



Ειδικός Σχεδιασμού Ιστοσελίδων & Εφαρμογών

Περιεχόμενα

ΠΕΡΙΓΡΑΦΗ.....	5
ΣΚΟΠΟΣ	6
ΜΑΘΗΣΙΑΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ	6
Εκπαιδευτική Ενότητα : 1 Εισαγωγή στην Πληροφορική	7
Σκοπός	7
Προσδοκώμενα Αποτέλεσμα	7
Έννοιες – Κλειδιά	7
Εκπαιδευτική Υποενότητα : 1.1 Δεδομένα και Υπολογιστές.....	8
Εκπαιδευτική Υποενότητα: 1.2 Ψηφιακή Τεχνολογία και οι Εφαρμογές της στο Κοινωνικό Γίνεσθαι. Κοινωνικές Επιπτώσεις από τις εξελίξεις της Πληροφορικής	19
Εκπαιδευτική Υποενότητα : 1.3 Υλικό και Λογισμικό Ηλεκτρονικών Υπολογιστών	83
Εκπαιδευτική Υποενότητα : 1.4 Η ψηφιακή τεχνολογίας και οι επιπτώσεις της	93
Σύνοψη Ενότητας.....	103
Ενότητα 1 ερωτήσεις	103
Άσκηση	104
Βιβλιογραφία.....	105
Προσδοκώμενα Ενότητα : 2 Υλικό και Δίκτυα Ηλεκτρονικών Υπολογιστών ..	105
Σκοπός	105
Μαθησιακά Αποτελέσματα	105
Έννοιες - Κλειδιά	105
Εκπαιδευτική Υποενότητα : 2.1 Δεδομένα – Πληροφορίες	107
Εκπαιδευτική Υποενότητα : 2.2 Δομικά στοιχεία ηλεκτρονικών υπολογιστών	119
Η RAM (Random Access Memory.....	123
Η ROM (Read-Only Memory, Μνήμη Μόνο Ανάγνωσης)	123
Εκπαιδευτική Υποενότητα : 2.3 Είδη δικτύων – Διαδίκτυο	132
Ο Παγκόσμιος Ιστός.....	159
Εκπαιδευτική Υποενότητα : 2.4 Δομικά στοιχεία δικτύων	164
Εκπαιδευτική Υποενότητα : 2.5 Θέματα Ασφάλειας στην πληροφορική	184
Αδυναμίες (vulnerabilities).....	201
Σύνοψη Ενότητας.....	233
Ενότητα 2 ερωτήσεις	233

Άσκηση	235
Βιβλιογραφία.....	235
Εκπαιδευτική Ενότητα : 3 Εισαγωγή στον προγραμματισμό	235
Σκοπός	235
Προσδοκώμενα Αποτελέσματα	236
Έννοιες – Κλειδιά	236
Εκπαιδευτική Υποενότητα: 3.1 Εισαγωγικές έννοιες	238
Εκπαιδευτική Υποενότητα: 3.2 Αλγόριθμοι και δομές δεδομένων	262
Αλγόριθμος Instagram.....	287
Εκπαιδευτική Υποενότητα: 3.3 Δομικά στοιχεία προγράμματος	293
Εκπαιδευτική Υποενότητα :3.4 Σχεδιασμός και υλοποίηση προγράμματος (Ψευδοκώδικας Pascal)	297
Σύνοψη	311
Ερωτήσεις ενότητας 3.....	312
Άσκηση	314
Βιβλιογραφία.....	314
Εκπαιδευτική Υποενότητα: 4 Λειτουργικά συστήματα και πληροφοριακά συστήματα	314
Σκοπός	315
Προσδοκώμενα Αποτελέσματα	315
Έννοιες – Κλειδιά	315
Εκπαιδευτική Υποενότητα : 4.1 Εισαγωγικές έννοιες.....	317
Εκπαιδευτική Υποενότητα : 4.2 Λειτουργικά συστήματα	323
Εκπαιδευτική Υποενότητα : 4.3 Λειτουργικά συστήματα πολλών χρηστών..	332
Εκπαιδευτική Υποενότητα : 4.4 Πληροφοριακά Συστήματα	338
Σύνοψη	380
Ερωτήσεις ενότητας 4.....	380
Άσκηση	382
Βιβλιογραφία.....	383
Εκπαιδευτική ενότητα: 5.Θεωρία βάσεων δεδομένων	383
Σκοπός	383
Προσδοκώμενα Αποτελέσματα	384
Έννοιες-κλειδιά	384
Εκπαιδευτική υποενότητα : 5.1 Μέθοδοι οργάνωσης δεδομένων	385
Φυσική ασφάλεια:.....	397
Λογική ασφάλεια:.....	397

Ακεραιότητα:	397
Εκπαιδευτική υποενότητα : 5.2 Μοντέλα Βάσεων Δεδομένων	404
Εκπαιδευτική υποενότητα : 5.3 Σχεδιασμός Μοντέλου Οντοτήτων-Συσχετίσεων	431
Εκπαιδευτική υποενότητα : 5.4 Συστήματα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ)	439
<i>Ποια είναι η σχέση μεταξύ Αυτόνομων Βάσεων Δεδομένων και Μηχανικής ...</i>	466
Εκπαιδευτική υποενότητα : 5.5 Εισαγωγή στη Δομημένη Γλώσσα Ερωταπαντήσεων (SQL)	478
Σύνοψη	500
Ερωτήσεις ενότητας 5	500
Άσκηση	502
Βιβλιογραφία	502
Εκπαιδευτική Υποενότητα : 6 Σχεδιασμός και ανάπτυξη ιστοτόπων	503
Σκοπός	503
Προσδοκώμενα αποτελέσματα	503
Έννοιες-κλειδιά	503
Εκπαιδευτική υποενότητα : 6.1 Παγκόσμιος Ιστός	505
Εκπαιδευτική Ενότητα: 6.2 Ανάπτυξη ιστοσελίδων (HTML, CSS)	520
Εκπαιδευτική Ενότητα: 6.3 Ανάπτυξη αλληλεπιδραστικών ιστοσελίδων (Javascript)	618
Εκπαιδευτική: Ενότητα 6.4 Ανάπτυξη δυναμικών ιστοσελίδων (PHP, MySQL)	656
Σύνοψη	670
Ερωτήσεις	671
Άσκηση	673
Βιβλιογραφία	673
Εκπαιδευτική Ενότητα 7. Σχεδιασμός και ανάπτυξη διαδικτυακών εφαρμογών	673
Σκοπός	673
Προσδοκώμενα αποτελέσματα	673
Έννοιες- κλειδιά	673
Εκπαιδευτική Υποενότητα 7.1 Περιβάλλον ανάπτυξης και αρχιτεκτονική μιας εφαρμογής	675
Εκπαιδευτική Υποενότητα 7.2 Κύκλος ανάπτυξης ολοκληρωμένης εφαρμογής	687
Εκπαιδευτική Υποενότητα 7.3 Εισαγωγή στον αντικειμενοστραφή προγραμματισμό (JAVA)	705

Εκπαιδευτική Υποενότητα 7.4 Εισαγωγή στην ανάπτυξη εφαρμογών για Android.....	736
Σύνοψη.....	787
Ερωτήσεις.....	787
Άσκηση.....	789
Βιβλιογραφία.....	789
Εκπαιδευτική ενότητα: 8 Διασύνδεση εφαρμογών με βάσεις δεδομένων...	790
Σκοπός.....	790
Προσδοκώμενα αποτελέσματα.....	790
Έννοιες-κλειδιά.....	790
Εκπαιδευτική υποενότητα : 8.1 Τεχνολογίες διαχείρισης δεδομένων.....	792
Εκπαιδευτική υποενότητα : 8.2 Διασύνδεση εφαρμογών με σχεσιακή βάση δεδομένων.....	886
Εκπαιδευτική Υποένότητα : 8.3 Διασύνδεση διαδικτυακών εφαρμογών με σχεσιακή βάση δεδομένων (PHP).....	915
Σύνοψη.....	979
Ερώτηση 8 ενότητας.....	979
Άσκηση.....	981
Βιβλιογραφία.....	981
Απαντήσεις.....	982

ΠΕΡΙΓΡΑΦΗ

Ο κόσμος των υπολογιστών και η πληροφορική έχει μπει πλέον στην καθημερινότητά μας και θα συνεχίσει πιο δραστικά και ουσιαστικά να διαφοροποιεί τη ζωή μας.

Το εκπαιδευτικό υλικό εισάγει τον εκπαιδευόμενο σε αυτό τον εξελισσόμενο κόσμο. Κατανοεί το τι είναι οι υπολογιστές, μαθαίνει για την γλωσσά τους την δομή τους. Μαθαίνει για τα δεδομένα το διαδίκτυο και όλες τις εφαρμογές και εργασίες που γίνονται μέσω των υπολογιστών.

Επίσης, το παρόν υλικό παρουσιάζει αναλυτικά τις διαδικασίες και τις τεχνικές σχεδιασμού ιστοσελίδων όχι μόνο θεωρητικά αλλά και πρακτικά με ασκήσεις και εφαρμογές.

Η τεχνολογία εξελίσσεται και το παρόν γνωστικό αντικείμενο είναι το θεμέλιο για κάποιον που θέλει όχι μόνο να παρακολουθεί τις εξελίξεις αλλά και να είναι μέσα σε αυτές.

ΣΚΟΠΟΣ

Σκοπός της εκπαίδευσης είναι η οικοδόμηση βήμα – βήμα, της τέχνης και της τεχνικής γύρω από τους υπολογιστές αλλά και τον σχεδιασμό τους, αναλύοντας την ιδιαιτερότητα της τεχνολογίας σαν μια νέα γλώσσα που έχει εισβάλει στην καθημερινότητα μας και προσπαθώντας να κοινοποιήσει αλλά και να γνωστοποιήσει τις δυνατότητές της στον σύγχρονο τρόπο ζωής μας.

Επιπλέον, διεισδύει στη μελέτη, στο σχεδιασμό και την κατασκευή των ιστοσελίδων, δίνοντας την βάση και την αρχή σε θέματα θεμελιώδους σημασίας σχετικά με την τεχνολογία και την αρχιτεκτονική αυτού του τομέα, παρέχοντας μια λεπτομερή ανάλυση για θέματα αιχμής.

ΜΑΘΗΣΙΑΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

Με την ολοκλήρωση της εκπαίδευσης, ο συμμετέχων μπορεί να εφαρμόζει τις απαραίτητες τεχνικές για την έρευνα, την συλλογή, την αποθήκευση, την επεξεργασία και την διανομή πληροφοριών, με τη βοήθεια υπολογιστικών συστημάτων.

Επιπλέον, θα μπορεί να εφαρμόζει τις μεθόδους σχεδιασμού των ιστοσελίδων όχι μόνο από την τεχνική πλευρά, αλλά και από την εικαστική, συμμετέχοντας στη δημιουργία ενός σωστού, εικαστικά και αισθητικά, αποτελέσματος.

Το τελικό αποτέλεσμα είναι να αποκτηθούν οι απαραίτητες γνώσεις ώστε ο εκπαιδευόμενος να γίνει ένας αποτελεσματικός συντονιστής στον τομέα της διαχείρισης και χρήσης των υπολογιστών και του σχεδιασμού ιστοσελίδων.

Εκπαιδευτική Ενότητα : 1 Εισαγωγή στην Πληροφορική

Σκοπός

Είναι να μάθει ο εκπαιδευόμενος για την δομή, το υλικό και το λογισμικό ενός υπολογιστή. Κατανοώντας φυσικά το τι είναι η πληροφορική, θα μπορέσει να χειριστεί τις δυνατότητες της, γνωρίζοντας την σαν νέα μια επιστήμη.

Έτσι ο εκπαιδευόμενος αποκτά εξοικείωση ,χτίζοντας τις βάσης του σε σχέση με την ψηφιακή τεχνολογία και τις εφαρμογές της.

Προσδοκώμενα Αποτέλεσμα

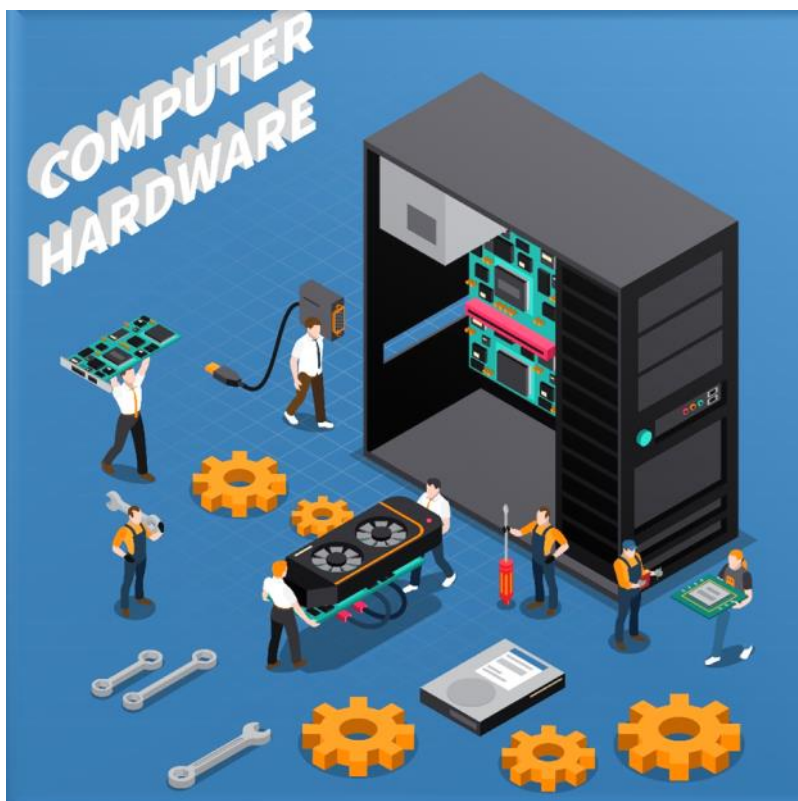
Ο εκπαιδευόμενος θα γνωρίσει και θα καταλαβαίνει τη διαδικασία επεξεργασίας δεδομένων. Θα περιγράφει αλλά και θα γνωρίζει τα μέρη που απαρτίζουν ένα υπολογιστικό σύστημα, γνωρίζοντας φυσικά και την χρήση τους. Τέλος θα μπορεί να αξιολογήσει και τις επιδράσεις της ψηφιακής τεχνολογίας στο σύγχρονο κοινωνικό, οικονομικό περιβάλλον, μπορώντας να στοχεύσει σε μια μελλοντικές της εξελίξεις.

Έννοιες – Κλειδιά

Hardware: Το υλικό μέρος του υπολογιστή, αυτό που βλέπουμε και αγγίζουμε είναι τα εξαρτήματα ενός υπολογιστή.

Software ή λογισμικό: Όλα τα προγράμματα, διαδικασίες και οδηγίες χρήσης του ηλεκτρονικού υπολογιστή που εκτελούν ορισμένες εργασίες.

Ψηφιακή τεχνολογία: Η ψηφιακή τεχνολογία αναμεταδίδει δεδομένα διαφόρων τύπων με δυαδικά ψηφία



Εκπαιδευτική Υποενότητα : 1.1 Δεδομένα και Υπολογιστές

Ο υλικός εξοπλισμός (hardware) του ηλεκτρονικού υπολογιστή αποτελείται από ένα σύνολο κατηγοριοποιημένων συσκευών, όπως:

- Συσκευές Εισόδου: Πληκτρολόγιο, Ποντίκι, Χειριστήριο (Joystick), Μικρόφωνο, Web camera, Σαρωτής (Scanner)
- Κεντρική Μονάδα Μητρική κάρτα, Κ.Μ.Ε. (Κεντρική Μονάδα Επεξεργασίας), ή πιο απλά Επεξεργαστής, Μνήμη RAM, Σκληρός Δίσκος (HDD=Hard Disk Drive), DVD / CD Drive, Floppy Disk Drive (οδηγός δισκέτας), Κάρτα γραφικών, Κάρτα ήχου, Κάρτα Δικτύου, ROM (όπου είναι αποθηκευμένο το BIOS), Μπαταρία, Τροφοδοτικό, κ.ά.
- Συσκευές Εξόδου: Οθόνη, Ηχεία, Εκτυπωτής.

Αυτός ο υλικός εξοπλισμός του ηλεκτρονικού υπολογιστή καθοδηγείται κατά τη λειτουργία του από το λογισμικό (software) που περιλαμβάνει:

- ✓ Το λογισμικό εφαρμογών, δηλαδή, οι επεξεργαστές κειμένου.
- ✓ Το ενδιάμεσο λογισμικό - middleware, που είναι ο συντονιστής και αυτό που ελέγχει τα καταναμημένα συστήματα.



- ✓ Το υλικολογισμικό που προγραμματίζει το υλικό ενός υπολογιστή ή τα περιφερειακά του.

Το Λογισμικό είναι κωδικοποιημένο με τη βοήθεια ενός δυαδικού ψηφιακού συστήματος, ώστε να είναι κατανοητό από το υλικό.

Περιλαμβάνει τα προγράμματα, τους ισότοπους, τα βιντεοπαιχνίδια και άλλα προγράμματα που έχουν αναπτυχθεί σε μια γλώσσα προγραμματισμού όπως για παράδειγμα η C, η C++ ή η Java.



HARDWARE

SOFTWARE

Λειτουργικό Σύστημα (Operating System), είναι το σύνολο των προγραμμάτων ενός υπολογιστικού συστήματος το οποίο ελέγχει, διαχειρίζεται και συντονίζει τη λειτουργία των επιμέρους τμημάτων του.

Τα προγράμματα του Λειτουργικού Συστήματος ελέγχουν τη λειτουργία των προγραμμάτων των χρηστών, παρέχοντας τους απαιτούμενους πόρους (resources) και εποπτεύοντας την αλληλοδιαδοχή τους.

Το σημαντικότερο πρόγραμμα του Λειτουργικού Συστήματος είναι ο επόπτης (supervisor) που διευθύνει το Λειτουργικό Σύστημα, και παραμένει μόνιμα στη μνήμη όσο ο υπολογιστής βρίσκεται σε λειτουργία.

Ο επόπτης ελέγχει το Λειτουργικό σύστημα και προωθεί στη μνήμη τα απαιτούμενα προγράμματα του Λειτουργικού για όλες τις επεξεργασίες.



Το Λειτουργικό Σύστημα βελτιστοποιεί την απόδοση της λειτουργίας του υπολογιστή και ενεργεί ως διασύνδεση (interface) χρήστη - μηχανής.

Τα υπόλοιπα προγράμματα του Λειτουργικού Συστήματος δε βρίσκονται συνέχεια στην κύρια μνήμη αλλά είναι αποθηκευμένα στη βοηθητική μνήμη.

Με τον τρόπο αυτό δημιουργείται ένα περιβάλλον συνεργασίας ανθρώπου και υπολογιστή, όπου ο προγραμματιστής μπορεί να δημιουργεί και να καλεί τα προγράμματα των εφαρμογών του, και ο υπολογιστής να τα εκτελεί.

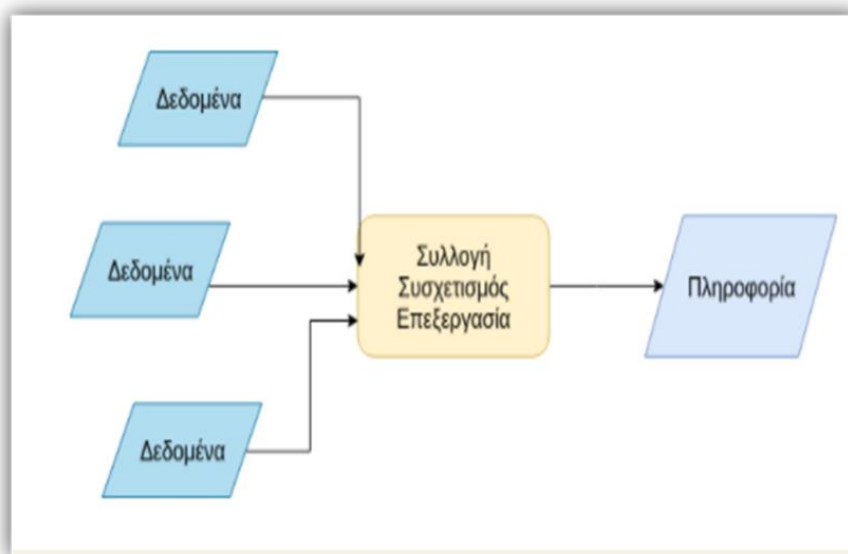
Δεδομένα Data:

Τα καθημερινά θέματα που μας απασχολούν αλλά και οι δραστηριότητες μας, μας αναγκάζουν να αναζητήσουμε πληροφοριακό υλικό, δηλαδή δεδομένα (Data).

Τα δεδομένα μπορεί να είναι σημεία πληροφοριών επί επιστημονικών παρατηρήσεων, ή συμπεριφοράς, να περιλαμβάνουν λέξεις, έννοιες, αριθμούς, σύμβολα, διαγράμματα, σχέδια, φωτογραφίες, κ.λπ. που περιγράφουν ή αντιπροσωπεύουν ποσότητες, έννοιες, ιδέες, αντικείμενα, γεγονότα, καταστάσεις και λειτουργίες.

Τα Δεδομένα μπορούν να μεταδοθούν ή να δεχτούν επεξεργασία.

Φυσικά δεν περιλαμβάνουν μόνο χρήσιμα αλλά και άσχετα ή περιττά στοιχεία για το υπό εξέταση.



Τα δεδομένα ως πληροφορία, μπορούμε να αποθηκεύσουμε σε βάσεις δεδομένων.

Τα αποτελέσματα που παίρνουμε, τα οποία μας μεταδίδουν κάποια επιπρόσθετη γνώση, τα χαρακτηρίζουμε ως Πληροφορίες (Information).

Η επεξεργασία των Data έχει διάφορες μορφές.

Οι επεξεργασίες στην συνέχεια αποτελούν δεδομένα σε μία νέα επεξεργασία.

Αυτή η εναλλαγή εξεργασίας από την μια πληροφορία στην άλλη, ονομάζεται Κύκλος Επεξεργασίας των Δεδομένων.

Στις μέρες μας για την επεξεργασία δεδομένων έχουμε τους υπολογιστές και τα Data είναι τα στοιχεία που χρησιμοποιούμε για επεξεργασία.

Αναφορά στην Επεξεργασία Δεδομένων (Data Processing)

Όλες λοιπόν οι εργασίες που γίνονται και αφορούν, τη συλλογή, την ταξινόμηση, την καταχώρηση, τη μεταβολή, την αποθήκευση, την αναζήτηση και την ανάκτηση δεδομένων με ή χωρίς τη βοήθεια του ηλεκτρονικού υπολογιστή για την παραγωγή πληροφοριών, ονομάζετε επεξεργασία δεδομένων.

Τα Big Data είναι τόσο ογκώδη δεδομένα που το παραδοσιακό λογισμικό επεξεργασίας δεν μπορεί να τα διαχειριστεί.



Από την επεξεργασία των δεδομένων τα αποτελέσματα που παίρνουμε και μας μεταδίδουν κάποια επιπρόσθετη γνώση ονομάζονται πληροφορίες, άρα:

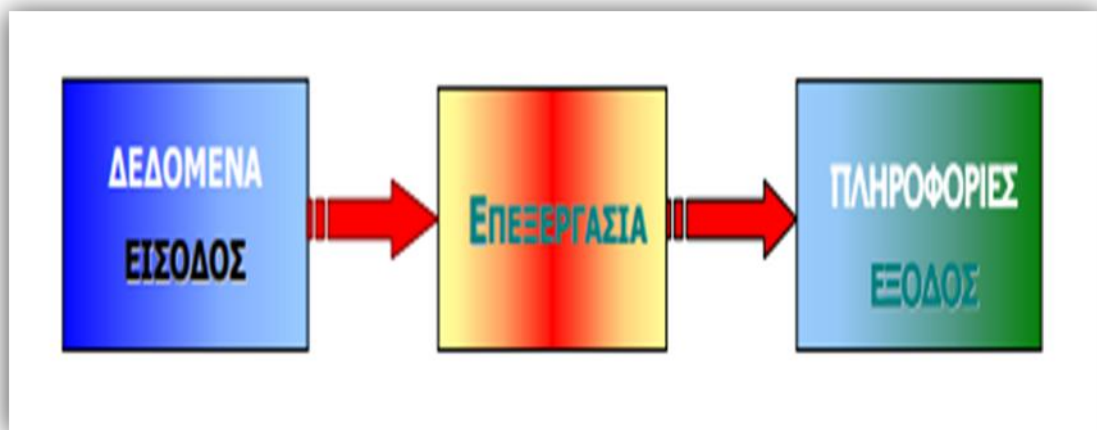
Η Πληροφορία Information είναι τα δεδομένα που έχουν δεχτεί τις αλλαγές και τις διορθώσεις, με λίγα λόγια έχουν γίνει οι επεξεργασίες για να γίνουν χρήσιμα για τον χρήστη δίνοντας τις απαντήσεις σε ερωτήσεις της μορφής, ποιος (who), τι (what), πού (where) και πότε (when).

Οι πληροφορίες που παράγονται πρέπει να έχουν τα χαρακτηριστικά που θα εξασφαλίζουν την ποιότητα τους όπως: Ακρίβεια, Σαφήνεια, Πληρότητα, Αξιοπιστία, Επικαιρότητα.

Η επεξεργασία των δεδομένων είναι απαραίτητη, ουσιαστικά αποτελεί την αφορμή και το κίνητρο για την δημιουργία των ηλεκτρονικών υπολογιστών που παρέχουν πια την δυνατότητα για ταχύτερη επεξεργασία τους με αποτέλεσμα για την εξαγωγή χρήσιμων πληροφοριών.

Αυτή η μονάδα αποθήκευσης δεδομένων στους υπολογιστές καλείται **δυναμικό ψηφίο ή μπιτ bit** (αποτελεί τη συντομογραφία του *Binary digit* συμβολίζεται και ως *b*). αλλά κυριολεκτικά σημαίνει και **«μικροσκοπικό τμήμα, κομματάκι».**

Στην γλώσσα μας αποδόθηκε με το όνομα **δυφίο**, που προέκυψε από τη σύντμηση των λέξεων **Δυναμικό ψηΦίο.**



Όμως σαν όρος δεν έχει χρησιμοποιηθεί καθόλου στη βιβλιογραφία των υπολογιστών.

Με το δυναμικό σύστημα αρίθμησης εργάζονται οι υπολογιστές, και χρησιμοποιούν δυναμικά ψηφία για να συμβολίζουν εντολές και δεδομένα.

Ως καταστάσεις ηλεκτρικής αγωγιμότητας ενός τρανζίστορ υλοποιούνται οι τιμές 0 και 1.

Συνήθως της υπολογιστές χρησιμοποιούμε τα **bit**, ομάδες των **8, 16, 32, 64, 128** κ.λπ., δηλαδή **δυνάμεις του 2**.

Το σύνολο των δυαδικών ψηφίων είναι κάθε χαρακτήρας, δηλαδή, είναι συνηθώς 8 και χαρακτηρίζουν της διαφορετικούς συνδυασμούς που χρειάζονται για τη σωστή και διαφορετική κωδικοποίηση.

8 δυαδικών ψηφίων – μπιτς (bits) κάθε χαρακτήρα του υπολογιστή αντιστοιχεί σε ένα **μπάιτ (byte)**.

Υπάρχουν πολλά πολλαπλάσια του Bit της Kbit, Mbit, Gbit, Tbit, Pbit κ.λπ.

W	O	R	D
01010111	01001111	01010010	01000100
1 BYTE			

Κωδικοποίηση της λέξης WORD στον υπολογιστή.

Ένα δυαδικό ψηφίο μπορεί να οριστεί ως μια μεταβλητή, ή ως μια υπολογίσιμη ποσότητα η οποία μπορεί να έχει μόνο δύο πιθανές τιμές και αναπαρίστανται με τους αριθμούς 0 και 1 συνήθως ερμηνεύονται ως **δυαδικά ψηφία**. Γι' αυτό, το σύστημα αποθήκευσης ονομάζεται Δυαδικό Σύστημα.

Έτσι φτιάχνονται ακολουθίες, ψηφίων **bits**, που ορίζουν τις πληροφορίες που αποθηκεύει ο υπολογιστής.

Άρα ένας υπολογιστής εκτελεί μόνο πράξεις αριθμητικές με βάση το δυαδικό σύστημα, επειδή ταιριάζει ακριβώς στις δυο καταστάσεις που μπορούν να λάβουν τα βασικά στοιχεία που τον απαρτίζουν.

Ενώ λοιπόν εμείς βλέπουμε ένα υπολογιστή να δημιουργήσει, να αντιγράψει, να συγκρίνει, να εκτελέσει διάφορες λειτουργίες, εσωτερικά όλο το λειτουργικό του, με βάση κάποια κωδικοποίηση μπορεί και χειρίζεται όλες αυτές τις πληροφορίες. Αρά κάθε εντολή που ο χρήστης του δίνει στον υπολογιστή, αυτός την μεταφράζει με την βοήθεια ενός κώδικα.

Τετράδα στο δυαδικό	Δεκαδικός αριθμός
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

Κωδικοποίηση Codification

Η κωδικοποίηση γίνεται για να αντιστοιχήσει σε κάθε χαρακτήρα έναν συγκεκριμένο αριθμό, ώστε να μπορεί ο υπολογιστής να κατανοήσει και το χειρίζεται ανάλογα με τις εντολές που δέχεται.

Γνωστός τέτοιος κώδικας είναι ο (American Standard Code for Information Interchange) ASCII.

Κωδικοποίηση ονομάζουμε την διαδικασία κατά την οποία κάθε χαρακτήρας αντιστοιχίζεται σε μια διαφορετική ομάδα δυαδικών ψηφίων. Ο υπολογιστής, για να μπορέσει να επεξεργαστεί, ή να διαχειριστεί ένα δεδομένο, μια εντολή, θα πρέπει

να έχουν δημιουργηθεί στην δική του γλώσσα ανάγνωσης και κατανόησης, ώστε να είναι αναγνωρίσιμα από αυτόν.



Ας δούμε ένα παράδειγμα:

Εάν θέλουμε ένας υπολογιστής να μεταδώσει την πληροφορία «ο λαμπτήρας ανάβει», θα μπορούσαμε να τη συμβολίσουμε με την κατάσταση «περνά ρεύμα υψηλής τάσης από έναν αγωγό».

Αντίστοιχα, εάν θέλουμε να μεταδώσει την πληροφορία «ο λαμπτήρας δεν ανάβει», θα μπορούσαμε να τη συμβολίσουμε με την κατάσταση «περνά ρεύμα χαμηλής τάσης από έναν αγωγό».

Όμως εάν θέλουμε να μεταδώσουμε και την πληροφορία «ο λαμπτήρας είναι χαλασμένος», δεν υπάρχει άλλη κατάσταση για να συμβολίσουμε την περίπτωση αυτή, εάν διαθέτουμε έναν μόνο αγωγό.

Άρα θα χρειαστούμε περισσότερους «αγωγούς» ταυτόχρονα. Συμπέρασμα, αν έχουμε δύο «αγωγούς πληροφορίας», μπορούμε να αναπαραστήσουμε και να συμβολίσουμε έως τέσσερις (4) διαφορετικές πληροφορίες (00, 01, 10, 11) μέσω των δύο διακριτών καταστάσεων.

Ο ρόλος της μνήμης

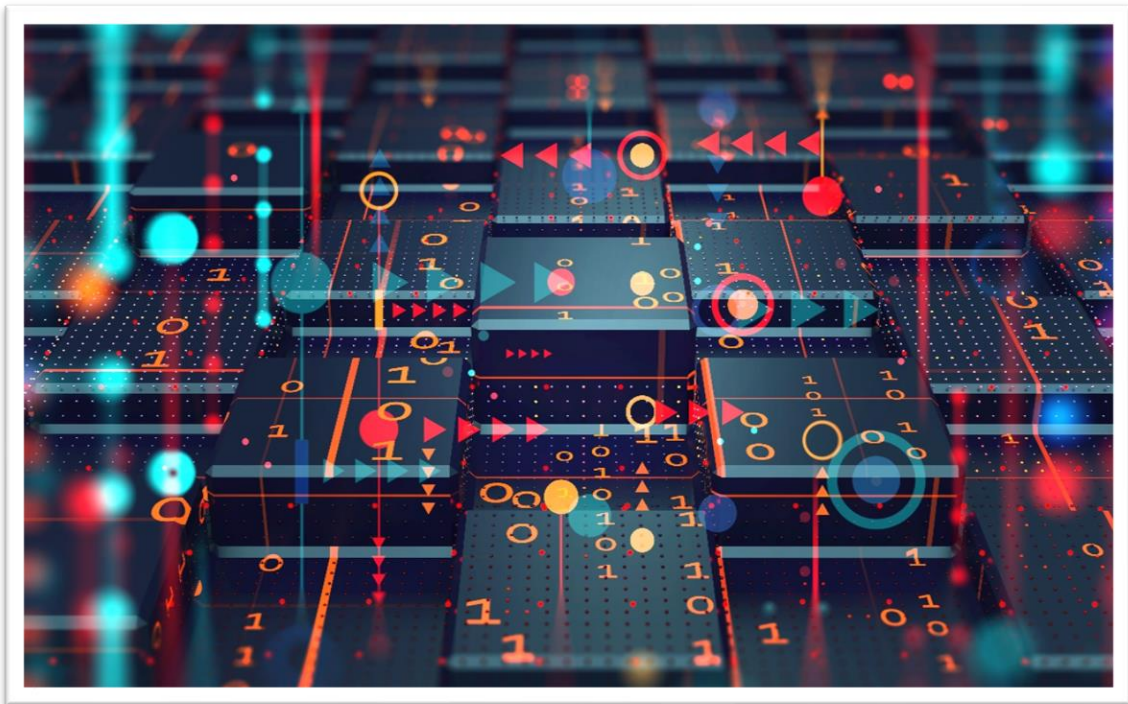
Η μνήμη είναι ένα βασικό εργαλείο για τον άνθρωπο, εκεί «αποθηκεύει» τις αναμνήσεις του πληροφορίες, τις γνώσεις του κτλ. Αποθηκεύει για να μπορεί ανά πασα ώρα και στιγμή να ανατρέξει σε αυτές και να τις χρησιμοποιήσει, να τις συσχετίσει με άλλες πληροφορίες από το εξωτερικό του περιβάλλον για να μπορεί να βγάλει κάποια συμπεράσματα.

Όπως και ο άνθρωπος έτσι και ο υπολογιστής έχει την μνήμη, το πιο βασικό εργαλείο του, εκεί αποθηκεύει καταγράφει πληροφορίες και στοιχεία που του δίνουμε και επεξεργάζεται κάθε στιγμή ότι νέα πληροφόρηση, στοιχείο, η εντολή που του δίνουμε.

Φυσικά σε αντίθεση με τον άνθρωπο ο υπολογιστής μπορεί μερικά από αυτά τα στοιχεία τα αποθηκεύει στην προσωρινή του μνήμη (RAM).

(Η συντομογραφία RAM σημαίνει Μνήμη Τυχαίας Προσπέλασης -Random Access Memory-Η μνήμη RAM είναι η βασική βραχυπρόθεσμη μνήμη.)

Κάποια άλλα στη μόνιμη μνήμη του (Περιφερειακή μνήμη, δίσκος, δισκέτα, CD).



Μονάδα Μέτρησης

Μονάδα μέτρησης στα υπολογιστικά συστήματα είναι το **bit** και **Byte**.

Σαν το πιο μικρότερο στοιχείο που μπορεί να επεξεργαστεί ένας υπολογιστής και που μπορεί να πάρει δύο μόνο τιμές, είναι το δυαδικό ψηφίο **Bit (Binary digit)** που το συμβολίζουμε με 0 και 1. π.χ. 00101110.

Ένα χαρακτήρα, δηλαδή, ένα γράμμα, ένας αριθμός ή κάποιο άλλο σύμβολο και αποτελείται από οκτώ (8) δυαδικά ψηφία (Bits), π.χ. 00101110.

Συνήθως ένα **Byte** αντιπροσωπεύει **256bit.10bytes** θα μπορούσε να είναι μια λέξη.

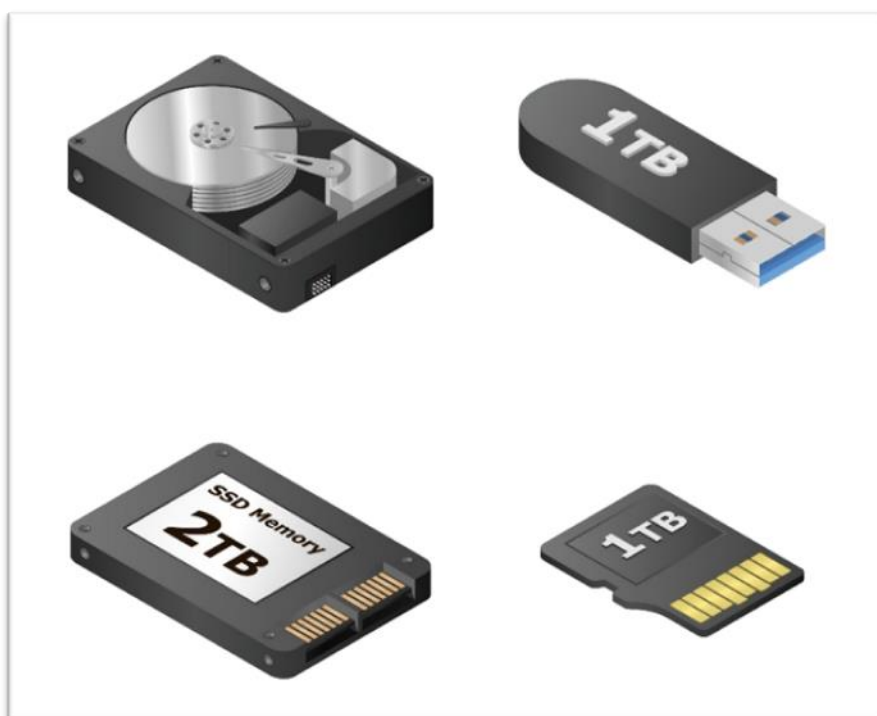
Η μονάδα μέτρησης της μνήμης που είναι το Byte, έχει πολλαπλάσια που είναι τα εξής:

Το **Kilobyte (Kb)**: αποτελείται από 1024 ή 210 bytes.

Το **Megabyte (Mb)**: αποτελείται από 1024 Kbytes ή 220 bytes (1.048.576bytes).

Το **Gigabyte (Gb)**: αποτελείται από 1024 Mbytes ή 220 Kbytes ή 230 bytes (1.073.741.824 bytes).

Το **Terabyte**: είναι ένα τρισεκατομμύρια bytes, ή 1.000 Gigabyte



- ✓ Kilobyte (Kb)= 1.000 Bytes. 1 Kilobyte θα μπορούσε να είναι ίσο με μια παράγραφο, 100 Kilobyte μπορεί να ισοδύναμη με μια σελίδα.
- ✓ Megabyte (Mb)=Στις μέρες μας που ένας σκληρός δίσκος είναι 1Terabyte σε ένα υπολογιστή το ένα Megabyte δεν είναι κάτι ιδιαίτερο. 100 Megabyte θα μπορούσαν να αποθηκεύσουν ένα όγκο από δυο εγκυκλοπαίδειες, ενώ 600 Megabyte είναι κοντά στην ποσότητα των δεδομένων που αποθηκεύτε σε ένα δίσκο CD-ROM.
- ✓ Gigabyte=είναι σχεδόν διπλάσιο από ποσό των δεδομένων που ένα CD-ROM μπορεί να αποθηκεύσει ή μπορεί να αποθηκεύσει βιβλία που έχει ένα ράφι βιβλιοθήκης 10 μέτρων.
- ✓ Terabyte= 2 Terabyte είναι οι συνηθισμένες προδιαγραφές στις μέρες μας για την χωρητικότητα ενός δίσκου για παράδειγμα ένα Terabyte θα μπορούσε να χωρέσει 1.000 αντίγραφα της εγκυκλοπαίδειας BRITANNICA.

Εκπαιδευτική Υποενότητα: 1.2 Ψηφιακή Τεχνολογία και οι Εφαρμογές της στο Κοινωνικό Γίγνεσθαι. Κοινωνικές Επιπτώσεις από τις εξελίξεις της Πληροφορικής

Από το 1945 που παρουσιάστηκε και χρησιμοποιήθηκε ο πρώτος υπολογιστής, μέχρι σήμερα η επιστήμη της Πληροφορικής έχει κάνει αλματώδη πρόοδο.

Η εξάπλωση και η χρήση υπολογιστών αλλά και «έξυπνων συσκευών», καθώς και η ανάπτυξη και καθιέρωση του Διαδικτύου έχει επηρεάσει και αλλάζει πολλούς τομείς της κοινωνίας, θετικά όσο και αρνητικά



Οι Τεχνολογίες Πληροφορικής και Επικοινωνιών είναι άμεσα συνδεδεμένες με την καθημερινότητα, δεν είναι απλά εργαλεία που χρησιμοποιούμε στις καθημερινές εργασίες μας, καθώς έχουν φέρει σημαντικές αλλαγές στη ζωή μας, αλλά και στον



τρόπο που εργαζόμαστε, επικοινωνούμε, διασκεδάζουμε και στις κοινωνικές συναναστροφές και όχι μόνο αφού έχουν αλλάξει μέχρι και τον τρόπο εκπαίδευσης μας κ.λπ.

Προσφέροντας όχι μόνο νέες ευκαιρίες, αλλά εισάγουν και νέους κινδύνους, όπως η μαζική παραβίαση της ιδιωτικότητας του ατόμου.

Παράλληλα, οι Τεχνολογίες Πληροφορικής και Επικοινωνιών υποστηρίζουν βασικούς θεσμούς και υποδομές στις σύγχρονες κοινωνίες, όπως οι θεσμοί διακυβέρνησης, η οικονομία, οι υποδομές των δικτύων ενέργειας, των συγκοινωνιών κ.ά.

Θα εστιάσουμε λοιπόν σε τρεις τύπους κοινωνικών επιπτώσεων που έχουν οι Τεχνολογίες Πληροφορικής και Επικοινωνιών. Τις επιπτώσεις στην εργασιακή ζωή των ανθρώπων, τις επιπτώσεις στην προσωπική και κοινωνική ζωή και τις επιπτώσεις στο επίπεδο των θεσμών και της κοινωνίας.

Εφαρμογές ψηφιακής τεχνολογίας

Η Ψηφιακή Τεχνολογία αφορά τις εφαρμογές, στην πληροφορική, στο διαδίκτυο και τους ηλεκτρονικούς υπολογιστές, τόσο σε επίπεδο υλικού (hardware) όσο και σε επίπεδο βασικού προγραμματισμού (software).

Στις μέρες μας οι Η/Υ είναι ένα αναπόσπαστο κομμάτι στην καθημερινότητα μας.

Αφού έχουν πάρει όχι άδικα, μια σημαντική θέση στη ζωή μας, βρίσκοντας εφαρμογή σε πολλούς τομείς, στην προσωπική και επαγγελματική μας ζωή καθημερινά.

Αποτελούν πλέον εργαλείο διερεύνησης, ενημέρωσης, δημιουργίας, επικοινωνίας, ψυχαγωγίας.

Αξιοποιώντας έτσι τις πολλαπλές δυνατότητες τους, μας προσφέρουν ευκαιρίες σε πολλές εφαρμογές:

- Εφαρμογές της πληροφορικής στις Επικοινωνίες
- Εφαρμογές της πληροφορικής στη Βιομηχανία
- Εφαρμογές της πληροφορικής στις Επιχειρήσεις
- Εφαρμογές της πληροφορικής στις Υπηρεσίες
- Εφαρμογές της πληροφορικής στην Υγεία
- Εφαρμογές της πληροφορικής στην Εκπαίδευση
- Εφαρμογές της πληροφορικής στις Επιστήμες και την Έρευνα
- Εφαρμογές της πληροφορικής στις Τέχνες
- Εφαρμογές της πληροφορικής στην Ψυχαγωγία

Οι υπηρεσίες του Internet

World Wide Web (παγκόσμιος ιστός), *e-mail* (ηλεκτρονικό ταχυδρομείο) και *newsgroups* (ομάδες ειδήσεων), *FTP* (πρωτόκολλο μεταφοράς αρχείων) *Telnet* (διαχείριση απομακρυσμένου υπολογιστή), τηλεδιάσκεψη είναι μερικές από τις υπηρεσίες που παρέχουν στους χρήστες διάφορες εφαρμογές που αξιοποιούν την τεχνολογία του διαδικτύου (Internet).



Όμως και στην τηλεπικοινωνιακή υποδομή, με τη χρήση υπολογιστών έχει εξελιχθεί καθώς με την εισαγωγή τους στα ψηφιακά τηλέφωνα παρέχουν πλέον πολλές δυνατότητες στους χρήστες τους π.χ. η ταυτόχρονη συνομιλία με δύο και με τρία άτομα καθώς και η εκτροπή κλήσεων και ένα σωρό άλλες δυνατότητες.

Το δημοκρατικό δίκτυο είναι το μέσο μαζικής επικοινωνίας που καθιστά τον άνθρωπο ισχυρότερο δίνοντάς του την δυνατότητα για πρόσβαση σε ένα μεγάλο όγκο πληροφοριών καθημερινά και επιτρέποντάς του να επιλέγει προσωπικά τις πληροφορίες που αυτός έχει ανάγκη.

Παραδείγμα, ο χρήστης ενός υπολογιστή μπορεί με πρόσβαση στο διαδίκτυο:

- Να αναζητήσει πληροφορίες, να φέρει στον υπολογιστή του κείμενα, ήχους και video
- Να δημοσιεύσει υλικό και να επικοινωνήσει με κάποιο άλλο χρήστη (e-mail)
- Να αγοράσει ή να πουλήσει προϊόντα ή να εκτελέσει τραπεζικές συναλλαγές
- Να διασκεδάση παρακολουθώντας ραδιοφωνικό ή τηλεοπτικό πρόγραμμα

Εφαρμογές της πληροφορικής στις Επικοινωνίες

Την δεκαετία του 1990, που χαρακτηρίστηκε από ένα πλήθος τεχνολογικών εξελίξεων που ακολούθησαν τους ιλιγγιώδεις ρυθμούς ανάπτυξης παγκοσμίως αυξήθηκαν οι επενδύσεις, σε Τεχνολογίες Πληροφορικής και Επικοινωνιών. Ακόμη και παραδοσιακοί κλάδοι μπορούν να ωφεληθούν και να βελτιωθούν εφαρμόζοντας αυτή την καινοτομική συμπεριφορά των εξελίξεων.



Οι Τεχνολογίες Πληροφορικής και Επικοινωνιών (Τ.Π.Ε.) είναι τεχνολογίες υποδομής που επηρεάζουν τις οικονομικές και κοινωνικές δραστηριότητες μιας χώρας και αφορούν όλους τους κλάδους της αλλά και την καθημερινή ζωή των ανθρώπων με

εφαρμογές όπως εκπαίδευση, εργασία και αγορές από απόσταση, παιχνίδια και ψυχαγωγία υπολογιστή, τηλεφωνική ηχητική κλήση ή βιντεοκλήση μέσω του Διαδικτύου, ηλεκτρονικό ταχυδρομείο, τηλεϊατρική, ψηφιακές βιβλιοθήκες, ηλεκτρονικό χρήμα, ηλεκτρονικό εμπόριο για γενικότερα κάθε αναζήτηση και άμεση ανάκτηση πληροφοριών κάθε είδους.

Η σχεδόν γενικευμένη παγκόσμια χρήση του δικτύου αυξάνει δραματικά τον βαθμό συνεργασίας μεταξύ εταίρων, συχνά γεωγραφικά απομακρυσμένων, και επιβάλλει νέα πρότυπα, νέες δυνατότητες αλλά και νέα προβλήματα για πολλές δραστηριότητες.

Οι τεχνολογίες πληροφορικής και επικοινωνιών προσφέρουν το τεχνολογικό υπόβαθρο και επηρεάζουν έτσι όλες τις επιχειρήσεις, ανεξάρτητα από μέγεθος, καθώς προσφέρουν τη δυνατότητα πρόσβασης σε πληροφόρηση, αγορές και πόρους με χαμηλό κόστος, πρόσβαση.

Κάτι που στο παρελθόν θεωρείτο προνόμιο των μεγάλων επιχειρήσεων.

Εφαρμογές της πληροφορικής στην Υγεία

Μια από τις επιστήμες που η εξέλιξη της συνδέεται άμεσα με την πληροφορική είναι η Ιατρική.

Οι εφαρμογές πληροφορικής που συναντώνται στην ιατρική είναι, οι χημικές αναλύσεις, το υπερηχογράφημα, τα ιατρικά μηχανήματα υψηλής τεχνολογίας αξονικός και μαγνητικός τομογράφος.

Υπάρχουν διαθέσιμες πολλές εφαρμογές για την παρακολούθηση της πορείας της υγείας των ασθενών και την τήρηση ιστορικού ακόμα και αν δεν βρίσκονται στον ίδιο χώρο με το γιατρό τους.

Η τηλεϊατρική είναι εφαρμογή ιδιαίτερα χρήσιμη στους κατοίκους απομακρυσμένων περιοχών.

Τα εργαλεία ηλεκτρονικής υγείας από απόσταση, μπορούν να βοηθήσουν τους ανθρώπους να έχουν πρόσβαση σε δυνητικά σωτήριες ιατρικές πληροφορίες όπου και όταν τις χρειάζονται, χωρίς να απαιτείται η μετακίνηση τους



Δραστική επιρροή δέχεται ο χώρος της κοινωνικής πολιτικής και ειδικότερα ο τομέας της κοινωνικής φροντίδας. Σημαντικός για χώρες όπως η Ελλάδα που εμφανίζουν δημογραφικά σημεία γήρανσης του πληθυσμού.

Έχει αναδιαμορφωθεί ο χάρτης κοινωνικών υπηρεσιών με ανάπτυξη νέων δομών αλλά και την αναβάθμιση και εκσυγχρονισμό των ήδη υφιστάμενων ώστε να γίνουν πιο αποδοτικές και να μπορούν να διαχειριστούν τις απαιτήσεις των πολιτών για καλύτερη ποιότητα ζωής ενώ συμβάλουν καθοριστικά στην αποφυγή του κοινωνικού αποκλεισμού ευπαθών ομάδων.

Η εξ αποστάσεως παροχή ιατρικών συμβουλών, με έμφαση στην προληπτική Ιατρική, διευκολύνει επίσης τη φροντίδα των κατοίκων απομακρυσμένων περιοχών.

Η ηλεκτρονική υγεία προάγει την ανθρωποκεντρική διαχείριση των ασθενειών και την εξατομικευμένη ιατρική περίθαλψη σε συνδυασμό με την ασφάλεια των ασθενών.



Επεκτείνετε ο θεσμός της κατ' οίκον φροντίδας που ανταποκρίνεται πλέον στους στόχους της Ευρωπαϊκής σύμπραξης, για μια ενεργή σωστά προστατευμένη και υγιή γήρανση του πληθυσμού.

Τεχνολογία φροντίδας στο σπίτι είναι κάθε τεχνολογικό μέσο, ιατρικό ή όχι, που παρέχει, ή διευκολύνει τη φροντίδα και τις καθημερινές δραστηριότητες στην οικία του χρήστη.

Οι υπηρεσίες της κατ' οίκον φροντίδας παρέχονται στο χώρο του ασθενή σύμφωνα με ιδιαίτερες ανάγκες και φροντίδες που χρειάζεται αλλά και την κατάσταση υγείας που είναι.

Οι χρήστες θα αποφύγουν την μετακίνηση (επίσκεψη σε γραφείο ιατρού, σε κλινική ή νοσοκομείο) προκειμένου να λάβουν την απαιτούμενη περίθαλψη, ειδικά για άτομα με χρόνιες παθήσεις.



Η εισαγωγή της ηλεκτρονικής υγείας στην πρωτοβάθμια ιατρική φροντίδα στοχεύει στη χρήση των τεχνολογικών καλύτερη πρόληψη, τη διάγνωση, τη θεραπεία, την παρακολούθηση και τη διαχείριση της υγείας του πληθυσμού.

Διαφορετικοί οργανισμοί υγειονομικής περίθαλψης, όπως ιδιωτικά νοσοκομεία, κλινικές, κατοικίες φροντίδας και οργανώσεις κοινωνικής πρόνοιας θα μπορούν να εξασφαλίσουν γρηγορά, αλλά και αξιόπιστα, άμεσα προσβάσιμη ιατρική πληροφόρηση για όλους.

Οι εφαρμογές της ηλεκτρονικής υγείας συμβάλλουν στην ακριβή διάγνωση με την βοήθεια των νέων συστημάτων που βασίζουν τη λειτουργία τους στην απεικόνιση. Διευρύνεται λοιπόν ο χώρος της πρωτοβάθμιας φροντίδας, με την ευκολία στην χρήση, τη διαθεσιμότητα, τις δυνατότητες ελέγχου και μείωσης του κόστους, καθώς και τη βελτίωση της εξυπηρέτησης των ασθενών-χρηστών.

Η ηλεκτρονική υγεία μπορεί να ωφελήσει τα συστήματα υγείας με επίκεντρο τον πολίτη και τη συνολική αποδοτικότητα, αποτελεσματικότητα και βιωσιμότητα του κλάδου υγείας.

Η ηλεκτρονική υγεία περιλαμβάνει οργανωμένα δίκτυα πληροφοριών υγείας, ηλεκτρονικά αρχεία υγείας, ηλεκτρονικός ιατρικός φάκελος, ηλεκτρονική κάρτα υγείας, ηλεκτρονική συνταγογράφηση, υπηρεσίες τηλεϊατρικής,

τηλεσυμβουλευτικής και τηλεπαρακολούθησης, και φορητά συστήματα επικοινωνίας για παρακολούθηση και υποστήριξη ασθενών.

Τα ηλεκτρονικά αρχεία υγείας μπορούν επίσης να χρησιμοποιηθούν για μη ιατρικούς σκοπούς, όπως η (ιατρική) βελτίωση της ποιότητας, η αναφορά αποτελεσμάτων, η (ιατρική) διαχείριση πόρων και η δημόσια υγεία.

Τα Ηλεκτρονικά Μητρώα Υγείας (EHR) ενσωματώνουν πληροφορίες για την υγεία των ασθενών από διαφορετικές πηγές, συμπεριλαμβανομένων όλων των ηλεκτρονικών ιατρικών αρχείων του ασθενούς, στα οποία μπορεί να έχει πρόσβαση ο πάροχος υγειονομικής περίθαλψης του ασθενούς από διαφορετικές τοποθεσίες. Ιδανικά, τα ηλεκτρονικά αρχεία υγείας θα πρέπει να ενημερώνονται συνεχώς σε πραγματικό χρόνο.



Οι εφαρμογές πληροφορικής στα μηχανήματα, και τις ιατρικές συσκευές

Το περιβάλλον επίσης ενός νοσοκομείου βομβαρδίζεται κάθε στιγμή από τεράστιο όγκο δεδομένων ποικίλης φύσης.

Διοικητικοί, οικονομικοί, τεχνικοί, ιατρικοί, νοσηλευτικοί, παραϊατρικοί, εργαστηριακοί και άλλοι πόροι αποτελούν τις πηγές των πληροφοριών. Οι δε πληροφορίες ποικίλουν σε μορφή, όγκο και ανάγκες, π.χ. οι ακτινογραφίες έχουν τη μορφή εικόνων, η εξέταση αίματος τιμές, κ.λπ.

Ιατρικά δεδομένα για τον ασθενή συλλέγονται από χημικές αναλύσεις, το υπερηχογράφημα και τα ιατρικά μηχανήματα υψηλής τεχνολογίας, όπως ο αξονικός και μαγνητικός τομογράφος. Συνήθως στην κατασκευή ιατρικών συσκευών αυτά

εφοδιάζονται με διάφορους αισθητήρες, π.χ. αισθητήρας μέτρησης του διαβήτη από ασθενή, για τους οποίους υπάρχει η πρόβλεψη διασύνδεσης τους, με τα πληροφοριακά συστήματα υγείας σε ένα νοσοκομείο μέσω δικτύου. Πολλά από αυτά τα πληροφοριακά συστήματα υγείας, είναι κρίσιμα και ζωτικής σημασίας για την υγεία ενός ασθενούς και ο σχεδιασμός και η λειτουργία τους υπόκεινται σε αυστηρούς κανονισμούς ποιότητας και εποπτείας.



Ενδεικτικά οι εφαρμογές και τα πληροφοριακά συστήματα ενός νοσοκομείου μπορούν να χωριστούν σε τέσσερις μεγάλες κατηγορίες:

- Εφαρμογές ζωτικής σημασίας, τα οποία διαθέτουν διάφορα συστήματα παρακολούθησης ασθενών που βρίσκονται συνήθως σε κρίσιμη κατάσταση και εφαρμογές συναγερμών μηνυμάτων κ.λπ.
- Κατ' απαίτηση εφαρμογές πολυμέσων
- Διοικητικές και λογιστικές εφαρμογές
- Εφαρμογές απομακρυσμένου ελέγχου.

Εφαρμογές της πληροφορικής στη Βιομηχανία

Η βιομηχανία δεν θα μπορούσε να μείνει ανεπηρέαστη από την ανάπτυξη της τεχνολογίας και από την ραγδαία εξέλιξη των υπολογιστικών συστημάτων με την εισαγωγή και χρήση των οποίων, κατάφερε να αναβαθμιστεί, ώστε να μπορέσει να ανταπεξέλθει στις αυξημένες παραγωγικές απαιτήσεις της εποχής. Βιομηχανική

πληροφορική ο κλάδος που ασχολείται με εφαρμογές της πληροφορικής στο χώρο της βιομηχανίας.

Οι προσπάθειες εκσυγχρονισμού στο χώρο των βιομηχανιών εστιάζονται σε τρεις τομείς:

- Στη διακίνηση και αξιοποίηση της πληροφορίας.
- Στο χρηματοοικονομικό τομέα με εφαρμογές που παρέχουν στις διοικήσεις οικονομικά στοιχεία που προκύπτουν από την λειτουργία των επιχειρήσεων τους, ώστε να μπορούν να λαμβάνουν αποφάσεις με γνώμονα την πραγματική απόδοση και τις δυνατότητες που έχουν, αλλά και να μπορούν να διαχειρίζονται αποτελεσματικά τα κόστη, ώστε να βελτιώνουν την κερδοφορία τους.
- Στον τομέα της παραγωγής η πληροφορική δημιούργησε αυτοματοποιήσεις



με την βοήθεια και χρήση εξελιγμένων και καινοτόμων συστημάτων για την βελτίωση της παρακολούθησης, της λειτουργίας και του ελέγχου της παραγωγικής διαδικασίας, με αποτέλεσμα την βελτίωση της ποιότητας και την μείωση του κόστους.

Τα Συστήματα παραγωγής με τη βοήθεια υπολογιστή (Computer Aided Manufacturing) χρησιμοποιούνται για την σχεδίαση και παραγωγή προϊόντων.

Εκεί αναλύονται οι τεχνικές προδιάγραφες τους, η αντοχή των υλικών κατασκευής τους, η αεροδυναμική συμπεριφορά τους, όλες οι τεχνικές τους προδιάγραφες. Σε αυτόματες γραμμές παραγωγής παράγονται προϊόντα με τη χρήση βιομηχανικών ρομπότ.

Οι νέες τεχνολογίες της πληροφορικής χαρακτηρίζονται από τις σημαντικά μειωμένες απαιτήσεις σε υλικό αλλά και τις συνεχώς αυξανόμενες απαιτήσεις σε λογισμικό.

Ψηφιακή επιχειρηματικότητα – Ψηφιακή αγορά

Για τους σύγχρονους επιχειρηματίες η χρήση του Διαδικτύου ήταν ένα μέσο για να δημιουργήσουν εμπορικές ευκαιρίες, να προωθήσουν την παραγωγή και τα προϊόντα τους και να επικοινωνήσουν και να συνεργαστούν μέσω του διαδικτύου με νέους πελάτες και συνεργάτες.



Η ψηφιακή επιχειρηματικότητα αξιοποιεί εφαρμογές τονώνοντας την επιχειρηματικότητα σε πιο ευρύ κύκλο καταναλωτών που δεν περιορίζονται από την απόσταση, ενώ οι συναλλαγές μπορούν να ολοκληρωθούν γρήγορα και η πληρωμή να είναι άμεσα διαθέσιμη χωρίς γραφειοκρατικές διαδικασίες τραπεζικών οργανισμών.

Στις περισσότερες μορφές ψηφιακής επιχειρηματικότητας συνδυάζονται τα φυσικά και τα ηλεκτρονικά καταστήματα. Υπάρχουν και μορφές επιχειρηματικότητας που

λειτουργούν μονάχα On-line, απαλλάσσοντας τον επιχειρηματία από μια σειρά λειτουργικών εξόδων (γραφεία, καταστήματα, πωλητές, εξοπλισμό κ.λπ.)

Το ψηφιακό εμπόριο

Το ψηφιακό εμπόριο είναι ένα σημαντικό κεφάλαιο της ψηφιακής επιχειρηματικότητας καθώς απευθύνεται σε όλο το εύρος των πολιτών και επισκεπτών μιας πόλης και όχι μόνο.

Οι επιχειρήσεις μιας τοπικής κοινωνίας που εστίαζαν στα προϊόντα της περιοχής τους δεν παραμένουν εγκλωβισμένες στην κλειστή κοινωνία τους και μέσα από τα σωστά σχεδιασμένα τους ηλεκτρονικά- ψηφιακά τους καταστήματα, έχουν ανοίξει το πελατολόγιο τους και σε άλλες αγορές εκτός της περιοχής τους.



Τα περισσότερα από τα ηλεκτρονικά καταστήματα χρησιμοποιούν μεγάλες διαδικτυακές πλατφόρμες συναλλαγών, e-shops, που συνδέουν άμεσα τους πελάτες, με τους προμηθευτές, επιτρέποντας τους να ενημερώνονται για τις τιμές των προϊόντων και να πραγματοποιούν ηλεκτρονικές αγορές.

Η πλατφόρμα βοηθά ώστε να γίνουν αγοραπωλησίες ειδών ή υπηρεσιών και να ολοκληρώνονται οι συναλλαγές μέσω διαφόρων τρόπων ηλεκτρονικής πληρωμής.

Πλεονεκτήματα του ηλεκτρονικού καταστήματος:

- ✓ Βολικό και γρήγορο. Οι διαδικασίες απλοποιούνται, για παράδειγμα τα φυσικά καταστήματα δεν χρειάζεται να διακοσμήσουν και να αλλάζουν βιτρίνα κάθε μήνα.
- ✓ Γρήγορες συναλλαγές. Αφού ο αγοραστής και ο πωλητής καταλήξουν σε συμφωνία, γίνεται ηλεκτρονικά η πληρωμή της συναλλαγής και τα αγαθά παραδίδονται στον αγοραστή μέσω της εφοδιαστικής ομάδας.
- ✓ Εύκολες αγορές όλη την διάρκεια της ημέρας. Χωρίς τους περιορισμούς του ωραρίου λειτουργίας, ο καταναλωτής μπορεί να κάνει τις αγορές του από όπου βρίσκεται με ένα κλικ από τον υπολογιστή ή το τηλέφωνο του.



Η τεράστια αύξηση του αριθμού των ηλεκτρονικών καταστημάτων τα έχει αναγκάσει, να μετατραπούν από μια απλή ηλεκτρονική προβολή προϊόντων, σε

μηχανές στρατηγικής μάρκετινγκ ,κάνοντας χρήση πολύπλοκων εργαλείων, όπως μηχανές CEO κ.λπ. για την εύκολη και ξεχωριστή προβολή τους μέσα στο διαδίκτυο. Ενώ αξιοποιούν κάθε εργαλείο του διαδικτύου και των μέσων κοινωνικής δικτύωσης για να είναι εύχρηστες και λειτουργικές οι ιστοσελίδες τους, χρησιμοποιούν φωτογραφίες, βίντεο και προσφορές χαμηλών τιμών για να προσελκύσουν πελάτες και χρησιμοποιούν μοντέλα και Influencers με αναγνωρισιμότητα για να αυξήσουν το κύρος των προϊόντων τους για να μπορέσουν να ξεχωρίσουν από τον τεράστιο αριθμό από ηλεκτρονικά καταστήματα που είναι ενεργά.

Το άνοιγμα ενός ηλεκτρονικού καταστήματος σημαίνει ότι είναι διαθέσιμος ένας ιστότοπος με λειτουργίες διαδικτυακών αγορών, δηλαδή ολόκληρη η βασική υποδομή που μπορεί να είναι ένας κεντρικός υπολογιστής, μια βάση δεδομένων και ένα πρόγραμμα λογισμικού.



Τριάντα χρόνια μετά την εμφάνιση των πρώτων πλατφορμών ηλεκτρονικού εμπορίου, σήμερα, το ηλεκτρονικό εμπόριο είναι μέρος της ανθρώπινης καθημερινότητας. Το μερίδιο ηλεκτρονικού εμπορίου στις συνολικές παγκόσμιες λιανικές πωλήσεις περιλαμβάνει το 14% όλων των εσόδων λιανικής το 2019, με τζίρο περίπου 3,5 τρις δολάρια από 2 δισεκατομμύρια παγκόσμιους ψηφιακούς αγοραστές, πριν αυτοί οι αριθμοί εκτοξευθούν από την κατακόρυφη άνοδο της

χρήσης του ηλεκτρονικού εμπορίου στην διάρκεια των περιοριστικών μέτρων που επιβλήθηκαν για την αντιμετώπιση της πανδημίας covid-19, το 2020 και το 2021.

Το ηλεκτρονικό εμπόριο είναι μια μέθοδος εμπορίου κατά την οποία είναι ψηφιακή ολόκληρη η διαδικασία της συναλλαγής, από την προσφορά του προϊόντος ή της υπηρεσίας έως την ολοκλήρωση της αγοράς με πληρωμή.

Στο ηλεκτρονικό εμπόριο, ο πελάτης αναζητά ένα προϊόν, έχει πρόσβαση σε πληροφορίες γι' αυτό, το επιλέγει, το τοποθετεί στο καλάθι αγορών του και ολοκληρώνει την αγορά του με την πληρωμή.

Σε αυτόν τον τύπο επιχείρησης, ολόκληρη η δομή είναι ψηφιοποιημένη και τα μόνα πράγματα που υπάρχουν στον φυσικό κόσμο είναι το απόθεμα όπου βρίσκεται το προϊόν και η διαδικασία παράδοσης στους αγοραστές.

Σε ορισμένες περιπτώσεις, ούτε καν αυτό, για παράδειγμα τα διαδικτυακά παιχνίδια αγοράζονται από το ηλεκτρονικό εμπόριο, και μεταφορτώνονται απευθείας στο βιντεοπαιχνίδι ή στον υπολογιστή, χωρίς να παραδοθεί φυσικό προϊόν.

Το ηλεκτρονικό εμπόριο λειτουργεί μέσω διαδικτυακών πλατφορμών πωλήσεων, οι οποίες μπορεί να είναι είτε μία μεμονωμένη εταιρεία (όπως συμβαίνει συχνά με μεγάλες εταιρείες λιανικής) είτε μικρότερες εταιρείες που χρησιμοποιούν τη δομή ενός ιστότοπου ή μιας εφαρμογής για να διαφημίσουν τα προϊόντα τους.



Στην πράξη, η αγορά λειτουργεί σαν ένα τεράστιο εμπορικό κέντρο στο οποίο πολλά διαφορετικά καταστήματα μοιράζονται τον ίδιο χώρο.

Το ηλεκτρονικό εμπόριο χρησιμεύει ως σημείο πώλησης ένα πλήρως ψηφιακό περιβάλλον και οι αγοραστές πραγματοποιούν τις αγορές τους εξ αποστάσεως, χωρίς να φύγουν από το σπίτι – για παράδειγμα μέσω υπολογιστή ή κινητού τηλεφώνου.

Το ηλεκτρονικό εμπόριο μπορεί να ωφελήσει τόσο τον αγοραστή όσο και τον πωλητή ενός προϊόντος. Πλέον το προϊόν, η διαδικασία πώλησης και ο μεσάζοντας είναι εξολοκλήρου ψηφιακοί.

Σε όσους αγοράζουν, το ηλεκτρονικό εμπόριο επιτρέπει σε κάθε χρονική στιγμή να επιλέξουν το προϊόν ήρεμα, αποκτούν πρόσβαση σε προϊόντα από άλλες περιοχές της χώρας (ή του κόσμου) και αφού συγκρίνουν τιμές και χαρακτηριστικά ανταγωνιστικών προϊόντων, μπορούν να ολοκληρώσουν την αγορά μέσω υπολογιστή ή κινητού τηλεφώνου, ανεξάρτητα από το πού βρίσκονται.



Για όσους πωλούν, το μεγάλο πλεονέκτημα είναι να εισάγουν την επιχείρησή τους στο ψηφιακό περιβάλλον και να προσεγγίζουν περισσότερους ανθρώπους.

Για παράδειγμα, εάν έχουν φυσικό κατάστημα, η επιχείρησή περιορίζεται σε άτομα που ζουν ή επισκέπτονται την περιοχή.

Με το ηλεκτρονικό εμπόριο, μπορεί να προσεγγίσουν άτομα σε άλλες πόλεις ή χώρες άλλες.

Για τους αγοραστές, τα κύρια μειονεκτήματα είναι ότι δεν γνωρίζουν προσωπικά το προϊόν και πρέπει να πληρώσουν το κόστος αποστολής για να το παραλάβουν.

Όποιος αγοράζει ένα προϊόν σε φυσικό κατάστημα μπορεί συνήθως να το αγγίξει, να το δοκιμάσει, να δει αν είναι πραγματικά αυτό που περίμενε, αν το υλικό είναι καλής ποιότητας κ.λπ. Όταν αγοράζει μέσω Διαδικτύου, δεν είναι δυνατό να γνωρίζει εάν το προϊόν είναι ακριβώς όπως φαντάζεται, ειδικά αν πρόκειται για αγορά ακριβών αντικειμένων π.χ. έπιπλα.

Γι' αυτό είναι σημαντικό για τους ψηφιακούς πωλητές να περιγράφουν το προϊόν με όσο το δυνατόν περισσότερες λεπτομέρειες. Όταν κάποιος είναι δυνητικός αγοραστής, η ανάγνωση της περιγραφής, η προβολή εικόνων αυτού του προϊόντος από διάφορες οπτικές γωνίες και η γνώση των διαστάσεων του μπορεί να είναι καθοριστικής σημασίας για να αποφασίσει αν θα αγοράσετε ή όχι από αυτό το ηλεκτρονικό κατάστημα.



Το κόστος μεταφοράς μπορεί επίσης να αποτελέσει εμπόδιο επειδή όταν αγοράζει σε ένα φυσικό κατάστημα, ο καταναλωτής βλέπει μια ενιαία τιμή, η οποία περιλαμβάνει όλα τα έξοδα.

Στο ηλεκτρονικό κατάστημα, η αποστολή χρεώνεται ξεχωριστά, συνήθως εξαρτάται από την απόσταση του σημείου βρίσκεται το εμπόρευμα και του τόπου παράδοσης του.

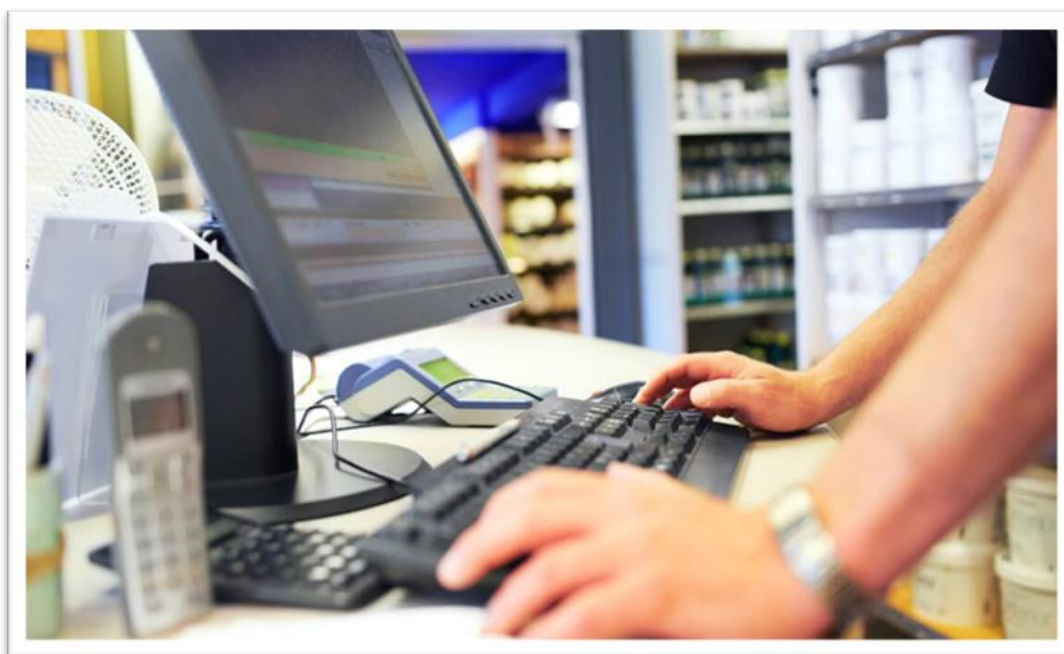
Αν το κόστος αποστολής είναι υψηλό, πολλές φορές αποθαρρύνει ένα άτομο από το να ολοκληρώσει μια αγορά.

Εκτός από το ηλεκτρονικό εμπόριο που απευθύνεται σε τελικούς καταναλωτές, οι οποίοι αγοράζουν προϊόντα και υπηρεσίες για προσωπική χρήση, υπάρχουν επίσης εταιρείες που πωλούν διαδικτυακά σε άλλες εταιρείες.

Αυτό ονομάζεται ηλεκτρονικό εμπόριο B2B (business to business). Μερικά παραδείγματα ηλεκτρονικού εμπορίου B2B είναι αυτά που πωλούν χονδρικά (όπως ρούχα για μεταπωλητές ή τρόφιμα για εστιατόρια και ζαχαροπλαστεία), καθώς και εταιρείες συμβούλων, ακόμη και εξοπλισμό για βιομηχανίες και γραφεία.

Εφαρμογές της πληροφορικής στις Επιχειρήσεις

Σε έναν κόσμο που ολόκληρο το επιχειρηματικό οικοσύστημα είναι πλέον διασυνδεδεμένο, οι εταιρείες πρέπει να είναι ευέλικτες ώστε να ανταποκρίνονται στις μεταβαλλόμενες επιχειρηματικές ανάγκες και απαιτήσεις της αγοράς. Ο ισχυρός ανταγωνισμός ανάγκασε τις επιχειρήσεις να επενδύσουν σε συστήματα και τεχνολογίες πληροφοριών για να επιτύχουν πιο αποτελεσματική διαχείριση των λειτουργιών τους.



Οι εφαρμογές της πληροφορικής στις επιχειρήσεις αναφέρονται συχνά σαν εφαρμογές αυτοματισμού γραφείου, δηλαδή, ένα πληροφοριακό σύστημα που υποστηρίζει με τη βοήθεια υπολογιστών τις εργασίες μιας επιχείρησης, για παράδειγμα, την παρακολούθηση στοιχείων για τις πωλήσεις, τις αγορές, τους

πελάτες και τους προμηθευτές, την ηλεκτρονική τήρηση των λογιστικών βιβλίων και τον υπολογισμό της μισθοδοσίας του προσωπικού.

Για παράδειγμα, στο ταμείο πολλών καταστημάτων υπάρχει μια οθόνη, ένα πληκτρολόγιο και ένας εκτυπωτής, στον οποίο τυπώνονται οι αποδείξεις. Ο ταμίας χειρίζεται επίσης μια συσκευή, με τη βοήθεια της οποίας «διαβάζει» τους κωδικούς των προϊόντων. Οι συσκευές αυτές (οθόνη, πληκτρολόγιο, εκτυπωτής, φορητή συσκευή) αποτελούν τμήματα του πληροφοριακού συστήματος της επιχείρησης.

Οι εφαρμογές αυτοματισμού γραφείου έχουν ως στόχο την αύξηση της αποδοτικότητας των εργαζομένων σε ένα γραφείο. Για παράδειγμα, στα σύγχρονα γραφεία οι γραφομηχανές έχουν αντικατασταθεί σε μεγάλο βαθμό από υπολογιστές και εκτυπωτές.

Οι εφαρμογές που χρησιμοποιούνται στην περίπτωση αυτή ονομάζονται συστήματα επεξεργασίας κειμένου (word processors) και αποτελούν την αντιπροσωπευτικότερη εφαρμογή αυτής της κατηγορίας.



Το πλήθος των εφαρμογών που έχουν αναπτυχθεί ως σήμερα έχει δημιουργήσει ένα αντίστοιχο πλήθος κατηγοριών. Κάποιες από τις κατηγορίες αυτές έχουν ευρεία χρήση, εμφανίζονται σχεδόν σε κάθε προσωπικό υπολογιστή, όπως είναι οι φυλλομετρητές, προγράμματα πλοήγησης στον Παγκόσμιο Ιστό (web browsers), τα

προγράμματα αναπαραγωγής πολυμέσων (media players), οι εφαρμογές γραφείου (office suites) και τα προγράμματα αντιμετώπισης κακόβουλου λογισμικού (antivirus).

Στις εφαρμογές γραφείου (Office Suites) περιλαμβάνονται εφαρμογές όπως:

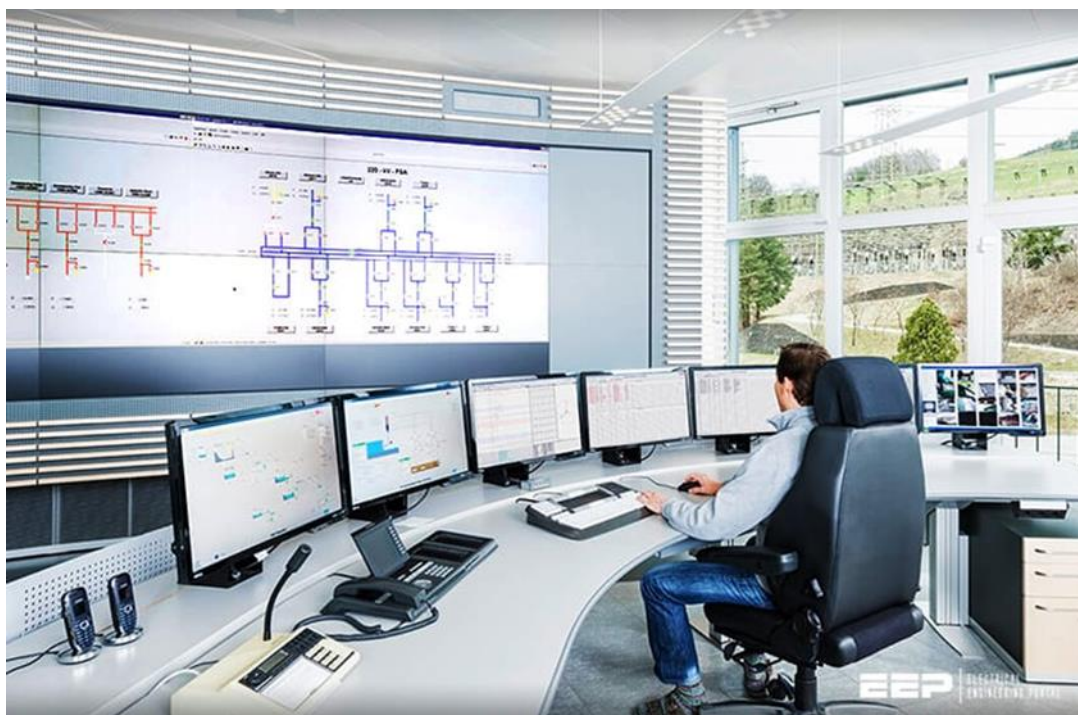
- ο επεξεργαστής κειμένου (word processor), που χρησιμοποιείται για τη σύνθεση, διόρθωση, μορφοποίηση και εκτύπωση εγγράφων.
- το υπολογιστικό φύλλο (spreadsheet), που διευκολύνει την οργάνωση αριθμητικών δεδομένων μέσω πινάκων, την αυτοματοποίηση πολύπλοκων υπολογισμών και τη δημιουργία γραφημάτων.
- το πρόγραμμα παρουσιάσεων, για την εύκολη δημιουργία εντυπωσιακών διαφανειών μιας παρουσίασης.
- το πρόγραμμα διαχείρισης βάσεων δεδομένων, που επιτρέπει την αποτελεσματική οργάνωση και διαχείριση της πληροφορίας.
- το πρόγραμμα ηλεκτρονικού ταχυδρομείου και οργάνωσης υποχρεώσεων.



Η χρήση των παραπάνω προγραμμάτων είναι τόσο διαδεδομένη, ώστε συνήθως τα προγράμματα αυτά δεν εμφανίζονται μεμονωμένα αλλά ως «πακέτο» (σουίτα). Οι πιο δημοφιλείς σουίτες είναι το Microsoft Office και το LibreOffice.

Στις επιχειρήσεις υπάρχουν τα συστήματα επιτραπέζιας τυπογραφίας (desktop publishing systems) και των ηλεκτρονικών εκδόσεων. Τα συστήματα επιτραπέζιας τυπογραφίας είναι εφαρμογές με τη χρήση των οποίων η διαδικασία της εκτύπωσης

απλοποιείται σημαντικά μειώνοντας το χρόνο και το κόστος εκτύπωσης. Πλέον οι ηλεκτρονικές εκδόσεις εκτός από κείμενο και εικόνα περιέχουν ήχο, video και κινούμενα γραφικά.



Η τεχνολογία σε μια επιχείρηση μπορεί να γίνει αντιληπτή σε δύο διαστάσεις:

- ✓ Τη **δημιουργική διάσταση** που αφορά το σχεδιασμό του συνολικού τεχνολογικού σχεδιασμού, που θα χρησιμοποιηθεί για να καλύψει όλες τις ανάγκες μιας επιχείρησης σε όλα τα στάδια των λειτουργιών της.
- ✓ Την **ουσιαστική διάσταση** που προσδιορίζει ακριβώς τα τεχνολογικά προϊόντα (hardware και software) που χρειάζονται για την εκτέλεση της κάθε συγκεκριμένης εργασίας.

Η τεχνολογία ασχολείται με τον μηχανολογικό εξοπλισμό που είναι υπεύθυνος για την παραγωγή, τα υλικά και την κατασκευή, την μορφοποίησή τους, τα logistics, τα πληροφοριακά συστήματα, το έλεγχο ποιότητας και τέλος τη συντήρηση των στοιχείων της παραγωγικής διαδικασίας.

Η τεχνολογία προϊόντος αποτελείται από:

- Τον προγραμματισμό προϊόντος το στάδιο που ελέγχονται οι ανάγκες, προσδιορίζονται και καλείται να τις καλύψει μελετώντας τα κατάλληλα χαρακτηριστικά του και το κόστος του.
- Την παραγωγή προϊόντος, τα στάδια από την έρευνα και τον σχεδιασμό του προϊόντος μέχρι τον σχεδιασμό της παραγωγής.
- Την εμπορία, με την προώθηση των πωλήσεων, την ικανοποίηση των απαιτήσεων του πελάτη και τρόπους επίδειξης της χρήσης του προϊόντος.
- Την εξυπηρέτηση των πελατών που περιλαμβάνει οδηγίες για τον τρόπο συντήρησης και λειτουργίας του προϊόντος.



Μάρκετινγκ και πληροφορική.

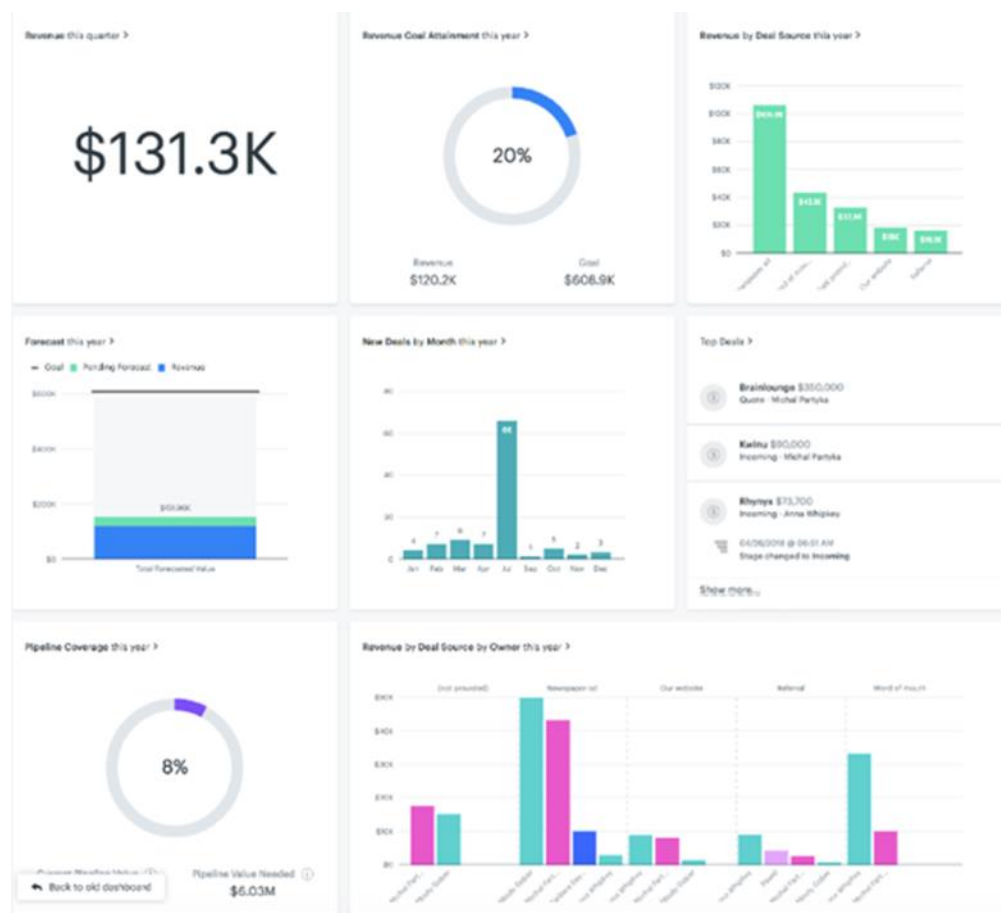
Οι υπολογιστές, το διαδίκτυο και τα μέσα κοινωνικής δικτύωσης αποτελούν αναπόσπαστο κομμάτι στην ανθρώπινη και έχουν αναδειχθεί σαν καθοριστικός τρόπος πραγματοποίησης καταναλωτικών επιλογών και αγορών προϊόντων.

Αυτό είχε σαν αποτέλεσμα την ανάπτυξη του ηλεκτρονικού μάρκετινγκ, το οποίο αποτελεί πλέον ένα εργαλείο που βρίσκεται στη διάθεση όλων των επιχειρήσεων, μεγαλύτερων ή μικρότερων.

Από την άποψη των επιχειρηματικών δραστηριοτήτων, τα συστήματα διαχείρισης πληροφοριών μάρκετινγκ και οι εφαρμογές επιχειρηματικού αυτοματισμού που διευκολύνει τη λήψη αποφάσεων, είναι προϊόν της σύγχρονης τεχνολογικής εξέλιξης.

Με τη βοήθεια της σύγχρονης τεχνολογίας πληροφοριών, οι δραστηριότητες μάρκετινγκ των επιχειρήσεων, μπορούν να ξεπεράσουν σε μεγάλο βαθμό, τους περιορισμούς του χρόνου και του χώρου του παρελθόν αξιοποιώντας τη

μικροηλεκτρονική τεχνολογία και την τεχνολογία επικοινωνίας. Το *μάρκετινγκ στην ηλεκτρονική αυτή μορφή* περιλαμβάνει την χρήση ηλεκτρονικών εφαρμογών για τον σχεδιασμό και την παράδοση των αγαθών και υπηρεσιών με τα χαρακτηριστικά που ικανοποιούν τις ανάγκες των καταναλωτών. Οι επιχειρήσεις το χρησιμοποιούν με στόχο να αυξήσουν την αποδοτικότητά τους και να βελτιώσουν την ποιότητα των προϊόντων και υπηρεσιών τους, με απώτερο σκοπό την μεγιστοποίηση των κερδών τους.



Το ηλεκτρονικό μάρκετινγκ παρέχει στις επιχειρήσεις άμεσες και έγκυρες πληροφορίες για την μελέτη του ανταγωνισμού και την έρευνα των καταναλωτών. Οι επιχειρήσεις όταν έχουν στη διάθεση τους τις σωστές πληροφορίες αγοράς, μπορούν να ικανοποιήσουν τις ανάγκες των πελατών με πιο κατάλληλα προϊόντα, καλύτερη ποιότητα, πιο ευνοϊκές τιμές, πιο αποτελεσματική διαφήμιση και πιο αποτελεσματικά κανάλια διανομής και να επεκτείνουν την παρουσία τους στην αγορά.

Το ηλεκτρονικό μάρκετινγκ αξιοποιείται στις δραστηριότητες:

- Της προβολής της επιχείρησης και των αγαθών.

- Της διανομής των αγαθών προς τους αγοραστές.
- Της τιμολόγησης των αγαθών και υπηρεσιών. Αυτό βέβαια αποσκοπεί στην αμοιβαία ικανοποίηση των επιχειρήσεων και των πελατών.

Ηλεκτρονική Διαφήμιση

Η διαδικτυακή διαφήμιση, ψηφιακή διαφήμιση ή διαφήμιση μέσω **web**, είναι μια μορφή μάρκετινγκ που χρησιμοποιεί το Διαδίκτυο για την παροχή διαφημιστικών μηνυμάτων στους καταναλωτές.

Η μορφή διαφήμισης στο διαδίκτυο, είναι μια εξελισσόμενη μορφή, που εξαρτάται από τις εξελίξεις της τεχνολογίας, άρα για να είναι αποτελεσματική απαιτεί από τις επιχειρήσεις διαρκή παρακολούθηση των εξελίξεων.



Ο πιθανός πελάτης στον οποίο στοχεύει το διαφημιστικό μήμα, *δεν είναι ένας παθητικός δέκτης μηνυμάτων, αλλά μέσα από τις εφαρμογές και τις δυνατότητες του διαδικτύου μπορεί να αντιδράσει στέλνοντας τα δικά του μηνύματα.*

Έτσι η στάση των καταναλωτών απέναντι σε μια διαδικτυακή διαφήμιση παρέχει χρήσιμα συμπεράσματα και πληροφορίες στην επιχείρηση.

Τα κυριότερα χαρακτηριστικά της διαδικτυακής διαφήμισης είναι:

- Δημιουργία άμεσης σχέσης με τον πελάτη, που ταυτίζεται με τη δημιουργία εξατομικευμένων διαφημίσεων, προσαρμοσμένες στα ιδιαίτερα χαρακτηριστικά, μεμονωμένων ατόμων, ή ομάδων κοινού.

- Δημιουργία πολύ ελκυστικού διαφημιστικού μηνύματος μέσω της ταυτόχρονης χρήσης εικόνας, ήχου, κειμένου, κίνησης και γραφικών.
- Μειωμένο κόστος, που αφορά τόσο τη μετάδοση της διαφήμισης, όσο και της επαναπροβολής ή τις ανανεώσεις του μηνύματος σε σύγκριση πάντα με τον απεριόριστο χρόνο προβολής.
- Απόλυτος έλεγχος του προϋπολογισμού της διαφημιστικής καμπάνιας, καθώς υπάρχει άμεση γνώση των αποδεκτών που είδαν την διαφήμιση.
- Ευελιξία πομπού και δέκτη. Ο πομπός έχει τη δυνατότητα για χρήση περισσότερων της μίας στρατηγικές μηνυμάτων, ενώ ο δέκτης ορίζει πότε θα δει και θα επεξεργαστεί το μεταδιδόμενο μήνυμα.



Τα στοιχεία περιεχομένου μιας διαδικτυακής διαφήμισης είναι:

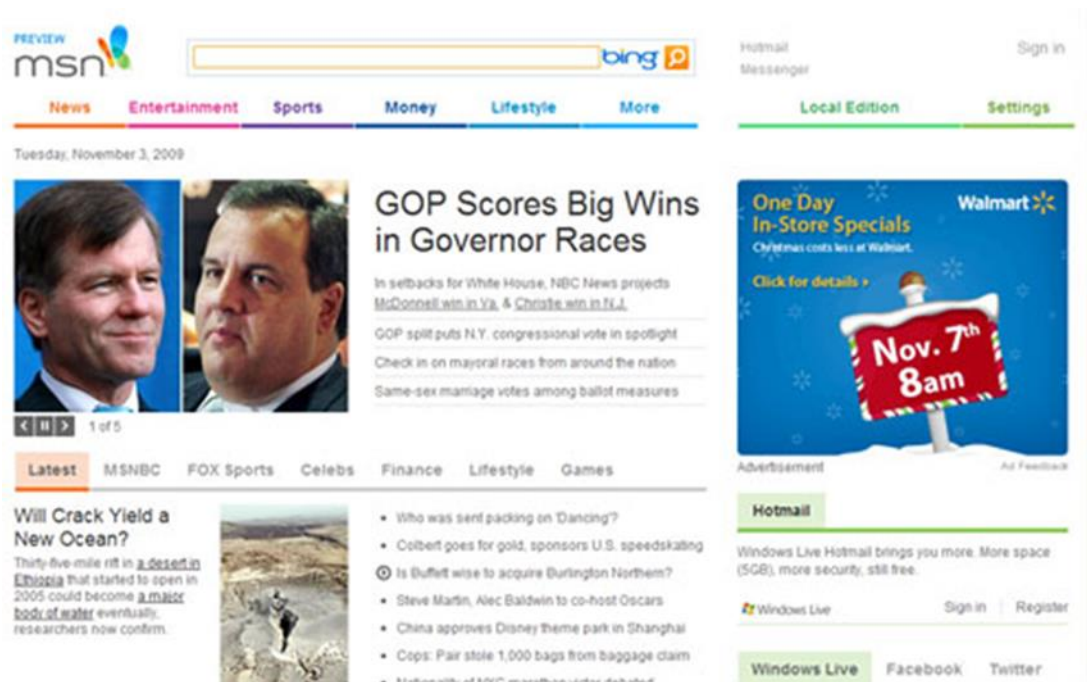
- η επωνυμία της επιχείρησης,
- η μάρκα, το λογότυπό της, κάποιο σλόγκαν αν υπάρχει,
- γραμμή πλοήγησης για την περιήγηση στις σελίδες του ιστότοπου που ο ενδιαφερόμενος θα δει τα προϊόντα και τις υπηρεσίες,
- ένα κείμενο με πληροφορίες, φωτογραφίες και βίντεο για τα προϊόντα.

Οι διαφημιζόμενοι (συνήθως μέσω των διακομιστών των διαφημίσεών τους) συχνά χρησιμοποιούν cookies.

Τα cookies είναι αναγνωριστικά συγκεκριμένων υπολογιστών, για να δουν ποιες διαφημίσεις θα εξυπηρετήσουν έναν συγκεκριμένο καταναλωτή.

Τα cookies μπορούν να δουν αν ένας χρήστης μπήκε και βγήκε από την σελίδα χωρίς να αγοράσει τίποτα, οπότε ο διαφημιζόμενος μπορεί αργότερα να τον προσεγγίσει πάλι, με διαφημίσεις από τον ιστότοπο που ήδη επισκέφθηκε. Η διαδικτυακή διαφήμιση εκτός από την προβολή της, στην επίσημη ιστοσελίδα της εταιρείας (Web Site), εμφανίζεται σε άλλες ιστοσελίδες με πολλούς διαφορετικούς τρόπους που καθένας έχει τα δικά του χαρακτηριστικά, εισάγεται με διαφορετικό τρόπο και έχει συγκεκριμένο σκοπό.

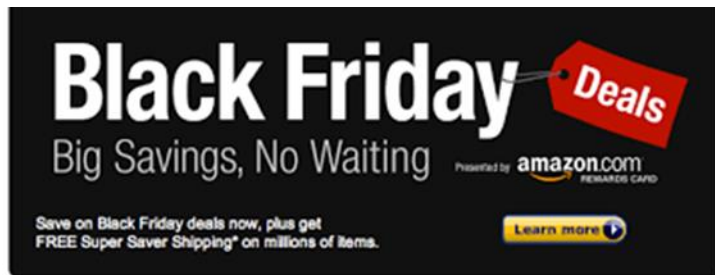
- **Banner.** Είναι η πρώτη και ίσως η πιο γνωστή και πιο κοινή μορφή διαδικτυακής διαφήμισης. Είναι ένας χώρος διαφορετικών μεγεθών και σχημάτων, τοποθετημένος σε στρατηγικό σημείο της ιστοσελίδας, που χρησιμοποιείται για την προβολή διαφημίσεων από άλλες εταιρείες ή άλλες ιστοσελίδες. Ο στόχος του είναι να μεταφέρει μια ιδέα και να κάνουν κλικ σε αυτήν οι ενδιαφερόμενοι πελάτες. Το banner είναι χρήσιμο σε μάρκες που έχουν καθιερωθεί στην αγορά, είναι γνωστές σε μεγάλο μέρος του κοινού και αρκεί η παρουσία και μόνο του λογότυπου τους για να πείσει τους χρήστες να αγοράσουν το προϊόν της.



- **Email.** Στο πλαίσιο της διαδικτυακής διαφήμισης, οι διαφημίσεις που αποστέλλονται σε email μπορεί να εξαφανιστούν στο πλαίσιο ανεπιθύμητων μηνυμάτων, να χαθούν, ή ο χρήστης να αποφασίσει να διαγράψει το μήνυμα χωρίς καν να το ανοίξει. Αυτό ανάγκασε πολλές εταιρείες να τελειοποιήσουν

την τεχνική και να σταματήσουν να στέλνουν διαφημίσεις, καμουφλάροντας τις προσφορές και τις διαφημίσεις τους μέσα σε καθημερινά και φιλικά μηνύματα που ο χρήστης δεν θα θέλει να διαγράψει.

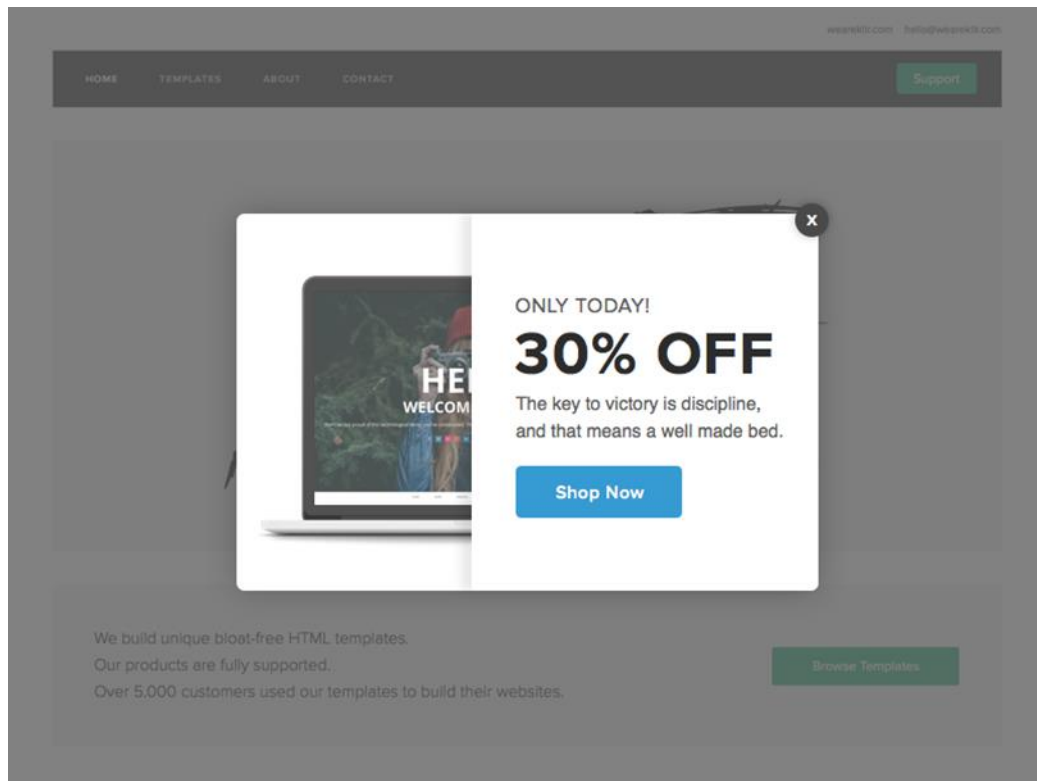
Χαρακτηριστικό παράδειγμα διαδικτυακής διαφήμισης μέσω email marketing είναι αυτό που συνήθως πραγματοποιεί η Amazon. Πάντα εκμεταλλεύεται τις ημέρες που υποδεικνύονται ως Black Friday για να λανσάρει τα προϊόντα της.



Έτσι οι πελάτες ταυτίζουν αυτόματα τις ημέρες κατανάλωσης με τη μάρκα, οπότε το είναι ήδη υψηλότερο το επίπεδο προσοχής και ενδιαφέροντος τους.

- Pop Up. Είναι ένα αναδυόμενο παράθυρο που εμφανίζεται όταν ο χρήστης ανοίγει έναν ιστότοπο, ιδιαίτερα ενοχλητικό επειδή μερικές φορές είναι δύσκολο να τον κλείσει. Ειδικά στις φορητές συσκευές λόγω του μεγέθους

της οθόνης, το κλείσιμο του αναδυόμενου παραθύρου είναι πολύ πιο περίπλοκο



- Διαφήμιση σε blogs είναι μια έμμεση διαδικτυακή διαφήμιση που χρησιμοποιεί την τεχνική αφήγησης, καλύπτοντάς την πίσω από μια εμπειρία, έναν προβληματισμό κ.λπ. Επειδή αυτοί οι ψηφιακοί χώροι έχουν γίνει δημοφιλείς, πολλές μάρκες εκμεταλλεύονται τις δυνατότητές τους για να μεταδώσουν τα μηνύματά τους ή να προωθήσουν τα προϊόντα τους κυρίως μέσω χορηγίας, αφού πολλοί blogger αποφασίζουν να προωθήσουν εκείνα τα προϊόντα που τους προσφέρουν οι εταιρείες για να διαφημίσουν.
- Διαφημιστικά κουμπιά (Buttons),
- Πλωτή διαφήμιση (floating),
- Δελτία τύπου (Advertorials),
- Μετάδοση από χρήστη σε χρήστη (Viral βίντεο),
- Παρουσία των επιχειρήσεων σε δημοφιλή Κοινωνικά Δίκτυα

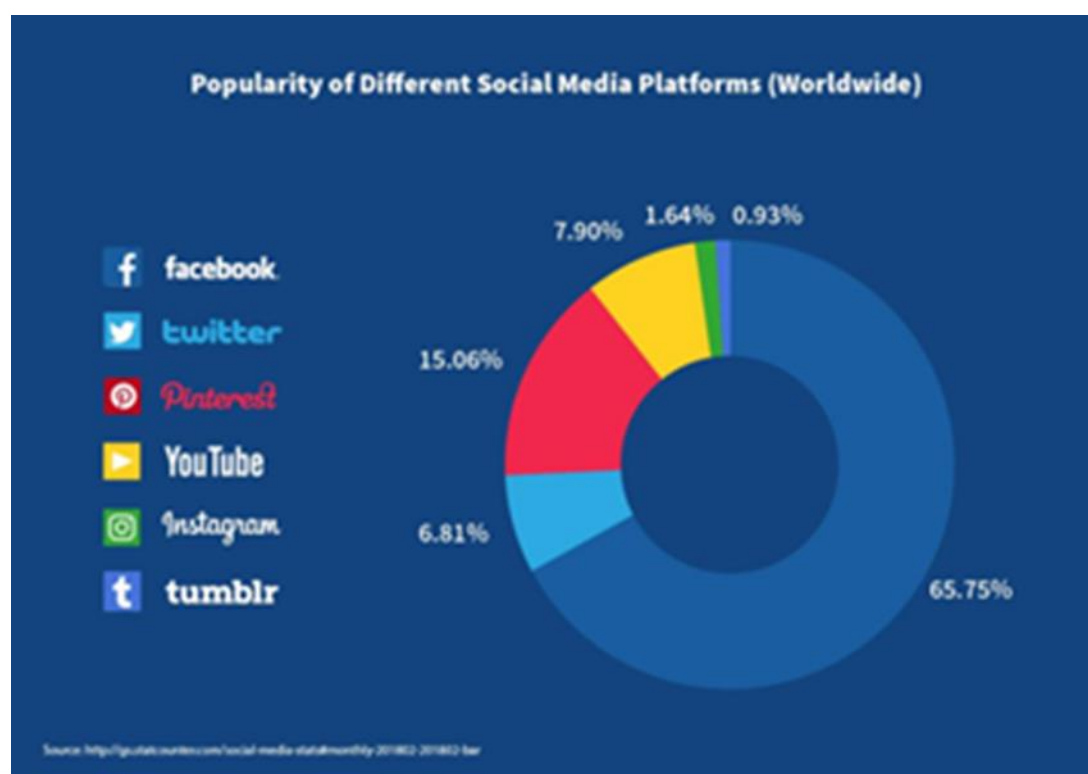
Μέσα κοινωνικής δικτύωσης (social media)

Τα μέσα κοινωνικής δικτύωσης (social media) είναι ψηφιακά μέσα αλληλεπίδρασης και επικοινωνίας ομάδων ανθρώπων μέσω διαδικτυακών κοινοτήτων. Τα μέσα

κοινωνικής δικτύωσης αποτελούν μια κοινωνική διάδραση μεταξύ ανθρώπων που δημιουργούν, μοιράζονται ή ανταλλάσσουν πληροφορίες και ιδέες μέσα σε εικονικές κοινότητες και δίκτυα και αποτελούν κυρίαρχο κομμάτι της καθημερινότητας των σύγχρονων ανθρώπων.

Τα social media σαν μορφή επικοινωνίας, έχει προσδώσει μία νέα ερμηνεία στην έννοια του προσωπικού χώρου και δημιουργούν ανησυχίες για παραβίαση της ιδιωτικότητας των χρηστών που έχουν δημοσιεύσει προσωπικά δεδομένα στο διαδίκτυο.

Το 2019 2,95 δισεκατομμύρια άνθρωποι ή το 77% των χρηστών του Διαδικτύου σε παγκόσμιο επίπεδο, χρησιμοποιούσαν κάποιο μέσο κοινωνικής δικτύωσης τουλάχιστον μία φορά τον μήνα.



Τα παγκόσμια έσοδα από διαφημίσεις κοινωνικών μέσων θα είναι περίπου 100 δισεκατομμύρια δολάρια το 2020, σύμφωνα με το Statista Digital Market Outlook. Από αυτό και μόνο φαίνεται ότι τα social media αποτελούν πλέον σημαντικό μέρος της ενεργειών προώθησης των επιχειρήσεων, επειδή προσφέρουν:

- ✓ Άμεση επαφή με τους καταναλωτές, η χρήση των εργαλείων κοινωνικής δικτύωσης, όπως το Facebook και το Twitter, ως πλατφόρμων συνεργασίας συνδέει τις επιχειρήσεις με τον υπόλοιπο κόσμο με δεκάδες διαφορετικούς

τρόπους. Τα εν λόγω εργαλεία φέρνουν την τεχνολογία σε επαφή με τις επιχειρήσεις, συνδέουν τους ανθρώπους με τις πληροφορίες, καθιερώνουν πιθανούς νέους δρόμους προς την αγορά και βελτιώνουν την επικοινωνία με τους πελάτες και τη διάδοση του εμπορικού σήματος.

- ✓ Επιχειρηματική Στρατηγική. Όσο αφορά στην επιχειρηματική στρατηγική, τα social media χρησιμοποιούνται ως μέσα δημιουργίας εταιρικής εικόνας, ενημέρωσης, επικοινωνίας και ανάπτυξης σχέσεων με τους πελάτες και έχουν απίστευτη δυναμική.
- ✓ Φήμη και Δημόσιες Σχέσεις. Οι εταιρείες που χρησιμοποιούν σωστά τα social networks μπορούν να «χτίσουν» την εικόνα τους, να αναπτύξουν τις δημόσιες σχέσεις τους καθώς και να δημιουργήσουν και ή να επηρεάσουν θετικά τις συζητήσεις που γίνονται γύρω από το brand τους «εκτοξεύοντας» την αναγνωσιμότητα και αξιοπιστία τους.
- ✓ Γεωγραφική κάλυψη. Κύριο πλεονέκτημα είναι η δυνατότητα που προσφέρεται στην επιχείρηση να απευθυνθεί σε αγορά τεράστιου μεγέθους, χωρίς γεωγραφικούς περιορισμούς. Επιπλέον η τμηματοποίηση της αγοράς με βάση διάφορα κριτήρια όπως γεωγραφικά, κοινωνικά, δημογραφικά, εθνικά, θρησκευτικά γίνεται πολύ πιο εύκολα, ενώ η «πληροφορία» μεταφέρεται εύκολα και με γρήγορους ρυθμούς.
- ✓ Οικονομική λύση - Value for money. Τα δίκτυα κοινωνικής δικτύωσης κάνουν εφικτή την παρουσίαση της επιχείρησης 24 ώρες του εικοσιτετράωρου όλο τον χρόνο, μειώνουν το λειτουργικό κόστος, ενώ τέλος διευκολύνεται η εύρεση νέου προσωπικού. Υπάρχουν πολλές μεγάλες εταιρίες που αυτή τη στιγμή απολαμβάνουν και εκμεταλλεύονται τα οφέλη του social media marketing. Χαρακτηριστικό είναι το παράδειγμα της Coca-Cola, που προσέγγισε το κοινό της μέσα από τα κοινωνικά δίκτυα, το photo και video sharing.
- ✓ Φιλικά προς το περιβάλλον. Το marketing γίνεται εξ ολοκλήρου online, οπότε δεν προκύπτουν μεγάλες ανάγκες για τη χρήση αναλώσιμων υλικών όπως χαρτί, πλαστικό, γυαλί ή αλουμίνιο. Ως εκ τούτου δεν υπάρχουν μεγάλης ποσότητας απόβλητα τα οποία απελευθερώνονται στο περιβάλλον και δεν προκαλείται μόλυνση.



Τα μέσα δικτύωσης αξιοποιούνται από κυβερνητικούς οργανισμούς, επιχειρήσεις, οργανισμούς αλλά κυρίως από απλούς χρήστες που οργανώνονται σε ομάδες κοινών ενδιαφερόντων και αλληλεπιδρούν.

Το 2019 2,95 δισεκατομμύρια άνθρωποι ή το 77% των χρηστών του Διαδικτύου σε παγκόσμιο επίπεδο, χρησιμοποιούσαν κάποιο μέσο κοινωνικής δικτύωσης τουλάχιστον μία φορά τον μήνα.

Το Facebook είναι μια πλατφόρμα κοινωνικής δικτύωσης που ξεκίνησε το 2004.

Οι 2,5 δισεκατομμύρια μηνιαίοι χρήστες, μπορούν να επικοινωνούν μεταξύ τους μέσω μηνυμάτων με τις επαφές τους και να τους ειδοποιούν όταν ανανεώνουν τις προσωπικές πληροφορίες τους.

Το Instagram είναι μια δωρεάν εφαρμογή κοινωνικής δικτύωσης που δίνει την δυνατότητα επεξεργασίας και κοινοποίησης φωτογραφιών και βίντεο στο διαδίκτυο. Οι 1 δισεκατομμύριο μηνιαίοι χρήστες μπορούν να μοιράζονται φωτογραφίες και βίντεο με τους ακολούθους (followers) ή με επιλεγμένη ομάδα φίλων, να σχολιάζουν και να δηλώνουν ότι μια δημοσίευση τους αρέσει.



Το Twitter δημιουργήθηκε το 2006 για να επιτρέπει στους 386 εκατομμύρια μηνιαίους χρήστες του να στέλνουν και να διαβάζουν σύντομα μηνύματα (μέχρι 280 χαρακτήρες), τα οποία ονομάζονται tweets. Τα μηνύματα μπορούν να αναγνωστούν και από μη συνδεδεμένους χρήστες, αλλά μόνο οι συνδεδεμένοι μπορούν να δημοσιεύσουν κείμενα.

Το Pinterest είναι ένα Κοινωνικό Μέσο στο οποίο κυριαρχούν οι εικόνες, στατικές και κινούμενες (GIF), ενώ επίσης υποστηρίζεται το βίντεο περιεχόμενο και έχει 336 εκατομμύρια μηνιαίους χρήστες.

Το Snapchat είναι μία εφαρμογή ανταλλαγής μηνυμάτων φωτογραφιών, βίντεο και κειμένου με 398 εκατομμύρια ενεργούς μηνιαίους χρήστες. Αυτό που την ξεχώρισε είναι το γεγονός πως τα μηνύματα τα οποία έχουν μικρή διάρκεια ζωής για προβολή από τον παραλήπτη έως ότου διαγραφούν αυτόματα.

Το TikTok είναι εφαρμογή για iOS και Android με 800 εκατομμύρια ενεργούς μηνιαίους χρήστες, η οποία επιτρέπει την δημιουργία και κοινοποίηση μικρών κωμικών και μη βίντεο. *Η εφαρμογή δημιουργήθηκε το 2017 για αγορές εκτός της Κίνας, στην οποία λειτουργούσε από το 2016.*

Το YouTube επιτρέπει κοινοποίηση, αποθήκευση, αναζήτηση και αναπαραγωγή ψηφιακών βίντεο και ψηφιακών ταινιών. Δημιουργήθηκε το 2005 και έχει 2 δισεκατομμύρια μηνιαίους χρήστες.

Στο Facebook Messenger 1,3 δισεκατομμύρια μηνιαίοι χρήστες συνομιλούν με φίλους τόσο στο κινητό όσο και στην κεντρική ιστοσελίδα. Είναι μια υπηρεσία άμεσων μηνυμάτων και εφαρμογή λογισμικού που παρέχει το κείμενο και τη φωνή. Το WhatsApp είναι μία εφαρμογή (ανήκει στο Facebook) άμεσης ανταλλαγής μηνυμάτων για smartphones. Εκτός από την ανταλλαγή μηνυμάτων κειμένου, οι 1 δισεκατομμύριο χρήστες μπορούν να στείλουν ο ένας στον άλλο εικόνες, βίντεο, ήχους και μηνύματα πολυμέσων.

Στο linkedin τα εγγεγραμμένα μέλη έχουν τη δυνατότητα να δημιουργήσουν το προσωπικό επαγγελματικό τους προφίλ, να συνδεθούν με άλλους χρήστες, να αναζητήσουν εργασία, αλλά και να δημιουργήσουν πελατολόγιο. Το LinkedIn είναι ιστοχώρος επαγγελματικής κοινωνικής δικτύωσης που ιδρύθηκε το 2002.



Το Social Media Marketing είναι το μοναδικό διαφημιστικό εργαλείο που μπορεί να φέρει αποτελέσματα σε πολύ μικρό χρονικό διάστημα. Βασικός στόχος της συγκεκριμένης υπηρεσίας είναι να αξιοποιήσει όλα τα νέα μέσα του διαδικτύου για να προσεγγίσει επισκέπτες – καταναλωτές ή να αυξήσει το brand name του πελάτη. Τα κοινωνικά μέσα (Social Media) είναι ο συνδυασμός της τεχνολογίας, της διαδικτυακής κοινωνικοποίησης και της πληροφορίας (κείμενο, ήχος, video και φωτογραφίες). Η ενίσχυση της επισκεψιμότητας είναι αποτέλεσμα μιας

στρατηγικής marketing για τα social media, η οποία, με τη σειρά της, οφείλει την επιτυχία της στην δημιουργία περιεχομένου που να ανταποκρίνεται στο ύφος του κάθε μέσου. Για παράδειγμα, οι χρήστες του Facebook δείχνουν να αγαπούν το περιεχόμενο σε μορφή video. Αυτό όμως μπορεί να μην ισχύει στην περίπτωση των χρηστών του Twitter.

Εφαρμογές της πληροφορικής στις Υπηρεσίες

Εκτός από τις επιχειρήσεις και οι υπηρεσίες του Δημοσίου έχουν επηρεαστεί σε μεγάλο βαθμό από την αξιοποίηση εφαρμογών της πληροφορικής και παρατηρήθηκαν σε αυτές θετικές αλλαγές, σε σχέση με την καλύτερη εξυπηρέτηση του πολίτη. Ηλεκτρονικά μητρώα, εφαρμογές οικονομικής διαχείρισης και μισθοδοσίας επιφέρουν σημαντικά οφέλη, όπως η εξοικονόμηση χρόνου, η ελαχιστοποίηση σφαλμάτων και ο αποτελεσματικότερος διοικητικός έλεγχος. Με στόχο την πιο άνετη και άμεση εξυπηρέτηση του πελάτη, οι τράπεζες άλλαξαν σημαντικά τον τρόπο λειτουργίας τους με την χρήση των εξελίξεων των επιστημών πληροφορικής και επικοινωνιών. Σήμερα όλες οι τραπεζικές συναλλαγές πραγματοποιούνται με τη βοήθεια υπολογιστή (web-banking) χωρίς χρονικό περιορισμό και φυσική παρουσία στο τραπεζικό κατάστημα και για αυτό, έχουν αναπτυχθεί κατάλληλα τραπεζικά και χρηματοπιστωτικά πληροφοριακά συστήματα, όπως παρακολούθηση καταθετών, συναλλαγών, δανείων, πιστώσεων και επενδύσεων.



Οι υπάλληλοι των τραπεζών χειρίζονται συστήματα συνδεδεμένα άμεσα με τον κεντρικό υπολογιστή της τράπεζας (on-line) ενώ οι καταναλωτές μπορούν να κάνουν συναλλαγές κάθε ώρα της ημέρας με την χρήση των αυτόματων μηχανές συναλλαγών (Automatic Teller Machines ATM). Τα **ATM** βρίσκονται εγκατεστημένα παντού και είναι συνδεδεμένα με τον κεντρικό υπολογιστή της τράπεζας, από τον οποίο ενημερώνονται για τα υπόλοιπα των λογαριασμών.



Εφαρμογές στον τουρισμό

Από τις δεκαετίες 1960 και 1970, τα συστήματα ηλεκτρονικών κρατήσεων που κατασκευάστηκαν από αεροπορικές εταιρείες και ομάδες τουριστικών ξενοδοχείων με βάση δίκτυα μπορούν να θεωρηθούν ότι σήμαναν την ανάπτυξη του ηλεκτρονικού τουρισμού, που όμως πήρε την σημερινή του μορφή με την εμφάνιση του Διαδικτύου και αποτελεί πλέον τον κυρίαρχο τρόπο μετακινήσεων και διακοπών. Η επικράτηση του ήταν αναμενόμενη από την στιγμή που προσέφερε τεράστια οφέλη τόσο τους καταναλωτές όσο και για τις επιχειρήσεις, όπως:

- ✓ Υπηρεσίες επικοινωνίας όπου αυτή πραγματοποιείται μέσω email και forums.
- ✓ Υπηρεσίες πληροφόρησης κυρίως για τα καταλύματα αλλά και τους προορισμούς όπως για παράδειγμα η πρόγνωση του καιρού.
- ✓ Υπηρεσίες συναλλαγών όπως πληρωμές δωματίων, εισιτηρίων κ.λπ.



Για τις τουριστικές επιχειρήσεις, η ηλεκτρονική διαχείριση τουρισμού απαλλάσσει το προσωπικό από δυσκίνητες χειρωνακτικές διαδικασίες και βελτιώνει την αποτελεσματικότητά του. Ακόμα, καθιστά πιο προσιτές και αξιόπιστες τις πληροφορίες της αγοράς.

Ο τουρισμός είναι μια ολοκληρωμένη βιομηχανία που αποτελείται από πολλές υποβιομηχανίες και απαιτεί τον συντονισμό και τη συνεργασία κάθε μιας από αυτές.

Υπάρχουν πολύπλοκες σχέσεις αντιπροσώπευσης, συναλλαγής και συνεργασίας μεταξύ διαφόρων τουριστικών επιχειρήσεων στην σίτιση, στην διαμονή, στις μεταφορές, στα τουριστικά είδη και σχετικές αγορές και στην διασκέδαση.

Σε αυτούς τους τομείς, ο ηλεκτρονικός τουρισμός σε σχέση με τον παραδοσιακό τουρισμό, έχει κατορθώσει να προσφέρει κάποια σημαντικά πλεονεκτήματα, όπως για παράδειγμα:

- ✓ Διευκόλυνση της πρόσβασης σε τουριστικά προϊόντα και υπηρεσίες,
- ✓ Λιγότερο χρόνο για παροχή υπηρεσιών και πληροφοριών,
- ✓ Πληροφορίες και οπτικό υλικό που μειώνουν το αίσθημα ανασφάλειας για την ορθότητα επιλογής ενός καταλύματος ή προορισμού
- ✓ Ευκολία στη χρήση
- ✓ Μείωση των μεσαζόντων, άρα μικρότερο κόστος για τον καταναλωτή και την επιχείρηση



Η ανάπτυξη εφαρμογών με πληροφορίες για αξιοθέατα, χάρτες και οδηγίες μετακίνησης, εντυπώσεις άλλων τουριστών που επισκέφτηκαν τον προορισμό, είναι μερικές από τις διευκολύνσεις που παρέχει η ψηφιακή τεχνολογία στον τουρίστα. Η ανάπτυξη του αστικού τουρισμού αποτελεί σημαντικό μέρος της οικονομικής ανάπτυξής της κάθε πόλης, είτε πρόκειται για μια πόλη με αρχαία ιστορία και πολιτισμό, είτε πρόκειται για μια πόλη με όμορφα φυσικά τοπία ή μια πόλη με χώρους ψυχαγωγίας.

Τα *περίπτερα πληροφοριών (information kiosks)* που έχουν δημιουργηθεί από τις αρχές κάθε πόλης, βρίσκονται εγκατεστημένα σε αεροδρόμια, λιμάνια ή τουριστικά θέρετρα. Είναι ένα ακόμα στοιχείο με το οποίο η τεχνολογία της πληροφορικής βοηθά τον τουρίστα στον οποίο θα δείξει τα χαρακτηριστικά της πόλης, θα τον καθοδηγήσει και θα ξεναγήσει εύκολα και γρήγορα τον επισκέπτη στην πόλη και στα σημεία τουριστικού ενδιαφέροντος που διαθέτει.

Τα *περίπτερα πληροφοριών (information kiosks)* είναι υπολογιστικά συστήματα που δίνουν πληροφορίες στην κύρια οθόνη τους που υπάρχουν οι επιλογές:



- **Οδηγίες και βοήθεια:** Παρουσιάζει τα σημεία τουριστικού ενδιαφέροντος με χρήσιμες πληροφορίες και παρέχει ένα τηλεφωνικό αριθμό για το δημοτικό κέντρο ταξιδιωτικής εξυπηρέτησης για όσους χρειάζονται μεγαλύτερη βοήθεια.
- **Ξενοδοχεία:** Αυτή η οθόνη περιέχει λίστες ξενοδοχείων ανάλογα με την τιμή και την κατηγορία στην οποία εντάσσονται. Κάνοντας κλικ σε οποιοδήποτε όνομα ξενοδοχείου παρέχονται περισσότερες πληροφορίες όπως είναι η διεύθυνση και ο τηλεφωνικός αριθμός. Υπάρχει επίσης πρόσβαση σε χάρτη και σε οδηγίες για την τοποθεσία, οι οποίες μπορούν να εκτυπωθούν.
- **Σίτιση:** σε αυτή την οθόνη είναι δυνατόν να επιλεγούν εστιατόρια βάσει τις γαστρονομικές απαιτήσεις του τουρίστα, και πληροφορίες όπως διεύθυνση και τηλέφωνο και πρόσβαση σε χάρτη και σε οδηγίες.

- **Διασκέδαση:** Μέσα από έναν αριθμό επιλογών όπως Ημερολόγιο Εκδηλώσεων, Αξιοθέατα, Σημεία Συνάντησης, Αναψυχή, Νυχτερινή ζωή, Ψώνια παρέχονται περισσότερες επιλογές για τα θέματα που θα επιλεγθούν.
- **Καιρός:** Παρέχει μια πρόβλεψη 1-5 ημερών για την μετεωρολογική κατάσταση στην ευρύτερη περιοχή.

Εφαρμογές της πληροφορικής στην Εκπαίδευση

Η σύγχρονες τεχνολογίες πληροφορικής και επικοινωνιών δεν αφορούν μόνο διαδικτυακές διδασκαλίες αλλά επεκτείνονται σε αυτό που ονομάζεται ηλεκτρονική μάθηση και περιλαμβάνει οτιδήποτε λαμβάνει χώρα στο διαδίκτυο και σε ψηφιακές πλατφόρμες που χρησιμοποιείται για εκπαίδευση, ή κατάρτιση. Ακόμα και μια εργασία που υποβάλλει ένας φοιτητής μέσω μιας διαδικτυακής πλατφόρμας είναι ηλεκτρονική μάθηση, όπως επίσης τα διαδικτυακά σεμινάρια ή η εκπαίδευση μέσω απλής τηλεδιάσκεψης κ.λπ.

Η διαδικτυακή μάθηση, εφαρμόστηκε σε μεγάλη έκταση την εποχή που εφαρμόζονταν σε πολλές χώρες τα περιοριστικά μέτρα για την αντιμετώπιση της πανδημίας covid-19. Η εμπειρία αυτή έφερε στην επιφάνεια διστάμενες απόψεις για την ηλεκτρονική μάθηση. Μετά από περισσότερο από ένα χρόνο εξ αποστάσεως εκπαίδευσης, οι μαθητές έχουν κουραστεί από τα διαδικτυακά μαθήματα και επιθυμούσαν την προσωπική επαφή με τους συμμαθητές τους και τους καθηγητές τους. Όμως έγιναν ορατά και πλεονεκτήματα που οφείλονται στις μεγάλες δυνατότητες που προσφέρει το e-learning:

- Παρέχει μαθησιακή ευελιξία με τη δυνατότητα χρήσης τεχνολογικών επιτευγμάτων (π.χ. virtual reality και προσομοιώσεις) που εμπλουτίζουν το εκπαιδευτικό υλικό και το καθιστούν πιο ενδιαφέρον και διαδραστικό.
- Κάνει την εκπαίδευση πιο προσιτή και προσβάσιμη ακόμα και σε όσους είχαν χρονικούς περιορισμούς από άλλες ενασχολήσεις. Η ηλεκτρονική μάθηση βοηθά επίσης μαθητές διαφορετικών ικανοτήτων να έχουν πιο εύκολη πρόσβαση στην εκπαίδευση, όπως η μάθηση από την άνεση του σπιτιού.
- Εξοικονομεί χρόνο σε διδάσκοντες και εκπαιδευόμενους. Οι μαθητές δεν



χρειάζεται πλέον να διανύουν μεγάλες αποστάσεις από και προς την τάξη, εξοικονομώντας έτσι έξοδα μεταφοράς.

Τα εκπαιδευτικά ιδρύματα χρειάζονται μόνο να διαχειρίζονται την υποδομή της τεχνολογίας και πληροφορικής τους και δεν χρειάζεται πλέον να μισθώνουν διδακτικό χώρο.

Ωστόσο, όλα αυτά τα πλεονεκτήματα θα προκύψουν μόνο όταν η ηλεκτρονική μάθηση εφαρμοστεί σωστά. Ως εκ τούτου, κατά το σχεδιασμό διαδικτυακών μαθημάτων, είναι σημαντικό να παρέχεται η ευρύτερη δυνατή επιλογή για να ταιριάζει στις ανάγκες διαφορετικών μαθητών.

Ένας ορισμός για την εκπαίδευση εξ αποστάσεως ή τηλε-εκπαίδευση (distance learning) είναι όταν ο εκπαιδευόμενος και ο εκπαιδευτής επικοινωνούν και συνεργάζονται μέσω δικτύου υπολογιστών.

Στην περίπτωση της σύγχρονης τηλεεκπαίδευσης, το μάθημα γίνεται κανονικά, αλλά οι εκπαιδευόμενοι και ο διδάσκων καθηγητής μπορούν να βρίσκονται σε διαφορετικό τόπο ο καθένας και χρησιμοποιώντας τεχνολογίες τηλεδιάσκεψης να βρίσκονται όλοι σε μία εικονική αίθουσα διδασκαλίας.

Για την πραγματοποίηση μαθήματος μέσω τηλεεκπαίδευσης, θα πρέπει η εικονική αίθουσα να είναι έτσι, ώστε να παρέχει και όλες τις τεχνικές δυνατότητες, που προσφέρει μία κανονική αίθουσα.

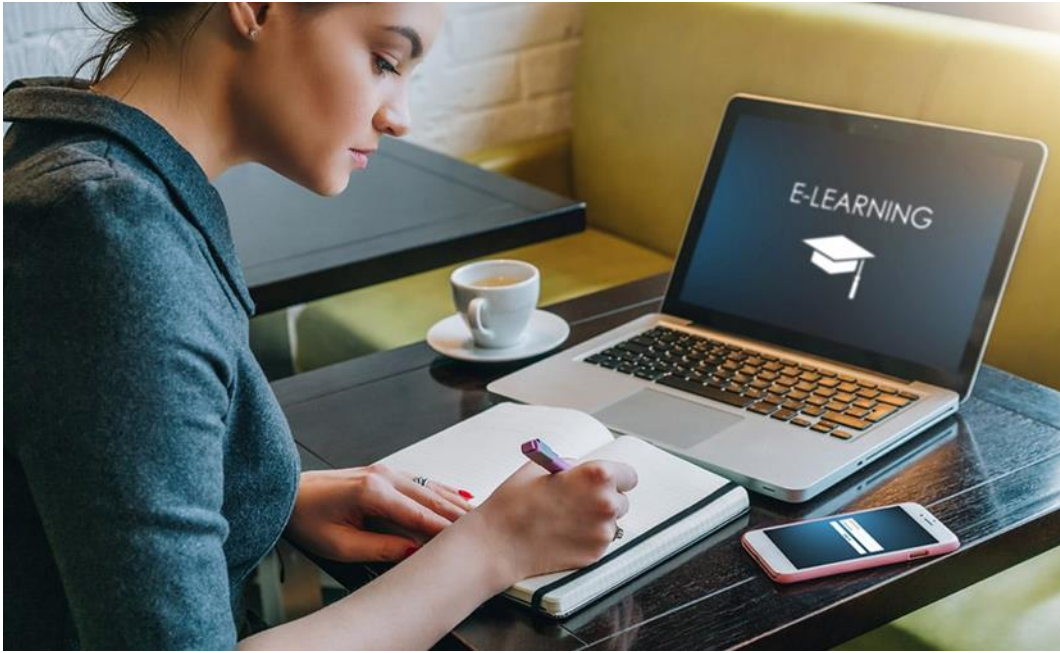
Στην ασύγχρονη εκπαίδευση, δεν υπάρχει ταυτόχρονη, έστω διαδικτυακή, παρουσία καθηγητή και μαθητών. Ο εκπαιδευόμενος μπορεί να παρακολουθήσει το βιντεοσκοπημένο μάθημα, οποιαδήποτε ώρα επιλέξει.



Αν και το πλεονέκτημα της χρονικής ευελιξίας είναι σημαντικό, η έλλειψη επικοινωνίας με τον διδάσκοντα μπορεί να δημιουργεί μαθησιακά κενά στον εκπαιδευόμενο.

Μια ακόμα συμβολή της σύγχρονης τεχνολογίας στην εκπαιδευτική διαδικασία είναι ότι οι εκπαιδευόμενοι έχουν πρόσβαση online ψηφιακή βιβλιοθήκη η οποία περιέχει όλα τα σχολικά βιβλία.

Η νέα ψηφιακή κουλτούρα επηρεάζει την εκπαίδευση, αλλάζοντας τις αλληλεπιδράσεις μεταξύ των ατόμων και μεταμορφώνοντας τον τρόπο διαμόρφωσης ενός πολίτη, επομένως είναι απαραίτητο να εστιάσουμε στα γεγονότα, τους φορείς και τις διαδικασίες που υπάρχουν στον σχολικό χώρο. Το κράτος και η αγορά εμφανίζονται ως διαμορφωτές του σχολείου, αλλάζοντας τη διδακτική του δομή κάθε φορά που επιβάλλονται νέες οικονομικές και κοινωνικές δυναμικές, δίνοντας αφορμή για την επικαιροποίηση της εκπαίδευσης για οικονομικούς σκοπούς. Οι δάσκαλοι και οι μαθητές που είναι οι πρωταγωνιστές στο χώρο του σχολείου, συχνά χρησιμοποιούν εργαλεία για να εισάγουν τεχνολογικές καινοτομίες που αναδιαμορφώνουν τον κόσμο στο περιβάλλον τους.



Η ψηφιακή τεχνολογία βοηθάει ώστε η εκπαίδευση να αντιμετωπίζεται:

- ✓ σαν αναφαίρετο ανθρώπινο δικαίωμα, ιδιαίτερο απαραίτητο στην εποχή της ραγδαίας εξέλιξης των τεχνολογιών πληροφορικής και επικοινωνίας
- ✓ σαν πολιτικό, οικονομικό, κοινωνικό και πολιτιστικό αγαθό, όπως απαιτεί η σημερινή κοινωνική δυναμική του σήμερα.

Εφαρμογές της πληροφορικής στις Επιστήμες και την Έρευνα

Η Πληροφορική αν και αποτελεί ένα μέλος του συνόλου των επιστημών και επικοινωνεί και συνεργάζεται με τις υπόλοιπες, διατηρεί την αυτονομία της και διαφέρει επειδή η δική της εξέλιξη, έχει άμεση εφαρμογή και αξιοποίηση από τις άλλες επιστήμες για να προχωρήσουν στα επιτεύγματα τους.

Γενικά, οι υπολογιστές είναι εργαλεία καταγραφής, δημιουργίας, ανάλυσης και διάδοσης της επιστημονικής γνώσης και της επιστημονικής έρευνας και συμμετέχουν με καθοριστικό τρόπο σε κάθε νέα ανακάλυψη .

Εφαρμογές της πληροφορικής στις Τέχνες

Οι υπολογιστές έχουν υψηλές δυνατότητες στην επεξεργασία και σύνθεση ήχου και εικόνας και στα Video και Animation (κινούμενα γραφικά), και με αυτό τον τρόπο συμβάλουν στις τέχνες. Η αισθητική αντανάκλαση της πραγματικότητας, είναι μια ιδιομορφία που υπάρχει σε όλα τα είδη τέχνης: Λογοτεχνία, ζωγραφική, χαρακτική, γλυπτική, αρχιτεκτονική, κινηματογράφος, θέατρο, μουσική, διακοσμητική τέχνη



κ.λπ. Μια σύγχρονη άποψη για την τέχνη, ιδιαίτερα ανάμεσα σε νέους καλλιτέχνες, είναι ότι δημιουργικότητα δεν είναι να δημιουργείς κάτι καινούργιο, αλλά να μαθαίνεις να χρησιμοποιείς ό,τι ήδη υπάρχει (conceptual art).

Ο διαδικτυακός καλλιτέχνης λοιπόν, αντί να παράγει, συνθέτει και αποσυνθέτει πληροφορίες που υπάρχουν, προσπαθώντας να αφυπνίσει τη συνείδηση και το



συναίσθημα του χρήστη.

Είναι πολύ σημαντική η συνεισφορά της ψηφιακής τεχνολογίας στο χώρο του πολιτισμού, όπου χρησιμοποιείται σαν:

- ✓ εργαλείο και εικαστικό μέσο δημιουργίας. Με τις υψηλές δυνατότητες στην επεξεργασία στην εικόνα βοηθάει πολύ τις τέχνες αφού πλέον οι τεχνικές, οι τεχνοτροπίες και οι επιρροές μεταδίδονται, επεξεργάζονται και εξελίσσονται σε πολύ γρήγορο χρόνο.
- ✓ μέσο για τη διάδοση του πολιτισμού μιας χώρας. Στο χώρο της προβολής της πολιτιστικής κληρονομιάς η συμβολή της ψηφιακής τεχνολογίας είναι σημαντική με την καταγραφή και ψηφιοποίηση εικόνων, πινάκων, μνημείων κ.λπ.

Εφαρμογές της πληροφορικής στην Ψυχαγωγία

Ο ηλεκτρονικός υπολογιστής είναι πλέον μέσο ψυχαγωγίας, το 'κατέβασμα' (downloading) των πολυάριθμων παιχνιδιών που μπορούν να ληφθούν από το διαδίκτυο δωρεάν, η μουσική και η αναπαραγωγή της, τα χόμπι και η αναπαραγωγή video η επίσκεψη στα δωμάτια συνομιλίας ή απλά το 'σερφάρισμα' στο διαδίκτυο είναι μερικές από τις χρήσεις που έχουν ανακαλύψει οι χρήστες ώστε να χαλαρώσουν και να ψυχαγωγηθούν κατά την πλοήγηση στο διαδίκτυο και από τα οποία μπορεί να επιλέξει για λήψη στον προσωπικό υπολογιστή του. Τα μέσα κοινωνικής δικτύωσης έχουν εξελιχθεί σε βασικό στοιχείο της επικοινωνίας μεταξύ χρηστών του διαδικτύου και δημιουργούν μια ψηφιακή κοινότητα χωρίς



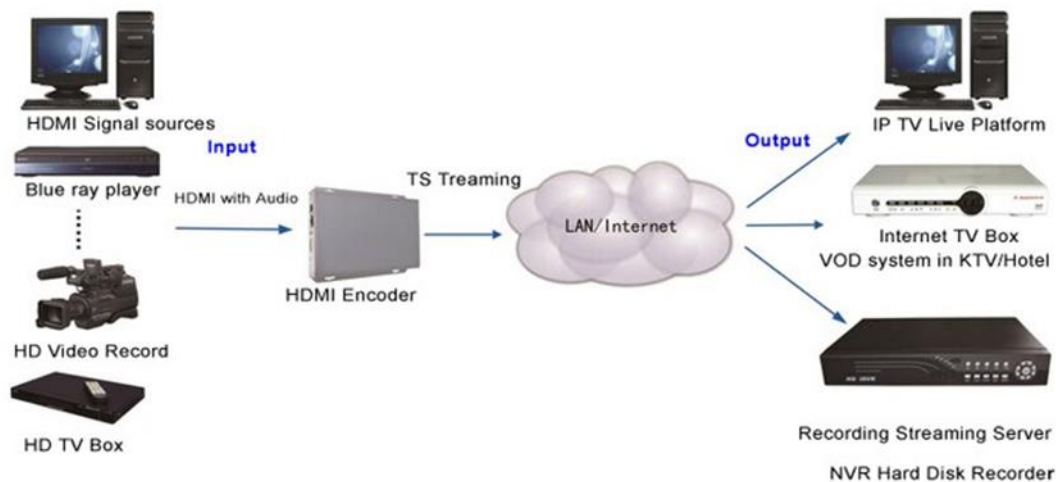
χιλιομετρικές αποστάσεις.

Η διανομή εικόνας, βίντεο ή ήχου μπορεί να έχει δύο μορφές:

- ✓ broadcasting όπου όλοι οι χρήστες, που είναι συνδεδεμένοι στο δίκτυο και έχουν τον κατάλληλο εξοπλισμό, μπορούν να λάβουν το υλικό.
- ✓ multicasting, όπου το υλικό διανέμεται μόνο σε χρήστες που ανήκουν σε συγκεκριμένες εξουσιοδοτημένες ομάδες.

Όσον αφορά στο εύρος ζώνης, **τα 200Kbps** που απαιτούνται για χαμηλής ποιότητας εικόνα μπορούν να διακινηθούν μέσω όλων των LAN. Όταν ζητείται καλή ποιότητα, η οποία απαιτεί εύρος ζώνης στα **6-10Mbps**. Το εύρος ζώνης της δορυφορικής μετάδοσης είναι ικανοποιητικό ενώ η αυξημένη καθυστέρηση (γύρω στα 570ms) είναι ανεκτή σε τέτοιες non-interactive εφαρμογές.

Όταν η διανομή ψηφιακού υλικού έχει στόχο διαφορετικές ομάδες χρησιμοποιούνται τα δίκτυα IP με μέσο μετάδοσης το Internet. Το κύριο πρόβλημα σε αυτήν την περίπτωση είναι το περιορισμένο εύρος ζώνης και η μεταβλητότητα της ποιότητας των υπηρεσιών. Αν το IP δίκτυο του χρήστη δεν έχει δυνατότητες broadcasting είναι εφικτό να προσομοιωθεί το broadcasting αν γίνει multicasting σε ανοικτή ομάδα, στην οποία μπορεί να συμμετάσχει οποιοσδήποτε, όπως λειτουργεί η τηλεόραση μέσω Internet.



Για τις multimedia εφαρμογές που είναι εγκατεστημένες σε κεντρικό server, όταν είναι προσπελάσιμες από τον χρήστη, η εφαρμογή χαρακτηρίζεται interactive, σε κάθε άλλη περίπτωση είναι μια εφαρμογή διανομής.

Ο χρήστης μπορεί να «κατεβάσει» την εφαρμογή και να την αποθηκεύσει και να την αναπαράγει τοπικά (downloading mode). Αυτό προϋποθέτει μικρούς όγκους πληροφορίας, γιατί διαφορετικά ο χρήστης θα περιμένει πολύ (περισσότερο από 5 δευτερόλεπτα) και εξασθενεί η αίσθηση της διαδραστικότητας. Όταν η καθυστέρηση μετάδοσης περιορίζεται σε 3-4sec τότε δεν υπάρχει πρόβλημα. Αν υπάρχει μεγαλύτερη καθυστέρηση τότε πρέπει να γίνει μετάδοση της πληροφορίας σε πραγματικό χρόνο, δηλαδή αναπαράγεται κάθε τμήμα της πληροφορίας μόλις φθάνει, χωρίς να περιμένει την ολοκλήρωση της μετάδοσης ολόκληρης της πληροφορίας.

Μια ειδική εφαρμογή αυτής της κατηγορίας, είναι το video-on-demand. Ο χρήστης επιλέγει την ταινία ή το βίντεο που θέλει να παρακολουθήσει και αυτή θα του αποστέλλεται μέσω δικτύου. Η ιδιαιτερότητα των VOD servers έγκειται στο ότι πρέπει να στέλνει διαφορετικά ρεύματα video για τις πολλές διαφορετικές αιτήσεις που λαμβάνουν με μικρή διαφορά χρόνου. Κάτι τέτοιο είναι πρακτικά ανέφικτο, γι' αυτό στην πράξη οι αιτήσεις ομαδοποιούνται και εξυπηρετούνται ανά τακτά (αλλά αρκετά μεγάλα) χρονικά διαστήματα. Για καλή ποιότητα εικόνας απαιτούνται **3-6Mbps** με συμπίεση **MPEG-2** ή **15-34Mbps** για ποιότητα τηλεόρασης υψηλής ανάλυσης.



Διαχείριση ψηφιακού υλικού

Ψηφιοποίηση υλικού είναι η εργασία μετατροπής έντυπου ή αναλογικού υλικού σε ψηφιακή μορφή. Η διεργασία αυτή γίνεται με μεθόδους και τεχνικές όπως είναι η

καταγραφή (βιντεοκάμερα) και η σάρωση (σαρωτής) κ.λπ. Η διαδικασία της ψηφιοποίησης είναι μια χρονοβόρα διαδικασία που απαιτεί πολύ χρόνο και υψηλό κόστος.

Η επιλογή του τμήματος του διαθέσιμου υλικού για ψηφιοποίηση εξαρτάται από τις ανάγκες και την πολιτική του φορέα για τον οποίο θα γίνει η διαδικασία.

Η εφαρμογή της Πληροφορικής στις ψηφιακές βιβλιοθήκες (digital libraries) επιτρέπει την εύκολη και ταχύτατη πρόσβαση σε πηγές πληροφοριών αποθηκευμένων σε ψηφιακή μορφή.

Πολύ τίτλοι δόθηκαν για να περιγράψουν αυτό το νέο τύπο βιβλιοθήκης, όπως εικονική βιβλιοθήκη (virtual library), ηλεκτρονική βιβλιοθήκη (electronic library) και βιβλιοθήκη χωρίς τοίχους (library without walls).

Τελικά καθιερώθηκε ο όρος Ψηφιακή Βιβλιοθήκη που χρησιμοποίησε ο οργανισμός Digital Libraries Initiative όταν το 1994 χρηματοδοτήθηκε από το Εθνικό ίδρυμα Ερευνών των Ηνωμένων Πολιτειών, με σκοπό τη δημιουργία και χρήση ψηφιακών βιβλιοθηκών.

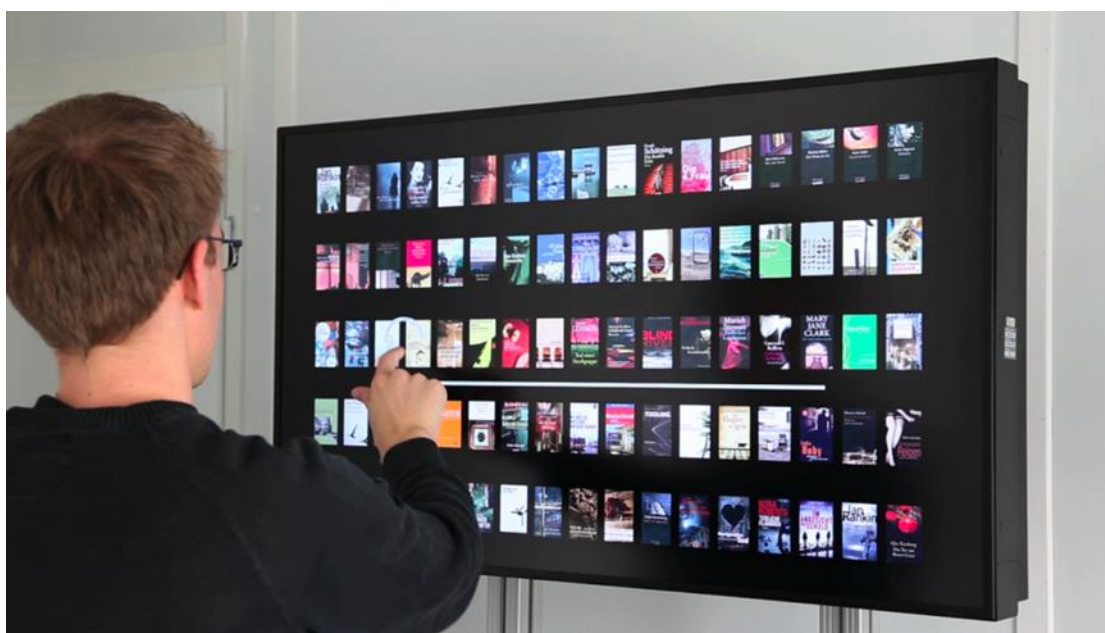


Η ψηφιακή βιβλιοθήκη είναι κατανεμημένο υπολογιστικό περιβάλλον που απλοποιεί σημαντικά τις λειτουργίες του ψηφιακού υλικού που περιέχει. Το περιβάλλον αυτό επιτρέπει την πραγματοποίηση διαδικασιών, όπως είναι η

αναζήτηση σε βάσεις δεδομένων, η ανάκτηση πληροφοριών και η εξόρυξη γνώσης από δεδομένα μέσα από το περιβάλλον του παγκόσμιου ιστού.

Η δομή της περιλαμβάνει τον ηλεκτρονικό εξοπλισμό για τη δημιουργία, αποθήκευση και προσπέλαση του ψηφιακού υλικού μαζί με το αναγκαίο σε κάθε περίπτωση λογισμικό, τη δημιουργία ευρετηρίων και καταλόγων, και την αναζήτηση πληροφοριών σε αυτό.

Ένα βασικό κριτήριο εγκυρότητας αλλά και αξιοπιστίας μιας ψηφιακής βιβλιοθήκης είναι η ακεραιότητα και η προστασία του αποθηκευμένου ψηφιακού υλικού.



Ψηφιακά αντικείμενα, ιδιότητες και μεταδεδομένα

Αυτό που χαρακτηρίζει την το περιεχόμενο μιας Ψηφιακής Βιβλιοθήκης είναι το Ψηφιακό Αντικείμενο (digital object), το σύνολο πληροφοριών που έχει υποστεί τη ψηφιοποίηση, έχει αποθηκευτεί με συγκεκριμένη μορφή (π.χ. με τη μορφή εικόνας, ήχου, κειμένου ή video) και χαρακτηρίζεται από ιδιότητες οι οποίες περιγράφονται με τη βοήθεια μεταδεδομένων, δηλαδή δεδομένων που περιγράφουν άλλα δεδομένα, όπως τίτλος, θέμα κ.λπ.

Τα πολυμέσα είναι ο κλάδος της πληροφορικής που ασχολείται με τον συνδυασμό ψηφιακών δεδομένων πολλαπλών μορφών, π.χ. εικόνας, ήχου, κειμένου, γραφικών, κινούμενης εικόνας (animation) και βίντεο, για την αναπαράσταση, παρουσίαση, αποθήκευση, μετάδοση και επεξεργασία πληροφοριών.

Τα συστήματα που χρησιμοποιούνται για την ανάπτυξη των πολυμέσων είναι:

- ✓ Συσκευή βίντεο
- ✓ Κάρτα βίντεο (video grabber)
- ✓ Βιντεοκάμερα
- ✓ Μικρόφωνο
- ✓ Ψηφιακό κασετόφωνο (DAT, Digital Audio Tape)
- ✓ Ψηφιακή φωτογραφική μηχανή
- ✓ Σαρωτής (scanner)
- ✓ Εγγραφέας CD (CD Recorder)
- ✓ Οδηγός DVD (Digital Video Disc)
- ✓ Εξωτερικός σκληρός δίσκος



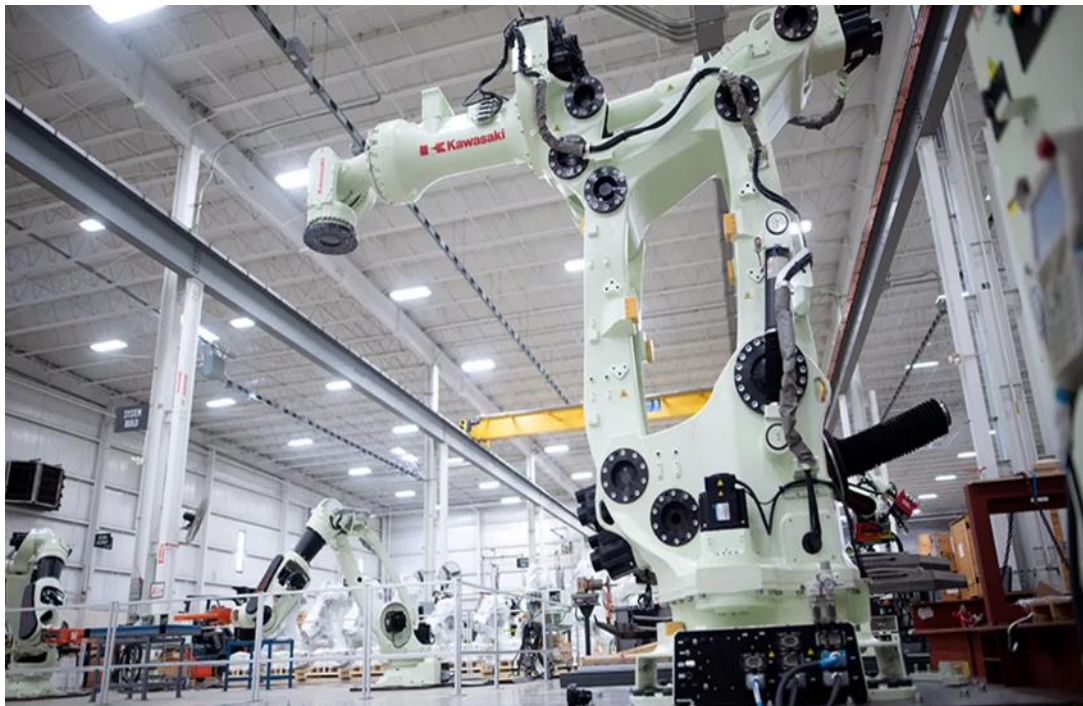
Μια εξέλιξη που αξιοποιεί την προστιθέμενη αξία των πολυμέσων είναι τα ψηφιακά βιβλία. Αρχικά το ψηφιακό βιβλίο ήταν μια ψηφιοποιημένη έκδοση του παραδοσιακού βιβλίου που μπορούσε να διαβαστεί με ειδικό αναγνώστη

ψηφιακών βιβλίων (ebook reader). Σήμερα τα ψηφιακά βιβλία είναι πλέον διαδραστικά. Δίνουν στον χρήστη την αίσθηση του ξεφυλλίσματος (flipping books), προσφέρουν τη δυνατότητα σελιδοδείκτη, επισημειώσεων και υπογραμμίσεων, ενώ ταυτόχρονα το περιεχόμενο είναι πολυμεσικό. Στις σελίδες ενός ψηφιακού βιβλίου ενσωματώνονται βίντεο, συλλογές από εικόνες, και διαδραστικά στοιχεία, π.χ. ένα τεστ ερωτήσεων. Τα βιβλία αυτά είναι εμπλουτισμένα, προσελκύουν την προσοχή και το ενδιαφέρον του αναγνώστη, ο οποίος μπορεί να τα διαβάσει μέσω ενός υπολογιστή, μιας ταμπλέτας ή ενός κινητού τηλεφώνου.

Εφαρμογές Ρομποτικής (Robotics)

Από το σλαβικό robota που σημαίνει εργασία γεννήθηκε η λέξη ρομπότ. Οι αυτόματες ακούραστες αυτές μηχανές με την προγραμματισμένη τους συμπεριφορά, δημιουργήθηκαν για να αντικαταστήσουν τον άνθρωπο και να τον απαλλάξουν από τις χειρωνακτικές εργασίες.

Η Ρομποτική είναι ένας συνδυασμός της πληροφορικής, της ηλεκτρονικής και της μηχανολογίας και φυσικά ανήκει στον κλάδο της τεχνολογίας. Η αλματώδης εξέλιξη



της ρομποτικής έχει προσφέρει σε πολλούς τομείς όπως στη βιομηχανία, την ιατρική, τη διασκέδαση καθώς και στην ανθρώπινη καθημερινότητα. Έχουν χρησιμότητα από το να καθαρίσουν και να σκουπίσουν ένα χώρο, μέχρι και να χειρουργήσουν, ή και να κατασκευάσουν αυτοκίνητα και πολλά άλλα. Τα ρομπότ

χρησιμοποιούνται καθημερινά στην κατασκευή μικροεπεξεργαστών, στην εξερεύνηση του διαστήματος και των βυθών, και γενικά σε εργασίες που πραγματοποιούνται σε επικίνδυνο χώρους με δύσκολες συνθήκες για την εξερεύνηση τους από το άνθρωπο.

Φυσικά αξιοσημείωτη παρουσία έχουν στον χώρο της ιατρικής και μάλιστα στο τομέα της χειρουργικής. Συστήματα ρομποτικής τηλεχειρουργικής έχουν ήδη χρησιμοποιηθεί για υπερατλαντικές επεμβάσεις. Ρομποτικοί βραχίονες που ενεργοποιούνται με τη φωνή μπορούν και χειρίζονται τη λαπαροσκοπική κάμερα.



Το Ινστιτούτο Ρομποτικής της Αμερικής, ονομάζει ρομπότ ένα επαναπρογραμματιζόμενο βραχίονα που εκτελεί πολλές λειτουργίες, σχεδιασμένο και προγραμματισμένο για την επίτευξη διάφορων εργασιών.

Αυτός ο ορισμός αποτέλεσε το κίνητρο για την ανάπτυξη όχι μόνο των ρομποτικών συστημάτων αλλά και των συστημάτων αυτοματισμού που εμφανίζονται και στις αυτοματοποιημένες γραμμές παραγωγής σε τομείς της βιομηχανίας όπως η αυτοκινητοβιομηχανία, η κλωστοϋφαντουργία, η βιομηχανία ηλεκτρικών και ηλεκτρονικών ειδών, κ.λπ.

Η τεχνολογική πρόοδος έφερε την ανάπτυξη των ευφυών και αυτόνομων ρομπότ που είναι σε θέση να παίρνουν αποφάσεις, να αυτενεργούν, να μαθαίνουν, και γενικά να επιτελούν εκείνες τις λειτουργίες που κατά κανόνα αποδίδονται στην

ανθρώπινη φύση. Αν και αναπτύχθηκαν ρομπότ πολύ αποτελεσματικά, χρησιμοποιώντας αλγοριθμικές προσεγγίσεις, η ευφυΐα που τους έχει αποδοθεί είναι στοιχειώδης και υπάρχει τεράστια απόσταση για την απόδοση νοημοσύνης σε ρομπότ - ανδροειδή, που συμπεριφέρονται όπως οι άνθρωποι.

Ψηφιακές τεχνολογίες και αλλαγές

Είναι γεγονός ότι οι ψηφιακές τεχνολογίες αλλάζουν με ραγδαίους ρυθμούς τις κοινωνίες και τις οικονομίες συνθήκες της πραγματικότητας. Η εξάπλωση και η εφαρμογή των ψηφιακών τεχνολογιών οδήγησε σε μια τεράστια αύξηση της παραγωγικότητας και επαναπροσδιόρισε τις δομές παραγωγής και τις αλληλεπιδράσεις ανθρώπου-μηχανής μεταβάλλοντας τον ρόλο που διαδραματίζουν οι άνθρωποι στην παραγωγή.

Υπάρχει σκεπτικισμός για τους κινδύνους που προκαλεί η τεχνολογία, όπως η πιθανότητα απώλειας θέσεων εργασίας και οι αλλαγές στη δομή της απασχόλησης. Ενώ οι κίνδυνοι είναι πραγματικοί, οι εκτιμήσεις για τις συνέπειες του είναι διαφορετικές και κυμαίνονται από την απώλεια θέσεων εργασίας για ειδικευμένους και ανειδίκευτους εργαζομένους, μέχρι την πιθανότητα δημιουργία θέσεων εργασίας λόγω της συμπληρωματικότητας ανθρώπου-μηχανής που αυξάνει την παραγωγικότητα. Το ηλεκτρονικό εμπόριο, για παράδειγμα, μπορεί να προσφέρει ευέλικτες ή μερικής απασχόλησης θέσεις εργασίας για την κάλυψη των ειδικών αναγκών και ικανοποίησης μειονεκτούντων ομάδων για απασχόληση.

Οι ψηφιακές τεχνολογίες μπορούν να δημιουργήσουν νέες επιχειρηματικές ευκαιρίες

- ✓ για υποστήριξη της καινοτομίας
- ✓ για νέες μορφές απασχόλησης
- ✓ για βελτίωση της ποιότητας των προϊόντων και των περιθώρια κέρδους τους
- ✓ οι πλατφόρμες ηλεκτρονικού εμπορίου μπορούν να ανοίξουν νέες αγορές για τα προϊόντα

Τα «κύματα» της ψηφιακής αλλαγής

Τα τελευταία χρόνια, έχει γίνει μεγάλος ο κατάλογος των προηγμένων ψηφιακών τεχνολογιών όπως η τεχνητή νοημοσύνη, τα μεγάλα δεδομένα, το cloud computing,

το διαδίκτυο των πραγμάτων, η ρομποτική, τα αυτοκίνητα χωρίς οδηγό, το blockchain και ο κβαντικός υπολογισμός και είναι διαρκής η μελέτη για δημιουργικές εφαρμογές τους. Όμως είναι δύσκολο προβλέψιμος ο αντίκτυπος που θα έχουν στην οικονομία, στην κοινωνία ή στην ανθρώπινη καθημερινότητα και η έκταση των αλλαγών που θα φέρουν. Παράδειγμα όταν παρουσιάστηκε το 2007 το πρώτο iPhone, δεν ήταν δυνατό να προβλεφθούν οι αλλαγές που θα έφερναν τα smartphones στην ζωή των ανθρώπων ή στο εμπόριο.



Οι digital natives (ψηφιακή γενιά), που γεννήθηκε και μεγάλωσε με τη νέα ψηφιακή τεχνολογία, ξεπερνούν σε αριθμό εκείνους που μεγάλωσαν χωρίς αυτές τις μορφές τεχνολογίας (digital immigrants) οι οποίοι πρέπει να προσαρμοστούν σε αυτήν τη νέα ψηφιακή πραγματικότητα:

- ✓ Τα μέσα κοινωνικής δικτύωσης δίνουν τη δυνατότητα στους οργανισμούς να προσεγγίσουν εκατομμύρια ανθρώπους με νέους τρόπους.
- ✓ Οι φορητές συσκευές βοηθούν τους ανθρώπους να παραμένουν συνεχώς συνδεδεμένοι στο Διαδίκτυο. Αυτή τη στιγμή οι χρήστες των έξυπνων τηλεφώνων ανέρχονται σε πολλά εκατομμύρια.
- ✓ Οι αναλύσεις δεδομένων (big data) προσφέρουν στους οργανισμούς λεπτομερή στοιχεία για τους πελάτες, αναλύσεις των συμπεριφορών τους και κατανόηση της εμπειρίας που έχουν όταν αλληλεπιδρούν με τους οργανισμούς .



- ✓ Το cloud επιτρέπει στις νέες επιχειρήσεις και στους οργανισμούς να αξιοποιούν την τεχνολογία και να αυξάνουν την επιχειρηματική τους αξία ταχύτερα από ποτέ, χρησιμοποιώντας την τεχνολογία cloud και τις ήδη υπάρχουσες υποδομές και λύσεις που παρέχονται από εταιρείες όπως η Amazon, η Microsoft και η Google.

Οι παραπάνω μορφές τεχνολογίας χρησιμοποιούνται όλο και περισσότερο και αποτελούν πλέον κομμάτι της καθημερινότητας.

Ψηφιακά προϊόντα

Ότι δεν έχει υλική υπόσταση και μεταφέρετε και χρησιμοποιείτε σε ψηφιακή μορφή ονομάζεται ψηφιακό. Τα ψηφιακά προϊόντα το περιεχόμενό τους, δημιουργείται διανέμονται και καταναλώνονται ψηφιακά, δηλαδή μέσω υπολογιστών, tablet ή smartphone.

Τα ψηφιακά προϊόντα διατίθενται στην αγορά σε διάφορες μορφές, όπως:

- **ηλεκτρονικά βιβλία.** Τα ebook, ή ψηφιακά βιβλία, είναι ενδιαφέροντα ψηφιακά προϊόντα, που πωλούνται ή παρέχονται δωρεάν με την πώληση άλλων προϊόντων. Η εμπορική τους επιτυχία, όπως και των φυσικών βιβλίων εξαρτάται από το θέμα και το περιεχόμενο που περιλαμβάνουν.

- **διαδικτυακά μαθήματα.** Η εξ αποστάσεως εκπαίδευση έφερε επανάσταση στην αγορά και έγινε ένα από τα πιο δημοφιλή και κερδοφόρα ψηφιακά προϊόντα.
- **διαδικτυακά σεμινάρια (webinars).** Πολλοί εξειδικευμένοι επαγγελματίες έχουν ήδη αρχίσει να αναπτύσσουν διαδικτυακές συναντήσεις στις οποίες παρέχουν συμβουλές για τις οποίες αμοίβονται από όσους συμμετέχουν.



- εφαρμογές και λογισμικό
- podcast. Το podcast είναι επίσης μια μορφή περιεχομένου πολύ δημοφιλής στους χρήστες του διαδικτύου, κυρίως λόγω της ροής μουσικής, όπως το Spotify για παράδειγμα.
- Infographics. Είναι υλικό που συνήθως χρειάζεται ο χρήστης του για μια ολοκληρωμένη παρουσίαση στοιχείων, εμπλουτίζοντας την εμφάνιση και το περιεχόμενό της. Η πώληση τους, γίνεται σαν πακέτο από διαφορετικές μακέτες infographic, στις οποίες ο καταναλωτής μπορεί να κάνει τροποποιήσεις.

Τα ψηφιακά προϊόντα διαφέρουν από τα συμβατικά, υλικά προϊόντα σε παράγοντες όπως:

- **Παραγωγική διαδικασία:** τα ψηφιακά προϊόντα δεν αντιμετωπίζουν τους περιορισμούς παραγωγής των υλικών αγαθών. Η παραγωγή ψηφιακών προϊόντων θεωρείται ευκολότερη διαδικασία, διότι η σύλληψη της ιδέας, ο σχεδιασμός και η κατασκευή μπορούν να γίνουν στον ίδιο χώρο από τους

ίδιους ανθρώπους. Από τη στιγμή που θα παραχθεί το πρωτότυπο, το κόστος αναπαραγωγής είναι πολύ χαμηλό.

Οι εργασίες αποθήκευσης και διανομής των ψηφιακών προϊόντων πραγματοποιούνται επίσης εύκολα, σε αντίθεση με τα συμβατικά προϊόντα.



- Πληροφοριακοί πόροι: η συλλογή πληροφοριών, η παραγωγή γνώσης και η επιχειρηματική καινοτομία είναι ζητήματα πρωταρχικής σημασίας στην παραγωγή ψηφιακών προϊόντων. Η αξία των πληροφοριακών πόρων αναβαθμίζεται σε σχέση με τους υπόλοιπους πόρους μιας επιχείρησης (ανθρώπινοι, οικονομικοί, τεχνολογικοί, υλικοί).
- Σημεία πώλησης: η εξασφάλιση του απαραίτητου χώρου στα σημεία λιανικής πώλησης είναι ένας κρίσιμος παράγοντας για την επιτυχία στην πώληση ενός υλικού αγαθού. Αντίθετα, η πώληση ενός ψηφιακού προϊόντος δεν υπόκειται σε περιορισμούς χώρου, γεωγραφικούς περιορισμούς, πρόσβασης κτλ., αφού τα σημεία πώλησης είναι εικονικά.
- Αφθαρσία: η εκτεταμένη χρήση των ψηφιακών προϊόντων δεν επηρεάζει τη μορφή, τη λειτουργικότητα και την ποιότητά τους. Αυτό φυσικά έχει οικονομική επίπτωση στις συνολικές πωλήσεις ενός τέτοιου προϊόντος, αφού οι καταναλωτές δεν προβαίνουν σε επαναλαμβανόμενες αγορές.



- Αξία προϊόντος: ενώ στην παραδοσιακή οικονομία ισχύει ότι, όσο πιο σπάνιο είναι ένα προϊόν τόσο μεγαλύτερη αξία έχει (π.χ. πολύτιμα μέταλλα, πετρέλαιο), ένα ψηφιακό προϊόν σε αφθονία έχει μεγαλύτερη αξία. Για παράδειγμα, η αξία των προγραμμάτων επεξεργασίας εικόνας Adobe Photoshop είναι πολύ μεγαλύτερη από παρόμοια προγράμματα, επειδή είναι εγκατεστημένα σχεδόν σε όλα τα εργαστήρια γραφιστικής, χρησιμοποιούνται από τους περισσότερους επαγγελματίες του χώρου και είναι πολύ ευκολότερο να βρεθούν εγχειρίδια εκμάθησής τους. Αυτό το χαρακτηριστικό των ψηφιακών προϊόντων αφορά μία ιδιότητα που είναι γνωστή ως εξωτερικότητες δικτύου (network externalities). Πρόκειται για την επίδραση που έχει στο χρήστη ενός προϊόντος ή μιας υπηρεσίας η ύπαρξη άλλων χρηστών που χρησιμοποιούν το ίδιο ή παρεμφερή προϊόντα/υπηρεσίες. Υπάρχουν τόσο θετικές όσο και αρνητικές εξωτερικότητες δικτύου. Θετικές, όταν όσο περισσότεροι χρησιμοποιούν ένα προϊόν τόσο μεγαλύτερη αξία έχει αυτό για τον κάθε χρήστη του (π.χ. κοινωνικά δίκτυα). Αρνητικές, όταν

οι περισσότεροι χρήστες οδηγούν στη μείωση της αξίας του προϊόντος (π.χ. όταν δημιουργείται συμφόρηση από την αυξημένη κίνηση σε ένα δίκτυο υπολογιστών).

- Εξατομίκευση: Το Διαδίκτυο προσφέρει τη δυνατότητα συλλογής των αναγκών και προτιμήσεων των καταναλωτών σε ατομικό επίπεδο και της προσφοράς εξατομικευμένων προϊόντων, τα οποία έχουν αυξημένες πιθανότητες να τύχουν καλύτερης αντιμετώπισης από τους καταναλωτές. Ακόμη και η αποστολή διαφημιστικών μηνυμάτων γίνεται σε εξατομικευμένο επίπεδο.



- Δημιουργία και διανομή υλικού από τους χρήστες: ψηφιακό υλικό, όπως βίντεο, φωτογραφίες, μηνύματα κτλ. παράγεται από τους ίδιους τους χρήστες και αναρτάται σε διάφορους διαδικτυακούς τόπους (π.χ. YouTube, Facebook, MySpace).
- Οι χρήστες συνεισφέρουν ακόμη και στη δημιουργία γνώσης (π.χ. Wikipedia). Αυτό είναι ένα πρωτόγνωρο χαρακτηριστικό των ψηφιακών προϊόντων και της ψηφιακής οικονομίας γενικότερα, καθώς στην παραδοσιακή οικονομία είναι, αν όχι αδύνατο, ή εξαιρετικά σπάνιο ο καταναλωτής-χρήστης να φτιάχνει από μόνος του προϊόντα και να τα μοιράζεται με άλλους καταναλωτές-χρήστες.

- Πώληση άδειας χρήσης αντί του προϊόντος: σε ορισμένες περιπτώσεις πώλησης ψηφιακών προϊόντων, όπως για παράδειγμα στα προγράμματα λογισμικού, ο πελάτης δεν αγοράζει το ίδιο το προϊόν, αλλά την άδεια χρήσης του για μία συγκεκριμένη

Ψηφιακή πειρατεία

Η ψηφιακή πειρατεία είναι η εμπορική διακίνηση ή διανομή ψηφιακού περιεχομένου χωρίς την έγκριση αυτού που διαθέτει πνευματικά δικαιώματα πάνω σε αυτό και αφορά ακόμα και υλικό που δεν είναι ψηφιακής προέλευσης, αλλά διακινείται ψηφιακά. Η ψηφιακή πειρατεία δεν περιορίζεται στην γνωστή από παλαιότερα πειρατεία ταινιών και τραγουδιών ή μουσικών άλμπουμ, αλλά πλέον εμφανίζεται πειρατεία λογισμικού, βιβλίων και εκπαιδευτικού υλικού όπως σημειώσεις διδασκαλίας και βιντεοσκοπημένα μαθήματα.

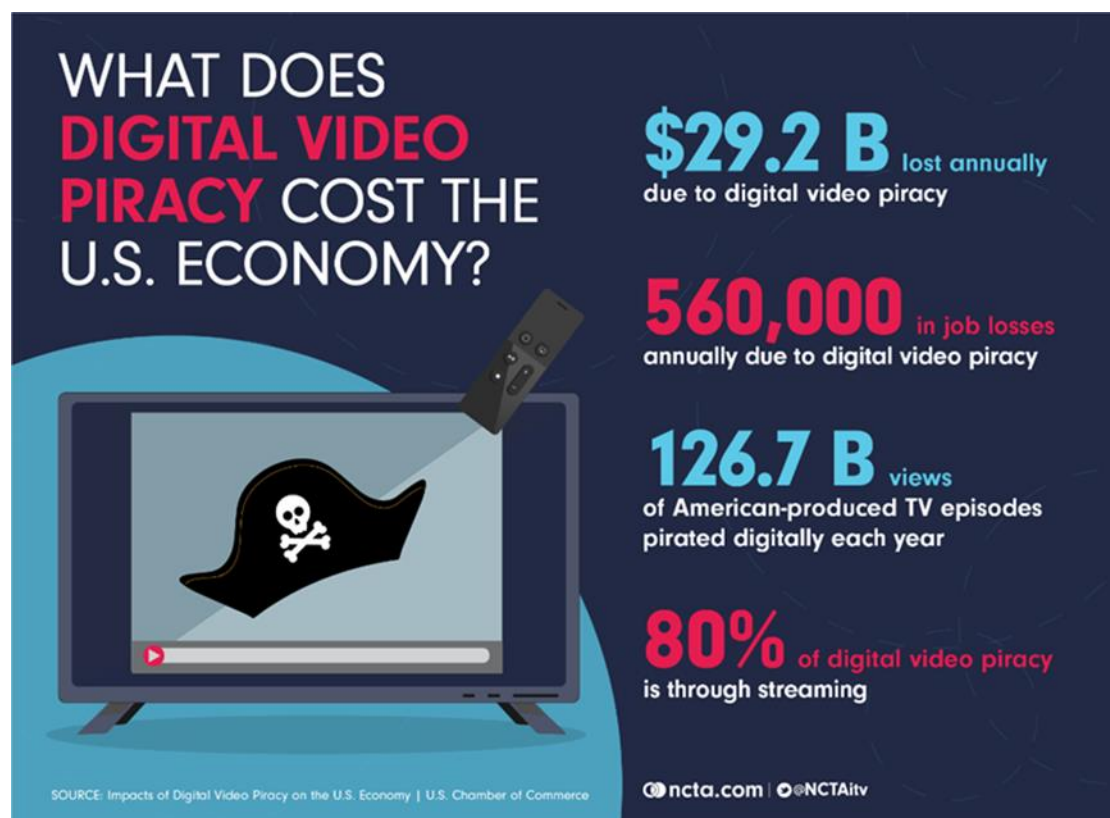
Το ποινικό αδίκημα διαπράττεται όταν προσφέρεται ψηφιακό αντίγραφο του προϊόντος μέσω δωρεάν λήψεων ή πώλησης σε πολύ χαμηλότερη τιμή. Σε αυτές τις περιπτώσεις, τόσο το άτομο που προσφέρει το υλικό όσο και το άτομο που το αγοράζει, γνωρίζοντας ότι εκτελούν παράνομη πράξη είναι νομικά υπεύθυνοι.



Ακόμα και αν δεν θεωρείται ψηφιακή πειρατεία, έχει πολλά παρόμοια στοιχεία η κατανάλωση ταινιών, σειρών και μουσικής σε εφαρμογές για κινητές συσκευές, κανάλια ή πλατφόρμες βίντεο (YouTube, κ.λπ.) και Streaming boxes (TV Box, IPTV) που δεν είναι εξουσιοδοτημένα ή κάτοχοι πνευματικών δικαιωμάτων.

Η πειρατεία μπορεί να επηρεάσει παραγωγούς περιεχομένου διαφορετικών τύπων και μεγεθών και να επηρεάσει όχι μόνο τη δουλειά τους, αλλά και τη δουλειά άλλων ατόμων που ασχολούνται με την παραγωγή, τη διάδοση και τη διανομή του υλικού αλλά έχει αρνητική επίδραση και στην οικονομία μιας χώρας.

Υπολογίστηκε στις Η.Π.Α., ότι ετησίως η ψηφιακή πειρατεία έχει ένα κόστος μεγαλύτερο από 29 δισεκατομμύρια δολάρια για την οικονομία της χώρας και απώλεια 560.000 θέσεων εργασίας.



Η ψηφιακή πειρατεία είναι μια πραγματικότητα που δεν βλάπτει μόνο τους μεγάλους οργανισμούς, αλλά πλήττει την κερδοφορία του μικρού παραγωγού που έχει κάνει μια προσπάθεια να προσφέρει το καλύτερο περιεχόμενο στο κοινό. Συχνά, μικροί παραγωγοί διαθέτουν τα μαθήματα και τα μαθήματά τους στο διαδίκτυο, αλλά υφίστανται μεγάλες απώλειες λόγω πειρατείας και παράνομης διανομής περιεχομένου, που μπορεί να τερματίσει την προσπάθειά τους. Οι πιο συνηθισμένες μέθοδοι που χρησιμοποιούνται από ψηφιακούς παραγωγούς περιεχομένου για να ελαχιστοποιηθεί η ζημιά που προκαλείται από την παράνομη διανομή του υλικού τους, είναι:

- Προστασία βίντεο με κωδικό πρόσβασης. Οι πλατφόρμες χρησιμοποιούν αυτήν τη λειτουργία για παράδειγμα, στην περίπτωση πώλησης περιεχομένου, οι κωδικοί πρόσβασης μπορούν να απελευθερωθούν μόνο στους χρήστες που πλήρωσαν για αυτό το περιεχόμενο, αποτρέποντας άλλα άτομα από την παράνομη παρακολούθηση του. Το ενδιαφέρον είναι ότι ο κωδικός πρόσβασης αλλάζει πάντα μετά την αποστολή του στον χρήστη.
- Κρυπτογράφηση περιεχομένου. Όλο το υλικό που τοποθετείται στον Ιστό θα κωδικοποιηθεί ειδικά για τον ιστότοπο ή το περιβάλλον κατανάλωσης περιεχομένου και στη συνέχεια, εάν κατεβάσουν το περιεχόμενο, δεν θα μπορούν να το αναπαράγουν, λόγω κρυπτογράφησης.



Ο αντίλογος στα επιχειρήματα των υπέρμαχων της ελεύθερης χρήσης της ψηφιακής πληροφορίας, έρχεται από το πλήθος των φορέων των *media* και των πολιτιστικών βιομηχανιών, από εταιρείες παραγωγής, διανομής και εκμετάλλευσης, καθώς και από επαγγελματικά σωματεία και δημιουργούς, οι οποίοι καταγγέλλουν με κάθε τρόπο τον παράνομο χαρακτήρα και τις αρνητικές επιπτώσεις της πειρατείας. Τα επιχειρήματά τους εστιάζονται όχι μόνο στις απώλειες εσόδων, της πολιτιστικής βιομηχανίας, αλλά και στις θέσεις εργασίας που χάνονται, και στην συρρίκνωση της καλλιτεχνικής παραγωγής καθώς και τις λιγότερες ευκαιρίες που δίνονται, τελικά, σε νέους και ανεξάρτητους δημιουργούς για επαγγελματική επιβίωση, ακριβώς εξαιτίας της πειρατείας.

Σύμφωνα με τις θέσεις τους, η πειρατεία στο πνευματικό έργο των δημιουργών δεν μπορεί να μπαίνει στην ίδια βάση συζήτησης όπως οι τεχνολογικές πατέντες και τα 'παράγωγα' καινοτομικά εμπορικά προϊόντα ούτε τα έργα πολιτισμού να θεωρούνται απλώς ως "πληροφορία"

Ο ρόλος του COVID-19 και η επίδραση της ψηφιακής τεχνολογίας στην κοινωνία.

Οι επιπτώσεις της πανδημίας Covid-19 άλλαξαν τα δεδομένα στην επιχειρηματικότητα. Ο φόβος και ο εγκλωβισμός του κόσμου έδωσε μια αύξηση στην αναζήτηση καταναλωτικών αγαθών από τα ηλεκτρονικά καταστήματα. Η υγειονομική κρίση άλλαξε και επαναπροσδιόρισε τις προτεραιότητες των επιχειρήσεων ειδικότερα όσων από αυτές υστερούσαν στην ψηφιοποίηση των εργασιακών και εμπορικών διαδικασιών.



Όλες οι επιχειρήσεις κατανόησαν την ανάγκη ύπαρξης μηχανισμών διαχείρισης κρίσεων και την ανάπτυξη σχετικών δεξιοτήτων, ώστε να εξασφαλίζεται η ετοιμότητα αντιμετώπισης καταστάσεων έκτακτης ανάγκης. Στο μέτρο του δυνατού η βασική προτεραιότητα είναι η συνέχιση λειτουργίας της επιχείρησης. Στην διάρκεια των περιοριστικών μέτρων αντιμετώπισης της πανδημίας, καταγράφηκε απότομη αύξηση των ψηφιακών εμπορικών κινήσεων, αφού για κάποιες χρονικές περιόδους τα φυσικά καταστήματα παρέμεναν κλειστά και υπήρχε διαθέσιμη μόνο η επιλογή της ηλεκτρονικής αγοράς. Οι πρακτικές «κοινωνικής

αποστασιοποίησης» (social distancing) που εφαρμόστηκαν στην πρωτοφανή υγειονομική κρίση είναι πολύ πιθανό να αφήσουν και μετά την πανδημία, ισχυρά ίχνη και στις καταναλωτικές, οικονομικές και κοινωνικές συμπεριφορές, π.χ. αύξηση των ηλεκτρονικών αγορών ακόμη και σε είδη καθημερινής ανάγκης και διατροφής, ανοδική τάση στη χρήση των πρακτικών τηλεκπαίδευσης και τηλεργασίας. Αυτό σε συνδυασμό με τις τεχνολογικές εξελίξεις, θα έχει αποτέλεσμα οι επιχειρήσει να αναζητήσουν νέες δεξιότητες του ανθρώπινου δυναμικού τους.

Εκπαιδευτική Υποενότητα : 1.3 Υλικό και Λογισμικό Ηλεκτρονικών Υπολογιστών
Ο ηλεκτρονικός υπολογιστής είναι μία συσκευή, η λειτουργία της οποίας βασίζεται στην τεχνολογία των ηλεκτρονικών και έχει ως βασικό σκοπό της την επεξεργασία δεδομένων.

Το υπολογιστικό σύστημα (ή υπολογιστής) αποτελείται από δύο μέρη:

- Το Υλικό (Hardware)
- Το Λογισμικό (software)



Το Υλικό (Hardware)

Το Hardware είναι το σύνολο των εξαρτημάτων που είναι απαραίτητα για την λειτουργία του υπολογιστή, ή συνδέονται εξωτερικά με αυτόν.

Στο Hardware διακρίνονται τέσσερα στοιχεία:

- Συσκευές εισόδου: Αυτά είναι τα στοιχεία που συνδέει ο χρήστης, όπως πληκτρολόγιο και ποντίκι.
- Συσκευές εξόδου: είναι τα στοιχεία που μεταφράζουν τα δεδομένα που λαμβάνονται σε μια γλώσσα προσβάσιμη στον χρήστη, όπως η οθόνη και τα ηχεία.
- Εσωτερικά εξαρτήματα: Αυτά είναι τα μέρη που συνδέονται μεταξύ τους για να λειτουργήσει ο υπολογιστής.
- Δευτερεύουσες συσκευές αποθήκευσης: Αυτά είναι τα στοιχεία που είναι υπεύθυνα για τη μόνιμη αποθήκευση δεδομένων στον υπολογιστή.

Μεταξύ των συσκευών του υπολογιστικού συστήματος διακρίνουμε ένα κουτί, που αποτελεί την Κεντρική Μονάδα του υπολογιστικού συστήματος. Σε αυτό βρίσκονται διάφορα εξαρτήματα με πιο σημαντικά την Κεντρική Μονάδα Επεξεργασίας (Κ.Μ.Ε., C.P.U.) και την Κύρια Μνήμη του υπολογιστή. Στην Κ.Μ.Ε. γίνεται, σύμφωνα με τις οδηγίες του χρήστη, η επεξεργασία των δεδομένων που εισάγονται στη μνήμη του υπολογιστή.



Η επικοινωνία του χρήστη με τον υπολογιστή γίνεται με τις συσκευές εισόδου και εξόδου. Συνδέονται με την Κεντρική Μονάδα του υπολογιστή, είτε με καλώδιο (ενσύρματα) είτε χωρίς καλώδιο (ασύρματα).

Για την εισαγωγή των δεδομένων χρησιμοποιούνται οι **συσκευές εισόδου**.

Για να είναι πρακτικά χρήσιμη η επεξεργασία των δεδομένων, πρέπει ο χρήστης να μπορεί να εξάγει τα αποτελέσματά της από τον υπολογιστή. Οι συσκευές στις οποίες αποτυπώνονται τα αποτελέσματα της επεξεργασίας ονομάζονται συσκευές εξόδου.

Τα δεδομένα που εισάγονται και οι πληροφορίες που εξάγονται από τον υπολογιστή, μπορούν να έχουν διάφορες μορφές (π.χ. κείμενο, εικόνα ή ήχο).

Ανάλογα με τη μορφή τους επιλέγεται και η κατάλληλη συσκευή εισόδου ή εξόδου.



❖ Οι σημαντικότερες συσκευές εισόδου και εξόδου ενός ηλεκτρονικού υπολογιστή, είναι:

- Πληκτρολόγιο(Keyboard). Είναι το στοιχείο που επιτρέπει την εισαγωγή δεδομένων και εντολών στον υπολογιστή. Στο πληκτρολόγιο, εκτός από τα πλήκτρα γραμμάτων, αριθμών και συμβόλων υπάρχουν ειδικά πλήκτρα, για πιο άμεσες εντολές στον υπολογιστή π.χ. το πλήκτρο F1, το πλήκτρο Esc κ.ά.

Το πληκτρολόγιο συνδέεται με τον υπολογιστή στην ειδική θύρα τύπου (AT ή PS/2) ή μέσω της θύρας USB, ή πλέον ακόμα και ασύρματα. Συνήθως στα πληκτρολόγια τα πλήκτρα κινούνται σε κατακόρυφο άξονα (motion sensitive), αλλά υπάρχουν και πληκτρολόγια που τα πλήκτρα τους είναι ευαίσθητα στην αφή (touch sensitive).

- Ποντίκι(Mouse). Είναι το στοιχείο που επιτρέπει στο χρήστη να κατευθύνει τον κέρσορα και να κάνει κλικ σε συγκεκριμένες τοποθεσίες για να εκτελέσει μια εργασία. Βοηθάει τον χρήστη να δίνει εντολές στον υπολογιστή, επιτρέποντας και να επιλέγει κάθε τις λειτουργίες που θέλει. Πήρε το όνομά του από το σχήμα του. Συνδέεται στον υπολογιστή μέσω της σειριακής θύρας PS/2, ή μέσω USB. Φυσικά υπάρχουν και τα ασύρματα Mouse.
- Σαρωτής. Ο σαρωτής είναι μια συσκευή που καταγράφει εικόνες. Ως οπτικό και μηχανικό προϊόν περιφερειακού υπολογιστή, ο σαρωτής μπορεί να μετατρέψει εικόνες σε οθόνες υπολογιστή, για επεξεργασία, αποθήκευση και έξοδο ψηφιακής μορφής. Χρησιμοποιώντας μια φωτοευαίσθητη συσκευή, το ανιχνευόμενο οπτικό σήμα μετατρέπεται σε ηλεκτρικό σήμα και, στη συνέχεια, το ηλεκτρικό σήμα μετατρέπεται σε ψηφιακό σήμα μέσω ενός μετατροπέα αναλογικού/ψηφιακού (A/D) και στη συνέχεια μεταδίδεται στον υπολογιστή.
- Το εξάρτημα που ολοκληρώνει τη φωτοηλεκτρική μετατροπή είναι η φωτοευαίσθητη συσκευή, η οποία είναι ο πυρήνας του σαρωτή. Τα χαρακτηριστικά φωτοηλεκτρικής μετατροπής του, όπως φασματική απόκριση, σταθερότητα φωτός, ευαισθησία, θόρυβος κ.λπ., είναι πολύ σημαντικά για τη μετάδοση πληροφοριών εικόνας. Ένας μετατροπέας A/D μετατρέπει μια συνεχώς μεταβαλλόμενη αναλογική ποσότητα σε μια διακριτή ψηφιακή ποσότητα και το κύριο καθήκον του είναι να μετατρέψει την αποκτηθείσα αναλογική τάση σε έναν αριθμό που αντιπροσωπεύει μια τιμή χρώματος ή γκρι. Οι κύριοι δείκτες απόδοσης είναι η ακρίβεια μετατροπής, ο χρόνος μετατροπής, η αντίσταση εισόδου κ.λπ.
- Οθόνη. Είναι ο εξοπλισμός που επιτρέπει την οπτικοποίηση των πληροφοριών που ζητά ο χρήστης. Υπάρχουν διάφοροι τύποι από οθόνες

αυτοί εξαρτώνται από το μέγεθος (15", 17", 19" και 20"), την ανάλυση της δηλαδή τον αριθμό των εικονοστοιχείων (pixel) και την τεχνολογία της π.χ. οι οθόνες αφής.

- Εκτυπωτής. Ο εκτυπωτής χρησιμοποιείται για την εκτύπωση σε σχετικά μέσα. Υπάρχουν πολλοί τύποι εκτυπωτών. Ανάλογα με το αν το στοιχείο εκτύπωσης έχει εντυπωσιακή δράση στο χαρτί, χωρίζεται σε κρουστικούς και μη κρουστικούς εκτυπωτές. Σύμφωνα με τη δομή των χαρακτήρων εκτύπωσης, μπορεί να χωριστεί σε εκτυπωτές χαρακτήρων πλήρους σχήματος και εκτυπωτές χαρακτήρων με κουκκίδες. Οι σειριακοί εκτυπωτές και οι εκτυπωτές γραμμής. Σύμφωνα με την τεχνολογία που χρησιμοποιείται, μπορεί να χωριστεί σε κυλινδρικούς, σφαιρικούς, inkjet, θερμικούς, λέιζερ, ηλεκτροστατικούς, μαγνητικούς, διόδους εκπομπής φωτός και άλλους εκτυπωτές. Η βασική αρχή λειτουργίας ενός εκτυπωτή inkjet είναι να δημιουργεί πρώτα μικρά σταγονίδια μελανιού και, στη συνέχεια, να χρησιμοποιεί την κεφαλή inkjet για να οδηγεί τα μικρά σταγονίδια μελανιού στην καθορισμένη θέση. Όσο μικρότερη είναι η σταγόνα μελάνης, τόσο πιο καθαρή είναι η εκτυπωμένη εικόνα.

Υπάρχουν τρεις δείκτες για τη μέτρηση της ποιότητας ενός εκτυπωτή:

- ✓ ανάλυση εκτύπωσης,
- ✓ ταχύτητα εκτύπωσης και
- ✓ θόρυβος.
- Ηχεία. Τα ηχεία υπολογιστή, μπορούν να χρησιμοποιηθούν και για σύνδεση με άλλες συσκευές αναπαραγωγής όπως κινητά τηλέφωνα. Πολλά ηχεία Bluetooth, ηχεία κάρτας κ.λπ. μπορούν να χρησιμοποιηθούν ως ενωμένα φορητά ηχεία υπολογιστή μετά την προσθήκη μιας εξωτερικής εισόδου. Το ηχείο είναι το τερματικό ολόκληρου του ηχητικού συστήματος και η λειτουργία του είναι να μετατρέπει την ηχητική ενέργεια στην αντίστοιχη ηχητική ενέργεια και να την εκπέμπει στο χώρο. Είναι ένα εξαιρετικά σημαντικό μέρος του ηχητικού συστήματος, επειδή είναι υπεύθυνο για το βασικό καθήκον της μετατροπής των ηλεκτρικών

Εκτός από αυτές τις συσκευές, υπάρχουν και πολλές άλλες συσκευές που συνδέονται με τον υπολογιστή και χρησιμοποιούνται για διάφορες εργασίες, όπως

η κάμερα, η φωτογραφική μηχανή, το μικρόφωνο, το στυλό γραφίδα, το χειριστήριο (joystick), που χρησιμοποιείται στα παιχνίδια κ.ά.



Μνήμη υπολογιστή (RAM) και αποθήκευση δεδομένων

Η μνήμη υπολογιστή ή η μνήμη τυχαίας πρόσβασης (RAM) είναι η βραχυπρόθεσμη αποθήκευση δεδομένων του συστήματος. Αποθηκεύει τις πληροφορίες που χρησιμοποιεί ενεργά ο υπολογιστής, ώστε να είναι δυνατή η γρήγορη πρόσβαση σε αυτές. Όσο περισσότερα προγράμματα εκτελεί το σύστημα, τόσο περισσότερη μνήμη χρειάζεται.

Η μνήμη RAM επιτρέπει στον υπολογιστή να εκτελεί πολλές από τις καθημερινές εργασίες, όπως τη φόρτωση εφαρμογών, την περιήγηση στο Διαδίκτυο, την επεξεργασία ενός υπολογιστικού φύλλου ή τη δοκιμή ενός παιχνιδιού. Η μνήμη στον χρήστη να πραγματοποιεί γρήγορη εναλλαγή μεταξύ αυτών των εργασιών, ενώ θυμάται πού βρισκόταν ο χρήστης σε μια εργασία πριν μεταβεί σε μια άλλη.

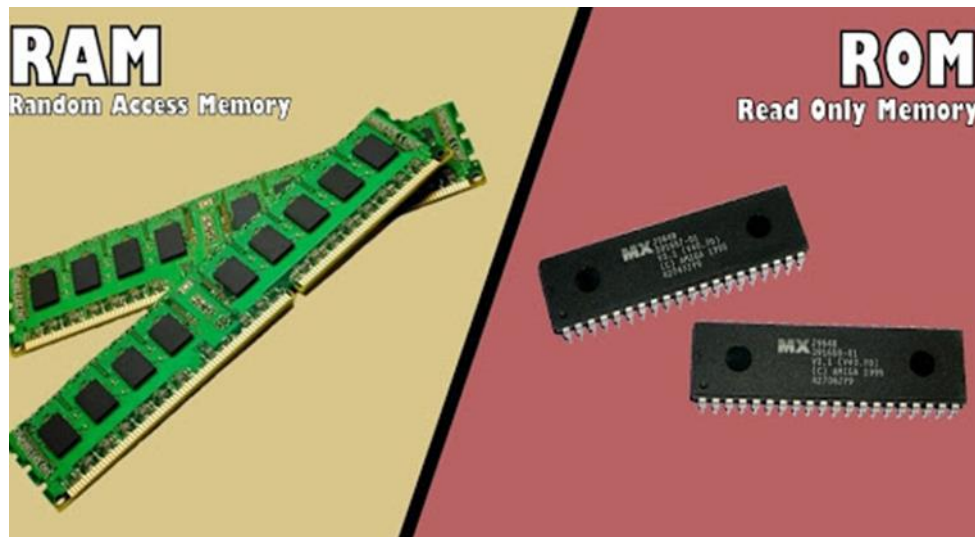
Η μνήμη χρησιμοποιείται για τη φόρτωση και εκτέλεση εφαρμογών όπως ένα πρόγραμμα υπολογιστικών φύλλων, την απόκριση σε εντολές (όπως αλλαγές στο υπολογιστικό φύλλο) ή την εναλλαγή μεταξύ των διαφόρων προγραμμάτων, όπως την μετάβαση από το *υπολογιστικό φύλλο* στο *πρόγραμμα διαχείρισης email*. Η μνήμη χρησιμοποιείται σχεδόν πάντα ενεργά από τον υπολογιστή.

Όταν ολοκληρώνεται ένα έργο, το σχετικό αρχείο αποθηκεύεται από τον υπολογιστή για φύλαξη σε μια μονάδα αποθήκευσης (σκληρός δίσκος).

Εάν το υπολογιστικό σύστημά είναι αργό ή δεν ανταποκρίνεται, η αναβάθμιση της μνήμης είναι ένας από τους ευκολότερους και πιο οικονομικούς τρόπους βελτίωσης της απόδοσης, μόνο όταν αυτό γίνεται επαναλαμβανόμενο πρόβλημα. Όταν ο υπολογιστής «κολλάει», συνήθως αρκεί μια επανεκκίνηση για να αποκατασταθεί το πρόβλημα.

Υπάρχουν τριών ειδών μνήμες σε κάθε υπολογιστή:

- ✓ Random Access Memory (RAM)
- ✓ Read Only Memory (ROM)
- ✓ CMOS



Οι χρήστες εισάγουν τα δεδομένα εισόδου απευθείας σε έναν υπολογιστή τα οποία βραχυχρόνια διατηρούνται στη μνήμη τυχαίας πρόσβασης (RAM). Ο υπολογιστής διαθέτει και τη μνήμη μόνο για ανάγνωση (ROM) όπου τα δεδομένα μπορούν μόνο να διαβαστούν και ελέγχεται και η λειτουργικότητα ενός υπολογιστή.

Η αποθήκευση δεδομένων υπολογιστή είναι μια τεχνολογία που αποτελείται από στοιχεία υπολογιστή και μέσα εγγραφής που χρησιμοποιούνται για τη διατήρηση ψηφιακών δεδομένων. Είναι μια βασική λειτουργία και ένα θεμελιώδες συστατικό των υπολογιστών. Η κεντρική μονάδα επεξεργασίας (CPU) ενός υπολογιστή, είναι αυτή που χειρίζεται τα δεδομένα εκτελώντας υπολογισμούς. Στην πράξη, σχεδόν όλοι οι υπολογιστές χρησιμοποιούν μια ιεραρχία αποθήκευσης, που τοποθετεί γρήγορες αλλά ακριβές μικρές επιλογές αποθήκευσης κοντά στην CPU και πιο αργές αλλά λιγότερο δαπανηρές επιλογές αποθήκευσης και μεγαλύτερες επιλογές πιο μακριά.



Η συνεχής εισαγωγή δεδομένων από τον χρήστη στον υπολογιστή κάνει εμφανή την περιορισμένη δυνατότητα του υπολογιστή σε χωρητικότητα αποθήκευσης και διαθεσιμότητα μνήμης.

Σήμερα, οι χρήστες απαιτούν μεγάλες χωρητικότητες για αποθήκευση δεδομένων που προκύπτουν από τις σύγχρονες υπολογιστικές ανάγκες. Έτσι, για να αποφευχθεί η απώλεια δεδομένων, οι οργανισμοί χρησιμοποιούν εφεδρικές λύσεις για την αποθήκευση δεδομένων.

Στην άμεση αποθήκευση περιοχής, ο χώρος αποθήκευσης συνδέεται απευθείας με την υπολογιστική μηχανή που έχει πρόσβαση σε αυτήν (π.χ. εξωτερικός σκληρός δίσκος)

Στην αποθήκευση που βασίζεται σε δίκτυο επιτρέπει σε περισσότερους από έναν υπολογιστές να έχουν πρόσβαση σε ένα κοινό δικτυακό αποθηκευτικό χώρο,

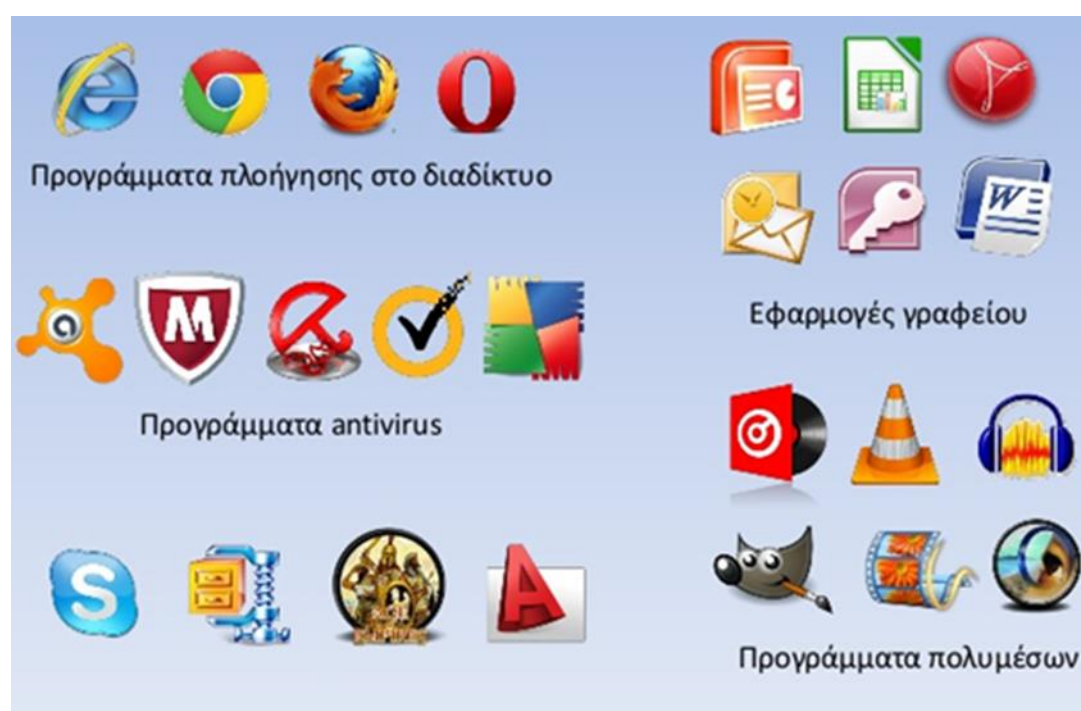
κατάλληλο για δημιουργία αντιγράφων ασφαλείας και μεγαλύτερη προστασία των δεδομένων.

Λογισμικό(Software)

Με τον όρο λογισμικό υπολογιστών (software) ορίζεται η συλλογή από προγράμματα υπολογιστών, διαδικασίες και οδηγίες χρήσης που εκτελούν ορισμένες εργασίες σε ένα υπολογιστικό σύστημα. Το λογισμικό περιλαμβάνει τα προγράμματα, τους ιστότοπους, τα βιντεοπαιχνίδια και άλλα προγράμματα που έχουν αναπτυχθεί σε μια γλώσσα προγραμματισμού όπως για παράδειγμα η C, η C++ ή η Java.

Το λογισμικό είναι κωδικοποιημένο με συγκεκριμένο τρόπο, με τη βοήθεια ενός δυαδικού ψηφιακού συστήματος, ώστε να είναι «κατανοητό» από το υλικό. Μια κατηγοριοποίηση του, είναι:

- ✓ το λογισμικό εφαρμογών, περιλαμβάνει μια μεγάλη ποικιλία προγραμμάτων:



- ✓ γραφείου με προγράμματα επεξεργασίας κειμένου, υπολογιστικών φύλλων, παρουσιάσεων, διαχείρισης emails και βάσεων δεδομένων
- ✓ πλοήγησης στο διαδίκτυο που φορτώνουν και απεικονίζουν σελίδες του internet με εικόνες, βίντεο κ.λπ.



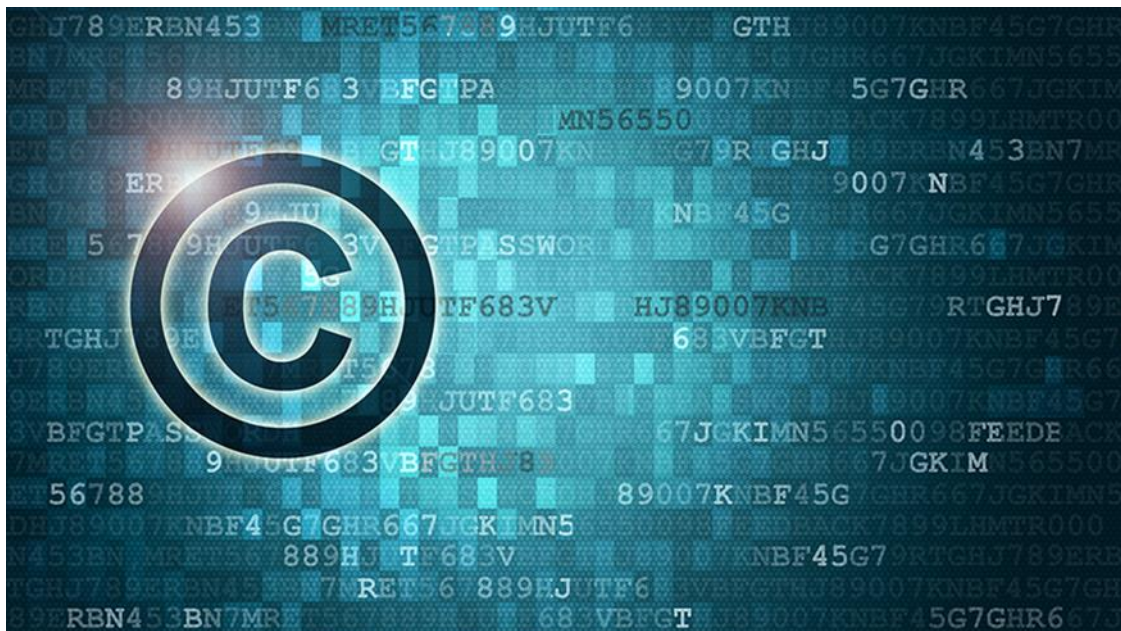
- προστασίας από κακόβουλο λογισμικό (τα γνωστά Antivirus στην απλή τους μορφή ή τα Internet Security με περισσότερες δυνατότητες), για να αποτρέψουν τις επιθέσεις κακόβουλου λογισμικού και να διασφαλίσουν τα προσωπικά τους δεδομένα. Τα εμπορικά προγράμματα αυτού του είδους διαθέτουν πολλές επιπλέον δυνατότητες όπως firewall (για έλεγχο εισερχόμενης και εξερχόμενης κίνησης στο ίντερνετ), γονικό έλεγχο για την προστασία των παιδιών στο διαδίκτυο, διαχείριση της ασφάλειας σε κινητές συσκευές και πολλές άλλες λειτουργίες.
- πολυμέσων και επεξεργασίας εικόνας, ήχου, βίντεο
- εκπαιδευτικά προγράμματα και εγκυκλοπαίδειες
- λειτουργίας επιχειρήσεων. Οι εμπορικές και λογιστικές εφαρμογές σε μία επιχείρηση είναι πανίσχυρα εργαλεία που διαχειρίζονται τις οικονομικές συναλλαγές με πελάτες και προμηθευτές ενώ έχουν άμεση εικόνα στα διάφορα τμήματα της εταιρίας σας όπως (ταμείο, αποθήκη, κλπ.). Το πληροφοριακό σύστημα ERP (Enterprise Resource Planning), έχει στόχο την υποστήριξη όλων των επιχειρησιακών δραστηριοτήτων. Τα συστήματα αυτά ενοποιούν όλες τις σημαντικές δραστηριότητες της επιχείρησης καθώς και όλες τις διαδικασίες σε ένα κεντρικό σύστημα ελέγχου που παρέχει μια συνολική εικόνα για τη λειτουργία της επιχείρησης.
- προγράμματα ψυχαγωγίας (κυρίως παιχνίδια),
- το λογισμικό συστήματος, όπως τα λειτουργικά συστήματα, παρέχει τις αναγκαίες υπηρεσίες του υλικού στο λογισμικό εφαρμογών, είναι προγράμματα που επιτρέπουν στον χρήστη να αλληλεπιδρά με τον υπολογιστή. Παράδειγμα τα Windows, τα οποία είναι ένα λογισμικό επί πληρωμή και το Linux, το οποίο είναι ελεύθερο λογισμικό.
- το ενδιάμεσο λογισμικό (αγγλ. middleware), που ελέγχει και συντονίζει τα καταναμημένα συστήματα, και

- το υλικολογισμικό (firmware) που προγραμματίζει σε χαμηλό επίπεδο το υλικό ενός υπολογιστή ή τα περιφερειακά του.

Εκπαιδευτική Υποενότητα : 1.4 Η ψηφιακή τεχνολογίας και οι επιπτώσεις της

Πνευματικά Δικαιώματα και Άδειες Χρήσης

Τα πνευματικά δικαιώματα είναι τα δικαιώματα που έχει κάθε συγγραφέας ενός πνευματικού έργου πάνω στην καλλιτεχνική, λογοτεχνική ή επιστημονική του δημιουργία, όπως: βιβλία, κείμενα, εικονογραφήσεις, μουσική, φωτογραφίες, λογισμικό, πίνακες ζωγραφικής, γλυπτά, κινηματογραφικά και οπτικοακουστικά έργα, μεταξύ άλλων.



Τα πνευματικά δικαιώματα διακρίνονται σε:

- ηθικά δικαιώματα που διασφαλίζουν ότι ο δημιουργός θα έχει πάντα το έργο που αποδίδεται στο όνομά του, εγγυάται την ακεραιότητα της δημιουργίας και τη δυνατότητα τροποποίησης από τον δημιουργό, πριν ή μετά τη χρήση. Πρόκειται για δικαιώματα μη μεταβιβάσιμα και αναπαλλοτρίωτα.
- δικαιώματα ιδιοκτησίας που αναφέρονται στην απόκτηση οικονομικών πλεονεκτημάτων με την εκμετάλλευση του έργου και μπορούν να εκχωρηθούν ή να παραχωρηθούν σε τρίτους, εφόσον γίνονται απαραίτητα με έγγραφη σύμβαση.

Αυτό σημαίνει ότι το άτομο που αποκτά πνευματικά δικαιώματα θα αποκτά μόνο τα ιδιοκτησιακά δικαιώματα για ένα δεδομένο έργο. Καθώς τα ηθικά δικαιώματα είναι μη μεταβιβάσιμα και αναπαλλοτριώτα, ο αγοραστής δεν μπορεί να μην αναφέρει ποιος είναι ο αρχικός δημιουργός του έργου, ούτε μπορεί να αλλάξει το πνευματικό έργο ή να εμποδίσει τον αρχικό δημιουργό να αντιταχθεί σε οποιοσδήποτε αλλαγές που επηρεάζουν την ακεραιότητα του έργου.



Τα συμβόλαια πνευματικής ιδιοκτησίας έχουν περιοριστική ερμηνεία, δηλαδή δεν χωρούν τεκμήρια. Όλες οι διατάξεις πρέπει να εκφράζονται δεόντως στη σύμβαση. Επομένως, η σύμβαση πρέπει να αναφέρει με πολύ σαφή και πολύ καλά περιγραφόμενο τρόπο ποια είναι τα εμπλεκόμενα δικαιώματα ιδιοκτησίας και ποιες είναι οι δυνατότητες χρήσης του πνευματικού έργου.

Σε περίπτωση παράλειψης, εννοείται ότι τα δικαιώματα ιδιοκτησίας που μεταβιβάστηκαν ήταν μόνο τα απαραίτητα και απαραίτητα για την εκπλήρωση του σκοπού της σύμβασης και όχι όλα καθολικά. Επιπλέον, μόνο τα δικαιώματα ιδιοκτησίας μπορούν να αποτελέσουν αντικείμενο διαπραγμάτευσης. Η σύμβαση που έχει ως αντικείμενο ηθικά δικαιώματα θα θεωρείται άκυρη.



Η συμφωνία άδειας χρήσης, ή εξουσιοδότηση χρήσης, είναι μια συμφωνία. Μέσω της αδειοδότησης, ο παραγωγός ή δημιουργός επιτρέπει σε έναν τρίτο να εκμεταλλευτεί οικονομικά ή/και να χρησιμοποιήσει οικονομικά το έργο του για μια συγκεκριμένη περίοδο, για συγκεκριμένους σκοπούς και για μια συγκεκριμένη μορφή εκμετάλλευσης. Μετά από αυτό το διάστημα, το τρίτο μέρος δεν θα μπορεί πλέον να χρησιμοποιήσει το αδειοδοτημένο έργο. Πρέπει, λοιπόν, να διακόψει αμέσως τη χρήση του, επιστρέφοντάς το στον αρχικό δημιουργό του.

Η σύμβαση άδειας χρήσης πνευματικών δικαιωμάτων μοιάζει λιγότερο με συμφωνία άδειας χρήσης και περισσότερο με σύμβαση μίσθωσης.

Στη μίσθωση ενός ακινήτου, με την πληρωμή των ενοικίων, ο μισθωτής χρησιμοποιεί το ακίνητο για ορισμένο χρονικό διάστημα. Μετά την προθεσμία που έχει συμφωνηθεί μεταξύ των μερών, ο μισθωτής επιστρέφει το ακίνητο στον ιδιοκτήτη του.

Παρόμοια στην άδεια χρήσης πνευματικών δικαιωμάτων, ο χρήστης πληρώνει για ορισμένο χρόνο το δημιουργό, για τη χρήση ενός έργου του. Το έργο αυτό, μετά το διάστημα που συμφωνήθηκε από τα μέρη, θα επιστρέψει στον ιδιοκτήτη του.

Στην περίπτωση συμφωνίας εκχώρησης πνευματικών δικαιωμάτων, σε αντίθεση με την αδειοδότηση, τα δικαιώματα ιδιοκτησίας του δημιουργού επί του έργου μεταβιβάζονται οριστικά.

Έτσι, ένα δεδομένο έργο, όταν μεταβιβαστεί σε τρίτο μέρος, δεν θα επανέλθει πλέον στην κυριότητα του αρχικού δημιουργού. Θα περιέλθει στην ιδιοκτησία του τρίτου που απέκτησε οριστικά το έργο άρα ο δημιουργός ή παραγωγός δεν θα μπορεί πλέον να αποφασίσει πώς θα πραγματοποιηθεί η αποκάλυψη, δημοσίευση, έκθεση, πώληση ή εμπορευματοποίηση. Ωστόσο, πρέπει να σημειωθεί ότι, ακόμη και αν υπάρξει αυτή η οριστική μεταβίβαση του έργου, θα εξακολουθούν να υπάρχουν ορισμένα δικαιώματα του αρχικού δημιουργού επί του έργου, όπως τα ηθικά δικαιώματα και το δικαίωμα μεταπώλησης.



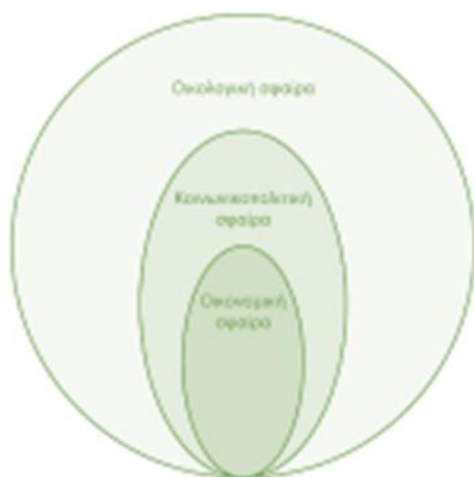
Η σύμβαση εκχώρησης είναι μια συμφωνία μεταβίβασης. Υπό αυτή την έννοια, η συμφωνία εκχώρησης πνευματικών δικαιωμάτων είναι παρόμοια με μια συμφωνία αγοράς και πώλησης. Επιπλέον, η ανάθεση μπορεί να είναι μερική ή ολική και πρέπει πάντα να γίνεται με γραπτή σύμβαση. Θα είναι μερική όταν αφορά μόνο ορισμένα δικαιώματα ιδιοκτησίας και θα είναι ολική όταν αφορά όλα τα δικαιώματα ιδιοκτησίας.

Κάθε εφαρμογή λογισμικού συνοδεύεται από την αντίστοιχη άδεια χρήσης που καθορίζει τη χρήση ή τη διανομή της. Μια τυπική άδεια χρήσης συνήθως συνοδεύεται από έναν κωδικό για κάθε νόμιμο αντίτυπο.

Μία εφαρμογή λογισμικού που έχει αγοραστεί νόμιμα μπορεί όμως τεχνικά να αναπαραχθεί σε περισσότερα αντίτυπα από όσα αναφέρει η άδεια, χωρίς να το γνωρίζει ο δημιουργός.

Τεχνολογία και Αειφορία

Η Βιώσιμη ή Αειφόρος Ανάπτυξη (sustainable development) είναι μια μορφή ανάπτυξης που ικανοποιεί τις ανάγκες της παρούσας γενιάς χωρίς να διακινδυνεύει την δυνατότητα των μελλοντικών γενεών να ικανοποιήσουν τις δικές τους ανάγκες.



Η Αειφόρος Ανάπτυξη είναι ένα σύστημα το οποίο θα πρέπει ταυτόχρονα να είναι:

- Οικονομικά βιώσιμο,
- Κοινωνικά βιώσιμο,
- Περιβαλλοντικά βιώσιμο.

Όταν συνδυάζεται με την τεχνολογία της πληροφορίας, η βιωσιμότητα αναφέρεται στις καλές πρακτικές που υιοθετούνται στον τομέα της πληροφορικής για να ευνοηθεί η επίτευξη αποτελεσμάτων που ευθυγραμμίζονται με τις τρεις συνιστώσες του όρου βιωσιμότητα:

1. αλλαγή συμπεριφοράς στο περιβάλλον των εταιρειών πληροφορικής,
2. χρήση τεχνολογίας για εξοικονόμηση φυσικών πόρων και
3. ανάπτυξη λύσεων που βοηθούν στην κοινωνική και τεχνολογική ανάπτυξη της κοινωνίας.

Εσωτερικά, μια εταιρεία πληροφορικής πρέπει να σκεφτεί λύσεις για την εξοικονόμηση ενέργειας, για την απόρριψη τεχνολογικών αποβλήτων και την επαναχρησιμοποίηση υλικών και χρήση τεχνολογίας. Επιπλέον όλο και περισσότερες εταιρείες πληροφορικής επιδιώκουν να παράγουν τεχνολογικές

λύσεις που βοηθούν από τη μείωση της χρήσης χαρτιού σε δημόσιους και ιδιωτικούς φορείς έως την ανάπτυξη λύσεων που προωθούν την κοινή χρήση οχημάτων, ρούχων, χώρων και παρακολούθησης αέρα.



Για περιβαλλοντικά προβλήματα, η πληροφορική μπορεί να συνεργαστεί τόσο αλυσιδωτά, όταν παράγει λογισμικό που εξαλείφει την ανάγκη για χαρτί, παρεμποδίζοντας κατά συνέπεια την κατανάλωση και την αποψίλωση των δασών για την παραγωγή του, όσο και στη γεωργία, με λύσεις γεωεντοπισμού που παρακολουθούν τις εκπομπές αερίων. και οι οποίες χρησιμοποιούνται όλο και περισσότερο. Η βιωσιμότητα στην πληροφορική βοηθά το περιβάλλον, αυξάνοντας την παραγωγικότητα και μειώνοντας το κόστος τόσο κατά τη λειτουργία όσο και για τους τελικούς δικαιούχους των ανεπτυγμένων συστημάτων.

Προσβασιμότητα από άτομα με ειδικές ανάγκες

Ο υπολογιστής μπορεί να προσφέρει μια σειρά από υπηρεσίες στα άτομα με ειδικές ανάγκες. Η προσβασιμότητα τους διευκολύνεται είτε με προσαρμογές στον εξοπλισμό (hardware) είτε στο λογισμικό (software) ή με συνδυασμό και των δύο. Με λογισμικό μετατρέπει κείμενο σε ηχητικά αποσπάσματα για όσους έχουν προβλήματα όρασης, ή προσθέτει αυτόματους υπότιτλους για αυτούς που έχουν προβλήματα ακοής. Τα προγράμματα αναγνώρισης φωνής επιτρέπουν στον χρήστη

να δίνει φωνητικές εντολές στον υπολογιστή και οι λέξεις εμφανίζονται αυτόματα στην οθόνη.



Οι περισσότερες προσαρμογές στον εξοπλισμό αφορούν την μικρότερων ή μεγαλύτερων σχεδιασμένων πληκτρολογίων και ποντικών, έτσι διαμορφωμένων ώστε να ταιριάζουν στις εκάστοτε ανάγκες. Στα πληκτρολόγια συνηθίζεται η εφαρμογή ενός οδηγού πληκτρολόγησης (σταθερή βάση με τρύπες που δεν αφήνει δυνατότητα ταυτόχρονου πατήματος δυο πλήκτρων) ενώ για τους μονόχειρες μπορεί να διαθέτουν λιγότερα πλήκτρα που με τον κατάλληλο συνδυασμό δημιουργούν ολόκληρες λέξεις. Τα ειδικά ποντίκια με τη στρογγυλή λαβή στο πάνω μέρος τους εξυπηρετούν άτομα με μειωμένη κινητικότητα στα χέρια και τα δάκτυλα, ενώ υπάρχουν ποντίκια που λειτουργούν με μια απλή κίνηση των ματιών ή του κεφαλιού.

Απεριόριστη πρόσβαση στην πληροφορία.

Από την αρχαιότητα, η κατοχή πληροφοριών θεωρούνταν προνόμιο που διέθεταν λίγοι και υπήρξαν πολλά παραδείγματα που κοινωνικές ελίτ διαφόρων εποχών εμπόδιζαν με κάθε τρόπο την πρόσβαση των πολλών στην πληροφορία και στην γνώση.

Η «κοινωνία της πληροφορίας» σαν όρος, εμφανίστηκε για πρώτη φορά στην Ιαπωνία την δεκαετία 1960. Η μετάβαση από τη βιομηχανική κοινωνία στην κοινωνία της πληροφορίας, συνδυασμένη με τις τεχνολογικές εξελίξεις έφερε αλλαγές στις μεθόδους παραγωγής, στην οικονομία, στην επικοινωνία, στην κοινωνία και άλλαξε δραματικά τον τρόπο ζωής των ανθρώπων



Μεγάλο μέρος της ανθρώπινης γνώσης πια είναι συσσωρευμένο αλλά και προσβάσιμο μέσω του Διαδικτύου και συνεχίζει να ανανεώνετε και να εμπλουτίζετε καθημερινά.

Στον σύγχρονο κόσμο της πληροφορίας, καθένας αποκτά πρόσβαση σε πληροφορίες και γνώσεις. Τα σύνορα και οι αποστάσεις έχουν καταργηθεί μέσω του Διαδικτύου και των εφαρμογών λογισμικού και κάθε επικοινωνία είναι δυνατή χωρίς περιορισμούς.

Με δύο υπολογιστές ή smartphones και σύνδεση με το Διαδίκτυο, καθένας μπορεί να μιλήσει και να δει, κάποιον άλλον άνθρωπο, όπου και αν βρίσκεται αυτός.

Ηλεκτρονική Διακυβέρνηση

Η τεχνολογική πρόοδος θα επηρεάσει την δημόσια διοίκηση, προς το καλύτερο. Ο πολίτης θα μπορεί να έχει πρόσβαση σε ηλεκτρονικές υπηρεσίες εξυπηρέτησης πολιτών και αυτό έχει ως αποτέλεσμα την άμεση εξυπηρέτηση του χωρίς να χάνει χρόνο σε γραφειοκρατικές διαδικασίες.

Τεχνολογοφοβία και Ψηφιακό Χάσμα

Υπάρχει μια γενιά (digital natives) που έχει μεγαλώσει μέσα στις τεχνολογικές εξελίξεις με αποτέλεσμα να θεωρεί δεδομένες και αυτονόητες τις αλλαγές που

δημιουργούνται από την χρήση της τεχνολογίας και του διαδικτύου. Όμως, προηγούμενες γενιές (digital immigrants) που δεν γεννήθηκαν στον ψηφιακό κόσμο παρουσιάζουν αρκετές φορές πρόβλημα προσαρμογής.



Η πρόσβαση στην τεχνολογία της πληροφορίας, αν και θεωρητικά είναι ελεύθερη από όλους, στην πράξη παρεμποδίζεται σε μεγαλύτερο ή μικρότερο βαθμό από διάφορα στοιχεία που δημιουργήσαν ομαδοποιήσεις χρηστών και μη χρηστών ανάλογα με την οικονομική και κοινωνική κατάσταση, την χώρα, τις πολιτικές αποφάσεις κ.λπ. Έτσι προέκυψε όρος ή ψηφιακό χάσμα ή ψηφιακός αποκλεισμός σχετικά με την ανισότητα πρόσβασης στην πληροφορία που δημιουργεί πρόβλημα στην ικανότητα του ανθρώπου να αποκτά γνώση.

Λογοκρισία και παραπληροφόρηση.

Με την χρήση του Παγκόσμιου Ιστού είναι πολύ εύκολο για όλους να πουν, να γράψουν και να καταθέσουν την άποψη και να την δημοσιοποιήσουν στο Διαδίκτυο. Το ανεξέλεγκτο περιεχόμενο στο διαδίκτυο, ειδικά διαχειριζόμενο από κακόβουλους χρήστες, έχει σαν αποτέλεσμα την διασπορά ψευδών ειδήσεων και σήμερα, περισσότερο από ποτέ στο παρελθόν, η παραπληροφόρηση να είναι πολύ εύκολη και αποτελεί όπλο σε όποιους επιθυμούν να χειραγωγήσουν το κοινό, ακόμα και σε παγκόσμιο επίπεδο.

Στο άλλο άκρο, η ίδια ελευθερία έκφρασης στο Διαδίκτυο αποτελεί πρόβλημα για αυταρχικά καθεστώτα, που επιδιώκοντας τον απόλυτο έλεγχο στην πληροφορία, συχνά εφαρμόζουν πολιτικές περιορισμού της πρόσβασης σε αυτό.

Ηλεκτρονικό έγκλημα

Οι παρανομίες που προέκυψαν από την ανάπτυξη των τεχνολογιών πληροφορικής μπορούν να χωριστούν σε διαδικτυακά και σε οικονομικά ηλεκτρονικά εγκλήματα.

Σήμερα, τα πλέον διαδεδομένα ηλεκτρονικά εγκλήματα είναι:

- η κλοπή συγκεκριμένων πληροφοριών που αφορούν σχέδια νέων προϊόντων, έρευνες αγοράς, καταλόγους πελατών με αγοραστική δύναμη κ.α. που χρησιμοποιούνται για την απόκτηση ανταγωνιστικού πλεονεκτήματος στην αγορά
- η παράνομη απόκτηση και εκμετάλλευση προσωπικών δεδομένων που μπορεί να δημιουργήσει προβλήματα στην ιδιωτική ζωή του πολίτη. Οι κίνδυνοι αυτοί αυξάνονται με τις νέες δυνατότητες ταχύτατης μεταφοράς πληροφοριών παγκοσμίως μέσω του διαδικτύου.



Τα οικονομικά εγκλήματα αφορούν:

- Απάτες μέσω ηλεκτρονικών υπολογιστών και διαδικτύου: παραποίηση λογιστικών λογαριασμών, την κατασκοπία επιχειρήσεων μέσω της υποκλοπής δεδομένων, παρακολούθηση και υποκλοπή τηλεφωνικών επικοινωνιών, παραποίηση ηλεκτρονικών πληρωμών.

- Ηλεκτρονική μετάλλαξη παραδοσιακών εγκλημάτων: ξέπλυμα χρήματος, κλοπή ηλεκτρονικών υπηρεσιών, λογιστική εξαπάτηση, εκβιασμός επιχειρήσεων.
- Πειρατεία λογισμικού: αφορά την κλοπή πνευματικής ιδιοκτησίας, την αντιγραφή ή τροποποίηση ψηφιακών δεδομένων και την μη εξουσιοδοτημένη χρήση λογισμικού.

Σύνοψη Ενότητας

Αναλύσαμε τι είναι ο υπολογιστής και τι είναι η πληροφορική. Την εξετάσαμε και την αναλύουμε σαν μια νέα επιστήμη.

Είδαμε ποιες είναι οι διαδικασίες του υπολογιστή και από τι απαρτίζεται.

Ποιες είναι οι εργασίες και η χρησιμότητα του, και ποιες είναι αναλυτικά οι λειτουργίες που γίνονται για να πάρουν οι χρήστες του τα επιθήματα αποτελέσματα.

Τέλος αναφερθήκαμε πως οι τεχνολογικές εξελίξεις και οι υπολογιστές έχουν επηρεάσει και έχουν διαφοροποιήσει την καθημερινότητα επίσης.

Ενότητα 1 ερωτήσεις

A. Τι είναι η πληροφορική;

1. Είναι η τεχνολογική εξέλιξη.
2. Είναι η ενημέρωση δεδομένων και αποθήκευση αυτών .
3. Είναι μια σύγχρονη επιστήμη με αρωγό την τεχνολογία.

B. Επιλέξτε ποια είναι η σωστή απάντηση στο τι είναι hardware;

1. Το Hardware είναι το υλικό μέρος του υπολογιστή.
2. Το Hardware είναι το λογισμικό μέρος του υπολογιστή.

Γ. Τι είναι η ψηφιακή τεχνολογία ;

1. Η αποθήκευση δεδομένων σε ψηφιακή μορφή.
2. Η αναμετάδοση δεδομένων σε ψηφιακή μορφή.
3. Η αναζήτηση και η τροποποίηση δεδομένων σε ψηφιακή μορφή.

Δ. Τι είναι Data;

1. Η χωρητικότητα
2. Τα δεδομένα

3. Τα αρχεία

Ε. Τι είναι Πληροφορία - Information; Επιλέξτε την σωστή απάντηση

1. Είναι τα δεδομένα που έχουν δεχτεί τις αλλαγές και τις διορθώσεις
2. Είναι ότι πληροφορία περνάει από το διαδίκτυο

Ζ. Επιλέξτε την σωστή ορολογία στα ελληνικά και στα αγγλικά της μονάδας αποθήκευσης δεδομένων στο υπολογιστή.

1. μπάιτ (byte) – στα ελληνικά μικροσκοπικό τμήμα, κομματάκι.
2. Δυφίο (ΔΥαδικό ψηΦΙΟ) - μπάιτ (byte).
3. Δυφίο (ΔΥαδικό ψηΦΙΟ)- bit (Binary digit)

Η. Ένα χαρακτήρα, δηλαδή ,ένα γράμμα, ένας αριθμός ή κάποιο άλλο σύμβολο ποσά δυαδικά ψηφιά είναι;

1. 9 bits
2. 2 bits
3. 8 bits

Θ. Ένα Kilobyte (Kb) από ποσά byte αποτελείται

1. 222-200 byte
2. 1024 bytes.
3. 2000-2030 bytes

Ι. Τα 1024 Mbytes είναι:

1. 1 Gigabyte (Gb)
2. 1 Terabyte
3. 1 Kilobyte

Κ. Τι είναι σχεδόν διπλάσιο από τα δεδομένων ενός CD-ROM;

1. Terabyte
2. Megabyte
3. Gigabyte

Άσκηση

Γράψτε την λέξης **WORD** Κωδικοποιημένα πχ **W=010101 O= 010011**

Βιβλιογραφία

Μπώκος Γεώργιος (2002) Τεχνολογία και Πληροφόρηση: Από τη διαχείριση του βιβλίου στη διαχείριση της γνώσης

David W. Wang (2020) Software Defined-WAN for the Digital Age : A Bold Transition to Next Generation Networking

Βραχάτης Μιχαήλ, Παπαδάκης (1995) Μικροϋπολογιστές

Μπώκος Γεώργιος (2020) Εισαγωγή στην Επιστήμη της Πληροφόρησης, 2η Έκδοση

Κουμπούλης Φώτης, Μέρτζιος Βασίλειος (2002) Εισαγωγή στη Ρομποτική

Laudon Kenneth, Carol Guercio Traver (2022) Ηλεκτρονικό Εμπόριο: Επιχειρήσεις, Τεχνολογία, Κοινωνία, 16η έκδοση

Προσδοκώμενα Ενότητα : 2 Υλικό και Δίκτυα Ηλεκτρονικών Υπολογιστών

Σκοπός

Η παροχή θεμελιωδών και εισαγωγικών γνώσεων για την επιστήμη της Πληροφορικής, έτσι ώστε ο εκπαιδευόμενος όχι μόνο να γνωρίσει και να κατανοεί τη δομή των ηλεκτρονικών υπολογιστών και των δικτύων, αλλά και την λειτουργία του διαδικτύου, επίσης θέματα πρωτοκόλλων επικοινωνίας και ασφαλείας στην πληροφορική.

Μαθησιακά Αποτελέσματα

Ο εκπαιδευόμενος θα μπορεί να αναγνωρίζει τα δομικά στοιχεία των υπολογιστών και των δικτύων αυτών. Να γνωρίζει εκτός από τις βασικές τους λειτουργίες και τον διαμερισμό των πόρων του συστήματος, τον τρόπο ανταλλαγής πληροφοριών κάθε μορφής (προγράμματα, αρχεία, δεδομένα). Έτσι θα έχει αποκτήσει τις απαραίτητες γνώσεις για τις επικοινωνίες και τα συστήματα επικοινωνιών

Έννοιες - Κλειδιά

- Κεντρική Μονάδα Επεξεργασίας/ΚΜΕ,

Θα μπορούσαμε να την χαρακτηρίσουμε σαν τον εγκέφαλο του υπολογιστή, γιατί εκτελεί τις αριθμητικές και λογικές πράξεις που περιγράφει το λογισμικό.

➤ ROM-BIOS (Read Only Memory)

Η προγραμματισμένη από τον κατασκευαστή μνήμη η οποία είναι με βασικές παραμέτρους λειτουργίας του υπολογιστή και δεν μπορεί να αλλάξει ο χρήστης

➤ RAM (Random Access Memory)

Η κύρια μνήμη του υπολογιστή, στην οποία αποθηκεύονται τα δεδομένα του χρήστη, τα λογισμικά λειτουργικού συστήματος και εφαρμογών, καθώς όλα τα αποτελέσματα από την επεξεργασία των δεδομένων.

➤ Δίκτυο υπολογιστών

Είναι ένα σύστημα από υπολογιστές διασυνδεδεμένους ή αυτονόμους, ή μη αυτόνομους που μπορούν να επικοινωνήσουν μεταξύ τους

➤ Επικοινωνιακός δίαυλος

Είναι ένα μέσο μεταφοράς δεδομένων ανάμεσα σε δύο υπολογιστικά συστήματα, ενσύρματο ή ασύρματο

➤ Πρωτόκολλο

Το σύνολο από συμβάσεις που ορίζουν το πώς ανταλλάσσουν μεταξύ τους δεδομένα οι υπολογιστές του δικτύου ονομάζουμε πρωτόκολλο

➤ Ασφάλεια πληροφοριακών συστημάτων, ασφάλεια υπολογιστικών συστημάτων ή ασφάλεια υπολογιστών

Είναι ένα τμήμα της πληροφορικής, ειδικότερα του κλάδου των υπολογιστικών συστημάτων, που ασχολείται με την προστασία των υπολογιστών, των δικτύων που τους διασυνδέουν και των δεδομένων σε αυτά τα συστήματα, αποτρέποντας τη μη εξουσιοδοτημένη πρόσβαση ή χρήση τους.

Εκπαιδευτική Υποενότητα : 2.1 Δεδομένα – Πληροφορίες

Η επανάσταση των πληροφοριών

Προτείνει μια ριζικά καινούρια ποριά όσον αφορά την τεχνολογία πληροφοριών άλλα και τον τρόπο που θα χρησιμοποιηθεί προκειμένου να θέσει τα συστήματα υπολογιστών στην υπηρεσία των ανθρώπων και όχι το αντίστροφο.

Η Αλλαγή έχει καθυστερήσει καθώς τα άτομα και οι οργανισμοί σε όλο τον κόσμο παλεύουν να εκμεταλλευτούν τις δυνατότητες του Web, του Διαδικτύου να μάθουν όλες τις νέες συσκευές που έχουν κατακλίσει την καθημερινότητα.

Οι άνθρωποι και οι υπολογιστές χρησιμοποιούν διαφορετικά γλώσσες και σήματα για επικοινωνία.

Μεταφράζοντας ένα τραγούδι μπορεί να αποτυπωθεί στη γραφική παράστασή του κάθε τόνος έχει μία αριθμητική τιμή που μπορεί να μετατραπεί σε μία ομάδα δυαδικών ψηφίων 0 και 1, στην ψηφιακή γλώσσα που χρησιμοποιεί ο υπολογιστής.

(Στα κυκλώματα της μνήμης του υπολογιστή, τα ψηφία 0 ταξιδεύουν ως ηλεκτρικά σήματα χαμηλής τάσης κλείνοντας ειδικούς διακόπτες (τρανζίστορ), ενώ τα ψηφία 1, σε υψηλότερη τάση, ανοίγουν τους ειδικούς διακόπτες.)

Λογισμικό για γενικές υπηρεσίες

Αυτή είναι μια αλήθεια για τον υπολογιστή, γιατί λειτουργεί τόσο έξυπνα όσο του επιτρέπει το διαθέσιμο λογισμικό του.

Μερικά προγράμματα ελέγχουν βασικές λειτουργικές εργασίες, όπως η επεξεργασία και η αποθήκευση δεδομένων ή ο προσδιορισμός σφαλμάτων.

Ενώ άλλα προγράμματα εκτελούν πολλές ειδικές απαιτήσεις χρηστών, π.χ. επικοινωνία με άλλους χρήστες, επεξεργασία κειμένου, δημιουργία εικόνων, οργάνωση δεδομένων, παίξιμο παιχνιδιών κτλ.

Χρήση διαλογικών πολυμέσων

Γίνεται διάκριση μεταξύ μονάδων ανάγνωσης που διαβάζουν από οπτικούς δίσκους και μονάδων ανάγνωσης και εγγραφής, τόσο μεμονωμένων όσο και πολλαπλών (+R, -R, +RW, -RW).

Τα τελευταία παρέχουν επίσης τη δυνατότητα οπτικής (λείζερ) εγγραφής δεδομένων σε κενά CD. Οι συσκευές εγγραφής CD μπορούν να είναι δύο τύπων: CD-R (Compact Disc Recordable), που σας επιτρέπει να κάνετε εγγραφή μία φορά σε δίσκους φορμά CD-R και CD-RW (Compact Disc ReWritable), που σας επιτρέπει να εγγράψετε πολλές φορές (με σβήσιμο το προηγούμενο). Ανάλογα με το σχεδιασμό της θήκης και την αρχή της σύνδεσης με τη μονάδα συστήματος, αυτές οι συσκευές μπορούν να είναι ενσωματωμένες (συνεργάζονται κυρίως με τη διεπαφή ATA) ή εξωτερικές, π.χ. κατασκευασμένο ως ξεχωριστή συσκευή συνδεδεμένη στη μονάδα συστήματος χρησιμοποιώντας διάφορες εξωτερικές διεπαφές (SCSI , USB , λιγότερο συχνά - οποιαδήποτε άλλη). Τα κύρια χαρακτηριστικά των μονάδων δίσκου περιλαμβάνουν ταχύτητα μεταφοράς δεδομένων (ταχύτητα ανάγνωσης / εγγραφής), ποιότητα ανάγνωσης / εγγραφής, μέσος χρόνος πρόσβασης, μέγεθος μνήμης buffer.

Ο ρυθμός μεταφοράς δεδομένων καθορίζεται από την ταχύτητα περιστροφής του δίσκου και την πυκνότητα των δεδομένων που καταγράφονται σε αυτόν. Συνήθως καθορίζεται σε σύγκριση με την τυπική ταχύτητα CD ήχου (CD-DA) περίπου 150 Kb / s: 2x (300 Kb / s), 4x (600 Kb / s), 8x (1200 Kb / s), 50x (7500 Kb/s) κ.λπ. Οι παλιοί δίσκοι χρησιμοποιούσαν σταθερή γραμμική ταχύτητα ανάγνωσης (CLV, Constant Linear Velocity), η οποία απαιτούσε αλλαγή στην ταχύτητα περιστροφής του δίσκου κατά την κίνηση της κεφαλής και αυξημένο χρόνο πρόσβασης λόγω αδράνειας δίσκου. Η ταχύτητα περιστροφής για συσκευές μονής ταχύτητας (δηλαδή 150 Kb / s) ήταν της τάξης των 200-500 rpm και για συσκευές 12 ταχυτήτων ήταν ήδη 2400-6400 rpm, γεγονός που άρχισε να δημιουργεί σοβαρά προβλήματα εξισορρόπησης, καθώς οι μονάδες είναι κακώς κεντραρισμένα. Ως αποτέλεσμα, οι κατασκευαστές έχουν στραφεί στις τεχνολογίες CAV και P-CAV. Το CAV (Constant Angular Velocity) σημαίνει σταθερή γωνιακή ταχύτητα του δίσκου (όπως στους σκληρούς δίσκους) και το P-CAV (Partial CAV) χρησιμοποιεί σταθερή γωνιακή ταχύτητα μόνο στις εξωτερικές ράγες του δίσκου. Ταυτόχρονα, ο προσδιορισμός του αριθμού των ταχυτήτων άρχισε να σημαίνει τη μέγιστη ταχύτητα ανάγνωσης στα εξωτερικά ίχνη του δίσκου. Η εγγραφή δεδομένων σε ένα CD ξεκινά από τα εσωτερικά κομμάτια, επομένως η μέγιστη ταχύτητα δεν επιτυγχάνεται σε δίσκους που δεν είναι εντελώς γεμάτοι. Έτσι, για μια μονάδα δίσκου 34 ταχυτήτων, η ταχύτητα ανάγνωσης μπορεί να κυμαίνεται από 2,8

MB / s σε εσωτερικό έως 5,3 MB / s σε εξωτερικά κομμάτια. Προς το παρόν, η ταχύτητα μετάδοσης κίνησης έχει αυξηθεί σε 70x και υψηλότερη (αντιστοιχεί σε ταχύτητα περιστροφής άνω των 12.000 σ.α.λ.). Ως αποτέλεσμα της εμφάνισης μοντέλων με τη διεπαφή Ultra ATA/33, η ταχύτητα ανάγνωσης και μεταφοράς δεδομένων μέσω του διαύλου ATA αυξήθηκε στα 33,3 MB/s. Το να κυνηγήσετε πολύ γρήγορα μια μονάδα CD-ROM συχνά οδηγεί σε κακή αναγνωσιμότητα δίσκων χαμηλής ποιότητας λόγω προβλημάτων ισορροπίας. Η ταχύτητα εγγραφής των δίσκων CD-RW είναι συνήθως πιο αργή από την ταχύτητα ανάγνωσης και εγγραφής των δίσκων CD-R. Η επισήμανση των επανεγγραφών μονάδων δίσκου προβλέπει τον τύπο εγγραφής 8X / 4X / 32X, όπου η πρώτη ομάδα αριθμών υποδεικνύει τη μέγιστη ταχύτητα εγγραφής δίσκων CD-R (στο παράδειγμά μας - 8X), η δεύτερη - τη μέγιστη ταχύτητα εγγραφής CD- Δίσκοι RW (4X), ο τρίτος - η ταχύτητα ανάγνωσης CD -ROM (32X). Η μέγιστη δυνατή επιλογή είναι 52X/32X/52X. όπου η πρώτη ομάδα αριθμών υποδεικνύει τη μέγιστη ταχύτητα εγγραφής δίσκων CD-R (στο παράδειγμά μας - 8X), η δεύτερη - τη μέγιστη ταχύτητα εγγραφής δίσκων CD-RW (4X), η τρίτη - την ταχύτητα ανάγνωσης των CD-ROM (32X). Η μέγιστη δυνατή επιλογή είναι 52X/32X/52X. Η ποιότητα της ανάγνωσης/εγγραφής καθορίζεται από το ποσοστό σφάλματος (Error Rate) και είναι μια εκτίμηση της πιθανότητας παραμόρφωσης 1 bit δεδομένων κατά την ανάγνωση. Οι συσκευές CD-ROM έχουν τη δυνατότητα να διορθώνουν σφάλματα ανάγνωσης/εγγραφής με περιττά δεδομένα που είναι γραμμένα στο δίσκο.

Ο μέσος χρόνος πρόσβασης είναι ο χρόνος που χρειάζεται η μονάδα δίσκου για να βρει τα απαιτούμενα δεδομένα στο δίσκο. Λόγω των χαρακτηριστικών σχεδιασμού, οι μονάδες CD είναι σημαντικά κατώτερες από τις μονάδες σκληρού δίσκου σε αυτόν τον δείκτη. Στις σύγχρονες μονάδες δίσκου, ο μέσος χρόνος πρόσβασης είναι της τάξης των 80-300 ms και οι καυστήρες είναι συνήθως κατώτεροι σε αυτόν τον δείκτη σε σχέση με τις "κανονικές" μονάδες CD-ROM.

Για να αυξηθεί η ταχύτητα πρόσβασης στα δεδομένα, τα τσιπ RAM είναι ενσωματωμένα σε μονάδες CD-ROM - buffer μνήμη (ή cache). Η βέλτιστη ποσότητα μνήμης buffer καθορίζεται από πολλούς παράγοντες και είναι συνεπής με τον ρυθμό μεταφοράς δεδομένων. Στις σύγχρονες μονάδες δίσκου, η χωρητικότητα της προσωρινής μνήμης είναι συνήθως αρκετά MB.

Ο μέσος χρόνος μεταξύ των αστοχιών καθορίζει τον μέσο χρόνο (σε ώρες) που μπορεί να είναι ενημερωμένος ο δίσκος. Οι σύγχρονοι δίσκοι έχουν σχεδιαστεί για περίπου 100.000 ώρες απρόσκοπτης λειτουργίας.

Οι μονάδες δίσκου μπορεί επίσης να διαφέρουν ως προς τις υποστηριζόμενες μορφές και μεθόδους εγγραφής, αν και τα κοινά μοντέλα από τις αρχές της δεκαετίας του 2000 υποστηρίζουν γενικά όλες τις δημοφιλείς μορφές και μεθόδους. Λόγω των διαφορών στην τεχνολογία εγγραφής, οι δίσκοι CD-R/RW που έχουν εγγραφεί σε σύγχρονες μονάδες δίσκου ενδέχεται να μην είναι αναγνώσιμοι σε ορισμένες παλαιότερες μονάδες.

Η πυκνότητα εγγραφής ή η ποσότητα των δεδομένων που έχουν εγγραφεί σε έναν δίσκο, καθορίζεται από το πρότυπο που χρησιμοποιείται. Το κύριο τρέχον διεθνές πρότυπο που καθορίζει τη μορφή εγγραφής είναι το ISO 9660. Υπάρχει επίσης η εκτεταμένη έκδοσή του - CD ROM XA (eXtended Architecture). Στα μέσα της δεκαετίας του 1990, η Philips και η Sony ανέπτυξαν ένα πρότυπο που ονομάζεται Multimedia CD (MMCD), σχεδιασμένο να εγγράφει έως και 7,4 GB δεδομένων σε ένα CD-ROM μονής όψης, που είναι περίπου 10 φορές περισσότερο από άλλα τρέχοντα πρότυπα.

Το 2000, η Sony άρχισε να εργάζεται σε μια μορφή διπλής πυκνότητας - **DDCD (Double Density CD)**, η οποία έχει μειωμένες παραμέτρους πλάτους και μήκους pit (εσοχές στο κομμάτι που αντιστοιχούν στη μονάδα εγγραφής). Οι μονάδες **DDCD** σάς επιτρέπουν να εγγράφετε τόσο ειδικούς δίσκους διπλής πυκνότητας (DD-R και DD-RW) όσο και παραδοσιακούς δίσκους (CD-R και CD-RW). Ένα σημαντικό μειονέκτημα του DDCD είναι η έλλειψη συμβατότητας προς τα πίσω, δηλ. τη δυνατότητα ανάγνωσης δίσκων DDCD σε συμβατικές μονάδες CD-ROM.

Το **Multilevel Recording Format (MLCD, MultiLevel CD)** είναι μια τεχνολογία εγγραφής CD που ονομάζεται **ML (MultiLevel Recording)** και ο αντίστοιχος τύπος **οπτικού μέσου** είναι το **MultiLevel Optical Storage**. Αυτή η τεχνολογία στοχεύει στην αύξηση της χωρητικότητας της μνήμης με την κωδικοποίηση των εγγεγραμμένων δεδομένων με το σχηματισμό μικροβόθων (pits) στην τροχιά ενός δίσκου μεταβλητού βάθους. Η βάση της τεχνολογίας ML είναι η χρήση ενός νέου τύπου κωδικοποίησης που ονομάζεται "**Pit Depth Modulation**" (**PDM**). Σε αυτή την

περίπτωση, χρησιμοποιούνται οκτώ διαφορετικές διαβαθμίσεις του συντελεστή ανάκλασης. Αυτό σας επιτρέπει να αποθηκεύσετε **τρία bit** πληροφοριών σε ένα κελί αντί για ένα. Ως αποτέλεσμα, η πυκνότητα εγγραφής ανά μονάδα μήκους διαδρομής τριπλασιάζεται χωρίς να αλλάζει το πλάτος της διαδρομής και το ελάχιστο μήκος *pit*. Αντίστοιχα, η ποσότητα των αποθηκευμένων δεδομένων και η ταχύτητα ανάγνωσης αυξάνονται επίσης κατά 3 φορές στην ίδια ταχύτητα ατράκτου κίνησης. Στα τέλη του 2000, για να φέρει την τεχνολογία ML σε εμπορική χρήση, δημιουργήθηκε μια συμμαχία ML, η οποία περιελάμβανε την Calimetrics (ο κύριος προγραμματιστής αυτής της τεχνολογίας), την TDK, τη Mitsubishi, την Plextor και τη Sanyo Semiconductor.

Μορφή εγγραφής πολλαπλών επιπέδων ή πολυστρωματικός δίσκος (Digital Multilayer Disk, DMD) είναι μια τεχνολογία που αναπτύχθηκε από την D Data, σύμφωνα με την οποία εφαρμόζονται έως και έξι επίπεδα δεδομένων σε ένα CD. Οι συσκευές DMD χρησιμοποιούν φθηνά λέιζερ «κόκκινης» δέσμης. Σε αυτή την περίπτωση, η συσκευή λήψης επεξεργάζεται επιλεκτικά την ακτινοβολία του φθορίζοντος υλικού που περιλαμβάνεται στο αντίστοιχο στρώμα. Σε σύγκριση με τις μορφές μπλε λέιζερ, αυτή η τεχνολογία είναι λιγότερο δαπανηρή, τόσο από πλευράς εξοπλισμού όσο και αποθηκευτικών μέσων. Στις αρχές της δεκαετίας του 2000, η χωρητικότητα ενός δίσκου DMD ήταν 15 GB, τα επόμενα χρόνια σχεδιάστηκε να αυξηθεί στα 60 GB.

Κίνηση σε δίκτυα υπολογιστών

Στο Διαδίκτυο, η κίνηση είναι μια ροή δεδομένων πληροφοριών (*traffic σημαίνει "κίνηση" από τα αγγλικά*), που σημαίνει τη μετακίνηση δεδομένων σε μια σύνδεση δικτύου από ένα σημείο σε άλλο σε μια συγκεκριμένη χρονική στιγμή και μετριέται σε πολλαπλάσια ενός byte (αντιπροσωπεύει το μήκος ενός μπλοκ δεδομένων). Με τον ίδιο τρόπο που η κίνηση σε έναν αυτοκινητόδρομο σχετίζεται με τον αριθμό των αυτοκινήτων που οδηγούν στο δρόμο, η κίνηση στο Διαδίκτυο σχετίζεται με τη ροή δεδομένων. Με απλά λόγια, η κίνηση στο Διαδίκτυο είναι μια περιγραφή όλης της μετάδοσης δεδομένων μέσα σε ένα δίκτυο (τοπικό ή παγκόσμιο). Σε περιόδους έντονης κυκλοφορίας, το δίκτυο μπορεί να επιβραδύνει και να υπερφορτωθεί εάν τα

φορτία δεν ταιριάζουν με τη χωρητικότητα του καναλιού (όπως ακριβώς ο αυτοκινητόδρομος δεν φιλοξενεί πάντα όλα τα αυτοκίνητα).

Η κυκλοφορία χωρίζεται σε:

- εξερχόμενες (πληροφορίες που εισέρχονται στο εξωτερικό δίκτυο).
- εισερχόμενες (πληροφορίες που προέρχονται από το εξωτερικό δίκτυο).
- εσωτερικό (μέσα σε ένα συγκεκριμένο δίκτυο, πιο συχνά τοπικό).
- εξωτερικό (εκτός συγκεκριμένου δικτύου, πιο συχνά - κίνηση στο Διαδίκτυο).

Ταχεία και οικονομική μεταφορά πληροφοριών

Το εύρος ζώνης μιας διεπαφής δεδομένων (Ρυθμός μεταφοράς) είναι ο μέγιστος ρυθμός με τον οποίο μπορούν να μεταδοθούν δεδομένα μέσω μιας δεδομένης διεπαφής. Μην συγχέετε το εύρος ζώνης μιας διεπαφής με το εύρος ζώνης των μεμονωμένων συσκευών που είναι συνδεδεμένες σε αυτήν. Πολλές διεπαφές δεν είναι σε θέση να μεταφέρουν δεδομένα όσο το δυνατόν γρηγορότερα λόγω της εγγενούς επιβάρυνσης τους. Κάποιες προσαρμογές έχουν δυνατότητες επεξεργασίας υλικού χωρίς CPU που βελτιώνουν την απόδοση, τη δυνατότητα διαχείρισης και την αξιοπιστία της μεταφοράς δεδομένων σε όλη τη διεπαφή. Για τις διεπαφές που αναφέρονται στον πίνακα, οι εντολές βασίζονται σε μετάδοση μίας θύρας σε λειτουργία ημιαμφίδρομης λειτουργίας.

(Εύρος ζώνης διακριτού (ψηφιακού) καναλιού χωρίς παρεμβολές

$$C = \log m \times Vt,$$

όπου m είναι η βάση του κωδικού σήματος που χρησιμοποιείται στο κανάλι. Ο ρυθμός μεταφοράς πληροφοριών σε ένα διακριτό κανάλι χωρίς θόρυβο (ιδανικό κανάλι) είναι ίσος με τη χωρητικότητά του, όταν οι χαρακτήρες στο κανάλι είναι ανεξάρτητοι και όλοι οι m χαρακτήρες του αλφαβήτου είναι εξίσου πιθανοί (χρησιμοποιούνται εξίσου συχνά). Το Vt είναι ο ρυθμός συμβόλων.

Εύρος ζώνης νευρωνικού δικτύου

Η απόδοση ενός νευρωνικού δικτύου είναι ο αριθμητικός μέσος όρος μεταξύ των όγκων των επεξεργασμένων και των παραγόμενων πληροφοριών από ένα νευρωνικό δίκτυο ανά μονάδα χρόνου. (από Βικιπαίδεια)

Επεξεργασία δεδομένων

Η επεξεργασία δεδομένων είναι μια σειρά από δραστηριότητες που εκτελούνται με τακτοποιημένο τρόπο, οι οποίες θα οδηγήσουν σε ένα είδος διευθέτησης πληροφοριών, όπου αρχικά συλλέγονται πληροφορίες ή δεδομένα, που διέρχονται από έναν οργανισμό όπου, τελικά, ο στόχος θα είναι ο χρήστης ή το σύστημα σκοπεύει να χρησιμοποιήσει. Η αρχική απόκτηση δεδομένων, πληροφοριών, ή η επεξεργασία τους μπορεί να πραγματοποιηθεί, με υπολογιστικές και τεχνολογικές μεθόδους, καθώς και με οποιαδήποτε άλλη μορφή συγγραφής και καταλογογράφησης.

Παράδειγμα: Ένα άλμπουμ αυτοκόλλητων είναι μια μορφή επεξεργασίας δεδομένων, καθώς η εισαγωγή ήταν τυχαία δεδομένα, όπου οργανώθηκαν, σχηματίστηκαν πληροφορίες σχετικές με το επιθυμητό θέμα. Ωστόσο, για καλύτερο ορισμό, οι υπολογιστές χρησιμοποιούνται για την εκτέλεση τέτοιων εργασιών, είναι ένα μηχάνημα που διαθέτει σύστημα συλλογής δεδομένων, επεξεργαστή και εργαλεία για το χειρισμό πληροφοριών.

Ηλεκτρονική Επεξεργασία Δεδομένων (PED)

Ονομάζεται επίσης αυτόματη επεξεργασία δεδομένων (PAD), ή δραστηριότητες ηλεκτρονικής επεξεργασίας δεδομένων (PED) που χρησιμοποιούν υπολογιστές στη διεργασία τους. Ωστόσο, ήταν ακριβώς η έλευση των υπολογιστών που έκανε την επεξεργασία των πληροφοριών τόσο δυναμική που, από τότε, η ορολογία Data processing έγινε δημοφιλής, έτσι ώστε αυτή η ονομασία συνδέεται, προς το παρόν, με την ιδέα της χρήσης υπολογιστών. Είναι η συστηματική επεξεργασία δεδομένων, μέσω ηλεκτρονικών υπολογιστών και άλλων ηλεκτρονικών συσκευών, με στόχο την παραγγελία, ταξινόμηση, ή πραγματοποίηση τυχόν μετασχηματισμών στα δεδομένα, σύμφωνα με ένα προηγουμένως προγραμματισμένο σχέδιο, με στόχο την επίτευξη ενός συγκεκριμένου αποτελέσματος.

Γενικά, μια επεξεργασία πραγματοποιείται σύμφωνα με το παρακάτω σχήμα:

Η είσοδος (εισαγωγή): Αναφέρεται σε κάποια δεδομένα εισόδου της επεξεργασίας, είναι τιμές στις οποίες θα ενεργήσει η διαδικασία. Για παράδειγμα, ένα αρχείο που αποστέλλεται σε συμπιεστή δεδομένων.

Επεξεργασία: Εδώ θα γίνει επεξεργασία των δεδομένων εισόδου για τη δημιουργία ενός συγκεκριμένου αποτελέσματος. Στην περίπτωση της επεξεργασίας ο υπολογιστής αυτή η εργασία την εκτελεί μέσω ενός αλγορίθμου γραμμένου σε μια γλώσσα προγραμματισμού που μεταγλωττίζεται και δημιουργεί τον κώδικα ενός προγράμματος που είναι υπεύθυνο για την επεξεργασία.

Η έξοδος: Είναι απλώς το αποτέλεσμα όλης της επεξεργασίας, σε κάθε επεξεργασία έχουμε δεδομένα που δημιουργούνται ως αποτέλεσμα, αυτές οι έξοδοι μπορούν να εκτυπωθούν στην οθόνη, σε χαρτί, να αποθηκευτούν σε ένα αρχείο ή ακόμη και να χρησιμεύσουν ως είσοδος σε μια άλλη διαδικασία. Ο υπολογιστής εμφανίζει τα αποτελέσματα που λαμβάνονται στην οθόνη.

Πώς λειτουργεί η επεξεργασία δεδομένων;

Η επεξεργασία δεδομένων είναι, γενικά, μια διαδικασία που λαμβάνει χώρα από κοινού μεταξύ υλικού (το οποίο μπορεί να είναι επιτραπέζιος υπολογιστής, smartphone, tablet ή άλλος τύπος διακομιστή) και λογισμικού (λειτουργικό σύστημα ή εφαρμογή που θα εμφανίσει τα δεδομένα).

Η εισαγωγή δεδομένων γίνεται μέσω του υλικού, το οποίο τα συλλέγει και τα διανέμει στον επεξεργαστή. Στη συνέχεια, το υπεύθυνο λογισμικό λαμβάνει, αναλύει, επεξεργάζεται και συμπεραίνει τις πληροφορίες.

Επεξεργασία δεδομένων

Η επεξεργασία δεδομένων είναι μια μέθοδος κατά την οποία ένας μεγάλος όγκος πληροφοριών οργανώνεται, καταλογίζεται και ενοποιείται για χάρη ορισμένων στόχων. Αυτή η διαδικασία υποστηρίζεται από ισχυρούς αλγόριθμους ανάλυσης εγγράφων, οι οποίοι συνδυάζουν Big Data, τεχνητή νοημοσύνη και μηχανική μάθηση. Έτσι, οι αναλυτές μπορούν να κατανοήσουν καλύτερα το περιβάλλον γύρω τους και να λάβουν πιο ισχυρές αποφάσεις.

Με άλλα λόγια, η επεξεργασία δεδομένων είναι οποιαδήποτε ρουτίνα που περιλαμβάνει την ανάλυση μεγάλου όγκου πληροφοριών. Προκειμένου να βοηθηθούν οι άνθρωποι να αποκτήσουν καλές πληροφορίες, αυτή η πρακτική συνήθως ακολουθεί τα ακόλουθα βήματα:

- προσδιορισμός στόχων·
- αναζήτηση πληροφοριών·
- συλλογή και αποθήκευση αρχείων·
- κατηγοριοποίηση των δεδομένων σύμφωνα με τις υπάρχουσες απαιτήσεις.

Οι 4 τύποι επεξεργασίας δεδομένων

Υπάρχουν διάφοροι τύποι επεξεργασίας δεδομένων. Καθένα από αυτά βοηθά τις εταιρείες να είναι πιο αποτελεσματικές στην καθημερινότητά τους. Εδώ είναι τα 4 πιο συνηθισμένα!

1. Επεξεργασία παρτίδων

Σε αυτήν την περίπτωση, οι πληροφορίες αποστέλλονται απευθείας σε έναν κεντρικό διακομιστή, μερικές φορές σε μια υποδομή υπολογιστικού νέφους. Τα αρχεία ομαδοποιούνται σε μια παρτίδα και αποθηκεύονται στο μηχάνημα. Έτσι, μπορούν να αναλυθούν σε μια συγκεκριμένη χρονική στιγμή για να μειωθεί ο αντίκτυπος που έχει η δραστηριότητα στη χρήση της υποδομής από άλλους.

2. Διαδικτυακή επεξεργασία

Σε αυτήν την περίπτωση, τα δεδομένα υποβάλλονται σε επεξεργασία όταν ο χρήστης εκτελεί μια ενέργεια. Τα δεδομένα επικυρώνονται σε διακομιστή και, ανάλογα με το αποτέλεσμα, λαμβάνονται μέτρα. Με αυτό, οι χρήστες μπορούν να πραγματοποιούν πληρωμές, να έχουν πρόσβαση σε ιδιωτικά συστήματα και να εκτελούν άλλες λειτουργίες γρήγορα.

3. Επεξεργασία εκτός σύνδεσης

Σε αυτόν τον τύπο επεξεργασίας δεδομένων, οι πληροφορίες δεν έρχονται σε επαφή με μια υποδομή δικτύου. Τα αρχεία αποθηκεύονται στον εξοπλισμό και

αξιολογούνται τοπικά. Με αυτόν τον τρόπο, οι ενέργειες μπορούν να αναληφθούν σε ένα πιο ελεγχόμενο περιβάλλον.

4. Επεξεργασία σε πραγματικό χρόνο

Γνωρίζετε πότε πρέπει να λαμβάνονται γρήγορα μέτρα με βάση ένα σύνολο παραγόντων; Αυτό συμβαίνει με την επεξεργασία σε πραγματικό χρόνο. Εδώ, το σύστημα λαμβάνει τα δεδομένα και αποφασίζει τι να κάνει το συντομότερο δυνατό.

Εμπειρογνώμονας στην επεξεργασία δεδομένων

Ένας εμπειρογνώμονας στην επεξεργασία δεδομένων θα είναι υπεύθυνος για τον εντοπισμό των απαιτήσεων της ομάδας και τη σύνταξη μιας έκθεσης που θα βασίζεται στην ανάλυση διαφοροποιημένων πληροφοριών. Αυτό το έγγραφο πρέπει να παρουσιάζει έξυπνες και αξιόπιστες πληροφορίες. Έτσι, οι ηγέτες θα είναι σε θέση να λαμβάνουν αξιόπιστες αποφάσεις ικανές να καθοδηγήσουν τον καθένα στον καλύτερο δυνατό δρόμο.

Αυτός ο τύπος μπορεί να είναι χρήσιμος σε τομείς όπως το μάρκετινγκ, η κατασκευή συστημάτων, ακόμη και οι οικονομικές επενδύσεις. *Για να είναι επιτυχημένοι, όσοι σκοπεύουν να εργαστούν σε αυτή τη θέση πρέπει να έχουν εκπαίδευση που περιλαμβάνει:*

- έννοιες στατιστικής
- προγνωστική ανάλυση
- επικοινωνία με επαγγελματίες με πολλαπλά προφίλ.
- τεχνολογίες ανάλυσης δεδομένων.
- προγραμματισμός
- μεγάλα δεδομένα
- τεχνητή νοημοσύνη;
- Business Intelligence.

Αυτά είναι μερικά μόνο από τα σημεία που βοηθούν στην εργασία σε αυτόν τον τομέα. Η εκπαίδευση, όπως μπορούμε να δούμε, είναι πολύπλευρη από τον

προγραμματισμό έως τις προηγμένες έννοιες των μαθηματικών. Άρα όποιος σκοπεύει να εργαστεί ως αναλυτής επεξεργασίας δεδομένων πρέπει να γνωρίζει.

Ωστόσο, η εργασία ως αναλυτής επεξεργασίας δεδομένων είναι δυναμική και μπορεί να φέρει νέες προκλήσεις στην καθημερινότητά σας, υπερβαίνοντας κατά πολύ τις δεξιότητές σας που αποκτήσατε προηγουμένως.

Φροντίστε λοιπόν να λάβετε υπόψη τον τομέα της επεξεργασίας δεδομένων για την καριέρα σας ως επαγγελματίας τεχνολογίας . Τα τελευταία χρόνια, έχει ξεχωρίσει ως μία από τις καλύτερες λύσεις που στοχεύουν στην εξαγωγή προστιθέμενης αξίας από εταιρικά δεδομένα έχουν αποκτήσει μεγάλη σημασία λόγω των δυνατοτήτων τους να κάνουν τις επιχειρήσεις ανταγωνιστικές .

Αρχεία

Οποιοδήποτε λογισμικό ή πληροφορίες που έχουν εγγραφεί στον υπολογιστή μας θα αποθηκευτούν σε μια μονάδα δίσκου, την οποία είδαμε νωρίτερα (HD, δισκέτα, CD, Zip, κ.λπ.). Αυτές οι πληροφορίες μπορούν να καταγραφούν μόνο με έναν τρόπο.

Το Αρχείο είναι απλώς η ονοματολογία που χρησιμοποιούμε για να ορίσουμε τις Καταγεγραμμένες Πληροφορίες . Όταν πληκτρολογούμε ένα κείμενο ή όταν σχεδιάζουμε μια εικόνα στον υπολογιστή, το πρόγραμμα (λογισμικό) που είναι υπεύθυνο για τη λειτουργία μας δίνει το δικαίωμα να καταγράψουμε τις πληροφορίες με τις οποίες εργαζόμαστε και μετά την εγγραφή, μετατρέπονται σε αρχείο και τοποθετούνται, κάπου στους δίσκους μας. Αυτή είναι η λειτουργία που ονομάζουμε αποθήκευση. *Είναι καλά! ΕΝΤΑΞΕΙ!* Μέχρι στιγμής, κανένα πρόβλημα, σωστά; Πού ακριβώς όμως γράφεται αυτό το αρχείο σε δίσκους;

Κατά τη στιγμή της εγγραφής, δηλαδή αφού ζητήσουμε την εντολή αποθήκευσης ο υπολογιστής μας ζητά δύο πληροφορίες για να προχωρήσει στην αποθήκευση: Το όνομα του αρχείου και το φάκελο (κατάλογος) όπου θα αποθηκευτεί.

Folder είναι το όνομα που δίνουμε σε ορισμένα «συρτάρια» στο δίσκο. Οι φάκελοι είναι δομές που χωρίζουν το δίσκο σε διάφορα μέρη διαφορετικών μεγεθών, όπως δωμάτια σε ένα σπίτι. Ένας φάκελος μπορεί να περιέχει αρχεία και άλλους φακέλους. Οι φάκελοι ονομάζονται συνήθως Κατάλογοι . Οι φάκελοι είναι "συρτάρια", τα

αρχεία είναι "έγγραφα". Άρα δεν θα υπάρχει ποτέ αρχείο που να έχει φάκελο μέσα. Οι φάκελοι κρατούν αρχεία, όχι το αντίστροφο! Τα αρχεία και οι φάκελοι πρέπει να έχουν όνομα. Το όνομα δίνεται τη στιγμή της δημιουργίας. Ο κανόνας για την ονομασία αρχείων και φακέλων διαφέρει για κάθε λειτουργικό σύστημα. Στα Windows, τα οποία πρόκειται να μελετήσουμε σε αυτό το υλικό, τα ονόματα μπορούν να περιέχουν έως και 256 χαρακτήρες (γράμματα, αριθμούς, λευκό διάστημα, σύμβολα), με εξαίρεση αυτούς τους / \ | > < * ? : " που είναι δεσμευμένα από τα Windows. Τα αρχεία εγγράφονται σε μονάδες δίσκου και παραμένουν εκεί μέχρι να διαγραφούν. Όταν ζητάμε να δουλέψουμε με ένα αρχείο που έχει εγγραφεί προηγουμένως (αυτή η διαδικασία ονομάζεται άνοιγμα του αρχείου), το αρχείο παραμένει στο δίσκο και ένα αντίγραφο των πληροφοριών του τοποθετείται στη μνήμη RAM για να μπορούμε να το επεξεργαστούμε. Όταν ανοίγουμε ένα αρχείο, μπορούμε να το αλλάξουμε αδιακρίτως, αλλά οι αλλαγές θα τεθούν σε ισχύ οριστικά μόνο εάν το αποθηκεύσουμε ξανά. Όταν αποθηκεύουμε ένα αρχείο για δεύτερη φορά και μετά, δεν θα μας ζητάει πλέον όνομα και τοποθεσία, αυτό συμβαίνει μόνο με την πρώτη αποθήκευση.

Με πολύ αντικειμενικό τρόπο, μπορούμε να πούμε ότι ένα αρχείο έχει τουλάχιστον τρία χαρακτηριστικά που το καθορίζουν:

Όνομα: Είναι η ετικέτα αρχείου, όταν επιλέγετε ένα όνομα για το αρχείο, συνιστάται να αποφεύγετε τη χρήση ειδικών χαρακτήρων όπως !@#\$%*+?:><|]. Ορισμένα λειτουργικά συστήματα επιτρέπουν τη χρήση τέτοιων συμβόλων, αλλά όταν αυτά τα αρχεία μεταδίδονται στο Διαδίκτυο ή μέσω pen-drives, δεν γνωρίζουμε ποτέ τον τελικό προορισμό και ούτε καν εάν η συμβατότητά τους θα ικανοποιηθεί σε άλλα λειτουργικά περιβάλλοντα.

Τύπος (επέκταση)

Μέγεθος

Κατάλογοι ή Φάκελοι

Αυτές είναι θέσεις σε μέσα αποθήκευσης δεδομένων, που χρησιμοποιούνται ειδικά για την ΟΡΓΑΝΩΣΗ αρχείων. Δεν καταναλώνουν φυσικό χώρο μόνο για την ύπαρξή τους, το μέγεθός τους είναι το άθροισμα των αρχείων που περιέχονται.

Η δομή οργάνωσης φακέλων σε ένα λειτουργικό σύστημα έχει συλληφθεί από την ιδέα ενός δέντρου αρχείων (root), όπου ο αρχικός φάκελος κάθε συστήματος ονομάζεται root, ή στα συστήματα Windows που προσδιορίζονται ως (C:), όπου έχει εκχωρηθεί κάθε συσκευή αποθήκευσης ένα γράμμα για την ταυτότητά του.

Εκπαιδευτική Υποενότητα : 2.2 Δομικά στοιχεία ηλεκτρονικών υπολογιστών

Οι υπολογιστές χρησιμοποιούνται για τον έλεγχο μεγάλων και μικρών μηχανών που στο παρελθόν ελεγχόταν από ανθρώπους.

Οι περισσότεροι άνθρωποι χρησιμοποιούν έναν προσωπικό υπολογιστή στο σπίτι ή στη δουλειά. Χρησιμοποιούνται για πράγματα όπως υπολογισμούς, ακρόαση μουσικής, ανάγνωση ειδήσεων, γραφή κ.λπ. Οι σύγχρονοι υπολογιστές είναι ηλεκτρονικό υλικό.

Δομικό στοιχείο

Τα δομικά στοιχεία είναι κομμάτια περιεχομένου εγγράφου ή προκαθορισμένα στοιχεία όπως πίνακες, λίστες, κεφαλίδες και πλαίσια κειμένου. Μπορούν να χρησιμοποιηθούν ξανά και ξανά σε αρχεία Word στον υπολογιστή σας.

Υπάρχουν τρία κύρια στοιχεία ενός συστήματος υπολογιστή, το υλικό, Λογισμικό, ανθρώπινου είδους. Οι εσωτερικές συσκευές που συνδέονται με τη μητρική πλακέτα περιλαμβάνουν, Μικροεπεξεργαστή (CPU). Σκληρός δίσκος. Μνήμη τυχαίας πρόσβασης (μονάδες μνήμης). Μονάδα τροφοδοσίας (PSU).

Ποια είναι η βασική αρχιτεκτονική ενός συστήματος υπολογιστών;

Τα κύρια στοιχεία σε ένα τυπικό σύστημα υπολογιστή είναι ο επεξεργαστής, η μνήμη, οι συσκευές εισόδου/εξόδου και τα κανάλια επικοινωνίας που τις συνδέουν. Ο επεξεργαστής είναι η δύναμη του συστήματος, είναι το στοιχείο που εκτελεί ένα πρόγραμμα, με αριθμητικές και λογικές πράξεις σε δεδομένα.

Θεμελιώδη δομικά στοιχεία δεδομένων

Ο Η/Υ ή ένα υπολογιστικό σύστημα, όπως αλλιώς λέγεται, περιλαμβάνει τα ακόλουθα μέρη:

- Κεντρική Μονάδα Επεξεργασίας
- Κύρια Μνήμη
- Περιφερειακή Μνήμη

Βασικό δομικό στοιχείο για την επικοινωνία σε έναν ηλεκτρονικό υπολογιστή.

Τα τρία βασικά δομικά στοιχεία είναι η μεταγωγή πακέτων, το πρωτόκολλο επικοινωνίας, το Πρωτοκόλλο Ελέγχου Μετάδοσης/Πρωτόκολλο Διαδικτύου (TCP/IP) και ο υπολογισμός πελάτη/διακομιστή.

Σε τι χρησιμεύει η μνήμη RAM σε έναν υπολογιστή;

Ο υπολογιστής χρειάζεται μνήμη RAM για *γρήγορη χρήση δεδομένων*. Με απλά λόγια, ο σκοπός της RAM είναι να παρέχει γρήγορη πρόσβαση ανάγνωσης και εγγραφής σε μια συσκευή αποθήκευσης. Ο υπολογιστής σας χρησιμοποιεί μνήμη RAM για τη φόρτωση δεδομένων, επειδή είναι πολύ πιο γρήγορη από την εκτέλεση των ίδιων δεδομένων απευθείας από έναν σκληρό δίσκο.

Πώς λειτουργεί η μνήμη του υπολογιστή

Η μνήμη του υπολογιστή είναι ένας χώρος προσωρινής αποθήκευσης. Περιέχει τα δεδομένα και τις οδηγίες που χρειάζεται η Κεντρική Μονάδα Επεξεργασίας (CPU). Πριν εκτελεστεί ένα πρόγραμμα, φορτώνεται από τον χώρο αποθήκευσης στη μνήμη. Κάθε ρύθμιση ενεργοποίησης/απενεργοποίησης στη μνήμη του υπολογιστή ονομάζεται δυαδικό ψηφίο ή bit.

Περιφερειακή Μνήμη

Στην Περιφερειακή Μνήμη ανήκουν το σύνολο των περιφερικών συσκευών αποθήκευσης πχ. CD – CD-R/RW- USB κτλ.

Ανάλυση Δομικών Υπολογιστή

Κεντρική Μονάδα Επεξεργασίας

Κεντρικός επεξεργαστής μια ηλεκτρονική μονάδα, ή ολοκληρωμένο κύκλωμα που εκτελεί οδηγίες μηχανής (κωδικός προγράμματος), το κύριο μέρος του υλικού του υπολογιστή, ή προγραμματιζόμενος λογικός ελεγκτής . Μερικές φορές αυτό το στοιχείο αναφέρεται απλώς ως επεξεργαστής . Αρχικά, ο όρος κεντρική μονάδα επεξεργασίας περιέγραφε ένα εξειδικευμένο σύστημα στοιχείων που είχαν

σχεδιαστεί για την κατανόηση και εκτέλεση του κώδικα μηχανής των προγραμμάτων ηλεκτρονικών υπολογιστών και όχι μόνο σταθερών λογικών λειτουργιών . Η αρχή της χρήσης του όρου και της συντομογραφίας του σε σχέση με τα υπολογιστικά συστήματα τέθηκε τη δεκαετία του 1960. Η συσκευή, η αρχιτεκτονική και η υλοποίηση των επεξεργαστών έχουν αλλάξει πολλές φορές από τότε. Στα σύγχρονα υπολογιστικά συστήματα, όλες οι λειτουργίες της κεντρικής μονάδας επεξεργασίας εκτελούνται συνήθως από ένα εξαιρετικά ενσωματωμένο μικροκύκλωμα - τον μικροεπεξεργαστή .

Τα κύρια χαρακτηριστικά της CPU είναι, η ταχύτητα , η απόδοση , η κατανάλωση ενέργειας, τα πρότυπα της λιθογραφικής διαδικασίας που χρησιμοποιείται στην παραγωγή (για μικροεπεξεργαστές) και στην αρχιτεκτονική. Οι πρώτες CPU σχεδιάστηκαν ως μοναδικά δομικά στοιχεία και μοναδικά συστήματα υπολογιστών. Αργότερα, από την ακριβή μέθοδο ανάπτυξης επεξεργαστών, που σχεδιάστηκαν για την εκτέλεση ενός μεμονωμένου, ή περισσότερων εξαιρετικά εξειδικευμένων προγραμμάτων. Οι κατασκευαστές υπολογιστών στράφηκαν στη σειριακή παραγωγή τυπικών κατηγοριών συσκευών επεξεργαστή πολλαπλών χρήσεων. Η τάση προς την τυποποίηση των εξαρτημάτων των υπολογιστών ξεκίνησε από την εποχή της ταχείας ανάπτυξης των στοιχείων ημιαγωγών , των κεντρικών υπολογιστών και των μικρών υπολογιστών και με την εμφάνιση των ολοκληρωμένων κυκλωμάτων, έχει γίνει ακόμη πιο δημοφιλής. Η δημιουργία μικροκυκλωμάτων επέτρεψε περαιτέρω αύξηση της πολυπλοκότητας της CPU μειώνοντας ταυτόχρονα το φυσικό τους μέγεθος. Η τυποποίηση και η σμίκρυνση των επεξεργαστών οδήγησαν σε μια βαθιά διείσδυση των ψηφιακών συσκευών που βασίζονται σε αυτές στην καθημερινή ζωή. Οι σύγχρονοι επεξεργαστές μπορούν να βρεθούν όχι μόνο σε συσκευές υψηλής τεχνολογίας όπως υπολογιστές, αλλά και σε αυτοκίνητα , αριθμομηχανές , κινητά τηλέφωνα , ακόμη και σε παιδικά παιχνίδια . Τις περισσότερες φορές αντιπροσωπεύονται από μικροελεγκτές , όπου, εκτός από την υπολογιστική συσκευή, βρίσκονται στο τσιπ πρόσθετα στοιχεία (μνήμη προγράμματος και δεδομένων, διεπαφές, θύρες εισόδου-εξόδου, χρονόμετρα κ.λπ.). Οι σύγχρονες υπολογιστικές δυνατότητες του μικροελεγκτή είναι συγκρίσιμες

με τους επεξεργαστές προσωπικών υπολογιστών πριν από τριάντα χρόνια και τις περισσότερες φορές υπερβαίνουν ακόμη και σημαντικά την απόδοσή τους.

Συσκευών εισόδου και εξόδου

Οι συσκευές εισόδου και εξόδου είναι οι συσκευές που επιτρέπουν την αλληλεπίδραση ενός επεξεργαστή, συνήθως ενός υπολογιστή, με τον άνθρωπο. Είναι επίσης γνωστά ως περιφερειακά και επιτρέπουν την είσοδο και την έξοδο δεδομένων.

Οι συσκευές εισόδου είναι αυτές που εισάγουν πληροφορίες σε έναν υπολογιστή από εξωτερική πηγή. *Παραδείγματα περιλαμβάνουν, πληκτρολόγιο, ποντίκι, μικρόφωνο, σαρωτή και οθόνες αφής.*

Οι συσκευές εξόδου είναι οι συσκευές που χρησιμοποιούνται από έναν υπολογιστή για την επικοινωνία πληροφοριών. Αυτά είναι σε μορφή που μπορεί να χρησιμοποιηθεί για τον χρήστη. *Παραδείγματα περιλαμβάνουν: οθόνες βίντεο, ηχεία, ηχεία, αισθητήρες, γυαλιά εικονικής πραγματικότητας και εκτυπωτές.*

Τι είναι η συσκευή εξόδου;

Οι συσκευές εξόδου παίζουν σημαντικό ρόλο στις πληροφορίες, μετατρέποντας τα δεδομένα στον υπολογιστή σε μια αναγνωρίσιμη γλώσσα για τον δέκτη.

Για παράδειγμα, κάθε φορά που παίζουμε ένα τραγούδι στον υπολογιστή, επεξεργάζεται το αρχείο και κάνει τον ήχο να βγαίνει από τα ηχεία.

Έτσι μπορούμε να ακούσουμε. Ένα άλλο παράδειγμα είναι η χρήση του εκτυπωτή, ο οποίος είναι ικανός να μετατρέψει ένα ψηφιακό αρχείο σε κάτι φυσικό.

Τι είναι η συσκευή εισόδου;

Οι συσκευές εισόδου είναι συσκευές που βοηθούν τον υπολογιστή να αποκτήσει δεδομένα και να τα εισάγει στο υπολογιστικό σύστημα. Μπορούν μόνο να στείλουν δεδομένα, όχι να λάβουν.

Είναι το υλικό που τραβάει δεδομένα μέσω της δράσης του χρήστη και επεξεργάζεται τις πληροφορίες. Παράδειγμα, όταν πληκτρολογείτε σε ένα πληκτρολόγιο, παίρνει τις πληροφορίες που θέλετε να μεταδώσετε και τις στέλνει στον υπολογιστή για να τις δούμε.

Αυτές οι συσκευές πρέπει να έχουν μια λειτουργία στην είσοδο. Δηλαδή, κάτι που τους συνδέει με τον υπολογιστή, όπως καλωδιακή ή ασύρματη σύνδεση.

Η RAM (Random Access Memory, Μνήμη Τυχαίας Προσπέλασης) είναι η μνήμη όπου αποθηκεύονται προσωρινά εντολές και δεδομένα, περιμένοντας την ανάκληση τους για επεξεργασία. Είναι επίσης ο χώρος όπου αποθηκεύονται τα αποτελέσματα της εκτέλεσης των εντολών.

Η RAM δεν παρέχει δυνατότητα μόνιμης αποθήκευσης δεδομένων. Όταν τελειώσει η εκτέλεση, τα δεδομένα μετακινούνται από τη Κ.Μ.Ε πίσω στην RAM και από εκεί αποστέλλονται στην κατάλληλη έξοδο ή προς μόνιμη αποθήκευση σε κάποια δευτερεύουσα μνήμη.

Για την κατασκευή της RAM χρησιμοποιούνται ολοκληρωμένα κυκλώματα ημιαγωγών (chips) τα οποία διατηρούν τα περιεχόμενα τους όταν υπάρχει ηλεκτρική ενέργεια για τη λειτουργία τους.

Για το λόγο αυτό, λέγεται συχνά ότι τα περιεχόμενα της χάνονται αν επέλθει διακοπή ηλεκτρικού ρεύματος και γι' αυτό συνίσταται η τακτική αποθήκευση των περιεχομένων της σε κάποια Δευτερεύουσα ή Περιφερειακή Μνήμη.

Η ROM (Read-Only Memory, Μνήμη Μόνο Ανάγνωσης) είναι μια μνήμη που δεν επιδέχεται αλλαγές. Τα περιεχόμενα της μπορούν να διαβαστούν όχι όμως και να μεταβληθούν παρά μόνο εάν αντικατασταθούν ολοσχερώς τα ίδια τα κυκλώματα των ημιαγωγών (chips) που την αποτελούν.

Η ROM χρησιμοποιείται για καταστάσεις όπου η πληροφορία χρειάζεται να φυλαχθεί χωρίς να δεχθεί μεταβολές. *Η πρωταρχική χρήση της ROM είναι κατά τη διάρκεια της διαδικασίας εκκίνησης τού υπολογιστή.*

Η ROM εμπεριέχει εντολές που υποδεικνύουν στη μηχανή πώς να εκτελέσει τη διαδικασία εκκίνησης ("boot process").

Οι εντολές αυτές συγκεκριμένα είναι POST (Power On Self Test),

Setup (παραμετροποίησης), εντολές οι οποίες ενεργοποιούν εντολές του CMOS BIOS, εντολές που αφορούν τη διαχείριση των περιφερειακών συσκευών του υλικού Boot, εντολές που καλούν το λειτουργικό σύστημα (DOS, OS/2, or Windows)

Η BIOS (Basic Input-Output Service) δίνει τη δυνατότητα ελέγχου της αλληλεπίδρασης του υπολογιστή με τον χρήστη.

Η CMOS (Complimentary Metal Oxide Semiconductor) μνήμη εμπεριέχει δεδομένα ζωτικής σημασίας για το **setup**. Η πρόσβαση σε αυτή γίνεται με την εκκίνηση για να προσδιορισθεί ποιο υλικό έχει ο υπολογιστής, και πως είναι διαμορφωμένο (configured).

Η χρήση αυτής της μνήμης, η οποία φυλάσσει τη διαμόρφωση του συστήματος, καθιστά δυνατή την αλλαγή του υλικού της μηχανής και μαζί την αναγνώριση της αλλαγμένης διαμόρφωσης από τον υπολογιστή. Επομένως, όταν αλλάζουμε τη διαμόρφωση του υλικού πρέπει να αλλάξουμε (manually) το περιεχόμενο της CMOS.

Οι σταθερές τιμές του CMOS Setup χρησιμοποιούνται για τη διαμόρφωση:

Ημερομηνίας και Ώρας

Πληκτρολογίου

Οθόνης

Οδηγού Δισκέττας

Σκληρού Δίσκου, Οδηγού CD-ROM, κλπ

Οι Προγραμματισμένες Μνήμες Μόνο Ανάγνωσης (Programmable Read Only Memory) είναι είδος της Μνήμης στην οποία μέσω ειδικής διαδικασίας, μπορούμε να γράψουμε δεδομένα και εντολές. Η (Primary Cache Memory) είναι μια μνήμη μεγάλης ταχύτητας προσπέλασης και χρησιμοποιείται για την αποθήκευση συχνά χρησιμοποιούμενων εντολών ώστε η Κ.Μ.Ε να μπορεί να έχει μια γρήγορη πρόσβαση, αποφεύγοντας τον σχετικά μικρής ταχύτητας δίαυλο μεταφοράς δεδομένων. Είναι ένα είδος φυσικής μνήμης η οποία προστίθεται σε αυτή της RAM.

Δίαυλοι (Buses)

Μία άλλη σημαντική έννοια είναι αυτή του διαύλου. Οι δίαυλοι χρησιμοποιούνται για την διακίνηση των δεδομένων μεταξύ των βασικών συστατικών του ηλεκτρονικού υπολογιστή. Μπορούν να χωρισθούν σε δύο βασικές κατηγορίες, *κεντρικός δίαυλος* (system bus), ο οποίος συνδέει την Κ.Μ.Ε και την RAM και *διαύλους εισόδου/εξόδου* (I/O buses), που συνδέουν την Κ.Μ.Ε με άλλα συστατικά.

Διαδικασία Εκκίνησης ενός Η/Υ (Boot Process)

Η εκκίνηση ή ο φορτωτής εκκίνησης, είναι μια διαδικασία προετοιμασίας του συστήματος ελέγχου υπολογιστή σε έναν υπολογιστή. Η διαδικασία εκκίνησης μπορεί να είναι "σκληρή εκκίνηση", όπως, διάγνωση υλικού μετά την ενεργοποίηση, μπορεί επίσης να είναι "soft boot", η οποία θα παρακάμψει την εκκίνηση και θα ξεκινήσει αυτόματα. Σε ορισμένα συστήματα, η μνήμη RAM ενδέχεται να μην εκκαθαριστεί κατά τη διαδικασία μαλακής εκκίνησης. Τόσο η μαλακή εκκίνηση όσο και η σκληρή εκκίνηση μπορούν να εκκινήθουν με υλικό, όπως το πάτημα του διακόπτη λειτουργίας. Όταν επιτευχθεί ένα κανονικό και έγκυρο περιβάλλον λειτουργίας, η εκκίνηση έχει ολοκληρωθεί.

Ο φορτωτής εκκίνησης είναι ένα πρόγραμμα υπολογιστή που φορτώνει ένα λειτουργικό σύστημα ή άλλο λογισμικό συστήματος, μετά την ολοκλήρωση του αυτοδιαγνωστικού ελέγχου ενεργοποίησης του υπολογιστή .

Ο φορτωτής εκκίνησης μπορεί να φορτωθεί από τον σκληρό δίσκο στην κύρια μνήμη. Σε ορισμένους παλαιότερους υπολογιστές, ο φορτωτής εκκίνησης μπορεί να φορτωθεί στην κύρια μνήμη από μια διάτρητη κάρτα, μια διάτρητη χαρτοταινία ή μια μαγνητική ταινία. Ο φορτωτής εκκίνησης φορτώνει και εκτελεί το πρόγραμμα που ολοκληρώνει την εκκίνηση. Παρόμοια με τη διαδικασία POST, ο κωδικός της συσκευής εκκίνησης μπορεί να αποθηκευτεί σε μια μόνιμη θέση τράπεζας μνήμης ή μπορεί να είναι ενσωματωμένος.

Στους σύγχρονους υπολογιστές γενικής χρήσης, η διαδικασία εκκίνησης συνήθως περιλαμβάνει έναν αυτοέλεγχο κατά την ενεργοποίηση (POST), την προετοιμασία των περιφερειακών και στη συνέχεια τη φόρτωση του λειτουργικού συστήματος.

Ορισμένα ενσωματωμένα συστήματα εκτελούν εκτελέσιμα προγράμματα που είναι αποθηκευμένα στη ROM απευθείας.

Το Boot στα αγγλικά, η οποία είναι μια συντομογραφία του *bootstrap*, που προέρχεται από τη φράση "Pull oneself up by one's bootstraps", που σημαίνει "τραβήξτε τον εαυτό σας από τα *bootstraps*". Εάν το μεγαλύτερο μέρος του λογισμικού πρέπει να φορτωθεί στον υπολογιστή από άλλα προγράμματα που εκτελούνται ήδη, πρέπει να υπάρχει ένας μηχανισμός για τη φόρτωση του αρχικού λογισμικού στον υπολογιστή.

Περιφερειακές Μονάδες Μνήμης

Η Κύρια Μνήμη ενός ηλεκτρονικού υπολογιστή όπως έχουμε ήδη αναφέρει έχει σχετικά μικρό μέγεθος και χρησιμοποιείται κυρίως για την προσωρινή αποθήκευση των εντολών που εκτελούνται και των δεδομένων πάνω στα οποία ενεργούν οι εντολές αυτές. Γενικά, όμως, ισχύει ότι το περιεχόμενο της Κύριας Μνήμης διατηρείται μόνο σε προσωρινή βάση. Για την αποθήκευση του μεγάλου όγκου δεδομένων που χρειάζεται να επεξεργαστούν και την αποθήκευσή τους σε μόνιμη βάση χρησιμοποιούνται οι βοηθητικές, ή δευτερεύουσες, ή περιφερειακές μονάδες μνήμης.

Η βοηθητική μνήμη διατηρεί το περιεχόμενο της σε μόνιμη βάση.

Τα τεχνικά χαρακτηριστικά των περιφερειακών μονάδων μνήμης είναι

- ✓ Χωρητικότητα Μνήμης
- ✓ Χρόνος Προσπέλασης
- ✓ Ταχύτητα Μεταφοράς
- ✓ Κόστος

Η Χωρητικότητα της μνήμης δηλώνει την ποσότητα των δεδομένων που μπορεί να διατηρηθεί στο μέσο αποθήκευσης. Η βασική μονάδα μέτρησης της χωρητικότητας μνήμης είναι το Bit (Binary Digit) δηλαδή ο χώρος στον οποίο μπορεί να αποθηκευτεί το 0 ή το 1. Συνήθως, όμως, χρησιμοποιείται το Byte (υπενθυμίζουμε ότι 1 Byte = 8Bits) και τα πολλαπλάσια του.

Ο Χρόνος Προσπέλασης της μνήμης συνήθως μετριέται σε χιλιοστά του δευτερολέπτου και αναφέρεται στον χρόνο που απαιτείται για τον εντοπισμό των δεδομένων στη βοηθητική μνήμη.

Η Ταχύτητα Μεταφοράς αναφέρεται στην ποσότητα δεδομένων ανά μονάδα χρόνου που μπορεί να μεταφερθεί από την περιφερειακή μνήμη στην κύρια. Μονάδα μέτρησης της ταχύτητας μεταφοράς είναι το Mbyte/sec.

Το Κόστος της περιφερειακής μνήμης εξαρτάται από το μέσο κατασκευής της περιφερειακής μνήμης και τα τεχνικά της χαρακτηριστικά. Έχει, όμως, παρατηρηθεί ότι το κόστος μνήμης σταδιακά μειώνεται, ενώ οι δυνατότητες των περιφερειακών μνημών αναβαθμίζονται ουσιαστικά.

Σήμερα, μπορεί κανείς να διακρίνει πολλές περιφερειακές μονάδες μνήμης που χρησιμοποιούνται σε υπολογιστικά συστήματα διαφόρων κατηγοριών. Όμως, οι πιο συνηθισμένες περιφερειακές μονάδες μνήμης είναι:

- Μαγνητικός Δίσκος
- Μονάδες Μνήμης USB (Flash USB Memory Sticks)
- Δισκέτες
- Οπτικοί ή Ψηφιακοί Δίσκοι (CDs)
- DVDs
- Έξυπνες Κάρτες



Ο δίσκος είναι το κύριο μέσο αποθήκευσης του υπολογιστή, το οποίο μπορεί να αποθηκεύσει μεγάλη ποσότητα δυαδικών δεδομένων και μπορεί να διατηρήσει τα δεδομένα να μην χαθούν μετά από διακοπή ρεύματος. Ο δίσκος που χρησιμοποιήθηκε στους πρώτους υπολογιστές ήταν μια δισκέτα (Floppy Disk, που

αναφέρεται ως δισκέτα) και ο δίσκος που χρησιμοποιείται συνήθως σήμερα είναι ένας σκληρός δίσκος (Σκληρός δίσκος, που αναφέρεται ως σκληρός δίσκος).

Δομή Δίσκου

Ένας δίσκος (όπως ένας μηχανικός σκληρός δίσκος 1T) σχηματίζεται με τη στοίβαξη. Η επιφάνεια της πλάκας είναι επικαλυμμένη με μαγνητικές ουσίες, οι οποίες χρησιμοποιούνται για την καταγραφή δυαδικών δεδομένων.

Πίστα, τομέας

Κάθε δίσκος χωρίζεται σε κομμάτια και κάθε κομμάτι χωρίζεται σε τομείς. Μεταξύ αυτών, η περιοχή τομέα στην πιο εσωτερική τροχιά είναι η μικρότερη, επομένως η πυκνότητα δεδομένων είναι η μεγαλύτερη.

Κύλινδρος

Κάθε επιφάνεια δίσκου αντιστοιχεί σε μια μαγνητική κεφαλή. Όλες οι μαγνητικές κεφαλές συνδέονται στον ίδιο μαγνητικό βραχίονα, έτσι όλες οι μαγνητικές κεφαλές μπορούν μόνο «να προχωρήσουν και να υποχωρήσουν μαζί». Τα ίχνη στην ίδια σχετική θέση σε όλες τις επιφάνειες του δίσκου σχηματίζουν κυλίνδρους.



Μονάδες Μνήμης USB (Flash USB Memory Sticks)

Το *Universal Serial Bus* (Αγγλικά: *Universal Serial Bus*, συντομογραφία *USB*) είναι ένα πρότυπο σειριακού διαύλου για τη σύνδεση συστημάτων υπολογιστών και εξωτερικών συσκευών και είναι επίσης μια τεχνική προδιαγραφή για διεπαφές εισόδου και εξόδου, η οποία χρησιμοποιείται ευρέως σε προσωπικούς υπολογιστές

και φορητές συσκευές, καθώς και άλλα προϊόντα πληροφόρησης και επικοινωνίας. Επεκτάθηκε και σε άλλους συναφείς τομείς όπως φωτογραφικός εξοπλισμός, ψηφιακή τηλεόραση (set-top boxes), κονσόλες παιχνιδιών κ.λπ. Όταν κυκλοφόρησαν για πρώτη φορά οι υπολογιστές πολυμέσων, οι διεπαφές μετάδοσης των εξωτερικών συσκευών ήταν διαφορετικές. Για παράδειγμα, ο εκτυπωτής μπορούσε να συνδεθεί μόνο σε LPT, το μόντεμ μπορούσε να συνδεθεί μόνο στο RS232 και το ποντίκι και το πληκτρολόγιο μπορούσαν να συνδεθούν μόνο στο PS/2. Το περίπλοκο σύστημα διεπαφής, σε συνδυασμό με τον περιορισμό ότι το πρόγραμμα οδήγησης πρέπει να εγκατασταθεί και να επανεκκινηθεί για να χρησιμοποιηθεί, θα προκαλέσει προβλήματα στους χρήστες. Ως εκ τούτου, είναι μια αναπόφευκτη τάση να δημιουργηθεί μια ενοποιημένη διεπαφή εξωτερικής μετάδοσης που υποστηρίζει εύκολη σύνδεση και αποσύνδεση.

Η τελευταία γενιά USB είναι USB4 με ταχύτητα μεταφοράς 40 Gbit/s. Η φυσική υποδοχή σύνδεσης USB Type-A και Type-B χωρίζεται σε θετική και αρνητική σε αντίθεση με τη νέα υποδοχή USB Type-C η οποία δεν χωρίζεται.

Οπτικοί ή Ψηφιακοί Δίσκοι (CDs)

Το CD, είναι ένας μικρός επίπεδος στρογγυλός πλαστικός δίσκος οπτικής αποθήκευσης που χρησιμοποιείται για την αποθήκευση και ανάκτηση δεδομένων υπολογιστή ή μουσικού φωτισμού. Τα CD μπορούν να αποθηκεύουν προγράμματα λογισμικού που πρόκειται να εγκατασταθούν σε έναν υπολογιστή. Αποθηκεύουν αρχεία για δημιουργία αντιγράφων ασφαλείας ή μεταφορά σε άλλον υπολογιστή και αποθηκεύουν μουσική για αναπαραγωγή σε ένα CD player.

Το Compact Disc (CD) αναπτύχθηκε από κοινού από τη Philips και τη Sony και κυκλοφόρησε το 1982. Αυτή η μορφή αναπτύχθηκε αρχικά για αποθήκευση και αναπαραγωγή μόνο ψηφιακών ηχογραφήσεων (CD-DA), αλλά αργότερα προσαρμόστηκε για αποθήκευση δεδομένων (CD-ROM).

Το 1982, τα CD αντικατέστησαν τις δισκέτες επειδή ήταν πιο προηγμένες και μπορούσαν να αποθηκεύσουν περισσότερες πληροφορίες. Ωστόσο, οι πιο κοινές εναλλακτικές λύσεις CD σήμερα είναι οι δίσκοι DVD και Blu-ray. Το JumpDrive είναι επίσης μια κοινή εναλλακτική λύση στα CD για μεγαλύτερο χώρο αποθήκευσης.

Τα CD μπορούν να αποθηκεύσουν έως και 700 MB δεδομένων, που ισοδυναμεί με 80 λεπτά μουσικής. Ένα CD με πληροφορίες υπολογιστή ονομάζεται CD-ROM ή μνήμη μόνο για ανάγνωση CD. Η διάμετρος ενός κανονικού CD είναι 120 mm. Η μεσαία τρύπα του CD είναι περίπου 1,5 cm.

DVDs

Ένας ψηφιακός ευέλικτος δίσκος ή ένας ψηφιακός δίσκος βίντεο είναι ένα οπτικό μέσο αποθήκευσης παρόμοιο με έναν οπτικό δίσκο, αλλά με βελτιωμένη χωρητικότητα αποθήκευσης δεδομένων βίντεο και ήχου υψηλότερης ποιότητας. Αναπτύχθηκε το 1995 από τις Sony, Panasonic, Philips και Toshiba. Ορισμένα DVD είναι διαμορφωμένα ειδικά για αναπαραγωγή βίντεο, ενώ άλλα μπορεί να περιέχουν διαφορετικούς τύπους δεδομένων, όπως προγράμματα λογισμικού και αρχεία υπολογιστή.

Ένα DVD έχει χωρητικότητα αποθήκευσης 4,7 GB σε δίσκο μονής όψης, μονής στρώσης, κάτι που δεν αρκεί για μια ταινία 133 λεπτών. Τα DVD μπορούν να είναι μονής ή διπλής όψης, με δύο στρώματα σε κάθε πλευρά. Ένα DVD διπλής όψης δύο



επιπέδων μπορεί να χωρέσει έως και 17 GB βίντεο, ήχου ή άλλες πληροφορίες.

Οι ψηφιακοί ευέλικτοι δίσκοι μπορούν να ταξινομηθούν με διαφορετικούς τρόπους ανάλογα με την εφαρμογή τους. Εάν χρησιμοποιούνται μόνο για ανάγνωση και όχι για γραφή, ταξινομούνται ως DVD-ROM.

Εάν το DVD μπορεί να χρησιμοποιηθεί για την εγγραφή οποιουδήποτε τύπου δεδομένων, ονομάζεται DVD-R. Εάν ένας οπτικός δίσκος μπορεί να διαβαστεί, να γραφτεί, να διαγραφεί και να επανεγγραφεί, ονομάζεται DVD-RW.

Άλλες μορφές DVD περιλαμβάνουν:

Το DVD-Video είναι μια μορφή σχεδιασμένη για ταινίες μεγάλου μήκους που παίζονται σε τηλεοράσεις.

Χρησιμοποιήστε το CD για να αντικαταστήσετε το διαμορφωμένο DVD-Audio.

Μια εγγράψιμη έκδοση DVD-RAM.

Το DVD-ROM είναι ο τύπος μονάδας δίσκου και δίσκου που χρησιμοποιούνται σε υπολογιστές. Οι μονάδες DVD συνήθως αναπαράγουν επίσης κανονικούς δίσκους CD-ROM και δίσκους DVD-Video.

Η δισκέτα

Είναι ένα φορητό μέσο αποθήκευσης δεδομένων που αποτελείται από ένα κυκλικό δίσκο, φτιαγμένο από μαγνητικό υλικό που βρίσκεται μέσα σε μια τετράγωνη πλαστική θήκη. Την δισκέτα μπορούμε να την διαβάσουμε ή να την εγγράψουμε με έναν οδηγό δισκέτας. (από Βικιπαίδεια)

Έξυπνες Κάρτες

Δεν τις χρησιμοποιούμε συχνά προβλέπετε ότι θα κάνουν αισθητή τη παρουσία τους σε ένα ευρύ φάσμα εφαρμογών. Οι έξυπνες κάρτες στην εμφάνιση μοιάζουν με τις γνωστές πιστωτικές κάρτες, αλλά έχουν ενσωματωμένο ένα μικροεπεξεργαστή. Ανάλογα με την εφαρμογή, τα δεδομένα στη μνήμη μπορεί να αντιπροσωπεύουν είτε το διαθέσιμο πιστωτικό υπόλοιπο για το χρήστη, ή στοιχεία για την υγεία του ή ακόμα και κάποιο συνδυασμό διαφόρων πληροφοριών για το συγκεκριμένο πρόσωπο. Όταν ο χρήστης θέλει να χρησιμοποιήσει την έξυπνη κάρτα του σε κάποια πράξη (π.χ. αγορά ενός προϊόντος) τα δεδομένα της κάρτας διαβάζονται, επεξεργάζονται και τα



αποτελέσματα της επεξεργασίας (π.χ. το τροποποιημένο πιστωτικό όριο του χρήστη) ξανά αποθηκεύονται στην κάρτα. Επειδή οι έξυπνες κάρτες προσφέρουν αυξημένη ευελιξία και ασφάλεια σε σχέση με τις κοινές πιστωτικές κάρτες, είναι βέβαιο πως σύντομα θα τις συναντάμε στη καθημερινή μας ζωή.

Εκπαιδευτική Υποενοότητα : 2.3 Είδη δικτύων – Διαδίκτυο

Ζούμε πλέον σε έναν διαδικτυακό κόσμο. Πληροφορίες, ανταλλάσσονται και ανιχνεύονται στον κόσμο σε πραγματικό χρόνο. Είναι πιθανό σχεδόν όλοι και τα πάντα να είναι συνδεδεμένοι στον ψηφιακό αυτό κόσμο με τον ένα ή τον άλλο τρόπο. Δίκτυο υπολογιστών είναι τηλεπικοινωνιακό σύστημα από αυτόνομους ή μη αυτόνομους διασυνδεδεμένους υπολογιστές.

Οι υπολογιστές ή αλλιώς **κόμβοι**, θεωρούνται διασυνδεδεμένοι όταν είναι σε θέση να ανταλλάξουν πληροφορίες σε μορφή κειμένου ήχου, ή και εικόνας μεταξύ τους. Αυτόνομοι όταν δεν είναι δυνατό κάποιος υπολογιστής να ελέγξει τη λειτουργία κάποιου άλλου.

Ένα δίκτυο υπολογιστών μπορεί να διασυνδέονται μεταξύ τους εκτός από τα παραδοσιακά επιτραπέζια PC και άλλου τύπου συσκευές όπως PDAs (Personal Digital Assistants), κινητά τηλέφωνα, τηλεοράσεις, εκτυπωτές, σαρωτές.

Ορισμός

Μια ομάδα από δύο ή περισσότερων όμοιων πραγμάτων διασυνδεδεμένα μεταξύ τους λέγονται δίκτυο.

Ένα δίκτυο υπολογιστών είναι μια διασύνδεση ανάμεσα σε δύο ή περισσότερους υπολογιστές ή υπολογιστικές συσκευές.

Μερικά από τα παραδείγματα δικτύου στην καθημερινότητά μας η ζωή περιλαμβάνει:

- Κοινωνικό δίκτυο
- Δίκτυο κινητής τηλεφωνίας
- Δίκτυο υπολογιστών
- Δίκτυα αεροπορικών εταιρειών, σιδηροδρόμων, τραπεζών, νοσοκομείων

Αυτή η διασύνδεση επιτρέπει στους υπολογιστές να μοιράζονται δεδομένα μεταξύ τους.

Ένα βασικό δίκτυο είναι μια σύνθεση μερικών υπολογιστών που είναι τοποθετημένοι σε ένα δωμάτιο.

Το μέγεθος του δικτύου μπορεί να ποικίλλει από μικρό σε μεγάλο ανάλογα με τον αριθμό των υπολογιστών που συνδέει.

Ένα δίκτυο υπολογιστών μπορεί να περιλαμβάνει διαφορετικούς τύπους κεντρικούς υπολογιστές (ονομάζονται επίσης κόμβοι).

Εκτός από τους υπολογιστές, τα δίκτυα περιλαμβάνουν τη δικτύωση συσκευές όπως δρομολογητής, μόντεμ κ.λπ.

Για επικοινωνία, δεδομένων σε δίκτυο χωρίζεται σε μικρότερα κομμάτια που ονομάζονται πακέτα. Στη συνέχεια, τα πακέτα μεταφέρονται μέσω ενός δικτύου. Το δίκτυο μπορεί να συνδεθεί είτε μέσω ενσύρματων μέσων όπως καλώδια ή με ασύρματα μέσα.

Σε ένα δίκτυο επικοινωνίας, κάθε συσκευή που είναι μέρος ενός δικτύου και που μπορεί να λάβει, να δημιουργήσει, να αποθηκεύσει, ή αποστολή δεδομένων σε διαφορετικές διαδρομές δικτύου ονομάζεται κόμβος.

Στο πλαίσιο της επικοινωνίας δεδομένων, ένας κόμβος μπορεί να είναι μια συσκευή όπως μόντεμ, διανομέας, δρομολογητής, ψηφιακό τηλέφωνο, εκτυπωτής, υπολογιστής κτλ.

Διασύνδεση υπολογιστικών συσκευών σε δίκτυο μας επιτρέπει να ανταλλάσσουμε πληροφορίες ταυτόχρονα σε πολλά μέρη μέσω email, ιστοσελίδων, ήχου/βίντεο κλήσεις, κ.λπ.

Το δίκτυο επιτρέπει την κοινή χρήση πόρων. *Για παράδειγμα, ένας εκτυπωτής μπορεί να είναι διαθέσιμος σε πολλούς υπολογιστές μέσω δικτύου, μια δικτυακή αποθήκευση μπορεί να είναι προσπελάσιμη από πολλούς υπολογιστές.*

Οι άνθρωποι συχνά συνδέουν τις συσκευές τους μέσω hotspot, σχηματίζοντας έτσι ένα μικρό προσωπικό δίκτυο.

Η Εξέλιξη της Δικτύωσης.

Αργά αλλά σταδιακά, όλο και περισσότεροι οργανισμοί δημιούργησαν πολλά ανεξάρτητα μικρότερα δίκτυα.

Λίγα από τα ορόσημα στο μαγευτικό ταξίδι εξέλιξης των δικτύων υπολογιστών

Είδη Δικτύων

Τα δίκτυα φέρουν τους εξής χαρακτηρισμούς, που καθορίζουν και την κατηγορία τους :

Ανάλογα με το φυσικό μέσο διασύνδεσής τους χαρακτηρίζονται ως «ενσύρματα» ή «ασύρματα».

Ανάλογα με τον τρόπο πρόσβασης σε αυτά χαρακτηρίζονται ως «δημόσια» ή «ιδιωτικά» δίκτυα.

Ανάλογα με την γεωγραφική κάλυψη του δικτύου χαρακτηρίζονται ως «τοπικά» (LAN και WLAN), «μητροπολιτικά» (MAN και WMAN), «ευρείας κάλυψης» (WAN και WWAN) και «προσωπικά» (PAN και WPAN).

Οι χαρακτηρισμοί με το πρόσθετο **W** ανταποκρίνονται στον ασύρματο (Wireless) τρόπο σύνδεσης.

Τύποι δικτύων υπολογιστών που κυμαίνονται από δίκτυο συσκευών χειρός (όπως κινητά τηλέφωνα ή tablet) που συνδέονται μέσω Wi-Fi ή Bluetooth και που είναι απλωμένοι σε όλη την υδρόγειο.

Με βάση τη γεωγραφική περιοχή που καλύπτεται και την ταχύτητα μεταφοράς δεδομένων, τα δίκτυα υπολογιστών κατηγοριοποιούνται ευρέως :

- PAN (Personal Area Network)
- LAN (τοπικό δίκτυο)
- MAN (Δίκτυο Μητροπολιτικής Περιοχής)
- WAN (Δίκτυο ευρείας περιοχής).

Προσωπικό Δίκτυο Περιοχής (PAN)

Το Προσωπικό Δίκτυο Περιοχής (Αγγλικά: Personal Area Network , συντομογραφία PAN) αναφέρεται σε ένα δίκτυο επικοινωνίας που αποτελείται από υπολογιστικές συσκευές (όπως υπολογιστές , τηλέφωνα , PDA , ψηφιακές κάμερες κ.λπ.) εντός μιας προσωπικής εμβέλειας (που μεταφέρεται μαζί σας ή εντός λίγων μέτρα) . Ένα προσωπικό δίκτυο περιοχής μπορεί να χρησιμοποιηθεί είτε για την ανταλλαγή δεδομένων μεταξύ αυτών των συσκευών είτε για σύνδεση σε δίκτυα υψηλότερου επιπέδου ή στο Διαδίκτυο .

Το προσωπικό δίκτυο περιοχής μπορεί να είναι ενσύρματο, όπως ο δίαυλος USB ή Firewire (IEEE1394), μπορεί επίσης να είναι ασύρματο, όπως υπέρυθρες (IrDA) ή Bluetooth. Ένα προσωπικό δίκτυο που χρησιμοποιεί ασύρματη τεχνολογία ονομάζεται ασύρματο προσωπικό δίκτυο (WPAN)

Οι συσκευές που βρίσκονται σε απόσταση περίπου 10 μέτρων, μέσω USB σχηματίζουν ένα ενσύρματο PAN ενώ δύο smartphone επικοινωνούν μεταξύ τους μέσω Bluetooth τεχνολογίας αποτελούν ένα ασύρματο PAN ή WPAN.

Τοπικό Δίκτυο (LAN)

Τα «τοπικά δίκτυα» ή «LAN» (local area networks) είναι δίκτυα που συνδέουν υπολογιστές σε κοντινές αποστάσεις. Ένα δίκτυο αποτελείται από δύο ή περισσότερους συνδεδεμένους υπολογιστές και ένα LAN είναι ένα δίκτυο που περιέχεται σε μια μικρή γεωγραφική περιοχή, συνήθως μέσα στο ίδιο κτίριο. Τα οικιακά δίκτυα WiFi και τα δίκτυα μικρών επιχειρήσεων είναι κοινά παραδείγματα LAN. Τα LAN μπορεί επίσης να είναι αρκετά μεγάλα σε μέγεθος, αν και εάν καταλαμβάνουν πολλά κτίρια, είναι συχνά πιο ακριβές να ταξινομηθούν ως Δίκτυο ευρείας περιοχής (WAN) ή Δίκτυο Μητροπολιτικής Περιοχής (MAN).

Πώς λειτουργεί το LAN;

Τα περισσότερα LAN συνδέονται στο Διαδίκτυο από μια κεντρική τοποθεσία , τον δρομολογητή. Τα οικιακά τοπικά LAN χρησιμοποιούν συνήθως έναν μόνο δρομολογητή, ενώ τα LAN σε μεγαλύτερους χώρους μπορούν επιπλέον να



χρησιμοποιούν μεταγωγείς δικτύου για να βελτιώσουν την αποτελεσματικότητα της μεταφοράς πακέτων.

Τα LAN χρησιμοποιούν σχεδόν πάντα Ethernet, WiFi ή και τα δύο για τη σύνδεση συσκευών σε ένα δίκτυο. Το Ethernet είναι το πρωτόκολλο που χρησιμοποιείται για φυσικές συνδέσεις δικτύου και απαιτεί τη χρήση καλωδίου Ethernet. Το WiFi είναι ένα πρωτόκολλο σύνδεσης σε δίκτυο μέσω ραδιοκυμάτων.

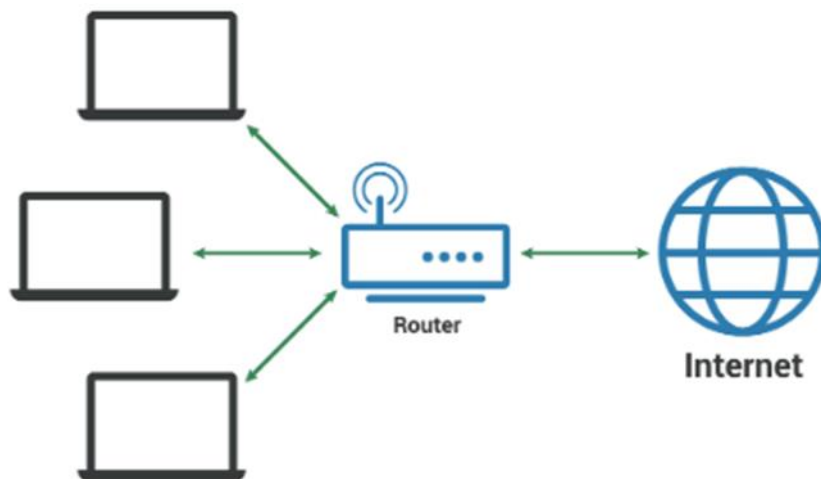
Μια μεγάλη ποικιλία συσκευών μπορούν να συνδεθούν στο LAN, συμπεριλαμβανομένων διακομιστών, επιτραπέζιων υπολογιστών, φορητών υπολογιστών, εκτυπωτών, συσκευών IoT, ακόμη και κονσόλες παιχνιδιών. Στα γραφεία, τα LAN χρησιμοποιούνται συχνά για να παρέχουν στους εσωτερικούς υπαλλήλους κοινή πρόσβαση σε συνδεδεμένους εκτυπωτές ή διακομιστές. Το απλούστερο LAN για σύνδεση στο Διαδίκτυο, που απαιτεί μόνο έναν δρομολογητή και έναν τρόπο σύνδεσης υπολογιστικών συσκευών στο δρομολογητή, όπως μέσω καλωδίου Ethernet ή σημείου πρόσβασης WiFi. Ένα LAN χωρίς σύνδεση στο Διαδίκτυο απαιτεί διακόπτη για την ανταλλαγή δεδομένων. Τα μεγάλα LAN, όπως αυτά σε μεγάλα κτίρια γραφείων, ενδέχεται να απαιτούν πρόσθετους δρομολογητές για πιο αποτελεσματική προώθηση δεδομένων στις σωστές συσκευές.

Δεν είναι όλα τα LAN συνδεδεμένα στο Διαδίκτυο. Στην πραγματικότητα, τα LAN προηγούνται του Διαδικτύου, τα LAN χρησιμοποιήθηκαν για πρώτη φορά σε επιχειρήσεις στα τέλη της δεκαετίας του 1970. (Τα πρωτόκολλα δικτύωσης που

χρησιμοποιούνται από αυτά τα παλιά LAN δεν χρησιμοποιούνται πλέον σήμερα.) Η μόνη απαίτηση για τη δημιουργία ενός LAN είναι οι συνδεδεμένες συσκευές να μπορούν να ανταλλάσσουν δεδομένα. Αυτό συνήθως απαιτεί μια συσκευή δικτύου για την ανταλλαγή πακέτων, όπως ένας μεταγωγέας δικτύου. Σήμερα, ακόμη και LAN που δεν είναι συνδεδεμένα στο διαδίκτυο χρησιμοποιούν τα ίδια πρωτόκολλα δικτύου (π.χ. IP) στο διαδίκτυο.

Τα δίκτυα LAN διακρίνονται από τα άλλα είδη δικτύων με βάση τρία χαρακτηριστικά:

- Το μέγεθος τους
- Την τεχνολογία μετάδοσης τους
- Την τοπολογία τους



Τι είναι εικονική μηχανή;

Ένα εικονικό LAN ή VLAN είναι μια μέθοδος διαίρεσης της κυκλοφορίας στο ίδιο φυσικό δίκτυο σε δύο δίκτυα.

Φανταστείτε να δημιουργήσετε δύο ξεχωριστά LAN στο ίδιο δωμάτιο, το καθένα με το δικό του δρομολογητή και σύνδεση στο διαδίκτυο. Τα VLAN είναι ακριβώς αυτό, αλλά στην πραγματικότητα οροθετούνται χρησιμοποιώντας λογισμικό και όχι φυσικό υλικό, το μόνο που χρειάζεστε είναι ένας δρομολογητής, για να συνδεθείτε στο διαδίκτυο.

Τα VLAN βοηθούν στη διαχείριση δικτύου, ειδικά για πολύ μεγάλα LAN. Τμηματοποιώντας το δίκτυο, οι διαχειριστές μπορούν να διαχειριστούν το δίκτυο πιο εύκολα. (Τα VLAN διαφέρουν αρκετά από τα υποδίκτυα, τα οποία είναι ένας άλλος τρόπος υποδιαίρεσης ενός δικτύου για μεγαλύτερη αποτελεσματικότητα.)

Ποια είναι η διαφορά μεταξύ LAN και WAN;

Ένα WAN, ή δίκτυο ευρείας περιοχής, είναι μια συλλογή από διασυνδεδεμένα LAN. Είναι ένα ευρύ δίκτυο τοπικών δικτύων. Ένα WAN μπορεί να έχει οποιοδήποτε μέγεθος, ακόμη και χιλιάδες μίλια πλάτος. Δεν περιορίζεται στην καθορισμένη περιοχή.

Πώς σχετίζεται το LAN με το υπόλοιπο Διαδίκτυο;

Το Διαδίκτυο είναι ένα δίκτυο δικτύων. Ένα LAN είναι συνήθως συνδεδεμένο σε ένα μεγαλύτερο δίκτυο, ένα αυτόνομο σύστημα (AS) . Τα AS είναι πολύ μεγάλα δίκτυα που έχουν τις δικές τους πολιτικές δρομολόγησης και ελέγχουν ορισμένες διευθύνσεις IP. Ένας πάροχος υπηρεσιών Διαδικτύου (ISP) είναι ένα παράδειγμα AS.

Σκεφτείτε ένα LAN ως ένα μικρό δίκτυο που είναι συνδεδεμένο σε ένα μεγαλύτερο δίκτυο, το οποίο είναι συνδεδεμένο με άλλα πολύ μεγάλα δίκτυα, τα οποία περιέχουν όλα ένα LAN. Αυτό είναι το Διαδίκτυο και δύο υπολογιστές που είναι συνδεδεμένοι σε δύο διαφορετικά LAN που απέχουν χιλιάδες μίλια μπορούν να επικοινωνούν μεταξύ τους στέλνοντας δεδομένα μέσω αυτών των συνδέσεων μεταξύ των δικτύων.

Πώς το Cloudflare προστατεύει το τοπικό δίκτυο;

Η εσωτερική επιχειρηματική υποδομή, όπως τα LAN και οι σχετικοί δρομολογητές, διακόπτες και διακομιστές συχνά εκτίθενται σε κακόβουλες επιθέσεις, συμπεριλαμβανομένων επιθέσεων DDoS . Το Cloudflare Magic Transit προστατεύει τα τοπικά δίκτυα και την υποδομή από κακόβουλες επιθέσεις, ενώ επιταχύνει τη νόμιμη κυκλοφορία ιστού. Το Cloudflare Magic Transit μπορεί επίσης να ασφαλίσει δίκτυα που φιλοξενούνται στο cloud και υβριδικά.

Μητροπολιτικό Δίκτυο Περιοχής (MAN) Το Metropolitan Area Network (MAN)

αναφέρεται σε ένα μεγάλο δίκτυο υπολογιστών , το οποίο ανήκει στο πρότυπο IEEE802.6 και είναι ένα δημόσιο δίκτυο μεταξύ LAN και WAN που μπορεί να

μεταδίδει φωνή και δεδομένα. Η MAN πρόκειται να βελτιώσει το μέσο μετάδοσης στο LAN (τοπικό δίκτυο), να επεκτείνει το εύρος του τοπικού δικτύου για να συμπεριλάβει μια πανεπιστημιούπολη, μια πόλη ή μια μητροπολιτική περιοχή . Είναι ένα μεγαλύτερο τοπικό δίκτυο που απαιτεί υψηλότερο κόστος, αλλά μπορεί να προσφέρει ταχύτερους ρυθμούς μεταφοράς. Για παράδειγμα, εάν μια εταιρεία συνδέει τις βάσεις υπηρεσιών της σε μια πόλη ή την ίδια χώρα, μπορεί να ονομαστεί ένα ή περισσότερα μητροπολιτικά δίκτυα.

Ορισμένες τεχνολογίες που χρησιμοποιούνται συνήθως σε δίκτυα αστικών περιοχών περιλαμβάνουν Ethernet (10Gbps/100Gbps), WiMAX (Παγκόσμια Διαλειτουργικότητα για Πρόσβαση σε Μικροκύματα).

Η τυπική εφαρμογή του δικτύου της μητροπολιτικής περιοχής είναι το ευρυζωνικό ένα δίκτυο επικοινωνίας πολυμέσων υψηλού εύρους ζώνης , πολλαπλών λειτουργιών, με πρόσβαση πολλαπλών υπηρεσιών .

Το ευρυζωνικό δίκτυο μητροπολιτικών περιοχών μπορεί να ανταποκριθεί στην αυξανόμενη ζήτηση για υπηρεσίες επικοινωνίας δεδομένων υψηλής ταχύτητας, υψηλής ποιότητας από κυβερνητικούς φορείς, χρηματοοικονομικές ασφάλειες, μεγάλα, μεσαία και δημοτικά σχολεία, εταιρείες και επιχειρήσεις , ιδιαίτερα τις ταχέως αναπτυσσόμενες ομάδες χρηστών του Διαδικτύου για ευρυζωνικές υπηρεσίες υψηλής ταχύτητας Πρόσβαση στο Διαδίκτυο με ταχύτητα.

Η ανάπτυξη του ευρυζωνικού δικτύου μητροπολιτικών περιοχών γνώρισε μακρά περίοδο, από παραδοσιακές υπηρεσίες φωνής έως υπηρεσίες εικόνας και βίντεο, από βασικές οπτικοακουστικές υπηρεσίες σε διάφορες υπηρεσίες προστιθέμενης αξίας, από 64 kb/s βασικές υπηρεσίες έως 2,5 Gb/s, 10 Gb/s και άλλες υπηρεσίες μισθωμένων γραμμών. Με την ανάπτυξη της τεχνολογίας και τη συνεχή αύξηση της ζήτησης, τα είδη των επιχειρήσεων επίσης αναπτύσσονται και αλλάζουν συνεχώς.

Το σταδιακά βελτιωμένο αστικό ευρυζωνικό έχει φέρει πολλές ευκολίες στη ζωή μας, πρόσβαση στο Διαδίκτυο υψηλής ταχύτητας, βίντεο κατά παραγγελία, βιντεοκλήσεις, τηλεόραση μέσω Διαδικτύου, εξ αποστάσεως εκπαίδευση, τηλεδιάσκεψη κ.λπ. Αυτές οι διάφορες εφαρμογές Διαδικτύου που χρησιμοποιούμε είναι στα παρασκήνια. Είναι το δίκτυο της μητροπολιτικής περιοχής που παίζει τεράστιο ρόλο.

Ένα τοπικό δίκτυο ή ένα δίκτυο ευρείας περιοχής εξυπηρετεί συνήθως μια μονάδα ή σύστημα, ενώ ένα δίκτυο μητροπολιτικών περιοχών εξυπηρετεί μια ολόκληρη πόλη και όχι ένα συγκεκριμένο τμήμα.

Το μητροπολιτικό δίκτυο (metropolitan area network), ή δίκτυο MAN είναι μια μεγαλύτερη εκδοχή ενός τοπικού δικτύου καθώς καλύπτει μεγαλύτερες αποστάσεις, π.χ. από μια ομάδα γειτονικών γραφείων μιας εταιρείας έως μια πόλη.

Το μητροπολιτικό δίκτυο(MAN) είναι στην ουσία μια εκτεταμένη μορφή LAN που καλύπτει μια μεγαλύτερη γεωγραφική περιοχή. Ο ρυθμός μεταφοράς δεδομένων στο MAN κυμαίνεται επίσης σε Mbps, αλλά είναι πολύ λιγότερο σε σύγκριση με το LAN.

Καλωδιακή τηλεόραση με υπηρεσίες ευρείας ζώνης διαδικτύου που συνδέεται με δίκτυο ή καλωδιακά, είναι παραδείγματα του MAN.

Αυτό το είδος δικτύου μπορεί να επεκταθεί έως 30-40 χλμ. Μερικές φορές, πολλά LAN συνδέονται μαζί για να σχηματίσουν τον MAN.

Το πιο γνωστό παράδειγμα μητροπολιτικού δικτύου είναι το δίκτυο καλωδιακής τηλεόρασης που υπάρχει σε πολλές πόλεις.

Αυτό το σύστημα είναι η εξέλιξη των παλαιότερων συστημάτων κοινοτικών κεραιών που χρησιμοποιούνταν σε περιοχές με κακή τηλεοπτική λήψη από αέρος.

Σε αυτά τα πρώιμα συστήματα, μια μεγάλη κεραία ήταν τοποθετημένη στην κορυφή ενός κοντινού λόφου και στην συνέχεια το σήμα στελνόταν στα σπίτια των συνδρομητών.

Αρχικά, τα συστήματα αυτά ήταν τοπικά σχεδιασμένα και εξειδικευμένα. Αργότερα κάποιες εταιρίες άρχισαν να ασχολούνται με αυτή τη δουλειά, υπογράφοντας συμβόλαια με τις δημοτικές αρχές για την καλωδίωση μιας ολόκληρης πόλης .

Το επόμενο βήμα ήταν τηλεοπτικά προγράμματα και ακόμη και ολόκληρα κανάλια που ήταν σχεδιασμένα μόνο για την καλωδιακή τηλεόραση.

Από τότε που το Internet άρχισε να προσελκύει το ενδιαφέρον του κόσμου, οι επιχειρήσεις δικτύου καλωδιακής τηλεόρασης άρχισαν να αντιλαμβάνονται ότι με ορισμένες αλλαγές στο σύστημα θα μπορούσαν να παρέχουν αμφίδρομες υπηρεσίες Internet σε μη χρησιμοποιούμενα τμήματα του φάσματος.

Σε αυτό το σημείο, το σύστημα καλωδιακής τηλεόρασης άρχισε να μεταλλάσσεται από έναν τρόπο διανομής τηλεόρασης σε ένα μητροπολιτικό δίκτυο.

Δίκτυο Ευρείας περιοχής (WAN)

Δίκτυο ευρείας περιοχής (αγγλικά: Wide Area Network , συντομογραφία WAN), γνωστό και ως δίκτυο ευρείας περιοχής , εξωτερικό δίκτυο , δημόσιο δίκτυο . Είναι ένα απομακρυσμένο δίκτυο που συνδέει τοπικά δίκτυα ή αστικά δίκτυα σε διαφορετικές περιοχές για επικοινωνία με υπολογιστή. Συνήθως εκτείνεται σε μια μεγάλη φυσική εμβέλεια, καλύπτοντας μια σειρά από δεκάδες χιλιόμετρα έως χιλιάδες χιλιόμετρα, μπορεί να συνδέσει πολλές περιοχές, πόλεις και χώρες ή να εκτείνεται σε πολλές ηπείρους και να παρέχει επικοινωνία μεγάλων αποστάσεων, σχηματίζοντας ένα διεθνές απομακρυσμένο δίκτυο. Το WAN δεν είναι το ίδιο με το Διαδίκτυο. Σε μια περιοχή που υπερβαίνει την απόσταση που συνδέεται με τον διανομέα, πρέπει να συνδεθεί μέσω δρομολογητή . Αυτός ο τύπος δικτύου ονομάζεται δίκτυο ευρείας περιοχής. Εάν υπάρχουν υποκαταστήματα στα βόρεια, στο κέντρο και στο νότο, ακόμη και στο εξωτερικό, αυτά τα υποκαταστήματα συνδέονται με αποκλειστικές γραμμές, το οποίο ονομάζεται «δίκτυο ευρείας περιοχής».

Το μέσο μετάδοσης του WAN χρησιμοποιεί κυρίως τηλεφωνικές γραμμές ή οπτικές ίνες και οι πάροχοι ISP συνδέουν τις επιχειρήσεις. Αυτές οι γραμμές είναι προ-θαμμένες κάτω από το δρόμο από τους φορείς εκμετάλλευσης ISP. Επειδή το έργο είναι τεράστιο, η συντήρηση δεν είναι εύκολη και Το εύρος ζώνης μπορεί να είναι εγγυημένο. , επομένως θα είναι πιο ακριβό.

Το Διαδίκτυο αναφέρεται γενικά σε ένα δημόσιο δίκτυο ευρείας περιοχής. Το κόστος ενός δημόσιου δικτύου ευρείας περιοχής είναι χαμηλότερο και είναι ένα φθηνότερο περιβάλλον δικτύου. Ωστόσο, σε σύγκριση με ένα δίκτυο ευρείας περιοχής, δεν υπάρχει τρόπος διαχείρισης του εύρους ζώνης. στο δημόσιο σύστημα δικτύου, το εύρος ζώνης οποιουδήποτε τμήματος δεν είναι εγγυημένο.

Στα περισσότερα δίκτυα ευρείας περιοχής, το υποδίκτυο αποτελείται από δύο διακριτά συστατικά:

- Τις γραμμές μετάδοσης

- Στοιχεία μεταγωγής.

Οι γραμμές μετάδοσης μετακινούν bit ανάμεσα στις μηχανές. Μπορεί να υλοποιούνται με χάλκινα σύρματα , οπτικές ίνες, ή ακόμη και με ασύρματες συνδέσεις. Τα στοιχεία μεταγωγής είναι εξειδικευμένοι υπολογιστές που συνδέουν τρεις ή περισσότερες γραμμές μετάδοσης.

Όταν φτάνουν δεδομένα σε μια εισερχόμενη γραμμή, το στοιχείο μεταγωγής πρέπει να επιλέξει την εξερχόμενη γραμμή στην οποία θα τα προωθήσει.

Η αρχή λειτουργίας ενός WAN μεταγωγής πακέτων είναι σημαντική.

Γενικά, όταν μια διεργασία σε έναν υπολογιστή υπηρεσίας πρέπει να στείλει ένα μήνυμα σε κάποιον άλλο υπολογιστή υπηρεσίας. Τότε ο υπολογιστής υπηρεσίας του αποστολέα, τεμαχίζει αρχικά το μήνυμα σε πακέτα, με κάθε πακέτο να σημειώνει τη θέση που κατέχει στην ακολουθία.

Στην συνέχεια αυτά τα πακέτα εισάγονται ένα προς ένα στο δίκτυο, σε γρήγορη διαδοχή.

Τα πακέτα μεταφέρονται αυτόνομα μέσα στο δίκτυο και παραδίδονται στον υπολογιστή υπηρεσίας του παραλήπτη, όπου συναρμολογείτε ξανά το αρχικό μήνυμα και παραδίδεται στη διεργασία παραλήπτη.

Στο επόμενο σχήμα απεικονίζεται μια ροή πακέτων που προκύπτει από κάποιο αρχικό μήνυμα.

Συσκευές Δικτύου

Για την επικοινωνία δεδομένων με διαφορετικά μέσα και να διαμορφώσετε δίκτυα με διαφορετικά λειτουργικότητα, χρειαζόμαστε διαφορετικές συσκευές όπως μόντεμ, Hub, Switch, Repeater, Router, Gateway, κ.λπ.

Modem

Το μόντεμ σημαίνει «Modulator DEModulator». Αναφέρεται σε μια συσκευή που χρησιμοποιείται για τη μετατροπή μεταξύ αναλογικών σημάτων και ψηφιακά bits. Γνωρίζουμε ότι οι υπολογιστές αποθηκεύουν και επεξεργάζονται δεδομένα σε όρους 0 και 1. Ωστόσο, για τη μετάδοση δεδομένων από ένας αποστολέας σε έναν παραλήπτη ή κατά την περιήγηση στο Διαδίκτυο, τα ψηφιακά δεδομένα μετατρέπονται σε αναλογικό σήμα και το μέσο (είτε είναι ελεύθερος χώρος είτε φυσικό μέσο) μεταφέρει το σήμα στον δέκτη.

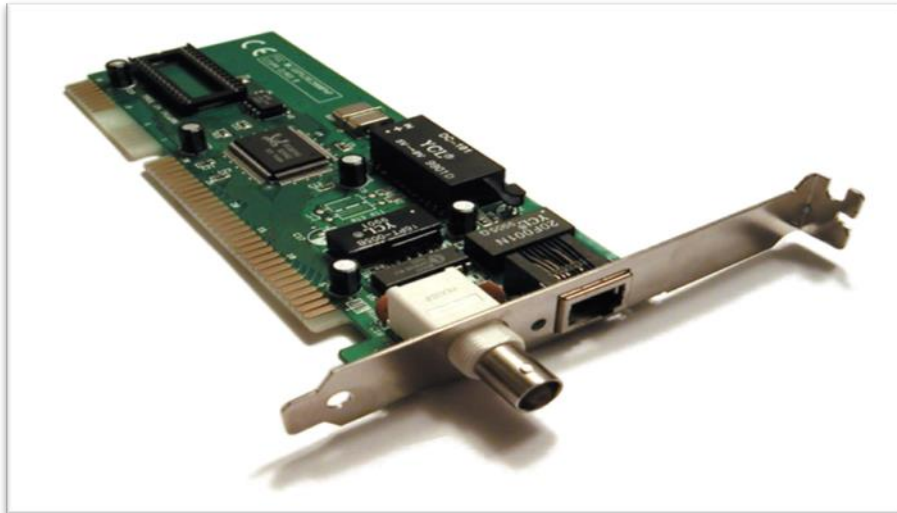
Υπάρχουν συνδεδεμένα μόντεμ και στους κόμβους προέλευσης και προορισμού.

Το μόντεμ στο τέλος του αποστολέα λειτουργεί ως διαμορφωτής που μετατρέπει τα ψηφιακά δεδομένα σε αναλογικά σήματα.

Το μόντεμ στο άκρο του δέκτη λειτουργεί ως αδιαμόρφωτής που μετατρέπει τα αναλογικά σήματα στα ψηφιακά δεδομένα για τον προορισμό κόμβος για να καταλάβεις τη συνδεσιμότητα χρησιμοποιώντας ένα Modem.

Κάρτα Ethernet

Κάρτα Ethernet, επίσης γνωστή ως κάρτα διασύνδεσης δικτύου (η κάρτα NIC εν συντομία) είναι ένας προσαρμογέας δικτύου που χρησιμοποιείται για τη ρύθμιση σε ένα ενσύρματο δίκτυο. Λειτουργεί ως επαφή μεταξύ υπολογιστή και το δίκτυο. Είναι ένα κύκλωμα πλακέτα τοποθετημένη στο μητρική πλακέτα του υπολογιστή Το καλώδιο Ethernet συνδέει τον υπολογιστή με το δίκτυο μέσω NIC. Οι κάρτες Ethernet μπορούν να υποστηρίξουν μεταφορά δεδομένων μεταξύ 1 Mbps και 1 Gbps (1000 Mbps). Κάθε NIC έχει ένα Διεύθυνση MAC, η οποία βοηθά στον μοναδικό προσδιορισμό του υπολογιστή στο δίκτυο.



RJ45

Το RJ 45 ή το Registered Jack-45 είναι ένα σύνδεσμος οκτώ ακίδων που χρησιμοποιείται αποκλειστικά με Καλώδια Ethernet για δικτύωση.

Είναι μια τυπική δικτύωση διεπαφή που στο το τέλος όλων των καλωδίων δικτύου.

Βασικά, είναι ένα μικρό πλαστικό βύσμα που χωράει στις υποδοχές RJ-45.

Κάρτες Ethernet υπάρχουν σε διάφορα υπολογιστικές συσκευές.

Repeater

Τα δεδομένα μεταφέρονται με τη μορφή σημάτων πάνω από το καλώδιο.

Αυτά τα σήματα μπορούν να διανύσουν μια καθορισμένη απόσταση (συνήθως περίπου 100 m). Τα σήματα χάνουν τη δύναμή τους πέρα από αυτή την περιορισμένη απόσταση. Σε τέτοιες συνθήκες, τα πρωτότυπα σήματα πρέπει να αναδημιουργηθούν.

Ο Repeater είναι μια αναλογική συσκευή που λειτουργεί με σήματα στα καλώδια τα οποία είναι συνδεδεμένο.

Τα εξασθενημένα σήματα που εμφανίζεται στο καλώδιο είναι σαν να αναγεννάτε και τοποθετείται πίσω στο καλώδιο από έναν Repeater.

Switch Διακόπτης

Ένας διακόπτης είναι μια συσκευή δικτύου που παίζει κεντρικό ρόλο σε ένα τοπικό δίκτυο (LAN).

Ένας Switch - Διακόπτης δικτύου χρησιμοποιείται για τη σύνδεση πολλαπλών υπολογιστών ή συσκευές επικοινωνίας.

Όταν τα δεδομένα φτάνουν το Switch - Διακόπτη, εξάγει στην διεύθυνση προορισμού από το πακέτο δεδομένων και το αναζητά σε έναν πίνακα για να δείτε πού να στείλετε το πακέτο.

Έτσι, στέλνει σήματα μόνο σε επιλεγμένα συσκευές αντί για να αποστολή σε όλες. Μπορεί να προωθήσει πολλά πακέτα στο την ίδια ώρα.

Οι διακόπτες Ethernet είναι συνηθισμένοι σε σπίτια/γραφεία για να συνδέσετε πολλές συσκευές δημιουργώντας έτσι LAN ή σε πρόσβαση στο διαδίκτυο.

Δρομολογητής- Router

Ένα Router είναι μια συσκευή δικτύου που μπορεί λαμβάνουν τα δεδομένα.

Το Router συνδέει ένα τοπικό δίκτυο με το Διαδίκτυο.

Σε σύγκριση με έναν διανομέα ή έναν διακόπτη, ένα Router έχει προηγμένες δυνατότητες καθώς μπορεί να αναλύσει τα δεδομένα που υπάρχουν για μεταφορά μέσω δικτύου.

Για παράδειγμα, τα δεδομένα έχουν χωριστεί σε πακέτα ορισμένων μεγεθών.

Ας υποθέσουμε ότι αυτά τα πακέτα πρόκειται να μεταφερθούν σε διαφορετικού τύπου δίκτυο που δεν μπορεί να χειριστεί μεγαλύτερο πακέτα.

Σε μια τέτοια περίπτωση, τα δεδομένα πρέπει να επανασυσκευαστούν ως μικρότερα πακέτα και στη συνέχεια αποστέλλονται μέσω του δικτύου από το Router

Ένα Router μπορεί να είναι ενσύρματο ή ασύρματο.

Ένας ασύρματος Router μπορεί να παρέχει πρόσβαση Wi-Fi σε smartphone και άλλα συσκευές.

Συνήθως, τέτοια Router περιέχουν επίσης ορισμένες θύρες για παροχή ενσύρματης πρόσβασης στο Διαδίκτυο.

Το Router συνδέονται με εισερχόμενες ευρείας ζώνης γραμμές, από τον ISP (Internet Service Provider) και μετατρέπει σε ψηφιακά δεδομένα για επεξεργασία υπολογιστικών συσκευών.

Gateway – Πύλη

Ο όρος «Gateway» υποδηλώνει ένα βασικό σημείο πρόσβασης που λειτουργεί ως «πύλη» μεταξύ του δικτύου ενός οργανισμού και τον έξω κόσμο του Διαδικτύου

Το Gateway χρησιμεύει ως το σημείο εισόδου και εξόδου ενός δικτύου, όπως όλα τα δεδομένα που εισέρχονται ή εξέρχονται από ένα δίκτυο πρέπει πρώτα να περάσει από την πύλη για να χρησιμοποιήσετε τη δρομολόγηση τα μονοπάτια. Εκτός από τη δρομολόγηση των δεδομένων, υπάρχουν και πύλες διατήρησης πληροφοριών σχετικά με το εσωτερικό δίκτυο του κεντρικού υπολογιστή.

Μονοπάτια (διαδρομές) σύνδεσης και τα προσδιορισμένα μονοπάτια άλλων απομακρυσμένων δικτύων.

Εάν ένας κόμβος από το ένα δίκτυο, θέλει να επικοινωνήσει με έναν κόμβο ενός ξένου δικτύου θα περάσει το πακέτο δεδομένων στην πύλη, η οποία στη συνέχεια θα το δρομολογήσει στον προορισμό που πρέπει, χρησιμοποιώντας την καλύτερη δυνατή διαδρομή.

Για μια απλή σύνδεση στο Διαδίκτυο από το σπίτι, η Gateway – Πύλη είναι συνήθως ο πάροχος υπηρεσιών Διαδικτύου που παρέχει πρόσβαση σε ολόκληρο το Διαδίκτυο.

Γενικά, το Router- δρομολογητής έχει ρυθμιστεί να λειτουργεί ως συσκευή πύλης σε δίκτυα υπολογιστών.

Επειδή μια Gateway - Πύλη δικτύου τοποθετείται στην άκρη του δικτύου, το τείχος προστασίας είναι συνήθως ενσωματώνεται με αυτό.

Χρήσεις των δικτύων υπολογιστών

Πριν αρχίσουμε να εξετάζουμε λεπτομερώς τα τεχνικά θέματα, θα εξετάσουμε τις πολλαπλές εφαρμογές των δικτύων υπολογιστών.

Θα ξεκινήσουμε με τις επιχειρηματικές εφαρμογές ,θα προχωρήσουμε στις οικιακές εφαρμογές και τέλος στις σχετικά πρόσφατες εξελίξεις που αφορούν τους μετακινούμενους χρήστες.

Επιχειρηματικές εφαρμογές

Πολλές εταιρίες έχουν σημαντικό πλήθος υπολογιστών. Για παράδειγμα, ένα εργοστάσιο μπορεί να έχει ξεχωριστούς υπολογιστές για το τμήμα του λογιστηρίου, για της αποθήκες , και για την παρακολούθηση της παραγωγής.

Αρχικά κάθε ένας από αυτούς τους υπολογιστές μπορεί να λειτουργούσε απομονωμένος από τους άλλους , αλλά σε κάποιο σημείο θα μπορούσε η διοίκηση να αποφασίσει να τους συνδέσει μεταξύ τους για να συλλέγει πληροφορίες για όλη την εταιρεία. Το ζητούμενο σε αυτή την περίπτωση είναι η κοινοχρησία πόρων (resource sharing), και ο στόχος είναι ο εξοπλισμός, τα προγράμματα και κυρίως τα δεδομένα να είναι διαθέσιμα σε οποιονδήποτε στο δίκτυο, χωρίς να έχει σημασία η φυσική θέση του πόρου. Ένα άλλο παράδειγμα μπορεί μια εταιρεία να έχει πολλά γραφεία διασκορπισμένα σε πολλές χώρες του κόσμου και ο αρμόδιος χρήστης να θέλει να συλλέξει

πληροφορίες που τυχαίνει να βρίσκεται 12.000 χιλιόμετρα μακριά από τα δεδομένα του , αυτό όμως δεν θα πρέπει να τον εμποδίζει να χρησιμοποιεί τα δεδομένα αυτά σαν να ήταν τοπικά χάρη στην κοινοχρησία πόρων.

Στο παραπάνω μοντέλο ,τα δεδομένα αποθηκεύονται σε ισχυρούς υπολογιστές που ονομάζονται **servers**.

Αυτοί συχνά βρίσκονται τοποθετημένοι σε ένα κεντρικό σημείο.

Αντίθετα, οι υπάλληλοι έχουν στα γραφεία τους απλούστερες μηχανές που ονομάζονται πελάτες(clients), με τις οποίες μπορούν να προσπελάσουν τα απομακρυσμένα δεδομένα.

Οι μηχανές πελάτη και ο servers συνδέονται με ένα δίκτυο, όπως φαίνεται στην εικόνα .

Αυτή η διάταξη ονομάζεται μοντέλο πελάτη- διακομιστή (client-server-model).

Το μοντέλο αυτό χρησιμοποιείται ευρύτατα και είναι η βάση για τις περισσότερες χρήσεις των δικτύων.

Το μοντέλο αυτό, έχει εφαρμογή και στις δύο περιπτώσεις που αναφέραμε παραπάνω, δηλαδή όταν πελάτης και διακομιστής βρίσκονται και οι δυο στο ίδιο κτίριο και επίσης όταν πελάτης και διακομιστής απέχουν χιλιόμετρα μακριά!

Το δίκτυο υπολογιστών μπορεί να παρέχει ένα ισχυρό μέσο επικοινωνίας ανάμεσα στους υπαλλήλους ,χάρη στο ηλεκτρονικό ταχυδρομείο(email).

Το ηλεκτρονικό ταχυδρομείο, όμως δεν είναι η μόνη μορφή βελτιωμένης επικοινωνίας που έγινε εφικτή χάρη στα δίκτυα υπολογιστών.

Η βιντεοδιάσκεψη (videoconferencing) είναι μια τεχνολογία όπου υπάλληλοι που βρίσκονται σε απομακρυσμένες τοποθεσίες μπορούν να οργανώσουν μια 'συνάντηση', βλέποντας και ακούγοντας ο ένας τον άλλον και γράφοντας σε ένα κοινόχρηστο εικονικό πίνακα .

Χρησιμοποιώντας τα δίκτυα υπολογιστών οι εταιρείες μπορούν να βλέπουν χιλιάδες αγαθά και τέλος να προμηθεύονται το αγαθό που προτιμούν στην καλύτερη τιμή γλιτώνοντας χρόνο και χρήμα.

Αυτό, οφείλεται στο ηλεκτρονικό εμπόριο (ecommerce).

Οικιακές εφαρμογές

Παλιότερα η οικιακή χρήση ηλεκτρονικών υπολογιστών συνέβαινε με στόχο την επεξεργασία κειμένου και τα παιχνίδια, αλλά τελευταία η κατάσταση έχει αλλάξει ριζικά. Σήμερα είναι πιθανό ότι ο σημαντικότερος λόγος είναι η πρόσβαση στο Internet. Μερικές από τις πιο δημοφιλείς χρήσεις του Internet για τους οικιακούς χρήστες είναι οι ακόλουθες:

- Πρόσβαση σε απομακρυσμένες πληροφορίες.
- Διαπροσωπική επικοινωνία.
- Αλληλεπιδραστική διασκέδαση.
- Ηλεκτρονικό εμπόριο.

Η πρόσβαση σε απομακρυσμένες πληροφορίες έχει πολλές μορφές. Μπορεί να είναι περιήγηση στον παγκόσμιο ιστό για πληροφόρηση ή ακόμα και απλώς για διασκέδαση.

Πολλές εφημερίδες, περιοδικά είναι σε άμεση σύνδεση και μπορούν αν εξατομικευτούν. Επίσης υπάρχει και η άμεσα συνδεδεμένη ψηφιακή βιβλιοθήκη.

Πολλοί επαγγελματικοί οργανισμοί όπως η IEEE Computer Society διαθέτουν ήδη σε άμεση σύνδεση πολλές επιστημονικές εκδόσεις και πρακτικά συνεδρίων.

Όλες οι παραπάνω εφαρμογές περιλαμβάνουν αλληλεπίδραση ανάμεσα σε ένα άτομο και μια απομακρυσμένη βάση δεδομένων γεμάτη με πληροφορίες.

Μια άλλη κατηγορία χρήσης του δικτύου είναι η διαπροσωπική (person to person) επικοινωνία. Μία άλλη, είναι τα άμεσα μηνύματα (instant messaging) όπου σχεδόν όλοι η σύγχρονοι έφηβοι είναι εθισμένοι σε αυτό.

Μια παραλλαγή αυτής της ιδέας είναι το δωμάτιο επικοινωνίας (chat room), όπου μια ομάδα ατόμων μπορεί να γράφει μηνύματα τα οποία είναι ορατά σε όλους.

Ένας άλλος τύπος διαπροσωπικής επικοινωνίας είναι η ομότιμη (peer to peer) επικοινωνία, ώστε να διακρίνετε από το μοντέλο πελάτη- διακομιστή.

Η «peer to peer» σημαίνει ότι δύο υπολογιστές επικοινωνούν μεταξύ τους, χωρίς να υπάρχει κάποιος server που να αναλαμβάνει τη μεταξύ τους επικοινωνία, αλλά άμεση σύνδεση.

Για να γίνει αυτό πρέπει οι δύο υπολογιστές να έχουν το ίδιο λογισμικό, μέσω του οποίου θα επικοινωνούν. Έτσι π.χ. μπορούν να σταλούν ηλεκτρονικά μηνύματα, ή να γίνει ανταλλαγή πληροφοριών ή αρχείων από τον έναν υπολογιστή απευθείας στον άλλον.

Μετακινούμενοι χρήστες

Επειδή η καλωδιακή σύνδεση είναι αδύνατη όταν βρισκόμαστε μέσα σε αυτοκίνητα και αεροσκάφη, υπάρχει μεγάλο ενδιαφέρον για τα ασύρματα δίκτυα.

Στην υποενότητα αυτή θα δούμε συνοπτικά μερικές χρήσεις των ασυρμάτων δικτύων. Οι άνθρωποι που βρίσκονται συχνά στο δρόμο θέλουν να χρησιμοποιούν το φορητό

ηλεκτρονικό εξοπλισμό τους για να στέλνουν και να λαμβάνουν τηλεφωνικές κλήσεις, φάξ, και μηνύματα ηλεκτρονικού ταχυδρομείου, να κάνουν περιήγηση στον ιστό, να έχουν πρόσβαση σε απομακρυσμένα αρχεία, και να συνδέονται σε απομακρυσμένες μηχανές.

Θέλουν επίσης να μπορούν να το κάνουν αυτό από οποιαδήποτε σημείο στην ξηρά τη θάλασσα, ή τον αέρα.

Για παράδειγμα τα πανεπιστήμια έχουν εγκαταστήσει ασύρματα δίκτυα στις πανεπιστημιούπολεις τους, ώστε οι φοιτητές να μπορούν να συμβουλευονται τον κατάλογο της βιβλιοθήκης ή να διαβάζουν το ηλεκτρονικό τους ταχυδρομείο.

Τα ασύρματα δίκτυα έχουν μεγάλη αξία για στόλους φορητών, ταξί, οχημάτων διανομής και τεχνικούς επισκευών που θέλουν να είναι σε διαρκή επικοινωνία με τη βάση τους. Επίσης τα ασύρματα δίκτυα είναι πολύ σημαντικά και για τις ένοπλες δυνάμεις.

Ένας άλλος τομέας στον οποίο τα ασύρματα δίκτυα θα μπορούσε να εξοικονομήσουν χρήματα είναι ο έλεγχος των μετρητών των υπηρεσιών κοινής ωφέλειας. Αν οι μετρητές ηλεκτρικού ρεύματος, φυσικού αερίου, και νερό στα σπίτια των καταναλωτών μπορούσαν να αναφέρουν τα στοιχεία κατανάλωσης μέσω ενός ασυρμάτου δικτύου, δεν θα υπήρχε ανάγκη να στέλνονται υπάλληλοι για καταμέτρηση.

Παρόμοια, οι ασύρματοι ανιχνευτές καπνού θα μπορούσαν να καλούν την πυροσβεστική υπηρεσία αντί να παράγουν απλώς έναν ήχο (ο οποίος έχει ελάχιστη αξία όταν δεν υπάρχει κανείς μέσα στο σπίτι).

Καθώς το κόστος τόσο των ασυρμάτων συσκευών όσο και του χρόνου σύνδεσης μειώνεται, όλο και περισσότερες μετρήσεις και αναφορές θα εκτελούνται μέσα από τα ασύρματα δίκτυα.

Ένας εντελώς διαφορετικός τομέας εφαρμογών για τα ασύρματα δίκτυα είναι η αναμενόμενη συγχώνευση των κινητών τηλεφώνων και των PDA σε μικροσκοπικούς ασύρματους υπολογιστές.

Μια πρώτη απόπειρα ήταν οι μικροσκοπικές ασύρματες συσκευές PDA που μπορούσαν να εμφανίζουν 'απογυμνωμένες' ιστοσελίδες στις ακόμη πιο μικροσκοπικές οθόνες τους.

Αυτό το σύστημα, το οποίο ονομαζόταν Πρωτόκολλο Ασύρματων Εφαρμογών 1.0 ή WAP 1.0(wireless Application Protocol), απέτυχε , κυρίως λόγω των μικροσκοπικών οθονών, του χαμηλού εύρους ζώνης, και των φτωχών παρεχόμενων υπηρεσιών. Οι νέες συσκευές και υπηρεσίες είναι όμως καλύτερες, χάρη στο πρωτόκολλο WAP 2.0.

Τεχνολογία μετάδοσης

Δεν υπάρχει κάποια γενικώς αποδεκτή ταξινόμηση στην οποία εντάσσονται όλα τα δίκτυα υπολογιστών, αλλά δύο ιδιότητες των δικτύων ξεχωρίζουν ως οι πλέον σημαντικές : η τεχνολογία μετάδοσης και η κλίμακα.

Σε γενικές γραμμές, υπάρχουν δύο είδη τεχνολογιών μετάδοσης που χρησιμοποιούνται ευρέως.

Οι τεχνολογίες αυτές είναι :

- Συνδέσεις εκπομπής.
- Συνδέσεις σημείο προς σημείο.

Τα Δίκτυα Εκπομπής(broadcast networks) έχουν έναν μοναδικό δίαυλο-κανάλι επικοινωνίας που τον μοιράζονται όλες οι μηχανές του δικτύου. Ένας υπολογιστής ενός τέτοιου δικτύου αποστέλλει μηνύματα με την μορφή πακέτων (packets) στο δίκτυο και αυτά λαμβάνονται από όλους τους υπόλοιπους υπολογιστές του δικτύου.

Με την παραλαβή του πακέτου κάθε υπολογιστής εξετάζει το πεδίο διεύθυνσης που αναγράφεται πάνω στο κάθε πακέτο και το οποίο είναι χαρακτηριστικό του τελικού προορισμού του. Αν το πακέτο προορίζεται για αυτόν τότε το επεξεργάζεται, ενώ σε αντίθετη περίπτωση το αγνοεί.

Τα συστήματα εκπομπής γενικά παρέχουν και τη δυνατότητα να απευθύνουμε ένα πακέτο σε όλους τους παραλήπτες, χρησιμοποιώντας έναν ειδικό κωδικό στο πεδίο διεύθυνσης.

Όταν ένα πακέτο μεταδίδεται με αυτόν τον κωδικό, το παραλαμβάνουν και το επεξεργάζονται όλες οι μηχανές του δικτύου. Αυτή η κατάσταση λειτουργίας ονομάζεται εκπομπή η ευρεία μετάδοση (broadcasting).

Μερικά συστήματα εκπομπής υποστηρίζουν και τη μετάδοση προς ένα υποσύνολο των μηχανών, δυνατότητα που ονομάζεται πολυδιανομή (multicasting).

Ένας πιθανός τρόπος με τον οποίο μπορεί να γίνεται αυτό είναι να δεσμεύεται ένα δυαδικό ψηφίο το οποίο θα υποδηλώνει πολυδιανομή.

Τα υπόλοιπα $n-1$ bit της διεύθυνσης περιέχουν έναν αριθμό ομάδας .Κάθε μηχανή μπορεί να 'γραφτεί συνδρομητής' σε οποιοσδήποτε ομάδες, ή και σε όλες τις ομάδες. Όταν ένα πακέτο στέλνεται σε μια συγκεκριμένη ομάδα, τότε παραδίδεται σε όλες τις μηχανές που είναι συνδρομητές στην ομάδα αυτή.

Αντίθετα, τα Δίκτυα Σημείου προς Σημείο(point-to-point)

Δίκτυο από σημείο σε σημείο, σύνδεση από σημείο σε σημείο , ο απλούστερος τύπος δικτύου υπολογιστών , στο οποίο δύο υπολογιστές συνδέονται απευθείας μεταξύ τους μέσω εξοπλισμού επικοινωνίας. Το πλεονέκτημα αυτού του τύπου σύνδεσης είναι η απλότητα και το χαμηλό κόστος, το μειονέκτημα είναι ότι δεν μπορούν να συνδεθούν περισσότεροι από δύο υπολογιστές με αυτόν τον τρόπο, σε αντίθεση με τέτοιες μεθόδους μετάδοσης δεδομένων όπως η εκπομπή και το σημείο σε πολλαπλά σημεία. Συχνά χρησιμοποιείται σε περιπτώσεις όπου είναι απαραίτητη η γρήγορη μεταφορά πληροφοριών από έναν υπολογιστή, για παράδειγμα, φορητό υπολογιστή σε έναν άλλο.

Εάν τα δύο σημεία βρίσκονται σε κοντινή απόσταση, τότε η επικοινωνία γίνεται συνήθως μέσω RS232 ή παρόμοιου πρωτοκόλλου (δείτε τη σύνδεση Null modem). Κατά τη σύνδεση εξ αποστάσεως, χρησιμοποιούνται μόντεμ .

Για συνδέσεις από σημείο σε σημείο , χρησιμοποιούνται πρωτόκολλα SLIP [2] , PPP , HDLC [1] και άλλα .Η σύγχρονη χρήση του όρου «δίκτυο από σημείο σε σημείο» αναφέρεται σε σταθερά σημεία ασύρματης πρόσβασης στο Διαδίκτυο ή VoIP μέσω ραδιοκυμάτων στην περιοχή πολλαπλών GHz, καθώς και σε τεχνολογίες όπως οι τηλεπικοινωνίες λέιζερ .

Τύποι δικτύων με βάση την κλίμακα

Ένα άλλο κριτήριο ταξινόμησης των δικτύων είναι η κλίμακά τους. Στο σχήμα ταξινομούμε τα συστήματα πολλών επεξεργαστών ανάλογα με το φυσικό τους μέγεθος.

Στην κορυφή βρίσκονται τα δίκτυα προσωπικής περιοχής, δηλαδή δίκτυα που προορίζονται για ένα άτομο.

Για παράδειγμα, ένα ασύρματο δίκτυο που συνδέει έναν υπολογιστή με το ποντίκι, το πληκτρολόγιο, και τον εκτυπωτή του είναι ένα δίκτυο προσωπική περιοχής. Μετά, τα δίκτυα προσωπικής περιοχής έχουμε δίκτυα με μεγαλύτερη ακτίνα δράσης.

Αυτά μπορούν να διακριθούν σε τοπικά δίκτυα, μητροπολιτικά δίκτυα, και δίκτυα ευρείας περιοχής.

Τέλος, η σύνδεση δύο ή περισσότερων δικτύων ονομάζεται διαδίκτυο (internet).

Τοπικά δίκτυα

Τα τοπικά δίκτυα (local area networks), που συνήθως αποκαλούνται δίκτυα LAN, είναι ιδιωτικά δίκτυα τα οποία βρίσκονται μέσα σε ένα κτίριο ή κτιριακό συγκρότημα, ή σε μια έκταση με μέγεθος μέχρι λίγα χιλιόμετρα.

Χρησιμοποιούνται ευρέως για τη διασύνδεση προσωπικών υπολογιστών και σταθμών εργασίας σε γραφεία και εργοστάσια εταιρειών, με στόχο την κοινοχρησία πόρων και την ανταλλαγή πληροφοριών.

Στα δίκτυα δακτυλίου (ring) το κάθε bit διαδίδεται μόνο του, χωρίς να περιμένει για το υπόλοιπο πακέτο στο οποίο ανήκει.

Συνήθως το κάθε bit μπορεί να καλύψει ολόκληρο το δακτύλιο στο διάστημα που απαιτείτε για τη μετάδοση λίγων μόνο bit, συχνά πριν καν μεταδοθεί ολόκληρο τα πακέτο.

Όπως σε όλα τα συστήματα εκπομπής, απαιτούνται κάποιοι κανόνες διαιτησίας ώστε να αποφεύγονται οι ταυτόχρονες μεταδόσεις στο δακτύλιο.

Χρησιμοποιούνται διάφορες μέθοδοι, όπως το να βάζουν τις μηχανές να μεταδίδουν με τη σειρά. Το πρότυπο IEEE 802.5 (το δίκτυο δακτυλίου με σκυτάλη) είναι ένα

δίκτυο LAN που βασίζεται σε δακτύλιο και λειτουργεί σε ταχύτητες από 4 Mbps έως 16 Mbps.

Ένα άλλο παράδειγμα α δακτυλίου είναι το FDDI. Τα δίκτυα εκπομπής μπορούν να υποδιαιρεθούν περαιτέρω σε στατικά και δυναμικά, ανάλογα με το πώς γίνεται η εκχώρηση του καναλιού.

Για τους υπολογιστές μεταγωγής έχουν χρησιμοποιηθεί διάφορα ονόματα στο παρελθόν. Το όνομα που χρησιμοποιείται περισσότερο σήμερα είναι 'δρομολογητής'(router).

Δίκτυο πρόσβασης- Δίκτυο κορμού



Ένα γεωγραφικά εκτεταμένο δίκτυο μπορεί να χωριστεί σε δίκτυο κορμού και δίκτυο πρόσβασης, όπως φαίνεται στην εικόνα.

Τα παγκόσμια δίκτυα χωρίζονται σε δίκτυα κορμού και δίκτυα πρόσβασης.

Το δίκτυο κορμού (Backbone network)

Συνιστάται να χωριστούν τα εδαφικά δίκτυα που χρησιμοποιούνται για τη δημιουργία ενός εταιρικού δικτύου σε δύο μεγάλες κατηγορίες:

- δίκτυα κορμού.
- δίκτυα πρόσβασης.

Δίκτυα ευρείας περιοχής κορμού χρησιμοποιούνται για το σχηματισμό peer-to-peer συνδέσεων μεταξύ μεγάλων τοπικών δικτύων που ανήκουν σε τμήματα μεγάλων επιχειρήσεων. Τα εδαφικά δίκτυα κορμού πρέπει να παρέχουν υψηλή απόδοση, καθώς οι ροές μεγάλου αριθμού υποδικτύων συνδυάζονται στον κορμό. Επιπλέον, τα δίκτυα κορμού πρέπει να είναι συνεχώς διαθέσιμα, δηλαδή να παρέχουν έναν πολύ υψηλό παράγοντα διαθεσιμότητας, αφού μεταφέρουν την κίνηση πολλών κρίσιμων για τις επιχειρήσεις εφαρμογών που είναι κρίσιμες για την επιτυχή λειτουργία της επιχείρησης. Λόγω της ιδιαίτερης σημασίας των βασικών κεφαλαίων, μπορούν να «αποχαιρετήσουν» το υψηλό κόστος. Δεδομένου ότι μια επιχείρηση συνήθως δεν έχει πολλά μεγάλα δίκτυα, τα δίκτυα κορμού δεν απαιτούνται για τη διατήρηση μιας εκτεταμένης υποδομής πρόσβασης. Συνήθως, τα δίκτυα κορμού είναι ψηφιακά μισθωμένα κυκλώματα με ταχύτητες από 2 έως 622 Mbps, τα οποία φέρουν πρωτόκολλα αρχιτεκτονικής IP, IPX ή IBM SNA, αναμετάδοση πλαισίου, δίκτυα μεταγωγής πακέτων ATM, X.25 ή TCP/IP. Με αποκλειστικούς συνδέσμους, χρησιμοποιείται μια μικτή πλεονάζουσα τοπολογία συνδέσμων για την παροχή υψηλής διαθεσιμότητας για τον κορμό.

Στην περιοχή Δίκτυα πρόσβασης αναφέρεται σε εδαφικά δίκτυα απαραίτητα για τη σύνδεση μικρών τοπικών δικτύων και μεμονωμένων απομακρυσμένων υπολογιστών με το κεντρικό τοπικό δίκτυο μιας επιχείρησης. Εάν η οργάνωση των βασικών επικοινωνιών δόθηκε πάντα μεγάλη προσοχή κατά τη δημιουργία ενός εταιρικού δικτύου, τότε η οργάνωση της απομακρυσμένης πρόσβασης για τους υπαλλήλους μιας επιχείρησης έχει γίνει ένα στρατηγικά σημαντικό ζήτημα μόλις πρόσφατα. Η γρήγορη πρόσβαση σε εταιρικές πληροφορίες από οποιαδήποτε γεωγραφική τοποθεσία καθορίζει την ποιότητα της λήψης αποφάσεων από τους υπαλλήλους της για πολλούς τύπους επιχειρηματικών δραστηριοτήτων. Η σημασία αυτού του παράγοντα αυξάνεται με την αύξηση του αριθμού των εργαζομένων που εργάζονται στο σπίτι (τηλεεργαζόμενοι - τηλεργαζόμενοι), συχνά σε επαγγελματικά ταξίδια, και με την αύξηση του αριθμού των μικρών υποκαταστημάτων επιχειρήσεων που βρίσκονται σε διαφορετικές πόλεις και, ίσως, διαφορετικές χώρες.

Τα ATM ή τα ταμειακά μηχανήματα μπορούν επίσης να λειτουργήσουν ως χωριστοί απομακρυσμένοι κόμβοι, απαιτώντας πρόσβαση σε μια κεντρική βάση δεδομένων

για τη λήψη πληροφοριών σχετικά με νόμιμους πελάτες τραπεζών των οποίων οι πλαστικές κάρτες πρέπει να έχουν εγκριθεί τοπικά. Τα ATM ή οι ταμειακές μηχανές είναι συνήθως σχεδιασμένα για να αλληλεπιδρούν με τον κεντρικό υπολογιστή μέσω του δικτύου X.25, το οποίο κάποτε αναπτύχθηκε ειδικά ως δίκτυο για απομακρυσμένη πρόσβαση μη ευφυούς τερματικού εξοπλισμού στον κεντρικό υπολογιστή.

Οι απαιτήσεις για δίκτυα πρόσβασης διαφέρουν σημαντικά από εκείνες για δίκτυα κορμού. Δεδομένου ότι μια επιχείρηση μπορεί να έχει πολλά σημεία απομακρυσμένης πρόσβασης, μία από τις κύριες απαιτήσεις είναι η παρουσία μιας εκτεταμένης υποδομής πρόσβασης που μπορεί να χρησιμοποιηθεί από τους υπαλλήλους της επιχείρησης τόσο όταν εργάζονται στο σπίτι όσο και σε επαγγελματικά ταξίδια. Επιπλέον, το κόστος της απομακρυσμένης πρόσβασης πρέπει να είναι μέτριο ώστε να δικαιολογείται οικονομικά το κόστος σύνδεσης δεκάδων ή εκατοντάδων απομακρυσμένων συνδρομητών. Ταυτόχρονα, οι απαιτήσεις εύρους ζώνης για έναν μόνο υπολογιστή ή ένα τοπικό δίκτυο που αποτελείται από δύο ή τρεις πελάτες συνήθως ταιριάζουν στο εύρος πολλών δεκάδων kilobits ανά δευτερόλεπτο (αν μια τέτοια ταχύτητα δεν ικανοποιεί πλήρως τον απομακρυσμένο πελάτη, τότε συνήθως η ευκολία του έργου της θυσιάζεται για χάρη της εξοικονόμησης των κεφαλαίων της επιχείρησης).

Ως δίκτυα πρόσβασης χρησιμοποιούνται συνήθως αναλογικά τηλεφωνικά δίκτυα, δίκτυα ISDN και, λιγότερο συχνά, δίκτυα αναμετάδοσης πλαισίου. Κατά τη σύνδεση τοπικών δικτύων υποκαταστημάτων, χρησιμοποιούνται επίσης αποκλειστικά κανάλια με ταχύτητες από 19,2 έως 64 Kbps. Ένα ποιοτικό άλμα στην επέκταση των δυνατοτήτων απομακρυσμένης πρόσβασης σημειώθηκε λόγω της ταχείας αύξησης της δημοτικότητας και της επικράτησης του Διαδικτύου. Οι υπηρεσίες μεταφοράς μέσω Διαδικτύου είναι φθηνότερες από τα υπεραστικά και διεθνή τηλεφωνικά δίκτυα και η ποιότητά τους βελτιώνεται ραγδαία.

Το λογισμικό και το υλικό που συνδέει τους υπολογιστές ή τα δίκτυα LAN απομακρυσμένων χρηστών σε ένα εταιρικό δίκτυο ονομάζονται εργαλεία απομακρυσμένης πρόσβασης. Συνήθως, στην πλευρά του πελάτη, αυτά τα εργαλεία αντιπροσωπεύονται από ένα μόντεμ και σχετικό λογισμικό.

Η οργάνωση της μαζικής απομακρυσμένης πρόσβασης από την πλευρά του κεντρικού τοπικού δικτύου παρέχεται από διακομιστή απομακρυσμένης πρόσβασης (*Remote Access Server, RAS*) .

Ο διακομιστής απομακρυσμένης πρόσβασης είναι ένα σύμπλεγμα λογισμικού και υλικού που συνδυάζει τις λειτουργίες ενός δρομολογητή, μιας γέφυρας και μιας πύλης. Ο διακομιστής εκτελεί τη μία ή την άλλη λειτουργία ανάλογα με τον τύπο του πρωτοκόλλου που χρησιμοποιείται από τον απομακρυσμένο χρήστη ή το απομακρυσμένο δίκτυο. Οι διακομιστές απομακρυσμένης πρόσβασης έχουν συνήθως αρκετές θύρες χαμηλής ταχύτητας ώστε οι χρήστες να συνδέονται μέσω αναλογικών τηλεφωνικών δικτύων ή ISDN.

Εμφανίζεται στο σχ. 6.5. Η δομή του παγκόσμιου δικτύου που χρησιμοποιείται για τον συνδυασμό μεμονωμένων τοπικών δικτύων και απομακρυσμένων χρηστών σε ένα εταιρικό δίκτυο είναι αρκετά χαρακτηριστική. Έχει μια έντονη ιεραρχία οχημάτων περιοχής, συμπεριλαμβανομένου κορμού υψηλής ταχύτητας (για παράδειγμα, συνδέσεις SDH 155-622 Mbps), δίκτυα πιο αργής πρόσβασης περιοχής για σύνδεση μεσαίου μεγέθους LAN (για παράδειγμα, ρελέ καρέ) και τηλέφωνο γενικής χρήσης δίκτυο για υπαλλήλους απομακρυσμένης πρόσβασης.

Συμπεράσματα

Τα δίκτυα υπολογιστών ευρείας περιοχής (WAN) χρησιμοποιούνται για τη συνένωση διαφορετικών τύπων συνδρομητών: μεμονωμένους υπολογιστές διαφορετικών κατηγοριών - από κεντρικούς υπολογιστές έως προσωπικούς υπολογιστές, τοπικά δίκτυα υπολογιστών, απομακρυσμένα τερματικά.

Λόγω του υψηλού κόστους της παγκόσμιας δικτυακής υποδομής, υπάρχει επείγουσα ανάγκη μεταφοράς όλων των τύπων κίνησης που εμφανίζονται στην επιχείρηση, και όχι μόνο της κίνησης υπολογιστών, σε ένα ενιαίο δίκτυο: φωνητική κίνηση του εσωτερικού τηλεφωνικού δικτύου που λειτουργεί σε αυτόματο γραφείο τηλεφωνικά κέντρα (PBX), κίνηση μηχανημάτων φαξ, βιντεοκάμερες, ταμειακές μηχανές, ATM και λοιπού εξοπλισμού παραγωγής.

Για την υποστήριξη τύπων κυκλοφορίας πολυμέσων, δημιουργούνται ειδικές τεχνολογίες: ISDN, B-ISDN. Επιπλέον, οι τεχνολογίες WAN, οι οποίες αναπτύχθηκαν

για να μεταφέρουν μόνο κίνηση υπολογιστών, προσαρμόστηκαν πρόσφατα για να μεταφέρουν φωνή και βίντεο. Για να γίνει αυτό, δίνεται προτεραιότητα στα πακέτα που μεταφέρουν φωνητικές μετρήσεις ή δεδομένα εικόνας και σε εκείνες τις τεχνολογίες που το επιτρέπουν, δημιουργείται μια σύνδεση με προδεδειγμένο εύρος ζώνης για τη μεταφορά τους. Υπάρχουν ειδικές συσκευές πρόσβασης - πολυπλέκτης "φωνή - δεδομένα" ή "βίντεο - δεδομένα", που συσκευάζουν πληροφορίες πολυμέσων σε πακέτα και τις στέλνουν μέσω του δικτύου, και στην άκρη λήψης αποσυσκευάζουν και τις μετατρέπουν στην αρχική τους μορφή - φωνή ή βίντεο.

Τα δίκτυα ευρείας περιοχής παρέχουν κυρίως υπηρεσίες μεταφοράς, μεταφέροντας δεδομένα κατά τη μεταφορά μεταξύ τοπικών δικτύων ή υπολογιστών. Υπάρχει μια αυξανόμενη τάση υποστήριξης υπηρεσιών σε επίπεδο εφαρμογής για συνδρομητές παγκόσμιων δικτύων: διανομή δημοσίως διαθέσιμων πληροφοριών ήχου, βίντεο και κειμένου, καθώς και οργάνωση διαδραστικής αλληλεπίδρασης μεταξύ συνδρομητών δικτύου σε πραγματικό χρόνο. Αυτές οι υπηρεσίες προέρχονται από το Διαδίκτυο και μεταφέρονται με επιτυχία σε εταιρικά δίκτυα σε αυτό που ονομάζεται τεχνολογία intranet.

Όλες οι συσκευές που χρησιμοποιούνται για τη σύνδεση των συνδρομητών στο παγκόσμιο δίκτυο χωρίζονται σε δύο κατηγορίες: DTE που παράγουν πραγματικά δεδομένα και DCE που μεταδίδουν δεδομένα σύμφωνα με τις απαιτήσεις της διεπαφής καθολικού καναλιού και τερματίζουν το κανάλι.

Οι τεχνολογίες WAN ορίζουν δύο τύπους διεπαφής: από χρήστη σε δίκτυο (UNI) και δίκτυο σε δίκτυο (NNI). Η διασύνδεση UNI είναι πάντα βαθιά λεπτή για να επιτρέψει την πρόσβαση σε εξοπλισμό από διαφορετικούς κατασκευαστές για σύνδεση στο δίκτυο. Η διεπαφή NNI μπορεί να μην είναι τόσο λεπτομερής καθώς τα μεγάλα δίκτυα μπορούν να διασυνδεθούν κατά περίπτωση.

Τα παγκόσμια δίκτυα υπολογιστών λειτουργούν με βάση την τεχνολογία μεταγωγής πακέτων, πλαισίου και κυψέλης. Τις περισσότερες φορές, ένα παγκόσμιο δίκτυο υπολογιστών ανήκει σε μια εταιρεία τηλεπικοινωνιών που ενοικιάζει τις υπηρεσίες δικτύου της. Ελλείψει τέτοιου δικτύου στην επιθυμητή περιοχή, οι επιχειρήσεις δημιουργούν ανεξάρτητα παγκόσμια δίκτυα ενοικιάζοντας αποκλειστικά κανάλια ή κανάλια μέσω τηλεφώνου από εταιρείες τηλεπικοινωνιών ή τηλεφωνίας.

Σε μισθωμένα κυκλώματα, μπορείτε να δημιουργήσετε ένα δίκτυο με ενδιάμεση μεταγωγή βάσει οποιασδήποτε τεχνολογίας δικτύου ευρείας περιοχής (X.25, frame relay, ATM) ή να συνδέσετε απευθείας δρομολογητές ή γέφυρες τοπικών δικτύων με μισθωμένα κυκλώματα. Η επιλογή του τρόπου χρήσης των μισθωμένων καναλιών εξαρτάται από τον αριθμό και την τοπολογία των συνδέσεων μεταξύ των τοπικών δικτύων.

Σύγχρονες υπηρεσίες δικτύων

Οι σύγχρονες υπηρεσίες των δικτύων υπολογιστών στηρίζονται κυρίως στην αλματώδη ανάπτυξη του Διαδικτύου, το οποίο από ένα ρόλο παρουσίασης πληροφοριών μεταλλάχθηκε σε ένα εργαλείο κοινωνικής δικτύωσης, αναμορφώνοντας τον τρόπο επικοινωνίας και συνεργασίας μεταξύ ανθρώπων.

Ο Παγκόσμιος Ιστός [Αγγλικά] World Wide Web, συντομ. - WWW ή Web (ιστός) - ένα κατακεκομμένο ετερογενές σύστημα υπολογιστή για τη συλλογική χρήση εγγράφων υπερμέσων, που λειτουργεί με βάση το Διαδίκτυο. Τα υπερσυνδεδεμένα έγγραφα αυτού του συστήματος, τα οποία σχηματίζουν μια σύνθετη διακλαδισμένη δομή, ονομάζονται μεταφορικά «ιστός» (ιστός). Μία από τις πιο δημοφιλείς υπηρεσίες Διαδικτύου (μαζί με e-mail, μηχανές αναζήτησης κ.λπ.). Τα έγγραφα υπερμέσου Ιστού, που ονομάζονται ιστοσελίδες, είναι μια ανάπτυξη εγγράφων υπερκειμένου (βλ. υπερκείμενο). Μπορεί να περιέχουν κείμενο, εικόνες, ήχο, βίντεο και άλλα στοιχεία. Κάθε ιστοσελίδα έχει μια μοναδική διεύθυνση - URL (συντομ. Universal Resource Locator - Uniform Resource Locator), με την οποία μπορεί να βρεθεί. Για παράδειγμα, το www.webopedia.com είναι η διεύθυνση URL της κύριας

ιστοσελίδας του λεξικού ηλεκτρονικού υπολογιστή. Οποιοδήποτε σύνολο παρακείμενων συμβόλων μιας ιστοσελίδας μπορεί να είναι υπερσύνδεσμος σε άλλη ιστοσελίδα ή άλλο πόρο πληροφοριών Διαδίκτυο. Οι υπερσύνδεσμοι μπορούν να οδηγούν όχι μόνο σε έγγραφα Ιστού, αλλά και σε άλλους πόρους πληροφοριών που παρέχονται από υπηρεσίες Διαδικτύου. Μια συλλογή ιστοσελίδων με υπερσυνδέσμους που μοιράζονται ένα κοινό μέρος μιας διεύθυνσης URL (και συνήθως ομαδοποιούνται θεματικά) ονομάζεται ιστότοπος. Για παράδειγμα, ο ιστότοπος του λεξικού ηλεκτρονικού υπολογιστή Webopedia είναι www.webopedia.com και η ιστοσελίδα του άρθρου με τον ορισμό του όρου Web είναι http://www.webopedia.com/TERM/W/World_Wide_Web.html. Το κοινό μέρος της διεύθυνσης όλων των σελίδων του ιστότοπου είναι η διεύθυνση του ιστότοπου.

Η τηλεφωνία μέσω διαδικτύου (Voice over IP ή VoIP)

Πολλοί άνθρωποι χρησιμοποιούν τους όρους Voice over Internet Protocol (VoIP) και IP Telephony (IPT) εναλλακτικά, εξισώνοντας το ένα με το άλλο. Ωστόσο, το VoIP είναι μόνο ένα μέρος της τηλεφωνίας IP. Η διαφορά μεταξύ αυτών των δύο όρων είναι τόσο λεπτή που πολλοί άνθρωποι πιστεύουν ότι σημαίνουν το ίδιο πράγμα. Το VoIP είναι ένας τύπος IP τηλεφωνίας. Μπορεί να ακούγεται περίεργο, αλλά η «τηλεφωνία» αναφέρεται στα τηλέφωνα, ενώ η τηλεφωνία του Πρωτοκόλλου Διαδικτύου ασχολείται με την ψηφιακή πλευρά των τηλεπικοινωνιών. Αυτό το κάνει χρησιμοποιώντας ένα πρωτόκολλο Διαδικτύου γνωστό ως Voice over IP (VoIP). Με κυριολεκτική έννοια, αυτό σημαίνει ότι μεταδίδετε φωνητικά σήματα μέσω του Διαδικτύου. Το πρωτόκολλο καθορίζει τον τρόπο διανομής της φωνής στο δίκτυο, παρόμοια με τον τρόπο με τον οποίο το Πρωτόκολλο μεταφοράς υπερκειμένου (HTTP) ορίζει τον τρόπο κατανόησης, μετάδοσης, μορφοποίησης και εμφάνισης των δεδομένων από διακομιστές Ιστού και προγράμματα περιήγησης.

Η τηλεφωνία IP είναι ένας τρόπος ψηφιοποίησης του τηλεφωνικού συστήματος προκειμένου να εκμεταλλευτείτε το Διαδίκτυο και κάθε εξοπλισμό ή εφαρμογή που συνδέεται με αυτό. Ο κύριος σκοπός της τηλεφωνίας IP είναι η αύξηση της παραγωγικότητας, γι' αυτό και ο όρος χρησιμοποιείται συχνότερα σε ένα επιχειρηματικό περιβάλλον. Από την άλλη πλευρά, το VoIP είναι απλώς ένα ψηφιακό όχημα για τηλεφωνικές κλήσεις. Σε διάφορες παραλλαγές, λειτουργεί για την προσφορά φθηνών ή δωρεάν κλήσεων και την προσθήκη πρόσθετων λειτουργιών για φωνητικές επικοινωνίες. Ορισμένοι περιγράφουν την τηλεφωνία IP ως μια γενική εμπειρία αποτελεσματικής και αξιόπιστης επικοινωνίας με χρήση πρωτοκόλλων Διαδικτύου. Το VoIP προσφέρει βολικές δυνατότητες για τη διευκόλυνση αυτής της διαδικασίας. Η διαφορά είναι τόσο λεπτή που η χρήση των δύο όρων εναλλακτικά μπορεί να είναι αποδεκτή σε πολλά πλαίσια, ακόμη και μόνο για να αποφευχθεί η σύγχυση.

Πώς να πραγματοποιήσετε δωρεάν κλήσεις μέσω Διαδικτύου;

Υπάρχουν πολλοί τρόποι για να κάνετε δωρεάν τηλεφωνικές κλήσεις μέσω Διαδικτύου. Ο ευκολότερος τρόπος είναι να κατεβάσετε μια εφαρμογή στο tablet ή το τηλέφωνό σας που θα σας επιτρέψει να τη χρησιμοποιήσετε σαν ένα κανονικό τηλέφωνο χωρίς να χάσετε ένα λεπτό συνομιλίας. Αλλά, εάν χρειάζεστε μια πραγματικά αξιόπιστη σύνδεση μέσω Διαδικτύου, με υψηλή ποιότητα επικοινωνίας, καθώς και με δυνατότητα χρήσης διαφόρων επιχειρηματικών λειτουργιών, τότε η IP-τηλεφωνία είναι απαραίτητη εδώ.

Οι τεχνολογίες DSL (Digital Subscriber Line)

Η ανάγκη για τεχνολογίες xDSL εμφανίστηκε κυρίως λόγω της πολύ γρήγορης ανάπτυξης του Διαδικτύου. Οι χρήστες πρέπει να αυξήσουν τις ταχύτητες μετάδοσης και να επεκτείνουν τις δυνατότητες απομακρυσμένης πρόσβασης - όλες οι οποίες δεν μπορούν να υποστηριχθούν από παραδοσιακά χρησιμοποιούμενες τεχνολογίες.

Τα συστήματα xDSL είναι κανάλια μετάδοσης δεδομένων υψηλής ταχύτητας που, εκτός από την πρόσβαση υψηλής ταχύτητας στο Διαδίκτυο, μπορούν επίσης να χρησιμοποιηθούν για την οργάνωση βιντεοδιασκέψεων, διαδραστικών

προγραμμάτων πολυμέσων και ως μέσο απομακρυσμένης πρόσβασης των εργαζομένων στο οικιακό γραφείο σε εταιρικά LAN. Φυσικά, υπάρχουν πολλές άλλες χρήσεις για τις τεχνολογίες xDSL. Για παράδειγμα, μια διαφημιστική εταιρεία μπορεί να χρησιμοποιεί συστήματα xDSL για την αποστολή και λήψη μεγάλων αρχείων γραφικών όταν είναι απαραίτητο να λάβει έγκριση πελάτη. Μετά από αυτό, μέσα σε λίγα δευτερόλεπτα, το αρχείο μπορεί να σταλεί για εκτύπωση.

Για να εγκαταστήσετε το DSL, πρέπει να έχετε πρόσβαση σε ένα καλωδιακό τηλεφωνικό δίκτυο. Τα μόντεμ DSL εγκαθίστανται και στα δύο άκρα της τηλεφωνικής γραμμής: το ένα μόντεμ είναι εγκατεστημένο στον συνδρομητή και το άλλο στο τηλεφωνικό κέντρο.

Σε αντίθεση με προηγούμενες τεχνολογίες που χρησιμοποιούν χάλκινη τηλεφωνική γραμμή, τα συστήματα xDSL δεν απαιτούν χειροκίνητη διαμόρφωση κατά την εγκατάσταση. Το μόντεμ αναλύει αυτόματα τη γραμμή και ρυθμίζει τη σύνδεση σε δευτερόλεπτα. Αυτή η διαδικασία συνεχίζεται κατά τη διάρκεια της σύνδεσης, καθώς το μόντεμ αντισταθμίζει τις αλλαγές που συμβαίνουν στη γραμμή (για παράδειγμα, που σχετίζονται με αλλαγές θερμοκρασίας). Τα μόντεμ χρησιμοποιούν προηγμένους αλγόριθμους επεξεργασίας ψηφιακού σήματος (DSP) που δημιουργούν μαθηματικά μοντέλα παραμόρφωσης γραμμής και εκτελούν αυτόματη διόρθωση. Ο ρυθμός μεταφοράς δεδομένων επηρεάζεται από το μήκος της γραμμής, το οποίο εξαρτάται από τη διατομή των πυρήνων του καλωδίου, τον τύπο μόνωσης και το επίπεδο παρεμβολής που υπάρχει στη γραμμή. Το Υπολογιστικό Νέφος ή σύννεφο

Το Υπολογιστικό Νέφος ή σύννεφο

Η τεχνολογία cloud είναι ένας γενικός όρος για την τεχνολογία δικτύου, την τεχνολογία πληροφοριών, την τεχνολογία ενοποίησης, την τεχνολογία πλατφόρμας διαχείρισης και την τεχνολογία εφαρμογών που βασίζεται σε επιχειρηματικά μοντέλα υπολογιστικού νέφους. Η τεχνολογία cloud computing θα γίνει σημαντική υποστήριξη. Οι υπηρεσίες παρασκηνίου των τεχνικών συστημάτων δικτύου απαιτούν πολλούς υπολογιστικούς και αποθηκευτικούς πόρους, όπως ιστότοπους βίντεο, ιστότοπους με εικόνες και περισσότερους ιστότοπους πύλης. Με την υψηλή ανάπτυξη και εφαρμογή της βιομηχανίας του Διαδικτύου, κάθε στοιχείο μπορεί να

έχει το δικό του αναγνωριστικό σήμα στο μέλλον και πρέπει να μεταδοθεί στο back-end σύστημα για λογική επεξεργασία. Δεδομένα διαφορετικών επιπέδων θα υποβάλλονται σε επεξεργασία ξεχωριστά και όλα Τα είδη των βιομηχανικών δεδομένων πρέπει να είναι ισχυρά. Η υποστήριξη του συστήματος μπορεί να πραγματοποιηθεί μόνο μέσω του cloud computing. Η απλούστερη τεχνολογία υπολογιστικού νέφους μπορεί να δει παντού στις υπηρεσίες δικτύου , όπως μηχανές αναζήτησης, γραμματοκιβώτια δικτύου κ.λπ., οι χρήστες μπορούν να λάβουν πολλές πληροφορίες απλά εισάγοντας απλές εντολές.

Στο μέλλον, οι κινητές συσκευές όπως τα κινητά τηλέφωνα και το GPS μπορούν να αναπτύξουν περισσότερες υπηρεσίες εφαρμογών μέσω της τεχνολογίας υπολογιστικού νέφους. Περαιτέρω υπολογιστικό νέφος δεν είναι μόνο για λειτουργίες αναζήτησης και ανάλυσης δεδομένων. Στο μέλλον, όπως η ανάλυση της δομής του DNA, η αλληλουχία χαρτών γονιδίων, η ανάλυση καρκινικών κυττάρων κ.λπ., μπορούν να επιτευχθούν εύκολα μέσω αυτής της τεχνολογίας.

Η πρώτη μεγάλη κλίμακας κατανεμημένη τεχνολογία υπολογιστών είναι η προέλευση της έννοιας του «υπολογιστικού νέφους».

Η τεχνολογία Cloud παρέχει δυναμικούς, επεκτάσιμους και συχνά εικονικοποιημένους πόρους για την εξυπηρέτηση των χρηστών μέσω του Διαδικτύου. Οι χρήστες δεν απαιτείται να έχουν τη γνώση, την εξειδίκευση ή τον έλεγχο για την υποστήριξη της τεχνικής αρχιτεκτονικής τους στο cloud. Για να μπορέσουμε να κάνουμε αυτό το νέο οικονομικό μοντέλο "ως υπηρεσία", υπάρχουν αρκετά βασικά στοιχεία στην τεχνολογία cloud που πρέπει να αντιμετωπιστούν.

(1) Μια κοινή υποδομή cloud: ικανή να παρέχει ασφάλεια, επεκτασιμότητα και ποιότητα υπηρεσιών (QoS) σε επίπεδο επιχείρησης.

(2) Περιβάλλον ανάπτυξης: Διευκολύνετε τους προγραμματιστές εταιρικών εφαρμογών που κάποτε είχαν συνηθίσει να αναπτύσσουν μικρές αυτόνομες εφαρμογές να μεταβούν εύκολα στην ανάπτυξη ασφαλών εφαρμογών "πολλαπλών μισθωτών" που μπορούν να επεκταθούν σε δυνητικά εκατομμύρια χρήστες .

(3) Ένα λειτουργικό περιβάλλον μπορεί να παρέχει και να ενημερώνει απρόσκοπτα τις υπηρεσίες cloud χωρίς να επηρεάζει την εμπειρία του χρήστη.

(4) Ένας απλός τρόπος συνδυασμού πολλαπλών υπηρεσιών cloud για την ολοκλήρωση επαγγελματικών ή προσωπικών εργασιών. Η τεχνολογία Cloud είναι ένα υπολογιστικό μοντέλο στο οποίο διάφοροι υπολογιστικοί πόροι εικονικοποιούνται ως υπηρεσίες και κατανέμονται δυναμικά μέσω της σύνδεσης της τεχνολογίας Ιστού για να ταιριάζουν στις ανάγκες των χρηστών.

Εκπαιδευτική Υποενότητα : 2.4 Δομικά στοιχεία δικτύων

Τα παρακάτω είναι τα γενικά βήματα για τη δημιουργία μιας υπηρεσίας τελικού σημείου για ένα τελικό σημείο διεπαφής.

Δημιουργήστε ένα Network Load Balancer για την εφαρμογή σας στο VPC σας και διαμορφώστε το για κάθε υποδίκτυο (Ζώνη Διαθεσιμότητας) που το εξυπηρετεί. Μια συσκευή εξισορρόπησης φορτίου λαμβάνει αιτήματα από καταναλωτές υπηρεσιών και τους δρομολογεί στην υπηρεσία σας. Εναλλακτικά, μπορείτε να διαμορφώσετε ένα Application Load Balancer ως στόχο ενός Network Load Balancer, το οποίο στη συνέχεια μπορεί να δρομολογήσει αιτήματα στις υπηρεσίες σας. Για περισσότερες πληροφορίες, ανατρέξτε στον Οδηγό χρήστη Network Load Balancer .

Συνιστούμε να διαμορφώσετε τις υπηρεσίες σας σε όλες τις Ζώνες Διαθεσιμότητας σε μια περιοχή.

Δημιουργήστε μια διαμόρφωση υπηρεσίας τελικού σημείου VPC και καθορίστε μια Εξισορρόπηση φόρτου δικτύου.

Ακολουθούν ορισμένα γενικά βήματα μέσω των οποίων οι καταναλωτές υπηρεσιών μπορούν να συνδεθούν στην υπηρεσία σας.

- Παραχωρήστε δικαιώματα σε συγκεκριμένους καταναλωτές υπηρεσιών (λογαριασμοί AWS, χρήστες IAM και ρόλοι IAM) για τη δημιουργία συνδέσεων με τις υπηρεσίες τελικού σημείου σας.

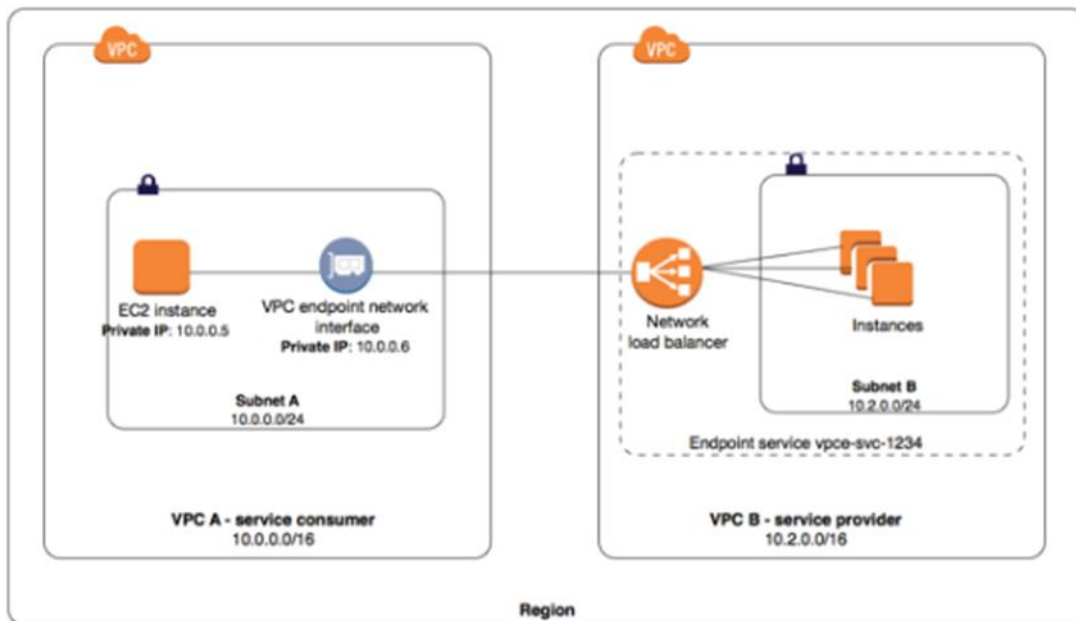
- Οι καταναλωτές υπηρεσιών στους οποίους έχουν παραχωρηθεί δικαιώματα μπορούν να δημιουργήσουν τελικά σημεία διεπαφής (προαιρετικά σε κάθε Ζώνη

Διαθεσιμότητας για την οποία έχετε διαμορφώσει υπηρεσίες) που συνδέονται με την υπηρεσία σας.

- Για να ενεργοποιήσετε τη σύνδεση, αποδεχτείτε το αίτημα σύνδεσης τερματικού σημείου διεπαφής. Από προεπιλογή, τα αιτήματα σύνδεσης πρέπει να γίνονται δεκτά με μη αυτόματο τρόπο. Ωστόσο, μπορείτε να διαμορφώσετε τις ρυθμίσεις αποδοχής της υπηρεσίας τελικού σημείου ώστε να αποδέχονται αυτόματα όλα τα αιτήματα σύνδεσης.

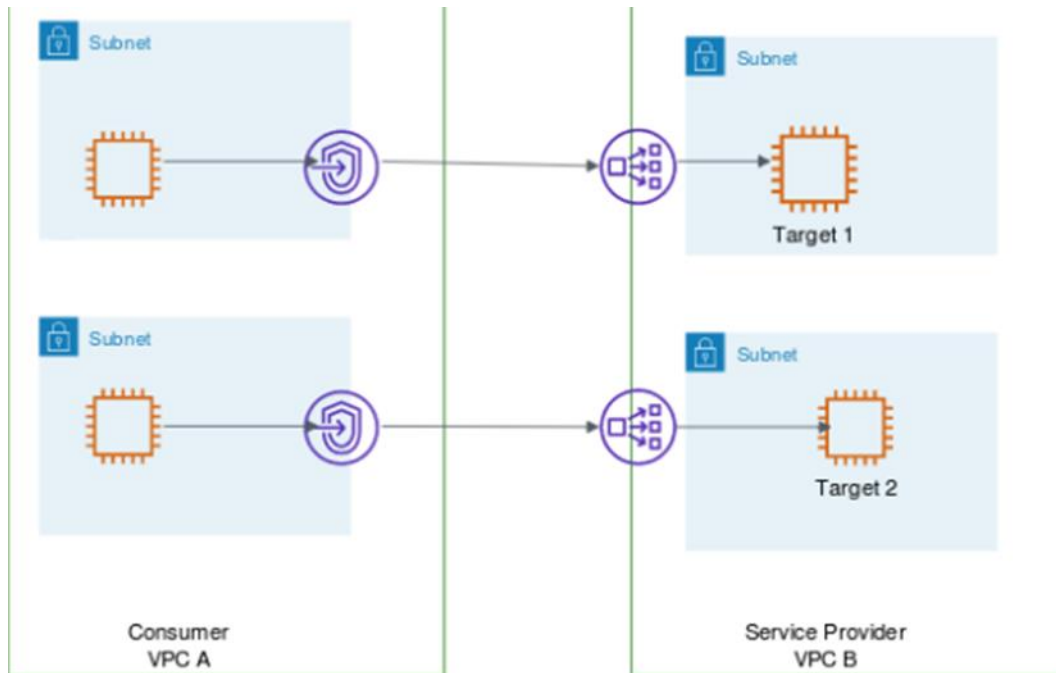
Ο συνδυασμός αδειών και ρυθμίσεων αποδοχής σας βοηθά να ελέγχετε ποιους καταναλωτές υπηρεσιών (αρχηγοί AWS) μπορούν να έχουν πρόσβαση στην υπηρεσία σας. Για παράδειγμα, μπορείτε να εκχωρήσετε δικαιώματα σε επιλεγμένους εντολείς που εμπιστεύεστε και να αποδέχεστε αυτόματα όλα τα αιτήματα σύνδεσης· μπορείτε επίσης να εκχωρήσετε δικαιώματα σε μια ευρύτερη ομάδα εντολέων και να αποδεχτείτε μη αυτόματα συγκεκριμένα αιτήματα σύνδεσης που εμπιστεύεστε.

Στο παρακάτω διάγραμμα, ο κάτοχος λογαριασμού του VPC B είναι πάροχος υπηρεσιών και υπάρχει μια υπηρεσία που εκτελείται σε παρουσίες στο υποδίκτυο B. Ο κάτοχος του VPC B έχει ένα τελικό σημείο υπηρεσίας (vrce-svc-1234) που σχετίζεται με έναν εξισορροπητή φόρτου δικτύου που δείχνει την παρουσία του υποδικτύου B ως στόχο. Οι παρουσίες στο υποδίκτυο A του VPC A χρησιμοποιούν τερματικά σημεία διεπαφής για πρόσβαση σε υπηρεσίες στο υποδίκτυο B.



Για χαμηλό λανθάνοντα χρόνο και ανοχή σφαλμάτων, συνιστάται ένα Network Load Balancer, που στοχεύει κάθε Ζώνη Διαθεσιμότητας σε μια Περιοχή AWS. Για να βοηθήσετε στην επίτευξη υψηλής διαθεσιμότητας για τους καταναλωτές υπηρεσιών που έχουν πρόσβαση σε υπηρεσίες χρησιμοποιώντας τοπικά ονόματα κεντρικών υπολογιστών DNS , μπορείτε να ενεργοποιήσετε την εξισορρόπηση φορτίου μεταξύ περιοχών. Με την εξισορρόπηση φορτίου μεταξύ ζωνών, το πρόγραμμα εξισορρόπησης φορτίου θα καταναείμει την κυκλοφορία μεταξύ των καταχωρημένων στόχων σε όλες τις ενεργοποιημένες Ζώνες Διαθεσιμότητας. Για περισσότερες πληροφορίες, ανατρέξτε στην ενότητα Εξισορρόπηση φορτίου μεταξύ περιοχών στον Οδηγό χρήσης του Network Load Balancer . Όταν είναι ενεργοποιημένη η εξισορρόπηση φορτίου μεταξύ περιοχών, οι λογαριασμοί ενδέχεται να χρεώνονται για περιφερειακή μεταφορά δεδομένων.

Στο παρακάτω διάγραμμα, το VPC B ανήκει σε έναν πάροχο υπηρεσιών και έχει διαμορφώσει ένα Network Load Balancer με στόχους σε δύο διαφορετικές Ζώνες Διαθεσιμότητας. Ο καταναλωτής υπηρεσιών (VPC A) έχει δημιουργήσει τελικά σημεία διεπαφής στις ίδιες δύο Ζώνες Διαθεσιμότητας στο VPC του. Τα αιτήματα για υπηρεσίες από παρουσίες στο VPC A μπορούν να χρησιμοποιούν οποιοδήποτε τελικό σημείο διεπαφής.



Για παραδείγματα διαμόρφωσης υπηρεσιών και παροχής δυνατότητας στους καταναλωτές υπηρεσιών να έχουν πρόσβαση σε υπηρεσίες μέσω της ανταλλαγής ομοτίμων VPC, ανατρέξτε στα Παραδείγματα: Χρήση των υπηρεσιών AWS PrivateLink και VPC Peering στον Οδηγό χρήστη Amazon VPC .

Όνομα DNS υπηρεσίας τελικού σημείου

Όταν δημιουργείτε μια υπηρεσία τελικού σημείου VPC, το AWS δημιουργεί ονόματα κεντρικού υπολογιστή DNS ειδικά για το τελικό σημείο που μπορείτε να χρησιμοποιήσετε για να επικοινωνήσετε με την υπηρεσία. Αυτά τα ονόματα περιλαμβάνουν το αναγνωριστικό τελικού σημείου VPC, το όνομα ζώνης διαθεσιμότητας και το όνομα περιοχής. Για παράδειγμα, `vrce-1234-abcdev-us-east-1.vrce-svc-123345.us-east-1.vrce.amazonaws.com`. Από προεπιλογή, οι καταναλωτές σας χρησιμοποιούν αυτό το όνομα DNS για πρόσβαση στην υπηρεσία, συνήθως απαιτώντας τροποποίηση της διαμόρφωσης της εφαρμογής.

Υπάρχει ένα προεπιλεγμένο όνομα DNS εάν η υπηρεσία τελικού σημείου προορίζεται για μια υπηρεσία AWS ή μια υπηρεσία διαθέσιμη στο AWS Marketplace. Για άλλες υπηρεσίες, οι πάροχοι υπηρεσιών μπορούν να διαμορφώσουν ιδιωτικά ονόματα DNS έτσι ώστε οι καταναλωτές να μπορούν να χρησιμοποιούν υπάρχοντα ονόματα DNS για πρόσβαση σε υπηρεσίες χωρίς να αλλάζουν τις εφαρμογές τους. Για περισσότερες

πληροφορίες, ανατρέξτε στην ενότητα Ιδιωτικά ονόματα DNS για υπηρεσίες τελικού σημείου .

Οι πάροχοι υπηρεσιών μπορούν να χρησιμοποιήσουν το κλειδί περιβάλλοντος υπό όρους `ec2:VpceServicePrivateDnsName` σε μια δήλωση πολιτικής IAM για να ελέγχουν ποια ιδιωτικά ονόματα DNS μπορούν να δημιουργηθούν. Για περισσότερες πληροφορίες, ανατρέξτε στο Amazon EC2-Defined Actions στον Οδηγό χρήστη IAM .

Τι είναι ένα πρωτόκολλο;

Τώρα που έχουμε μια αίσθηση για το τι είναι το Διαδίκτυο, ας εξετάσουμε ένα άλλο σημαντικό τσιτάτο στη δικτύωση υπολογιστών: πρωτόκολλο.

Τι είναι ένα πρωτόκολλο; Τα πρωτόκολλα του επιπέδου σύνδεσης ορίζουν έναν βολικό τρόπο παρουσίασης πληροφοριών για ανταλλαγή δικτύου, καθώς και το απαραίτητο σύνολο κανόνων που σας επιτρέπει να βελτιστοποιήσετε την αλληλεπίδραση των συνδρομητών.

Στο επίπεδο σύνδεσης δεδομένων, τα δεδομένα αντιμετωπίζονται ως σειριακή ροή bit. Πριν από τη μετάδοση μέσω φυσικών καναλιών, αυτή η ροή, σύμφωνα με την αρχή της μεταγωγής πακέτων, χωρίζεται σε "τμήματα", καθένα από τα οποία παρέχεται με μια κεφαλίδα που περιέχει ορισμένες πληροφορίες υπηρεσίας, π.χ. σχηματίζεται ένα πακέτο. Στο επίπεδο σύνδεσης δεδομένων, ένα πακέτο ονομάζεται πλαίσιο .

Η δομή της κεφαλίδας πλαισίου εξαρτάται από το σύνολο εργασιών που επιλύει το πρωτόκολλο. Η πολυπλοκότητα των πρωτοκόλλων καναλιών καθορίζεται σε μεγάλο βαθμό από την πολυπλοκότητα της τοπολογίας του δικτύου. Προφανώς, είναι πολύ πιο εύκολο να οργανωθεί η επικοινωνία μεταξύ δύο μόνο συνδρομητών παρά να εξορθολογιστεί η ανταλλαγή πληροφοριών σε δίκτυα όπου είναι δυνατή η παράλληλη αλληλεπίδραση πολλών ζευγών συνδρομητών. Επομένως, τα πρωτόκολλα καναλιών χωρίζονται εύκολα σε δύο ομάδες:

- πρωτόκολλα για συνδέσεις από σημείο σε σημείο.

- πρωτόκολλα για δίκτυα πολύπλοκων τοπολογιών.

Η σύνθεση της κεφαλίδας πλαισίου εξαρτάται από πολλούς παράγοντες που καθορίζονται από το σύνολο των λειτουργιών που εκτελεί το πρωτόκολλο. Ωστόσο, υπάρχει ένας αριθμός πεδίων πληροφοριών που συνήθως υπάρχουν στην κεφαλίδα του πλαισίου. Αυτά τα πεδία περιλαμβάνουν:

1. Ειδικά πεδία σχεδιασμένα για να ορίζουν όρια πλαισίου. Δεδομένου ότι τα σήματα μπορεί να περνούν συνεχώς μέσα από το φυσικό μέσο, οι προσαρμογείς δικτύου πρέπει να είναι σε θέση να καταλάβουν πότε ξεκινά και πότε τελειώνει μια μετάδοση πλαισίου.

2. Ένα πεδίο σχεδιασμένο για να ορίζει το πρωτόκολλο του επιπέδου δικτύου στο οποίο πρέπει να μεταφερθούν δεδομένα. Εφόσον οι μονάδες προγράμματος διαφόρων πρωτοκόλλων επιπέδων δικτύου μπορούν να λειτουργούν στον ίδιο υπολογιστή, τα πρωτόκολλα του επιπέδου σύνδεσης πρέπει να μπορούν να διανέμουν δεδομένα σε αυτά τα πρωτόκολλα.

3. Ένα άθροισμα ελέγχου (ή ένας ειδικός κωδικός) των περιεχομένων του πλαισίου, το οποίο επιτρέπει στην πλευρά λήψης να προσδιορίσει την παρουσία σφαλμάτων στα λαμβανόμενα δεδομένα. Η αρχή της χρήσης του είναι η εξής. Ο προσαρμογέας δικτύου του υπολογιστή αποστολής, αφού σχηματίσει το πλαίσιο, υπολογίζει την τιμή του αθροίσματος ελέγχου του με βάση τα περιεχόμενα και τοποθετεί αυτήν την τιμή στην κεφαλίδα του πλαισίου. Η πλευρά λήψης υπολογίζει επίσης το άθροισμα ελέγχου του ληφθέντος πλαισίου και το συγκρίνει με την τιμή που τοποθετείται στην κεφαλίδα. Εάν δεν ταιριάζουν, τότε αυτό σημαίνει ότι παρουσιάστηκε σφάλμα κατά τη μετάδοση του πλαισίου.

4. Πεδία που προορίζονται για τη διευθυνσιοδότηση συνδρομητών σε πολύπλοκα δίκτυα (καθορίζονται για πρωτόκολλα που χρησιμοποιούνται σε δίκτυα που βασίζονται σε σύνθετες τοπολογίες).

Network Protocols Πρωτόκολλα Δικτύου

Υπάρχουν πάνω από 7.000 πρωτόκολλα στον κόσμο και ο αριθμός συνεχίζει να αυξάνεται. Μιλάμε για τους πιο συχνά χρησιμοποιούμενους κανόνες για την

αλληλεπίδραση συσκευών στο δίκτυο. Ένα πρωτόκολλο δικτύου είναι ένα σύνολο κανόνων που διέπουν τον τρόπο αλληλεπίδρασης των συσκευών σε ένα δίκτυο. Για να είναι επιτυχής η αποστολή και η λήψη πληροφοριών, όλες οι συσκευές που συμμετέχουν στη διαδικασία πρέπει να αποδέχονται τους όρους του πρωτοκόλλου και να τους ακολουθούν. Στο δίκτυο, η υποστήριξή τους είναι ενσωματωμένη είτε στο υλικό (στο υλικό), είτε στο λογισμικό (στον κωδικό συστήματος) ή και στα δύο.

Για την αλληλεπίδραση των πρωτοκόλλων μεταξύ τους, υπάρχει το μοντέλο OSI, ή Open Systems Interconnection. Κυριολεκτικά, το όνομα μεταφράζεται ως "αλληλεπίδραση ανοιχτών συστημάτων". Το μοντέλο OSI είναι ένα μοντέλο που επιτρέπει σε διαφορετικά συστήματα επικοινωνίας να επικοινωνούν μεταξύ τους σύμφωνα με γενικά αποδεκτά πρότυπα. Μπορεί να συγκριθεί με τα αγγλικά, δηλαδή την παγκόσμια, καθολική γλώσσα στον κόσμο των δικτύων.

Το μοντέλο βασίζεται στην αρχή της διαίρεσης του συστήματος επικοινωνίας σε επτά ξεχωριστά επίπεδα.

Εάν υπάρχουν αστοχίες στη μετάδοση πληροφοριών, το μοντέλο βοηθά στον γρήγορο και εύκολο εντοπισμό του προβλήματος σε ένα συγκεκριμένο επίπεδο και επιταχύνει σημαντικά τη διαδικασία επαναφοράς του συστήματος σε λειτουργική ικανότητα.

Το μοντέλο OSI είναι το πρότυπο αναφοράς, αλλά αυτή τη στιγμή είναι ξεπερασμένο, καθώς τα σύγχρονα πρωτόκολλα λειτουργούν σε πολλά επίπεδα του μοντέλου OSI ταυτόχρονα. Το μοντέλο OSI έχει αντικατασταθεί από το μοντέλο TCP / IP, βάσει του οποίου λειτουργούν οι περισσότερες συσκευές στον σύγχρονο κόσμο.

Πρωτόκολλο TCP/IP

Το TCP IP είναι ένα μοντέλο δικτύου που βασίζεται σε μια στοίβα πρωτοκόλλων που περιγράφουν τη μεταφορά ψηφιακών δεδομένων μεταξύ συσκευών στο Διαδίκτυο. Τα πρωτόκολλα αναπτύχθηκαν στη δεκαετία του '70 του περασμένου αιώνα για να εξασφαλίσουν την ανταλλαγή πληροφοριών μεταξύ διαφορετικών δικτύων. Το μοντέλο δικτύου TCP IP είναι ένα σύνολο κανόνων που διέπουν τη μεταφορά

δεδομένων μεταξύ συσκευών: e-mail, αρχεία πολυμέσων, απομακρυσμένη πρόσβαση σε ένα σταθμό εργασίας.

Το μοντέλο δικτύου TCP IP περιέχει τυποποιημένες συμβάσεις δρομολόγησης και δικτύωσης που επιτρέπουν την απρόσκοπτη επικοινωνία μεταξύ των κεντρικών υπολογιστών. Το κύριο πλεονέκτημα του μοντέλου δικτύου είναι η ανεξαρτησία μεταξύ πλατφορμών και υλικού: η στοίβα πρωτοκόλλου μπορεί να χρησιμοποιηθεί σε υπολογιστές οποιασδήποτε διαμόρφωσης και σε διαφορετικά λειτουργικά συστήματα.

Το μοντέλο δικτύου αναπτύχθηκε από μια ομάδα με επικεφαλής τους Vinton Cerf και Bob Kahn το 1972. Προϋποθέσεις για τη δημιουργία του μοντέλου ήταν οι απαιτήσεις του Υπουργείου Άμυνας ότι το δίκτυο συνέχιζε να λειτουργεί υπό οποιεσδήποτε συνθήκες, ανεξάρτητα από εξωτερικούς παράγοντες. Η πρώτη επίδειξη εργασίας περιελάμβανε την παράδοση ενός πακέτου δεδομένων σε τρία διαφορετικά κοινά δίκτυα, διανύοντας πάνω από 150.000 km χωρίς απώλεια δεδομένων. Το μοντέλο δικτύου του TCP IP έχει κάποιες ομοιότητες με το μοντέλο OSI, αλλά το επίπεδο εφαρμογής συνδυάζει τρία. Το TCP/IP είναι μια οικογένεια πρωτοκόλλων επικοινωνίας που χρησιμοποιούνται στο Διαδίκτυο. Το όνομα TCP/IP συνδυάζει δύο σημαντικά πρωτόκολλα που ήταν τα πρώτα αυτής της οικογένειας.

Από ιστορικής αναφοράς, τα πρωτόκολλα TCP (Transmission Control Protocol) και IP (Internet Protocol) αναπτύχθηκαν από το Υπουργείο Άμυνας των ΗΠΑ, η στοίβα περιλαμβάνει πολλά πρωτόκολλα που χρησιμοποιούνται για συγκεκριμένες εργασίες, όπως η αποστολή email μέσω SMTP. Το TCP IP είναι το επίσημο πρότυπο για το παγκόσμιο Διαδίκτυο, βασίζεται στο μοντέλο δικτύου OSI και έχει τέσσερα επίπεδα, καθένα από τα οποία εκτελεί τις δικές του λειτουργίες.

Η αρχιτεκτονική επιπέδου του TCP IP περιγράφεται στο RFC 1122, αλλά σε ορισμένες άλλες πηγές υπάρχουν πέντε επίπεδα, αφού το φυσικό εκχωρείται ξεχωριστά.

Το σχήμα λειτουργίας αυτών των πρωτοκόλλων είναι μεταξύ τους, σε μια στοίβα (στοίβα). Το ένα είναι πάνω, το άλλο, αντίστοιχα κάτω. Το επάνω μέρος χρησιμοποιεί μηχανισμούς ενθυσιάκωσης. Τα πρωτόκολλα αυτών των επιπέδων υλοποιούν όλα τα χαρακτηριστικά της αρχιτεκτονικής δικτύου OSI.

Πλεονεκτήματα του TCP/IP:

- ελάχιστο κόστος για τη δημιουργία πρωτοκόλλων αλληλεπίδρασης,
- μεγάλη απλοποίηση της διαδικασίας δρομολόγησης,
- τη δυνατότητα κατασκευής και λειτουργίας IVS μεγάλης κλίμακας,
- αλληλεπίδραση διαφορετικών δικτύων χρησιμοποιώντας απλούς αλγόριθμους αντιστοίχισης.

Μειονεκτήματα του TCP/IP:

- την ανάγκη για ευρείες ειδικές διαύλους επικοινωνίας,
- το πρόβλημα της συναρμολόγησης πακέτων που φτάνουν στο στρώμα μεταφοράς που πρέπει να λυθεί,
- ο κίνδυνος απώλειας ενός μηνύματος λόγω της πιθανότητας απώλειας ενός από τα πακέτα του,
- υψηλή πολυπλοκότητα εφαρμογών χρήστη.

Η οικογένεια TCP / IP είναι περιζήτητη κατά τη δημιουργία παγκόσμιων δικτύων, όπου είναι απαραίτητος ο συντονισμός της λειτουργίας άλλων μικρότερων δικτύων. Βασικά, χρησιμοποιούνται διαφορετικοί τύποι εξοπλισμού για αυτό. Σε τέτοιες συνθήκες, τα πρωτόκολλα TCP / IP είναι αυτά που φέρνουν το μεγαλύτερο αποτέλεσμα.



Το πρωτόκολλο DHCP

Με τον όρο DHCP (Dynamic Host Configuration Protocol) αναφερόμαστε σε ένα μηχανισμό διαχείρισης TCP/IP πρωτοκόλλων. Το DHCP είναι ένα πρωτόκολλο για την

αυτοματοποίηση της εκχώρησης μιας διεύθυνσης IP σε έναν πελάτη. Χρησιμοποιείται ευρέως στα σύγχρονα δίκτυα. Στο άρθρο, θα εξετάσουμε τις αρχές λειτουργίας, τη διαδικασία DORA, τις κύριες επιλογές και άλλες πτυχές του πρωτοκόλλου.

Το **Dynamic Host Configuration Protocol (DHCP)** είναι ένα πρωτόκολλο πελάτη/διακομιστή που παρέχει αυτόματα σε έναν κεντρικό υπολογιστή πρωτοκόλλου Internet (IP) τη διεύθυνση IP του και άλλες σχετικές πληροφορίες διαμόρφωσης, όπως η μάσκα υποδικτύου και η προεπιλεγμένη πύλη. Τα RFC 2131 και 2132 ορίζουν το DHCP ως πρότυπο Internet Engineering Task Force (IETF) που βασίζεται στο Bootstrap Protocol (BOOTP), ένα πρωτόκολλο με το οποίο το DHCP μοιράζεται πολλές λεπτομέρειες υλοποίησης. Το DHCP επιτρέπει στους κεντρικούς υπολογιστές να λαμβάνουν τις απαιτούμενες πληροφορίες διαμόρφωσης TCP/IP από έναν διακομιστή DHCP.

Ο Windows Server 2016 περιλαμβάνει διακομιστή DHCP, ο οποίος είναι ένας προαιρετικός ρόλος διακομιστή δικτύωσης που μπορείτε να αναπτύξετε στο δίκτυό σας για να εκμισθώσετε διευθύνσεις IP και άλλες πληροφορίες σε πελάτες DHCP. Όλα τα λειτουργικά συστήματα-πελάτες που βασίζονται σε Windows περιλαμβάνουν τον πελάτη DHCP ως μέρος του TCP/IP και ο πελάτης DHCP είναι ενεργοποιημένος από προεπιλογή.

Γιατί να χρησιμοποιήσετε το DHCP;

Κάθε συσκευή σε δίκτυο που βασίζεται σε TCP/IP πρέπει να έχει μια μοναδική διεύθυνση IP unicast για πρόσβαση στο δίκτυο και στους πόρους του. Χωρίς DHCP, οι διευθύνσεις IP για νέους υπολογιστές ή υπολογιστές που μετακινούνται από το ένα υποδίκτυο στο άλλο πρέπει να ρυθμιστούν με μη αυτόματο τρόπο. Οι διευθύνσεις IP για υπολογιστές που αφαιρούνται από το δίκτυο πρέπει να ανακτηθούν με μη αυτόματο τρόπο.

Με το DHCP, όλη αυτή η διαδικασία αυτοματοποιείται και διαχειρίζεται κεντρικά. Ο διακομιστής DHCP διατηρεί μια ομάδα διευθύνσεων IP και μισθώνει μια διεύθυνση σε οποιονδήποτε πελάτη με δυνατότητα DHCP κατά την εκκίνηση του στο δίκτυο. Επειδή οι διευθύνσεις IP είναι δυναμικές (μισθωμένες) και όχι στατικές (μόνιμα

εκχωρημένες), οι διευθύνσεις που δεν χρησιμοποιούνται πλέον επιστρέφονται αυτόματα στο pool για ανακατανομή.

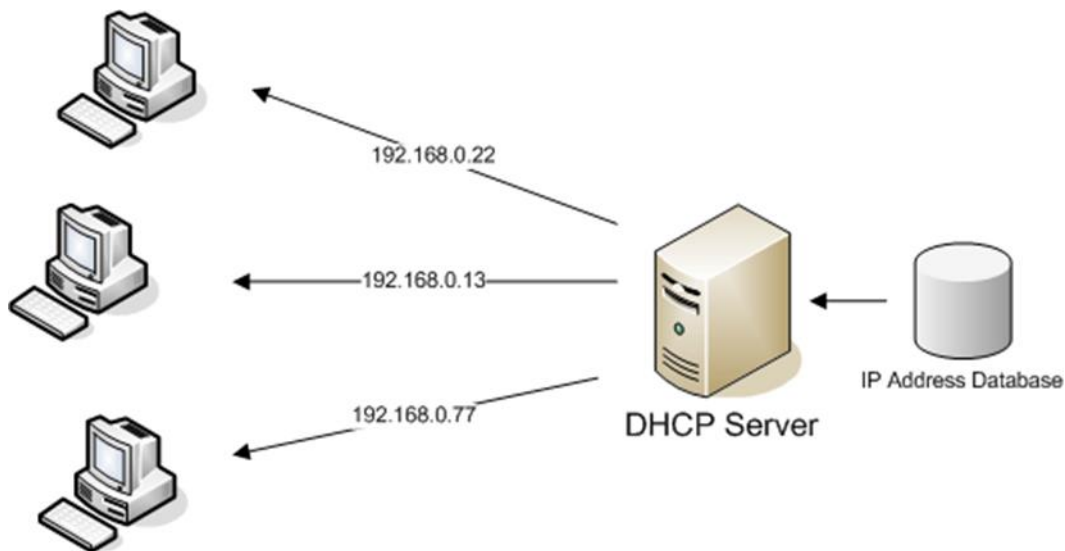
Ο διαχειριστής δικτύου δημιουργεί διακομιστές DHCP που διατηρούν πληροφορίες διαμόρφωσης TCP/IP και παρέχουν διαμόρφωση διεύθυνσης σε πελάτες με δυνατότητα DHCP με τη μορφή προσφοράς μίσθωσης.

Ο διακομιστής DHCP αποθηκεύει τις πληροφορίες διαμόρφωσης σε μια βάση δεδομένων που περιλαμβάνει:

- Έγκυρες παράμετροι διαμόρφωσης TCP/IP για όλους τους πελάτες στο δίκτυο.
- Έγκυρες διευθύνσεις IP, που διατηρούνται σε μια ομάδα για εκχώρηση σε πελάτες, καθώς και εξαιρούμενες διευθύνσεις.
- Δεσμευμένες διευθύνσεις IP που σχετίζονται με συγκεκριμένους πελάτες DHCP. Αυτό επιτρέπει τη συνεπή εκχώρηση μιας μεμονωμένης διεύθυνσης IP σε έναν πελάτη DHCP.
- Η διάρκεια μίσθωσης ή το χρονικό διάστημα για το οποίο μπορεί να χρησιμοποιηθεί η διεύθυνση IP πριν απαιτείται ανανέωση μίσθωσης.

Ένας πελάτης με δυνατότητα DHCP, με την αποδοχή μιας προσφοράς μίσθωσης, λαμβάνει:

- Μια έγκυρη διεύθυνση IP για το υποδίκτυο στο οποίο συνδέεται.
- Ζητούμενες επιλογές DHCP, οι οποίες είναι πρόσθετες παράμετροι που ένας διακομιστής DHCP έχει ρυθμιστεί να εκχωρεί σε πελάτες. Μερικά παραδείγματα επιλογών DHCP είναι ο δρομολογητής (προεπιλεγμένη πύλη), οι διακομιστές DNS και το όνομα τομέα DNS.



Οφέλη του DHCP

Το DHCP παρέχει τα ακόλουθα πλεονεκτήματα.

- Αξιοπίστη διαμόρφωση διεύθυνσης IP . Το DHCP ελαχιστοποιεί τα σφάλματα διαμόρφωσης που προκαλούνται από τη μη αυτόματη ρύθμιση παραμέτρων διεύθυνσης IP, όπως τυπογραφικά σφάλματα ή διενέξεις διευθύνσεων που προκαλούνται από την εκχώρηση μιας διεύθυνσης IP σε περισσότερους από έναν υπολογιστές ταυτόχρονα.
- Μειωμένη διαχείριση δικτύου.

Μέθοδοι διανομής DHCP

Ανάλογα με τη διαμόρφωση, ο διακομιστής DHCP μπορεί να λειτουργήσει με τρεις τρόπους:

Δυναμική κατανομή

Όταν ένας διακομιστής DHCP έχει ρυθμιστεί να χρησιμοποιεί δυναμική κατανομή, σημαίνει ότι χρησιμοποιεί πολιτική μίσθωσης. Με αυτόν τον τρόπο, όταν μια εκχωρημένη διεύθυνση IP από ένα διαθέσιμο χώρο συγκέντρωσης δεν χρησιμοποιείται πλέον, θα μετακινηθεί ξανά στο χώρο συγκέντρωσης, καθιστώντας τη διαθέσιμη σε κάποιον άλλο. Το πλεονέκτημα αυτής της μεθόδου είναι ότι οι διευθύνσεις IP χρησιμοποιούνται στο μέγιστο - όταν δεν χρησιμοποιούνται πλέον

από τον πελάτη, είναι άμεσα διαθέσιμες σε άλλους. Το μειονέκτημα αυτής της μεθόδου είναι ότι ο πελάτης θα έχει πάντα μια τυχαία διεύθυνση IP.

Αυτόματη διανομή

Η μέθοδος αυτόματης κατανομής είναι πολύ παρόμοια με τη μέθοδο δυναμικής εκχώρησης - μόλις συνδεθεί ένας πελάτης, ο διακομιστής DHCP του παρέχει μια διεύθυνση IP από μια ομάδα διευθύνσεων IP. Ωστόσο, όταν χρησιμοποιείται η αυτόματη κατανομή, ο διακομιστής DHCP διατηρεί μια βάση δεδομένων με προηγούμενες επιχορηγήσεις IP και προσπαθεί να δώσει στον πελάτη την ίδια διεύθυνση IP που χρησιμοποίησε τελευταία, εάν είναι διαθέσιμη.

Στατική κατανομή

Η μέθοδος στατικής διανομής είναι πολύ δημοφιλής στα σημερινά δίκτυα ISP που δεν χρησιμοποιούν μεθόδους dial-up. Με τη στατική εκχώρηση, ο διακομιστής DHCP διατηρεί μια βάση δεδομένων με όλες τις διευθύνσεις MAC των τοπικών δικτύων των πελατών και τους δίνει μια διεύθυνση IP μόνο εάν η διεύθυνση MAC τους βρίσκεται στη βάση δεδομένων. Έτσι, οι πελάτες μπορούν να είναι σίγουροι ότι θα λαμβάνουν την ίδια διεύθυνση IP κάθε φορά. Ένας διακομιστής DHCP μπορεί να ρυθμιστεί ώστε να λειτουργεί χρησιμοποιώντας έναν συνδυασμό μεθόδων διανομής. Για παράδειγμα, σε ένα δημόσιο δίκτυο Wi-Fi, όλοι οι γνωστοί κεντρικοί υπολογιστές και οι επαναλαμβανόμενοι πελάτες μπορούν να χρησιμοποιούν στατική κατανομή, ενώ οι επισκέπτες χρησιμοποιούν δυναμική κατανομή. Με αυτόν τον τρόπο, οι γνωστοί κεντρικοί υπολογιστές μπορούν πάντα να χρησιμοποιούν την ίδια διεύθυνση IP και η ομάδα των διευθύνσεων IP είναι εξίσου προσβάσιμη σε όλους.

HTTP(S) - Πρωτόκολλο μεταφοράς υπερκειμένου

Το HTTP είναι η ραχοκοκαλιά του Διαδικτύου και χρησιμοποιείται για τη λήψη ιστοσελίδων χρησιμοποιώντας συνδέσμους υπερκειμένου. Αναφέρεται σε πρωτόκολλα εφαρμογών και λειτουργεί πάνω από άλλα επίπεδα της στοίβας πρωτοκόλλων δικτύου.

Συνήθως, το πρωτόκολλο HTTP περιλαμβάνει έναν υπολογιστή-πελάτη (όπως ο υπολογιστής σας) που στέλνει ένα αίτημα σε έναν διακομιστή, ο οποίος στη συνέχεια

επιστρέφει μια απάντηση. Κάθε αίτημα HTTP περιλαμβάνει μια σειρά από κωδικοποιημένα δεδομένα που περιέχουν διάφορες πληροφορίες, όπως:

- http έκδοση,
- url,
- Μέθοδος αιτήματος HTTP - ένδειξη αναμονής για αίτημα από τον διακομιστή (για παράδειγμα, αιτήματα PUT και GET),
- Κεφαλίδα - μεταφέρει βασικές πληροφορίες σχετικά με το αίτημα και περιέχει ζεύγη κλειδιών-τιμών,
- Φορέας αιτήματος (προαιρετικό, πρόκειται για οποιαδήποτε πληροφορία αποστέλλεται).

Μετά τη λήψη του αιτήματος, ο διακομιστής πρέπει να απαντήσει. Η τυπική του δομή περιλαμβάνει: έναν κωδικό κατάστασης, μια κεφαλίδα και ένα σώμα απόκρισης.

Ένας κωδικός κατάστασης αιτήματος HTTP είναι ένας κωδικός τριών χαρακτήρων που συνήθως υποδεικνύει την επιτυχία του αιτήματος. Χωρίζονται σε πέντε κύρια μπλοκ:

1xx* Πληροφορίες (Πληροφοριακά),

2xx Επιτυχία (Επιτυχία),

3xx Ανακατεύθυνση,

4xx Σφάλμα πελάτη,

5xx Σφάλμα διακομιστή,

*Το XX σημαίνει αριθμούς από το 00 έως το 99.

Παρόμοια με ένα αίτημα, η απάντηση έχει μια κεφαλίδα που περιέχει διάφορες πληροφορίες, όπως τη γλώσσα των δεδομένων που αποστέλλονται. Στις περισσότερες περιπτώσεις, περιέχει δεδομένα HTML που το πρόγραμμα περιήγησης ιστού του πελάτη αποδίδει σε μια σελίδα.

Όταν μιλάμε για HTTP, δεν μπορούμε να μην αναφέρουμε μια σημαντική πτυχή - την ανασφάλεια του πρωτοκόλλου. Όταν μεταδίδονται δεδομένα, όλα γίνονται ανοιχτά, με αποτέλεσμα ένας εισβολέας να μπορεί να υποκλέψει τα δεδομένα. Για την

εξάλειψη αυτού του προβλήματος, αναπτύχθηκε το πρωτόκολλο HTTPS. Μια λεπτομερής σύγκριση αυτών των πρωτοκόλλων βρίσκεται στο ιστολόγιό μας .

Το SSH είναι το κύριο μέσο σύνδεσης με διακομιστές

Το SSH ή το Secure Shell είναι ένα ασφαλές πρωτόκολλο που χρησιμοποιείται ως το κύριο μέσο σύνδεσης με διακομιστές. Με αυτό, κατά τη σύνδεση στον διακομιστή, ο χρήστης συνδέεται σε έναν ήδη υπάρχοντα λογαριασμό, όπου εκτελούνται όλες οι εντολές που αποστέλλονται.

Στον πίνακα ελέγχου Selectel, ο χρήστης μπορεί να ανεβάσει το δικό του κλειδί SSH για να αυξήσει την αξιοπιστία και την ασφάλεια της σύνδεσης ή να το δημιουργήσει επιτόπου. Μάθετε περισσότερα στη γνωσιακή μας βάση .

Πρωτόκολλα παλαιού τύπου: Telnet, Gopher, FTP

Σήμερα υπάρχουν περισσότερα από 7.000 διαφορετικά πρωτόκολλα. Το δίκτυο εξελίσσεται συνεχώς, επομένως ορισμένα πρωτόκολλα γίνονται απαρχαιωμένα - για παράδειγμα, Gopher, FTP, Telnet. Ας ρίξουμε μια πιο προσεκτική ματιά στο τελευταίο.

Το Telnet είναι ένα παλιό αλλά πολύ αξιόπιστο πρωτόκολλο επικοινωνίας. Αρχικά αναπτύχθηκε ως πρωτόκολλο εξομοίωσης τερματικού προσανατολισμού χαρακτήρων που χρησιμοποιείται στο περιβάλλον UNIX. Σήμερα, το Telnet χρησιμοποιείται ευρέως για τη διαχείριση συστήματος δρομολογητών, μεταγωγέων και απομακρυσμένων διακομιστών, καθώς και για βασική επικοινωνία κειμένου όπου δεν απαιτούνται γραφικά.

Το πρωτόκολλο Network Address Translation.

Ο σχεδιασμός δικτύου χρησιμοποιεί συνήθως ιδιωτικές διευθύνσεις IP 10.0.0.0/8, 172.16.0.0/12 και 192.168.0.0/16. Χρησιμοποιούνται εντός του δικτύου μιας τοποθεσίας ή οργανισμού για την υποστήριξη τοπικής επικοινωνίας μεταξύ συσκευών και όχι για δρομολόγηση σε ένα παγκόσμιο δίκτυο. Για να μπορεί μια συσκευή με διεύθυνση IPv4 να έχει πρόσβαση σε άλλες συσκευές ή πόρους μέσω του Διαδικτύου, η ιδιωτική της διεύθυνση πρέπει να αντιστοιχιστεί σε μια δημόσια διεύθυνση. Τέτοια μετάφραση είναι το κύριο πράγμα που κάνει το NAT, ένας ειδικός μηχανισμός μετατροπής ιδιωτικών διευθύνσεων σε δημόσιες διευθύνσεις.

Η υποστήριξη NAT, σε συνδυασμό με ιδιωτικές διευθύνσεις IPv4, έχει γίνει ένας αποτελεσματικός τρόπος διατήρησης των δημόσιων διευθύνσεων IPv4. Ο μηχανισμός επιτρέπει σε πολλές συσκευές, η καθεμία με τη δική της ιδιωτική διεύθυνση, να χρησιμοποιούν μια ενιαία δημόσια διεύθυνση IPv4. Ένα πρόσθετο πλεονέκτημα του NAT είναι ότι αυξάνει την ασφάλεια και το απόρρητο του δικτύου αποκρύπτοντας ιδιωτικές διευθύνσεις IPv4.

Ένας δρομολογητής NAT μπορεί να διαμορφωθεί με μία ή περισσότερες δημόσιες διευθύνσεις IPv4, η οποία ονομάζεται συγκέντρωση NAT. Όταν μια συσκευή στέλνει κίνηση από το εσωτερικό δίκτυο στο εξωτερικό δίκτυο, ο δρομολογητής μεταφράζει την εσωτερική του διεύθυνση IPv4 σε μία από τις ομαδοποιημένες διευθύνσεις. Ως αποτέλεσμα αυτού του μηχανισμού, όλη η κίνηση που εξερχόμενη από το δίκτυο «βλέπεται» από εξωτερικές συσκευές με δημόσια διεύθυνση IPv4, η οποία μπορεί να ονομαστεί διεύθυνση IP NAT.

Ο δρομολογητής NAT λειτουργεί στην άκρη ενός δικτύου stub που ονομάζεται δίκτυο Stub. Συνδέεται στο γειτονικό δίκτυο με μία σύνδεση, έχει μία είσοδο και μία έξοδο. Όταν δημιουργηθεί σύνδεση μεταξύ συσκευών εντός και εκτός του δικτύου Stub, το πακέτο αποστέλλεται στον δρομολογητή συνόρων, ο οποίος εκτελεί τη διαδικασία NAT. Αυτή η διαδικασία μετατρέπει την εσωτερική ιδιωτική διεύθυνση της συσκευής σε μια δημόσια εξωτερική διεύθυνση με δυνατότητα δρομολόγησης.

Πλεονεκτήματα και μειονεκτήματα του NAT

Κατανοώντας πώς λειτουργεί το NAT, μπορεί κανείς να ξεχωρίσει μια σειρά από σοβαρά πλεονεκτήματα που παρέχει αυτός ο μηχανισμός για τη δικτύωση.

Μεταξύ των κύριων πλεονεκτημάτων είναι οι ακόλουθες ιδιότητες της Μετάφρασης Διευθύνσεων Δικτύου:

1. Διατήρηση του καταχωρημένου συστήματος διευθύνσεων επιτρέποντας την ιδιωτικοποίηση εσωτερικών δικτύων. Όταν χρησιμοποιείτε NAT, είναι δυνατό για εσωτερικούς κεντρικούς υπολογιστές να χρησιμοποιούν μια ενιαία δημόσια διεύθυνση IPv4 για εξωτερικές συνδέσεις. Αυτό το σχήμα απαιτεί μικρό αριθμό

εξωτερικών διευθύνσεων για την υποστήριξη σημαντικού αριθμού εσωτερικών κεντρικών υπολογιστών.

2. Αύξηση της ευελιξίας επικοινωνίας με το Διαδίκτυο. Η αξιόπιστη συνδεσιμότητα δικτύου επιτυγχάνεται μέσω πολλαπλών ομάδων διευθύνσεων, ομάδων εξισορρόπησης φορτίου και δημιουργίας αντιγράφων ασφαλείας.

3. Υποστήριξη για συνεπή λειτουργία σχημάτων διευθυνσιοδότησης εσωτερικού δικτύου. Εάν το δίκτυο δεν χρησιμοποιεί NAT και ιδιωτικές διευθύνσεις IPv4, τότε για να αλλάξετε το γενικό σχήμα διευθύνσεων, είναι απαραίτητο να ανακατευθύνετε ολόκληρο το συγκρότημα κεντρικών υπολογιστών. Αυτό μπορεί να αυξήσει σημαντικά το κόστος της ανακατεύθυνσης. Όταν χρησιμοποιείτε το NAT, είναι δυνατή η διατήρηση του τρέχοντος συστήματος ιδιωτικών διευθύνσεων, γεγονός που καθιστά πολύ πιο εύκολη την πραγματοποίηση αλλαγών στο σύστημα δημόσιων διευθύνσεων. Στην πράξη, αυτό σημαίνει, για παράδειγμα, τη δυνατότητα αλλαγής του παρόχου της εταιρείας χωρίς να κάνει αλλαγές στους εσωτερικούς πελάτες της.

4. Διατηρήστε υψηλό επίπεδο ασφάλειας δικτύου. Όταν χρησιμοποιούν NAT, τα ιδιωτικά δίκτυα δεν μεταφράζουν την εσωτερική τους τοπολογία και διευθύνσεις, γεγονός που αυξάνει την αξιοπιστία τους. Λάβετε υπόψη ότι το NAT δεν αντικαθιστά λύσεις ασφάλειας δικτύου, όπως ένα τείχος προστασίας.

Ένα χαρακτηριστικό του NAT είναι η οργάνωση της άμεσης αλληλεπίδρασης των κεντρικών υπολογιστών στο Διαδίκτυο με μια συσκευή που υποστηρίζει τη μετάφραση διευθύνσεων δικτύου και όχι με τον πραγματικό κεντρικό υπολογιστή στο τοπικό δίκτυο.

Εδώ, αποκαλύπτονται ορισμένες αδυναμίες του μηχανισμού, όπως:

- Μια ορισμένη μείωση στην απόδοση του δικτύου λόγω της αύξησης της καθυστέρησης μεταγωγής κατά τον μετασχηματισμό πακέτων κάθε διεύθυνσης IPv4. Η υποβάθμιση της απόδοσης μπορεί να είναι ευαίσθητη στο VoIP και σε άλλα πρωτόκολλα σε πραγματικό χρόνο.
- Απαιτείται απώλεια διευθυνσιοδότησης από άκρο σε άκρο για έναν αριθμό εφαρμογών και πρωτοκόλλων. Εάν η εφαρμογή δεν χρησιμοποιεί αναγνωρισμένο

όνομα τομέα, αλλά φυσικές διευθύνσεις, τότε τα πακέτα δεν φτάνουν στους παραλήπτες μέσω του δρομολογητή NAT. Σε ορισμένες περιπτώσεις, το πρόβλημα επιλύεται χρησιμοποιώντας στατικές αντιστοιχίσεις NAT.

- Απώλεια ανίχνευσης IPv4 από άκρο σε άκρο. Οι πολλές αλλαγές διεύθυνσης πακέτων καθιστούν δύσκολο τον εντοπισμό, τον εντοπισμό και την αντιμετώπιση προβλημάτων.
- Περιπλέκοντας το IPsec και άλλα πρωτόκολλα σήραγγας αλλάζοντας τις τιμές της κεφαλίδας, καθιστώντας τους ελέγχους ακεραιότητας πιο δύσκολους.
- Δυνατότητα διακοπής πρωτοκόλλων χωρίς κατάσταση ή υπηρεσιών που απαιτούν εκκίνηση συνδέσεων TCP από το εξωτερικό δίκτυο, εάν ο δρομολογητής NAT δεν έχει διαμορφωθεί για να τις υποστηρίξει.

Ταυτόχρονα, το NAT παραμένει ένας αποτελεσματικός και βολικός μηχανισμός για δικτύωση με χρήση διευθύνσεων IPv4.

Συμπέρασμα

Εξετάσαμε δημοφιλή πρωτόκολλα για την αλληλεπίδραση συσκευών και προγραμμάτων στο δίκτυο. Στην πραγματική ζωή, εκατοντάδες πρωτόκολλα χρησιμοποιούνται μόνο στον υπολογιστή σας, καθώς το καθένα εκτελεί μια συγκεκριμένη εργασία. Μπορείτε να μάθετε για τις εργασίες και τις αρχές του πρωτοκόλλου σε ένα ειδικό έγγραφο που ονομάζεται πρότυπο RFC.

Domain Name Service

Το Σύστημα Ονομάτων Τομέα (DNS) είναι ένα από τα βιομηχανικά πρότυπα πρωτόκολλα που συνθέτουν το TCP/IP και μαζί ο πελάτης DNS και ο διακομιστής DNS παρέχουν υπηρεσίες ανάλυσης ονομάτων υπολογιστή και χρήση. Στον Windows Server 2016, το DNS είναι ένας ρόλος διακομιστή που μπορεί να εγκατασταθεί χρησιμοποιώντας εντολές Server Manager ή Windows PowerShell. Όταν εγκαθιστάτε ένα νέο σύμπλεγμα δομών και τομέα, το Active Directory DNS εγκαθίσταται αυτόματα με την Active Directory ως διακομιστή καθολικού καταλόγου για το σύμπλεγμα δομών και τον τομέα.

Οι υπηρεσίες τομέα Active Directory (AD DS) χρησιμοποιούν DNS ως μηχανισμό για τον εντοπισμό ενός ελεγκτή τομέα. Κατά την εκτέλεση οποιασδήποτε από τις βασικές λειτουργίες της υπηρεσίας καταλόγου Active Directory, όπως έλεγχος ταυτότητας, ενημέρωση ή αναζήτηση, οι υπολογιστές χρησιμοποιούν DNS για να βρουν ελεγκτές τομέα Active Directory. Επίσης, οι ελεγκτές τομέα χρησιμοποιούν DNS για να αναζητήσουν ο ένας τον άλλον.

Η υπηρεσία DNS Client περιλαμβάνεται σε όλες τις εκδόσεις πελάτη και διακομιστή του λειτουργικού συστήματος Windows και εκτελείται από προεπιλογή όταν είναι εγκατεστημένο το λειτουργικό σύστημα. Όταν ρυθμίζετε μια σύνδεση δικτύου TCP/IP με μια διεύθυνση IP διακομιστή DNS, ο πελάτης DNS ζητά από τον διακομιστή DNS να εντοπίσει ελεγκτές τομέα και να επιλύσει τα ονόματα των υπολογιστών σε διευθύνσεις IP. Για παράδειγμα, όταν ένας χρήστης δικτύου με λογαριασμό χρήστη Active Directory συνδέεται σε έναν τομέα Active Directory, η υπηρεσία DNS Client υποβάλλει ερώτημα στον διακομιστή DNS για να βρει τον ελεγκτή τομέα για τον τομέα Active Directory. Όταν ο διακομιστής DNS απαντήσει στο ερώτημα και παρέχει στον Πελάτη τη διεύθυνση IP του ελεγκτή τομέα, ο πελάτης επικοινωνεί με τον ελεγκτή τομέα και η διαδικασία ελέγχου ταυτότητας μπορεί να ξεκινήσει.

Οι υπηρεσίες Windows Server 2016 DNS Server και DNS Client χρησιμοποιούν το πρωτόκολλο DNS που περιλαμβάνεται στη σουίτα πρωτοκόλλων TCP/IP.

Πολιτικές DNS

- Μπορείτε να χρησιμοποιήσετε την πολιτική DNS για να ελέγξετε την επισκεψιμότητα με βάση τη Γεω-Τοποθεσία, έξυπνες αποκρίσεις DNS με βάση την ώρα της ημέρας, να ελέγξετε έναν μεμονωμένο διακομιστή DNS που έχει διαμορφωθεί για χωριστή ανάπτυξη, να εφαρμόσετε φίλτρα σε ερωτήματα DNS και πολλά άλλα. Τα ακόλουθα στοιχεία περιέχουν πιο λεπτομερείς πληροφορίες σχετικά με αυτές τις δυνατότητες.

- Εξισορρόπηση φόρτου εφαρμογής. Όταν αναπτύσσετε πολλές παρουσίες εφαρμογών σε διαφορετικές τοποθεσίες, μπορείτε να χρησιμοποιήσετε την πολιτική DNS για να φορτώσετε την κυκλοφορία ισορροπίας σε διαφορετικές παρουσίες εφαρμογών, εκχωρώντας δυναμικά φόρτο κυκλοφορίας στην εφαρμογή.

- Διαχείριση κυκλοφορίας με βάση τη Γεω-Τοποθεσία. Με την πολιτική DNS, μπορείτε να επιτρέψετε στους κύριους και δευτερεύοντες διακομιστές DNS να ανταποκρίνονται σε αιτήματα πελάτη DNS με βάση τη γεωγραφική θέση του πελάτη και τον πόρο στον οποίο προσπαθεί να συνδεθεί, παρέχοντας στον πελάτη τη διεύθυνση IP του πλησιέστερου πόρου.

- Διαχωρισμός του διακομιστή DNS του εγκεφάλου. Οι διακομιστές DNS διαχωρισμένου εγκεφάλου χωρίζονται σε διαφορετικές περιοχές ζώνης στον ίδιο διακομιστή DNS και οι πελάτες DNS λαμβάνουν μια απάντηση με βάση το εάν οι πελάτες είναι εσωτερικοί ή εξωτερικοί πελάτες. Μπορείτε να διαμορφώσετε διαχωρισμένο DNS για ενσωματωμένες ζώνες Active Directory ή για ζώνες σε αυτόνομους διακομιστές DNS.

- Φιλτράρισμα Μπορείτε να διαμορφώσετε την πολιτική DNS για τη δημιουργία φίλτρων ερωτημάτων με βάση τις συνθήκες που καθορίζετε. Τα φίλτρα ερωτημάτων στην πολιτική DNS σας επιτρέπουν να διαμορφώσετε έναν διακομιστή DNS ώστε να στέλνει μια προσαρμοσμένη μέθοδο με βάση το ερώτημα DNS και τον πελάτη DNS που στέλνει το ερώτημα DNS.

- Forensics Μπορείτε να χρησιμοποιήσετε την πολιτική DNS για να ανακατευθύνετε κακόβουλα προγράμματα-πελάτες DNS σε μια ανύπαρκτη διεύθυνση IP αντί να τους κατευθύνετε στον υπολογιστή στον οποίο προσπαθούν να προσεγγίσουν.

- Ανακατεύθυνση με βάση την ώρα της ημέρας. Η πολιτική DNS μπορεί να χρησιμοποιηθεί για τη διανομή της κυκλοφορίας εφαρμογών σε διαφορετικές γεωγραφικά διασκορπισμένες παρουσίες εφαρμογών χρησιμοποιώντας πολιτικές DNS που βασίζονται στην ώρα της ημέρας.

Οι πολιτικές DNS μπορούν επίσης να χρησιμοποιηθούν για ενσωματωμένες ζώνες DNS της Active Directory..

Τα γενικού τύπου domain είναι:

- Com (commercial – εμπορικά)
- Edu (educational – ακαδημαϊκά)
- Org (organizational – οργανισμοί μη κερδοσκοπικού χαρακτήρα)
- Net (network providers – πάροχοι δικτύου)
- Mil (military – κυρίες Αμερικάνικες στρατιωτικές υπηρεσίες)
- Gov (government – Αμερικάνικες κυβερνητικές υπηρεσίες)
- Int (international – διεθνείς οργανισμοί)

Εκπαιδευτική Υποενότητα : 2.5 Θέματα Ασφάλειας στην πληροφορική

Ασφάλεια Πληροφοριακού Συστήματος

Η έννοια της ασφάλειας των πληροφοριακών συστημάτων είναι γνωστή στους περισσότερους χρήστες. Η συνάφεια της κατασκευής ενός συστήματος ασφάλειας πληροφοριών επιβεβαιώνεται από τη συνεχή αύξηση των επιθέσεων χάκερ, των επιθέσεων σε τραπεζικά, εταιρικά δίκτυα και υπολογιστές ιδιωτών χρηστών. Ακόμη και οι συσκευές IoT εγκυμονούν κινδύνους σε περίπτωση μη εξουσιοδοτημένης πρόσβασης. Όλα αυτά δίνουν ιδιαίτερη προσοχή στην κατασκευή συστημάτων ασφαλείας.

Τύποι απειλών για την ασφάλεια στον κυβερνοχώρο

Η ασφάλεια των πληροφοριών σε οποιοδήποτε επίπεδο περιλαμβάνει την εφαρμογή τριών συνιστωσών:

- το απόρρητο των πληροφοριών ή τη μη προσβασιμότητά τους σε τρίτους·
- ακεραιότητα ή απουσία παραμόρφωσης ή υποκατάστασης·
- διαθεσιμότητα ή μόνιμη ευκαιρία για τον χρήστη να έχει πρόσβαση στα δεδομένα που χρειάζεται.

Οι χάκερ μπορεί να ενδιαφέρονται να καταστρέψουν οποιοδήποτε από αυτά τα στοιχεία, ανάλογα με τους στόχους που θα είναι διαφορετικοί για τον διοργανωτή της επιχειρηματικής κατασκοπείας, έναν εισβολέα που κλέβει αριθμούς τραπεζικών καρτών και ένα ξένο κράτος που έχει θέσει ως στόχο την αποδιοργάνωση του συστήματος ελέγχου του εχθρού χρησιμοποιώντας τεχνολογία πληροφοριών.

Ανάλογα με το επίπεδο των δικτύων, κατατάσσεται και το μοντέλο απειλών.

Η έννοια της ασφάλειας πληροφοριών περιγράφει τους κύριους τύπους απειλών που αντιμετωπίζει το κράτος, και συνεπώς η κοινωνία και οι επιχειρήσεις

Το έγγραφο προσφέρει ένα ολιστικό σύστημα προστασίας από κινδύνους υψηλού επιπέδου. Σε ιδιωτικό επίπεδο, προσδιορίζονται τα είδη των απειλών. Σε διάφορες μελέτες, τα συστημικά προβλήματα ασφάλειας συστημάτων πληροφοριών ονομάζονται:

- χειραγώγηση της πρόσβασης στον εσωτερικό χώρο πληροφοριών·
- κλοπή πληροφοριών από εταιρικά δίκτυα και βάσεις δεδομένων·
- αλλαγή πληροφοριών, πλαστογραφία εγγράφων σε ηλεκτρονική μορφή·
- βιομηχανική κατασκοπεία;
- κλοπή κεφαλαίων από τραπεζικούς λογαριασμούς·
- απειλές από ιούς·

Τα τρωτά σημεία λογισμικού, η παρουσία μη δηλωμένων χαρακτηριστικών επιτρέπουν τη χρήση διαφόρων μεθόδων παραβίασης βάσεων δεδομένων. Σε εθνικό επίπεδο, προτιμάται όλο και περισσότερο το ρωσικό λογισμικό που αναπτύσσεται λαμβάνοντας υπόψη τις τρέχουσες απειλές και δεν περιέχει νομικά έγγραφα.

Από τι να προστατευτείτε όσον αφορά τις εξωτερικές απειλές

Κατά την ανάπτυξη της αρχιτεκτονικής ενός συστήματος ασφάλειας πληροφοριών, είναι απαραίτητο να παρέχονται μέθοδοι προστασίας που ελαχιστοποιούν τους κινδύνους των πιο διαδεδομένων προβλημάτων - ιών και ανεπιθύμητων μηνυμάτων.

Αυτό το πρόβλημα επιλύεται με την εγκατάσταση:

- τείχη προστασίας (firewalls);
- Antivirus?
- λογισμικό φιλτραρίσματος email.

Αλλά εάν ένας χρήστης ανοίξει κατά λάθος ένα συνημμένο που περιέχει έναν ιό και έχει έρθει στο ταχυδρομείο από έναν γνωστό ανταποκριτή ή από κάποιον του οποίου η διεύθυνση φαίνεται οικεία, ο ιός ή ο δούρειος ίππος μπορεί να υπονομεύσει την ασφάλεια ολόκληρης της IP. Ως εκ τούτου, τα κύρια καθήκοντα θα είναι η εκπαίδευση και εκπαίδευση των χρηστών, η εγκατάσταση εργαλείων λογισμικού που παρέχουν προστασία από ακούσια σφάλματα.

Στο μέλλον, ο ορισμός αυτών των δύο ομάδων θα καταστήσει δυνατή τη συσχέτιση ήδη σε επίπεδο προγράμματος, χτίζοντας τον βαθμό διαφοροποίησης της πρόσβασης, το μοντέλο περιορισμού των δικαιωμάτων για εργασία με δεδομένα. Ένα τέτοιο σύστημα οικοδομείται πρώτα σε επίπεδο έγκρισης τοπικής κανονιστικής πράξης και μετά σε επίπεδο αποφάσεων προγράμματος. Αυτό όμως δεν είναι αρκετό. Η εταιρεία πρέπει να αποδέχεται και να αναπτύξει έγγραφα όπως:

1. Πολιτική ασφαλείας για εργασία με πληροφορίες.
2. Κανόνες εργασίας με το Διαδίκτυο και το εξωτερικό e-mail.
3. Κανόνες χειρισμού φορέων πληροφοριών, διαδικασία παρακολούθησης της χρήσης τους.
4. Κανόνες χειρισμού χάρτινων εγγράφων, ασφάλειά τους.

Σε ορισμένες περιπτώσεις, καθίσταται απαραίτητος ο έλεγχος της χρήσης συστημάτων αντιγραφής, εκτυπωτών και φωτοαντιγραφικών μηχανημάτων. Σε κάθε μία από αυτές τις περιπτώσεις, οι σύγχρονες λύσεις λογισμικού μπορούν να ελαχιστοποιήσουν τον κίνδυνο διαρροής πληροφοριών σε χαρτί. Για τους χειριστές προσωπικών δεδομένων, είναι σημαντικό να προετοιμάσουν μια πολιτική επεξεργασίας PD και να τη διαθέσουν στο κοινό.

Αλλά οι καθαρά τεχνικές λύσεις δεν αρκούν, είναι απαραίτητο να εισαχθεί ένας λειτουργικός μηχανισμός για την προσαγωγή του παραβάτη στη δικαιοσύνη. Για αυτό χρειάζεστε:

έκδοση εντολής στην εταιρεία για την εισαγωγή καθεστώτος προστασίας εμπορικού μυστικού, εξοικείωση των εργαζομένων με αυτό.

να αναπτύξει έναν κατάλογο πόρων πληροφοριών που σχετίζονται με εμπορικά απόρρητα, συμπεριλαμβανομένων προσωπικών δεδομένων σε αυτόν.

να συμπεριλάβει στις συμβάσεις εργασίας με το προσωπικό μια προϋπόθεση ευθύνης για την ασφάλεια των εμπορικών μυστικών.

Κατά την εφαρμογή του νομικού μηχανισμού για τη διασφάλιση της ασφάλειας των πληροφοριακών συστημάτων, κάθε περίπτωση εσκεμμένης ή ακούσιας διαρροής δεδομένων θα αποτελεί τη βάση για εσωτερική έρευνα ή εμπλοκή των υπηρεσιών επιβολής του νόμου. Ο δράστης όχι μόνο θα φέρει πειθαρχικές ευθύνες, έως και την απόλυση, αλλά θα υποχρεωθεί και σε αποζημίωση για τη ζημία που προκλήθηκε, ενώ σε κρίσιμες περιπτώσεις θα φέρει και ποινική ευθύνη. Η πεποίθηση του προσωπικού ότι οποιαδήποτε απόπειρα κλοπής πληροφοριών θα τιμωρηθεί από οποιονδήποτε υπάλληλο μειώνει τον βαθμό κινδύνου στις περισσότερες περιπτώσεις. Και εάν οι εργαζόμενοι γνωρίζουν ότι όλες οι ενέργειές τους εντός της περιμέτρου πληροφοριών της εταιρείας παρακολουθούνται, το επίπεδο ασφάλειας του πληροφοριακού συστήματος θα αυξηθεί περαιτέρω.

Απειλή είναι κάθε οντότητα ή γεγονός που μπορεί να προκαλέσει:

- Μη διαθεσιμότητα του συστήματος και των υπηρεσιών τυχαία ή με πρόθεση
- Τη μη εξουσιοδοτημένη τροποποίηση των δεδομένων
- Την καταστροφή των δεδομένων ή του συστήματος
- Την μη εξουσιοδοτημένη αποκάλυψη ευαίσθητων πληροφοριών



Κατ' αυτή την έννοια απειλή μπορεί να είναι ένας χρήστης του πληροφοριακού συστήματος που σκόπιμα ή από λάθος παραποιεί ή καταστρέφει δεδομένα, όπως επίσης ένας εξωτερικός εισβολέας ο οποίος αποκτά παράνομα πρόσβαση στο πληροφοριακό σύστημα.

Επίσης απειλή είναι κάθε πρόγραμμα ή διεργασία (γενικά λογισμικό) που έχει δημιουργηθεί για να:

- Προκαλέσει βλάβη ή δυσλειτουργία στο πληροφοριακό σύστημα και στα δεδομένα του (πχ ιοί, σκουλήκια, κλπ).
- Να υποκλέψει ευαίσθητα δεδομένα χρηστών όπως κωδικούς πρόσβασης στοιχεία πιστωτικών καρτών κλπ (πχ key loggers).
- Να πάρει υπό τον έλεγχό του το υπολογιστικό σύστημα και να ενεργοποιήσει διαδικασίες χωρίς την συναίνεση του χρήστη.

Ένα μεγάλο ποσοστό των ιών δεν έχει σκοπό την καταστροφή των δεδομένων του χρήστη ή την παρενόχλησή του, αλλά την κλοπή



Τι είναι ιοί και τι προκαλούν

Ένας ιός υπολογιστή είναι ένα σύνολο οδηγίων υπολογιστή ή κώδικα προγράμματος που εισάγεται σε ένα πρόγραμμα υπολογιστή από τον μεταγλωττιστή για να καταστρέψει τις λειτουργίες του υπολογιστή ή να καταστρέψει δεδομένα, να επηρεάσει τη χρήση του υπολογιστή και να μπορεί να αναπαραχθεί. Λόγω των ομοιοτήτων τους με βιολογικούς ιούς, οι άνθρωποι αναφέρονται σε αυτά τα κακόβουλα προγράμματα υπολογιστών ως «ιούς υπολογιστών» και μερικές φορές απλώς ως «ιούς». Οι ιοί υπολογιστών έχουν γενικά τα χαρακτηριστικά της μολυσματικότητας, της παρασιτικής, της απόκρυψης και της ενεργοποίησης (λανθάνουσα) και εξαπλώνονται γρήγορα και έχουν διάφορες μεθόδους επίθεσης. Αφού ο ιός εισέλθει στον κεντρικό υπολογιστή του χρήστη, θα προκαλέσει διαφορετικούς βαθμούς βλάβης στον κεντρικό υπολογιστή του χρήστη, που κυμαίνονται από την κατάληψη της μνήμης του συστήματος έως την επιβράδυνση της ταχύτητας λειτουργίας του συστήματος και, σε σοβαρές περιπτώσεις, θα οδηγήσει σε απώλεια σημαντικών δεδομένων, διαρροή πληροφοριών, συντριβή συστήματος και άλλες συνέπειες, με αποτέλεσμα ανεκτίμητες απώλειες.

Χαρακτηριστικά των ιών υπολογιστών

Ο ιός υπολογιστή είναι ένα είδος εικονικού ιού, ο οποίος έχει χαρακτηριστικά παρόμοια με τους βιολογικούς ιούς και μπορεί να ολοκληρώσει την αυτο-αντιγραφή και να εξαπλωθεί στον υπολογιστή. Έχει τα ακόλουθα χαρακτηριστικά.

Μολυσματικότητα: Ορισμένοι ιοί υπολογιστών μπορούν να αναπαράγουν και να αναπαράγουν τους δικούς τους κώδικες προγράμματος και στη συνέχεια να μολύνουν άλλους υπολογιστές μέσω δικτύων, συστημάτων ασύρματης επικοινωνίας και φορητών συσκευών αποθήκευσης, όπως σκληροί δίσκοι και δίσκοι U, και άλλοι υπολογιστές θα γίνουν νέες πηγές μόλυνσης, και σύντομο χρονικό διάστημα. ευρεία διάδοση εντός.

Καταστροφικότητα: Όταν ένας ιός εισβάλλει σε έναν υπολογιστή, συχνά καταστρέφει τους πόρους του υπολογιστή. Σε ελαφριές περιπτώσεις, μπορεί να μειώσει τον χώρο του υπολογιστή και να μειώσει την ταχύτητα του υπολογιστή. Σε σοβαρές περιπτώσεις, θα προκαλέσει καταστροφικές συνέπειες, όπως απώλεια δεδομένων αρχεία και σφάλματα συστήματος.

Παρασιτισμός: Το πρόγραμμα του ιού είναι ενσωματωμένο στο πρόγραμμα υποδοχής και εξαρτάται από το πρόγραμμα υποδοχής για να επιβιώσει. Όταν εκτελείται το πρόγραμμα υποδοχής, ο ιός θα διεγείρεται, διαφορετικά ο ιός θα είναι πάντα παρασιτικός στο πρόγραμμα υποδοχής και δεν θα εμφανίζεται.

Απόκρυψη: Οι ιοί συνήθως υπάρχουν ως κωδικοί προγραμμάτων σε άλλα προγράμματα ή με τη μορφή κρυφών αρχείων. Συνήθως είναι πολύ κρυφοί και είναι δύσκολο να εντοπιστούν ακόμη και από λογισμικό προστασίας από ιούς.

Ενεργοποίηση (λανθάνουσα): Ορισμένοι ιοί τείνουν να κρύβονται στο σύστημα για μεγάλο χρονικό διάστημα πριν επιτεθούν, με μια συγκεκριμένη περίοδο λανθάνουσας κατάστασης, και θα επιτεθούν όταν συναντήσετε μια συνθήκη ενεργοποίησης.

Ποια είναι τα σημάδια ότι μπορείτε να έχετε ιό υπολογιστή

Εάν εμφανιστούν τα ακόλουθα σημάδια στον υπολογιστή σας, είναι πιθανό να υποδηλώνουν ότι ο υπολογιστής έχει μολυνθεί από ιό υπολογιστή και πρέπει να δοθεί προσοχή και να αντιμετωπιστεί έγκαιρα.

- Το λειτουργικό σύστημα δεν εκκινεί σωστά ή εκτελείται αργά.
- Οι υπολογιστές συχνά παγώνουν ή κάνουν επανεκκίνηση ξαφνικά.
- Ο χώρος στο δίσκο πέφτει κατακόρυφα χωρίς προφανή λόγο.
- Παρουσιάζεται ένα παράνομο σφάλμα σε ένα πρόγραμμα που εκτελείται κανονικά και δεν μπορεί να εκτελεστεί κανονικά ή διακόπτεται.
- Ένα αρχείο λείπει, ένα αρχείο είναι κατεστραμμένο ή εμφανίζεται ένα νέο αρχείο.
- Η εξωτερική συσκευή είναι εκτός ελέγχου, όπως το ίδιο το ποντίκι κινείται ή ο εκτυπωτής δεν είναι φυσιολογικός.
- Στην οθόνη εμφανίζεται ανεξήγητο κείμενο ή εικόνες ή μουσική.
- Η ημερομηνία, η ώρα και τα χαρακτηριστικά του αρχείου έχουν αλλάξει.

Πώς να αμυνθείτε από ιούς υπολογιστών

Το κλειδί για την άμυνα από ιούς υπολογιστών έγκειται στην αμυντική απόδοση του συστήματος υπολογιστών και στην επίγνωση της προσωπικής ασφάλειας.

Συνιστάται η λήψη των παρακάτω μέτρων για την άμυνα από ιούς υπολογιστών.

- Εγκαταστήστε γνήσια λειτουργικά συστήματα και λογισμικό, χρησιμοποιήστε λογισμικό προστασίας από ιούς και ενημερώστε εγκαίρως τις ενημερώσεις κώδικα.
- Ενισχύστε την ευαισθητοποίηση σχετικά με την ασφάλεια, μην ανοίγετε μηνύματα ηλεκτρονικού ταχυδρομείου από άγνωστες πηγές και μην κάνετε τυχαία κλικ σε αβέβαιους συνδέσμους κατά την περιήγηση στον Ιστό.
- Η λειτουργία προστασίας από ιούς αναπτύσσεται στο τείχος προστασίας. Το πρόγραμμα προστασίας από ιούς μπορεί να ανιχνεύσει αρχεία ιών συγκρίνοντας τη δική του βάση δεδομένων υπογραφών ιών και στη συνέχεια να παρέμβει ή να υπενθυμίσει τα αρχεία ιών που εντοπίστηκαν μέσω αποκλεισμού και συναγερμού.

Ο πρώτος ιός που εμφανίστηκε στους προσωπικούς υπολογιστές ήταν ο ιός

Brain (γνωστός και ως Ashar, (C)Brain, Clone, Nipper, Pakistani, Pakistani, Lahore,

Pakistani flu, Pakistani Brain). Δημιουργήθηκε στο Πακιστάν το 1986 από τους

αδελφούς Basit και Amjad Farooq Alvi. Προσέβαλε τον τομέα εκκίνησης (boot sector) του σκληρού δίσκου.

Σκουλήκι (Worm): Το Worm είναι ένας αυτοαναπαραγόμενος ιός που εξαπλώνεται κυρίως αναζητώντας τρωτά σημεία του συστήματος (όπως ευπάθειες συστήματος των Windows, ευπάθειες διακομιστή δικτύου κ.λπ.). Διαφορετικά από τους γενικούς ιούς, τα σκουλήκια δεν απαιτούν χειροκίνητη παρέμβαση, μπορούν να εκμεταλλευτούν τα κενά για να επιτεθούν ενεργά και να έχουν ισχυρή ανεξαρτησία. Ένας υπολογιστής που έχει μολυνθεί με σκουλήκι θα αντιμετωπίσει αργή λειτουργία του συστήματος, χαμένα αρχεία, κατεστραμμένα αρχεία ή νέα αρχεία. Όταν ο ιός χτυπήσει, ένα πράγμα που μοιάζει με σκουλήκι θα εμφανιστεί στην οθόνη, το οποίο σέρνεται στην οθόνη. Τα σκουλήκια εξαπλώνονται πολύ πιο γρήγορα από τους παραδοσιακούς ιούς, επειδή τα σκουλήκια μπορούν να εξαπλωθούν μέσω διαφόρων καναλιών, όπως ευάλωτων διακομιστών, αρχείων δικτύου και ηλεκτρονικών μηνυμάτων, και οι επιθέσεις δεν περιορίζονται από προγράμματα κεντρικού υπολογιστή. Για παράδειγμα, ο περίφημος ιός «Panda Burning Incense». Ο ιός ανακαλύφθηκε το 2006 και εξαπλώθηκε γρήγορα μέσω της περιήγησης στον ιστό, των μονάδων flash USB και της κοινής χρήσης δικτύου. Χρειάστηκαν μόνο δύο μήνες για να εξαπλωθεί σε ολόκληρη τη χώρα. Όλα τα εικονίδια αρχείων στον μολυσμένο από ιό υπολογιστή θα μετατραπούν σε ένα panda που το κρατούσε. Τρία μπουκάλια λιβάνι, και μετά θα υπάρξουν καταστάσεις όπως αργή λειτουργία, συντριβή, μπλε οθόνη κ.λπ.

Δούρειος ίππος (Trojan horse):

Ορισμένα ονομάζονται επίσης λογισμικό απομακρυσμένου ελέγχου . Εάν ο Trojan μπορεί να συνδεθεί, τότε μπορεί να ειπωθεί ότι ο ελεγκτής έχει αποκτήσει όλα τα δικαιώματα λειτουργίας και ελέγχου του απομακρυσμένου υπολογιστή. Βασικά δεν υπάρχει μεγάλη διαφορά μεταξύ του χειρισμού του απομακρυσμένου υπολογιστή και του χειρισμού του Ο δικός του υπολογιστής. Μπορεί να καταγράψει την κάμερα του κατηγορούμενου χρήστη και να υποκλέψει τον κωδικό πρόσβασης , κ.λπ., και να εκτελέσει σχεδόν όλες τις λειτουργίες που μπορεί να εκτελέσει ο χρήστης (σύνδεση και αποσύνδεση υλικού , το σύστημα δεν μπορεί να ξεκινήσει ή δεν μπορεί να ελεγχθεί όταν δεν είναι συνδεδεμένο στο Διαδίκτυο). Η " Σύνδεση απομακρυσμένης

επιφάνειας εργασίας " που συνοδεύει τις εκδόσεις των Windows NT και νεότερες εκδόσεις ή κάποιο άλλο κανονικό λογισμικό τηλεχειρισμού, μπορεί επίσης να διαδραματίσει παρόμοιο ρόλο εάν δεν έχει ρυθμιστεί σωστά για ασφάλεια ή έχει παραβιαστεί από κακούς χρήστες. Αλλά συνήθως δεν ονομάζονται ιοί ή δούρειοι ίπποι, και η βάση για την κρίση εξαρτάται κυρίως από το για ποιο σκοπό έχει σχεδιαστεί το λογισμικό και εάν ο ιδιοκτήτης του υπολογιστή είναι ρητά ενημερωμένος.

Μακροϊός

Το μολυσμένο αντικείμενο του ιού μακροεντολών είναι το λογισμικό της σειράς γραφείου που αναπτύχθηκε από τη Microsoft και υπάρχουν επίσης παραλλαγές που απευθύνονται ειδικά σε άλλο λογισμικό. Το λογισμικό του Office, όπως το Microsoft Word και το ίδιο το Excel , υποστηρίζει εκτελούμενες εντολές που μπορούν να εκτελέσουν ορισμένες λειτουργίες αρχείων, επομένως χρησιμοποιούνται επίσης από κακόβουλους ιούς μακροεντολών στα έγγραφα του Office. Το openoffice.org υποστηρίζει μόνο επεξεργασία και δεν εκτελεί τις μακροεντολές VBS της Microsoft, επομένως τα έγγραφα του MS Office που περιέχουν ιούς μακροεντολών δεν μπορούν να εκτελεστούν αφού ανοίξουν στο openoffice.org.

Εγκαταστήστε και ενημερώστε το λογισμικό προστασίας από ιούς και τα προϊόντα τείχους προστασίας εγκαίρως

Διατηρήστε την πιο πρόσφατη βάση δεδομένων ιών, ώστε να μπορεί να εντοπιστεί ο πιο πρόσφατος ιός. Για παράδειγμα, ορισμένοι διακομιστές αναβάθμισης λογισμικού προστασίας από ιούς διαθέτουν νέα πακέτα βάσης δεδομένων ιών για ενημέρωση των χρηστών κάθε ώρα. Κατά τη χρήση τείχους προστασίας, θα πρέπει να σημειωθεί ότι απαγορεύεται η πρόσβαση στο δίκτυο λογισμικού από άγνωστες πηγές. Για λόγους όπως το anti-kill και η διεργασία έγχυσης, ορισμένοι ιοί μπορούν εύκολα να περάσουν από τη διπλή άμυνα του anti-virus και του firewall. είδη και υπηρεσίες εκκίνησης. Μερικά ειδικά τείχη προστασίας μπορούν να «υπερασπιστούν ενεργά» και να παρακολουθούν το μητρώο σε πραγματικό χρόνο. Κάθε κακόβουλη λειτουργία κακόβουλων προγραμμάτων στον υπολογιστή μπορεί να υποκλαπεί και να αποκλειστεί.

Μην κάνετε κλικ σε άγνωστες συνδέσεις και εκτελείτε άγνωστα προγράμματα

Άγνωστες συνδέσεις είναι πιθανό να αποστέλλονται αυτόματα από worms μέσω email ή λογισμικού ανταλλαγής άμεσων μηνυμάτων , όπως το QQ tail , ένας από τους ιούς QQ . Στους ιστότοπους, οι χρήστες ενδέχεται να λάβουν περισσότερους ιούς, χωρίς λήψη απευθείας μετά την επίσκεψη σε αυτούς τους ιστότοπους.



Ποιοι δημιουργούν τους ιούς (τα πρώτα χρόνια);

Πολλοί χάκερ που παράγουν ιούς έχουν συλληφθεί και διωχθεί, και η σοβαρότητα της απόφασης ποικίλλει από χώρα σε χώρα. Για παράδειγμα , η παραλλαγή MSBlast.F που γράφτηκε από τον Siobani στη Ρουμανία χρειάστηκε 15 λεπτά για να μολύνει μόνο περίπου 1.000 υπολογιστές. Σύμφωνα με τη ρουμανική νομοθεσία , μπορεί να έχει την υψηλότερη θα καταδικαστεί σε 15 χρόνια φυλάκιση . Το 1998, ο ιός CIH που γράφτηκε από τον Ταϊβανέζο συγγραφέα ιών Chen Yinghao υποτίθεται ότι είχε παραλύσει 6 εκατομμύρια υπολογιστές σε όλο τον κόσμο, αλλά εξαιρέθηκε από τις νομικές κυρώσεις επειδή δεν διώχθηκε μετά τη σύλληψή του. Το 2001, κάποιος μήνυσε τον Chen Yinghao ως Θύμα CIH . Συνελήφθη εκ νέου σύμφωνα με τους νόμους της Ταϊβάν εκείνη την εποχή, καταδικάστηκε για ζημιά και αντιμετώπιζε ποινή φυλάκισης έως και τριών ετών.

Ο Zhang Yong , ο συγγραφέας του προγράμματος Trojan Horse " Securities Thief " στην ηπειρωτική Κίνα , καταδικάστηκε σε ισόβια κάθειρξη επειδή χρησιμοποίησε το

πρόγραμμα του δούρειου ίππου για να υποκλέψει κωδικούς πρόσβασης λογαριασμών μετόχων, να κλέψει μετοχές αξίας 11,419 εκατομμυρίων Γιουάν και να αποκομίσει παράνομα κέρδη 386.000 γιουάν . Η έκθεση ανάλυσης μιας εταιρείας ανέφερε ότι αυτή τη στιγμή υπάρχουν 2 εκατομμύρια προγραμματιστές στον κόσμο που έχουν την ικανότητα να γράφουν πιο ώριμους ιούς υπολογιστών.

Κακόβουλο Λογισμικό – Διαφημιστικό Λογισμικό

Ο όρος "κακόβουλη διαφήμιση" περιγράφει τη διαδικτυακή διαφήμιση που διανέμει κακόβουλο λογισμικό. Η διαδικτυακή διαφήμιση είναι ένα περίπλοκο οικοσύστημα που περιλαμβάνει ιστότοπους διαφημιζόμενων, ανταλλαγές διαφημίσεων, διακομιστές διαφημίσεων, δίκτυα επαναστόχευσης και δίκτυα παράδοσης περιεχομένου. Οι εισβολείς χρησιμοποιούν αυτήν την πολυπλοκότητα για να τοποθετήσουν κακόβουλο κώδικα εκεί όπου οι διαφημιστές και τα δίκτυα διαφημίσεων δεν μπορούν πάντα να τον βρουν. Οι χρήστες που αλληλεπιδρούν με κακόβουλες διαφημίσεις μπορούν να κατεβάσουν κακόβουλο λογισμικό στη συσκευή τους ή να μεταβούν σε κακόβουλους ιστότοπους.

Στην διεθνή βιβλιογραφία σαν κακόβουλο λογισμικό (malware) ορίζεται οποιοδήποτε πρόγραμμα σκοπό έχει να επιφέρει δυσλειτουργία σε ένα υπολογιστικό σύστημα ή δίκτυο. Σύμφωνα με αυτό τον ορισμό και τα προγράμματα της προηγούμενης παραγράφου ανήκουν στην κατηγορία του κακόβουλου λογισμικού. Όμως στην καθημερινή πρακτική όταν αναφερόμαστε σε πρόγραμμα malware εξειδικεύουμε στην κατηγορία του κακόβουλου λογισμικού που σκοπό έχει την υποκλοπή προσωπικών πληροφοριών του χρήστη και την πρόσβαση στους πόρους του υπολογιστικού συστήματος. Τα πιο γνωστά είδη προγραμμάτων malware είναι.

Λογισμικό Κατασκοπίας (Spyware):

Το λογισμικό κατασκοπείας, εγκατεστημένο κρυφά, επιτρέπει στους διαχειριστές να κατασκοπεύουν τους χρήστες ή να ελέγχουν προσωπικά έναν προσωπικό υπολογιστή. Ωστόσο, το όνομα έχει βαθύτερο νόημα από την απλή παρακολούθηση της δραστηριότητας των χρηστών. Το λογισμικό κατασκοπείας μπορεί να συλλέξει διάφορες προσωπικές πληροφορίες, όπως δεδομένα περιήγησης στο Διαδίκτυο ή

πληροφορίες σχετικά με τοποθεσίες που επισκέπτεστε. Αυτό το λογισμικό βοηθά επίσης στη διαχείριση του υπολογιστή, στον έλεγχο των εργασιών του χρήστη (για παράδειγμα: εγκατάσταση πρόσθετου λογισμικού, αλλαγή της αρχικής σελίδας ενός προγράμματος περιήγησης Ιστού, ανακατεύθυνση συνδέσμων προγράμματος περιήγησης Ιστού).

Οι κακόβουλοι κατάσκοποι μπορούν να κατευθύνουν τα ερωτήματα αναζήτησης του χρήστη σε παράνομους ιστότοπους που περιέχουν παγίδες Ιστού, ιούς. Επιπλέον, το παράνομο λογισμικό υποκλοπής spyware επιτρέπει στους διαχειριστές να αλλάζουν τις ρυθμίσεις του υπολογιστή, με αποτέλεσμα πιο αργές διαδικασίες και σφάλματα που σχετίζονται με τη σύνδεση στο Διαδίκτυο.



Η εμφάνιση μη εξουσιοδοτημένων προγραμμάτων οδήγησε στη δημιουργία λογισμικού anti-spyware, καθώς και στην έγκριση ορισμένων νομοσχεδίων που σχετίζονται με την εγγύηση της ασφάλειας των ηλεκτρονικών δεδομένων σε διάφορες χώρες. Το Spyware είναι ένας τύπος προγράμματος υπολογιστή που μπορεί να παρακολουθεί ορισμένους τρόπους με τους οποίους λειτουργεί ένα σύστημα. Οι υπολογιστές που αντιλαμβάνονται πληροφορίες παρακολουθούνται εύκολα λόγω της εισαγωγής spyware στα συστήματα υπολογιστικής ισχύος. Το λογισμικό κατασκοπείας βοηθά στην απομακρυσμένη παρακολούθηση των διαδικτυακών δραστηριοτήτων του χρήστη σε μυστική λειτουργία. Σύμφωνα με ευρέως διαδεδομένα δεδομένα, πάνω από τα 2/3 των προσωπικών υπολογιστών περιέχουν κρυμμένους κατασκόπους στα συστήματά τους. Ωστόσο, ακόμη και τώρα, πολλοί χρήστες δεν κατανοούν πλήρως τι είναι ένα λογισμικό υποκλοπής spyware, τις μεθόδους και τις αρχές λειτουργίας του, τις κύριες διαφορές μεταξύ

εξουσιοδοτημένων και κακόβουλων προγραμμάτων κ.λπ. Ορισμένοι συγγέουν αυτό το λογισμικό με ιούς.

Τι είναι όμως ακριβώς το spyware; Είναι ιός; Οι ιοί υπολογιστών είναι ένα κομμάτι κωδικών που έχουν σχεδιαστεί για να αναπαράγουν και να διαδίδουν τις δραστηριότητές τους από τον έναν υπολογιστή στον άλλο. Συχνά, μετακινούμενος μεταξύ υπολογιστών, ο ιός καταστρέφει τα προσωπικά αρχεία των χρηστών ή βλάπτει το λειτουργικό σύστημα. Το spyware δεν έχει σχεδιαστεί για να καταστρέφει δεδομένα χρήστη. Το κύριο χαρακτηριστικό τους είναι η κρυφή διείδυση στο σύστημα των χρηστών, η υποκλοπή εμπιστευτικών δεδομένων χρήστη και η εξίσου ανεπαίσθητη εξαφάνιση από το σύστημα υπολογιστή. Από μόνο του, το spyware είναι νόμιμο, αλλά μόνο οι προθέσεις μισθοφόρων διαχειριστών που, αφού υποκλέψουν δεδομένα, το χρησιμοποιούν όχι για ανάλυση μάρκετινγκ, αλλά για κέρδος μπορεί να είναι παράνομες .

Επί του παρόντος, το πιο χρήσιμο λογισμικό υποκλοπής spyware, συμπεριλαμβανομένου του δημοφιλούς spyware NeoSpy, είναι γραμμένο για Windows.

Πρόγραμμα καταγραφής πληκτρολογήσεων (Key loggers):

Το Keylogger είναι μια από τις πιο επικίνδυνες κακόβουλες εφαρμογές. Μέσω αυτού, οι χάκερ μπορούν να λάβουν οποιαδήποτε εμπιστευτική πληροφορία, συμπεριλαμβανομένων των στοιχείων πληρωμής του χρήστη. Το keylogger είναι οποιοδήποτε στοιχείο λογισμικού ή υλικού που μπορεί να υποκλέψει και να καταγράψει όλους τους χειρισμούς με ένα πληκτρολόγιο υπολογιστή. Συχνά, ένα keylogger βρίσκεται μεταξύ του πληκτρολογίου και του λειτουργικού συστήματος και παρεμποδίζει όλες τις ενέργειες του χρήστη. Αυτό το εργαλείο είτε αποθηκεύει τις πληροφορίες που έχουν υποκλαπεί στον μολυσμένο υπολογιστή ή, εάν είναι μέρος μιας μεγαλύτερης επίθεσης, όλα τα δεδομένα μεταφέρονται αμέσως στον απομακρυσμένο υπολογιστή των εισβολέων. Αν και ο όρος "keylogger" χρησιμοποιείται συνήθως για να αναφέρεται σε κακόβουλο λογισμικό, μερικές φορές χρησιμοποιείται και από τις υπηρεσίες επιβολής του νόμου.

Διαφημιστικό Λογισμικό (Adware): (από τα αγγλικά. Ad vertisement - διαφήμιση και Software - λογισμικό) - ανεπιθύμητο λογισμικό που περιέχει διαφημίσεις. Το Adware συνοδεύεται από προϊόντα λογισμικού, συνήθως δωρεάν ή κοινόχρηστο λογισμικό. Στο μέλλον, κατά τη χρήση του προϊόντος λογισμικού, η διαφήμιση εμφανίζεται αναγκαστικά στον χρήστη. Ορισμένα στοιχεία adware είναι συνήθως κρυμμένα, γεγονός που καθιστά δύσκολη την αφαίρεσή τους. Επίσης, τα διαφημιστικά συστήματα μπορούν να συλλέγουν εμπιστευτικές πληροφορίες σχετικά με τον υπολογιστή και τον χρήστη:

- *διεύθυνση IP υπολογιστή.*
- *έκδοση του εγκατεστημένου λειτουργικού συστήματος και του προγράμματος περιήγησης στο Διαδίκτυο.*
- *μια λίστα με τους πόρους του Διαδικτύου που επισκέπτεται συχνά ο χρήστης.*
- *ερωτήματα αναζήτησης.*
- *άλλα δεδομένα που μπορούν να χρησιμοποιηθούν σε επόμενες διαφημιστικές καμπάνιες.*

Τις περισσότερες φορές, τα στοιχεία adware ενσωματώνονται σε δωρεάν εφαρμογές. Το MyWay Searchbar, το οποίο διαθέτει βοηθητικά προγράμματα SmileyCentral και Zwinky, και το WeatherBug (μια λειτουργική μονάδα διαφημίσεων που μπορεί να απενεργοποιηθεί μόνο αφού πληρώσετε για μια αναβάθμιση) είναι παραδείγματα εφαρμογών που αναφέρθηκαν ότι περιέχουν adware. Επίσης, προγράμματα Adware και Spyware (spyware) μπορούν να εγκατασταθούν μέσα από «τρύπες» στην ασφάλεια του προγράμματος περιήγησης ή του λειτουργικού συστήματος.

Επιθέσεις Εισβολής

Κάθε μη εξουσιοδοτημένη πρόσβαση σε έναν υπολογιστή ή δίκτυο υπολογιστών είναι Επίθεση Εισβολής. Μερικές μέθοδοι για την πραγματοποίηση επίθεσης εισβολής είναι:

Ασύμμετρη Δρομολόγηση (Asymmetric Routing)

Ο επιτιθέμενος προσπαθεί να στείλει πακέτα δεδομένων χρησιμοποιώντας περισσότερες από μία διαδρομές για τον στοχο του που είναι η συσκευή δικτύου.

Ο σκοπός είναι να αποφευχθεί η ανίχνευση της επίθεσης έτσι στέλνοντας από διαφορετικές διαδρομές ένα σημαντικό μέρος των παράνομων πακέτων

παρακάμπτοντας ορισμένα τμήματα του δικτύου και κατά συνέπεια τους αισθητήρες παραβίασης του δικτύου.

Επιθέσεις Υπερχείλισης (Buffer Overflow Attacks):

Επιθέσεις υπερχείλισης buffer είναι για να εκτελέσουν εκτελέσιμους κώδικες υπερχειλίζοντας σταθερές προσωρινές ενδιάμεσες μνήμη που προορίζονται για τη διαδικασία εισαγωγής. Αυτοί οι κωδικοί επιτρέπουν σε έναν εισβολέα να εισέλθει στο σύστημα του υπολογιστή-στόχου, ή να βλάψει τα δεδομένα που είναι αποθηκευμένα σε αυτόν.

Υπάρχουν δύο τύποι επιθέσεων υπερχείλισης buffer.

1. Δυναμικές επιθέσεις μνήμης : Γεμίζει η μνήμη που είναι δεσμευμένη για το πρόγραμμα. Αυτές είναι δύσκολες επιθέσεις και εκτελούνται σπάνια.
2. Επιθέσεις στοίβας: Τα αντικείμενα αποθήκευσης στοίβας χρησιμοποιούνται για την αποθήκευση των δεδομένων χρήστη. Αυτός είναι ο πιο κοινός τύπος επίθεσης.

Εξειδικευμένες Επιθέσεις στα Πρωτόκολλα (Protocol-Specific Attacks):

Οι επιθέσεις στα πρωτόκολλα μπορούν να χωριστούν σε δύο κατηγορίες:

1. Σε μια παθητική επίθεση , ο cracker δεν συμμετέχει στο πρωτόκολλο, παρακολουθεί μόνο το πρωτόκολλο και προσπαθεί να πάρει πολύτιμες πληροφορίες με βάση το κρυπτογραφημένο κείμενο.
2. Σε μια ενεργή επίθεση , ο εισβολέας προσπαθεί να αλλάξει το πρωτόκολλο προς όφελός του. Για το σκοπό αυτό, ένας ενεργός κροτίδα μπορεί να μιμηθεί ένα άλλο άτομο, να επαναλάβει μηνύματα, να αντικαταστήσει μηνύματα, να σπάσει τη γραμμή, να τροποποιήσει πληροφορίες.

Παραδείγματα ενεργών επιθέσεων πρωτοκόλλου είναι η επίθεση man-in-the-middle, η επίθεση λεξικού, η επίθεση επανάληψης, η επίθεση επανάληψης μηνυμάτων, η επίθεση συστήματος ρολογιού. Μια επίθεση man-in-the-middle μπορεί να πετύχει εάν τα μέρη δεν έχουν τρόπο να επαληθεύσουν ότι πράγματι επικοινωνούν μεταξύ τους. Η επίθεση man-in-the-middle μπορεί να αποτραπεί χρησιμοποιώντας ψηφιακές υπογραφές κατά τη διάρκεια του πρωτοκόλλου ανταλλαγής κλειδιών

συνεδρίας ή χρησιμοποιώντας την ταυτόχρονη μεταφορά κλειδιών και μηνυμάτων χωρίς προηγούμενη ανταλλαγή κλειδιών.

Μια επίθεση λεξικού σάς επιτρέπει να συγκρίνετε ένα κλεμμένο αρχείο κρυπτογραφημένου κωδικού πρόσβασης με ένα έτοιμο κρυπτογραφημένο αρχείο υποψηφίου κωδικού πρόσβασης, αναζητώντας ένα ταίριασμα. Οι επιθέσεις λεξικών μπορούν να γίνουν πιο δύσκολες χρησιμοποιώντας άγκυρες - τυχαίες συμβολοσειρές που συνδέονται με κωδικούς πρόσβασης πριν υποβληθούν σε επεξεργασία από μια μονόδρομη συνάρτηση.

Η άμυνα ενάντια στις επιθέσεις επανάληψης είναι ότι οι λειτουργίες κρυπτογράφησης και ψηφιακής υπογραφής πρέπει να είναι διαφορετικές. Για να το κάνετε αυτό, πρέπει να χρησιμοποιήσετε διαφορετικά κλειδιά για κάθε συναλλαγή ή διαφορετικούς αλγόριθμους σε κάθε συναλλαγή ή επισήμανση ή δημιουργία ψηφιακών υπογραφών χρησιμοποιώντας συναρτήσεις κατακερματισμού.

Τρόποι αντιμετώπισης επιθέσεων επανάληψης είναι η χρήση αριθμών ακολουθίας μηνυμάτων, σφραγίδων ημερομηνίας/ώρας, μοναδικών αιτημάτων (περιπτώσεις) και απαντήσεων που περιέχουν τη σωστή τιμή εμφάνισης.

Συμπεριλαμβάνοντας την τρέχουσα ώρα στα κρυπτογραφικά πρωτόκολλα, εμποδίζουμε έναν εισβολέα να προωθήσει παλιά μηνύματα υπό το πρόσχημα νέων.

Ωστόσο, μια επιτυχημένη επίθεση στο σύστημα του ρολογιού (γυρίζοντας το ρολόι προς τα πίσω ή προς τα εμπρός, σταματά το ρολόι) έχει ως αποτέλεσμα την αποστολή μηνυμάτων με λάθος ημερομηνία/ώρα, κάτι που μπορεί να έχει μεγάλες οικονομικές επιπτώσεις.

Κυκλοφοριακή Πλημμύρα (Traffic Flooding):

Traffic Flooding Δημιουργεί πολύ μεγάλο φόρτο από κυκλοφορούντα πακέτα τα οποία το σύστημα αδυνατεί να παρακολουθήσει επαρκώς είναι μια μεθοδος διείσδυση σε ένα δίκτυο η οποία απλά στοχεύει στα συστήματα ανίχνευσης.

Οι επιτιθέμενοι στοχεύουν σε άρνηση υπηρεσίας (DoS) και επίσης μπορούν να προκαλέσουν μία κατάσταση «fail- open».

Αδυναμίες Προγραμμάτων και Λειτουργικών Συστημάτων. Ένα λειτουργικό σύστημα είναι ένα σύνολο λογισμικού που ελέγχει τις βασικές λειτουργίες ενός υπολογιστή, συμπεριλαμβανομένων προγραμμάτων που εκτελούνται, διαχείρισης αποθήκευσης, εισόδου/εξόδου και πόρων επικοινωνίας. Σύμφωνα με το W3Schools, τα Windows XP και τα Windows 7 είναι τα πιο δημοφιλή λειτουργικά συστήματα, με την οικογένεια των Windows, η οποία περιλαμβάνει επίσης τα Windows Vista και τα Windows 2003, να αντιπροσωπεύουν περισσότερο από το 80 τοις εκατό της χρήσης του λειτουργικού συστήματος. Άλλα λειτουργικά συστήματα για προσωπικούς υπολογιστές περιλαμβάνουν το Linux, το λειτουργικό σύστημα Apple Macintosh, το OS X και πολλά άλλα, το καθένα με τα δικά του δυνατά και αδύνατα σημεία.

Αδυναμίες (vulnerabilities) είναι ελαττώματα στο λογισμικό των υπολογιστών που δημιουργούν αδυναμίες στη συνολική ασφάλεια του υπολογιστή ή του δικτύου. Τρωτά σημεία μπορούν επίσης να δημιουργηθούν από ακατάλληλη διαμόρφωση του υπολογιστή ή της ασφάλειας.

Οι απειλές εκμεταλλεύονται τις αδυναμίες των τρωτών σημείων με αποτέλεσμα πιθανή βλάβη στον υπολογιστή ή τα προσωπικά δεδομένα.

Οι αδυναμίες είναι συνδυασμός τριών στοιχείων:

(α) Ύπαρξη ευαισθησίας ή ελαττώματος στο σύστημα,

(β) Γνώση από τον εισβολέα της ύπαρξης του ελαττώματος και

(γ) Ικανότητα του εισβολέα να εκμεταλλευτεί το ελάττωμα.

Για να εκμεταλλευτεί την αδυναμία, ο εισβολέας πρέπει να έχει τουλάχιστον ένα κατάλληλο εργαλείο ή τεχνική που μπορεί να συνδεθεί σε μια αδυναμία του συστήματος.

Τρωτά σημεία σε ένα λειτουργικό σύστημα ή ένα πρόγραμμα μπορούν να προκύψουν από:

- Τα λάθη του προγράμματος (bugs). Ένα λάθος στον κώδικα του προγράμματος μπορεί να επιτρέψει σε έναν επιτιθέμενο να αποκτήσει πρόσβαση σε έναν υπολογιστή ή μια συσκευή δικτύου και να πάρει τον έλεγχο
- Διαδικασίες Επικοινωνίας. Στα προγράμματα υπάρχουν χαρακτηριστικά που έχουν προβλεφθεί από τους δημιουργούς τους επιτρέποντας πρόσβαση στο σύστημα από τις νόμιμες εφαρμογές. Αυτά τα χαρακτηριστικά είναι πλήρως τεκμηριωμένα δίνοντας τις απαραίτητες πληροφορίες στους επιτιθέμενους για να εντοπίσουν πιθανά τρωτά σημεία.

Αν τα τρωτά σημεία είναι γνωστό ότι υπάρχουν σε ένα λειτουργικό σύστημα ή μια εφαρμογή – ανεξάρτητα από το κατά πόσον οι αδυναμίες οφείλονται σε προβλεπόμενα χαρακτηριστικά ή όχι - το λογισμικό θα είναι ανοιχτό σε επιθέσεις από κακόβουλα προγράμματα.

Πολλά εργαλεία λογισμικού υπάρχουν που μπορεί να βοηθήσουν στην ανακάλυψη (και μερικές φορές στην απομάκρυνση) των τρωτών σημείων σε ένα σύστημα υπολογιστή.

Αν και τα εργαλεία αυτά μπορούν να παρέχουν έναν είδος ελεγκτή παρέχοντας μια καλή επισκόπηση των πιθανών τρωτών σημείων που υπάρχουν, δεν μπορούν να αντικαταστήσουν την ανθρώπινη κρίση. Στηριζόμενοι αποκλειστικά στους σαρωτές αδυναμιών είναι πιθανόν να υπάρξουν ψευδείς συναγερμοί ή και περιορισμένη αντίληψη των προβλημάτων που υπάρχουν στο σύστημα.

Αδυναμίες έχουν βρεθεί σε κάθε μεγάλη εφαρμογή και λειτουργικό σύστημα, συμπεριλαμβανομένων των Windows, Mac OS, διάφορες διανομές των Unix και Linux, OpenVMS, και άλλα.

Ο μόνος τρόπος για να μειωθεί η πιθανότητα να χρησιμοποιηθεί μια αδυναμία για επίθεση σε ένα σύστημα είναι μέσω της συνεχούς επαγρύπνησης, συμπεριλαμβανομένης της προσεκτικής συντήρησης του συστήματος όπως είναι η τακτική εγκατάσταση ενημερώσεων λογισμικού (software patches), βέλτιστες πρακτικές ανάπτυξης του δικτύου (π.χ. με τη χρήση firewalls για έλεγχο πρόσβασης) και έλεγχος του λογισμικού τόσο κατά τη διάρκεια της ανάπτυξη, αλλά και σε όλη τη διάρκεια του κύκλου ζωής του.

Πώς να ασφαλίσετε το email

Το email έχει σχεδιαστεί για να είναι όσο το δυνατόν πιο ανοιχτό και προσβάσιμο και να επιτρέπει στους ανθρώπους να επικοινωνούν μεταξύ τους. Το μειονέκτημα αυτής της διαθεσιμότητας είναι η ευπάθεια ορισμένων πτυχών του email. Αυτό επιτρέπει στους εισβολείς να χρησιμοποιούν email για να παραβιάζουν την ασφάλεια του Διαδικτύου.

Τι είναι η ασφάλεια email;

Η ασφάλεια email είναι ένα σύνολο μεθόδων που χρησιμοποιούνται για την προστασία των λογαριασμών email και των επικοινωνιών από μη εξουσιοδοτημένη πρόσβαση, απώλεια και συμβιβασμό. Δεδομένου ότι το ηλεκτρονικό ταχυδρομείο χρησιμοποιείται συχνά για τη διάδοση επιθέσεων κακόβουλου λογισμικού, ανεπιθύμητης αλληλογραφίας και *phishing*, η ασφάλεια του ηλεκτρονικού ταχυδρομείου είναι μια σημαντική πτυχή της διαδικτυακής ασφάλειας.

Πώς να αντιμετωπίσετε το spam email;

Τα ανεπιθύμητα μηνύματα είναι ανεπιθύμητα μηνύματα που αποστέλλονται μαζικά. Οι περισσότεροι πάροχοι email χρησιμοποιούν αλγόριθμους φιλτραρίσματος ανεπιθύμητων μηνυμάτων, αλλά παρόλα αυτά, τα ανεπιθύμητα μηνύματα μπορούν να συνεχίσουν να φτάνουν. Για να απαλλαγείτε από ανεπιθύμητα μηνύματα, μπορείτε να ακολουθήσετε τα ακόλουθα βήματα:

Επισήμανση ανεπιθύμητων μηνυμάτων. Αυτό θα βοηθήσει τον πάροχο email να βελτιώσει το φιλτράρισμα ανεπιθύμητων μηνυμάτων. Ο τρόπος επισήμανσης ενός μηνύματος ως ανεπιθύμητου εξαρτάται από το πρόγραμμα-πελάτη ηλεκτρονικού ταχυδρομείου που χρησιμοποιείτε: Outlook, Gmail, Apple Mail, Yahoo Mail, κ.λπ.

• Ποτέ μην κάνετε κλικ σε συνδέσμους ή μην ανοίγετε συνημμένα σε ανεπιθύμητα μηνύματα. Ως αποτέλεσμα τέτοιων ενεργειών, μπορεί να γίνει λήψη κακόβουλου λογισμικού στη συσκευή. Τουλάχιστον, τέτοιες ενέργειες χρησιμεύουν για να

επιβεβαιώσουν στους αποστολείς ανεπιθύμητης αλληλογραφίας ότι πρόκειται για ενεργό λογαριασμό email και να τους ενθαρρύνουν να στείλουν περισσότερα ανεπιθύμητα μηνύματα.

- Να είστε προσεκτικοί όταν χρησιμοποιείτε μια διεύθυνση email. Είναι χρήσιμο να έχετε έναν πρόσθετο προσωρινό λογαριασμό email που χρησιμοποιείται αποκλειστικά για σκοπούς εγγραφής και συνδρομής. Θα πρέπει να είναι διαφορετικό από το εργασιακό και από αυτό που χρησιμοποιείται για αλληλογραφία με φίλους και συγγενείς.

- Οι περισσότεροι πάροχοι email έχουν ρυθμίσεις απορρήτου . Βεβαιωθείτε ότι έχουν ρυθμιστεί σε ένα άνετο επίπεδο για εσάς.

- Εξερευνήστε φίλτρα ανεπιθύμητης αλληλογραφίας τρίτων. Παρέχουν ένα επιπλέον επίπεδο ασφάλειας στον κυβερνοχώρο, καθώς τα μηνύματα ηλεκτρονικού ταχυδρομείου πρέπει να περάσουν από δύο φίλτρα ανεπιθύμητης αλληλογραφίας πριν φτάσουν στον παραλήπτη: το φίλτρο ανεπιθύμητης αλληλογραφίας του παρόχου αλληλογραφίας και ένα φίλτρο τρίτου μέρους.

Τα πάρα πολλά ανεπιθύμητα μηνύματα ηλεκτρονικού ταχυδρομείου μπορεί να είναι σημάδι ότι η διεύθυνση email σας έχει παραβιαστεί λόγω παραβίασης δεδομένων. Σε αυτήν την περίπτωση, συνιστάται να αλλάξετε τη διεύθυνση email σας.

Μέθοδοι Προστασίας

Αν και η τεχνολογία εξελίσσεται με πολύ γρήγορους ρυθμούς θεωρικά αυτούς τους ρυθμούς θα έπρεπε να ακολουθούν και τα μετρά προστασίας. Δυστυχώς όμως δεν υπάρχει ένα πρόγραμμα ή κάποιο εργαλείο που να παρέχει καθολική προστασία σε ένα σύστημα υπολογιστή ή ένα δίκτυο. Με κάθε είδος κινδύνου υπάρχουν διαφορετικά εργαλεία και διαφορετικές μεθοδολογίες για την προστασία.

Διαχείριση Καταστροφών

Σε ένα πληροφοριακό σύστημα συμβούν ανά πάσα στιγμή μια ζημιά ,συνήθως μάλιστα χωρίς καμία προειδοποίηση.

Ζημιά – Καταστροφή ονομάζουμε ότι προκαλεί διακοπή, ή αδυναμία κάποιας λειτουργίας ζωτικής σημασίας, για απροσδιόριστο χρονικό διάστημα.

Οι καταστροφές διακρίνονται:

(α) Σε αυτές που γίνονται από τον άνθρωπο π.χ όπως από λανθασμένη χρήση. Για οποιαδήποτε περίπτωση καταστροφής στο πληροφοριακό σύστημα ή σε μέρος αυτού, είναι απαραίτητο ο οργανισμός να έχει εκπονήσει ένα σχέδιο Ανάκαμψης από Καταστροφή.

Το σχέδιο (Disaster Recovery) αναλύει το σύνολο των ενεργειών που θα πρέπει να γίνουν ώστε σε περίπτωση καθολικής καταστροφής, ή μερικής καταστροφής πληροφοριακού συστήματος να είναι δυνατόν αυτό να επανέλθει σε κατάσταση λειτουργίας το συντομότερο δυνατό.

Ένα σχέδιο ανάκαμψης από καταστροφή μπορεί να περιλαμβάνει:

Αντίγραφα ασφαλείας.



Ένα αντίγραφο ασφαλείας(Back Up), στην τεχνολογία των πληροφοριών (information technology), άλλα και η διαδικασία δημιουργίας αντιγράφων ασφαλείας, αναφέρεται στην φύλαξη των δεδομένων ενός πληροφοριακού

συστήματος, για να είναι δυνατή η επαναφορά των δεδομένων στην αρχική τους κατάσταση μετά από μια απώλειά τους.

Η πολιτική διατήρησης των δεδομένων καθορίζεται από το σχέδιο

ανάκαμψης ανάλογα με το πόσο χρόνο είναι απαραίτητο αυτά να διαφυλάσσονται.

Μερικά από τα ερωτήματα που πρέπει να απαντηθούν όταν αποφασίζεται η πολιτική λήψης αντιγράφων ασφαλείας είναι:

- Τι είδους αντίγραφα ασφαλείας λαμβάνονται (πλήρες back up, σταδιακό back up κλπ);
- Με ποια μέθοδο θα λαμβάνονται τα αντίγραφα ασφαλείας;
- Ποιος είναι υπεύθυνος για την λήψη των αντιγράφων;
- Κάθε πότε θα γίνεται λήψη αντιγράφων;
- Σε τι αποθηκευτικό μέσο θα αποθηκεύονται τα αντίγραφα;
- Σε ποιο χώρο θα φυλάγονται τα αντίγραφα;
- Πόσο χρόνο πρέπει να διατηρούνται τα αντίγραφα;

Όσον αφορά **τα είδη των αντιγράφων ασφαλείας** οι δυνατές επιλογές είναι

Πλήρες αντίγραφο (Full Back Up)

ένα πλήρες αντίγραφο ασφαλείας δεδομένων αποθηκεύει ένα ακριβές αντίγραφο ολόκληρου του αρχικού συνόλου δεδομένων στην καθορισμένη τοποθεσία κάθε φορά. Δεν έχει σημασία τι έχει συμβεί σε αυτές τις πληροφορίες από το προηγούμενο πλήρες αντίγραφο ασφαλείας. όλα θα αντιγραφούν μέχρι το τελευταίο byte. Αυτή η προσέγγιση έχει τόσο προφανή πλεονεκτήματα όσο και ορισμένα μειονεκτήματα. *Πλεονεκτήματα* : Ο χρήστης μπορεί να είναι σίγουρος ότι όλα τα απαραίτητα δεδομένα αποθηκεύονται με εγγύηση.

Μειονεκτήματα : Ένα πλήρες αντίγραφο ασφαλείας απαιτεί πολύ χρόνο για να ολοκληρωθεί και καταλαμβάνει επίσης πολύ χώρο αποθήκευσης, ειδικά εάν πολλά αντίγραφα ενός συνόλου δεδομένων αποθηκεύονται στο ίδιο μέσο.

Αυξητικό Αντίγραφο (Incremental Back Up)

Καθώς το μέγεθος των βάσεων δεδομένων, ειδικά η αποθήκευση, συνεχίζει να αυξάνεται σε κλίμακα terabyte και petabyte, ο χρόνος και οι πόροι υλικού που απαιτούνται για τη δημιουργία αντιγράφων ασφαλείας και την επαναφορά αυτών των βάσεων δεδομένων αυξάνονται ταυτόχρονα. Όταν ασχολούμαστε με μεγάλες βάσεις δεδομένων, η πλήρης δημιουργία αντιγράφων ασφαλείας βάσεων δεδομένων και επιτραπέζιων χώρων δεν είναι πάντα η καλύτερη προσέγγιση, επειδή οι απαιτήσεις αποθήκευσης για πολλαπλά αντίγραφα τέτοιων βάσεων δεδομένων είναι απαγορευτικές. Λάβετε υπόψη τις ακόλουθες σκέψεις:

- Όταν αλλάζει μόνο ένα μικρό ποσοστό των δεδομένων στην αποθήκη, θα πρέπει να είναι δυνατό να γίνει χωρίς δημιουργία αντιγράφων ασφαλείας ολόκληρης της βάσης δεδομένων.
- Η προσθήκη επιτραπέζιων χώρων σε υπάρχουσες βάσεις δεδομένων και στη συνέχεια η δημιουργία αντιγράφων ασφαλείας μόνο του τραπεζικού χώρου είναι επικίνδυνη, επειδή δεν υπάρχει καμία εγγύηση ότι τίποτα εκτός των επιτραπέζιων χώρων που αντιγράφονται δεν έχει αλλάξει μεταξύ δύο αντιγράφων ασφαλείας.

Πλεονάζων Εξοπλισμός Πλεονασμός (redundancy)

Ο σχεδιασμός ενός συστήματος στην πληροφορική έτσι ώστε ένα ή περισσότερα συστατικά του είναι διπλά έτσι ώστε σε περίπτωση αποτυχίας να υπάρχει ένα αντίγραφο του εξοπλισμού.

Ο πλεονάζων εξοπλισμός (redundant infrastructure) είναι επιπλέον εξοπλισμός ο οποίος κατά κανόνα δεν χρησιμοποιείται υπό κανονικές συνθήκες.

Μόνο σε περίπτωση βλάβης του κύριου εξοπλισμού τότε αυτόματα, είτε με ανθρώπινη παρέμβαση ενεργοποιείται. Φυσικά ο σκοπός είναι να μειωθεί στο ο χρόνο εκτός λειτουργίας (downtime) του πληροφοριακού συστήματος.

Μια απλή εφαρμογή πλεονάζοντος εξοπλισμού είναι η ύπαρξη δεύτερου σκληρού δίσκου σε ένα σύστημα υπολογιστή ο οποίος είναι το ακριβές αντίγραφο του βασικού δίσκου.

Η τεχνολογία που χρησιμοποιείται είναι η τεχνολογία RAID (Redundant Array of Independent Disks). Η τεχνολογία αυτή δημιουργεί συστοιχίες από δύο ή περισσότερους δίσκους που τα περιεχόμενά τους είναι πανομοιότυπα. Στην περίπτωση που ο ένας παρουσιάσει βλάβη συνεχίζει ο δεύτερος να λειτουργεί.

Ειδικά σε περιπτώσεις διακομιστών ιστοσελίδων (Web Servers) μπορεί να εγκατασταθεί πλήρες πληροφοριακό σύστημα αντίγραφο του βασικού σε διαφορετικό χώρο, ή και διαφορετική γεωγραφική περιοχή, για να υπάρξει συνέχεια λειτουργίας σε οτιδήποτε είδους βλάβη και να προκύψει, ακόμα και σε περίπτωση αδυναμίας του τηλεπικοινωνιακού παρόχου να προσφέρει σύνδεση.

Cloud computing.

Το cloud computing, το οποίο προσελκύει την προσοχή ως τεχνολογία και υπηρεσία υποδομής υπηρεσιών πληροφορικής επόμενης γενιάς, βρίσκεται υπό

έρευνα και ανάπτυξη προς την ευρεία χρήση του. Σύμφωνα με την οποία, οι υπολογιστικές εργασίες διαμοιράζονται σε έναν μεγάλο αριθμό υπολογιστών, έτσι ώστε όλες οι εφαρμογές να έχουν πρόσβαση στην υπολογιστική ισχύ, τον αποθηκευτικό χώρο και το λογισμικό.

Η βασική αρχή πίσω από αυτό το μοντέλο είναι η παροχή της υπολογιστικής ισχύος, της αποθήκευσης και του λογισμικού «ως υπηρεσίες».

Τι είναι το cloud computing;

Το Cloud computing είναι η κατ' απαίτηση παράδοση πόρων πληροφορικής μέσω του Διαδικτύου με pay-as-you-go τιμολόγηση. Αντί να αγοράζετε, να κατέχετε και να διατηρείτε φυσικά κέντρα δεδομένων και διακομιστές, μπορείτε να αποκτήσετε πρόσβαση σε τεχνολογικές υπηρεσίες όπως υπολογιστές, αποθήκευση και βάσεις δεδομένων, όπως απαιτείται, χρησιμοποιώντας έναν πάροχο cloud όπως το Amazon Web Services (AWS).

Η υπολογιστική νέφος και ειδικά η αποθήκευση στο νέφος (cloud storage) προσφέρει πολλά πλεονεκτήματα όταν χρησιμοποιείται στο σχεδιασμό ανάκαμψης από καταστροφές.

Ποιος χρησιμοποιεί το cloud computing;

Οργανισμοί όλων των τύπων, μεγεθών και βιομηχανιών χρησιμοποιούν το cloud για μια ευρεία ποικιλία περιπτώσεων χρήσης, όπως δημιουργία αντιγράφων ασφαλείας δεδομένων, ανάκτηση καταστροφών, email, εικονικούς επιτραπέζιους υπολογιστές, ανάπτυξη και δοκιμή λογισμικού, αναλυτικά δεδομένα μεγάλων δεδομένων και εφαρμογές ιστού.

Για παράδειγμα, οι εταιρείες υγειονομικής περίθαλψης χρησιμοποιούν το cloud για να αναπτύξουν πιο εξατομικευμένες θεραπείες για τους ασθενείς. Οι εταιρείες χρηματοοικονομικών υπηρεσιών χρησιμοποιούν το cloud ως βάση για τον εντοπισμό και την πρόληψη της απάτης σε πραγματικό χρόνο. Και οι κατασκευαστές βιντεοπαιχνιδιών χρησιμοποιούν το cloud για να προσφέρουν διαδικτυακά παιχνίδια σε εκατομμύρια παίκτες σε όλο τον κόσμο.

Οι επιχειρήσεις με χαμηλά έσοδα μπορούν να επωφεληθούν από την μαζική επεξεργασία και την αποθήκευση μεγάλου όγκου δεδομένων με πολύ χαμηλό κόστος. Επίσης, η αρχιτεκτονική «αποθήκευσης στο νέφος» διακρίνεται από υψηλή αποδοτικότητα και επεκτασιμότητα, ενώ ταυτόχρονα μειώνουν την ανάγκη ύπαρξης του πλεονάζοντος εξοπλισμού.

Οφέλη υπολογιστικού νέφους

Ευκινησία

Το cloud σας παρέχει εύκολη πρόσβαση σε ένα ευρύ φάσμα τεχνολογιών, ώστε να μπορείτε να καινοτομείτε πιο γρήγορα και να δημιουργείτε σχεδόν οτιδήποτε μπορείτε να φανταστείτε. Μπορείτε να δημιουργήσετε γρήγορα πόρους όπως απαιτείται, από υπηρεσίες υποδομής όπως υπολογιστές, αποθήκευση και βάσεις δεδομένων έως το Διαδίκτυο των πραγμάτων, τη μηχανική εκμάθηση, τις λίμνες δεδομένων, την ανάλυση δεδομένων και πολλά άλλα.

Μπορείτε να αναπτύξετε υπηρεσίες τεχνολογίας μέσα σε λίγα λεπτά και να μεταβείτε από την ιδέα στην εφαρμογή με ευελιξία αρκετές τάξεις μεγέθους μεγαλύτερη από πριν. Έτσι έχετε την ελευθερία να πειραματιστείτε, να δοκιμάσετε νέες ιδέες για να

διαφοροποιήσετε τις εμπειρίες των πελατών και να μεταμορφώσετε την επιχείρησή σας.

Ελαστικότητα

Με το cloud computing, δεν χρειάζεται να διαθέτετε υπερβολικά πόρους για να απορροφήσετε μελλοντικές αιχμές επιχειρηματικής δραστηριότητας. Αντίθετα, παρέχετε το ποσό των πόρων που χρειάζεστε πραγματικά. Μπορείτε να κλιμακώσετε αμέσως αυτούς τους πόρους προς τα πάνω ή προς τα κάτω για να προσαρμόσετε τη χωρητικότητα καθώς εξελίσσονται οι ανάγκες της επιχείρησής σας.

Εξοικονόμηση κόστους

Το cloud σας επιτρέπει να ανταλλάσσετε σταθερά έξοδα (κέντρα δεδομένων και φυσικούς διακομιστές) με μεταβλητά έξοδα και να πληρώνετε μόνο για το IT που καταναλώνεται. Επίσης, τα μεταβλητά έξοδα είναι πολύ χαμηλότερα από αυτά που θα πληρώνατε μόνοι σας λόγω οικονομίων κλίμακας.

Παγκόσμια ανάπτυξη σε λίγα λεπτά

Με το cloud, μπορείτε να κλιμακώσετε τις δραστηριότητες σε νέες γεωγραφικές περιοχές και να τις αναπτύξετε παγκοσμίως μέσα σε λίγα λεπτά. Για παράδειγμα, το AWS διαθέτει υποδομή σε όλο τον κόσμο, η οποία σας επιτρέπει να αναπτύξετε εφαρμογές σε πολλές φυσικές τοποθεσίες με λίγα μόνο κλικ. Η προσέγγιση των εφαρμογών στους τελικούς χρήστες μειώνει τον λανθάνοντα χρόνο και βελτιώνει την εμπειρία για τους τελικούς χρήστες.

Τύποι Cloud Computing

Το cloud computing ενδυναμώνει τους προγραμματιστές και τα τμήματα πληροφορικής, ώστε να μπορούν να επικεντρωθούν σε ό,τι πραγματικά έχει σημασία και να αποφεύγουν παρόμοιες εργασίες όπως η απόκτηση, η συντήρηση και ο προγραμματισμός χωρητικότητας.

Καθώς η δημοτικότητα του υπολογιστικού νέφους έχει αυξηθεί, έχει προκύψει μια σειρά από διαφορετικά μοντέλα και στρατηγικές ανάπτυξης για την αντιμετώπιση αυτών των ειδικών αναγκών διαφορετικών χρηστών. Κάθε τύπος υπηρεσίας cloud και

μέθοδος ανάπτυξης παρέχει διαφορετικά επίπεδα ελέγχου, ευελιξίας και διαχείρισης. Η κατανόηση των διαφορών μεταξύ του Infrastructure as a Service, του Platform as a Service και του Software as a Service, καθώς και των στρατηγικών ανάπτυξης που μπορείτε να χρησιμοποιήσετε, μπορεί να σας βοηθήσει να αποφασίσετε ποιο σύνολο υπηρεσιών είναι κατάλληλο για τις ανάγκες σας

Μοντέλα υπολογιστικού νέφους

Υπάρχουν τρία κύρια μοντέλα υπολογιστικού νέφους. Κάθε μοντέλο αντιπροσωπεύει ένα διαφορετικό μέρος της στοίβας υπολογιστικού νέφους.

Υποδομή ως υπηρεσία (IaaS)

Το Infrastructure as a Service, που μερικές φορές συντομεύεται ως IaaS, περιέχει τα δομικά στοιχεία του *cloud IT* και γενικά παρέχει πρόσβαση (εικονική ή σε αποκλειστικό υλικό) σε πόρους δικτύου και υπολογιστή, καθώς και χώρο για αποθήκευση δεδομένων. Το Infrastructure as a Service σας προσφέρει το υψηλότερο επίπεδο ευελιξίας και διαχειριστικού ελέγχου των πόρων πληροφορικής σας και μοιάζει πολύ με τους σημερινούς πόρους πληροφορικής που γνωρίζουν σήμερα πολλά τμήματα πληροφορικής και προγραμματιστές.

Πλατφόρμα ως υπηρεσία (PaaS):

Με την πλατφόρμα ως υπηρεσία, οι εταιρείες δεν χρειάζεται πλέον να διαχειρίζονται την υποκείμενη υποδομή (συνήθως υλικό και λειτουργικά συστήματα), επιτρέποντάς σας να εστιάσετε στην ανάπτυξη και τη διαχείριση των εφαρμογών σας. Αυτό σας βοηθά να γίνετε πιο αποτελεσματικοί καθώς εξαλείφει τις ανησυχίες σας σχετικά με την απόκτηση πόρων, τον προγραμματισμό χωρητικότητας, τη συντήρηση λογισμικού, την επιδιόρθωση ή οποιοδήποτε άλλο παρόμοιο είδος βαριάς ανύψωσης που εμπλέκεται στην εκτέλεση της εφαρμογής σας.

Λογισμικό ως υπηρεσία (SaaS)

Το λογισμικό ως υπηρεσία προσφέρει ένα πλήρες προϊόν, το οποίο λειτουργεί και διαχειρίζεται ο πάροχος υπηρεσιών. Στις περισσότερες περιπτώσεις, τα άτομα που αναφέρονται στο λογισμικό ως υπηρεσία αναφέρονται σε εφαρμογές τελικού χρήστη. Με μια προσφορά SaaS, δεν χρειάζεται να σκεφτείτε πώς διατηρείται η υπηρεσία ή πώς γίνεται η διαχείριση της υποκείμενης υποδομής, απλά πρέπει να σκεφτείτε πώς

θα χρησιμοποιήσετε αυτόν τον συγκεκριμένο τύπο λογισμικού. Ένα συνηθισμένο παράδειγμα εφαρμογής SaaS είναι το webmail, όπου μπορείτε να στέλνετε και να λαμβάνετε email χωρίς να χρειάζεται να διαχειρίζεστε πρόσθετους πόρους για το προϊόν email ή να συντηρείτε τους διακομιστές και τα λειτουργικά συστήματα στα οποία εκτελείται το πρόγραμμα email.

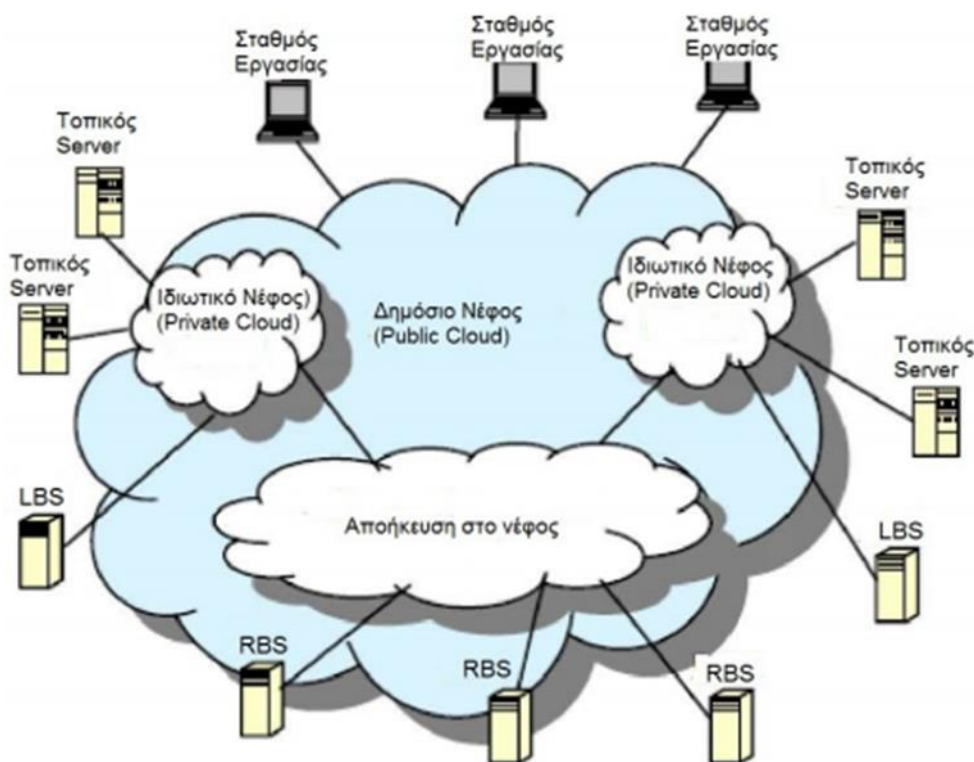
Μοντέλα ανάπτυξης υπολογιστικού νέφους

- Μια εφαρμογή που βασίζεται σε σύννεφο έχει αναπτυχθεί πλήρως στο cloud και όλες οι πτυχές της εφαρμογής εκτελούνται σε αυτό. Οι εφαρμογές στο cloud είτε δημιουργήθηκαν σε αυτό είτε μεταφέρθηκαν από προηγούμενη υποδομή για να απολαύσουν τα οφέλη του υπολογιστικού νέφους. Οι εφαρμογές που βασίζονται στο cloud μπορούν να επωφεληθούν από δευτερεύοντα τμήματα υποδομής ή μπορούν να χρησιμοποιήσουν υπηρεσίες υψηλότερου επιπέδου που μειώνουν τις ανάγκες διαχείρισης, αρχιτεκτονικής και επεκτασιμότητας της βασικής υποδομής.
- Η υβριδική ανάπτυξη είναι ένας τρόπος σύνδεσης υποδομής και εφαρμογών μεταξύ πόρων ιστού και υπαρχόντων πόρων που δεν βρίσκονται στο cloud. Η πιο κοινή μέθοδος υβριδικής ανάπτυξης είναι μεταξύ του cloud και της τρέχουσας εσωτερικής υποδομής για την επέκταση και ανάπτυξη της υποδομής cloud μιας εταιρείας, ενώ οι πόροι cloud συνδέονται με το εσωτερικό σύστημα. Για περισσότερες πληροφορίες σχετικά με το πώς το AWS μπορεί να σας βοηθήσει με την υβριδική σας ανάπτυξη, επισκεφθείτε τη σελίδα μας για υβριδικά.
- Η ανάπτυξη πόρων εσωτερικής εγκατάστασης, χρησιμοποιώντας εργαλεία εικονικοποίησης και διαχείρισης πόρων, μερικές φορές αναφέρεται ως "ιδιωτικό σύννεφο". Η εγκατάσταση εντός εγκατάστασης δεν προσφέρει πολλά από τα πλεονεκτήματα του υπολογιστικού νέφους, αλλά μερικές φορές προτιμάται λόγω της ικανότητάς της να προσφέρει αποκλειστικούς πόρους. Στις περισσότερες περιπτώσεις, αυτό το μοντέλο ανάπτυξης είναι το ίδιο με την παλαιού τύπου υποδομή πληροφορικής, καθώς χρησιμοποιεί τεχνολογίες διαχείρισης εφαρμογών και εικονικοποίησης για να προσπαθήσει να αυξήσει τη χρήση πόρων.

Με την ανάπτυξη μοντέλων υπολογιστικού νέφους σε άνοδο, είναι απαραίτητο να κατανοήσουμε τους διαφορετικούς τύπους επιλογών ανάπτυξης που μπορεί να εξερευνήσει μια επιχείρηση.

Τα δεδομένα των εφαρμογών του συστήματος αποθηκεύονται στους διακομιστές (servers). Όλοι οι διακομιστές έχουν κάποιους back-up διακομιστές. Οι back-up διακομιστές και οι βασικοί διακομιστές μπορούν να βρίσκονται σε διαφορετικές πόλεις.

Ο back-up διακομιστής αποτελείται από τον τοπικό back-up διακομιστή (Local Backup Server - LBS) και τον απομακρυσμένο back-up διακομιστή (Remote Backup Server - RBS). Τα δεδομένα αντιγράφονται πρώτα στον τοπικό back-up διακομιστή και στην συνέχεια στον απομακρυσμένο ώστε οι διακομιστές να παραμένουν συγχρονισμένοι και τα δεδομένα να παραμένουν ασφαλή στην περίπτωση μιας καταστροφής.



Έλεγχος Πρόσβασης στους πόρους του δικτύου

Για την πρόσβαση στους πόρους ενός δικτύου είναι απαραίτητο να έχει προηγηθεί Εξουσιοδότηση (Authorization).

Για την εξουσιοδότηση πρέπει να γίνει αυθεντικοποίηση (authentication) δηλαδή με κάποιο τρόπο να γίνει έλεγχος ταυτότητας αυτού που θέλει να έχει πρόσβαση.

Για κάθε χρήστη ενός πληροφοριακού συστήματος είναι απαραίτητο να τίθενται περιορισμοί σύμφωνα με διάφορα κριτήρια.

Οι περιορισμοί αυτοί πρέπει να λαμβάνουν υπόψη το είδος της συνόδου (Session) (σύνδεση σε τερματικό, δυνατότητα σύνδεσης στο Internet, σε ασύρματο δίκτυο κλπ) και τις εφαρμογές μαζί με τα δεδομένα που μπορεί να έχει πρόσβαση ο χρήστης.

Είναι απαραίτητο να ορισθούν διαφορετικοί περιορισμοί σύνδεσης (login) για να διασφαλιστεί ότι κάθε χρήστης στον οργανισμό έχει επαρκή δικαιώματα πρόσβασης για να εκτελέσει πλήρως τα καθήκοντά του χωρίς περιορισμούς, αλλά τίποτε περισσότερο.

Επειδή κανένα σύνολο των μέτρων ασφαλείας δεν είναι το 100% τέλειο και ένα περιστατικό μπορεί πάντα να συμβεί, πρέπει να γίνεται συλλογή και ανάλυση των δεδομένων που σχετίζονται με την ιστορία της δραστηριότητας του δικτύου.

Αυτό προϋποθέτει ότι πρέπει να καταγράφονται όλα τα γεγονότα πρόσβασης και ότι ένας συνολικός και λεπτομερής κατάλογος συνδέσεων πρέπει να είναι πάντα διαθέσιμος για να διευκολυνθεί η αποτελεσματική έρευνα για τυχόν παραβιάσεις ασφάλειας.

Για τον σκοπό αυτό χρησιμοποιούνται τα αρχεία καταγραφής (log files)

Τα αρχεία καταγραφής περιέχουν πληροφορίες χρήσιμες για την παρακολούθηση της δραστηριότητας στο δίκτυο, όπως είναι χρόνος σύνδεσης, χρήστης, σταθμός εργασίας, αρχεία που έγινε πρόσβαση κλπ.

Για τον έλεγχο ταυτότητας υπάρχουν αρκετές μέθοδοι. Ο πιο συνηθισμένος τρόπος πρόσβασης σε ένα πληροφοριακό σύστημα είναι η πιστοποίηση χρηστών με χρήση κωδικών πρόσβασης.

Ο κωδικός πρόσβασης συνδέεται με ένα όνομα χρήστη και σε συνδυασμό με τα δικαιώματα (ή περιορισμούς) πρόσβασης καθορίζουν τον λογαριασμό χρήστη. Υπάρχουν αρκετοί κανόνες για την δημιουργία ισχυρών κωδικών πρόσβασης ώστε να γίνει δύσκολη η αποκάλυψή τους από επίδοξους εισβολείς. Μερικοί κανόνες είναι:

- Όσο περισσότερους χαρακτήρες έχει ένας κωδικός τόσο πιο δύσκολο είναι να αποκαλυφθεί.
- Πρέπει να περιέχει εκτός από γράμματα αριθμούς και άλλα σύμβολα.
- Δεν περιέχει το όνομα χρήστη, το πραγματικό όνομα ή το όνομα της εταιρίας του ιδιοκτήτη του λογαριασμού.
- Δεν περιέχει μια ολόκληρη λέξη.
- Είναι πολύ διαφορετικός από προηγούμενους κωδικούς πρόσβασης.
- Πρέπει να αλλάζει ανά τακτά χρονικά διαστήματα

Τα τελευταία χρόνια έχουν αρχίσει να χρησιμοποιούνται βιομετρικές μέθοδοι πιστοποίησης χρηστών οι οποίες αντικαθιστούν ή και συμπληρώνουν τους κωδικούς πρόσβασης. Ως βιομετρικές μέθοδοι νοούνται οι τεχνικές πιστοποίησης της ταυτότητας των ατόμων μέσω]ανάλυσης σταθερών χαρακτηριστικών τους. Οι βιομετρικές μέθοδοι μπορούν να ταξινομηθούν σε δύο κατηγορίες:

- α) Στις τεχνικές που στηρίζονται στην ανάλυση φυσικών ή γενετικών χαρακτηριστικών (όπως δακτυλικών αποτυπωμάτων, γεωμετρίας της παλάμης, ανάλυσης της κόρης του ματιού, των χαρακτηριστικών του προσώπου, του DNA) και
- β) Στις τεχνικές που στηρίζονται στην ανάλυση συμπεριφοράς, όπως υπογραφής, φωνής, τρόπου πληκτρολόγησης.

Έλεγχος Πρόσβασης σε Ασύρματο Δίκτυο

Κανένα δίκτυο ή σύστημα δεν είναι απολύτως ασφαλές. Ωστόσο, τα ασύρματα δίκτυα προσθέτουν έναν επιπλέον παράγοντα ασφάλειας σε σύγκριση με τα ενσύρματα δίκτυα. Επειδή χρησιμοποιούν ηλεκτρομαγνητικά κύματα ως μέσο πρόσβασης, είναι πολύ πιο δύσκολο να ελεγχθεί η εμβέλειά τους και μπορούν εύκολα να υπερβούν τα φυσικά όρια της Μονάδας, του Πανεπιστημίου ή ακόμα και του σπιτιού σας, επιτρέποντας έτσι τον εντοπισμό ή τη χρήση τους από μη εξουσιοδοτημένους πρόσωπα. Δεδομένου του χαμηλού κόστους, σε συνδυασμό με την ευκολία εγκατάστασης, χωρίς ουσιαστικά να απαιτείται διαμόρφωση, ακόμη και ένας απλός άνθρωπος μπορεί να εγκαταστήσει γρήγορα μια ασύρματη βάση (σημείο

πρόσβασης), που αγοράστηκε στην τοπική αγορά, χρησιμοποιώντας την τυπική της διαμόρφωση. Η πρακτική αυτή οδηγεί στην υλοποίηση ενός λειτουργικού αλλά ανοιχτού δικτύου, χωρίς ελάχιστα αποδεκτά κριτήρια ασφάλειας και ελέγχου στο δίκτυο.

Η πρόσβαση στους πόρους ενός τοπικού δικτύου μέσω ενός ασύρματου δικτύου προϋποθέτει, την σύνδεση της τερματικής συσκευής με ένα ασύρματο σημείο πρόσβασης (access point).

Στο ενσύρματο τοπικό δίκτυο για την πρόσβαση στο δίκτυο χρειάζεται να χρησιμοποιηθεί ένα καλώδιο το οποίο θα συνδέει τη τερματική συσκευή με τον διανομέα. Αυτό από μόνο του εισαγάγει περιορισμούς στην πρόσβαση, αφού απαιτεί φυσική πρόσβαση στο χώρο (θα πρέπει να συνδεθεί ένα καλώδιο σε κάποια πρίζα δικτύου) που καλύπτεται από το δίκτυο και άρα φυσική παρουσία του χρήστη στον χώρο.

Για να επιτραπεί μόνο η πρόσβαση μόνο σε εξουσιοδοτημένους χρήστες, οι διαχειριστές του δικτύου, μπορούν να επιλέξουν ανάμεσα από έναν αριθμό μεθόδων που μπορούν να χρησιμοποιηθούν για την πρόσβαση. Για την εφαρμογή της επιθυμητής μεθόδου πρέπει ο διαχειριστής να κάνει τις απαραίτητες ρυθμίσεις στο access point.

Οι μηχανισμοί ελέγχου πρόσβασης που υποστηρίζουν σχεδόν όλα τα access points είναι οι παρακάτω:

Αναγνωριστικό συνόλου υπηρεσιών (Service Set Identifier - SSID): Το SSID περιγράφεται από το 802.11 σαν ένα είδος κωδικού πρόσβασης για το χρήστη ώστε να συνδεθεί με κάποιο WLAN. Για να γίνει εφικτή η επικοινωνία πρέπει το access point και η κάρτα δικτύου του εξοπλισμού χρήστη να έχουν το ίδιο SSID. Στην πραγματικότητα το SSID είναι η μόνη υποχρεωτική δικλείδα ασφαλείας που θέσπισε το I.E.E.E. κατά την επικύρωση του πρωτοκόλλου 802.11.

Το πρόβλημα είναι ότι το SSID μεταδίδεται ανοιχτά στον αέρα πολλές φορές το δευτερόλεπτο από το access point μέσα σε κάθε πλαίσιο σινιάλου⁷⁹ (beacon frame).

Πολλοί κατασκευαστές δίνουν την επιλογή να απενεργοποιηθεί το πλαίσιο σινιάλου (να μην αποστέλλεται δηλαδή πλαίσιο για εντοπισμό χρηστών στο χώρο), όμως ο

επίδοξος εισβολέας μπορεί να περιμένει υπομονετικά μέχρι να αναγκαστεί το access point να στείλει πλαίσιο σινιάλου λόγω της μεταγωγής κάποιου χρήστη από άλλο access point (roaming) ή λόγω εισόδου κάποιου νέου χρήστη στο δίκτυο.

Φιλτράρισμα Διευθύνσεων MAC.

Αναφέρεται στο φιλτράρισμα των διευθύνσεων MAC των κεντρικών υπολογιστών που έχουν πρόσβαση στο δίκτυο μέσω της συσκευής, απαγορεύοντας ή επιτρέποντας μόνο ορισμένους κεντρικούς υπολογιστές να έχουν πρόσβαση στο δίκτυο μέσω της συσκευής.

Εφαρμογή φιλτραρίσματος διευθύνσεων MAC

Το πρώτο βήμα είναι να προβάλετε τη διεύθυνση MAC του τοπικού κεντρικού υπολογιστή . Εδώ είναι μια μέθοδος γραμμής εντολών. Στα windows98 και windows me, μέσω της συνάρτησης "run" στο μενού "start", πληκτρολογήστε "winipcfg" και πατήστε Enter για να λάβετε τη διεύθυνση IP του τοπικού κεντρικού υπολογιστή: στα windows2000, windows xp, windows7 και vista, εκτελέστε πρώτα το "cmd Ανοίξτε το παράθυρο της γραμμής εντολών και, στη συνέχεια, εκτελέστε την εντολή "ipconfig/all, θα λάβετε δύο λειτουργίες, συγκεκριμένα Windows IP Configuration και Ethernetadapter, ο 12ψήφιος δεκαεξαδικός αριθμός μετά τη PhysicalAddress κάτω από το στοιχείο του προσαρμογέα Ethernet (τοπική σύνδεση) είναι τη διεύθυνση MAC.

Το δεύτερο βήμα είναι να διαμορφώσετε το φιλτράρισμα διευθύνσεων MAC στο AP (Wireless Access Point) . Μια ασύρματη συσκευή πρέπει να περάσει από δύο βήματα για να αποκτήσει πρόσβαση στο WLAN : έλεγχος ταυτότητας και εξουσιοδότηση. Το φιλτράρισμα της διεύθυνσης MAC πραγματοποιείται μετά τη διαδικασία επαλήθευσης. Η διαδικασία έχει ως εξής:

Όταν η τερματική συσκευή εντοπίσει το σήμα του AP, μπορεί να στείλει ένα αίτημα σύνδεσης. Αρχικά, στέλνει ένα αίτημα ελέγχου ταυτότητας στο AP. Αυτό το αίτημα περιλαμβάνει το SSID του δικτύου προορισμού. Το SSID πρέπει να οριστεί στο AP. Εάν το δίκτυο είναι ανοιχτός δρόμος δικτύου, το SSID είναι κενό, διαφορετικά δεν είναι κενό. Το AP καθορίζει εάν ο έλεγχος ταυτότητας μεταβιβάζεται με βάση το SSID που

αποστέλλεται από τον πελάτη . Αφού περάσει ο έλεγχος ταυτότητας, η συσκευή πελάτη μπορεί να συνδεθεί στο AP. Σε αυτό το σημείο, εμφανίζεται το φιλτράρισμα διευθύνσεων MAC. Υπάρχουν δύο μέθοδοι για το φιλτράρισμα διευθύνσεων MAC. Η πρώτη είναι η μέθοδος της λίστας επιτρεπόμενων, που σημαίνει ότι επιτρέπεται να περάσουν οι πληροφορίες της καθορισμένης διεύθυνσης MAC και οι άλλες διευθύνσεις MAC απορρίπτονται, δηλαδή η άλλο είναι η μαύρη λίστα. Με αυτόν τον τρόπο, οι πληροφορίες της καθορισμένης διεύθυνσης MAC μπορούν να διαβιβαστούν εκτός από το ότι οι πληροφορίες της καθορισμένης διεύθυνσης MAC απορρίπτονται. Μεταξύ αυτών, η μέθοδος της λευκής λίστας έχει σχετικά περισσότερους περιορισμούς και υψηλότερη ασφάλεια και είναι πιο κατάλληλη για χρήση σε καταστάσεις όπου ο τερματικός εξοπλισμός είναι σχετικά σταθερός, ενώ η μέθοδος μαύρης λίστας εφαρμόζεται κυρίως για τον αποκλεισμό ορισμένων χρηστών.

Το φιλτράρισμα διευθύνσεων MAC βρίσκεται στο AP και θα αποτρέψει την πρόσβαση σε μη αξιόπιστες τερματικές συσκευές. Πριν η τελική συσκευή επιχειρήσει να συνδεθεί στο AP, το φιλτράρισμα διευθύνσεων MAC προσδιορίζει μη αξιόπιστες διευθύνσεις MAC και αποκλείει την επικοινωνία , εμποδίζοντάς την να αποκτήσει πρόσβαση στο αξιόπιστο δίκτυο, αλλά η τελική συσκευή μπορεί να συνδεθεί στο AP, αλλά η περαιτέρω πρόσβαση απαγορεύεται. Μπορεί να φανεί ότι δεν αρκεί να βασιζόμαστε μόνο στο φιλτράρισμα διευθύνσεων MAC και θα πρέπει να χρησιμοποιηθούν άλλες στρατηγικές για να επιτευχθεί ένα πιο ασφαλές επίπεδο.

Χρήση πρωτοκόλλων κρυπτογράφησης.

Κατά τη διάρκεια του ελέγχου ταυτότητας συστήματος, ο πελάτης στέλνει ένα μήνυμα στον διακομιστή και ο διακομιστής απαντά στον πελάτη με ένα μήνυμα αυτοελέγχου. Ο πελάτης και ο διακομιστής ολοκληρώνουν μια πρόσθετη ανταλλαγή κλειδιών συνεδρίας, τερματίζοντας ολόκληρη τη διαδικασία ελέγχου ταυτότητας. Αφού ολοκληρωθεί ο έλεγχος ταυτότητας, η ασφαλής επικοινωνία μεταξύ του διακομιστή και του πελάτη χρησιμοποιώντας το συμμετρικό κλειδί που καθιερώθηκε κατά τη διαδικασία ελέγχου ταυτότητας ονομάζεται πρωτόκολλο προστασίας κρυπτογράφησης.

Τα κοινά πρωτόκολλα που μπορούν να χρησιμοποιηθούν για την προστασία κρυπτογράφησης είναι το TLS και το SSL.

Προβλήματα ασφαλείας προκύπτουν όταν τα εμπιστευτικά δεδομένα πρέπει να σταλούν σε ένα μη αξιόπιστο δίκτυο.

Προς το παρόν, το TLS/SSI (Transport Layer Security/Secure Sockets Layer, Transport Layer Security/Secure Sockets Layer) μπορεί να χρησιμοποιηθεί για τον έλεγχο ταυτότητας της ταυτότητας του διακομιστή και του πελάτη, και ταυτόχρονα, των δεδομένων και των δύο μερών επικοινωνίας μπορεί να κρυπτογραφηθεί. Οι εκδόσεις 2.0/3.0 του πρωτοκόλλου SSL και του πρωτοκόλλου PCT (Private Communications Transport, private communications transport) βασίζονται στην τεχνολογία κρυπτογράφησης δημόσιου κλειδιού και παρέχονται στη σουίτα πρωτοκόλλων ελέγχου ταυτότητας Schannel (Κανάλι ασφαλείας, ασφαλές κανάλι). Όλα τα πρωτόκολλα ασφαλών καναλιών βασίζονται στο μοντέλο πελάτη/διακομιστή.

Περιγραφή: Το Secure Channel είναι μέρος του πακέτου ασφαλείας της Microsoft που παρέχει υπηρεσίες ελέγχου ταυτότητας για την παροχή ασφαλών επικοινωνιών μεταξύ πελατών και διακομιστών. Το πακέτο ασφαλείας των Windows είναι ένας πάροχος υποστήριξης ασφαλείας (SSP) που εφαρμόζει τα τυπικά πρωτόκολλα ελέγχου ταυτότητας Διαδικτύου Secure Sockets Layer (SSL) και Transport Layer Security (TLS). Η τεχνολογία Schannel υποστηρίζει τα ακόλουθα πρωτόκολλα υποδομής δημόσιου κλειδιού: SSL2.0/3.0, TLS1.0 και PCT1.0.

Το TLS/SSL περιλαμβάνει στην πραγματικότητα δύο πρωτόκολλα, το TLS και το SSL. Το TLS είναι χτισμένο με βάση την υποδομή δημόσιου κλειδιού, χρησιμοποιώντας ζεύγη δημόσιου/ιδιωτικού κλειδιού για την παροχή κρυπτογράφησης και ελέγχου ακεραιότητας για δεδομένα και πιστοποιητικά X.509 για έλεγχο ταυτότητας. Η υψηλότερη έκδοση του πρωτοκόλλου TLS είναι η 1.0 και η υψηλότερη έκδοση του SSL είναι η 3.0. Το TLS1.0 έχει αναπτυχθεί με βάση το SSL3.0. Οι κύριες λειτουργίες και οι αρχές λειτουργίας των TLS1.0 και SSL3.0 είναι βασικά οι ίδιες. Επειδή το TLS είναι ένα ανοιχτό πρότυπο που δημοσιεύεται από το IETF, υποστηρίζεται ευρέως. Στο μέλλον, πολλά πρωτόκολλα ασφαλείας του SSL θα αντικατασταθούν πλήρως. Για παράδειγμα, το Kerberos v5 κρυπτογραφεί και αποκρυπτογραφεί δεδομένα με βάση ένα μόνο κλειδί. Επομένως, αυτά τα πρωτόκολλα βασίζονται στην ασφαλή ανταλλαγή

κλειδιών. Στο πρωτόκολλο Kerberos v5, αυτό επιτυγχάνεται μέσω διακριτικών που λαμβάνονται από το KDC (Κέντρο Διανομής Κλειδιών). Αυτό απαιτεί από κάθε χρήστη που χρησιμοποιεί το πρωτόκολλο Kerberos να εγγραφεί στο KDC, κάτι που δεν είναι πρακτικό για ιστότοπους ηλεκτρονικού εμπορίου με εκατομμύρια χρήστες από όλο τον κόσμο. Το TLS βασίζεται στην υποδομή δημόσιου κλειδιού (PKI), η οποία χρησιμοποιεί δύο κλειδιά για την κρυπτογράφηση και την αποκρυπτογράφηση δεδομένων. Όταν το ένα χρησιμοποιείται για κρυπτογράφηση, το άλλο πρέπει να χρησιμοποιείται για αποκρυπτογράφηση. Το όφελος μιας τέτοιας αρχής κρυπτογράφησης είναι ότι δεν υπάρχει ανάγκη παροχής ασφαλούς ανταλλαγής κλειδιών.

Το ιδιωτικό κλειδί στο PKI μπορεί να ανήκει μόνο στον καταχωρίζοντα και το δημόσιο κλειδί μπορεί να δημοσιευτεί στο κοινό. Εάν κάποιος έστειλε ένα ιδιωτικό μήνυμα στον κάτοχο του κλειδιού, τα δεδομένα θα μπορούσαν να κρυπτογραφηθούν χρησιμοποιώντας το δημόσιο κλειδί του παραλήπτη και ο παραλήπτης θα μπορούσε στη συνέχεια να αποκρυπτογραφήσει τα δεδομένα χρησιμοποιώντας το ιδιωτικό του κλειδί. Τα ζεύγη κλειδιών μπορούν επίσης να χρησιμοποιηθούν για ελέγχους ακεραιότητας δεδομένων. Ο κάτοχος του κλειδιού μπορεί να επισυνάψει μια υπογραφή με το δικό του ιδιωτικό κλειδί πριν στείλει δεδομένα. Δημιουργήστε μια ψηφιακή υπογραφή που περιλαμβάνει τον υπολογισμό ενός κατακερματισμού των δεδομένων και την κρυπτογράφηση των δεδομένων με το ιδιωτικό κλειδί. Οποιοσδήποτε έχει το δημόσιο κλειδί του αποστολέα των δεδομένων μπορεί να επαληθεύσει ότι τα δεδομένα προέρχονται από τον σωστό αποστολέα αποκρυπτογραφώντας την υπογραφή δεδομένων. Επιπλέον, ο παραλήπτης μπορεί να χρησιμοποιήσει τον ίδιο αλγόριθμο με τον αποστολέα για να κατακερματίσει τα ληφθέντα δεδομένα. Εάν το αποτέλεσμα του υπολογισμού είναι συνεπές με τον κατακερματισμό που επισυνάπτεται στην υπογραφή δεδομένων, ο παραλήπτης μπορεί να επιβεβαιώσει ότι τα δεδομένα δεν έχουν τροποποιηθεί κατά τη μετάδοση. .

Στην περίπτωση πολλών διακομιστών που παρέχουν έλεγχο ταυτότητας σε πελάτες, το TLS/SSL δεν απαιτεί την αποθήκευση κλειδιών διακομιστή σε έναν ελεγκτή τομέα (όπως η υπηρεσία Windows Active Directory της Microsoft) ή σε μια βάση δεδομένων. Ο πελάτης χρησιμοποιεί το πιστοποιητικό που εκδόθηκε από την αξιόπιστη αρχή

έκδοσης πιστοποιητικών (που φορτώνεται αυτόματα μετά την εγκατάσταση του συστήματος Windows Server 2003) για να επιβεβαιώσει την αξιοπιστία . Δεν χρειάζεται να δημιουργήσετε αντίστοιχο λογαριασμό για να συνδεθείτε με ασφάλεια.

Τα πρωτόκολλα TLS και SSL χρησιμοποιούνται ευρέως στη μετάδοση ασφάλειας στο Διαδίκτυο με βάση το πρωτόκολλο HTTPS μεταξύ προγραμμάτων περιήγησης ιστού και διακομιστών ιστού (δηλαδή, η λεγόμενη λειτουργία B/S, η οποία είναι στην πραγματικότητα η λειτουργία πελάτη/διακομιστή (C/S). Το TLS/SSL μπορεί επίσης να εφαρμοστεί σε άλλα πρωτόκολλα επιπέδου εφαρμογής , όπως πρωτόκολλα FTP, LDAP και SMTP . Το TLS/SSI μπορεί να παρέχει μέτρα ασφαλείας, όπως έλεγχο ταυτότητας διακομιστή, έλεγχο ταυτότητας πελάτη και έλεγχο ακεραιότητας δεδομένων για επικοινωνίες δικτύου μέσω δημόσιων δικτύων όπως το Διαδίκτυο. Επιπλέον, δεν μπορεί να εφαρμοστεί μόνο σε ενσύρματα δίκτυα, αλλά έχει επίσης ένα παρόμοιο εκτεταμένο πρότυπο μετάδοσης ασφαλείας-WTLS σε ασύρματη μετάδοση δικτύου WLAN

Προγράμματα Προστασίας από Ιούς

Η εργασία που δίνεται από το λογισμικό προστασίας από ιούς είναι να παρακολουθεί τη συμπεριφορά των προγραμμάτων υπολογιστή ανά πάσα στιγμή και να σαρώνει το σύστημα για κακόβουλα προγράμματα όπως ιούς.

Ορισμένα λογισμικά προστασίας από ιούς μπορούν να ξεκινήσουν με μόνιμα προγράμματα μετά την ενεργοποίηση του λειτουργικού συστήματος. Το λογισμικό προστασίας από ιούς έχει διαφορετικές τεχνικές για παρακολούθηση σε πραγματικό χρόνο. Θα χρησιμοποιήσει μέρος του χώρου για να συγκρίνει τα χαρακτηριστικά του τρέχοντος προγράμματος με τη βάση δεδομένων ιών για να προσδιορίσει εάν πρόκειται για κακόβουλο πρόγραμμα.

Άλλο λογισμικό προστασίας από ιούς θα χρησιμοποιήσει κάποιο χώρο για να προσομοιώσει τις ενέργειες που επιτρέπονται από το σύστημα ή τον χρήστη, έτσι ώστε το δοκιμασμένο πρόγραμμα να εκτελεί τις απαιτήσεις του εσωτερικού κώδικα και να μπορεί να κριθεί εάν είναι ιός σύμφωνα με τις ενέργειες του προγράμματος. Τελικά τα πρόγραμμα προστασίας από ιούς είναι απαραίτητα πρόγραμμα που πρέπει

να είναι εγκατεστημένα σε κάθε συσκευή που συνδέεται στο δίκτυο (τοπικό ή Internet).

Η αντιμετώπιση των ιών έχει δύο σκέλη:

- τον εντοπισμό του ιού και
- την απάλειψή του. Τα προγράμματα προστασίας από ιούς πραγματοποιούν έλεγχο των αρχείων της συσκευής για τον εντοπισμό μολυσματικού λογισμικού. Τα αρχεία αυτά μπορεί να είναι αρχεία δεδομένων, αρχεία συστήματος, ή αρχεία εφαρμογών. Επίσης, μπορεί να είναι αποθηκευμένα σε κάποια μονάδα βοηθητικής μνήμης ή να εισέρχονται στο σύστημα μέσω δικτύου (Τοπικού ή Internet). Τα σημερινά προγράμματα προστασίας από ιούς εντοπίζουν σχεδόν όλα τα είδη κακόβουλου λογισμικού (ιούς, σκουλήκια, key loggers κ.α.)

Ο τρόπος σάρωσης του σκληρού δίσκου είναι ο ίδιος με το πρώτο πρόγραμμα εκτέλεσης παρακολούθησης σε πραγματικό χρόνο που αναφέρθηκε παραπάνω, αλλά εδώ, το λογισμικό προστασίας από ιούς θα εκτελέσει έναν έλεγχο σύμφωνα με τις ανάγκες του χρήστη (το εύρος ορισμού της σάρωσης).

Επιπλέον, το λογισμικό προστασίας από ιούς περιλαμβάνει περισσότερη τεχνολογία σάρωσης:

- Τεχνολογία σάρωσης συμπιεσμένων αρχείων : Είναι μια τεχνολογία που αναλύει και ελέγχει συμπιεσμένα αρχεία και συσκευασμένα αρχεία.
- Προστασία από παραβίαση προγραμμάτων: Αυτό αποτρέπει τα κακόβουλα προγράμματα από το να καταστρέψουν τον υπολογιστή σας καταργώντας προγράμματα εντοπισμού ιών.
- Τεχνολογία επιδιόρθωσης: δηλαδή επαναφορά αρχείων που έχουν καταστραφεί από κακόβουλα προγράμματα
- Emergency Disk Antivirus: Χρησιμοποιήστε έναν κενό δίσκο U για να δημιουργήσετε έναν δίσκο εκκίνησης έκτακτης ανάγκης για τον εντοπισμό ιών υπολογιστών.

- Έξυπνη σάρωση: Σαρώνστε τους πιο συχνά χρησιμοποιούμενους δίσκους και βασικές τοποθεσίες συστήματος σε σύντομο χρονικό διάστημα.
- Πλήρης σάρωση: Σάρωση όλων των δίσκων του υπολογιστή, η οποία διαρκεί πολύ.
- Ransomware Protection: Προστατέψτε τα αρχεία στον υπολογιστή σας από κακόβουλη κρυπτογράφηση από χάκερ.
- Σάρωση ενεργοποίησης: Σαρώνει αυτόματα όταν ο υπολογιστής είναι ενεργοποιημένος, ο οποίος μπορεί να σαρώσει συμπιεσμένα έγγραφα και δυνητικά ανεπιθύμητα προγράμματα.

Το λογισμικό προστασίας από ιούς είναι ένα σύστημα ανάλυσης πληροφοριών, το οποίο παρακολουθεί όλες τις ροές δεδομένων (συμπεριλαμβανομένων: μνήμη - δίκτυο σκληρού δίσκου - δίκτυο μνήμης - σκληρός δίσκος), όταν διαπιστώσει ότι ορισμένες πληροφορίες έχουν μολυνθεί, θα αφαιρέσει τον ιό. Ο τρόπος με τον οποίο αναλύονται (ή σαρώνονται) οι πληροφορίες εξαρτάται από την προέλευσή τους και το λογισμικό προστασίας από ιούς λειτουργεί διαφορετικά κατά την παρακολούθηση μονάδων οπτικού δίσκου, email ή κίνησης δεδομένων μεταξύ LAN.

Τοποθεσία παρακολούθησης λογισμικού προστασίας από ιούς:

- Παρακολούθηση μνήμης: Όταν εντοπιστεί ένας ιός στη μνήμη, θα ειδοποιήσει ενεργά, θα παρακολουθεί όλες τις διαδικασίες, θα παρακολουθεί τα αρχεία που διαβάζονται στη μνήμη, θα παρακολουθεί τα δεδομένα δικτύου που διαβάζονται στη μνήμη.
- Παρακολούθηση αρχείου: Όταν διαπιστώσει ότι υπάρχει ιός στο αρχείο γραμμένο στο δίσκο ή έχει μολυνθεί από ιό, θα αναφέρει ενεργά την αστυνομία.
- Παρακολούθηση email: Αποκλεισμός όταν εντοπίζονται ιοί στα συνημμένα email.
- Προστασία Ιστού: Αποκλείει επιθέσεις δικτύου και μη ασφαλείς λήψεις.
- Προστασία συμπεριφοράς: Ειδοποιεί τους χρήστες για ύποπτη συμπεριφορά εφαρμογής.

Internet Security

Ασφάλεια Διαδικτύου είναι η ασφάλεια των ενεργειών και των συναλλαγών που γίνονται στο Διαδίκτυο. Η ασφάλεια στο Διαδίκτυο αποτελεί μέρος ευρύτερων εννοιών, όπως η ασφάλεια στον κυβερνοχώρο και η ασφάλεια των υπολογιστών και περιλαμβάνει την ασφάλεια του προγράμματος περιήγησης και του δικτύου, καθώς και τη σωστή συμπεριφορά στο διαδίκτυο. Όταν περνάτε σημαντικό χρόνο στο διαδίκτυο, ενδέχεται να αντιμετωπίσετε τις ακόλουθες απειλές για την ασφάλεια του Διαδικτύου:

- Hacking - απόκτηση πρόσβασης από μη εξουσιοδοτημένους χρήστες σε συστήματα υπολογιστών, λογαριασμούς email και ιστότοπους.
- Ιοί και κακόβουλο λογισμικό που μπορούν να καταστρέψουν δεδομένα και να καταστήσουν τα συστήματα ευάλωτα σε άλλες απειλές.
- Κλοπή ταυτότητας, όπως προσωπικές και οικονομικές πληροφορίες από χάκερ.

Τα άτομα και οι οργανισμοί μπορούν να προστατευτούν από τέτοιες απειλές χρησιμοποιώντας τεχνικές ασφάλειας στο Διαδίκτυο.

Έλεγχος περιεχομένου:

Έλεγχος περιεχομένου είναι ο περιορισμός της πρόσβασης ενός χρήστη σε συγκεκριμένο πληροφοριακό υλικό (π.χ. μία ιστοσελίδα). Πολλές φορές αναφέρεται και ως Γονικός Έλεγχος, όταν ο περιορισμός της πρόσβασης αφορά ανηλίκους.

Για την εφαρμογή του ελέγχου περιεχομένου χρησιμοποιούνται προγράμματα ελέγχου περιεχομένου τα οποία ρυθμίζουν την πρόσβαση σε πληροφορίες ή υπηρεσίες στο Διαδίκτυο σύμφωνα με καθορισμένα κριτήρια.

Μπορούν να εγκατασταθούν στον υπολογιστή του χρήστη, σε ένα κεντρικό υπολογιστή που ανήκει σε κάποιο φορέα όπως είναι ένας διακομιστής μεσολάβησης

(proxy server) σε ένα σχολείο ή τους υπολογιστές ενός παρόχου υπηρεσιών Διαδικτύου (ISP) και μπορούν να αντιδράσουν ποικιλοτρόπως:

Μπορούν να προειδοποιήσουν για προβληματικές ιστοσελίδες, να καταγράψουν λεπτομερώς τις κινήσεις ενός χρήστη στο Διαδίκτυο, να μπλοκάρουν ύποπτους ιστοχώρους, να επιτρέπουν την πρόσβαση συγκεκριμένες ώρες και ημέρες, ακόμα και να κλείσουν τελείως τον υπολογιστή.

Τυπικό παράδειγμα εφαρμογής ελέγχου περιεχομένου είναι υπηρεσία η «Ασφαλούς Πρόσβασης στον Παγκόσμιο Ιστό» (αποκλεισμός ακατάλληλου περιεχομένου) που έχει υλοποιήσει το Πανελλήνιο Σχολικό Δίκτυο (ΠΣΔ).

Με την υπηρεσία αυτή αποκόπτεται η πρόσβαση σε ιστοσελίδες που περιέχουν:

- μηνύματα για το μίσος, τη βία και προπαγανδίζουν την επιθετική συμπεριφορά,
- μηνύματα για την προώθηση ή τη χρήση ναρκωτικών ουσιών
- τυχερά παιχνίδια
- πορνογραφικό περιεχόμενο
- ρατσιστικά μηνύματα ή προωθούν το ρατσισμό
- Η υπηρεσία είναι διαφανής (transparent proxy), δηλαδή οι χρήστες δεν μπορούν να την παρακάμψουν.

Η σημασία της ενημέρωσης (update)

Η χρήση του πιο πρόσφατου λειτουργικού συστήματος είναι ένας από τους σημαντικούς τρόπους διατήρησης της ασφάλειας των πληροφοριών του υπολογιστή και αποφυγής σφαλμάτων του συστήματος. Σε αυτό το άρθρο, θα εξηγήσουμε συγκεκριμένα πώς να διατηρήσετε τον υπολογιστή στην καλύτερη κατάσταση ενημερώνοντας το σύστημα και τα προγράμματα του υπολογιστή.

Βεβαιωθείτε ότι το σύστημα του υπολογιστή σας είναι πάντα ενημερωμένο

Κάθε ενημέρωση του λειτουργικού συστήματος μπορεί να φέρει περισσότερο λογισμικό αυτόματης εκκίνησης, αλλά τα περισσότερα από αυτά είναι απαραίτητο

λογισμικό εργασιών για τη βελτίωση της απόδοσης του συστήματος και τη βελτίωση της ταχύτητας του συστήματος.

Η έγκαιρη ενημέρωση του συστήματος μπορεί να αποφύγει καλύτερα την εισβολή ιών και άλλου κακόβουλου λογισμικού, επειδή οι χάκερ συνήθως χρησιμοποιούν τα τρωτά σημεία του λειτουργικού συστήματος για να στείλουν λογισμικό ή προγράμματα ιών. Η έγκαιρη ενημέρωση του συστήματος με επιδιορθωμένα τρωτά σημεία μπορεί να προστατεύσει καλύτερα την ασφάλεια και χρήση προσωπικών πληροφοριών Ασφάλεια.

Ενημέρωση υπολογιστών Microsoft® Windows®

Τα Windows 10 είναι ένα σύστημα που ενημερώνεται εύκολα. Το Windows Update, που βρίσκεται στην περιοχή Ενημέρωση και ασφάλεια του Πίνακα Ελέγχου, εγκαθιστά αυτόματα τις ενημερώσεις των Windows και τις ενημερώσεις για ορισμένες εφαρμογές. Εάν θέλετε να βεβαιωθείτε ότι το λογισμικό σας είναι ενημερωμένο, μπορείτε επίσης να ελέγξετε μη αυτόματα για ενημερώσεις κάνοντας κλικ στο εικονίδιο των Windows, επιλέγοντας Πίνακας Ελέγχου, στη συνέχεια Ενημέρωση και ασφάλεια και, στη συνέχεια, Windows Update. Κάντε κλικ στην επιλογή Έλεγχος για ενημερώσεις για να ελέγξετε μη αυτόματα για πρόσθετες ενημερώσεις.

Για παλαιότερες εκδόσεις των Windows, θα πρέπει να ελέγξετε για να βεβαιωθείτε ότι το Windows Update δεν είναι απενεργοποιημένο. Εάν λάβετε ένα μήνυμα που σας ζητά να εγκαταστήσετε μια σημαντική ενημέρωση, μπορείτε να εγκαταστήσετε την ενημέρωση κατά τον ελεύθερο χρόνο σας.

Όταν εμφανίζεται μια προτροπή επανεκκίνησης κατά τη διαδικασία ενημέρωσης, προσπαθήστε να κάνετε επανεκκίνηση αμέσως σύμφωνα με την προτροπή. Αυτή η διαδικασία θα διαρκέσει πολύ, αλλά εάν η επανεκκίνηση καθυστερήσει, η ουρά του έργου μπορεί να είναι μεγαλύτερη ή να σταματήσει να αναζητά άλλες ενημερώσεις.

Ενημέρωση λογισμικού προστασίας από ιούς

Το λογισμικό προστασίας από ιούς θα πρέπει να λειτουργεί και να ενημερώνεται συνεχώς για να διατηρείται η ασφάλεια του υπολογιστή σας. Υπάρχουν πολλοί τύποι λογισμικού προστασίας από ιούς, συμπεριλαμβανομένου λογισμικού προστασίας

από spyware, anti-adware και anti-malware. Οι χρήστες μπορούν να ρυθμίσουν τον αυτόματο έλεγχο και εγκατάσταση ενημερώσεων για να διασφαλίσουν ότι ο υπολογιστής είναι απαλλαγμένος από κακόβουλο λογισμικό.

Τείχος Προστασίας

Το τείχος προστασίας αποτελεί αναπόσπαστο μέρος των λειτουργικών συστημάτων των Windows από το XP SP2. Τα παλαιότερα συστήματα χρησιμοποιούσαν Τείχος προστασίας σύνδεσης στο Internet, το οποίο ήταν απενεργοποιημένο από προεπιλογή. Αυτό οδήγησε σε παγκόσμιες επιθέσεις από σκουλήκια όπως τα Blaster και Sasser, τα οποία μόλυναν συλλογικά περισσότερους από 350.000 υπολογιστές παγκοσμίως το 2003 και το 2004. Το τείχος προστασίας μπορεί να είναι είτε για έναν υπολογιστή είτε για ολόκληρο το τοπικό δίκτυο. Γενικά, εκτελεί τις ακόλουθες λειτουργίες:

- Προστασία του συστήματος από εξωτερικές επιθέσεις. Η λίστα τέτοιων απειλών περιλαμβάνει σάρωση θυρών, πλαστογράφηση IP, επιθέσεις DDoS, εικασία κωδικού πρόσβασης.
- Μπλοκάρισμα διαρροών. Εάν έχει εισέλθει κακόβουλο λογισμικό στον υπολογιστή μέσω USB ή CD, τότε το τείχος προστασίας, εάν έχει ρυθμιστεί σωστά, θα αποτρέψει περαιτέρω εξάπλωση στο δίκτυο.
- Έλεγχος εφαρμογών. Το τείχος προστασίας σας επιτρέπει να διαμορφώσετε την πρόσβαση στο δίκτυο για κάθε μεμονωμένη εφαρμογή.
- Προστασία ζώνης. Παροχή διαφορετικών επιπέδων πρόσβασης στο τοπικό δίκτυο.
- Καταγραφή και προειδοποίηση. Το τείχος προστασίας όχι μόνο συλλέγει στατιστικά στοιχεία, αλλά προειδοποιεί επίσης τους χρήστες για διάφορες ενέργειες.

Το τείχος προστασίας δεν είναι μόνο σε λειτουργικά συστήματα. Το λογισμικό του δρομολογητή περιλαμβάνει επίσης ένα ενσωματωμένο τείχος προστασίας, το οποίο συνήθως ρυθμίζεται μέσω μιας διεπαφής web.

Από τι προστατεύει ένα τείχος προστασίας και σε τι δεν βοηθά;

Το τείχος προστασίας είναι η πρώτη γραμμή άμυνας του υπολογιστή σας και μπορεί να αντιμετωπίσει αποτελεσματικά τους ακόλουθους τύπους απειλών:

- Σκουλήκια υπολογιστών και ορισμένοι ιοί. Τα σκουλήκια έχουν τον δικό τους κώδικα, επομένως δεν χρειάζονται συγκεκριμένα αρχεία για να μολύνουν. Από αυτή την άποψη, τέτοιες απειλές είναι πιο επικίνδυνες.
- Hacking με χρήση απομακρυσμένης επιφάνειας εργασίας. Με το τείχος προστασίας απενεργοποιημένο, οι εισβολείς μπορούν να αποκτήσουν πρόσβαση στα αρχεία σας και ακόμη και να πάρουν τον έλεγχο.
- Διάφορα spyware. Ορισμένα προγράμματα στέλνουν πληροφορίες σχετικά με το σύστημα ή τις ενέργειες του χρήστη σε τρίτους εν αγνοία σας. Ένα τείχος προστασίας μπορεί να αποτρέψει τη διαρροή δεδομένων περιορίζοντας την εξερχόμενη κυκλοφορία.
- Πρόσβαση από κερκόπορτες. Οι χάκερ συχνά χρησιμοποιούν διάφορα τρωτά σημεία σε λογισμικό, συμπεριλαμβανομένων των ανοιχτών θυρών. Το τείχος προστασίας αποκλείει οποιαδήποτε μη εξουσιοδοτημένη κυκλοφορία, μειώνοντας την πιθανότητα εκμετάλλευσης τέτοιων τρωτών σημείων.
- Επιθέσεις DDoS. Οι αλγόριθμοι που χρησιμοποιούνται ανιχνεύουν αποτελεσματικά τέτοιες επιθέσεις αναλύοντας επαναλαμβανόμενα αιτήματα από συγκεκριμένες διευθύνσεις IP.

Ένα τείχος προστασίας δεν είναι ικανό να παρέχει πλήρη προστασία στον υπολογιστή σας. Υπάρχει μια σειρά από απειλές που δεν μπορεί να αντιμετωπίσει. Το πρώτο είναι ιοί και σκουλήκια που έχουν ήδη εισέλθει στον υπολογιστή. Το τείχος προστασίας σαρώνει μόνο την κυκλοφορία δικτύου και δεν αναλύει απευθείας το σύστημα αρχείων. Αυτός είναι ο λόγος για τον οποίο οι υπολογιστές πρέπει να διαθέτουν ένα πλήρες πρόγραμμα προστασίας από ιούς που εντοπίζει και αφαιρεί υπάρχοντες ιούς.

Το τείχος προστασίας δεν μπορεί να σας προστατεύσει από κακόβουλους συνδέσμους που λαμβάνετε μέσω spam email. Επίσης, ένας υπολογιστής μπορεί να μολυνθεί από κακόβουλο λογισμικό όχι μέσω του δικτύου - μονάδες USB, μονάδες οπτικού δίσκου, κάρτες μνήμης κ.λπ. Η ανάγνωση και η αντιγραφή αρχείων από αυτά τα μέσα δεν ελέγχεται από το τείχος προστασίας.

Πλεονεκτήματα και μειονεκτήματα της χρήσης τείχους προστασίας

Το κύριο πλεονέκτημα της χρήσης του είναι η αυξημένη ασφάλεια. Στον εταιρικό τομέα, αυτή είναι μια απαραίτητη προστασία που θα αποτρέψει τις εξωτερικές εισβολές, θα περιορίσει την πρόσβαση στο Διαδίκτυο στους υπαλλήλους και θα κάνει ασφαλείς τις μεταφορές αρχείων μέσω FTP και άλλων πρωτοκόλλων. Για τους απλούς χρήστες, το τείχος προστασίας θα μειώσει την πιθανότητα εμφάνισης σκουληκιών και επίσης θα περιορίσει τη δραστηριότητα των "ύποπτων" προγραμμάτων.

Η χρήση τείχους προστασίας σε ένα λειτουργικό σύστημα έχει πολλά μειονεκτήματα:

- Πτώση απόδοσης. Ένα τείχος προστασίας που λειτουργεί καταναλώνει πόρους CPU και RAM, και λόγω της συνεχούς σάρωσης της κυκλοφορίας, οι χρήστες ενδέχεται να αντιμετωπίσουν μια μικρή πτώση στην ταχύτητα πρόσβασης στο Διαδίκτυο.
- Ψευδοθετικά. Οι αλγόριθμοι του τείχους προστασίας δεν είναι τέλειοι, επομένως μπορεί να "ορκιστεί" στη δουλειά των προγραμμάτων προστασίας από ιούς, torrent και άλλων αξιόπιστων προγραμμάτων.
- Δυσκολία ρύθμισης. Εάν θέλετε να επιτύχετε το μέγιστο επίπεδο προστασίας, τότε οι κανόνες για τις εισερχόμενες και εξερχόμενες συνδέσεις θα πρέπει να ρυθμιστούν χειροκίνητα.

Πρωτόκολλα Ασφαλείας Ασύρματων Δικτύων

Ο πλέον συνηθισμένος τρόπος προστασίας ενός οικιακού ασύρματου δικτύου ή του ασύρματου δικτύου μιας μικρής επιχείρησης είναι με τη χρήση των πρωτοκόλλων ασφαλείας που είναι ενσωματωμένα σε κάθε σημείο πρόσβασης. Τα πρωτόκολλα αυτά προσφέρουν κρυπτογράφηση των δεδομένων που διακινούνται, και μόνο οι χρήστες που γνωρίζουν το κλειδί της κρυπτογράφησης έχουν την δυνατότητα να συνδεθούν και να επικοινωνήσουν με το σημείο πρόσβασης και κατά συνέπεια με το τοπικό ασύρματο δίκτυο.

Τα κυριότερα πρωτόκολλα είναι:

WEP (Wired Equivalent Privacy):

Το Wired Equivalent Privacy συντομογραφία WEP είναι ένα σύστημα για την προστασία της ασφάλειας πληροφοριών ασύρματου δικτύου .

Το WEP χρησιμοποιεί κρυπτογράφηση κλειδιού 64 bit ή 128 bit (24 bit χρησιμοποιείται για το αρχικό διάνυσμα (Initialization Vector, IV)), επομένως ο διαθέσιμος κωδικός πρόσβασης 64 bit (Secret Key) είναι $64-24=40$ bit (WEP-40), δηλαδή 10 Δεκαεξαδικός αριθμός . Ο διαθέσιμος κωδικός πρόσβασης 128 bit είναι $128-24=104$ bit (WEP-104), δηλαδή 26 δεκαεξαδικά ψηφία. Αυτή η χρήση έχει χρησιμοποιηθεί ευρέως και είναι επίσης η προεπιλεγμένη διαμόρφωση των δρομολογητών για τους χρήστες εκείνη την εποχή.

Επειδή τα ασύρματα δίκτυα εκπέμπουν μηνύματα μέσω ραδιοφώνου , είναι ιδιαίτερα ευάλωτα στην υποκλοπή . Το WEP έχει σχεδιαστεί για να παρέχει το ίδιο επίπεδο εμπιστευτικότητας με τα παραδοσιακά ενσύρματα LAN , εξ ου και το όνομα. Ωστόσο , οι κρυπτοαναλυτές έχουν εντοπίσει αρκετές αδυναμίες στην ενσύρματη ισοδύναμη κρυπτογράφηση, η οποία αντικαταστάθηκε από την προστατευμένη πρόσβαση Wi-Fi (WPA) το 2003 και με την κυκλοφορία του πλήρους προτύπου IEEE 802.11i το 2004,

Αρχικά το WEP ήταν εφαρμόσιμο μόνο στα 802.11a και 802.11b και η πρώτη γενιά του WPA που ακολούθησε ήταν εφαρμόσιμη στα 802.11g. Ωστόσο, καθώς ορισμένοι κατασκευαστές αναβαθμίζουν το υλικολογισμικό των προϊόντων τους 802.11b, μπορεί επίσης να χρησιμοποιηθεί κρυπτογράφηση WPA.

Το Wired Equivalent Privacy (WEP) είναι ένας αλγόριθμος ασφαλείας πίσω από το ασύρματο δίκτυο IEEE 802.11 . Καθιερώθηκε ως μέρος του αρχικού προτύπου 802.11, το οποίο επικυρώθηκε το 1997, πρόθεσή του ήταν να παρέχει απόρρητο δεδομένων συγκρίσιμο με ένα παραδοσιακό ενσύρματο δίκτυο .

WPA (Wi-Fi Protected Access) και WPA2:

Είναι ένα πρόγραμμα πιστοποίησης που δημιουργήθηκε από τη Wi-Fi Alliance για να αποδείξει τη συμμόρφωση με τα πρωτόκολλα ασφαλείας που δημιουργήθηκαν από

τη Wi-Fi Alliance για την προστασία των ασύρματων δικτύων υπολογιστών. Το πρωτόκολλο δημιουργήθηκε ως απάντηση σε αρκετές σοβαρές αδυναμίες που ανακαλύφθηκαν από ερευνητές σε ένα προηγούμενο σύστημα, το Wired Equivalent. Μία από τις πιο σημαντικές αλλαγές μεταξύ WPA και WPA2 είναι η υποχρεωτική χρήση του αλγορίθμου AES και η εισαγωγή του CCMP (Counter Mode Cipher Block Chaining Message Complete Code Protocol) αντί του TKIP.

Επί του παρόντος, τα κύρια ελαττώματα ασφαλείας στα συστήματα WPA2 είναι ασήμαντα (ο εκμεταλλευτής πρέπει να εκτελέσει μια επίθεση man-in-the-middle, να λάβει εξουσιοδότηση από το δίκτυο και στη συνέχεια να συνεχίσει να επιτίθεται σε άλλες συσκευές στο δίκτυο). Ως αποτέλεσμα, τα γνωστά τρωτά σημεία στο WPA2 περιορίζονται σχεδόν αποκλειστικά σε δίκτυα εταιρικής ποιότητας. Επομένως, είναι αμφισβητήσιμο να συζητηθεί εάν το WPA2 είναι ασφαλές στο οικιακό δίκτυο.

Δυστυχώς, το WPA2 πάσχει από την ίδια αδυναμία που προκαλεί το WPA, τον φορέα επίθεσης με προστασία Wi-Fi Setup (WPS). Παρόλο που η επίθεση σε ένα προστατευμένο δίκτυο WPA/WPA2 απαιτεί 2 έως 14 ώρες συνεχούς επίθεσης χρησιμοποιώντας έναν σύγχρονο υπολογιστή, πρέπει να ανησυχούμε για αυτό το ζήτημα ασφαλείας και οι χρήστες θα πρέπει να απενεργοποιήσουν το WPS (εάν είναι δυνατόν, το υλικολογισμικό θα πρέπει να ενημερωθεί έτσι ώστε η συσκευή να μην υποστηρίζει πλέον WPS by Αυτό εξαλείφει εντελώς το διάνυσμα επίθεσης).

Η μέθοδος κρυπτογράφησης WPA2 είναι αυτή τη στιγμή η πιο ευρέως χρησιμοποιούμενη μέθοδος ασύρματης κρυπτογράφησης.

Με το πρότυπο 802.11i, ολόκληρη η αλυσίδα της μονάδας ασφαλείας γίνεται πιο αξιόπιστη και αποτελεσματική προστασία από μη στοχευμένες και στοχευμένες επιθέσεις. Το σύστημα WPA2 επιτρέπει στον διαχειριστή του δικτύου Wi-Fi να μεταβεί από ζητήματα ασφαλείας στη λειτουργία και τη διαχείριση συσκευών.

Το πρότυπο 802.11r είναι μια τροποποίηση του προτύπου 802.11i. Αυτό το πρότυπο επικυρώθηκε τον Ιούλιο του 2008. Η τυπική τεχνολογία μεταφέρει πιο γρήγορα και αξιόπιστα βασικές ιεραρχίες που βασίζονται στην τεχνολογία Handoff (μεταφορά ελέγχου) κατά τη μετακίνηση του χρήστη μεταξύ σημείων πρόσβασης. Το πρότυπο 802.11r είναι πλήρως συμβατό με τα πρότυπα WiFi 802.11a/b/g/n.

Υπάρχει επίσης το πρότυπο 802.11w, το οποίο έχει σχεδιαστεί για να βελτιώνει τον μηχανισμό ασφαλείας που βασίζεται στο πρότυπο 802.11i. Αυτό το πρότυπο έχει σχεδιαστεί για την προστασία των πακέτων ελέγχου.

Τα 802.11i και 802.11w είναι οι μηχανισμοί ασφαλείας για δίκτυα WiFi 802.11n.

Κρυπτογράφηση αρχείων και φακέλων στα Windows 7

Η δυνατότητα κρυπτογράφησης σας επιτρέπει να κρυπτογραφείτε αρχεία και φακέλους που αργότερα θα είναι δυσανάγνωστα σε άλλη συσκευή χωρίς ειδικό κλειδί. Αυτή η δυνατότητα υπάρχει σε εκδόσεις των Windows 7 όπως Professional, Enterprise ή Ultimate. Τα παρακάτω θα καλύψουν τρόπους για να ενεργοποιήσετε την κρυπτογράφηση αρχείων και φακέλων.

Ενεργοποίηση κρυπτογράφησης αρχείων:

Έναρξη -> Υπολογιστής (επιλέξτε το αρχείο για κρυπτογράφηση)-> κάντε δεξί κλικ στο αρχείο->Ιδιότητες->Για προχωρημένους(καρτέλα Γενικά)->Πρόσθετα χαρακτηριστικά->Βάλτε έναν δείκτη στο Κρυπτογράφηση περιεχομένου για προστασία δεδομένων->OK->Εφαρμογή- > Ok (Επιλέξτε την εφαρμογή μόνο για αρχείο)->Close Properties (Κλείσιμο διαλόγου) (Κάντε κλικ στο Ok ή Close).

Ενεργοποίηση κρυπτογράφησης φακέλου:

Έναρξη -> Υπολογιστής (επιλέξτε τον φάκελο για κρυπτογράφηση)-> κάντε δεξί κλικ στο φάκελο-> Ιδιότητες->Για προχωρημένους(καρτέλα Γενικά)->Πρόσθετα χαρακτηριστικά->Βάλτε έναν δείκτη στο Κρυπτογράφηση περιεχομένου για προστασία δεδομένων->OK->Εφαρμογή- > Ok (Επιλέξτε την εφαρμογή μόνο για αρχείο)->Close Properties (Κλείσιμο διαλόγου) (Κάντε κλικ στο Ok ή Close).

Και στις δύο περιπτώσεις, η περιοχή ειδοποιήσεων θα σας ζητήσει να δημιουργήσετε ένα αντίγραφο ασφαλείας του κλειδιού των κρυπτογραφημένων αρχείων ή φακέλων. Στο παράθυρο διαλόγου, μπορείτε να επιλέξετε πού θα αποθηκεύσετε το αντίγραφο του κλειδιού (συνιστάται αφαιρούμενο μέσο). Το αρχείο ή ο φάκελος θα αλλάξει το χρώμα του μετά την επιτυχή κρυπτογράφηση.

Η κρυπτογράφηση είναι εξαιρετικά χρήσιμη εάν μοιράζεστε τον υπολογιστή εργασίας σας με πολλά άτομα.

Πεδίο εφαρμογής

Στις περισσότερες περιπτώσεις, τα ασύρματα δίκτυα (χρησιμοποιώντας σημεία πρόσβασης και δρομολογητές) κατασκευάζονται για εμπορικούς σκοπούς για να προσελκύσουν κέρδη από πελάτες και ενοικιαστές. Οι εργαζόμενοι της εταιρείας Get Wifi έχουν εμπειρία στην προετοιμασία και υλοποίηση των ακόλουθων έργων για την υλοποίηση δικτυακής υποδομής με βάση ασύρματες λύσεις.

Σύνοψη Ενότητας

Τελειώνοντας αυτή η ενότητα ο εκπαιδευόμενος κατανοεί και γνωρίζει, τι είναι δεδομένα, τα δομικά στοιχεία των ηλεκτρονικών υπολογιστών. Γνωρίζει για τα είδη δικτύων, για το διαδίκτυο και μαθαίνει για τα συστήματα ασφαλείας για την πληροφορική

Ενότητα 2 ερωτήσεις

A . Τι εκτελεί η ΚΜΕ;

1. Εκτελεί τις αριθμητικές και λογικές πράξεις που περιγράφει το λογισμικό.
2. Εκτελεί τις εντολές του χρήστη.
3. Δεν εκτελεί αποθηκεύει

B. Επιλέξτε την σωστή απάντηση στο τι είναι RAM;

1. Η προγραμματισμένη από τον κατασκευαστή μνήμη η οποία είναι με βασικές παραμέτρους λειτουργίας του υπολογιστή
2. Η κύρια μνήμη του υπολογιστή, στην οποία αποθηκεύονται τα δεδομένα του χρήστη, τα λογισμικά λειτουργικού συστήματος και εφαρμογών, καθώς όλα τα αποτελέσματα από την επεξεργασία των δεδομένων.

Γ. Επιλέξτε τι είναι Δίκτυο υπολογιστών;

1. Είναι ένα σύστημα από υπολογιστές διασυνδεδεμένους που μπορούν να επικοινωνήσουν μεταξύ τους
2. Είναι ένα σύστημα από υπολογιστές αυτονόμους, ή μη αυτόνομους που μπορούν να επικοινωνήσουν μεταξύ τους
3. Όλα τα παραπάνω

Δ. Τι είναι Επικοινωνιακός δίαυλος;

1. Είναι ένα σύστημα από υπολογιστές διασυνδεδεμένους ή αυτονόμους, ή μη αυτόνομους που μπορούν να επικοινωνήσουν μεταξύ τους.
2. Είναι ένα μέσο μεταφοράς δεδομένων ανάμεσα σε δύο υπολογιστικά συστήματα, ενσύρματο ή ασύρματο.

Ε. Επιλέξτε τι ονομάζουμε πρωτόκολλο;

1. Το σύνολο από συμβάσεις που ορίζουν το πώς ανταλλάσσουν μεταξύ τους δεδομένα οι υπολογιστές του δικτύου.
2. Το σύνολο των δεδομένων που καταχωρούνται και αποθηκεύονται στην κεντρική μνήμη του υπολογιστή

Ζ. Επιλέξτε τι ονομάζουμε κίνηση σε δίκτυα υπολογιστών;

1. Σημαίνει τη μετακίνηση δεδομένων σε μια σύνδεση δικτύου από ένα σημείο σε άλλο σε μια συγκεκριμένη χρονική στιγμή και μετριέται σε πολλαπλάσια ενός byte.
2. Η κίνηση είναι μια ροή δεδομένων πληροφοριών (traffic σημαίνει "κίνηση" από τα αγγλικά) από το διαδίκτυο στον υπολογιστή.

Η. Σε τι χωρίζεται η κυκλοφορία ;

1. εξερχόμενες (πληροφορίες που εισέρχονται στο εξωτερικό δίκτυο).
2. εισερχόμενες (πληροφορίες που προέρχονται από το εξωτερικό δίκτυο).
3. Όλα τα παραπάνω

Θ. Με άλλα λόγια τι είναι η επεξεργασία δεδομένων;

1. Η επεξεργασία δεδομένων είναι οποιαδήποτε ρουτίνα που περιλαμβάνει την ανάλυση μεγάλου όγκου πληροφοριών.

2. Είναι η ανεύρεση διαδικτυακών πληροφοριών .
3. Είναι η αποθήκευση των διαδικτυακών πληροφοριών

Ι. Ένα υπολογιστικό σύστημα, ποια μέρη περιλαμβάνει ;

1. Κεντρική Μονάδα Επεξεργασίας
2. Κύρια Μνήμη, Περιφερειακή Μνήμη
3. Όλα τα παραπάνω

Κ. Τι χρησιμεύει η μνήμη RAM σε έναν υπολογιστή;

1. Χρειάζεται για γρήγορη χρήση δεδομένων.
2. Για ενίσχυση του λογισμικού του υπολογιστή .
3. Για πιο μεγάλο χώρο αποθήκευσης

Άσκηση

Ένας υπολογιστής έχει μνήμη 128 MB. Αν κάθε λέξη μνήμης είναι 8 bytes, πόσα bit πρέπει να έχει ο δίαυλος διευθύνσεων?

Βιβλιογραφία

Γκρίτζαλης Στέφανος, Κάτσικας Σωκράτης, Γκρίτζαλης Δημήτρης (2003) Ασφάλεια Δικτύων Υπολογιστών: Τεχνολογίες και Υπηρεσίες σε περιβαλλοντα ηλεκτρονικού Επιχειρείν & Ηλεκτρονικής Διακυβέρνησης

Φραντζεσκάκης Ιωάννης, Γκόλιας Ιωάννης (1994) Οδική Ασφάλεια

Laudon, Jane Price & Laudon, Kenneth C. (2021) Πληροφοριακά συστήματα διοίκησης.

ΚΟΜΕΡ Ε. ΝΤΑΓΚΛΑΣ (2001) Διαδίκτυα με TCP/ IP Αρχές Πρωτοκόλλα και Αρχιτεκτονικές .

Βικιπαίδεια

Εκπαιδευτική Ενότητα : 3 Εισαγωγή στον προγραμματισμό

Σκοπός

Στην ενότητα αυτή ο εκπαιδευόμενος θα αντιληφθεί και θα γνωρίσει την έννοια του προγράμματος. Θα γνωρίσει την ιστορία των γλωσσών προγραμματισμού και των ειδών προγραμματισμού, καθώς και τις τεχνικές που χρησιμοποιούνται για τη σωστή δημιουργία προγραμμάτων. Θα κατανοήσει τις τεχνικές της ιεραρχικής

σχεδίασης προγραμμάτων και τη διαδικασία που ακολουθείται από τη σύνταξη του προγράμματος μέχρι την τελική του εκτέλεση.

Προσδοκώμενα Αποτελέσματα

Τελειώνοντας την ενότητα ο εκπαιδευόμενος θα μπορεί να χρησιμοποιεί τα στάδια αντιμετώπισης ενός προβλήματος, μπορεί να αναφέρει τα χαρακτηριστικά ενός αλγορίθμου, θα περιγράψει έναν αλγόριθμο με λεκτική (ψευδοκώδικα) και συμβολική αναπαράσταση, θα σχεδιάζει αλγορίθμους με εφαρμογή βασικών πρακτικών προγραμματισμού, και με εφαρμογή μεθοδολογιών σχεδιασμού προγραμμάτων και να εφαρμόζουν τεχνικές προγραμματισμού.

Έννοιες – Κλειδιά

Γλώσσα προγραμματισμού

Ο άνθρωπος μιλάει έχει σαν μέσω μιας γλώσσας, ο υπολογιστής καταλαβαίνει δυαδικά.

Για να μπορέσουν λοιπόν και τα δύο μέρη να επικοινωνήσουν, χρειάζεται ένας ενδιάμεσος, μια γλώσσα πραγματισμού.

Η γλώσσα προγραμματισμού είναι μια γραπτή, επίσημη γλώσσα που καθορίζει ένα σύνολο οδηγιών και κανόνων που χρησιμοποιούνται για τη δημιουργία προγραμμάτων (λογισμικό). Ένα λογισμικό μπορεί να αναπτυχθεί για να τρέχει σε υπολογιστή, φορητή συσκευή ή οποιονδήποτε εξοπλισμό που επιτρέπει την εκτέλεσή του. Υπάρχουν πολλές γλώσσες και εξυπηρετούν πολλούς σκοπούς. Κάποια προφανή, όπως η δημιουργία λογισμικού, άλλα λιγότερο, όπως ο έλεγχος ενός αυτοκινήτου ή μιας τοστιέρας.

Μια εφαρμογή που εκτελείται στο τηλέφωνό σας είναι ένας τύπος λογισμικού που δημιουργήθηκε με μια γλώσσα προγραμματισμού. Ένα παιχνίδι, που τρέχει σε υπολογιστή ή σε άλλο εξοπλισμό, επίσης.

Μια έξυπνη τηλεόραση, η οποία έχει τη λειτουργικότητα να συνδέεται στο διαδίκτυο και να επιτρέπει στον καταναλωτή να χρησιμοποιεί πρόγραμμα περιήγησης, χρησιμοποιεί γλώσσες προγραμματισμού τόσο σε υλικό όσο και σε λογισμικό.

Μια τοστιέρα δεν έχει ψηφιακή διεπαφή επικοινωνίας, αλλά διαθέτει μικροελεγκτή με λογισμικό που διαχειρίζεται τον εξοπλισμό. Αυτός ο τύπος λογισμικού, ενσωματωμένος σε ηλεκτρονικά κυκλώματα, ονομάζεται υλικολογισμικό και είναι επίσης γραμμένος χρησιμοποιώντας μια γλώσσα προγραμματισμού. Το διαδίκτυο των πραγμάτων (IoT) απαιτεί πολλές λύσεις που χρησιμοποιούν αυτόν τον τύπο τεχνολογίας.

Γλώσσα μηχανής

Η γλώσσα προγραμματισμού που περιλαμβάνει εντολές γραμμένες σε μορφή ακολουθιών bit εκτελέσιμες από την Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ).

Γλώσσα προγραμματισμού χαμηλού επιπέδου

Η γλώσσα που είναι κοντά στην γλώσσα μηχανής γι' αυτό και ονομάζεται χαμηλού επιπέδου.

Υψηλού επιπέδου γλώσσα προγραμματισμού (high-level programming language)

Επιτρέπει τη μεταφερσιμότητα ενός προγράμματος από έναν ,σε έναν άλλο υπολογιστή . Χαρακτηριστικά

- Τρόπος έκφρασης κοντά στον ανθρώπινο
- Ευκολία εκμάθησης
- Ευκολότερη διόρθωση λαθών και συντήρηση προγραμμάτων σε σχέση με χαμηλότερα επίπεδα.

Μεταγλωττιστής (Compiler)

Μεταφράζει μια γλώσσα υψηλού επιπέδου σε γλώσσα επιπέδου μηχανής.

Διερμηνευτής (Interpreter) εκτελεί άμεσα ένα πρόγραμμα σε γλώσσα υψηλού επιπέδου.

Λογικό διάγραμμα είναι μια διαγραμματική μορφή απεικόνισης μιας διαδικασίας η οποία αφορά σε σύστημα ή πρόγραμμα.

Αλγόριθμος ονομάζεται μία σειρά από εντολές που έχουν αρχή και τέλος, είναι σαφείς και έχουν ως σκοπό την επίλυση κάποιου προβλήματος

Κάθε γλώσσα προγραμματισμού έχει το δικό της σύνολο τυπικών προδιαγραφών (ή κανόνων) που αφορούν το συντακτικό, το λεξιλόγιο και το νόημα της.

Για πολλές γλώσσες που χρησιμοποιούνται ευρέως και έχουν χρησιμοποιηθεί για αρκετό χρονικό διάστημα (π.χ. C, C++, Java, Scheme), υπάρχουν ειδικοί φορείς τυποποίησης, οι οποίοι μέσα από τακτές συναντήσεις δημιουργούν, τροποποιούν ή επεκτείνουν τις τυπικές προδιαγραφές που διέπουν τη χρήση μιας γλώσσας προγραμματισμού.

Άλλες γλώσσες δεν περιγράφονται σε κάποιο επίσημο πρότυπο αλλά ορίζονται μόνο με βάση κάποια υλοποίησή τους (που αποτελεί το ντε φάκτο πρότυπο), όπως η Python που περιγράφεται από την υλοποίηση CPython.

Εκπαιδευτική Υποενότητα: 3.1 Εισαγωγικές έννοιες

Προγραμματισμός Η/Υ

Ο υπολογιστής

Ένας υπολογιστής (γνωστός και ως ηλεκτρονικός υπολογιστής) είναι μια συσκευή που χρησιμοποιεί ψηφιακή ηλεκτρονική τεχνολογία για να καθοδηγήσει και να εκτελέσει αυτόματα οποιαδήποτε ακολουθία αριθμητικών ή λογικών πράξεων σύμφωνα με μια σειρά οδηγιών. Οι υπολογιστές γενικής χρήσης είναι ικανοί να εκτελούν ένα εξαιρετικά ευρύ φάσμα εργασιών λόγω της ικανότητάς τους να ακολουθούν ένα γενικό σύνολο λειτουργιών που ονομάζονται " προγράμματα ". Οι υπολογιστές χρησιμοποιούνται ως συστήματα ελέγχου για διάφορες βιομηχανικές και καταναλωτικές συσκευές. Αυτό περιλαμβάνει απλές συσκευές ειδικής χρήσης (όπως φούρνοι μικροκυμάτων και τηλεχειριστήρια), βιομηχανικές συσκευές (όπως βιομηχανικά ρομπότ και σχεδιασμός με τη βοήθεια υπολογιστή) και συσκευές γενικής χρήσης (όπως προσωπικούς υπολογιστές και κινητές συσκευές όπως smartphone). Αν και υπάρχουν πολλά είδη υπολογιστών, σύμφωνα με τη θεωρία των μηχανών Turing , ένας υπολογιστής με βασικές λειτουργίες θα πρέπει να μπορεί να κάνει ό,τι μπορεί να κάνει οποιοσδήποτε άλλος υπολογιστής. Επομένως, θεωρητικά τα πάντα, από ένα smartphone έως έναν υπερυπολογιστή θα πρέπει να μπορούν να

κάνουν την ίδια δουλειά (ανεξαρτήτως χρόνου και αποθήκευσης). Λόγω της ταχείας τεχνολογικής προόδου, η επόμενη γενιά υπολογιστών ήταν πάντα σε θέση να ξεπερνά σημαντικά τους προκατόχους τους, ένα φαινόμενο που μερικές φορές αναφέρεται ως « Νόμος του Μουρ ». Μέσω του Διαδικτύου , οι υπολογιστές συνδέονται μεταξύ τους, γεγονός που βελτιώνει σημαντικά την ταχύτητα της ανταλλαγής πληροφοριών, η οποία με τη σειρά της προωθεί την ανάπτυξη της τεχνολογίας. Στον 21ο αιώνα, η εφαρμογή των υπολογιστών έχει εμπλακεί σε όλες τις πτυχές και σε όλους τους τομείς της ζωής.

Απλές χειροκίνητες συσκευές - όπως ο άβακας - έχουν βοηθήσει τους ανθρώπους να κάνουν υπολογισμούς από την αρχαιότητα. Στις πρώτες μέρες της Βιομηχανικής Επανάστασης , δημιουργήθηκαν μηχανές κάθε είδους για να αυτοματοποιήσουν κουραστικές και κουραστικές εργασίες, όπως η ύφανση μοτίβων στον αργαλειό. Πιο πολύπλοκες μηχανές εμφανίστηκαν στις αρχές του 20ου αιώνα, εκτελώντας σύνθετους και συγκεκριμένους υπολογισμούς μέσω αναλογικών κυκλωμάτων. Οι πρώτοι ψηφιακοί υπολογιστές εμφανίστηκαν κατά τον Β' Παγκόσμιο Πόλεμο. Από τότε, η ταχύτητα, η κατανάλωση ενέργειας και η ευελιξία των υπολογιστών συνέχισαν να αυξάνονται. Στη σύγχρονη εποχή, η εφαρμογή των μηχανικών υπολογιστών έχει αντικατασταθεί πλήρως από τους υπολογιστές.

Οι υπολογιστές διαφέρουν ως προς τη σύνθεση. Υπάρχει ακόμη ένας μεγάλος αριθμός μεγάλων, γιγάντιων υπολογιστών που εξυπηρετούν *ad hoc* επιστημονικές ανάγκες υπολογιστών, ή επεξεργασίας συναλλαγών για μεγάλους οργανισμούς. Οι μικρότεροι σχεδιασμένοι για προσωπικές εφαρμογές ονομάζονται Προσωπικοί Υπολογιστές (PC), οι οποίοι αναφέρονται ως «μικροϋπολογιστές» στην Κίνα. Αυτό αναφέρεται επίσης συνήθως σήμερα στην καθημερινή χρήση της λέξης "υπολογιστής", αλλά η πιο κοινή μορφή εφαρμογής υπολογιστή σήμερα είναι ενσωματωμένη, η οποία είναι συνήθως σχετικά απλή, μικρή και χρησιμοποιείται για τον έλεγχο άλλων συσκευών—είτε αεροσκαφών, βιομηχανικών ρομπότ, ή ψηφιακές φωτογραφικές μηχανές .

Η τεχνική έρευνα που σχετίζεται με τους υπολογιστές ονομάζεται επιστήμη των υπολογιστών και η «τεχνολογία υπολογιστών» αναφέρεται στο άθροισμα πολλών τεχνικών και εμπειρικών αποτελεσμάτων που προέρχονται από την εφαρμογή των αποτελεσμάτων της επιστήμης των υπολογιστών στην πρακτική μηχανικής. Η «τεχνολογία υπολογιστών» και η «επιστήμη των υπολογιστών» είναι δύο έννοιες σχετικές αλλά διαφορετικές. Η διαφορά είναι ότι η πρώτη είναι πιο πρακτική και η δεύτερη πιο θεωρητική. Η έρευνα με επίκεντρο τα δεδομένα ονομάζεται τεχνολογία πληροφοριών .

Παραδοσιακά, οι σύγχρονοι υπολογιστές περιέχουν τουλάχιστον μία μονάδα επεξεργασίας (συνήθως μια κεντρική μονάδα επεξεργασίας (CPU)) και κάποια μορφή μνήμης. Τα στοιχεία επεξεργασίας εκτελούν αριθμητικές και λογικές πράξεις και η μονάδα αλληλουχίας και ελέγχου μπορεί να αλλάξει τη σειρά των λειτουργιών σε απόκριση στις αποθηκευμένες πληροφορίες. Τα περιφερειακά περιλαμβάνουν συσκευές εισόδου (πληκτρολόγιο , ποντίκι , joystick , κ.λπ.), συσκευές εξόδου (οθόνη οθόνης, εκτυπωτής κ.λπ.) και συσκευές εισόδου/εξόδου που εκτελούν και τις δύο λειτουργίες (π.χ. οθόνη αφής). Τα περιφερειακά επιτρέπουν την ανάκτηση πληροφοριών από εξωτερικές πηγές και επιτρέπουν την αποθήκευση και ανάκτηση των αποτελεσμάτων των λειτουργιών.

Γλώσσα μηχανής και γλώσσες .

Τι είναι η γλώσσα μηχανής;

Ίσως όλοι γνωρίζουν ότι οι υπολογιστές δεν μπορούν να αναγνωρίσουν τη φυσική γλώσσα όπως οι άνθρωποι. Αν δεν με πιστεύετε, δοκιμάστε το. Εάν πείτε στον υπολογιστή, "*Παρακαλώ ανοίξτε το WeChat*", οι άνθρωποι πιθανότατα σκέφτονται στην καρδιά τους: "Είμαι φοβάστε ότι αυτό το άτομο δεν είναι ανόητο!". Εφόσον ένας υπολογιστής δεν μπορεί να καταλάβει τη φυσική γλώσσα, ποια γλώσσα μπορεί να καταλάβει; Ναι, είναι γλώσσα μηχανής.

Η γλώσσα μηχανής, όπως υποδηλώνει το όνομα, είναι μια γλώσσα που οι μηχανές μπορούν να καταλάβουν, αλλά οι άνθρωποι δεν πρέπει να καταλάβουν. Δεδομένου

ότι η γλώσσα μηχανής εκφράζεται σε δυαδικό και το δυαδικό έχει μόνο 0 και 1, έχει τα χαρακτηριστικά της απλότητας και της υψηλής απόδοσης και είναι τέλεια για έναν σπασίτσα σαν υπολογιστή.

Αλλά το πρόβλημα εμφανίζεται ξανά. Εάν ανοίξετε τη δυαδική αναπαράσταση του WeChat "10100101" (στην πραγματικότητα μπορεί να είναι μεγαλύτερη), θα ήταν πολύ ενοχλητικό να εισάγετε μια τέτοια σειρά από "αστρονομικούς αριθμούς" κάθε φορά που ανοίγετε το WeChat, οπότε οι άνθρωποι (δεν 't know Who, με ανθρώπινους όρους, είναι σωστό) επινόησε τη γλώσσα assembly.

τι είναι η γλώσσα assembly

Επειδή η γλώσσα μηχανής αντιπροσωπεύεται από το 0 και το 1, δεν είναι εύκολο να το θυμόμαστε, η γλώσσα assembly μπορεί να γίνει κατανοητή ως μνημονική για τη γλώσσα μηχανής. Η γλώσσα συναρμολόγησης αποτελείται από μια σειρά οδηγιών (μνημονικά). Υπάρχει ένας μεγάλος αριθμός δυαδικών πράξεων στη διαδικασία λειτουργίας του υπολογιστή, όπως πρόσθεση, αφαίρεση, πολλαπλασιασμός και διαίρεση κ.λπ., και αυτές οι πράξεις μπορούν να ολοκληρωθούν με οδηγίες, όπως η εντολή ADD για πρόσθεση, ακολουθούμενη από τελεστές, όπως π.χ. ως ΠΡΟΣΘΗΚΗ 9, 9 για την ολοκλήρωση της λειτουργίας προσθήκης μίας φορές.

Η γλώσσα assembly μπορεί επίσης να γίνει άμεσα κατανοητή (εκτελεσμένη) από υπολογιστές, αλλά στην ουσία, η γλώσσα assembly θα εξακολουθεί να μεταφράζεται σε δυαδικό, αλλά είναι ευκολότερο για τους ανθρώπους να την κατανοήσουν μέσω οδηγιών.

Τι είναι μια γλώσσα υψηλού επιπέδου;

Στη συνέχεια, ας μιλήσουμε για γλώσσες υψηλού επιπέδου. Εξάλλου, η γλώσσα assembly εξακολουθεί να είναι δυσνόητη και πολύ περίπλοκη. Μπορεί να απαιτεί μεγάλο αριθμό οδηγιών για την ολοκλήρωση μεγάλου αριθμού λειτουργιών για να

ανοίξετε το WeChat. Επομένως, η γλώσσα assembly είναι επίσης μια γλώσσα χαμηλού επιπέδου και όσο πιο κοντά είναι στη φυσική γλώσσα, τόσο πιο προηγμένη είναι. Με την ανάπτυξη των υπολογιστών, υπάρχουν αμέτρητες γλώσσες υψηλού επιπέδου. Οι πιο δημοφιλείς είναι η γλώσσα C, η C++, η Java, η Python κ.λπ. Όλοι αυτοί είναι επαγγελματικοί όροι. Όσοι δεν έχουν μάθει ποτέ γλώσσες προγραμματισμού μπορεί να μην έχουν ακούσει τους, αλλά δεν πειράζει. Εάν υπάρχει μια τόσο υψηλού επιπέδου γλώσσα, το "open WeChat" μπορεί να αντιπροσωπεύεται από μια τέτοια γραμμή κώδικα: "start wechat" και κοιτάξτε, η μετάφραση είναι απλώς για να ανοίξετε το WeChat, κάτι που είναι υπέροχο.

Οι γλώσσες υψηλού επιπέδου μεταφράζονται απευθείας σε δυαδικά και, σε διαφορετικά λειτουργικά συστήματα, τα αποτελέσματα της μετάφρασης είναι διαφορετικά, κάτι που καθορίζεται από τα χαρακτηριστικά του λειτουργικού συστήματος. Για παράδειγμα, στο σύστημα των Windows, το start wechat θα μεταφραστεί ως: 10100101 και στο macOS θα μεταφραστεί ως: 10101111. Αν και τα αποτελέσματα μετάφρασης και των δύο είναι διαφορετικά, μπορούν και τα δύο να ανοίξουν το WeChat. Εάν μεταφέρετε το μεταφρασμένο 10100101 στο σύστημα των Windows στο macOS και το εκτελέσετε, δεν θα μπορείτε να ανοίξετε το WeChat, επειδή το WeChat μπορεί να ανοίξει μόνο με την εκτέλεση του 10101111 στο macOS.

Γενική κατάσταση

1. Ένας μεγάλος αριθμός ασήμαντων λεπτομερειών συγκρατεί τους προγραμματιστές, καθιστώντας τους αδύνατο να έχουν περισσότερο χρόνο και ενέργεια για να ασχοληθούν με δημιουργική εργασία και να εκτελέσουν εργασίες που είναι πιο σημαντικές για αυτούς. Έτσι ώστε να διασφαλίζεται η ορθότητα και η αποτελεσματικότητα του προγράμματος.
2. Οι προγραμματιστές δεν πρέπει μόνο να ελέγχουν τη συνολική κατάσταση του σχεδιασμού του προγράμματος, αλλά και να εμβαθύνουν σε κάθε μέρος μέχρι τις λεπτομέρειες της εφαρμογής. είναι φτωχή και ο κύκλος ανάπτυξης είναι μακρύς. .
3. Επειδή ο τρόπος σκέψης και έκφρασης του προγραμματισμού στη γλώσσα μηχανής είναι πολύ διαφορετικός από τις συνήθειες των ανθρώπων, μόνο οι προγραμματιστές που έχουν παρακολουθήσει μακροχρόνια επαγγελματική

κατάρτιση μπορούν να είναι ικανοί, γεγονός που κάνει τον προγραμματισμό υψηλό και χαμηλό.

4. Επειδή η γραπτή του μορφή είναι όλος κωδικός «password», άρα η αναγνωσιμότητα είναι κακή, και δεν βολεύει για επικοινωνία και συνεργασία.

5. Επειδή εξαρτάται σε μεγάλο βαθμό από τον συγκεκριμένο υπολογιστή, έχει κακή φορητότητα και κακή επαναχρησιμοποίηση.

Η Γλώσσα Υψηλού και Χαμηλού Επιπέδου και ποια είναι η καλύτερη;

Τι είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου;

Οι γλώσσες προγραμματισμού υψηλού επιπέδου είναι εκείνες που είναι περίπου πιο κοντά στην ανθρώπινη γλώσσα από τη γλώσσα μηχανής.

Φανταστείτε ότι οι υπολογιστές, ή πιο συγκεκριμένα οι επεξεργαστές τους, καταλαβαίνουν μόνο μία γλώσσα: αυτή των δυαδικών αριθμών .

Επομένως, ανεξάρτητα από το μηχάνημα και όποιον κι αν είναι ο επεξεργαστής του, παρά τις θεμελιώδεις διαφορές μεταξύ των μητρικών γλωσσών καθεμιάς από αυτές, όλες μπορούν να επικοινωνούν μόνο μέσω των αριθμών 0 και 1 .

Αυτή η γλώσσα, που ονομάζεται γλώσσα μηχανής , είναι αυτό που ονομάζουμε γλώσσα χαμηλού επιπέδου .

Από την άλλη πλευρά, οι γλώσσες υψηλού επιπέδου , σε αντίθεση με τους αντιπάλους τους, είναι αυτές που, αν και επικοινωνούν με λογισμικό διερμηνείας, έχουν σύνταξη πολύ πιο κοντά σε αυτές που χρησιμοποιούνται από τις γλώσσες που μιλούν οι άνθρωποι.

Φυσικά, κανείς δεν θα επικοινωνεί σε Python σε κύκλο φίλων, αλλά επειδή η Python είναι μια γλώσσα προγραμματισμού που χρησιμοποιεί τα αγγλικά ως βασική γλώσσα, ακόμα κι αν δεν έχετε μεγάλες γνώσεις σχετικά με την ανάπτυξη ιστού, όταν δείτε έναν κώδικα Python θα να είναι σε θέση να εξάγει τη μία ή την άλλη γλωσσική κατανόηση.

Αυτές οι γλώσσες ονομάζονται γλώσσες υψηλού επιπέδου.

Ποιες είναι οι κύριες γλώσσες υψηλού επιπέδου;

Οι κορυφαίες γλώσσες υψηλού επιπέδου είναι επίσης οι πιο δημοφιλείς γλώσσες προγραμματισμού στην αγορά σήμερα.

Παρακάτω, θα απαριθμήσουμε τα τρία που χρησιμοποιούνται περισσότερο από τους επαγγελματίες ανάπτυξης Ιστού σήμερα.

JavaScript

Η JavaScript είναι μια από τις παλαιότερες γλώσσες προγραμματισμού στην αγορά, που δημιουργήθηκε στα μέσα της δεκαετίας του 1990, πιο συγκεκριμένα το 1995.

Για να φέρει περισσότερο δυναμισμό σε ιστοσελίδες που μέχρι τότε ήταν στατικές και με λίγους οπτικούς πόρους, το JS άλλαξε πολύ με τα χρόνια.

Σήμερα, η JavaScript είναι μια από τις πιο ευέλικτες γλώσσες προγραμματισμού στην αγορά και, που αρχικά αναπτύχθηκε ως γλώσσα front-end , είναι ήδη δυνατό να βρείτε εντολές JS ακόμη και σε εφαρμογές back-end.

Πύθων

Η Python είναι μια σχετικά νέα γλώσσα προγραμματισμού που εκτός από υψηλού επιπέδου είναι και cross-platform.

Αυτό σημαίνει ότι εκτελείται σε οποιοδήποτε λειτουργικό σύστημα, εφόσον ο διερμηνέας του είναι εγκατεστημένος εκεί.

Πολύ δημοφιλής στον τομέα της τεχνητής νοημοσύνης , η Python είναι μια ελαφριά γλώσσα με πολύ γρήγορη καμπύλη εκμάθησης και απλή σύνταξη.

Όλα αυτά, προστιθέμενα στο γεγονός ότι υπάρχει μεγάλος αριθμός βιβλιοθηκών και πλαισίων που βασίζονται στην Python, καθιστούν αυτήν τη γλώσσα προγραμματισμού μία από τις πιο αγαπημένες από τους επαγγελματίες ανάπτυξης σήμερα.

PHP

Πού είναι το κείμενο για την PHP

Ποια είναι τα πλεονεκτήματα και ποια τα μειονεκτήματα της γλώσσας υψηλού επιπέδου;

Οι γλώσσες υψηλού επιπέδου έχουν το προφανές πλεονέκτημα ότι είναι πιο κατανοητές από τις γλώσσες χαμηλού επιπέδου.

Επιπλέον, διαθέτουν επίσης περισσότερους πόρους και λειτουργίες που στοχεύουν στη δημιουργία ενός τεράστιου φάσματος διαδικτυακών εφαρμογών, όπως λογισμικό, ιστότοποι, εφαρμογές, λειτουργικά συστήματα, βάσεις δεδομένων κ.λπ.

Φυσικά, άλλες γλώσσες, όπως οι γλώσσες μηχανής και οι γλώσσες συναρμολόγησης, όπως η περίφημη Συνέλευση, για παράδειγμα, έχουν πολύ σημαντικές λειτουργίες στην τεχνολογία.

Χωρίς το πρώτο, δεν θα ήταν δυνατή η επικοινωνία με έναν επεξεργαστή. και χωρίς το δεύτερο, δεν θα ήταν δυνατή η επικοινωνία με τους ελεγκτές με κατανοητό τρόπο.

Αλλά μόνο μέσω γλωσσών υψηλού επιπέδου μπορούμε να συνδεθούμε με το διαδίκτυο και την τεχνολογία όπως τη γνωρίζουμε σήμερα.

Τι είναι η γλώσσα χαμηλού επιπέδου;

Γλώσσες χαμηλού επιπέδου είναι εκείνες των οποίων η σύνταξη είναι πιο μακριά από αυτό που γνωρίζουμε ως γλώσσα.

Αυτές οι γλώσσες ταξινομούνται σε δύο τύπους: γλώσσες μηχανής και γλώσσες συναρμολόγησης, όπως αναφέρθηκε στο προηγούμενο θέμα.

Το πρώτο βασίζεται μόνο σε δυαδικούς αριθμούς και μεταγραφές όπως το 010101000110010101, για παράδειγμα.

Οι γλώσσες συναρμολόγησης, γνωστές και ως γλώσσες ενδιάμεσου επιπέδου, είναι ένα υβρίδιο μεταξύ δυαδικών γλωσσών και γλωσσών υψηλού επιπέδου, επιτρέποντας στους ανθρώπους να επικοινωνούν με μηχανές μέσω μιας συγκεκριμένης κωδικοποίησης που μεταφράζεται από το ένα άκρο στο άλλο.

Ποιες είναι οι κύριες γλώσσες χαμηλού επιπέδου;

Η πιο δημοφιλής γλώσσα χαμηλού επιπέδου σήμερα είναι η Assembly.

Υπάρχει από την αρχή αυτού που γνωρίζουμε ως υπολογιστές και αναπτύχθηκε τη δεκαετία του 1950 με στόχο να κάνει τη δυαδική γλώσσα αναγνώσιμη στον άνθρωπο.

Ήταν ουσιαστικά η πρώτη δομημένη γλώσσα στην ιστορία του προγραμματισμού, δηλαδή μια γλώσσα της οποίας η λογική βασίζεται σε ακολουθίες και επαναλήψεις.

Υψηλό ή χαμηλό επίπεδο, ποιο είναι καλύτερο;

Δεν είναι δυνατός ο διαχωρισμός αυτών των γλωσσών σε καλύτερες ή χειρότερες επειδή, όπως εξηγήθηκε μερικές γραμμές παραπάνω, η καθεμία από αυτές εξυπηρετεί διαφορετικό σκοπό.

Χωρίς γλώσσες χαμηλού επιπέδου, γλώσσες υψηλού επιπέδου πιθανότατα δεν θα υπήρχαν, γιατί η επικοινωνία με επεξεργαστές και μικροεπεξεργαστές δεν θα ήταν δυνατή.

Ωστόσο, χωρίς γλώσσες υψηλού επιπέδου, δεν θα ήταν δυνατός ο πραγματικός προγραμματισμός, με την έννοια που το γνωρίζουμε σήμερα και όπως το γνωρίζουμε σήμερα.

Όταν κάθε γλώσσα εξυπηρετεί τον δικό της σκοπό, και οι δύο πάνε χέρι-χέρι.

Γλώσσες υψηλού επιπέδου (3η γενιά)

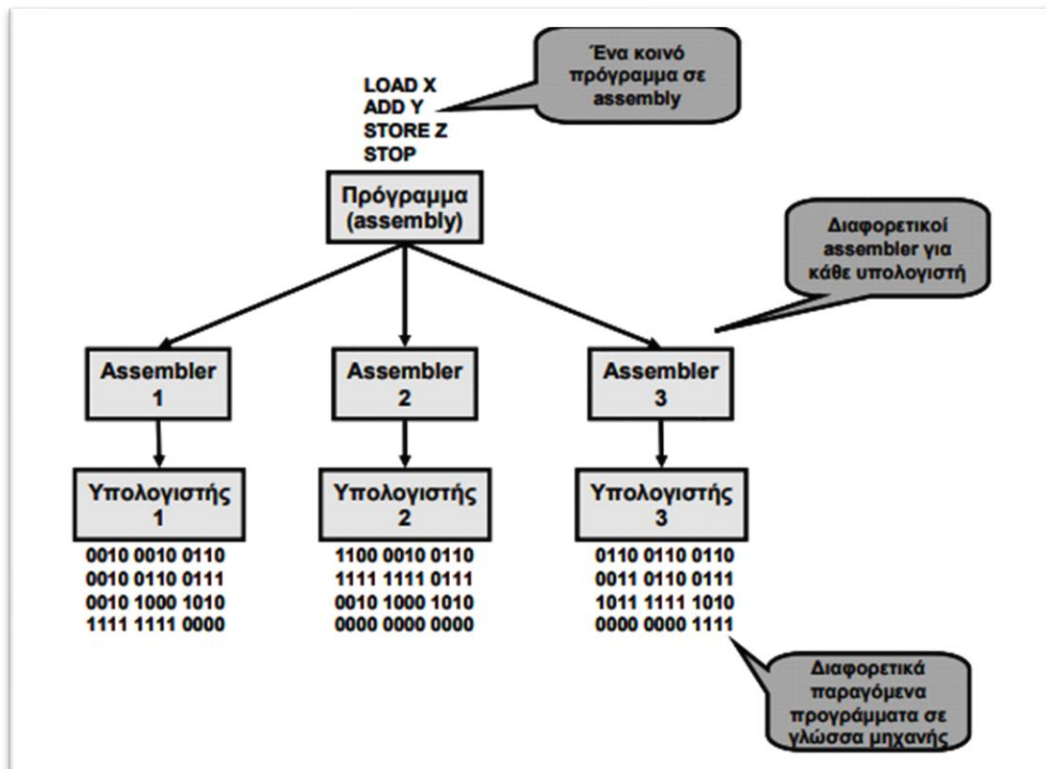
Η συμβολική γλώσσα βελτίωσε σημαντικά το έργο του προγραμματισμού, όμως



συνέχιζε να έχει ένα πολύ σημαντικό μειονέκτημα.

Επειδή ο προγραμματισμός γίνονταν ακόμα σε επίπεδο εντολών της γλώσσας μηχανής, έστω κι αν αυτές γράφονταν συμβολικά, τα προγράμματα ήταν μεγάλα σε μέγεθος και δυσνόητα καθώς έπρεπε να γραφτούν πολλές εντολές ακόμα και για απλές διαδικασίες.

Για παράδειγμα, η απλή πρόσθεση δυο αριθμών απαιτούσε τρεις εντολές (LOAD X – ADD Y – STORE Z), ενώ θα μπορούσε να γραφτεί πολύ πιο απλά, για παράδειγμα ως $Z = X + Y$ (δηλαδή, θέσε στη θέση μνήμης Z το άθροισμα των περιεχομένων των θέσεων X και Y).



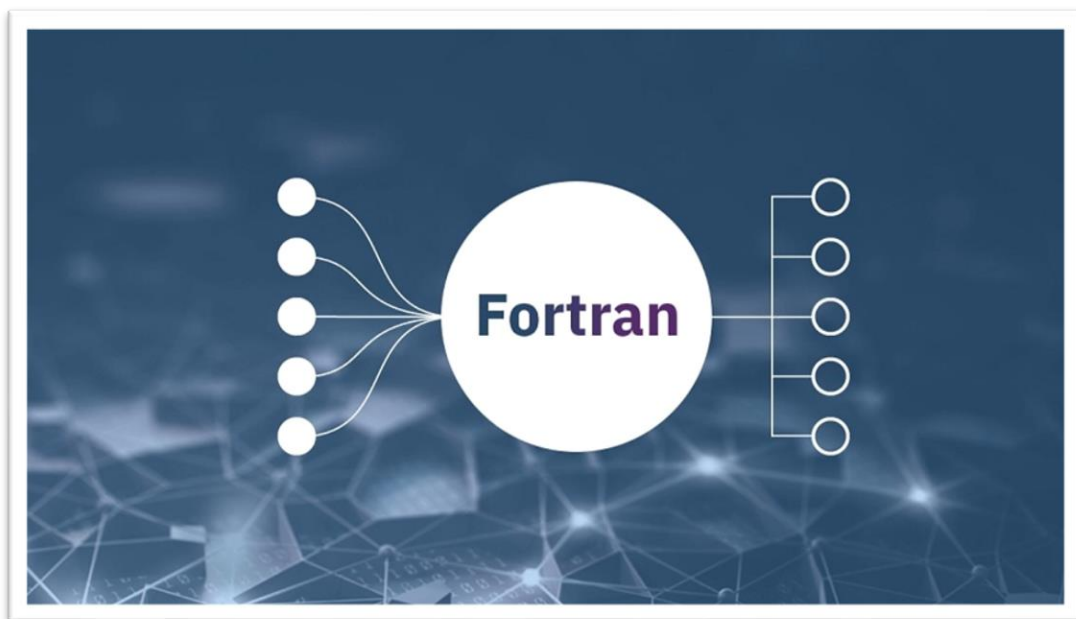
Έτσι, άρχισαν προσπάθειες δημιουργίας αφαιρετικών γλωσσών προγραμματισμού, δηλαδή γλωσσών που θα έκαναν το έργο συγγραφής προγραμμάτων ακόμα πιο εύκολο για τον προγραμματιστή, παρέχοντας εντολές που θα αντιστοιχούσαν σε ολόκληρες ομάδες εντολών της γλώσσας μηχανής (αντί της μια-προς-μια αντιστοίχισης που υποστηρίζει η assembly). Επειδή οι γλώσσες αυτές είναι πιο φιλικές στον προγραμματιστή, ονομάζονται γλώσσες υψηλού επιπέδου, σε αντίθεση με τη γλώσσα μηχανής και την assembly, που είναι πιο φιλικές στη μηχανή και ονομάζονται γλώσσες χαμηλού επιπέδου.

Στα πλεονεκτήματα των γλωσσών προγραμματισμού υψηλού επιπέδου σε σχέση με τις συμβολικές γλώσσες μπορούν να αναφερθούν:

- Ο φυσικότερος τρόπος προγραμματισμού με χρήση ευκολομνημόνευτων εντολών.
- Η ευκολία εκμάθησης.
- Η ευκολία διόρθωσης λαθών (debugging) και συντήρησης των προγραμμάτων.

- Η ανεξαρτησία από τον τύπο του υπολογιστή (μεταφερσιμότητα των προγραμμάτων).

Συνολικά οι γλώσσες υψηλού επιπέδου ελάττωσαν σημαντικά το χρόνο και το κόστος παραγωγής νέων προγραμμάτων, αφού λιγότεροι προγραμματιστές μπορούν σε μικρότερο χρόνο να αναπτύξουν προγράμματα που χρησιμοποιούνται σε περισσότερους υπολογιστές.



Το 1957 η IBM ανέπτυξε την πρώτη γλώσσα υψηλού επιπέδου, η οποία ονομάστηκε *FORTTRAN (από τις λέξεις formula translation)*.

Η FORTTRAN δημιουργήθηκε για την επίλυση μαθηματικών και επιστημονικών προβλημάτων και παρέχει ιδιαίτερα ισχυρές δομές αριθμητικής υψηλής ακρίβειας και χειρισμού πολύπλοκων μαθηματικών συναρτήσεων. Νεότερες εκδόσεις της χρησιμοποιούνται ακόμα και σήμερα σε επιστημονικές εφαρμογές.

Το 1960 παρουσιάστηκε η γλώσσα COBOL (common business oriented language), η οποία, όπως υπονοεί και το όνομα της, δημιουργήθηκε για ανάπτυξη εμπορικών εφαρμογών. Οι εφαρμογές αυτές έχουν τελείως διαφορετικές απαιτήσεις από τις επιστημονικές, όπως χειρισμό αρχείων και βάσεων δεδομένων, παραγωγή αναφορών (reports) και άλλα.

Η γλώσσα παρείχε σημαντικές διευκολύνσεις για το χειρισμό τέτοιων δεδομένων και για το λόγο αυτό έγινε πολύ δημοφιλής τις δεκαετίες του 1970 και 1980.

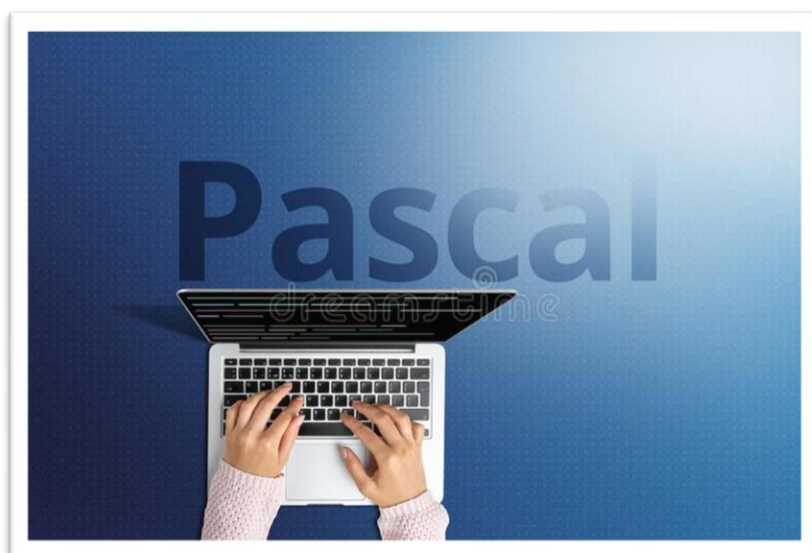
Σήμερα δε χρησιμοποιείται πια.



Μια από τις σημαντικότερες γλώσσες προγραμματισμού, η οποία δε γνώρισε ποτέ εμπορική επιτυχία, αλλά ο σχεδιασμός της επηρέασε ιδιαίτερα την ανάπτυξη μελλοντικών γλωσσών προγραμματισμού, είναι η ALGOL (algorithmic language) που πρωτοπαρουσιάστηκε το 1962 και σήμερα δε χρησιμοποιείται πια.

Μια από τις πιο χαρακτηριστικές γλώσσες προγραμματισμού, που χρησιμοποιείται ακόμα και σήμερα, είναι η **BASIC (beginner's all-purpose symbolic instruction code)**, η οποία παρουσιάστηκε το 1964 με σκοπό να αποτελέσει εργαλείο εκπαίδευσης προγραμματιστών.

Η γλώσσα Pascal παρουσιάστηκε το 1971 και ο σχεδιασμός της στηρίχθηκε στην ALGOL. Η γλώσσα είναι κατάλληλη τόσο για εκπαίδευση όσο και για τη δημιουργία ισχυρών προγραμμάτων με κύριο χαρακτηριστικό της τη δυνατότητα δημιουργίας



ιδιαίτερα δομημένων τμημάτων κώδικα. Χρησιμοποιήθηκε ευρέως ως πρώτη γλώσσα εκμάθησης προγραμματισμού για πολλά χρόνια, μέχρι που ξεπεράστηκε από γλώσσες όπως η C και η Java για το σκοπό αυτό.

Μία ακόμη γλώσσα με μεγάλη διάδοση ακόμα και σήμερα, παρόλο που παρουσιάστηκε για πρώτη φορά το 1978, είναι η γλώσσα C. Η C είναι μια γλώσσα προγραμματισμού γενικής χρήσης, που δημιουργήθηκε αρχικά για συστήματα Unix. Χρησιμοποιείται συνήθως σε συστήματα cross-platform, για κωδικοποίηση συστημάτων Unix, παιχνιδιών και ειδικών εφαρμογών (automotive, ενσωματωμένα συστήματα, ρομποτική κτλ.). Επιλέγεται πιο συχνά, επειδή είναι πιο συμπαγής από την C++ και τρέχει πιο γρήγορα. Είναι, η δεύτερη πιο συνηθισμένη γλώσσα προγραμματισμού ακολουθώντας την Java. Η C αποτελεί τη βάση για να δημιουργηθούν πολλές άλλες γλώσσες προγραμματισμού συμπεριλαμβανομένων των C#, Java, JavaScript, Perl, PHP και Python. Είναι γλώσσα με ιδιαίτερα ισχυρά χαρακτηριστικά, παρέχοντας ταυτόχρονα τα πλεονεκτήματα του δομημένου προγραμματισμού που είχε και η Pascal, αλλά και πολλές δυνατότητες εντολών χαμηλού επιπέδου, οι οποίες επιτρέπουν άμεση πρόσβαση στο υλικό του υπολογιστή. Η C αργότερα εξελίχτηκε σε διάφορες παραλλαγές, όπως η C++ και η C#, που είναι αντικειμενοστραφείς (objectoriented) γλώσσες (θα εξηγήσουμε τι σημαίνει αυτό στη συνέχεια του κεφαλαίου).

Η **Objective-C** Όπως και οι περισσότερες από αυτές τις γλώσσες προγραμματισμού που αναφέρονται σε αυτό το άρθρο, προέρχεται από την C. Είναι κώδικας γενικής χρήσης, υψηλού επιπέδου που έχει μια πρόσθετη λειτουργία ανταλλαγής μηνυμάτων. Είναι γνωστή, αφού είναι η γλώσσα προγραμματισμού που επιλέγεται να χρησιμοποιείται για τις εφαρμογές OS X και iOS της Apple, μέχρι που αντικαταστάθηκε από την Swift.

Η **JavaScript** δημιουργήθηκε ως ένας κώδικας με πρόσθετα για να επεκτείνει τη λειτουργικότητα των ιστοσελίδων. Προσθέτει δυναμικά χαρακτηριστικά όπως φόρμες υποβολής, διαδραστικότητα, κινούμενα σχέδια, παρακολούθηση χρηστών κλπ. Χρησιμοποιείται κυρίως για front-end development ή για λύσεις προγραμματισμού με τις οποίες αλληλεπιδρούν οι πελάτες. Είναι συμβατή με όλους τους browsers, καθιστώντας την έναν καλό κώδικα γενικής χρήσης για web development, παρόλο που είναι επίσης γνωστή για τη δυσκολία της να κάνει debugging.

Η **Swift** αντιμετωπίστηκε ως η αντικατάσταση της Objective-C όσον αφορά τα προγράμματα της Apple. Έχει γίνει πιο δημοφιλής τα τελευταία χρόνια, καθώς είναι εύκολη στο να διαβαστεί, εύκολη στη συντήρηση και πιο γρήγορη από την Objective-C. Εάν θέλετε να γίνετε προγραμματιστής της Apple ή να γράψετε προγράμματα/εφαρμογές για iOS, αυτή είναι η γλώσσα που χρειάζεται να μάθετε. Παρόλο που η Objective-C εξακολουθεί να χρησιμοποιείται, η Swift γίνεται με γρήγορους ρυθμούς η γλώσσα προγραμματισμού που επιλέγεται να χρησιμοποιείται από τους προγραμματιστές που δημιουργούν προγράμματα για συσκευές Apple.

Η **C#** (προφέρεται C Sharp) δημιουργήθηκε από τη Microsoft το 2001. Παρόλο που ονομαστικά αναφέρεται στην 'οικογένεια' C των γλωσσών προγραμματισμού, έχει περισσότερα κοινά με την Java παρά με άλλες γλώσσες C. Η C# χρησιμοποιείται κυρίως για εσωτερικές/επιχειρηματικές λύσεις και είναι αρκετά δημοφιλής, αποτελώντας το 'αντίπαλο δέος' της Java.

Η Ruby είναι μια αντικειμενοστρεφής, γενικής χρήσης back-end γλώσσα προγραμματισμού που αναπτύχθηκε στα μέσα της δεκαετίας του '90 στην Ιαπωνία. Είναι μια από τις απλούστερες γλώσσες προγραμματισμού για να μάθει κάποιος και χρησιμοποιείται συχνά ως 'stepping stone' για το Ruby on Rails. Αναπτύχθηκε τόσο για να είναι ευχάριστη στο να κωδικοποιηθεί όσο και στο να αυξήσει την παραγωγικότητα. Είναι γνωστή, καθώς είναι εύκολο να διαβαστεί και ως αποτέλεσμα πολλοί προγραμματιστές συνιστούν την εκμάθηση της Ruby ως την πρώτη γλώσσα προγραμματισμού. Χρησιμοποιείται από sites όπως το Hulu, το Shopify, το Airbnb και πολλά άλλα.



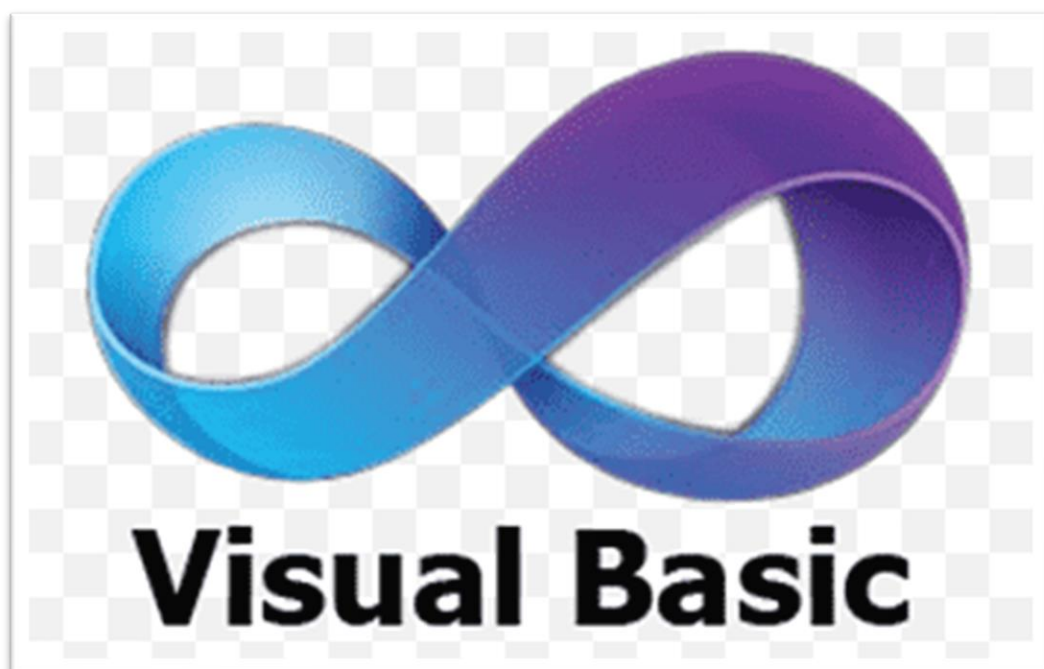
Τα τελευταία χρόνια χρησιμοποιείται ιδιαίτερα η Java. Η Java, σήμερα, είναι η πιο δημοφιλής και ευρέως χρησιμοποιούμενη γλώσσα στον κόσμο. Παρόλο που δημιουργήθηκε αρχικά για τη διαδραστική τηλεόραση, έγινε γνωστή ως η γλώσσα που επιλέγεται να χρησιμοποιείται στις συσκευές Android και γενικά στις web εφαρμογές.

Είναι, επίσης, η γλώσσα προγραμματισμού που επιλέγεται για το λογισμικό πολλών επιχειρήσεων. Είναι μια καλή γλώσσα προγραμματισμού πολλαπλών χρήσεων, επειδή μπορεί να χρησιμοποιηθεί σε cross-platform (το οποίο σημαίνει ότι είναι εξίσου εύκολο να χρησιμοποιηθεί τόσο στις εφαρμογές των smartphone όσο και στις εφαρμογές desktop). Μοιάζει με τη C++ στη σύνταξη και στη δομή, καθιστώντας, έτσι, εύκολη την εκμάθησή της, αν γνωρίζετε ήδη τις γλώσσες της 'οικογένειας' C.

Η Java είναι και αυτή μια αντικειμενοστραφής γλώσσα που αρχικά αναπτύχθηκε από την εταιρία Sun Microsystems με σκοπό την ανάπτυξη εφαρμογών που θα εκτελούνται σε κατανομημένα περιβάλλοντα, για παράδειγμα σε υπολογιστές που είναι συνδεδεμένοι με το Διαδίκτυο.

Έχει δανειστεί πολλά χαρακτηριστικά της C και της C++ και ένα από τα βασικά της χαρακτηριστικά είναι ότι επιτρέπει, εκτός από τη συγγραφή ανεξάρτητων εφαρμογών (applications), και τη συγγραφή μικροεφαρμογών (applets) που είναι ενσωματωμένες σε κώδικα HTML και εκτελούνται μέσα από οποιοδήποτε web browser έχει δυνατότητες Java.

Τέλος, η Visual Basic είναι άλλη μια αντικειμενοστραφής (object-oriented) γλώσσα που επιτρέπει τη γρήγορη ανάπτυξη εφαρμογών με γραφικό περιβάλλον διεπαφής με το χρήστη. Οι χρήστες μπορούν να δημιουργήσουν απλά προγράμματα διαμορφώνοντας τμήματα (components) σε προσχεδιασμένες φόρμες, απλά ορίζοντας τη λειτουργικότητα των τμημάτων αυτών.



Με άλλα λόγια, ένας προγραμματιστής μπορεί να δημιουργήσει ένα πρόγραμμα χωρίς να χρειαστεί να γράψει καθόλου κώδικα (αν και έχει φυσικά αυτή τη δυνατότητα εφόσον επιθυμεί).

Γλώσσες πολύ υψηλού επιπέδου (4η γενιά)

Η προσπάθεια να αναπτυχθούν γλώσσες προγραμματισμού ολοένα και φιλικότερες προς τον προγραμματιστή έχει οδηγήσει τα τελευταία χρόνια στην ανάπτυξη γλωσσών στις οποίες ο προγραμματιστής δε χρειάζεται σε πολλές περιπτώσεις να καθορίσει πως θα εκτελεστεί μια δραστηριότητα, αλλά πρέπει απλώς να ορίσει τι θέλει να γίνει.

Για παράδειγμα, δε χρειάζεται να γραφτεί ένα πρόγραμμα που να δίνει στον υπολογιστή εντολές για το πως θα ταξινομηθούν τα περιεχόμενα ενός πίνακα, καθώς αρκεί να δοθεί μια εντολή να γίνει η ταξινόμηση και ο υπολογιστής αναλαμβάνει να την εκτελέσει.

Για το λόγο αυτό, οι γλώσσες αυτές ονομάζονται γλώσσες πολύ υψηλού επιπέδου ή γλώσσες τέταρτης γενιάς, καθώς είναι ακόμα φιλικότερες προς τον προγραμματιστή. Μάλιστα, είναι τόσο φιλικές που μπορούν να χρησιμοποιηθούν όχι μόνο από προγραμματιστές αλλά και από απλούς χρήστες.

Παράδειγμα τέτοιας γλώσσας αποτελεί η SQL.

Διαβάζεται "sequel" SQL σημαίνει Structured Query Language είναι μια γλώσσα προγραμματισμού ειδικού σκοπού που χρησιμοποιείται για τη λήψη πληροφοριών και την ενημέρωση βάσεων δεδομένων. Λαμβάνοντας υπ' όψιν ότι σχεδόν όλες οι επιχειρήσεις διαθέτουν κάποιο σύστημα βάσης δεδομένων, η SQL είναι ένα καλό στοιχείο που μπορείτε να εντάξετε στο βιογραφικό σας. Είναι επίσης μία από τις ευκολότερες δεξιότητες Πληροφορικής που μπορείς να μάθεις, με πολλά μαθήματα και δωρεάν πηγές στο διαδίκτυο.

Εάν εξοικειωθείς με τη συγκεκριμένη γλώσσα, θα μπορούσες να γίνεις ιδανικός/η υποψήφιος/α για θέσεις εργασίας, όπως Database Developer και Database Engineer

- αλλά και πολλές γενικές εργασίες αναλυτών δεδομένων (Data Engineers / Data Analysts) θα ζητήσουν επίσης να γνωρίζεις την SQL, καθώς χρησιμοποιείται ευρέως σε εφαρμογές βάσεων δεδομένων και web frameworks.

Η SQL (structured query language) είναι μια γλώσσα υποβολής ερωτημάτων σε βάσεις δεδομένων. Η γλώσσα έχει σχεδιαστεί για να υποστηρίζει τη δημιουργία και διαχείριση σχεσιακών βάσεων δεδομένων (relational databases) και παρέχει εργαλεία για την ανάκτηση και παρουσίαση υποσυνόλων από τα δεδομένα που περιέχονται σε τέτοιες βάσεις. Υποστηρίζεται από όλα σχεδόν τα συστήματα διαχείρισης βάσεων δεδομένων.



Για τους λόγους που αναφέραμε στην εισαγωγή, ο προγραμματισμός σε γλώσσα μηχανής δεν εξασκείται πια παρά μόνο σε πολύ εξειδικευμένες περιπτώσεις (για παράδειγμα, απευθείας προγραμματισμός ολοκληρωμένων κυκλωμάτων). Αντίθετα, αναπτύχθηκαν και εξελίχθηκαν διάφορα είδη γλωσσών προγραμματισμού που θα δούμε στη συνέχεια.

Σε αυτή ακριβώς τη δυσκολία οφείλεται, κατά κύριο λόγο, η ύπαρξη των γλωσσών προγραμματισμού *υψηλού επιπέδου* όπως οι Fortran (χρονικά η πρώτη τέτοια γλώσσα), BASIC, C, Pascal, Lisp και δεκάδες άλλες.

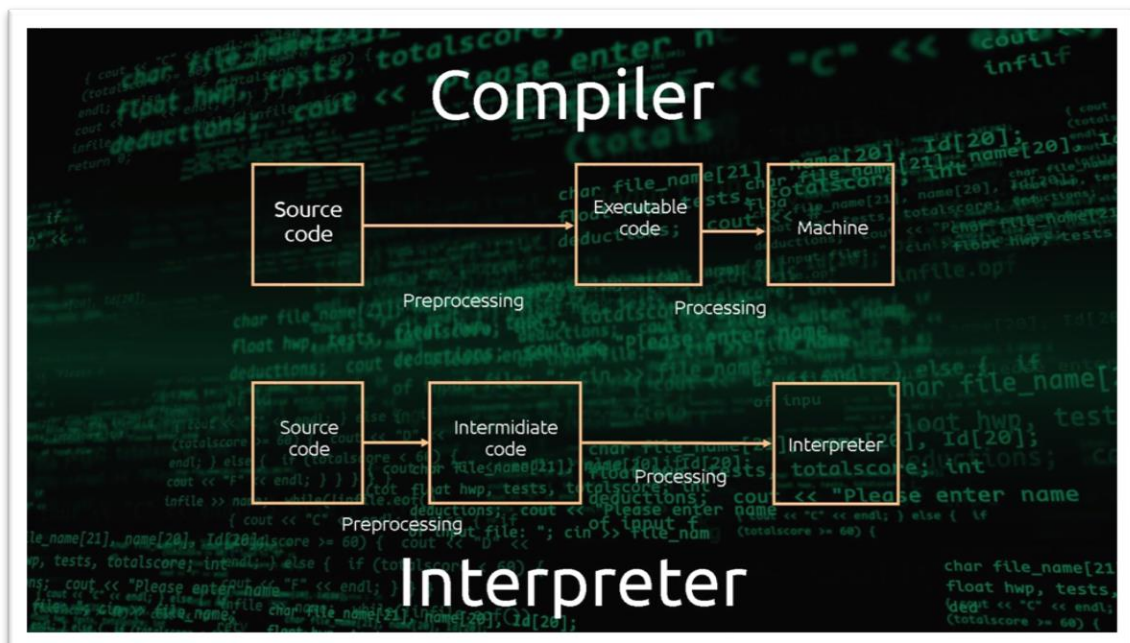


Όπως θα δούμε παρακάτω οι γλώσσες αυτές παρέχουν σημαντικές ευκολίες στον προγραμματιστή, οι οποίες δεν υφίστανται στη γλώσσα μηχανής, ή έστω στη γλώσσα *assembly* που αποτελεί ένα ενδιάμεσο σημείο ανάμεσα στη γλώσσα μηχανής και στις γλώσσες υψηλού επιπέδου (πολύ πλησιέστερα όμως στη γλώσσα μηχανής απ' ότι στις τελευταίες).

Επειδή όμως κάθε υπολογιστής μια γλώσσα μόνο "καταλαβαίνει" κάθε πρόγραμμα που έχει γραφεί σε μια γλώσσα προγραμματισμού (C, BASIC, Fortran κλπ, ή ακόμη και σε γλώσσα *assembly*) πρέπει πρώτα να μεταγλωττιστεί σε γλώσσα μηχανής προτού μπορέσει να εκτελεστεί. Η διαδικασία αυτή της μεταγλώττισης συναντάται σε τρεις, λίγο-πολύ, διαφορετικές μορφές:

- Προγράμματα που έχουν γραφεί σε γλώσσα *assembly* "περνάνε" από το λεγόμενο *assembler*, που δεν είναι τίποτε άλλο από ένα εκτελέσιμο (δηλ. ήδη μεταφρασμένο σε γλώσσα μηχανής) πρόγραμμα που διαβάζει το πρόγραμμα *assembly* και παράγει ένα λειτουργικά ισοδύναμο εκτελέσιμο πρόγραμμα.
- Προγράμματα που έχουν γραφεί σε μια γλώσσα υψηλού επιπέδου περνάνε από το λεγόμενο *compiler* (γίνονται *compiled*, όπως λέμε).

- Υπάρχει και μια τρίτη μορφή μεταγλώττισης ενός προγράμματος, αυτή στην οποία δεν δημιουργείται κάποιο εκτελέσιμο πρόγραμμα το οποίο στη συνέχεια εκτελείται, αλλά ο κύκλος μεταγλώττιση-εκτέλεση είναι πολύ πιο βραχύς, και συνήθως εκτελείται μια ``εντολή" της γλώσσας υψηλού επιπέδου αμέσως μόλις μεταγλωττιστεί και προτού μεταγλωττιστεί η ``επόμενη" εντολή. Προγράμματα που κάνουν αυτού του είδους τη μεταγλώττιση-εκτέλεση λέγονται *interpreters* και χαρακτηριστικό τους είναι ότι συνήθως δε δημιουργούν κάποια μορφή εκτελέσιμου προγράμματος που μπορεί να εκτελεστεί χωρίς τη βοήθεια του interpreter και μόνο από τη μηχανή.



Το φάσμα βέβαια ανάμεσα στο compilation και στο interpretation είναι γεμάτο από ενδιάμεσες καταστάσεις, με πιο συχνή, τελευταία, τη μετατροπή του προγράμματος από τη γλώσσα υψηλού επιπέδου σε μια γλώσσα χαμηλού επιπέδου (που έχει επικρατήσει να λέγεται *bytecode*) η οποία όμως είναι ανεξάρτητη μηχανής. Γι' αυτό το πρόγραμμα σε μορφή *bytecode* μπορεί να εκτελεστεί σε οποιαδήποτε μηχανή, αρκεί βέβαια να υπάρχει σε αυτή τη μηχανή ένα πρόγραμμα που διαβάζει *bytecode* και το εκτελεί. Τέτοια προγράμματα είναι συνήθως πολύ απλά να γραφούν, γιατί

ακριβώς το bytecode είναι μια πολύ απλή γλώσσα που γράφεται και διαβάζεται εύκολα από μηχανή.

Το bytecode interpretation προσφέρει μια ενδιαμέση αποδοτικότητα (προγράμματα που έχουν γίνει fully compiled και έχουν μετατραπεί σε εκτελέσιμο κώδικα σε γλώσσα μηχανής είναι πολύ ταχύτερα από προγράμματα που διαβάζονται και εκτελούνται ταυτόχρονα από interpreters, είναι όμως καταδικασμένα να τρέχουν μόνο στον τύπο μηχανής (η έννοια "τύπος της μηχανής" εδώ περιλαμβάνει το είδος του hardware αλλά και το είδος και τις διάφορες παραμέτρους του λειτουργικού συστήματος) για τον οποίο έχουν μεταγλωττιστεί. Αντίθετα, ένα πρόγραμμα που τρέχει μέσα από interpreter τρέχει οπουδήποτε υπάρχει εγκατεστημένος ένας interpreter για τη γλώσσα στην οποία έχει το πρόγραμμα γραφεί.

Είναι όμως πολύ πιο αργό. Το bytecode προσφέρει μια μέση οδό ανάμεσα στο *portability* (μεταφερισιμότητα) και στο *efficiency*, που είναι μάλιστα ιδιαίτερα σημαντική στην εκτέλεση προγραμμάτων διά μέσου του Internet για τον απλούστατο λόγο ότι θέλει κανείς να κάνει κλικ σε ένα πρόγραμμα που βρίσκεται στο τάδε site και αυτό να τρέχει τοπικά, στη μηχανή του. Εκεί έχει σημασία τα προγράμματα να είναι αποθηκευμένα σε γλώσσα ανεξάρτητη μηχανής, μια και δεν μπορεί κανείς να υποθέσει κάτι για ένα remote site, αλλά είναι επίσης σημαντικό να είναι αποθηκευμένα σε μια μορφή που να γίνεται interpreted αποτελεσματικά. Το bytecode προσφέρει μια καλή μέση οδό.

Ταξινόμηση γλωσσών προγραμματισμού

Στην προηγούμενη ενότητα είδαμε μια χρονολογική ταξινόμηση των γλωσσών προγραμματισμού σε γλώσσες πρώτης, δεύτερης, τρίτης και τέταρτης γενιάς. Όμως οι γλώσσες προγραμματισμού μπορούν να ταξινομηθούν και με πολλά άλλα κριτήρια.

Στη συνέχεια θα δούμε συνοπτικά κάποιες τέτοιες ταξινομήσεις.

Η βασικότερη κατηγοριοποίηση των γλωσσών προγραμματισμού μπορεί να γίνει με βάση τη φιλοσοφία προγραμματισμού που πρεσβεύουν. Έχουν αναπτυχθεί διάφορες τέτοιες φιλοσοφίες, κάθε μια εκ των οποίων μπορεί να ταιριάζει καλύτερα σε

διαφορετικά είδη προβλημάτων ή/και στον τρόπο που βολεύει την προσέγγιση σε ότι αφορά τη λύση προβλημάτων διαφορετικών προγραμματιστών.

Ας δούμε τις κατηγορίες των γλωσσών προγραμματισμού ανάλογα με αυτό το κριτήριο.

Η πρώτη κατηγορία είναι οι **διαδικασιακές (procedural) γλώσσες**.

Οι γλώσσες αυτές εκτελούν ακολουθίες βημάτων για την επίλυση κάποιου προβλήματος. Είναι γνωστές και ως προστακτικές γλώσσες γιατί κάθε εντολή τους αποτελεί και μια προσταγή προς τον υπολογιστή να εκτελέσει κάποια συγκεκριμένη ενέργεια. Ένα πρόγραμμα γραμμένο σε μια διαδικασιακή γλώσσα αποτελείται από διαδικασίες, οι οποίες επικοινωνούν μεταξύ τους και κάθε μια εκτελεί μια συγκεκριμένη δραστηριότητα.

Ο διαδικασιακός τρόπος προγραμματισμού αποτελεί τον παραδοσιακό τρόπο με τον οποίο γράφονταν προγράμματα στις πρώτες δημοφιλής γλώσσες προγραμματισμού και χρησιμοποιείται κυρίως ακόμα και σήμερα. Χαρακτηριστικά παραδείγματα διαδικασιακών γλωσσών προγραμματισμού είναι οι Fortran, Cobol, Algol, Pascal και C.

Τα τελευταία χρόνια έχει γίνει ιδιαίτερα δημοφιλής η κατηγορία γλωσσών προγραμματισμού, οι οποίες ονομάζονται **αντικειμενοστραφείς γλώσσες (object-oriented languages)**.

Ο αντικειμενοστραφής προγραμματισμός αποτελεί μια φιλοσοφία διαφορετική από αυτή του διαδικασιακού, καθώς στηρίζεται στα δεδομένα και όχι στις διαδικασίες.

Στις αντικειμενοστραφείς γλώσσες τα δεδομένα ονομάζονται αντικείμενα (objects) και περιέχουν ενσωματωμένες στον ορισμό τους τις διαδικασίες που μπορούν να εφαρμοστούν σε κάθε αντικείμενο. Τα αντικείμενα επικοινωνούν μεταξύ τους και μπορούν να κληροδοτήσουν τις ιδιότητές τους σε άλλα αντικείμενα. Χαρακτηριστικά παραδείγματα αντικειμενοστραφών γλωσσών είναι η C++, η Visual Basic και η Java.

Μια άλλη ακόμα κατηγορία γλωσσών που έχουν γίνει πολύ δημοφιλείς τα τελευταία χρόνια με την ανάπτυξη του Internet είναι οι **γλώσσες σήμανσης ή παρουσίασης**.

Οι γλώσσες αυτές δεν είναι ακριβώς γλώσσες προγραμματισμού, με την έννοια ότι αποσκοπούν στον καθορισμό του τρόπου δόμησης ή εμφάνισης δεδομένων, παρά στην εκτέλεσή μιας διαδικασίας.

Ενδεικτικά παραδείγματα γλωσσών σήμανσης αποτελούν οι γλώσσες XML (γλώσσα σήμανσης δεδομένων) και HTML (γλώσσα μορφοποίησης ιστοσελίδων).



Τα υπόλοιπα είδη γλωσσών προγραμματισμού έχουν πιο συγκεκριμένες χρήσεις και για το λόγο αυτό είναι λιγότερο δημοφιλή.

Σε αυτά ανήκουν:

⇐ Οι συναρτησιακές γλώσσες (*functional languages*), όπου ένα πρόγραμμα έχει ουσιαστικά τη μορφή μιας μαθηματικής συνάρτησης. Χρησιμοποιούνται σε μαθηματικές εφαρμογές και χαρακτηριστικότερο παράδειγμα τους είναι η γλώσσα LISP.

⇐ Οι λογικές γλώσσες (*logic languages*), που είναι γνωστές και ως δηλωτικές (*declarative*), χρησιμοποιούν λογικούς συλλογισμούς για τη δημιουργία προγραμμάτων. Χρησιμοποιούνται σε εφαρμογές τεχνητής νοημοσύνης και έχουν ως χαρακτηριστικό εκπρόσωπο τους τη γλώσσα PROLOG.

⇐ Γλώσσες ερωταποκρίσεων (*query languages*), οι οποίες αποτελούνται από ερωτήματα προς τον υπολογιστή που καθορίζουν το επιθυμητό αποτέλεσμα αλλά όχι και το πως αυτό θα παραχθεί (το τελευταίο αποτελεί απόφαση του υπολογιστή). Χαρακτηριστικό παράδειγμα αποτελεί η γλώσσα SQL.

Μια άλλη ταξινόμηση των γλωσσών προγραμματισμού μπορεί να γίνει με βάση το είδος των προβλημάτων τα οποία έχει σχεδιαστεί να λύνει κάθε γλώσσα.

Με αυτό το κριτήριο διακρίνουμε τις γλώσσες σε γλώσσες γενικής χρήσης, οι οποίες θεωρητικά μπορούν να χρησιμοποιηθούν για την επίλυση οποιουδήποτε προβλήματος και περιλαμβάνουν γλώσσες όπως η Basic, η Pascal, η C και η Java.

Από την άλλη μεριά, υπάρχουν οι γλώσσες ειδικής χρήσης που έχουν σχεδιαστεί ειδικά για συγκεκριμένα είδη προβλημάτων.

Τέτοιες γλώσσες είναι η FORTRAN (για μαθηματικές εφαρμογές), η COBOL (για εμπορικές εφαρμογές), η HTML (για μορφοποίηση ιστοσελίδων), η SQL (για ερωτήματα σε βάσεις δεδομένων), η MODSIM (γλώσσα προσομοίωσης) και άλλες.

Εκπαιδευτική Υποενότητα: 3.2 Αλγόριθμοι και δομές δεδομένων

Δομές δεδομένων: εισαγωγή

Οι δομές δεδομένων, μαζί με τη μελέτη των αλγορίθμων, αποτελούν μέρος των θεμελιωδών αρχών του προγραμματισμού και πολλά διαβάζονται/ακούγονται για τη σημασία της μελέτης αυτών των θεμάτων. Σε αυτό το άρθρο, θα καλύψουμε τις δομές δεδομένων από την αρχή: ποιες είναι, μερικά παραδείγματα και γιατί είναι σημαντικές.

Τι είναι τα δεδομένα; Τα δεδομένα (και οι διάφοροι τύποι τους) είναι τα δομικά στοιχεία του προγραμματισμού. Αντιπροσωπεύουν μια μονάδα ή ένα στοιχείο

πληροφοριών που μπορεί να προσπελαστεί μέσω ενός αναγνωριστικού - για παράδειγμα, μιας μεταβλητής.

Οι περισσότερες γλώσσες προγραμματισμού λειτουργούν με παραλλαγές που βασίζονται στους τέσσερις αρχέγονους τύπους παρακάτω:

INT ή ακέραιος: ακέραιες αριθμητικές τιμές (θετικές ή αρνητικές).

FLOAT ή το λεγόμενο "floating point": αριθμητικές τιμές με θέσεις μετά το κόμμα (θετικό ή αρνητικό).

BOOLEAN ή booleans: αντιπροσωπεύεται από δύο μόνο τιμές, "true" και "false". Ονομάζονται επίσης λογικοί τελεστές.

KEIMENO: ακολουθίες ή συμβολοσειρές χαρακτήρων, που χρησιμοποιούνται για τον χειρισμό κειμένου ή/και άλλων μη αριθμητικών ή boolean τύπων δεδομένων, όπως κρυπτογραφικούς κατακερματισμούς.

Η JavaScript, για παράδειγμα, έχει πρωτόγονους τύπους ενσωματωμένους στη βασική δομή της γλώσσας: αριθμό, συμβολοσειρά, boolean και σύμβολο (από το "συμβολικό όνομα", που χρησιμοποιείται μεταξύ άλλων για τη δημιουργία μοναδικών ιδιοτήτων σε αντικείμενα). Το C# (C-Sharp) λειτουργεί με μεγαλύτερο αριθμό πρωτόγονων τύπων, ανάλογα με το χώρο μνήμης που θα καταλαμβάνει η μεταβλητή: Boolean, Byte, SByte, Int16, UInt16, Int32, UInt32, Int64, UInt64, IntPtr, UIntPtr, Char, Double και Single. Το C, από την άλλη πλευρά, δεν έχει δικό του τύπο δεδομένων Boolean. false αντιπροσωπεύεται από τον αριθμό 0 και οποιοδήποτε άλλο ψηφίο αντιπροσωπεύει true. Άλλες γλώσσες μπορεί να λειτουργούν με άλλες παραλλαγές.

Τι είναι οι δομές δεδομένων;

Στους υπολογιστές, συνήθως χρησιμοποιούμε δεδομένα μαζί. Ο τρόπος με τον οποίο θα συγκεντρωθούν και θα οργανωθούν αυτά τα δεδομένα εξαρτάται σε μεγάλο βαθμό από τον τρόπο χρήσης και επεξεργασίας τους, λαμβάνοντας υπόψη, για παράδειγμα, την αποτελεσματικότητα των αναζητήσεων, τον όγκο των δεδομένων που εργάστηκαν, την πολυπλοκότητα της υλοποίησης και τον τρόπο στην οποία σχετίζονται τα δεδομένα. Αυτές οι διάφορες μορφές οργάνωσης ονομάζονται δομές δεδομένων.

Μπορούμε να πούμε ότι ένα πρόγραμμα αποτελείται από αλγόριθμους και δομές δεδομένων, που μαζί κάνουν το πρόγραμμα να λειτουργεί όπως θα έπρεπε.

Κάθε δομή δεδομένων έχει το δικό της σύνολο μεθόδων για την εκτέλεση λειτουργιών όπως:

Εισαγωγή ή διαγραφή στοιχείων.

Αναζήτηση και εντοπισμός στοιχείων.

Ταξινόμηση (ταξινόμηση) στοιχείων σύμφωνα με κάποια καθορισμένη σειρά.

Χαρακτηριστικά δομών δεδομένων

Οι δομές δεδομένων μπορεί να είναι:

γραμμικό (π.χ. πίνακες) ή μη γραμμικό (π.χ. γραφήματα).

ομοιογενή (όλα τα δεδομένα που απαρτίζουν τη δομή είναι του ίδιου τύπου) ή ετερογενή (μπορεί να περιέχουν δεδομένα διαφόρων τύπων).

στατική (έχουν σταθερό μέγεθος/χωρητικότητα μνήμης) ή δυναμική (μπορεί να επεκταθεί).

Θα δούμε παρακάτω μια λίστα και περιγραφή ορισμένων δομών:

Πίνακας

Ο πίνακας που ονομάζεται επίσης διάνυσμα , μήτρα ή πίνακας , είναι η πιο κοινή από τις δομές δεδομένων και είναι συνήθως η πρώτη που μελετάμε.

Με τον όρο παραγγελία , εννοούμε εδώ μια λίστα όπου η πρόσβαση στις τιμές γίνεται πάντα με την ίδια σειρά. Δηλαδή, εκτός εάν χρησιμοποιείται κάποια συνάρτηση ή μέθοδος για την αλλαγή της σειράς, το πρώτο στοιχείο του πίνακα `listaNumeros` είναι πάντα 4 και το τελευταίο, 0.

Κανονικά εργαζόμαστε μόνο με έναν τύπο δεδομένων ανά πίνακα. αν και η JavaScript επιτρέπει τη δήλωση πινάκων περισσότερων του ενός τύπων δεδομένων,

για παράδειγμα ["banana", 5, true], αυτό δεν συμβαίνει στις περισσότερες γλώσσες προγραμματισμού.

Καθώς είναι μια βασική και ευρέως χρησιμοποιούμενη δομή δεδομένων, οι γλώσσες προγραμματισμού συνήθως έχουν ήδη υλοποιήσει αυτόν τον τύπο, με εγγενείς μεθόδους δημιουργίας και χειρισμού πινάκων.

Χρήσεις

Όντας η πιο κοινή από τις δομές, οι πίνακες χρησιμοποιούνται σχεδόν σε κάθε περίπτωση που περιλαμβάνει την οργάνωση δεδομένων του ίδιου τύπου. είτε τα δεδομένα που λαμβάνονται από ένα API είτε αποστέλλονται σε μια βάση δεδομένων, είτε ακόμη περνούν μέσω παραμέτρου σε μια συνάρτηση ή μέθοδο. Οι πίνακες μπορούν επίσης να είναι πολυδιάστατοι, να χρησιμοποιούνται όποτε υπάρχει ανάγκη για πίνακα δεδομένων και οι δισδιάστατοι πίνακες (πίνακες) χρησιμοποιούνται για την επεξεργασία εικόνας.

Μπαταρία

Σε έναν πίνακα, είναι δυνατό να χρησιμοποιήσει τις δικές του συναρτήσεις για να χειριστεί στοιχεία σε οποιαδήποτε θέση στη λίστα. Ωστόσο, υπάρχουν περιπτώσεις (θα δούμε παραδείγματα αργότερα) όπου είναι επιθυμητός περισσότερος έλεγχος των λειτουργιών που μπορούν να γίνουν στο πλαίσιο. Εδώ μπαίνει η υλοποίηση δομών δεδομένων όπως η στοίβα και η ουρά .

Μια στοίβα είναι μια δομή δεδομένων που, όπως ένας πίνακας, είναι παρόμοια με μια λίστα. Το κύριο παράδειγμα πίσω από τη στοίβα είναι το LIFO - Last In, First Out σε ελεύθερη μετάφραση.

Για να καταλάβετε καλύτερα τι σημαίνει αυτό, σκεφτείτε ένα σωρό βιβλία ή πιάτα. Κατά τη στοίβαξη βιβλίων, για παράδειγμα, το πρώτο βιβλίο που αφαιρείται από τη στοίβα είναι απαραίτητα το τελευταίο που τοποθετείται. Αν προσπαθήσουμε να βγάλουμε το τελευταίο βιβλίο από το σωρό, όλα θα καταρρεύσουν. Δηλαδή το τελευταίο βιβλίο που στοιβάζεται είναι το πρώτο που βγαίνει.

Αφαιρώντας αυτήν την αρχή σε κώδικα, μπορεί να φανεί ότι υπάρχουν μόνο δύο δυνατές μέθοδοι για να χειριστείτε τα δεδομένα μιας στοίβας: 1) εισαγάγετε ένα

στοιχείο στην κορυφή της στοίβας και 2) αφαιρέστε ένα στοιχείο από την κορυφή της στοίβας.

Σε αντίθεση με τους πίνακες, οι γλώσσες προγραμματισμού συνήθως δεν έχουν εγγενείς μεθόδους για τη δημιουργία και το χειρισμό στοίβων. Ωστόσο, είναι δυνατή η χρήση μεθόδων πίνακα για την υλοποίηση στοίβων.

Χρήσεις

Η πιο διάσημη περίπτωση χρήσης της στοίβας είναι η στοίβα κλήσης ενός προγράμματος που εκτελείται: η σειρά εκτέλεσης των διεργασιών που «καλούνται» από ένα πρόγραμμα μέσω συναρτήσεων ή μεθόδων υπακούει στην αρχή της στοίβας.

Ένας άλλος πόρος που χρησιμοποιούμε καθημερινά και που χρησιμοποιεί μπαταρίες για να λειτουργήσει είναι ο μηχανισμός «πίσω» και «εμπρός» του προγράμματος περιήγησης (συνήθως αντιπροσωπεύεται με αριστερό και δεξί βέλος). Οι διευθύνσεις που επισκέφθηκαν συσσωρεύονται. Όταν καλούμε τη συνάρτηση «πίσω», η τελευταία διεύθυνση που επισκεφτήκαμε - δηλαδή αυτή που βρίσκεται στην κορυφή της στοίβας - είναι η πρώτη που εμφανίζεται.

Σειρά

Η ουρά έχει παρόμοια δομή με τη στοίβα, αλλά με μια σημαντική εννοιολογική διαφορά: το παράδειγμα πίσω από την ουρά είναι το FIFO - First In, First Out , σε ελεύθερη μετάφραση. Σκεφτείτε μια ουρά ταμείου, για παράδειγμα. Το άτομο που έφτασε πρώτο θα εξυπηρετηθεί (και θα αγοράσει το εισιτήριό του) πριν από το άτομο που έφτασε αργότερα και έμεινε πίσω στην ουρά. Η ουρά ως δομή δεδομένων ακολουθεί την ίδια αρχή.

Επομένως, υπάρχουν επίσης μόνο δύο τρόποι χειρισμού μιας ουράς: 1) Εισαγάγετε ένα στοιχείο στο τέλος της ουράς και 2) αφαιρέστε ένα στοιχείο από την αρχή της ουράς.

Κατάστρωμα Η δομή δεδομένων deque (συντομογραφία για ουρά διπλού άκρου ή "ουρά διπλού τέλους") είναι μια παραλλαγή της ουράς που υποστηρίζει την

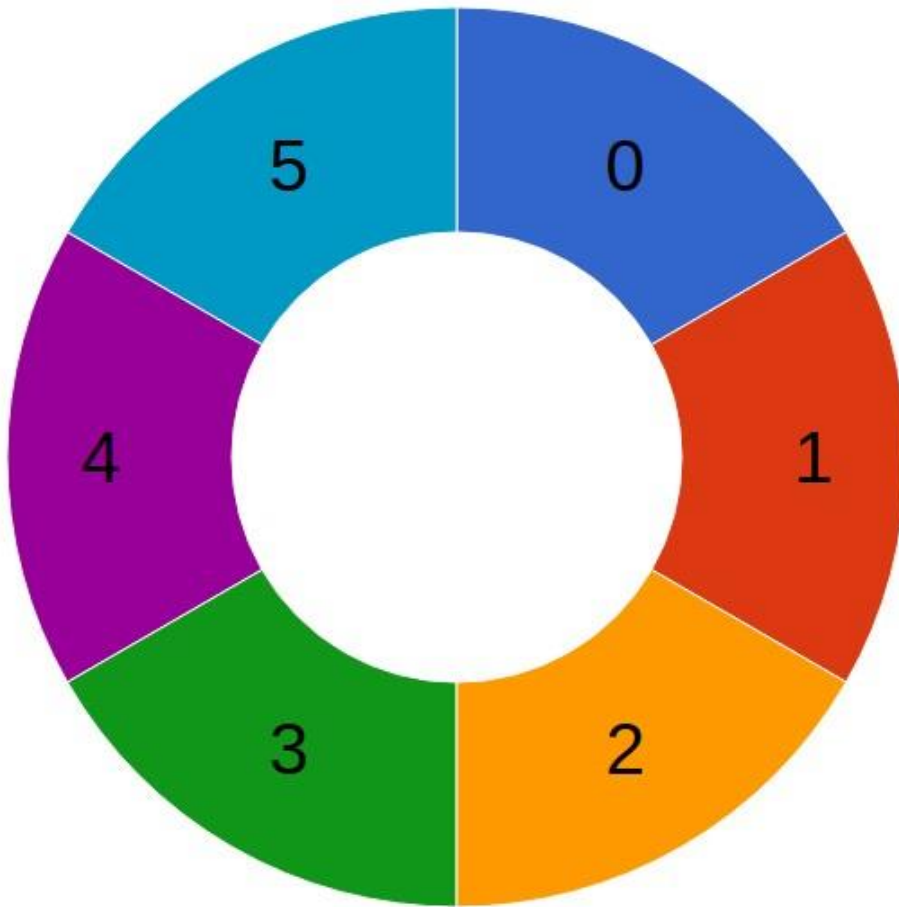
εισαγωγή και αφαίρεση στοιχείων τόσο από την αρχή όσο και από το τέλος της ουράς.

Μπορούμε να συγκρίνουμε, πάλι, με μια ουρά ατόμων σε ένα γραφείο εξυπηρέτησης: ένας ηλικιωμένος που φτάνει εξυπηρετείται πριν (δηλαδή, δεν μπορεί να τοποθετηθεί στο τέλος της ουράς), ταυτόχρονα με ένα άτομο που μπήκε στο τέλος της ουράς. μπορείτε να σταματήσετε την αναμονή και να φύγετε (σε αυτήν την περίπτωση, δεν μπορούμε να περιμένουμε το άτομο να φτάσει στο μπροστινό μέρος της γραμμής για να το βγάλουμε από εκεί).

Ένας άλλος τρόπος για να κατανοήσουμε τη δομή deque είναι ως ένωση των δομών στοίβας και ουράς.

κυκλική ουρά

Μια άλλη παραλλαγή της ουράς είναι η κυκλική ουρά , όπου το τελευταίο στοιχείο συνδέεται με το πρώτο στοιχείο - όπως σε κύκλο:



Η κυκλική ουρά επιδιώκει να λύσει έναν περιορισμό της γραμμικής ουράς, που είναι η αντιμετώπιση των κενών χώρων που μπορούν να σχηματίσουν ουρά μετά την αφαίρεση στοιχείων από την κεφαλή της ουράς.

χρήσεις

Μια εύκολη στην απομνημόνευση χρήση της ουράς είναι ακριβώς η ουρά εκτύπωσης των λειτουργικών συστημάτων: η τελευταία εργασία εκτύπωσης που θα προστεθεί στην ουρά θα είναι η τελευταία που θα εκτυπωθεί.

Επιπλέον, τα αιτήματα που γίνονται σε έναν διακομιστή οργανώνονται επίσης σε μια ουρά για να απαντηθούν και όταν κάνουμε εναλλαγή μεταξύ προγραμμάτων

χρησιμοποιώντας τη συντόμευση alt+tab, το λειτουργικό σύστημα διαχειρίζεται τη σειρά χρησιμοποιώντας την αρχή της κυκλικής λίστας.

συνδεδεμένη λίστα

Έχουμε ήδη δει ότι οι περισσότερες γλώσσες προγραμματισμού έχουν εγγενείς μεθόδους χειρισμού πινάκων, όπως η εισαγωγή και η αφαίρεση στοιχείων. Επίσης, αυτές οι μέθοδοι κάνουν πολύ τη δουλειά της ταξινόμησης και της ανάκτησης στοιχείων για εμάς.

Ωστόσο, υπάρχουν τρία σημαντικά πράγματα που πρέπει να γνωρίζετε για τους πίνακες: 1) στις περισσότερες γλώσσες προγραμματισμού, οι πίνακες έχουν σταθερό μέγεθος. 2) όλα τα στοιχεία καταλαμβάνουν διαδοχικούς χώρους στη μνήμη και 3) η εισαγωγή ή η αφαίρεση στοιχείων από τη μέση του πίνακα δεν είναι πολύ απλή καθώς απαιτεί τη μετατόπιση αυτών των στοιχείων.

Λεξικό ή hashmap

Το λεξικό (γνωστό και ως χάρτης, χάρτης ή hashmap) είναι μια δομή που αποθηκεύει δεδομένα σε ζεύγη κλειδιών και τιμών και χρησιμοποιεί αυτά τα κλειδιά για να βρει τα στοιχεία που σχετίζονται με αυτά, σε αντίθεση με τις δομές που έχουμε δει μέχρι τώρα, που λειτουργούν με λίστες (διαδοχικές ή όχι) μόνο αξίες. Αυτή η περιγραφή μοιάζει πολύ με μια άλλη δομή που ήδη γνωρίζουμε, το αντικείμενο. Υπάρχουν όμως αρκετές διαφορές μεταξύ λεξικών/χαρτών και αντικειμένων. Για παράδειγμα, είναι δυνατό να αντιστοιχιστεί το μέγεθος ενός λεξικού (δηλαδή, ο αριθμός των ζευγών κλειδιών/τιμών) και τα λεξικά μπορούν να δέχονται οποιονδήποτε τύπο δεδομένων ως κλειδί (τα αντικείμενα δέχονται μόνο συμβολοσειρές ή σύμβολα). Τα λεξικά μπορούν επίσης να έχουν καλύτερες επιδόσεις στην ανάκτηση και τον χειρισμό δεδομένων από τα αντικείμενα, καθώς χρησιμοποιούν αναφορές σε κλειδιά - με παρόμοιο τρόπο με τους δείκτες, τα κλειδιά δείχνουν τη διεύθυνση μνήμης των τιμών τους.

Μια σημαντική πτυχή σχετικά με τα λεξικά έναντι των αντικειμένων: η απόφαση για το ποιο θα χρησιμοποιηθεί θα εξαρτηθεί πολύ από τη γλώσσα. Κάθε γλώσσα εφαρμόζει τις δικές της μεθόδους για τη δημιουργία περιπτώσεων αντικειμένων ή λεξικών και για τον χειρισμό δεδομένων. Για παράδειγμα, τα λεξικά Python δεν ήταν

ταξινομημένα από προεπιλογή μέχρι την έκδοση 3.6, ενώ Map()τα αντικείμενα JavaScript ταξινομούνται από προεπιλογή. άλλες γλώσσες μπορεί να υποστηρίζουν λεξικά αλλά όχι αντικείμενα.

Χρήσεις

Η μορφή λεξικού chave:valογειναι η πιο κοινή για την αντιμετώπιση δεδομένων που είναι αποθηκευμένα σε βάσεις δεδομένων, αλλά όχι μόνο για αυτό: η μορφή χρησιμοποιείται κάθε φορά που υπάρχει ανάγκη συσχέτισης δεδομένων και παραπομπών για μεταγενέστερη αναφορά, είτε πρόκειται για πληροφορίες από μια βάση δεδομένων ενός προϊόντος είτε ακόμη τη σχέση μεταξύ του ονόματος ενός αρχείου και της εσωτερικής διαδρομής του στη μνήμη του υπολογιστή, που δημιουργείται από το λειτουργικό σύστημα.

Δέντρο

Το δέντρο είναι μια μη διαδοχική δομή, πολύ χρήσιμη για την αποθήκευση δεδομένων με ιεραρχικό τρόπο στον οποίο είναι δυνατή η γρήγορη πρόσβαση.

Ένα δέντρο μπορεί να οριστεί ως μια συλλογή δεδομένων που αντιπροσωπεύονται από κόμβους και είναι διατεταγμένα σε ιεραρχικά επίπεδα.

δυναδικός σωρός

Ο δυναδικός σωρός είναι ένας ειδικός τύπος δυναδικού δέντρου, που χρησιμοποιείται συνήθως στους υπολογιστές για την υλοποίηση ουρών προτεραιότητας, επειδή ένας σωρός μπορεί να εξάγει πιο αποτελεσματικά την ελάχιστη ή τη μέγιστη τιμή από μια λίστα. Κάποιος μπορεί να μεταφράσει το σωρό, πολύ χαλαρά, ως «σωρό» ή «τμήμα» δεδομένων.

Ο δυναδικός σωρός διαφέρει από το δυναδικό δέντρο σε δύο κύρια χαρακτηριστικά:

Όλα τα επίπεδα εκτός από το τελευταίο έχουν παιδιά τόσο στα αριστερά όσο και στα δεξιά της ρίζας. Στο τελευταίο επίπεδο, τα παιδιά τοποθετούνται όσο πιο αριστερά γίνεται. Αυτό είναι αυτό που λέμε πλήρες δέντρο.

μπορεί να είναι ένας ελάχιστος σωρός (min heap), για να εξαγάγετε τη μικρότερη τιμή από το δέντρο ή έναν μέγιστο σωρό (max heap) για να εξαγάγετε τη μεγαλύτερη τιμή. Όλοι οι κόμβοι πρέπει να είναι είτε \geq (σε περίπτωση max-heap) είτε \leq (σε περίπτωση min-heap) από τις τιμές των θυγατρικών κόμβων.

χρήσεις

Η δενδρική δομή έχει πολλές διαφορετικές χρήσεις, όπως αλγόριθμους λήψης αποφάσεων στη μηχανική εκμάθηση, ευρετηρίαση βάσεων δεδομένων, ευρετηρίαση και εμφάνιση αρχείων και φακέλων στον εξερευνητή αρχείων λειτουργικών συστημάτων, μεταξύ πολλών άλλων περιπτώσεων.

Ο δυαδικός σωρός, όπως αναφέρθηκε ήδη, χρησιμοποιείται σε ουρές προτεραιότητας (ειδικός τύπος ουράς όπου τα στοιχεία δεν τοποθετούνται στην ουρά στο μοτίβο FIFO, αλλά μάλλον ταξινομούνται κατά προτεραιότητα: υψηλότερη προτεραιότητα στην αρχή της ουράς και χαμηλότερη στο τέλος) και επίσης ένας συγκεκριμένος αλγόριθμος ταξινόμησης, ταξινόμηση σωρού.

Πιρούνι

Μια άλλη μη διαδοχική δομή, το γράφημα είναι ένα σύνολο κόμβων (ή κορυφών), ταξινομημένων ή μη και συνδεδεμένων με ακμές, σχηματίζοντας μια δομή με τη μορφή δικτύου.

χρήσεις

Τα κοινωνικά δίκτυα χρησιμοποιούν γραφήματα για να χειριστούν τον μεγάλο όγκο δεδομένων που σχετίζονται μεταξύ τους που λαμβάνουν κάθε στιγμή. Το πιο διάσημο παράδειγμα, η γλώσσα ερωτημάτων GraphQL, δημιουργήθηκε από το Facebook για να χρησιμοποιεί γραφήματα για πρόσβαση και συσχέτιση δεδομένων.

Μια άλλη διάσημη χρήση των γραφημάτων είναι το σύστημα πλοήγησης των εφαρμογών χαρτών/GPS, που χρησιμοποιούν γραφήματα και τον αλγόριθμο συντομότερης διαδρομής για να σχεδιάσουν διαδρομές.

συμπέρασμα

Όπως ίσως έχετε παρατηρήσει, μόνο μία από κάθε σύντομη δομή θα μπορούσε να πάρει ώρες και ώρες μελέτης για να κατανοήσει πλήρως το παράδειγμα πίσω από κάθε μία και να εφαρμόσει κάθε απαιτούμενη τάξη και μέθοδο. Αρκετά πλαίσια

έχουν ήδη τις δικές τους μεθόδους και βιβλιοθήκες που υλοποιούνται στις πιο χρησιμοποιούμενες γλώσσες προγραμματισμού, αλλά είναι πάντα ενδιαφέρον να κατανοήσουμε τι συμβαίνει «κάτω από την κουκούλα».

Αλγόριθμος

Αν και η αρχή είναι παρόμοια με αυτή των μαθηματικών, ένας τεχνολογικός αλγόριθμος εκτελεί αυτοματοποιημένες ενέργειες λαμβάνοντας τιμές εισόδου και παράγοντας τιμές εξόδου. Ένα σαφές παράδειγμα αυτού είναι ο αλγόριθμος της Google.

Οι αλγόριθμοι είναι μέρος της καθημερινότητάς μας.

Ακόμα κι αν δεν το ξέρετε, υπάρχουν σε μερικά από τα πιο διάσημα εργαλεία του διαδικτύου.

Αυτό περιλαμβάνει μέσα κοινωνικής δικτύωσης, μηχανές αναζήτησης, πλατφόρμες βίντεο και μουσικής και πολλά άλλα.

Αλλά, τελικά, γιατί είναι σημαντικά;

Λοιπόν, όταν σκεφτόμαστε το μάρκετινγκ, είναι απαραίτητο να κατανοήσουμε πώς λειτουργεί.

Αυτό συμβαίνει επειδή η μελέτη του σας επιτρέπει να βρείτε πλεονεκτήματα για τη βελτίωση της απόδοσης των καμπανιών ιστού σας.

Με άλλα λόγια, η γνώση των μυστικών τους είναι ένας ανταγωνιστικός παράγοντας διαφοροποίησης.

Όταν τα ανακαλύπτουμε, σχεδιάζουμε πιο αποτελεσματικές και βελτιστοποιημένες στρατηγικές, που μας φέρνουν πιο κοντά σε καλύτερα αποτελέσματα.

Και ποιος δεν το θέλει αυτό για την εταιρεία, το εμπορικό σήμα ή το έργο του;

Αν θέλετε να μάθετε τι είναι και πώς να χρησιμοποιείτε αλγόριθμους υπέρ των ενεργειών σας στο διαδίκτυο, έχετε έρθει στο σωστό μέρος.

Από εδώ και πέρα, θα μιλήσω για αυτό το ρομποτικό σύμπαν και θα υποδείξω τρόπους επιτυχούς χρήσης αυτών των πληροφοριών.

Τι είναι ένας αλγόριθμος στο διαδίκτυο;

Στην αρχική του ιδέα, ένας αλγόριθμος είναι μια ακολουθία κανόνων που στοχεύουν στην επίλυση προβλημάτων.

Δηλαδή, είναι σχηματικά βήματα που εκτελούν μια εργασία.

Στα μαθηματικά, χρησιμοποιούνται ως εναλλακτικός τρόπος για την εκτέλεση υπολογισμών.

Ωστόσο, όταν μεταφέραμε την έννοια στην τεχνολογία της πληροφορίας, επαναπροσδιορίστηκε.

Αν και η αρχή είναι παρόμοια, ένας τεχνολογικός αλγόριθμος εκτελεί αυτοματοποιημένες ενέργειες λαμβάνοντας τιμές εισόδου και παράγοντας τιμές εξόδου.

Για να γίνει πιο κατανοητό, ας χρησιμοποιήσουμε ένα παράδειγμα με ένα εργαλείο που χρησιμοποιούμε στην καθημερινή μας ζωή: το Google , του οποίου ο αλγόριθμος ονομάζεται Googlebot.

Επομένως, θα θεωρήσουμε ότι οι μαθηματικές εξισώσεις που το συνθέτουν είναι ρομπότ.

Όταν ένας χρήστης πραγματοποιεί μια αναζήτηση, ενεργοποιείται.

Σαρώνουν το διαδίκτυο αναζητώντας τη σελίδα που απαντά καλύτερα στην ερώτηση του χρήστη του Διαδικτύου.

Στη συνέχεια σαρώνει ξανά το διαδίκτυο για να βρει τη δεύτερη καλύτερη σελίδα.

Έτσι, δημιουργείται η κατάταξη της Google.

Φυσικά, όλες αυτές οι διαδικασίες συμβαίνουν εν ριπή οφθαλμού, οπότε δεν συνειδητοποιούμε πόσες ενέργειες εκτελούνται.

Τώρα με ρωτάτε: αλλά πώς αποφασίζουν τα ρομπότ ποια σελίδα είναι η καλύτερη;

Λοιπόν, εκεί μπαίνουν τα κριτήρια για το **bot** οι παράγοντες κατάταξης .

Υπάρχουν περισσότερα από 200 από αυτά, δεν είναι όλα γνωστά, οπότε αξίζει να αναφέρουμε μόνο μερικά από τα πιο σημαντικά.

Το πρώτο είναι η εμφάνιση της λέξης-κλειδιού .

Όταν τα ρομπότ εντοπίζουν μια αντιστοίχιση μεταξύ του όρου αναζήτησης και των λέξεων-κλειδιών στον τίτλο, τους υπότιτλους, τις ετικέτες, τις μετα-ετικέτες , την περιγραφή και σε όλο το κείμενο, εκχωρούν μια συγκεκριμένη βαθμολογία.

Ομοίως, όταν εντοπίζουν ότι υπάρχει καλή ποσότητα και ποιότητα εξωτερικών συνδέσμων που οδηγούν στη σελίδα, κερδίζει κάποιους πόντους.

Όταν συνειδητοποιούν ότι οι χρήστες που εισέρχονται σε μια σελίδα και περνούν χρόνο ανάλογο με το μέγεθος του περιεχομένου και συνεχίζουν την περιήγηση εντός του τομέα, προσθέτουν μερικούς ακόμη αριθμούς στον πίνακα αποτελεσμάτων.

Βλέπετε, εδώ, μόλις ανέφερα τρία από αυτά.

Υπάρχουν πολλά άλλα όπως η ταχύτητα φόρτωσης της σελίδας , ο αποκριτικός σχεδιασμός , ο χάρτης ιστότοπου , η ηλικία σελίδας και μια σειρά από άλλες στρατηγικές λευκού καπέλου .

Αυτές είναι, επομένως, μερικές από τις πιο γνωστές τεχνικές SEO (Search Engine Optimization) .

Τώρα, συνειδητοποιήστε επίσης ότι τα ρομπότ είναι έξυπνα.

Αυτό σημαίνει ότι είναι προσαρμόσιμοι.

Εντοπίζουν νέες σελίδες και τις καταχωρούν αυτόματα στην αναζήτηση, συλλέγοντας καθημερινά δεδομένα που αυξάνουν ή μειώνουν τη θέση τους στα αποτελέσματα αναζήτησης, σελίδες γνωστές ως SERP (Searching Engine Result Pages).

Όλες αυτές οι πληροφορίες αποθηκεύονται σε μια μεγάλη βάση δεδομένων που ανατροφοδοτείται καθημερινά.

Φαίνεται λίγο περίπλοκο και κάθε πλατφόρμα έχει τον δικό της αλγόριθμο.

Αλλά αυτός είναι ακριβώς ο τρόπος με τον οποίο καθίσταται δυνατό να αξιολογηθούν αυτές οι εξισώσεις με βάση μετρήσεις και να βρεθούν κενά για να ξεχωρίσετε στο ψηφιακό περιβάλλον.

Πώς προέκυψαν οι αλγόριθμοι;

Ίσως δεν το ξέρατε αυτό: αλγόριθμοι υπήρχαν για πολλά χρόνια πριν δημιουργηθούν οι υπολογιστές.

Είναι παρόντα στα μαθηματικά εδώ και αιώνες – όπως στο κόσκινο του Ερατοσθένη και στον αλγόριθμο του Ευκλείδη .

Με την πάροδο του χρόνου, η μαθηματική έννοια ενσωματώθηκε στο σύμπαν των υπολογιστών.

Το 1968, η Barbara Liskov προσαρμοσε τον αλγόριθμο για χρήση στον προγραμματισμό.

Το PhD στην Επιστήμη των Υπολογιστών από το Πανεπιστήμιο του Στάνφορντ, στις Ηνωμένες Πολιτείες, άρχισε να μελετά τα προγράμματα όχι ως τεχνικά θέματα, αλλά ως μαθηματικά.

Με αυτόν τον τρόπο κατάφερε να βελτιώσει τη γλώσσα προγραμματισμού εισάγοντας εντολές μέσω αλγορίθμων.

Η προσαρμογή του στους υπολογιστές θεωρήθηκε επαναστατική για τους προγραμματιστές και, το 2008, η Barbara έλαβε το βραβείο Turing - το Νόμπελ Υπολογιστών - για το κατόρθωμά της.

Πώς λειτουργούν οι αλγόριθμοι;



Ένας αλγόριθμος λειτουργεί με αυτοματοποιημένο και έξυπνο τρόπο.

Ωστόσο, οι προγραμματιστές δεν μας παρέχουν τις πληροφορίες και τις εξισώσεις που χρησιμοποιούνται για τον υπολογισμό του περιεχομένου που παραδίδεται στον χρήστη.

Αυτό, φυσικά, δημιουργεί μια εντελώς νέα επιχειρηματική θέση.

Γι' αυτό υπάρχουν εκατομμύρια ερευνητές αφοσιωμένοι στον πειραματισμό και τη μελέτη αυτών των περιοχών.

Κάθε κινητήρας έχει τα δικά του χαρακτηριστικά.

Στο παράδειγμά μας, χρησιμοποιούμε τη μηχανή αναζήτησης της Google, αλλά το Yahoo!, το Bing και το Baidu έχουν όλες τις δικές τους μηχανές.

Το ίδιο ισχύει για κάθε κοινωνικό δίκτυο, το οποίο λαμβάνει υπόψη διαφορετικούς παράγοντες.

Σε αυτήν την περίπτωση, οι αλληλεπιδράσεις (ή η δέσμευση) έχουν πολύ μεγαλύτερη βαρύτητα, αφού τέτοιες πλατφόρμες χρησιμοποιούνται για κοινωνικοποίηση.

Εν ολίγοις, πρόκειται για συνεχή εργασία που αξιολογεί τη συμπεριφορά των χρηστών και επιτρέπει νέες ρυθμίσεις, παρέχοντας την καλύτερη δυνατή εμπειρία.

Πόσο σημαντικοί είναι οι αλγόριθμοι για τη στρατηγική ψηφιακού μάρκετινγκ;



Όπως παρατηρήσατε, οι αλγόριθμοι είναι πολύ σημαντικοί για το ψηφιακό μάρκετινγκ .

Είναι απαραίτητα για την εκπλήρωση των στόχων της εκστρατείας.

Για παράδειγμα, εάν ο στόχος σας είναι να δημιουργήσετε οργανική (ή δωρεάν) επισκεψιμότητα, θα πρέπει να επενδύσετε στη βελτιστοποίηση μηχανών αναζήτησης.

Εξάλλου, είναι υπεύθυνοι για ένα σημαντικό μέρος των κλικ στο διαδίκτυο και ένας φυσικός τρόπος για να ανακαλύψουν οι ξένοι τις σελίδες του ιστότοπού σας.

Ωστόσο, εάν θέλετε να αυξήσετε την αναγνωρισιμότητα της επωνυμίας σας και να αναβαθμίσετε την επωνυμία σας, τα μέσα κοινωνικής δικτύωσης είναι μια εξαιρετική επιλογή.

Η γνώση του αλγόριθμου της επιλεγμένης πλατφόρμας αλληλεπίδρασης θα κάνει τις αναρτήσεις σας να έχουν μεγαλύτερη αφοσίωση, απήχηση και θετική εικόνα στην αντίληψη του χρήστη.

Επιπλέον, φυσικά, για να αυξήσετε τον αριθμό των followers σας.

Αυτά είναι ανταγωνιστικά διαφοροποιητικά.

Όταν επενδύετε στην κατανόηση του αλγορίθμου, οι δείκτες των ενεργειών σας στο διαδικτυακό μάρκετινγκ τείνουν να αυξάνονται.

Η σημασία των αλγορίθμων στους υπολογιστές

Έχετε ήδη δει μέχρι στιγμής πώς υπάρχουν οι αλγόριθμοι στην καθημερινότητά σας και πόσο σημαντικοί είναι για τη στρατηγική σας για το ψηφιακό μάρκετινγκ.

Αλλά, φυσικά, είναι επίσης πολύ σχετικά με τον προγραμματισμό.

Εξάλλου, οι αλγόριθμοι περνούν από τους υπολογιστές σε άλλους τομείς χρήσης.

Αρχικά, πρέπει να κατανοήσουμε πώς λειτουργεί μια γλώσσα προγραμματισμού.

Είναι βασικά ένα σύνολο «λέξεων» που κάνουν το μηχάνημα να λειτουργεί – υπολογιστής, κινητό τηλέφωνο ή οτιδήποτε άλλο.

Στη συνέχεια, η γλώσσα επικοινωνεί με το αντικείμενο με τον τρόπο που θα έπρεπε να εκτελεί τις συνήθεις εργασίες της – όπως ενεργοποίηση, απενεργοποίηση, άνοιγμα ενός προγράμματος περιήγησης, αναπαραγωγή μουσικής, μεταξύ άλλων.

Κάθε μία από αυτές τις λειτουργίες χρειάζεται κάποιου είδους εντολή. Και στους υπολογιστές, οι αλγόριθμοι δίνουν τη σειρά.

Αν πρόκειται να κατανοήσουμε τους αλγόριθμους με πιο οπτικό ή γραφικό τρόπο, μπορούμε να τους φανταστούμε ως ένα μεγάλο διάγραμμα ροής.

Έτσι, σηματοδοτούν ποιες και πώς πρέπει να εκτελούνται οι λειτουργίες από τη μηχανή, μέσω μιας πολύπλοκης γλώσσας προγραμματισμού.

Παραδείγματα αλγορίθμων

Δεν υπάρχει μόνο ένας τύπος αλγορίθμου στο ψηφιακό μάρκετινγκ.

Κάθε σύστημα και κάθε εταιρεία θα το αναπτύξει σύμφωνα με τις ανάγκες της διοίκησης.

Έτσι, υπάρχουν πολλές μορφές αλγορίθμων, οι οποίες εξυπηρετούν διαφορετικά συστήματα.

Μερικά από τα πιο διάσημα παραδείγματα σε αυτό το σύμπαν είναι αυτά που αναπτύχθηκαν και χρησιμοποιούνται από τις ακόλουθες εταιρείες:

Google (Googlebot)

Facebook (EdgeRank)

κελάδημα

Ίνσταγκραμ

YouTube

LinkedIn

Spotify

Τικ Τοκ.

Γνωρίζατε ότι ακόμη και μια συνταγή για κέικ μπορεί να θεωρηθεί αλγόριθμος;

Τα εγχειρίδια οδηγιών, οι μαθηματικές συναρτήσεις και οποιαδήποτε άλλη λογική ακολουθία συλλογισμών που οδηγεί σε μια ενέργεια είναι παραδείγματα αλγορίθμων.

Ποια είναι τα κύρια οφέλη από τη χρήση ενός αλγορίθμου προς όφελός σας;

Υπάρχουν πολλά οφέλη που προκύπτουν από την κατανόηση του τρόπου λειτουργίας των αλγορίθμων.

Φανταστείτε, για παράδειγμα, ότι επενδύετε σε δραστηριότητες εισερχόμενου μάρκετινγκ .

Σε αυτήν την τακτική, οι εντελώς άγνωστοι αναμένεται να γίνουν πελάτες μέσω ενός εργαλείου που είναι γνωστό ως διοχέτευση πωλήσεων .

Τώρα, κάντε μια ερώτηση: πώς μπορούν άνθρωποι που δεν γνωρίζουν καν τον ιστότοπό σας να καταλήξουν στις σελίδες σας;

Υπάρχουν δύο πιθανές απαντήσεις σε αυτό: μέσω πληρωμένης διαφήμισης (ή συνδέσμων χορηγίας) ή οργανικά.

Στη δεύτερη περίπτωση, θα χρειαστεί να βασιστείτε στη βοήθεια ψηφιακών εργαλείων και οι αλγόριθμοι είναι οι κύριοι.

Έτσι, μπορείτε να προσελκύσετε επισκέπτες στις σελίδες σας, όπου χρησιμοποιούνται άλλες μέθοδοι για να τους πείσετε να γίνουν δυνητικοί πελάτες , να δημιουργήσετε μια διαρκή σχέση και, τέλος, να κάνετε μια προσφορά.

Εν ολίγοις, η κατανόηση του αλγορίθμου σημαίνει κατανόηση του τρόπου λειτουργίας του μυαλού του χρήστη και προσαρμογή στις προτιμήσεις του.

Ως εκ τούτου, πολλοί ειδικοί υπερασπίζονται τη θεωρία ότι οι μηχανές αναζήτησης είναι η μηχανική αναπαράσταση της σκέψης του χρήστη του Διαδικτύου.

Είναι το ίδιο εργαλείο που κάνει προτάσεις για νέο περιεχόμενο σε κανάλια όπως το YouTube, για παράδειγμα.

Με την ερμηνεία τους, μπορούμε να αναπτύξουμε νέες ενέργειες που πολλαπλασιάζουν τα κέρδη σας από τις επενδύσεις μάρκετινγκ.

Τα ακόλουθα είναι μερικά από τα κύρια οφέλη που αποκτήθηκαν.

τάσεις προβλέψεων

Η γνώση αλγορίθμων σάς επιτρέπει να είστε ένα βήμα μπροστά στο στάδιο του σχεδιασμού.

Αυτό σημαίνει ότι είναι δυνατό να προβλέψετε τις τάσεις και να τοποθετήσετε τον εαυτό σας ακόμη και πριν πραγματοποιηθούν ενημερώσεις.

Αυτό είναι σωστό: αυτοί οι μηχανισμοί υπόκεινται σε συνεχείς ενημερώσεις.

Για παράδειγμα, πριν από λίγο καιρό η Google ανανέωσε τον αλγόριθμό της, δίνοντας μεγαλύτερη σημασία σε σελίδες που προσαρμόζονται σε διαφορετικούς τύπους οθόνης.

Όσοι είχαν βελτιστοποιήσει προηγουμένως σελίδες ιστολογίου ή ιστότοπου κέρδισαν αρκετούς βαθμούς κατάταξης και είδαν τους δείκτες απόδοσής τους να αυξάνονται.

Αυτοματοποίηση

Οι αλγόριθμοι είναι αυτόματα εργαλεία.

Φανταστείτε πόση δουλειά θα ήταν αν υπήρχε μια ομάδα αφιερωμένη στη μη αυτόματη αξιολόγηση κάθε αναζήτησης που εκτελείται από εκατομμύρια χρήστες του Διαδικτύου σε όλο τον κόσμο.

Από τη μία πλευρά, αυτό δυσκολεύει τις ενέργειες πληροφοριών.

Αλλά, από την άλλη πλευρά, μας επιτρέπει να κατανοήσουμε τη λειτουργία του εντοπίζοντας πρότυπα συμπεριφοράς του μηχανισμού.

Με αυτόν τον τρόπο, μπορούμε να σχεδιάσουμε ενέργειες μάρκετινγκ που λειτουργούν στον αυτόματο πιλότο.

Αυτό είναι πολύ συνηθισμένο, για παράδειγμα, σε εικονικά καταστήματα, τα οποία χρησιμοποιούν ρομποτικά εργαλεία για τη δημιουργία διευθύνσεων URL που ταιριάζουν με την πρόθεση αναζήτησης του χρήστη.

Εξέλιξη

Όπως αναφέρθηκε, οι αλγόριθμοι είναι προσαρμοστικά εργαλεία.

Δηλαδή, διαβάζοντας τη συμπεριφορά του χρήστη, αναπροσαρμόζονται για να τον εξυπηρετήσουν καλύτερα.

Προφανώς, αυτό αντιπροσωπεύει μια εντελώς νέα σειρά ευκαιριών για τους επαγγελματίες του μάρκετινγκ .

Μια κορυφαία σελίδα της Google δεν θα υπάρχει για πάντα.

Μόλις ο κινητήρας εντοπίσει μια άλλη ιστοσελίδα υψηλότερης ποιότητας, κερδίζει τη θέση της.

Ομοίως, οι σχέσεις τους με τα μέσα κοινωνικής δικτύωσης αλλάζουν με την πάροδο του χρόνου.

Χάρη σε αυτό, έχουμε την ευκαιρία να εξερευνήσουμε αυτό το χαρακτηριστικό και να εξελίξουμε τις στρατηγικές μάρκετινγκ.

Σύσταση με βάση το περιεχόμενο

Να θυμάστε ότι οι αλγόριθμοι είναι προσανατολισμένοι στην εμπειρία χρήστη .

Δηλαδή κάνουν εξατομικευμένες συστάσεις ανάλογα με τη διαδικτυακή συμπεριφορά του καθενός.

Σύσταση βασισμένη στη συνεργασία

Καθώς περιηγείστε σε έναν ιστότοπο, είτε πρόκειται για ιστολόγιο, είτε για κοινωνικό δίκτυο είτε για ηλεκτρονικό εμπόριο , οι αλγόριθμοι συλλέγουν πληροφορίες.

Αυτό σημαίνει ότι οι συνεισφορές σας έχουν άμεσο αντίκτυπο στα αποτελέσματα και τις δημοσιεύσεις που θα σας εμφανιστούν.

Μεταφέρετε το στη μαζική μάζα των ανθρώπων που χρησιμοποιούν τον Ιστό ταυτόχρονα και έχουμε συστάσεις με βάση τις δραστηριότητές τους.

Πώς χρησιμοποιούνται οι αλγόριθμοι στο διαδίκτυο;

Τώρα, ίσως αναρωτιέστε πώς χρησιμοποιούνται ούτως ή άλλως οι αλγόριθμοι στο διαδίκτυο;

Λοιπόν, μέχρι στιγμής, έχουμε δει ότι κάθε πλατφόρμα έχει το δικό της σύνολο αναλύσεων.

Έτσι, θα μιλήσω για τα κύρια ξεχωριστά.

Αλγόριθμος Google

Στο αρχικό παράδειγμα αυτού του κειμένου, μίλησα πολύ για τον αλγόριθμο της Google.

Βασίζεται σε προθέσεις αναζήτησης, λαμβάνοντας υπόψη τη συλλογική συμπεριφορά των χρηστών.

Με άλλα λόγια, μπορώ να πω ότι οι σελίδες που απαντούν στις ερωτήσεις ενός ατόμου έχουν επιρροή στα αποτελέσματα που εμφανίζονται σε άλλους.

Αλγόριθμος Facebook

Ο αλγόριθμος του Facebook, με τη σειρά του, είναι προσανατολισμένος στις αλληλεπιδράσεις.

Δηλαδή, τα likes, τα shares, τα σχόλια, τα κλικ, οι προβολές και άλλες κοινωνικές αντιδράσεις τροφοδοτούν τη μηχανή.

Εκτός βέβαια από την ιστορία του.

Μπορείτε εύκολα να παρατηρήσετε ότι οι αναρτήσεις από φίλους με τους οποίους αλληλεπιδράτε τακτικά εμφανίζονται πιο συχνά στη ροή σας.

Ένας από τους κύριους παράγοντες που αναλύουν οι αλγόριθμοι του Facebook για να αποφασίσουν σχετικά με τη συνάφεια μιας ανάρτησης είναι η ημερομηνία δημοσίευσής της.

Στην περίπτωση αυτή, ο κανόνας είναι σαφής: όσο πιο πρόσφατη είναι η δημοσίευση, τόσο μεγαλύτερη είναι η διανομή της.

Δηλαδή, καθώς η ανάρτηση παλιώνει, ο αλγόριθμος την δείχνει σε λιγότερους ανθρώπους.

Το Facebook το κάνει αυτό λαμβάνοντας επίσης υπόψη την τελευταία ημερομηνία σύνδεσης του χρήστη.

Έτσι, όσο περισσότερες επισκέψεις κάνει, τόσο πιο πρόσφατο τείνει να είναι το περιεχόμενο που θα δει στο timeline του.

Τύπος περιεχομένου



Σημειώστε ότι στο Facebook, καθώς και στο Google και σε άλλα κοινωνικά δίκτυα, τα κριτήρια που χρησιμοποιεί ο αλγόριθμος για την εμφάνιση περιεχομένου ακολουθούν μια αμφίδρομη οδό.

Στη μία πλευρά βρίσκονται τα χαρακτηριστικά και οι ιδιότητες του περιεχομένου.

Το περιεχόμενο, η ημερομηνία δημοσίευσης, τα κείμενα και οι σύνδεσμοι είναι παράγοντες που επηρεάζουν όταν το ρομπότ του Facebook αποφασίζει τι θα δείξει και τι όχι.

Από την άλλη είναι ο χρήστης. Με τις προτιμήσεις, τις αλληλεπιδράσεις και τη συμπεριφορά του ασκεί και κάποια επιρροή.

Γι' αυτό, γενικά, ο αλγόριθμος του Facebook δίνει προτίμηση στο εγγενές περιεχόμενο έναντι των εξωτερικών συνδέσμων, εκείνων στους οποίους η εικόνα που εμφανίζεται έχει δυνατότητα κλικ.

Είναι λογικό, εξάλλου, είναι πολύ πιο ασφαλές να επισημαίνουμε δημοσιεύσεις στο ίδιο το Facebook παρά σε εξωτερικό περιεχόμενο που δεν υπόκειται στους ίδιους μηχανισμούς ελέγχου.

Πρόελευση της ανάρτησης

Το περιεχόμενο στα προσωπικά προφίλ έχει προτεραιότητα στη ροή σε σχέση με το περιεχόμενο που δημοσιεύεται από εταιρείες .

Σε κάθε περίπτωση, ο αλγόριθμος λαμβάνει υπόψη και τις προηγούμενες αλληλεπιδράσεις του χρήστη.

Έτσι, εάν αλληλεπιδράτε με εταιρείες περισσότερο από ανθρώπους, είναι πιθανό να δείτε περισσότερο περιεχόμενο από σχετικές επιχειρηματικές σελίδες.

Το ίδιο ισχύει και για τις αλληλεπιδράσεις με τα προσωπικά προφίλ.

Έτσι, όσο περισσότερο παρακολουθείτε τη σελίδα κάποιου, τόσο περισσότερο θα βλέπετε περιεχόμενο από αυτό το άτομο.

Οι αναρτήσεις που γίνονται viral είναι πιο πιθανό να διανέμονται σε περισσότερα άτομα για μεγαλύτερο χρονικό διάστημα.

Επομένως, εάν μια δημοσίευση «σκάσει» με πολλά likes , σχόλια και κοινοποιήσεις, πιθανότατα θα ενισχυθεί η απήχυσή της.

Ένας άλλος τρόπος με τον οποίο μπορεί να συμβεί αυτό είναι εάν μια ανάρτηση έχει μεγάλη απήχηση μεταξύ των ατόμων που είναι συνδεδεμένα με το προφίλ σας.

Με άλλα λόγια, εάν είστε δημοφιλείς μεταξύ των φίλων σας, οι πιθανότητες είναι πολύ μεγάλες να δείτε και εσείς αυτό το περιεχόμενο.

Αλγόριθμος Twitter

Υπήρχε μια εποχή που το Twitter δεν είχε αλγόριθμο.

Δηλαδή όλες οι δημοσιεύσεις εμφανίζονταν χρονολογικά.

Σήμερα, ο χρήστης έχει τη δυνατότητα να προσαρμόσει τη ροή του ώστε να εμφανίζεται σύμφωνα με τις αλληλεπιδράσεις (με αλγόριθμο) ή με τη σειρά με την οποία δημοσιεύτηκαν.

Κατάταξη tweets

Σε αντίθεση με το Facebook, το Twitter έχει ένα είδος εσωτερικής «επιμέλειας».

Αυτό κάνει ότι, όταν έχει πρόσβαση στο δίκτυο, ο χρήστης βλέπει πρώτα μερικά tweets που είχαν επιλεγεί προηγουμένως στο χρονοδιάγραμμά του πριν δει αυτά των φίλων του και τις σελίδες που ακολουθούν.

Σε αυτή την περίπτωση, το Twitter υιοθετεί ένα κριτήριο παρόμοιο με αυτό του Facebook για να καθορίσει τι θα εμφανιστεί, με βάση την απήχηση συγκεκριμένου περιεχομένου μεταξύ των φίλων του χρήστη.

Με άλλα λόγια: εάν μια ανάρτηση έχει καλή απήχηση και αφοσίωση μεταξύ των ατόμων στο δίκτυό σας, είναι πολύ πιθανό να εμφανιστεί ανάμεσα στα tweets που έχουν καταταχθεί.

χρονοδιάγραμμα

Στην προεπιλεγμένη γραμμή χρόνου, ο αλγόριθμος κάνει να εμφανίζονται πρώτα τα πιο πρόσφατα tweets από τα άτομα και τις σελίδες που ακολουθείτε.

Επίσης, δίνει προτεραιότητα στα tweets που αρέσουν ή στα οποία απαντούν τα άτομα που ακολουθείτε ή που ακολουθούν το προφίλ σας – δηλαδή, είναι περισσότερο ένα θέμα αφοσίωσης .

Η ίδια αρχή ισχύει για τα tweets που έχουν αλληλεπιδράσει με τα likes της σελίδας και τις διαφημίσεις.

Σε περίπτωση που έχασες

Ο αλγόριθμος του Twitter έχει επίσης ένα άλλο κριτήριο εμφάνισης περιεχομένου, σύμφωνα με το οποίο ο χρήστης βλέπει παλαιότερα tweets.

Αυτό που έχει σημασία σε αυτήν την περίπτωση είναι η συνάφεια του περιεχομένου με το άτομο σύμφωνα με τις προηγούμενες αλληλεπιδράσεις του και τις δικές του δημοσιεύσεις.

Για παράδειγμα: αν δημοσιεύσετε κάτι σχετικά με το κέικ καλαμποκάλευρου, σε μια επόμενη πρόσβαση, υπάρχει μεγάλη πιθανότητα να δείτε κάποιο σχετικό περιεχόμενο στη ροή σας.

Σε αυτή την ενότητα ισχύει και η συχνότητα προσβάσεων. Επομένως, θα πρέπει να λαμβάνετε ενδείξεις παλιών αναγνώσεων μόνο εάν δεν έχετε συνδεθεί για λίγο.

Αλγόριθμος Instagram



Το Instagram είναι μια περίεργη εφαρμογή, καθώς έχει διαφορετικούς αλγόριθμους που επηρεάζουν ο ένας τον άλλον: ένας για Ιστορίες, ένας για ροή, ένας άλλος για τροχούς και ένας τέταρτος για το IGTV .

Βασικά λειτουργούν παρόμοια με το Facebook και διεγείρονται μέσω των αλληλεπιδράσεων.⁰¹²

Στη ροή, έχουμε τις ετικέτες, τα σχόλια και τα likes.

Στις Ιστορίες, οι προβολές έχουν μεγαλύτερο αντίκτυπο, καθώς ο ίδιος ο χρήστης επιλέγει χειροκίνητα εκείνες στις οποίες θέλει να έχει πρόσβαση.

Στην περίπτωση δημοσιεύσεων που εμφανίζονται σε τροχούς, εμφανίζεται μόνο περιεχόμενο από λογαριασμούς που δεν ακολουθεί ο χρήστης, εφόσον ο αλγόριθμος κατανοεί ότι μπορεί να είναι σχετικοί.

Με τη σειρά τους, τα βίντεο IGTV που εμφανίζονται είναι όλα από διαδοχικούς λογαριασμούς, αρκεί ο αλγόριθμος να τα θεωρεί ενδιαφέροντα για τον χρήστη

Κτήματα

Ένα αξιοπερίεργο για το Instagram είναι ότι όταν κυκλοφόρησε το 2010, δεν υπήρχε αλγόριθμος για τον προσδιορισμό του τι εμφανιζόταν σε μια ροή, παρά μόνο χρονολογική σειρά.

Αυτό θα άλλαζε μόλις έξι χρόνια αργότερα, όταν εφαρμόστηκαν αλγόριθμοι που χρησιμοποιούσαν κριτήρια παρόμοια με εκείνα του Facebook, το οποίο παρεμπιπτόντως απέκτησε το κοινωνικό δίκτυο το 2012.

Εν μέσω αυτών των αλλαγών, η χρονολογική σειρά διατηρήθηκε ως κριτήριο για την επισήμανση περιεχομένου στη γραμμή χρόνου, αν και χωρίς την ίδια συνάφεια με πριν.

Σύμπλεξη

Το κριτήριο εμπλοκής, από την άλλη πλευρά, ακολουθεί ένα μοτίβο πρακτικά πανομοιότυπο με αυτό που χρησιμοποιείται από τον αλγόριθμο του Facebook και άλλων δικτύων.

Επομένως, όσο περισσότερες αφοσίωση έχει μια ανάρτηση τα πρώτα λεπτά μετά τη δημοσίευσή της, τόσο περισσότερα άτομα τείνουν να επηρεάζονται.

Σημειώστε ότι, σε αυτή την περίπτωση, μιλάμε για δημοσιεύσεις που εμφανίζονται στο κοινό χρονοδιάγραμμα.

Όπως είδαμε, για κάθε τύπο δημοσίευσης, το Instagram υιοθετεί αλγόριθμους με διαφορετικά κριτήρια.

Ούτως ή άλλως, η ενασχόληση των φίλων του χρήστη με συγκεκριμένο περιεχόμενο είναι ένας παράγοντας πρακτικά κοινός σε όλους, συμπεριλαμβανομένων των τροχών.

Σχέση

Με βάση τα κριτήρια της σχέσης, το περιεχόμενο που δημοσιεύεται από άτομα ή σελίδες με τις οποίες ο χρήστης έχει αλληλεπιδράσει θα έχει προτεραιότητα να εμφανίζεται σε μια ροή.

Το ίδιο ισχύει και για τις ιστορίες. Έτσι, όσο περισσότερες επισκέψεις και δεσμεύσεις έχει ένα άτομο με ένα συγκεκριμένο προφίλ, τόσο μεγαλύτερη προτεραιότητα έχουν οι αναρτήσεις του και στις ιστορίες του.

Αλγόριθμος YouTube

Το YouTube έχει πολύ μεγαλύτερη επιρροή στη μηχανή αναζήτησης από άλλα κοινωνικά δίκτυα, καθώς μοιάζει περισσότερο με το Google παρά με το Facebook.

Γι' αυτό υπάρχει το SEO για το YouTube, το οποίο λαμβάνει υπόψη τους όρους αναζήτησης που χρησιμοποιούνται για την εκτέλεση της αναζήτησης.

Επιπλέον, οι αυτόματες συστάσεις λαμβάνουν υπόψη τη συμπεριφορά των ατόμων εντός της πλατφόρμας.

Αν και το YouTube μπορεί να θεωρηθεί ακόμη και κοινωνικό δίκτυο από ορισμένες απόψεις, σε αυτό, τα κριτήρια αλγορίθμου πρέπει να λάβουν υπόψη άλλους παράγοντες.

Αυτό συμβαίνει επειδή η ποιότητα και η συνάφεια ενός βίντεο σε αυτήν την πλατφόρμα είναι ακόμη πιο σημαντική για τη διανομή και την απήχυσή του από τις αλληλεπιδράσεις που λαμβάνει.

Με αυτόν τον τρόπο, τα ρομπότ που παρακολουθούν τις προσβάσεις πρέπει να καθοδηγούνται από μετρήσεις που εκφράζουν την απόδοση ενός βίντεο.

Αν και δεν είναι γνωστά όλα τα μυστικά του αλγορίθμου YouTube, είναι γνωστό ότι υπάρχουν τρεις θεμελιώδεις μετρήσεις απόδοσης.

Είναι:

- Χρόνος συνεδρίας : στην οποία καταγράφεται η ώρα κάθε συνεδρίας, μετρώντας την ώρα που ο χρήστης δεν παρακολουθεί βίντεο

- Χρόνος παρακολούθησης : καταγράφεται ο χρόνος που αφιερώνει ένας χρήστης παρακολουθώντας ένα βίντεο
- Συνέπεια : στην οποία αξιολογείται η συχνότητα των αναρτήσεων για ένα κανάλι. Όσο πιο συχνά, τόσο πιο πιθανό είναι τα βίντεό σας να προσεγγίσουν περισσότερα άτομα.

Εκτός από αυτές τις μετρήσεις, ο αλγόριθμος Youtube λαμβάνει επίσης υπόψη:

Δημοτικότητα βίντεο : Όπου ο αλγόριθμος αξιολογεί τον ρυθμό ανάπτυξης και αν το βίντεο δέχεται επιτυχίες με γρήγορο ρυθμό

Προσωρικότητα : Όσο πιο καινούργιο, τόσο μεγαλύτερη απήχηση μπορεί να έχει ένα βίντεο

Κλικ : στα οποία τα δεδομένα εμφανίσεων διασταυρώνονται για τον υπολογισμό του CTR , της αναλογίας κλικ προς αριθμό εμφανίσεων

Αλληλεπιδράσεις : Οι κοινωνικές μετρήσεις του βίντεο, όπως σχόλια, "μου αρέσει" και "δεν μου αρέσει".

Εξατομίκευση

Όπως τα κύρια κοινωνικά δίκτυα, το YouTube καθορίζει επίσης τις προτιμήσεις των χρηστών ως κριτήριο για τον προσδιορισμό της απήχησης ενός βίντεο.

Σε αυτή την περίπτωση, σταθμίζει παράγοντες όπως:

- Πόσες φορές εμφανίστηκε ένα βίντεο στον χρήστη
- Βίντεο με ετικέτα "δεν ενδιαφέρομαι"
- Προηγούμενες δεσμεύσεις με άλλα βίντεο
- Θέματα και κανάλια πρόσβασης
- Ο χρόνος που αφιερώθηκε παρακολουθώντας ένα βίντεο.

Είναι πάντα καλό να επισημαίνουμε ότι τα κριτήρια που χρησιμοποιούν οι αλγόριθμοι (όχι μόνο αυτά του YouTube, αλλά των κοινωνικών δικτύων και της Google) δεν είναι απολύτως γνωστά.

Άλλωστε, οι αλγόριθμοι προγραμματίζονται με σχετικά σύνθετες γραμμές υπολογισμού, στις οποίες εισάγονται μια σειρά από παράγοντες, τύπους και συντακτικά κριτήρια.

Αλγόριθμος LinkedIn

Το μεγαλύτερο κοινωνικό δίκτυο της αγοράς στον κόσμο συνδέει επαγγελματίες σε όλο τον κόσμο.

Εδώ λαμβάνονται υπόψη τα μέσα αλληλεπίδρασης, όπως τα likes, τα σχόλια και τα shares, καθώς και άλλες μορφές, όπως η προβολή περιεχομένου στο LinkedIn Pulse.

Πρόσφατα, προστέθηκαν και άλλες κοινωνικές αντιδράσεις στην πλατφόρμα, όπως «συγχαρητήρια», «το λατρεύω», «ιδιοφυΐα» και «ενδιαφέρον».

Το ίδιο σκεπτικό ισχύει και για το LinkedIn, το κοινωνικό δίκτυο που απευθύνεται σε επαγγελματικές επαφές, τους αλγόριθμους του οποίου θα μάθουμε παρακάτω.

Ποιότητα περιεχομένου

Τα ρομπότ που καθορίζουν εάν το περιεχόμενο αξίζει περισσότερο ή λιγότερο να προσεγγιστεί στο LinkedIn πρώτα αξιολογούν τη συνάφειά του.

Για αυτό, επιδιώκουν να μάθουν εάν η δημοσίευση είναι χαμηλής ποιότητας ή αν είναι ανεπιθύμητη.

Εάν δεν είναι κανένα από τα δύο, ο αλγόριθμος προσπαθεί να εκτιμήσει αν αξίζει ή όχι να αυξήσει την εμβέλεια του περιεχομένου, με βάση επίσης τα ενδιαφέροντα του κάθε χρήστη.

Σύμπλεξη

Ως κοινωνικό δίκτυο, στο LinkedIn, τα δεδομένα αφοσίωσης είναι το κλειδί για τον προσδιορισμό της απήχησης μιας ανάρτησης.

Επομένως, οι αλγόριθμοί του θεωρούν επίσης τον αριθμό των likes, των σχολίων, των αλληλεπιδράσεων και των κοινοποιήσεων ως δείκτες ότι η ανάρτηση αξίζει να προσεγγίσει περισσότερα άτομα.

Τι είναι σε άνοδο;

Όπως και το Twitter, το LinkedIn εκτιμά τις αναρτήσεις που είναι σε άνοδο ή ασχολούνται με ιογενή θέματα .

Παράλληλα, οι αλγόριθμοί του λαμβάνουν επίσης υπόψη τη φήμη του προφίλ που δημιουργεί τη δημοσίευση, καθώς και τη συνάφεια του περιεχομένου με μεμονωμένους χρήστες.

Αλγόριθμος Spotify

Το Spotify είναι μια από τις μεγαλύτερες υπηρεσίες ροής σήμερα. Δίνει στους χρήστες πρόσβαση σε εκατομμύρια τραγούδια, podcast και άλλο ηχητικό περιεχόμενο.

Ένα από τα μεγάλα πλεονεκτήματα της πλατφόρμας είναι ότι επιτρέπει τη δημιουργία εξατομικευμένων και προσβάσιμων λιστών αναπαραγωγής ακόμα και σε λειτουργία εκτός σύνδεσης.

Έτσι, ο χρήστης μπορεί να επιλέξει τα αγαπημένα του περιεχόμενα από τις δικές του λίστες.

Όμως, εκτός από αυτά τα συγκεκριμένα αρχεία, όσοι έχουν πρόσβαση στο Spotify παρατηρούν ότι η πλατφόρμα προσφέρει περιστασιακά νέα τραγούδια ή podcast.

Και, περιέργως, είναι περιεχόμενα που σχετίζονται με το μουσικό στυλ ή το θέμα που ακούει συχνά ο χρήστης.

Φυσικά, οι προτάσεις του Spotify δεν έρχονται «από το πουθενά». Δημιουργούνται από τον αλγόριθμο της πλατφόρμας ροής.

Ουσιαστικά λοιπόν, ο αλγόριθμος του Spotify παρακολουθεί τις ακουστικές σας συνήθειες και προτείνει νέους καλλιτέχνες και παραγωγούς περιεχομένου ήχου σύμφωνα με τις προτιμήσεις σας.

Αλγόριθμος TikTok

Το Tik Tok είναι ένα από τα πιο πρόσφατα και διάσημα κοινωνικά δίκτυα .

Η εφαρμογή είναι ένας χώρος για κοινή χρήση βίντεο που επεξεργάζεται ο χρήστης.

Ως κοινωνικό δίκτυο, οι αλληλεπιδράσεις μεταξύ των «tik tokers» είναι θεμελιώδεις για την τροφοδοσία των αλγορίθμων της κινεζικής εφαρμογής.

Στη συνέχεια, εξετάζει την αφοσίωση περιεχομένου για τη δημιουργία νέων προτάσεων χρήστη.

Εν ολίγοις, ο αλγόριθμος του Tik Tok λειτουργεί πολύ παρόμοια με αυτόν του YouTube και πολλοί χρήστες χρησιμοποιούν τον αλγόριθμο για να αποκτήσουν περισσότερους οπαδούς στο Tik Tok .

Πώς να πουλήσετε περισσότερα στο διαδίκτυο με τη βοήθεια αλγορίθμων;

Οι αλγόριθμοι είναι ισχυροί σύμμαχοι όσον αφορά τη βελτιστοποίηση των πωλήσεων.

Όταν χρησιμοποιούνται σωστά, εκπληρώνουν διαφορετικές λειτουργίες μάρκετινγκ.

Δηλαδή, μπορούν να βοηθήσουν, να επιταχύνουν και να βελτιστοποιήσουν τις μετατροπές σε οποιοδήποτε διαδικτυακές στρατηγικές μάρκετινγκ.

Εξαιτίας αυτού, η κατανόησή τους πρέπει να αξιολογείται καθημερινά, ενισχύοντας τη σημασία της συνεχούς ανάπτυξης.

Συμπέρασμα

Σε αυτό το άρθρο, μίλησα για αλγόριθμους, δίνοντας ιδιαίτερη έμφαση στο τι χρειάζεται για να επιτευχθούν καλύτερα αποτελέσματα μάρκετινγκ.

Ελέγξτε τις έννοιες και καταλάβετε πώς λειτουργούν αυτά τα εργαλεία.

Διαβάστε επίσης για τη σημασία τους, τα κύρια οφέλη και τον τρόπο χρήσης τους στο διαδίκτυο σε διαφορετικές πλατφόρμες.

Εκπαιδευτική Υποενότητα: 3.3 Δομικά στοιχεία προγράμματος

Δομή Προγράμματος

Ένα πρόγραμμα λοιπόν είναι ένα σύνολο από οδηγίες που μπορούν να εκτελεστούν μηχανικά, χωρίς την απαίτηση για την ύπαρξη κάποιας ``κρίσης" εκ μέρους του μηχανήματος που εκτελεί.

Το πρόγραμμα που περιγράφηκε αμέσως προηγούμενα δεν είναι γραμμένο σε κάποια γλώσσα προγραμματισμού. Οι γλώσσες αυτές έχουν πολύ αυστηρό συντακτικό το οποίο είναι αναγκαστικά περιοριστικό στο προγραμματιστή ακριβώς για να είναι ερμηνεύσιμο από μια μηχανή.

Για παράδειγμα, στη γλώσσα προγραμματισμού BASIC, μια από τις απλούστερες ίσως γλώσσες, το παραπάνω πρόγραμμα μπορεί να γραφεί ως εξής:

```
10 print "Give a: "  
20 input a  
30 print "Give b: "  
40 input b  
50 x = 0  
60 if a<b then goto 100  
70 x = x + 1  
80 a = a - b  
90 goto 60  
100 print "The quotient a/b is "; x
```

Η γλώσσα BASIC, στη μορφή αυτή που φαίνεται στο πιο πάνω πρόγραμμα συναντάται πλέον σπάνια. Υπάρχουν πιο εξελιγμένες μορφές της γλώσσας που χρησιμοποιούνται σήμερα.

Παρ' όλα αυτά, στο παραπάνω πρόγραμμα, όπου υπολογίζεται το πηλίκο της διαίρεσης του ακεραίου a διαιρούμενου από τον ακέραιο b (θετικοί ακέραιοι κι οι δυο), εμφανίζονται τα βασικότερα στοιχεία ενός προγράμματος και θα το χρησιμοποιήσουμε ως το βασικό παράδειγμα για την εισαγωγή τους.

Η γλώσσα BASIC, λόγω της απλότητάς της, είναι μάλλον προσφορότερη για την εισαγωγή αυτών των βασικών εννοιών από τη γλώσσα C στην οποία θα μεταπηδήσουμε σύντομα. Προχωρούμε λοιπόν τώρα να δούμε πώς δουλεύει το παραπάνω πρόγραμμα και καθ' οδόν θα δούμε ποια είναι τα βασικά στοιχεία ενός προγράμματος γενικά.

Ας κάνουμε όμως πρώτα μια περιγραφή της στρατηγικής που χρησιμοποιείται στον αλγόριθμο που υλοποιείται στο παραπάνω πρόγραμμα BASIC. Πώς δηλ. υπολογίζεται το πηλίκo της διαίρεσης a/b ;

Το πηλίκo k της διαίρεσης του a διά του b είναι ο μέγιστος ακέραιος τέτοιος ώστε $k \cdot b \leq a$. Ο τρόπος που έχουμε επιλέξει εδώ για να υπολογίσουμε το k είναι το να αρχίσουμε να προσθέτουμε το b στον εαυτό του μέχρι αυτό το άθροισμα να ξεπεράσει το a . Το πλήθος των φορών που προσθέσαμε πριν ξεπεραστεί το a είναι το πηλίκo k .

Μεταβλητές

Οι μεταβλητές (variables) είναι το πιο βασικό κομμάτι ενός προγράμματος. Τις φανταζόμαστε σαν ένα κουτί, που έχει πάντα ένα, συνήθως περιγραφικό, όνομα, και κάποια περιεχόμενα που μπορεί να αλλάζουν από βήμα σε βήμα του προγράμματος. Ακόμη, σε πολλές γλώσσες προγραμματισμού (η C και η παλιά αυτή BASIC που δείχνουμε εδώ συμπεριλαμβάνονται σε αυτές) κάθε μεταβλητή έχει το δικό της σταθερό τύπο δεδομένων που μπορεί να κρατήσει.

Για παράδειγμα μια μεταβλητή μπορεί να έχει τύπο ακέραιο αριθμό, πραγματικό αριθμό (συνήθως λέγονται αριθμοί κινητής υποδιαστολής ή floating point numbers), λογική τιμή (true ή false, δηλ. να είναι αληθής ή ψευδής), λέξη (string), κ.ά. Το προηγούμενο πρόγραμμα σε BASIC έχει τρεις μεταβλητές: τα a , b , x , που και οι τρεις είναι τύπου ακέραιου.

Οι δύο μεταβλητές a και b χρησιμοποιούνται κατ' αρχήν ως μεταβλητές εισόδου: εκεί κρατάει το πρόγραμμα, στην αρχή τουλάχιστον, τους δύο ακεραίους των οποίων το πηλίκo θέλουμε να υπολογίσουμε. Αντιθέτως η μεταβλητή x είναι μεταβλητή που χρησιμοποιείται και για να κρατήσει μερικά ενδιάμεσα αποτελέσματα, αλλά είναι και

η μεταβλητή που στο τέλος των βασικών υπολογισμών κρατάει το επιθυμητό αποτέλεσμα.

I/O (input/output)

Οι εντολές στις γραμμές 10, 20, 30 και 100 είναι εντολές που είτε διαβάζουν (εντολή input της BASIC) κάποια στοιχεία που ο χρήστης γράφει στο πληκτρολόγιο είτε τυπώνουν (εντολή print της BASIC) στην οθόνη του υπολογιστή ή του τερματικού κάποιο κείμενο. Είναι προφανές ότι δεν μπορεί να υπάρξει χρήσιμο πρόγραμμα χωρίς κάποια τέτοια αλληλεπίδραση με το χρήστη. Οι εντολές αυτές λέγονται εισόδου/εξόδου (I/O).

Εντολές ανάθεσης τιμής σε μεταβλητές

Αυτές είναι οι εντολές 50, 70 και 80. Η 50 θέτει τη μεταβλητή x ίση με 0, η 70 αυξάνει την τιμή της ίδιας μεταβλητής κατά 1 και η 80 αφαιρεί από τη μεταβλητή a την τιμή της μεταβλητής b.

Η χρήση του συμβόλου = που μοιάζει κατ' αρχήν αντιφατική σε σχέση με όσα έχει κανείς συνηθίσει από την αλγεβρική του χρήση, πρέπει στη γλώσσα BASIC (όπως και στη γλώσσα C) να ερμηνεύεται ως εξής: αριστερά του = υπάρχει πάντα μια μεταβλητή και δεξιά του υπάρχει πάντα μια παράσταση.

Το πρόγραμμα πρώτα κάνει όσους υπολογισμούς χρειάζεται για να υπολογίσει την τιμή της παράστασης (ενδεχομένως χρησιμοποιώντας και την μεταβλητή που υπάρχει αριστερά του =, και το αποτέλεσμα αυτό *εκχωρεί* στη μεταβλητή που βρίσκεται αριστερά).

Εντολές εκτελούμενες υπό συνθήκη

Η εντολή στη γραμμή 60 αποτελείται από μια συνθήκη ($a < b$) και μια εντολή (goto 100) που εκτελείται μόνο αν ισχύει τη συγκεκριμένη εκείνη χρονική στιγμή εκτέλεσης του προγράμματος η συνθήκη.

Εντολές μεταφοράς ροής προγράμματος

Τέλος η εντολή goto <line number> που εμφανίζεται στις γραμμές 60 και 90 είναι μια εντολή που δεν κάνει τίποτε άλλο από το να μεταφέρει τον έλεγχο ροής του

προγράμματος σε μια συγκεκριμένη γραμμή. Αμέσως μετά τη γραμμή 90, για παράδειγμα, εκτελείται η γραμμή 60 και ποτέ η 100.

Ας δώσουμε τώρα ένα παράδειγμα της ροής του προγράμματος όταν το πρόγραμμα τρέξει με a ίσο με 13, και b ίσο με 10 (εδώ εννοούμε ότι ο χρήστης πληκτρολογεί τους αριθμούς 13 και 10 στο πληκτρολόγιο και με αυτή τη σειρά). Η ροή λοιπόν του προγράμματος σε αριθμούς γραμμών είναι η εξής:

10, 20 (ο χρήστης πληκτρολογεί 13), 30, 40 (ο χρήστης πληκτρολογεί 10), 50, 60, 70, 80, 90, 60, 100 (εκτυπώνεται The quotient a/b is 1).

Εκπαιδευτική Υποενοότητα :3.4 Σχεδιασμός και υλοποίηση προγράμματος
(Ψευδοκώδικας Pascal)

Κύκλος ανάπτυξης προγράμματος/λογισμικού

Η διαδικασία ανάπτυξης λογισμικού εξελίσσεται σε διακριτές φάσεις ή στάδια. Θεωρείται υποσύνολο του κύκλου ζωής ενός συστήματος λογισμικού που ξεκινά από την ανάλυση απαιτήσεων και τελειώνει με την παύση λειτουργίας του.

Αρκετές φορές, οι θεωρητικές προσεγγίσεις για την ανάπτυξη λογισμικού έχουν δεχθεί κριτική για την αποτελεσματικότητά τους, καθώς συχνά, στη πράξη, παρουσιάζονται σημαντικές αποκλίσεις από τον αρχικό σχεδιασμό.

Παρόλα αυτά είναι ιδιαίτερα χρήσιμο, η ανάπτυξη ενός λογισμικού να ακολουθεί συγκεκριμένα μοντέλα, τα οποία έχουν προταθεί ανάλογα με τα προτερήματα και τις αδυναμίες τους.

Μεταξύ των βασικών μεθοδολογιών (μοντέλων) που έχουν προταθεί και ακολουθούνται, θα αναφερθούμε στο μοντέλο του καταρράκτη (waterfall model) και της σπειροειδούς προσέγγισης (spiral model).

Μοντέλο του καταρράκτη

Πρόκειται για το μοντέλο που υποδιαιρεί τη διαδικασία ανάπτυξης ενός συστήματος λογισμικού στις ακόλουθες φάσεις:

- Καθορισμού απαιτήσεων.

- Ανάλυσης απαιτήσεων.
- Σχεδίασης.
- Υλοποίησης.
- Ολοκλήρωσης. Λειτουργίας και συντήρησης.

Οι φάσεις εφαρμόζονται διαδοχικά με τη σειρά (γραμμικά) με κύκλους ανατροφοδότησης.

Μοντέλο σπείρας

Στο μοντέλο της σπείρας, η ανάπτυξη ακολουθεί μια εξελικτική διαδικασία με την επαναληπτική εκτέλεση ενός κύκλου φάσεων, που σε καθεμιά δημιουργείται μια ενδιάμεση έκδοση του τελικού προϊόντος, η οποία βελτιώνεται κατά τον επόμενο κύκλο Κ.Ο.

Η διαδικασία αυτή συνεχίζεται μέχρι να παραχθεί μια έκδοση που να ικανοποιεί τις απαιτήσεις των χρηστών.

Το πλεονέκτημα αυτού του μοντέλου είναι η γρήγορη ανάπτυξη ενός πρωτότυπου και η ανάλυση όλων των κινδύνων, πριν από κάθε φάση.

Το μοντέλο αυτό χρησιμοποιείται περισσότερο σε έργα μεγάλης κλίμακας. Κατά την ανάπτυξη ενός προγράμματος, εργαζόμαστε παρόμοια με τη διαδικασία ανάπτυξης ενός λογισμικού και ακολουθείται η μεθοδολογία υλοποίησης του κύκλου ανάπτυξης προγράμματος (program development life cycle-PDLC). Μια τέτοια μεθοδολογία ανάπτυξης προγράμματος μπορεί να αποτελείται από φάσεις, όπως:

- Ανάλυση του προβλήματος.
- Σχεδίαση.
- Συγγραφή κώδικα.
- Έλεγχος και εκσφαλμάτωση.
- Τεκμηρίωση.

Η λογική συγγραφής προγράμματος ανάλογα με το είδος προγραμματισμού

Από τη δεκαετία του 1960 και μέχρι σήμερα, έχουν αναπτυχθεί διάφορα είδη προγραμματισμού που τα υποστήριξαν πολλές γλώσσες προγραμματισμού. Μπορούμε να τα κατηγοριοποιήσουμε σε μεγάλες κατευθύνσεις, τα λεγόμενα προγραμματιστικά υποδείγματα (programming paradigms).

Τα βασικά προγραμματιστικά υποδείγματα είναι τα ακόλουθα:

- Ο Προστακτικός προγραμματισμός.
- Ο Δηλωτικός προγραμματισμός.

Προστακτικός προγραμματισμός

Ο Προστακτικός προγραμματισμός (Imperative programming) βασίζεται σε εντολές που υλοποιούν τα βήματα ενός αλγόριθμου, και βρίσκεται πιο κοντά στη λογική λειτουργίας του υπολογιστή.

Γλώσσες που ακολούθησαν το είδος αυτό είναι οι κλασικές γλώσσες προγραμματισμού, όπως Cobol, Fortran, Pascal, C κ.ά.

Δομημένος και μη προγραμματισμός

Κατά την αρχική περίοδο του προγραμματισμού, η διακλάδωση και η αλλαγή της ροής εκτέλεσης ενός προγράμματος γίνονταν με τη χρήση της εντολής got. Με την εντολή αυτή, η ροή εκτέλεσης μεταφερόταν σε οποιοδήποτε σημείο του προγράμματος, γεγονός που οδηγούσε σε μη δομημένα προγράμματα (μη δομημένος προγραμματισμός - Unstructured programming). Στη συνέχεια δημιουργήθηκε το πρότυπο του Δομημένου προγραμματισμού (Structured programming), όπου η αλλαγή της ροής εκτέλεσης γίνεται με συγκεκριμένους κανόνες.

Ο Διαδικαστικός προγραμματισμός (Procedural programming) αποτελεί μια υποκατηγορία του δομημένου προγραμματισμού, με το πρόγραμμα να αποτελείται από ομάδες εντολών, τις διαδικασίες (procedures). Η γλώσσα προγραμματισμού που αξιοποίησε, ήδη από το 1970, το είδος αυτό είναι η Pascal.

Ο Αντικειμενοστρεφής προγραμματισμός (Object-oriented programming) βασίζεται - σε αντίθεση με το διαδικαστικό προγραμματισμό- σε αντικείμενα που αλληλεπιδρούν μεταξύ τους. Ο Αντικειμενοστρεφής προγραμματισμός αποτελεί ένα

πρότυπο που ταιριάζει περισσότερο στη λογική οργάνωσης και λειτουργίας του πραγματικού κόσμου.

Δηλωτικός προγραμματισμός

Ο Δηλωτικός προγραμματισμός (Declarative programming) βασίζεται στην περιγραφή του σκοπού, ο οποίος ζητείται να επιτύχει το πρόγραμμα. Στο γενικό αυτό υπόδειγμα ανήκουν διάφορες υποκατηγορίες προγραμματισμού, όπως είναι ο Συναρτησιακός και ο Λογικός.

Συναρτησιακός προγραμματισμός

Ο Συναρτησιακός προγραμματισμός (Functional programming) βασίζεται σε μαθηματικές συναρτήσεις, με γλώσσες όπως Lisp, Logo κ.ά.

Λογικός προγραμματισμός

Στο Λογικό προγραμματισμό (Logic programming), ένα πρόγραμμα είναι ένα σύνολο από αξιώματα ή κανόνες οι οποίοι καθορίζουν σχέσεις ανάμεσα σε αντικείμενα. Υπολογισμός ενός λογικού προγράμματος είναι ένα συμπέρασμα που συνάγεται από τα αποτελέσματά του, όπως για παράδειγμα γίνεται στην Prolog.

Στο προγραμματιστικό πρότυπο του δηλωτικού προγραμματισμού μπορούμε να θεωρήσουμε ότι ανήκουν και άλλες γλώσσες που δεν υπάγονται στις δύο προηγούμενες κατηγορίες.

Χαρακτηριστικές είναι η HTML (HyperText Markup Language), γλώσσα σήμανσης-χαρακτηρισμού υπερκειμένου και εν μέρει η SQL (Structured Query Language) γλώσσα για τη διαχείριση δεδομένων, σε ένα Σύστημα Διαχείρισης Σχεσιακών Βάσεων Δεδομένων (RDBMS Relational Database Management System).

Λοιπά πρότυπα και τεχνικές προγραμματισμού

Εκτός από τα παραπάνω υποδείγματα προγραμματισμού υπάρχουν και άλλα τα οποία, είτε δεν μπορούν να χαρακτηριστούν πλήρως ως προγραμματιστικά υποδείγματα, είτε αποτελούν τεχνικές και μεθοδολογίες προγραμματισμού, που θα αναφέρουμε στη συνέχεια.

Παράλληλος προγραμματισμός (Parallel programming). Επιτρέπει ταυτόχρονη εκτέλεση διαδικασιών από διαφορετικούς επεξεργαστές, όπως στην γλώσσα προγραμματισμού Occam.

Προγραμματισμός οδηγούμενος από γεγονότα (Event-driven programming).

Αποτελεί περισσότερο τεχνική αρχιτεκτονικής ενός προγράμματος σχετικά με τη ροή του, παρά προγραμματιστικό υπόδειγμα. Η ροή του προγράμματος εξαρτάται από την ύπαρξη γεγονότων (events), όπως είναι, για παράδειγμα, ένα μήνυμα ενός αισθητήρα ή μια ενέργεια του χρήστη με το πάτημα του ποντικιού ή ενός πλήκτρου. Παραδείγματα γλωσσών που το ενσωματώνουν είναι η Python, Java κ.ά.

Οπτικός προγραμματισμός (Visual programming).

Εκφράζει τη δυνατότητα γλωσσών ή περιβαλλόντων προγραμματισμού να παρέχουν τη δυνατότητα δημιουργίας του προγράμματος μέσω γραφικών αντικειμένων, αντί της πληκτρολόγησης του κώδικα ο οποίος αντιστοιχεί σε εντολές. Για παράδειγμα, στην κατηγορία αυτή ανήκουν περιβάλλοντα δημιουργίας σεναρίων όπως το Kodu (Kodu, 2016) και το Alice (Alice, 2016) ή το Scratch (Scratch, 2016).

Προγραμματισμός Δέσμης ενεργειών (Script programming)

Είναι τύπος προγραμματισμού και όχι υπόδειγμα, δημιουργίας μικρών τμημάτων κώδικα και όχι ολοκληρωμένων προγραμμάτων. Είναι υψηλού επιπέδου προγραμματισμός που διερμηνεύεται κατά την εκτέλεση από ένα άλλο πρόγραμμα, όπως ένας φυλλομετρητής.

Αρθρωτός ή Τμηματικός Προγραμματισμός (Modular programming)

Σχετίζεται περισσότερο με τεχνική σχεδίασης λογισμικού παρά με πρότυπο. Χαρακτηρίζεται από τη διαίρεση του προβλήματος σε απλούστερα τμήματα, αυτά με τη σειρά τους σε επί μέρους μικρότερα κ.ο.κ. Παρέχει απλούστευση της επίλυσης ενός προβλήματος, ευκολία κωδικοποίησης και συντήρησης. Γενικά ως τμήμα θεωρούμε ένα σύνολο ενεργειών το οποίο εκτελεί μια καθορισμένη λειτουργία ενός προγράμματος και είναι κατά το δυνατόν ανεξάρτητο από τα άλλα τμήματα.

Ιεραρχικός σχεδιασμός

Η μέθοδος ανάλυσης ενός προβλήματος σε μικρότερα, είναι εκείνη με την οποία αντιμετωπίζουμε το πρόβλημα ως μια πολυεπίπεδη δομή. Έτσι, για τη σχεδίασή του,

ξεκινάμε από το υψηλότερο επίπεδο και στη συνέχεια το αναλύουμε σε όλο και χαμηλότερα, έως ότου φθάσουμε στο κατώτερο επίπεδο ανάλυσης. Η τεχνική αυτή ονομάζεται ιεραρχικός σχεδιασμός (Top down design).

Ψευδοκώδικας/Pascal

Ο **ψευδοκώδικας** είναι εργαλείο που χρησιμοποιείται από προγραμματιστές, κυρίως στα αρχικά στάδια της σχεδίασης και κατασκευής ενός προγράμματος. Με τον ψευδοκώδικα, τα βήματα μιας λύσης ή η μορφή ενός αλγόριθμου περιγράφονται με σύντομες και περιεκτικές προτάσεις που όμως υπακούουν σε μια τυποποίηση που πλησιάζει την τυποποίηση μιας γλώσσας προγραμματισμού.

Είναι ένα από τα πιο συνηθισμένα εργαλεία για τον ορισμό αλγορίθμων. Ο ψευδοκώδικας είναι μια αναπαράσταση σε φυσική γλώσσα του κώδικα που απαιτείται για έναν αλγόριθμο. Είναι κατά ένα ποσοστό φυσική γλώσσα και κατά το υπόλοιπο δομημένος κώδικας.

Ο ψευδοκώδικας είναι κάτι που μοιάζει με κώδικα προγραμματισμού, αλλά δεν είναι.

Είναι μια δομημένη γλώσσα που χρησιμοποιεί:

- Στοιχεία (συντακτικές δομές) από τις γλώσσες προγραμματισμού
- Συμβολισμούς (σημασιολογικά στοιχεία) από τα Μαθηματικά και τη Μαθηματική Λογική
- Στοιχεία (λεκτικές περιγραφές) από τη φυσική γλώσσα

Το μέρος της φυσικής γλώσσας παρέχει μια ανεπίσημη σύνταξη η οποία διαβάζεται εύκολα, ενώ το μέρος του κώδικα συνίσταται σε μια εκτεταμένη έκδοση των βασικών αλγοριθμικών δομών, της ακολουθίας, της επιλογής και της επανάληψης.

Η γλώσσα **Pascal** παρουσιάστηκε το 1971 και ο σχεδιασμός της στηρίχθηκε στην ALGOL. Η γλώσσα είναι κατάλληλη τόσο για εκπαίδευση όσο και για τη δημιουργία ισχυρών προγραμμάτων με κύριο χαρακτηριστικό της τη δυνατότητα δημιουργίας ιδιαίτερα δομημένων τμημάτων κώδικα. Χρησιμοποιήθηκε ευρέως ως πρώτη γλώσσα εκμάθησης προγραμματισμού για πολλά χρόνια, μέχρι που ξεπεράστηκε από γλώσσες όπως η C και η Java για το σκοπό αυτό.

Βασικοί Τύποι Δεδομένων

Στον προγραμματισμό υπολογιστών, ένας τύπος δεδομένων (data type) είναι μία κατηγοριοποίηση που προσδιορίζει διαφορετικούς τύπους από δεδομένα.

Οι τύποι δεδομένων χρησιμοποιούνται μέσα σε συστήματα τύπων (type systems), τα οποία προσφέρουν διάφορους τρόπους για τον ορισμό, την υλοποίηση και την χρήση τους.

Διαφορετικά συστήματα τύπων εξασφαλίζουν διαφορετικούς βαθμούς ασφάλειας τύπων (type safety). Τυπικά, ένας τύπος μπορεί να οριστεί ως "οποιαδήποτε ιδιότητα ενός προγράμματος την οποία μπορούμε να προσδιορίσουμε πριν την εκτέλεση του προγράμματος"

Οι συνήθεις τύποι δεδομένων είναι:

- Ακέραιοι (integers),
- Τύποι δεδομένων αληθείας (booleans),
- Χαρακτήρες (characters),
- Κινητής υποδιαστολής (floating-point numbers),
- Αλφαριθμητικές συμβολοσειρές (strings).

Ακέραιος (integer)

Δήλωση τύπου

shortint

integer

longint

byte

word

Επιτρεπτές πράξεις ακεραίων

+ πρόσθεση

- αφαίρεση

* πολλαπλασιασμός

div ακέραια διαίρεση (πηλίκο)

mod υπόλοιπο διαίρεσης

Πραγματικός (real)

Χρησιμοποιείται εκεί που οι αριθμητικές τιμές δεν είναι ακέραιοι αριθμοί ή οι αναμενόμενες τιμές του είναι εκτός των ορίων του ακεραίου τύπου.

Τύποι Πραγματικών Αριθμών

- real
- single
- Double
- extended
- Comp

Επιτρεπτές πράξεις πραγματικών αριθμών

+ πρόσθεση

- αφαίρεση

* πολλαπλασιασμός

/ διαίρεση (πηλίκο)

Αριθμητικές εκφράσεις είναι οι απεικονίσεις αριθμητικών παραστάσεων που μπορεί να περιέχουν σταθερές, μεταβλητές, συναρτήσεις, αριθμητικά σύμβολα και παρενθέσεις.

Στον πίνακα επίσης βλέπουμε την προτεραιότητα των μαθηματικών τελεστών

Χαμηλότερη	Υψηλότερη
+	*
-	/ div mod

Λογικός Τελεστής

Έχει δύο μόνο τιμές: **Αληθή (true), Ψευδή (false)**

Επιτρεπτές πράξεις:

Σύζευξη (and)

Διάζευξη (or)

Αποκλειστική διάζευξη (xor)

Άρνηση (not)

Λογικές εκφράσεις είναι οι απεικονίσεις παραστάσεων που μπορεί να περιέχουν σταθερές, μεταβλητές, συναρτήσεις, αριθμητικά σύμβολα και παρενθέσεις και μπορούν να πάρουν μια λογική τιμή (true ή false).

Μια λογική έκφραση παράγεται από δύο μεταβλητές ή σταθερές μέσω των σχεσιακών τελεστών.

Σχεσιακοί τελεστές

Περιγραφή	Pascal
Μεγαλύτερο από	>
Μικρότερο από	<
Μεγαλύτερο ή ίσο	>=

Μικρότερο ή ίσο	<=
Διάφορο	<>
Ανήκει	in

Προτεραιότητα τελεστών

Μεγαλύτερη προτεραιότητα έχουν οι πράξεις μέσα σε παρενθέσεις.

Όταν τα σύμβολα έχουν την ίδια προτεραιότητα τότε οι πράξεις εκτελούνται από αριστερά προς τα δεξιά.

Χαρακτήρας

Περιγράφει δεδομένα ενός χαρακτήρα

Π.χ.: 'A', 'B', '@', '!

Ο τύπος char είναι ένας διατεταγμένος τύπος ο οποίος περιλαμβάνει το σύνολο των χαρακτήρων του υπολογιστή μας.

Διάταξη:

0,1,2,3,...9

A,B,C,...Z, a,b,c,...z και ακολουθούν οι ελληνικοί χαρακτήρες

A,B,Γ,...Ω, α,β,γ,...ω

Αλφαριθμητικός (string)

Είναι μία σειρά από 255 το πολύ χαρακτήρες.

Π.χ.: 'Turbo Pascal', 'Bπ2', 'Ελένη', '1346', "

Δομή προγράμματος Pascal

Ένα πρόγραμμα Pascal αποτελείται από 3 τμήματα:

- **Τίτλος Προγράμματος**

Αρχίζει με τη λέξη-κλειδί **program** και ακολουθείται από ένα **όνομα** που χαρακτηρίζει όλο το πρόγραμμα.

- **Τμήμα δηλώσεων**

Δηλώνουμε όλα τα στοιχεία που πρόκειται να χρησιμοποιηθούν στο πρόγραμμα

Σταθερές

Τύπους

Μεταβλητές

Υποπρογράμματα

- **Τμήμα εντολών**

Περιέχει τις εκτελέσιμες προτάσεις του προγράμματος.

Αρχίζει με τη λέξη-κλειδί **begin** και τελειώνει με τη λέξη-κλειδί **end**.

Το τέλος των δηλώσεων και των προτάσεων δηλώνεται με το διαχωριστικό χαρακτήρα (;)

Στο τέλος του προγράμματος, δηλ. μετά το **end** βάζουμε τελεία.

Παράδειγμα

Εύρεση n!

Παραγοντικό του N στο X

Πριν: N: φυσικός αριθμός

Μετά: X = N!

```
Είσοδος:  
  N : Φυσικός αριθμός  
Εξοδος:  
X : Φυσικός αριθμός  
K : Φυσικός αριθμός  
  
X <- 1  
K <- N  
Όσο K > 0  
  X <- X * K  
  K <- K - 1  
Τέλος (όσο)
```

Pascal

```
(* Παραγοντικό του N *)  
function factorial(N : Integer) : Integer;  
var  
  Count, Result : Integer;  
begin  
  Count := N;  
  Result := 1;  
  While Count > 0 Do  
  begin  
    Result := Result * Count;  
    Count := Count - 1  
  end;  
  Factorial := Result  
end (* Factorial *);
```

Ψευδοκώδικας

Ο ψευδοκώδικας (pseudocode) επιτρέπει την περιγραφή των αλγορίθμων.

Στοιχεία εισόδου, εξόδου και μεταβλητές

Είσοδος:

πλευρά α, πλευρά β, πλευρά γ : Φυσικός αριθμός

Εξοδος:

εμβαδό : Φυσικός αριθμός

αριθμός πλακών : Φυσικός αριθμός

συνολικό βάρος : Πραγματικός αριθμός

είναι ισοσκελές : Λογική τιμή

Προϋποθέσεις και μετά-συνθήκες

Πριν: N Φυσικός αριθμός

Μετά: X = N!

Εκχωρήσεις

ύψος <- συν(κλίση) * πλευρά

μέσο <- ύψος / 2

Βρόχοι

Όσο υπόλοιπο χρημάτων > 0

δώσε χαρτονόμισμα

Τέλος (όσο)

Επανάλαβε

μείωσε αέρα

Όσο στροφές κινητήρα < 900

Για πάντα

έλεγξε παραμέτρους λειτουργίας εργοστασίου

Τέλος (για πάντα)

Αποφάσεις

Αν ζωές παίκτη > 0 Τότε

ξεκίνα νέο παιχνίδι

Αλλιώς

εμφάνισε "Game Over"

Τέλος (Αν)

Αν στροφές > 8700 Τότε

Διάκοψε την τροφοδοσία

Τέλος (Αν)

Αν ομάδα ΠΑΟ Τότε

χρώμα <- πράσινο

Αλλιώς Αν ομάδα Ολυμπιακός Τότε

χρώμα <- κόκκινο

Αλλιώς Αν ομάδα ΑΕΚ Τότε

χρώμα <- κίτρινο

Τέλος (Αν)

Διαδικασίες, συναρτήσεις και τάξεις

- Οι διαδικασίες (procedures) επιτρέπουν την οργάνωση των εντολών ενός προγράμματος που εκτελούν μια συγκεκριμένη λειτουργία σε μια ενότητα.
- Οι συναρτήσεις (functions) επιτρέπουν την οργάνωση των εντολών ενός προγράμματος που υπολογίζουν ένα συγκεκριμένο αποτέλεσμα σε μια ενότητα.
- Οι τάξεις (classes) επιτρέπουν την οργάνωση των στοιχείων και των εντολών που σχετίζονται με ένα αντικείμενο σε μια ενότητα.

Παράδειγμα συνάρτησης και διαδικασίας σε Pascal

```

program fun;

(* Παραγοντικό του N *)
function factorial(N : Integer) : Integer;
var
  Count, Result : Integer;
begin
  Count := N;
  Result := 1;
  While Count > 0 Do
  begin
    Result := Result * Count;
    Count := Count - 1
  end;
  Factorial := Result
end (* Factorial */;

procedure clear_screen;
const number_of_lines = 24;
var i : integer;

begin
  for i := 0 to number_of_lines do
    writeln("");
  end (*clear_screen*);

begin
  clear screen;
  writeln(factorial(5));
end.

```

Σύνοψη

Τελειώνοντας την ενότητα αυτή, έχει γίνει μια αναφορά και εισαγωγή στον προγραμματισμό, αναφέρθηκαν και αναλύθηκαν οι αλγόριθμοι και οι δομές δεδομένων. Η παρουσίαση των αλγορίθμων περιλαμβάνει ρητά διατυπωμένες αναλλοίωτες ιδιότητες μαζί με σχόλια για σύγχρονες τάσεις, όπως οι τεχνολογίες υλοποίησης. Για την περιγραφή και την εξήγηση των αλγορίθμων χρησιμοποιείται ψευδοκώδικας υψηλού επιπέδου, ενώ στη συνέχεια παρουσιάζονται περισσότερες

λεπτομέρειες για την αποδοτική υλοποίησή τους σε πραγματικές γλώσσες προγραμματισμού όπως η C++ και η Java. Επίσης έγινε αναφορά για τα δομικά στοιχεία προγράμματος και για τον σχεδιασμό και την υλοποίηση.

Ερωτήσεις ενότητας 3

A. Τι είναι γλώσσα προγραμματισμού;

1. Η γλώσσα προγραμματισμού είναι μια γραπτή, γλώσσα που καταλαβαίνει και διαχειρίζεται ένας υπολογιστής
2. Η γλώσσα προγραμματισμού είναι μια γραπτή, επίσημη γλώσσα που καθορίζει ένα σύνολο οδηγιών και κανόνων που χρησιμοποιούνται για τη δημιουργία προγραμμάτων (λογισμικό).
3. Η γλώσσα προγραμματισμού είναι γλώσσα που απαρτίζεται και σχεδιάζεται από προγραμματιστές .

B. Τι είναι γλώσσα μηχανής;

1. Η γλώσσα προγραμματισμού που περιλαμβάνει εντολές γραμμένες σε μορφή ακολουθιών bit εκτελέσιμες από την Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ).
2. Είναι μια γραπτή, γλώσσα που καταλαβαίνει και διαχειρίζεται ένας υπολογιστής

Γ. Τι γλώσσα προγραμματισμού χαμηλού επιπέδου;

1. Είναι μια γραπτή, γλώσσα που καταλαβαίνει και διαχειρίζεται ένας υπολογιστής
2. Η γλώσσα προγραμματισμού που περιλαμβάνει εντολές γραμμένες σε μορφή ακολουθιών bit εκτελέσιμες από την Κεντρική Μονάδα Επεξεργασίας (ΚΜΕ).
3. Η γλώσσα που είναι κοντά στην γλώσσα μηχανής γι' αυτό και ονομάζεται χαμηλού επιπέδου.

Δ. Τι είναι Μεταγλωττιστής (Compiler);

1. Μεταφράζει μια γλώσσα υψηλού επιπέδου σε γλώσσα επιπέδου μηχανής.
2. Μεταφράζει μια γλώσσα χαμηλού επιπέδου σε γλώσσα επιπέδου μηχανής.

3. Μεταφράζει μια γλώσσα υψηλού επιπέδου σε γλώσσα χαμηλού.

E. Τι είναι Διερμηνευτής (Interpreter);

1. Διερμηνευτής (Interpreter) εκτελεί άμεσα ένα πρόγραμμα σε γλώσσα υψηλού επιπέδου.

2. Διερμηνευτής (Interpreter) εκτελεί άμεσα ένα πρόγραμμα σε γλώσσα χαμηλού επιπέδου.

3. Διερμηνευτής (Interpreter) εκτελεί άμεσα ένα πρόγραμμα σε γλώσσα γλώσσα μηχανής.

Z. Τι είναι αλγόριθμος;

1. Ονομάζεται μία σειρά από δεδομένα που έχουν αρχή και τέλος, έχουν ως σκοπό την επίλυση κάποιου προβλήματος

2. Ονομάζεται μία σειρά από εντολές που έχουν αρχή και τέλος, είναι σαφείς και έχουν ως σκοπό την επίλυση κάποιου προβλήματος.

H. Τι είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου;

1. Η γλώσσα που είναι κοντά στην γλώσσα μηχανής γι' αυτό και ονομάζεται χαμηλού επιπέδου.

2. Είναι μια γραπτή, γλώσσα που καταλαβαίνει και διαχειρίζεται ένας υπολογιστής

3. Η γλώσσα υψηλού επιπέδου είναι εκείνη που είναι περίπου πιο κοντά στην ανθρώπινη γλώσσα από τη γλώσσα μηχανής.

Θ. Επιλέξτε ποιες είναι οι κύριες γλώσσες υψηλού επιπέδου;

1. JavaScript – Python

2. PHP

3. Όλες οι παραπάνω

I. Κάθε δομή δεδομένων έχει το δικό της σύνολο μεθόδων για την εκτέλεση λειτουργιών όπως επιλέξτε ποιες είναι ;

1. Εισαγωγή ή διαγραφή στοιχείων.

2. Αναζήτηση και εντοπισμός στοιχείων.
3. Ταξινόμηση (ταξινόμηση) στοιχείων σύμφωνα με κάποια καθορισμένη σειρά.
4. Όλα τα παραπάνω

Κ. Τι διαφορετικό έχει το Instagram σε σχέση με τους αλγόριθμους ;

1. Το Instagram είναι μια περίεργη εφαρμογή, καθώς έχει διαφορετικούς αλγόριθμους που επηρεάζουν ο ένας τον άλλον.
2. Το Instagram είναι μια εφαρμογή, που δεν αλληλεπιδρά με αλγόριθμους.

Άσκηση

Να γράψετε αλγόριθμο που να διαβάζει 2 αριθμούς και να εμφανίζει.

- α. Το άθροισμα τους
- β. Το γινόμενο τους.
- γ. Το μέσο όρο τους

Δώσε 2 αριθμούς:

2

4

Το άθροισμά τους είναι 6

Το γινόμενο τους είναι 8

Ο μέσος όρος τους είναι 3

Βιβλιογραφία

Αριστέιδης Σ. Μπούρας, (2017) Αλγοριθμική και προγραμματισμός υπολογιστών σε Python, Οδηγός για νέους και έμπειρους προγραμματιστές.

Bjarne Stroustrup (2014) Η γλώσσα προγραμματισμού C++

Eric Matthes επιστημονική επιμέλεια Χρήστος Καλλονιάτης (2020) Η Γλώσσα Προγραμματισμού PYTHON

Mehlhorn, Kurt Αλγόριθμοι και δομές δεδομένων -Τα βασικά εργαλεία

Εκπαιδευτική Υποεπάρκεια: 4 Λειτουργικά συστήματα και πληροφοριακά συστήματα

Σκοπός

Η ενότητα εισάγει τον εκπαιδευόμενο σε βασικά θέματα της αρχιτεκτονικής των λειτουργικών συστημάτων. Ειδικότερα όπως οι διεργασίες, η διαχείριση μνήμης κ.λ.π. Καθώς και τα χαρακτηριστικά ενός συστήματος πολλών χρηστών. Επίσης εισάγει σε θέματα της αρχιτεκτονικής των λειτουργικών συστημάτων και στην έννοια των πληροφοριακών συστημάτων.

Προσδοκώμενα Αποτελέσματα

Με το πέρας της ενότητας εκπαιδευόμενος θα έχει την ευχέρεια να περιγράψει ένα λειτουργικό σύστημα, να γνωρίζει τις βασικές αρχές λειτουργίας ενός λειτουργικού Συστήματος ώστε να εμβαθύνει σε περαιτέρω γνώσεις σχεδίασης λειτουργικών Συστημάτων. Θα έχει διευρύνει τις γνώσεις του και τις ικανότητές του σε σχέση με τους τρόπους χρήσης υπολογιστών πέρα από την κλασσική χρήση μέσω των παραθυρικών περιβαλλόντων.

Τυπική θέση του λειτουργικού συστήματος σε ένα υπολογιστικό σύστημα



Έννοιες – Κλειδιά

Λειτουργικό σύστημα (αγγλικά: Operating System ή OS) ονομάζεται στην επιστήμη της πληροφορικής το λογισμικό του υπολογιστή που είναι υπεύθυνο για τη διαχείριση και τον συντονισμό των εργασιών, καθώς και την κατανομή των διαθέσιμων πόρων. Είναι το πρόγραμμα το οποίο εκτελείται πρώτο όταν ανοίξετε

τον υπολογιστή σας και το οποίο διαχειρίζεται όλα τα προγράμματα που τρέχετε στον υπολογιστή σας (εφαρμογές).

Παρακολουθητής (monitor) Η χρήση ενός λογισμικού που λέγεται παρακολουθητής (monitor), είναι ένα ειδικό λογισμικό που ελέγχει τα προγράμματα που τρέχουν και ομαδοποιεί τις εργασίες. Κάθε πρόγραμμα είναι φτιαγμένο έτσι ώστε όταν ολοκληρώνεται να επιστρέφει στο monitor για να φορτωθεί άλλο πρόγραμμα

Πυρήνας (Kernel) το μόνο πρόγραμμα που τρέχει συνέχεια, όσο βρίσκεται σε λειτουργία το ΥΣ. Είναι το κυριότερο τμήμα ενός ΛΣ. Το τμήμα αυτό φορτώνεται πρώτο στην κύρια μνήμη και εκτελείται συνεχώς σε όλη τη διάρκεια λειτουργίας του υπολογιστή.

Σύστημα Αρχείων (File System). Είναι το τμήμα του ΛΣ το οποίο διαχειρίζεται τα αρχεία (ονοματοδοσία, καταχώριση, ανάκτηση κ.λπ.) και φροντίζει επίσης για τη διάθεσή τους στους χρήστες,

Διεπαφή χρήστη (user interface). Είναι το τμήμα που αναλαμβάνει να δέχεται και να δίνει στο σύστημα του υπολογιστή τα αιτήματα (εντολές) του χρήστη και επίσης να μεταφέρει στο χρήστη μηνύματα από το σύστημα.

Διεργασία (process) είναι ένας όρος της πληροφορικής ο οποίος περιγράφει το στιγμιότυπο ενός προγράμματος που εκτελείται σε έναν υπολογιστή.

MIS. Τα συστήματα πληροφόρησης της διοίκησης, MIS (Management Information Systems), αναφέρονται σε πληροφοριακά συστήματα τα οποία παρέχουν σε διευθυντές εργαλεία για την οργάνωση, την παρακολούθηση και την αξιολόγηση των τμημάτων τους.

Εκπαιδευτική Υποενότητα : 4.1 Εισαγωγικές έννοιες

Ο όρος Λειτουργικό Σύστημα (Operating System), εκφράζει το σύνολο των προγραμμάτων ενός υπολογιστικού συστήματος το οποίο ελέγχει, διαχειρίζεται και συντονίζει τη λειτουργία των επιμέρους τμημάτων του. Τα προγράμματα του Λειτουργικού Συστήματος ελέγχουν τη λειτουργία των προγραμμάτων των χρηστών, παρέχοντας τους απαιτούμενους πόρους (resources) και εποπτεύοντας την αλληλοδιαδοχή τους.

Στόχοι ενός ΛΣ:

- Εκτέλεση προγραμμάτων χρηστών
- Ευκολία χρήσης του ΥΣ
- Χρήση του υλικού και των περιφερειακών του ΥΣ με αποτελεσματικό / αποδοτικό τρόπο
- Προστασία των προγραμμάτων και δεδομένων των διαφόρων χρηστών του ΥΣ

Η ύπαρξη πολλών προγραμμάτων στους σύγχρονους υπολογιστές δικαιολογείται από το γεγονός ότι τα ΛΣ σήμερα υποστηρίζουν τη σύνδεση πολλών χρηστών στο ίδιο μηχάνημα. Το σημαντικότερο πρόγραμμα του Λειτουργικού Συστήματος είναι ο επόπτης (supervisor). Ο Επόπτης διευθύνει το Λειτουργικό Σύστημα, και παραμένει μόνιμα στη μνήμη όσο ο υπολογιστής βρίσκεται σε λειτουργία. Τα υπόλοιπα προγράμματα του Λειτουργικού Συστήματος δε βρίσκονται συνέχεια στην κύρια μνήμη αλλά είναι αποθηκευμένα στη βοηθητική μνήμη. Ο επόπτης ελέγχει το Λειτουργικό Σύστημα και προωθεί στη μνήμη τα απαιτούμενα προγράμματα του Λειτουργικού για όλες τις επεξεργασίες.

Το Λειτουργικό Σύστημα βελτιστοποιεί την απόδοση της λειτουργίας του υπολογιστή και ενεργεί ως διασύνδεση (interface) χρήστη - μηχανής. Με τον τρόπο αυτό δημιουργείται ένα περιβάλλον συνεργασίας ανθρώπου και υπολογιστή, όπου ο προγραμματιστής μπορεί να δημιουργεί και να καλεί τα προγράμματα των εφαρμογών του, και ο υπολογιστής να τα εκτελεί.

Εξέλιξη λειτουργικών συστημάτων

Το λειτουργικό σύστημα αποτελεί τη διεπαφή μεταξύ των εφαρμογών των χρηστών και του υλικού, υλοποιώντας ένα είδος ιδεατής μηχανής (virtual machine) που είναι ευκολότερη στον προγραμματισμό σε σχέση με το υλικό. Ένα λειτουργικό σύστημα είναι ένα πρόγραμμα που ελέγχει την εκτέλεση των προγραμμάτων εφαρμογών και ενεργεί ως ενδιάμεσος μεταξύ εφαρμογών και υλικού μέρους του υπολογιστή.

Αποτελεί το λογισμικό που τοποθετείται στην κορυφή του υλικού ώστε να διαχειρίζεται όλα τα συστατικά μέρη του συστήματος και να προσφέρει στο χρήστη τη διεπαφή (interface) ή μια ιδεατή μηχανή, που είναι πιο εύκολη στην κατανόηση και τον προγραμματισμό. Το λειτουργικό σύστημα καθορίζει ένα πλαίσιο για τους χρήστες και τα προγράμματά τους ώστε να συνυπάρχουν, να συνεργάζονται και να λειτουργούν ταυτόχρονα και αποδοτικά, υποστηρίζοντας:



- Ταυτόχρονη εκτέλεση και αλληλεπίδραση πολλών προγραμμάτων των χρηστών
- Διαμοιραζόμενες εφαρμογές που καλύπτουν συνήθεις απαιτούμενες διευκολύνσεις
- Μηχανισμούς διαμοίρασης και συνδυασμού συστατικών λογισμικού
- Πολιτικές για ασφαλή και δίκαιη διαμοίραση των πόρων
 - ο φυσικών πόρων (π.χ. χρόνος της CPU και χώρος αποθήκευσης)
 - ο λογικών πόρων (π.χ. αρχεία δεδομένων, προγράμματα)

Ιστορική Εξέλιξη των ΛΣ

Πρώτο στάδιο εξέλιξης

Το πρώτο στάδιο εξέλιξης στην ιστορία των λειτουργικών συστημάτων εκτείνεται από τις αρχές της δεκαετίας 1940 έως τα μέσα της δεκαετίας 1950 και περιλαμβάνει την σειριακή επεξεργασία. Τα βασικά προβλήματα του σταδίου αυτού αφορούν:

Προβλήματα :

- **Δρομολόγηση**

- **Χρόνος**

Δρομολόγηση εργασιών – προγραμμάτων λόγω της σπατάλης σε υπολογιστικό χρόνο, αφού τα προγράμματα που έμεναν στη μέση της εκτέλεσής του λόγω του μεγέθους του μπλοκ χρόνου

Χρόνος εγκατάστασης ένα μοναδικό πρόγραμμα, η εργασία, φόρτωνε τον μεταγλωττιστή και το πρόγραμμα σε γλώσσα υψηλού επιπέδου στη μνήμη, αποθήκευε τον αντικειμενικό κώδικα (object program) και στη συνέχεια το φόρτωνε και το συνέδεε με τις λοιπές συναρτήσεις.

Δεύτερο στάδιο εξέλιξης

Το δεύτερο στάδιο εξέλιξης (αρχές–μέσα της δεκαετίας 1960) περιλαμβάνει τα απλά μαζικά συστήματα επεξεργασίας (batch process). Ο λόγος ανάπτυξης τους ήταν η μείωση του νεκρού χρόνου στις πολύ ακριβές υπολογιστικές μηχανές της εποχής και βασίζονται στην κεντρική ιδέα της χρήση ενός λογισμικού που λέγεται παρακολουθητής (monitor). Ο χρήστης υποβάλλει την εργασία του σε ένα χειριστή, ο οποίος τις βάζει όλες μαζί σε μια συσκευή εισόδου για να χρησιμοποιηθεί από τον παρακολουθητή, ενώ κάθε πρόγραμμα είναι φτιαγμένο, έτσι ώστε όταν ολοκληρώνεται να επιστρέφει στο monitor για να φορτωθεί άλλο πρόγραμμα. Ο παρακολουθητής (monitor) είναι ένα ειδικό λογισμικό που ελέγχει τα προγράμματα που τρέχουν και ομαδοποιεί τις εργασίες. Επίσης, ο παρακολουθητής παραμένει

στην κεντρική μνήμη και είναι πάντοτε διαθέσιμος ενώ μόλις ένα πρόγραμμα ολοκληρωθεί ο έλεγχος μεταφέρεται σε αυτόν. Η υποβολή εργασιών στα συστήματα επεξεργασίας του δευτέρου σταδίου εξέλιξης πραγματοποιείται με μια ειδική γλώσσα ελέγχου εργασιών, την Job Control Language (JCL), η οποία παρέχει εντολές ελέγχου στον παρακολουθητή, επιτρέπει τον καθορισμό των μεταγλωττιστών που θα χρησιμοποιηθούν αλλά και των δεδομένων που απαιτούνται από τις εργασίες.

Τα χαρακτηριστικά του υλικού μέρους που είναι επιθυμητά στα απλά μαζικά συστήματα επεξεργασίας είναι τα ακόλουθα:

- Προστασία μνήμης: το πρόγραμμα χρήστη που εκτελεί ο παρακολουθητής δεν πρέπει να τροποποιεί την περιοχή μνήμης όπου υπάρχει ο παρακολουθητής.
- Χρονομετρητής: πρέπει να αποφεύγεται η μονοπώληση του συστήματος από μια μοναδική εργασία (π.χ. κάρτες σε προγράμματα FORTRAN).
- Προνομιούχες εντολές: εκτελούνται μόνον από τον παρακολουθητή, κυρίως για διαδικασίες I/O
- Διακοπές: μηχανισμοί του ΛΣ για παραχώρηση και ανάκτηση του ελέγχου του συστήματος

Τρίτο στάδιο εξέλιξης

Το τρίτο στάδιο εξέλιξης (1965-1980) περιλαμβάνει τα πολυπρογραμματιζόμενα μαζικά συστήματα με βασικό λόγο ανάπτυξης το γεγονός ότι ο επεξεργαστής παραμένει ανενεργός λόγω διαφοράς ταχύτητας με τις συσκευές I/O. Τα απλά μαζικά συστήματα επεξεργασίας χαρακτηρίζονταν από τον Μονοπρογραμματισμό, σύμφωνα με τον οποίον όταν η εκτελούμενη εργασία διέκοπτε, αναμένοντας να ολοκληρωθεί μια διεργασία I/O, η CPU έμενε ανενεργή. Ενώ σε επιστημονικές εφαρμογές η σπατάλη χρόνου είναι ελάχιστη αφού διαδικασίες I/O συμβαίνουν σπάνια, ενώ σε εμπορικές εφαρμογές η σπατάλη είναι 80-90%.

Αντίθετα, στην περίπτωση του Πολυπρογραμματισμού έχουμε διαμερισμό της μνήμης σε τμήματα ώστε κάθε διαφορετική διεργασία να καταλαμβάνει άλλο

τμήμα μνήμης. Όταν μια διεργασία περιμένει μια I/O, κάποια άλλη χρησιμοποιεί την CPU. Αν μπορούσαν να παραμείνουν στη μνήμη πολλές διεργασίες ταυτόχρονα, τότε το ποσοστό χρήσης της CPU έφθανε και το 100%. Η βασική απαίτηση για την υποστήριξη πολυπρογραμματισμού είναι η προστασία της μνήμης από επικαλύψεις, ένας μηχανισμός που ήταν διαθέσιμος μέσω του υλικού.

Στην περίπτωση του πολυπρογραμματισμού, τα μεγέθη που χαρακτηρίζουν την απόδοση ενός λειτουργικού συστήματος είναι η μέση χρησιμοποίηση των πόρων, η ρυθμοαπόδοση (throughput) και ο χρόνος απόκρισης των εργασιών. Τα πολυπρογραμματιζόμενα μαζικά συστήματα βασίζονται σε συγκεκριμένα χαρακτηριστικά του υλικού μέρους του Η/Υ (διακοπές, I/O, DMA), στη διαχείριση μνήμης και στους αλγόριθμους δρομολόγησης. Τα συστήματα διαμοιραζόμενου χρόνου προέκυψαν από τις διαμαρτυρίες όσων περίμεναν ώρες ή ημέρες για να πάρουν αποτελέσματα από τα πολυπρογραμματιζόμενα μαζικά συστήματα.

Στην περίπτωση τους, κάθε χρήστης συνδέεται μέσω τερματικού και η CPU εξυπηρετεί εκ περιτροπής κάθε πρόγραμμα χρήστη με ένα σύντομο καταιγισμό (burst) ή ένα κβάντα (quantum) υπολογισμού. Με τον τρόπο αυτό, ο ΗΥ προσφέρει μια διαλογική εξυπηρέτηση με την τεχνική του καταμερισμού χρόνου και εκμεταλλεύεται τον σχετικά βραδύ χρόνο της ανθρώπινης αντίδρασης. Με τον τρόπο αυτό, ο ΗΥ προσφέρει μια διαλογική εξυπηρέτηση με την τεχνική του καταμερισμού χρόνου και εκμεταλλεύεται τον σχετικά βραδύ χρόνο της ανθρώπινης αντίδρασης. Συγκρίνοντας τον πολυπρογραμματισμό με τη διαμοίραση χρόνου, διακρίνεται, στην περίπτωση του πρώτου, η μεγιστοποίηση της χρησιμοποίησης επεξεργαστή και την εισαγωγή οδηγιών προς το λειτουργικό σύστημα μέσω γλώσσας ελέγχου εργασιών. Αντίθετα στην διαμοίραση χρόνου η ελαχιστοποίηση του χρόνου απόκρισης και τις εντολές προς το λειτουργικό σύστημα που εισάγονται άμεσα από το τερματικό.

Τέταρτο στάδιο εξέλιξης

Η τέταρτη γενιά των λειτουργικών συστημάτων (1980 – 1990) επηρεάζεται από την εμφάνιση των ολοκληρωμένων κυκλωμάτων τύπου LSI. Την περίοδο αυτή τα λειτουργικά συστήματα γίνονται φιλικά προς το χρήστη ενώ κάνουν την εμφάνισή

τους τα λειτουργικά συστήματα δικτύων, όπου κάθε ΗΥ τρέχει το δικό του ΛΣ, και τα κατανεμημένα (distributed) λειτουργικά συστήματα. Τα τελευταία εμφανίζονται ως παραδοσιακά συστήματα ενός επεξεργαστή, στα οποία οι χρήστες δεν ενδιαφέρονται που εκτελούνται τα προγράμματά τους ή που βρίσκονται τα αρχεία τους. Τα κατανεμημένα λειτουργικά συστήματα επιτρέπουν στα προγράμματα να εκτελούνται σε διαφορετικούς επεξεργαστές την ίδια χρονική στιγμή και απαιτούν πολύπλοκους αλγόριθμους χρονοδρομολόγησης (scheduling). Παράλληλα κάνουν την εμφάνιση τους και τα συστήματα συναλλαγής πραγματικού χρόνου (π.χ. σύστημα κρατήσεων αεροπορικών εταιρειών).

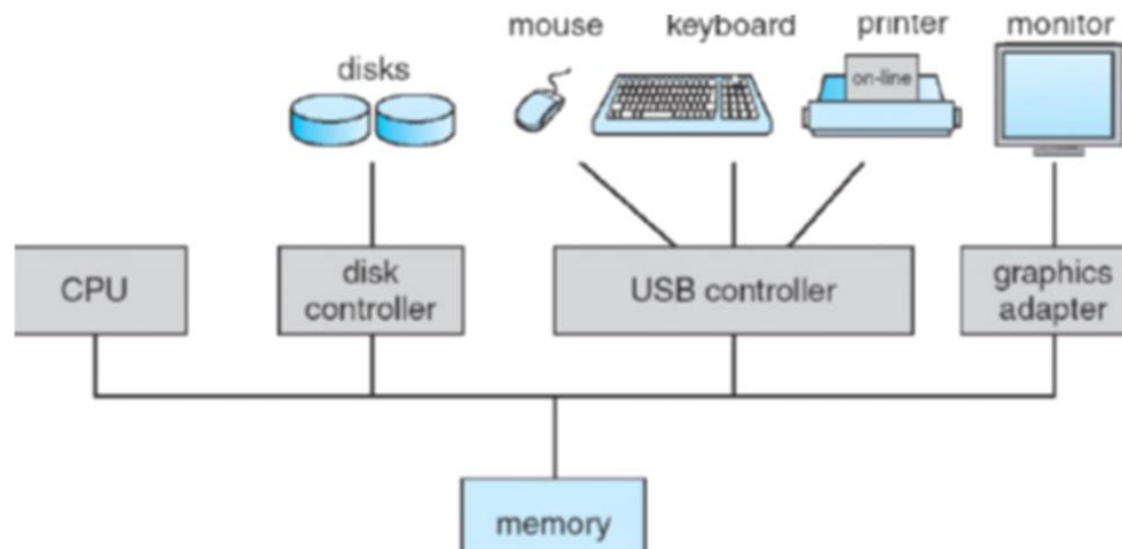
Πέμπτο στάδιο εξέλιξης

Το Πέμπτο στάδιο εξέλιξης των λειτουργικών συστημάτων περιλαμβάνει τις σύγχρονες εξελίξεις τους, οι οποίες μπορούν να διαχωριστούν σε αυτές της δεκαετίας του 1990 και αυτών που διανύουμε. Στην δεκαετία του 1990 έχουμε την αλατώδη αύξηση της απόδοσης του υλικού (MIPS), ενώ η δημιουργία του παγκοσμίου ιστού αύξησε την κατανεμημένη επεξεργασία και οδήγησε στην ανάγκη ενσωμάτωσης διαδικτυακών διεργασιών. Παράλληλα έχουμε την καθιέρωση της αντικειμενοστραφούς τεχνολογίας, την διάδοση και ανάπτυξη της τεχνολογίας ανοικτού κώδικα με σημαντικότερο γεγονός την εμφάνιση του LINUX.

Κατά την τρέχουσα δεκαετία έχει εμφανιστεί το middleware, το οποίο αποτελεί κατάλληλο λογισμικό που συνδέει δύο ξεχωριστές εφαρμογές, που συχνά βρίσκονται σε δίκτυο. Επίσης, κυριαρχούν εφαρμογές (web services) που δημοσιεύονται στο Internet και χρησιμοποιούνται από χρήστες μέσω συνδέσεων υψηλών ταχυτήτων ενώ κάνουν την εμφάνιση τους βελτιωμένες αρχιτεκτονικές δικτύων. Σημειώνεται σημαντική πρόοδος και αύξηση της παράλληλης επεξεργασίας ενώ χρησιμοποιούνται λειτουργικά συστήματα τύπου POSIX (Portable Operating System Interface). Τέλος, εμφανίζεται η υπολογιστική δυνατότητα σε φορητές συσκευές.

Η ταχύτερη πλέον εξέλιξη της τεχνολογίας τόσο στο υλικό όσο και στο λογισμικό και η ανάπτυξη των δικτύων οδηγούν τις εξελίξεις. Προσωπικοί υπολογιστές αρχικά και προσωπικές έξυπνες συσκευές τα τελευταία χρόνια αποκτούν τεράστιες

υπολογιστικές ικανότητες σε σύγκριση με τους υπολογιστές της 4ης γενεάς. Τα λειτουργικά συστήματα εξελίσσονται και ενσωματώνουν τα περισσότερα από τα γνωρίσματα που έχουν σήμερα (φιλικότητα, πολυπρογραμματισμό, πολυχρησία, δικτύωση, ασφάλεια). Συστήματα τύπου «πελάτη-εξυπηρετητή» είναι βασικά μοντέλα που ακολουθούνται με κατάληξη τα σημερινά συστήματα νέφους (cloud computing), με χρήση του διαδικτύου.



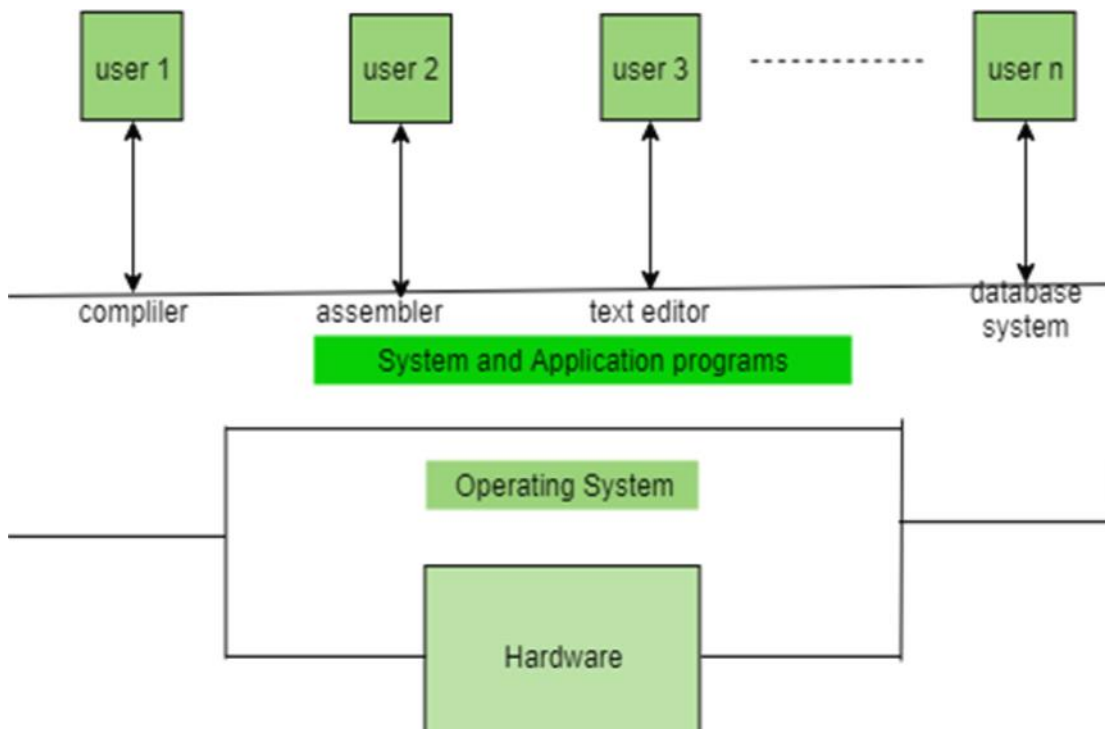
Εκπαιδευτική Υποενότητα : 4.2 Λειτουργικά συστήματα

Βασικές αρμοδιότητες και λειτουργίες του Λειτουργικού Συστήματος

Οι βασικές "αρμοδιότητες" του λειτουργικού συστήματος είναι :

- Να λειτουργεί ως ενδιάμεσος (Διεπαφή ή Interface) ανάμεσα στον άνθρωπο και στη μηχανή.
- Να διαχειρίζεται τις δυνατότητες και τους πόρους (resources) του συστήματος υπολογιστή έτσι, ώστε να παράγεται χρήσιμο έργο.
- Να μεταφέρει εντολές ή/και απαιτήσεις του χρήστη στον Η/Υ.
- Να δίνει χρήσιμες πληροφορίες για την κατάσταση του συστήματος, μεταφέρει μηνύματα του Η/Υ προς το χρήστη για λάθη ή προβλήματα που εμφανίζονται.
- Να διαχειρίζεται την Κεντρική Μνήμη (RAM) του συστήματος.

- Να ενεργοποιεί και δίνει οδηγίες στην ΚΜΕ. κατανέμοντας το χρόνο λειτουργίας της στους χρήστες και στα διάφορα προγράμματα που εκτελούνται.
- Να διαχειρίζεται τις συσκευές εισόδου και εξόδου ελέγχοντας τη ροή των δεδομένων (είσοδος) και την έξοδο των πληροφοριών (έξοδος).
- Να οργανώνει και διαχειρίζεται τα αρχεία του συστήματος μέσω του συστήματος αρχείων.
- Να ελέγχει την εκτέλεση των προγραμμάτων των χρηστών.
- Να εφαρμόζει μηχανισμούς που βελτιώνουν την ασφάλεια του υπολογιστή από διάφορους κινδύνους.



Ένα ΛΣ έχει δυο βασικούς σκοπούς:

Ο πρώτος βασικός σκοπός ενός λειτουργικού συστήματος είναι η διευκόλυνση του χρήστη στην επικοινωνία του με τον υπολογιστή. Η διευκόλυνση αυτή επιτυγχάνεται με τη δημιουργία ενός περιβάλλοντος επικοινωνίας (φλοιού). Αυτό μεσολαβεί ανάμεσα στο χρήστη και τον υπολογιστή και έτσι, δεν είναι απαραίτητο ο χρήστης να γνωρίζει λεπτομέρειες του υλικού για να κάνει χρήση του. Ο χρήστης, λοιπόν, μπορεί να εστιάσει σε αυτό που επιθυμεί να γίνει με τις εντολές του και τα

προγράμματα εφαρμογών, τα οποία χρησιμοποιεί για να λύσει ένα πρόβλημά του. Το ΛΣ είναι στη συνέχεια υπεύθυνο για την μεταβίβαση στο υλικό των ενεργειών, που πρέπει να γίνουν. Για παράδειγμα, ο χρήστης δίνει μια εντολή για την αντιγραφή ενός αρχείου από το σκληρό δίσκο σε ένα flash disk. Το ΛΣ:

- αναγνωρίζει την εντολή του χρήστη
- αναζητά το αρχείο στο δίσκο
- εντοπίζει τη θέση του
- ελέγχει αν υπάρχει ελεύθερος χώρος στο flash disk
- αρχίζει τη μεταφορά του αρχείου από το ένα μέσο στο άλλο

Ο δεύτερος σκοπός ενός λειτουργικού συστήματος είναι η αξιόπιστη και η αποδοτική λειτουργία του συστήματος του υπολογιστή και η καλύτερη αξιοποίηση των πόρων του (ΚΜΕ, μνήμη, δίσκοι, περιφερειακές συσκευές). Η κατανομή αυτών των πόρων γίνεται με βάση κριτήρια, τα οποία εξασφαλίζουν ίση χρήση από όλους τους ενδιαφερόμενους (χρήστες και εφαρμογές) και αποτελεσματικότητα στην αξιοποίησή τους.

Πυρήνας (Kernel) του λειτουργικού συστήματος

Ο Πυρήνας του λειτουργικού συστήματος είναι ένα σύνθετο πρόγραμμα, το οποίο διαχειρίζεται αιτήματα χρήσης συσκευών εισόδου/εξόδου από τις εφαρμογές και ελέγχει την κατανομή της μνήμης και της κεντρικής μονάδας επεξεργασίας (ΚΜΕ) στα προγράμματα που εκτελούνται. Αποτελεί το πιο χαμηλό (κοντά στη μηχανή) επίπεδο του ΛΣ και είναι το πρόγραμμα, το οποίο εκκινεί άμεσα με το άνοιγμα του υπολογιστή και τερματίζει τελευταίο. Ο πυρήνας χειρίζεται αυτό που ονομάζεται κλήσεις συστήματος. Αυτές είναι αιτήματα από τις εφαρμογές για χρήση του υλικού, δημιουργία νέων διεργασιών, που θα εκτελούνται στην ΚΜΕ, και διαχείριση της μνήμης του συστήματος. Η επικοινωνία με το υλικό, άλλα και με τις εφαρμογές, πραγματοποιείται δια μέσου ενός συστήματος διακοπών, που είναι ένας βασικός μηχανισμός του ΛΣ. Μόλις δημιουργείται μια διακοπή, η οποία αντιστοιχεί σε

κάποιο αίτημα, καλείται το αντίστοιχο πρόγραμμα για να διαχειριστεί το αίτημα αυτό.

Η επικοινωνία του χρήστη ή Διεπαφή χρήστη (User Interface)

Ότι ο πρώτος βασικός σκοπός του ΛΣ είναι η διευκόλυνση του χρήστη στην επικοινωνία με τον υπολογιστή. Αυτός ο ρόλος καλύπτεται από τη λεγόμενη διεπαφή χρήστη (user interface), η οποία είναι ένας μηχανισμός που επιτρέπει στον χρήστη:

- να χρησιμοποιεί αποδοτικά το σύστημα αρχείων,
- να εκκινεί και να διαχειρίζεται τις εφαρμογές που εκτελούνται στον υπολογιστή και
- να έχει πληροφορίες για τη λειτουργία των μονάδων του υπολογιστή με δυνατότητα να προβεί σε ρυθμίσεις.

Η διεπαφή χρήστη μπορεί να πραγματοποιηθεί με δύο κυρίως τρόπους:

- Μέσω ενός διερμηνευτή εντολών.
- Μέσω ενός γραφικού περιβάλλοντος επικοινωνίας.

Διερμηνευτής εντολών

Ο διερμηνευτής εντολών είναι ο πρώτος ιστορικά μηχανισμός επικοινωνίας με τον υπολογιστή. Ο χρήστης χρησιμοποιεί ένα προκαθορισμένο σύνολο εντολών, τις οποίες μπορεί να δώσει στον υπολογιστή, δια μέσου μιας τερματικής συσκευής χαρακτήρων (ή ενός αναγνώστη καρτών στις πολύ παλιές εποχές). Αυτές οι εντολές μπορούν να παραμετροποιηθούν ως προς τη λειτουργία τους. Επίσης, ανάλογα με το ΛΣ, παρέχεται η δυνατότητα συγγραφής σεναρίων.

Σε αυτή την περίπτωση μπορεί να αυτοματοποιηθούν πολλές εργασίες που απαιτούν τη χρήση πολλών από αυτές τις εντολές. Λόγω των πολύ ισχυρών

δυνατοτήτων και της οικονομίας χρόνου, που είναι δυνατόν να υπάρχει με τη χρήση αυτών των εντολών, η χρήση του διερμηνευτή εντολών συνεχίζει να υπάρχει ακόμα και στα σύγχρονα λειτουργικά συστήματα. Αυτό γίνεται με τη χρήση ενός ειδικού προγράμματος εξομοίωσης τερματικού. Παρά το ότι ένας διερμηνευτής εντολών δίνει πολλές δυνατότητες μαζικής κυρίως επεξεργασίας αρχείων, εντούτοις είναι δύσκολη η χρήση του, διότι απαιτεί εξοικείωση με τις εντολές και τις παραμέτρους του. Αποτελεί σημαντικό εργαλείο για διαχειριστές συστημάτων και πεπειραμένους χρήστες και μπορεί να εξοικονομήσει χρόνο και να αυτοματοποιήσει διαδικασίες. Στα δε συστήματα UNIX ή στα βασιζόμενα σε αυτό (π.χ LINUX) υπάρχουν περισσότεροι του ενός διερμηνευτές εντολών με πολύ προχωρημένες δυνατότητες προγραμματισμού (sh, csh, bash).



Γραφικό περιβάλλον επικοινωνίας

Το γραφικό περιβάλλον επικοινωνίας (Graphical User Interface, GUI) εμφανίζεται από τα μέσα της δεκαετίας του 80 και *έπειτα και έχει γίνει πλέον ο βασικός μηχανισμός επικοινωνίας με τον υπολογιστή.*

Βασικά στοιχεία του είναι τα εξής:

- Η χρήση περιβάλλοντος γραφικών και όχι απλής γραμμής κειμένου.

- Η χρήση «παραθύρων» που είναι ορθογώνιες περιοχές στην οθόνη εντός των οποίων μπορούν να εκτελούνται εφαρμογές. Τα παράθυρα αυτά έχουν έναν καθορισμένο τρόπο εμφάνισης και χειρισμού (τίτλος παραθύρου, χρώματα πλαισίου, μεγιστοποίηση, ελαχιστοποίηση, κλείσιμο).
- Ύπαρξη μιας «επιφάνειας εργασίας» στην οποία μπορούν να υπάρχουν τα παράθυρα των εκτελούμενων εφαρμογών καθώς και εικονίδια που αντιστοιχούν σε εφαρμογές, αρχεία και φακέλους.
- Πλήρης έλεγχος μέσω συσκευών κατάδειξης (π.χ ποντίκι, light pen, touch pad, οθόνη αφής). Ο χρήστης μπορεί να εκκινήσει και να τερματίσει εφαρμογές, να αλλάξει θέση και μέγεθος στα παράθυρα και γενικά να προκαλεί «γεγονότα» (π.χ πάτημα πλήκτρου ποντικιού, τοποθέτηση δείκτη ποντικιού σε μια περιοχή, κ.ά) τα οποία διαχειρίζονται από το ΛΣ και οδηγούνται προς τις αντίστοιχες εφαρμογές για έλεγχο και ανταπόκριση.
- Δυνατότητα ύπαρξης μιας περιοχής όπου υπάρχει ένα «μενού» των εφαρμογών.
- Δυνατότητα ύπαρξης μιας περιοχής όπου εμφανίζονται (σε μορφή εικονιδίων) οι εφαρμογές που εκτελούνται και διάφορες άλλες πληροφορίες (π.χ ώρα)

Για τη χρήση όλων αυτών των δυνατοτήτων από τις εφαρμογές το ΛΣ παρέχει μια σειρά κλήσεων οι οποίες είναι διαθέσιμες δια μέσου βιβλιοθηκών λογισμικού. Το γραφικό περιβάλλον επικοινωνίας μπορεί να είναι αναπόσπαστο κομμάτι του ΛΣ (π.χ όπως συμβαίνει σε όλα τα Microsoft Windows) ή να επιλέγεται/εγκαθίσταται από τον χρήστη (π.χ λειτουργικά Linux με δυνατότητα χρήσης του KDE, του Gnome, κ.ά).

Κατηγορίες λειτουργικών συστημάτων

Σε μια προσπάθεια κατηγοριοποίησης των λειτουργικών συστημάτων θα μπορούσαμε να κάνουμε διακρίσεις ανάλογα με τα παρακάτω:

- Τύπο της επεξεργασίας πληροφοριών που υποστηρίζουν

- Υποστήριξη ενός ή πολλών χρηστών
- Ανοικτό η κλειστό λογισμικό

Εννοείται ότι ένα Λειτουργικό μπορεί να κατατάσσεται ταυτόχρονα σε διάφορες κατηγορίες ανάλογα με τα συγκεκριμένα γνωρίσματα του.

Κατάταξη με τύπο επεξεργασίας πληροφοριών. Ανάλογα με τον τύπο επεξεργασίας διακρίνουμε τις κατηγορίες που θα αναφερθούν παρακάτω και οι οποίες διαφοροποιούνται βασικά στο χρόνο απόκρισης και στη γεωγραφική διασπορά των μονάδων. Θα πρέπει να τονιστεί, ότι αυτή η κατηγοριοποίηση συμπεριλαμβάνει κάποιους τύπους λειτουργικών οι οποίοι ουσιαστικά δεν υφίστανται πλέον λόγω:

- της τεράστιας εξέλιξης της τεχνολογίας
- της αύξησης της υπολογιστικής ισχύος
- της μνήμης
- του αποθηκευτικού χώρου
- της ταχύτητας της δικτυακής επικοινωνίας

Κατά δέσμες (batch)

Αν και συναντάται σε παλαιότερα συστήματα, υπάρχουν και σήμερα κατά κάποιο τρόπο στα συστήματα GRID (υπολογιστικά πλέγματα). Πάρα πολλοί χρήστες αναθέτουν τις συνήθως απαιτητικές σε πόρους εργασίες τους και αυτές εκτελούνται, όποτε είναι δυνατό, από το σύστημα με κεντρική διαχείριση

Συναλλαγών (transaction)

Εδώ υπάρχει συνεχής επικοινωνία χρήστη-συστήματος και η απόκριση θα πρέπει να δίνεται όσο πιο γρήγορα γίνεται. Αυτό το χαρακτηριστικό συναντάται και στα διαλογικά (interactive) συστήματα.

Μερισμού χρόνου (time sharing)

Το σύστημα διαμοιράζεται σε πολλούς χρήστες και είναι δυνατό να υπάρχει χρέωση για τις υπηρεσίες του.

Πραγματικού χρόνου (real time)

Το σύστημα πρέπει να εξασφαλίζει άμεση απόκριση σε προκαθορισμένο και συνήθως πολύ μικρό χρονικό διάστημα καθώς η λειτουργία του επηρεάζει κρίσιμες διαδικασίες όπως π.χ έλεγχος βιομηχανικών δραστηριοτήτων, έλεγχος αεροπλάνων, διαστημοπλοίων κτλ.

Ανοχής σφαλμάτων ή άνευ παύσης (fault tolerant ή non stop)

Εδώ πρόκειται για συστήματα τα οποία δεν επιτρέπεται να διακόψουν τη λειτουργία τους λόγω βλαβών υλικού ή άλλων λόγων. Προφανώς ένα σύστημα πραγματικού χρόνου θα πρέπει να παρέχει και αυτή τη δυνατότητα.

Κατανεμημένα (distributed)

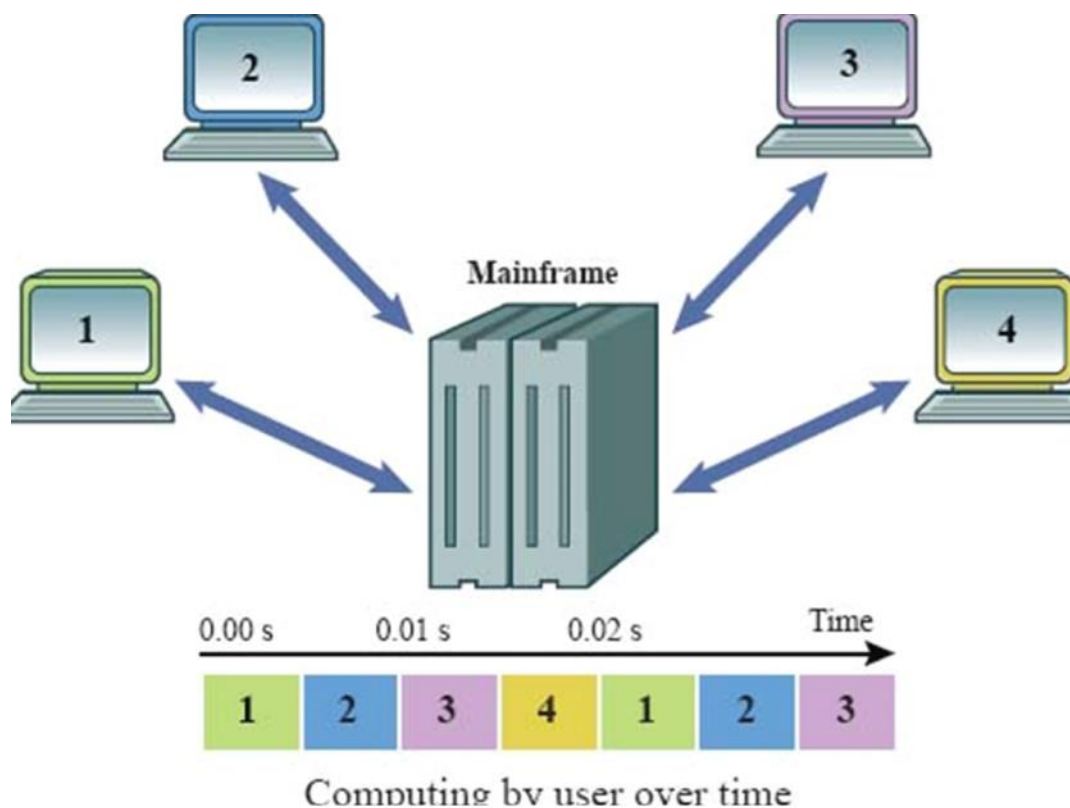
Πρόκειται για συστήματα τα οποία έχουν γεωγραφική διασπορά των σταθμών εργασίας σε διάφορα σημεία.

Συστήματα πελάτη-εξυπηρετητή (client-server)

Αποτελούν την τελευταία εξέλιξη ενσωματώνοντας πολλά από τα παραπάνω χαρακτηριστικά. Πρόκειται για συστήματα όπου υπάρχει ένας ή περισσότεροι κεντρικοί υπολογιστές με επαυξημένες δυνατότητες (εξυπηρετητές/servers) οι οποίοι δέχονται απομακρυσμένες συνδέσεις από άλλους υπολογιστές (πελάτες/clients) και διαμοιράζουν υπολογιστικούς πόρους (όπως αποθηκευτικό χώρο, εκτυπωτές κ.λπ.) και εφαρμογές (όπως βάσεις δεδομένων, εφαρμογές γραφείου κ.λπ.). Ο εξυπηρετητής και οι σταθμοί εργασίας (πελάτες) ελέγχονται

συνήθως από διαφορετικά λειτουργικά συστήματα καθώς πρέπει να εξυπηρετήσουν διαφορετικές ανάγκες ο καθένας.

Ένας νέος όρος στην Πληροφορική, η υπολογιστική νέφους (cloud computing) είναι μια ακόμα εξέλιξη ενός τέτοιου συστήματος που βασίζεται στη χρήση υπηρεσιών του παγκόσμιου ιστού και τη δυνατότητα χρήσης αποθηκευτικού χώρου και εφαρμογών δια μέσου του διαδικτύου.



Κατάταξη με πλήθος χρηστών.

Ανάλογα με τον αριθμό των χρηστών που υποστηρίζουν τα λειτουργικά συστήματα διακρίνονται σε δύο κατηγορίες:

- **Ενός Χρήστη (Single User):** Τα λειτουργικά συστήματα αυτά μπορούν να εξυπηρετήσουν μόνο ένα χρήστη σε κάθε χρονική στιγμή. Χαρακτηριστικά παραδείγματα τέτοιων λειτουργικών συστημάτων είναι τα Windows 95/98/2000, το MS-DOS και το Λειτουργικό Σύστημα των Apple Macintosh.

• **Πολλών Χρηστών (Multiuser):** Τα λειτουργικά συστήματα αυτά μπορούν να εξυπηρετήσουν πολλούς χρήστες το ίδιο χρονικό διάστημα. Χαρακτηριστικά παραδείγματα τέτοιων λειτουργικών συστημάτων είναι τα Windows NT/2000/Server, το UNIX, το LINUX, το NOVELL, το VMS της DEC, το OS/400 της IBM και τα λειτουργικά συστήματα των mainframes (IBM MVS, IBM VM, CDC NOS κ.λπ.).

Κατάταξη με καθεστώς λειτουργίας.

Το καθεστώς λειτουργίας καθορίζεται από την ανάγκη ύπαρξης ή όχι μιας άδειας χρήσης του λογισμικού του λειτουργικού συστήματος η οποία έχει κάποιο κόστος. Όπως και σε πολλές κατηγορίες λογισμικού εφαρμογών, από τις αρχές της δεκαετίας του 90 και έπειτα με την ανάπτυξη και του διαδικτύου, υπάρχει μια νέα φιλοσοφία στην ανάπτυξη λογισμικού βασιζόμενη στη δημιουργία ανοιχτών κοινοτήτων ανάπτυξης λογισμικού. Σε αυτές τις κοινότητες η συμμετοχή είναι συνήθως εθελοντική.

Ομάδες προγραμματιστών δουλεύουν σε κοινές εργασίες σχεδιασμού και ανάπτυξης λογισμικού. Αποτέλεσμα αυτών των δράσεων είναι η δημιουργία λογισμικού το οποίο παρέχεται για χρήση δωρεάν και ταυτόχρονα είναι διαθέσιμος και ο πηγαίος κώδικας του (Ελεύθερο Λογισμικό/Λογισμικό Ανοιχτού Κώδικα, ΕΛΛΑΚ). Συνήθως η μόνη υποχρέωση είναι η συνέχιση της δωρεάν παροχής σε περίπτωση περαιτέρω ανάπτυξης του, ενώ μπορεί να υπάρχει χρέωση για υποστήριξη στη χρήση αυτής της κατηγορίας λογισμικού. Εκπρόσωπος αυτού του τύπου λογισμικού στα λειτουργικά συστήματα είναι το LINUX το οποίο παρέχεται δωρεάν σε διάφορες εκδόσεις/διανομές και βασίζεται στη φιλοσοφία του UNIX.

Εκπαιδευτική Υποενότητα : 4.3 Λειτουργικά συστήματα πολλών χρηστών

Το Λειτουργικό Σύστημα επικοινωνεί με τον ή τους χρήστες μέσα από ένα πρόγραμμα που λέγεται «Διερμηνευτής Εντολών (Command Interpreter)» ή

«Φλοιός (Shell)». Το πρόγραμμα αυτό αναλαμβάνει την ερμηνεία των εντολών του χρήστη και την αποστολή τους στον πυρήνα του λειτουργικού συστήματος για εκτέλεση.

Τα λειτουργικά συστήματα κατηγοριοποιούνται ως εξής σε σχέση με το πλήθος των χρηστών και των διαδικασιών που υποστηρίζουν:

•Λειτουργικό σύστημα ενός χρήστη μίας εφαρμογής (single user single application)

Αυτό το λειτουργικό σύστημα διαχειρίζεται έναν χρήστη και μία μόνο εφαρμογή κάθε φορά. Τέτοιου είδους λειτουργικό σύστημα έχουν σήμερα τα έξυπνα κινητά τηλέφωνα, ενώ η εικόνα που παρουσιάζεται σε ορισμένα από αυτά ότι εκτελούν ταυτόχρονα περισσότερες από μία εφαρμογές είναι πλασματική.

•Λειτουργικό σύστημα ενός χρήστη πολλών διαδικασιών (single user multitasking)

Αυτό το λειτουργικό σύστημα υποστηρίζει έναν χρήστη, αλλά πολλές διαφορετικές διαδικασίες. Οι πολλές διαδικασίες μπορεί να είναι το αποτέλεσμα πολλών διαφορετικών εφαρμογών ή μίας εφαρμογής, η οποία μπορεί να σπάσει σε πολλές διαδικασίες. Παράδειγμα αυτού του λειτουργικού συστήματος είναι το λειτουργικό σύστημα των περισσότερων προσωπικών υπολογιστών.

Η έννοια της πολυεπεξεργασίας (multitasking) συνίσταται στο ότι το λειτουργικό σύστημα μπορεί να διαχειριστεί ταυτόχρονα περισσότερες από μία διαδικασίες, χωρίς να χάνει τον έλεγχό τους. Επιπλέον, μπορεί να μεταπηδά από τη μία στην άλλη, χωρίς να γίνεται αντιληπτό στο χρήστη. Με τον τρόπο αυτό μπορούν να εκτελούνται πολλές εφαρμογές στον ίδιο υπολογιστή. Για παράδειγμα, ένας χρήστης μπορεί να δουλεύει με έναν κειμενογράφο, ενώ ταυτόχρονα να ελέγχει τα δεδομένα ενός λογιστικού φύλλου.

•Λειτουργικό σύστημα πολλών χρηστών και πολλών διαδικασιών (multi-user multitasking)

Ένα λειτουργικό σύστημα αυτής της κατηγορίας μπορεί να διαχειρίζεται πολλούς χρήστες και πολλές διαφορετικές διαδικασίες. Συνήθως, το λειτουργικό σύστημα αυτής της κατηγορίας αφορά υπολογιστές με υψηλές δυνατότητες, δηλαδή υπολογιστές με μεγάλη επεξεργαστική ισχύ και μεγάλη χωρητικότητα μνήμης. Αφορά κυρίως τους μεγάλους υπολογιστές που μπορούν να υποστηρίξουν πλήθος εργασιών με μεγάλες απαιτήσεις και πολλούς χρήστες.

Για παράδειγμα, ένας χρήστης μπορεί να εκτελεί ένα πρόγραμμα εξομοίωσης κάποιου πειράματος, ένας άλλος να εκτελεί ένα πρόγραμμα πρόγνωσης του καιρού, ένας άλλος ένα σχεδιαστικό πρόγραμμα με τρισδιάστατα γραφικά κ.λπ. Το λειτουργικό σύστημα είναι υπεύθυνο για θέματα κατανομής πόρων, διαχείρισης διαδικασιών, απόδοσης, δικαιωμάτων πρόσβασης, γενικότερα θέματα ασφάλειας και πλήθος άλλα για όλους τους χρήστες.

Σε σχέση με το είδος της αλληλεπίδρασης των χρηστών και του λειτουργικού συστήματος διακρίνονται οι εξής κατηγορίες λειτουργικών συστημάτων

•Λειτουργικό σύστημα ομαδικής επεξεργασίας (batch operating system)

Στα λειτουργικά συστήματα αυτά κάθε χρήστης προετοιμάζει τα προγράμματά του και τα υποβάλλει όλα μαζί. Από τη στιγμή που έχει γίνει η υποβολή, ο χρήστης δεν μπορεί να έχει κανενός είδους αλληλεπίδραση με το λειτουργικό σύστημα.

Περιμένει μόνο να λάβει τα αποτελέσματα. Τα λειτουργικά συστήματα είναι σχετικά εύκολα στην υλοποίησή τους, αλλά παρουσιάζουν μειονεκτήματα, καθώς τα προγράμματα πρέπει να εκτελεστούν με τη σειρά. Ο επεξεργαστής συχνά μένει άεργος. Για παράδειγμα, εάν ένα πρόγραμμα εκτελεί λειτουργίες εισόδου/εξόδου, τότε όλα τα υπόλοιπα απλώς περιμένουν την ολοκλήρωση των αργών λειτουργιών εισόδου/εξόδου. Η έλλειψη αλληλεπίδρασης με τον χρήστη οδηγεί σε μεγάλη αναμονή τους χρήστες μέχρι να πάρουν κάποιο αποτέλεσμα. Δεν εφαρμόζεται κάποια πολιτική προτεραιοτήτων, με αποτέλεσμα η απόδοση του συστήματος να

είναι μικρή. Στη σημερινή εποχή τα λειτουργικά συστήματα αυτού του τύπου δεν χρησιμοποιούνται.

•Λειτουργικό σύστημα διανομής χρόνου (time sharing operating system)

Το λειτουργικό σύστημα αυτού του είδους επιτρέπει σε πολλούς χρήστες να αλληλοεπιδρούν μαζί του.

Κάθε χρήστης χρησιμοποιεί ένα κλάσμα του χρόνου του επεξεργαστή. Έτσι, ο επεξεργαστής μπορεί να εκτελεί πολλές διαδικασίες.

Ο χρόνος που διαθέτει σε καθεμία είναι μικρός και η εναλλαγή από τη μια διαδικασία στην άλλη γίνεται γρήγορα. Αυτό δημιουργεί την ψευδαίσθηση στους χρήστες ότι το λειτουργικό σύστημα τους επιτρέπει να δουλεύουν ταυτόχρονα όλοι μαζί στο ίδιο μηχάνημα.

Επιπλέον, οι χρήστες έχουν γρήγορη απόκριση από το σύστημα, αφού, αν μια διαδικασία καθυστερεί να εκτελεστεί, τη θέση της στον επεξεργαστή παίρνει κάποια άλλη διαδικασία, ενώ αυτή περιμένει μέχρι να εκλείψουν οι λόγοι της καθυστέρησής της. Έτσι, μειώνεται ο χρόνος κατά τον οποίο ο επεξεργαστής παραμένει άεργος. Βέβαια, τα συστήματα αυτής της κατηγορίας οφείλουν να αντιμετωπίσουν θέματα ασφάλειας και αξιοπιστίας, καθώς απαιτείται αυστηρός συγχρονισμός των διαδικασιών που είναι ενεργές την ίδια χρονική στιγμή.

•Λειτουργικά συστήματα πραγματικού χρόνου (real time operating systems)

Τα λειτουργικά συστήματα πραγματικού χρόνου πρέπει να αποκρίνονται σε αυστηρούς χρονικούς περιορισμούς. Απόκριση του συστήματος είναι ο χρόνος που μεσολαβεί από τη στιγμή που το σύστημα δέχεται μια αίτηση για εξυπηρέτηση μέχρι να ικανοποιηθεί η αίτηση αυτή. Η χρήση τέτοιων συστημάτων είναι επιβεβλημένη σε περιπτώσεις, για παράδειγμα, επιστημονικών πειραμάτων, ελέγχου της εναέριας κυκλοφορίας ή σε ιατρικά συστήματα ελέγχου.

Σε σχέση με τη θέση στην οποία είναι εγκατεστημένο το λειτουργικό σύστημα διακρίνονται οι εξής κατηγορίες

• Τοπικό λειτουργικό σύστημα (local operating system)

Ο όρος αυτός αναφέρεται στο λειτουργικό σύστημα το οποίο είναι εγκατεστημένο σ' έναν υπολογιστή μόνο και δεν επεκτείνεται η δράση του σε άλλο υπολογιστή.

• Κατανεμημένο λειτουργικό σύστημα (distributed operating system)

Το λειτουργικό σύστημα αυτού του είδους εγκαθίσταται σε πολλούς υπολογιστές, στον καθένα με τη δική του μονάδα επεξεργασίας. Τα κατανεμημένα λειτουργικά συστήματα εξυπηρετούν πολλούς χρήστες, πολλές εφαρμογές και διαχειρίζονται πολλούς επεξεργαστές ταυτόχρονα. Επιπλέον, διαχειρίζονται περισσότερη μνήμη σε σχέση με τα άλλα είδη λειτουργικών συστημάτων. Οι επεξεργαστές επικοινωνούν μεταξύ τους μέσα από διάφορα επικοινωνιακά κανάλια και οι διαδικασίες κατανέμονται στους επεξεργαστές ανάλογα με το ποιος μπορεί να τις διεκπεραιώσει πιο αποδοτικά.

Οι χρήστες έχουν την εντύπωση ότι πρόκειται για ένα σύστημα και όχι για πολλά συνδεδεμένα συστήματα. Ακόμα και στην περίπτωση που σε κάποιον επεξεργαστή παρουσιαστεί πρόβλημα, οι υπόλοιποι συνεχίζουν να λειτουργούν κανονικά. Ο καταμερισμός των διαδικασιών είναι καλύτερος και οι υπηρεσίες προς τους χρήστες ταχύτερες. Ωστόσο, είναι συστήματα δυσκολότερα στην υλοποίηση και στη συντήρηση και απαιτούν γρήγορες γραμμές επικοινωνίας ανάμεσα στους επεξεργαστές.

• Δικτυακό λειτουργικό σύστημα (network operating system)

Αυτά τα λειτουργικά συστήματα είναι εγκατεστημένα σ' έναν διακομιστή (server). Έχουν την ικανότητα να διαχειρίζονται πολλούς χρήστες, εφαρμογές και δικτυακές

λειτουργίες. Πρόκειται για σταθερά συστήματα που παρέχουν κεντρική διαχείριση των υπηρεσιών τους.

Χαρακτηρίζονται από υψηλή αξιοπιστία και αυξημένες δυνατότητες ασφάλειας. Επιπλέον, επιτρέπουν την απομακρυσμένη πρόσβαση από διαφορετικές τοποθεσίες και από διαφορετικούς τύπους μηχανημάτων.

Από την άλλη πλευρά, σε περίπτωση προβλήματος στον διακομιστή, η λειτουργία του συστήματος καταρρέει. Επίσης, χρειάζεται εξειδικευμένο προσωπικό για την αναβάθμιση και τη συντήρησή του. Η διαφορά από τα καταναμημένα λειτουργικά συστήματα είναι πως δεν δίνουν την εντύπωση ενός ενιαίου συστήματος και κάθε τοπικός ηλεκτρονικός υπολογιστής διατηρεί την αυτονομία του.

•Λειτουργικό σύστημα ενεργής σύνδεσης (online operating system)

Τα λειτουργικά συστήματα αυτής της κατηγορίας ονομάζονται και cloud operating systems (λειτουργικό σύστημα νέφους) ή web operating systems (λειτουργικό σύστημα παγκόσμιου ιστού). Δεν πρόκειται για πραγματικό λειτουργικό σύστημα, αλλά για περιβάλλον εικονικού λειτουργικού συστήματος. Οι όροι αυτοί αναφέρονται ουσιαστικά σε διαδικτυακές υπηρεσίες ενός καταναμημένου μοντέλου υπολογισμού. Τα συστήματα αυτής της κατηγορίας δημιουργούν ένα εικονικό περιβάλλον εργασίας.

Ο χρήστης συνδέεται μέσω διαδικτύου στον διακομιστή που προσφέρει εφαρμογές και τις εκτελεί. Τα δεδομένα αποθηκεύονται στον διακομιστή.

Ο υπολογιστής του χρήστη δεν είναι υπεύθυνος για τη διαχείριση των εφαρμογών που εκτελούνται, ούτε για τη διαχείριση του συστήματος αρχείων στο οποίο αποθηκεύονται τα αρχεία.

Το μηχάνημα του χρήστη γίνεται το σημείο διεπαφής όπου παρουσιάζονται οι διαδικτυακές εφαρμογές, αλλά στην πραγματικότητα έχει το δικό του ανεξάρτητο λειτουργικό σύστημα.

Εκπαιδευτική Υποενότητα : 4.4 Πληροφοριακά Συστήματα

Τι σημαίνει σύστημα πληροφοριών (αγγλ. Information Systems ή IS)

Σύστημα πληροφοριών ονομάζεται ένα σύνολο διαδικασιών, ανθρώπινου δυναμικού και αυτοματοποιημένων υπολογιστικών συστημάτων, που προορίζονται για τη συλλογή, εγγραφή, ανάκτηση, επεξεργασία, αποθήκευση και ανάλυση πληροφοριών.

Τα συστήματα αυτά μπορούν να περιλαμβάνουν λογισμικό, υλικό και τηλεπικοινωνιακό σκέλος. Αποτελούν το μέσο για την αρμονική συνεργασία ανθρώπινου δυναμικού, δεδομένων, διαδικασιών και τεχνολογιών πληροφορίας και επικοινωνιών. Προέκυψαν ως γέφυρα μεταξύ των πρακτικών εφαρμογών της επιστήμης υπολογιστών και του επιχειρηματικού κόσμου. «Τα στοιχεία αυτά συλλέγονται, επεξεργάζονται, αποθηκεύονται και μοιράζονται έτοιμες πληροφορίες οι οποίες είναι χρήσιμες για την υποστήριξη στη λήψη αποφάσεων, για τον έλεγχο σε έναν οργανισμό, για το συντονισμό, για τη βοήθεια των στελεχών, για την ανάλυση των προβλημάτων, την απεικόνιση σύνθετων θεμάτων και για τη δημιουργία νέων προϊόντων.»

Τα πληροφοριακά συστήματα περιέχουν διάφορες πληροφορίες, οι οποίες είναι χρήσιμες για τη σωστή εργασία του ανθρώπινου δυναμικού και για την επιχείρηση ή τον οργανισμό για την επίτευξη των στόχων τους οποίους έχει θέσει, αλλά και για το περιβάλλον της επιχείρησης. Με τον όρο πληροφορία εννοούνται τα δεδομένα τα οποία έχουν επεξεργαστεί έτσι ώστε να βγάζουν νόημα και να είναι χρήσιμα στους ανθρώπους. Ενώ τα δεδομένα είναι ροές διάφορων πρωτογενών στοιχείων που αντιπροσωπεύουν γεγονότα τα οποία συμβαίνουν σε οργανισμούς ή στο φυσικό περιβάλλον πριν ακόμα αυτά γίνουν διαθέσιμα σε κατανοητή και χρήσιμη μορφή για τους ανθρώπους. *Οι δραστηριότητες με τις οποίες τα πληροφοριακά συστήματα παράγουν πληροφορίες, που χρειάζεται η οργάνωση για να παίρνει αποφάσεις, να ελέγχει τις λειτουργίες, να αναλύει προβλήματα και να παράγει νέα προϊόντα ή υπηρεσίες, είναι τρεις:*

- η είσοδος
- η επεξεργασία
- η έξοδος

Η **είσοδος** συλλέγει τα δεδομένα μέσα και έξω από την επιχείρηση. Η επεξεργασία μετατρέπει τα δεδομένα της εισόδου σε πιο κατανοητή μορφή. Η έξοδος κάνει τη μεταφορά των επεξεργασμένων δεδομένων στους ανθρώπους, έτσι ώστε να προχωρήσουν στις δραστηριότητες, τις οποίες τους έχουν ανατεθεί. Για τη σωστή συνεργασία των πληροφοριακών συστημάτων με τους χρήστες, θα πρέπει να γίνεται η διαδικασία αν πληροφόρησης, έτσι ώστε να μπορούν να αξιολογηθούν και να διορθωθούν πιθανά λάθη στις πληροφορίες και να συνεχιστεί η διαδικασία εισόδου - εξόδου της πληροφορίας.

Ένα πληροφοριακό σύστημα γνωστό σε όλους είναι και το eBay το οποίο πραγματοποιεί ηλεκτρονικές δημοπρασίες για διάφορες επιχειρήσεις. Σαν είσοδος δίνεται ο κωδικός αναγνώρισης του αγοραστή και το ποσό που προσφέρει. Στη συνέχεια, ο υπολογιστής επεξεργάζεται τα δεδομένα, που έχει πάρει σαν είσοδο και δημιουργεί προσφορές για το αντικείμενο της δημοπρασίας όπου οι προσφορές αυτές είναι η **έξοδος**. Ακολουθώντας τη διαδικασία αυτή το σύστημα παρέχει αναλυτικές πληροφορίες για το υψηλότερο ποσό της προσφοράς, αυτός που έκανε την προσφορά, τον αριθμό των προσφορών και όλες τις υπόλοιπες πληροφορίες σχετικά με το προϊόν. Ένα τέτοιο σύστημα, όπως το eBay, θα πρέπει να είναι κατάλληλο για περιβάλλον πολλών αγοραστών και πολλών πωλητών και να μπορεί να συμμορφώνεται με τις νομοθεσίες των ηλεκτρονικών δημοπρασιών.



Είναι πολύ σημαντικό να γνωρίζεις κανείς τον τρόπο με τον οποίο λειτουργούν οι υπολογιστές και τα προγράμματα, για να μπορεί να σχεδιάζει λύσεις. Οι υπολογιστές όμως αποτελούν μόνο ένα μέρος από τα πληροφοριακά συστήματα.

Αυτό γίνεται κατανοητό ευκολότερα με ένα χαρακτηριστικό παράδειγμα, το σπίτι. Για να χτιστεί ένα σπίτι χρειάζεται μπετονιέρα, τσιμέντο, τούβλα, σίδερα, το σπίτι όμως δεν είναι μόνο αυτά τα υλικά. Σημαντικά είναι η αρχιτεκτονική, ο σχεδιασμός, η διάταξη, η τοπογραφία, οι αποφάσεις οι οποίες οδηγούν στο να γίνει το σπίτι και αυτά είναι μέρος του σπιτιού και είναι πολύ σημαντικά για να βρεθούν οι κατάλληλες λύσεις. Άρα οι υπολογιστές και τα προγράμματα είναι όπως η μπετονιέρα, τα τούβλα, το τσιμέντο και τα σίδερα του πληροφοριακού συστήματος όμως δεν παράγουν από μόνα τους πληροφορίες όπως και τα υλικά δε χτίζουν σπίτι από μόνα τους. Για αυτό πρέπει πρώτα να γίνει πλήρως κατανοητό το πρόβλημα, που χρειάζεται να λυθεί, την αρχιτεκτονική, τα στοιχεία για το σχεδιασμό και τις διεργασίες οι οποίες πρέπει να γίνουν για να φτάσουν στη λύση.

Με τον όρο Πληροφοριακό Σύστημα, γίνεται κατανοητό κάτι, το οποίο έχει σχέση με τον υπολογιστή. Ωστόσο, δεν είναι απαραίτητο κάθε φορά να εννοείται ένα ηλεκτρονικός υπολογιστής με την έννοια Πληροφοριακό Σύστημα, αφού τα

Πληροφοριακά Συστήματα υπήρχαν και πολύ πριν από αυτούς. Από πολύ πριν οι επιχειρήσεις, οι οργανισμοί και οι άνθρωποι χρησιμοποιούσαν διάφορα μικρά και άγνωστα Πληροφοριακά Συστήματα έτσι ώστε να συλλέξουν και να επεξεργαστούν πληροφορίες. Ύστερα άρχισαν να χρησιμοποιούν χειρόγραφα τα Πληροφοριακά Συστήματα μέχρι να βγουν οι ηλεκτρονικοί υπολογιστές, που από τότε και πέρα πραγματοποιήθηκαν μεγάλες εξελίξεις στον τομέα αυτό.

Ο άνθρωπος από μόνος του θα μπορούσε να θεωρηθεί ένα μέσο αποθήκευσης και μεταφοράς δεδομένων. Ένα απλό παράδειγμα είναι ένας άνθρωπος πηγαίνει, μαθαίνει (συλλέγει) διάφορα πράγματα από την καθημερινή του ζωή τα κρατάει (αποθηκεύει) στο μυαλό του. Τη στιγμή που μετακινείται ο άνθρωπος μεταφέρει και τις πληροφορίες μαζί του, χωρίς να τις επεξεργαστεί και τα διανέμει σε άλλους ανθρώπους. Έτσι ακριβώς το ίδιο πράγμα γίνεται μεταξύ πωλητή και πελάτη, όπου ο πωλητής αρχικά συλλέγει πληροφορίες για τη ζήτηση των προϊόντων και στη συνέχεια τις αποθηκεύει στις παραγγελίες και είναι σε αναμονή μέχρι να χρειαστούν για να ολοκληρωθεί η παραγγελία.

Στη δεύτερη περίπτωση με τον πωλητή και τον πελάτη αν ο πωλητής αποφασίσει να περάσει τις πληροφορίες σε έναν ηλεκτρονικό υπολογιστή και ύστερα σε μία βάση δεδομένων, τότε υπάρχει σύζευξη δύο Πληροφοριακών Συστημάτων, με σκοπό τη δημιουργία ενός μεγαλύτερου Πληροφοριακού Συστήματος. Τέτοιες συζεύξεις συμβαίνουν σε επιχειρήσεις και οργανισμούς. Σύμφωνα με τα παραπάνω παραδείγματα, τα Πληροφοριακά Συστήματα διακρίνονται σε δύο κατηγορίες σε χειρογραφικά και μηχανογραφικά τα οποία αποτελούνται από τέσσερα στοιχεία:

•Συλλογή Δεδομένων: όπου αφορά αριθμούς, έρευνες, συζητήσεις και γεγονότα.

•Αποθήκευση Δεδομένων: όπου εδώ μας δίνεται η δυνατότητα να αποθηκεύονται τα δεδομένα στο μυαλό του ανθρώπου, σε αρχεία, σε ηλεκτρονικούς υπολογιστές και σε βάσεις δεδομένων.

•Επεξεργασία Δεδομένων: σε αυτό το στοιχείο βρίσκονται η ανάλυση, η κωδικοποίηση, η ταξινόμηση και η σύνθεση των δεδομένων.

•Παρουσίαση της πληροφορίας: στο στοιχείο αυτό η πληροφορία θα πρέπει να παρουσιάζεται όπως τη θέλει ο χρήστης.

Ο ορισμός των Πληροφοριακών Συστημάτων

Ο όρος πληροφοριακό σύστημα μπορεί να είναι παραπλανητικός, δεδομένου ότι δεν ανακοινώνουν με ακρίβεια τη φύση του θέματος. Σύμφωνα με τον Keen, ένας από τους ηγέτες στον τομέα αυτό, καθορίζει ότι τα πληροφοριακά συστήματα είναι ειδικά σχεδιασμένα, για την επεξεργασία των δεδομένων και τη χρήση τους σε οργανισμούς και επιχειρήσεις. Ο ορισμός αυτός περιλαμβάνει τη λέξη οργάνωση, αλλά δεν περιλαμβάνει τη λέξη διοίκηση. Έτσι το σύστημα αυτό θα περιλαμβάνει διαχειριστές, αλλά θα περιλαμβάνει επίσης και όλα τα άλλα άτομα σε έναν οργανισμό που έχουν σχέση με την δομή και τον σχεδιασμό του οργανισμού.

Ο ολοκληρωμένος ορισμός για τα πληροφοριακά συστήματα διοίκησης είναι ότι, «τα πληροφοριακά συστήματα είναι η ανάπτυξη και η χρήση αποτελεσματικών συστημάτων πληροφόρησης σε οργανισμούς και επιχειρήσεις. Ένα σύστημα είναι αποτελεσματικό αν βοηθάει τον οργανισμό να πετύχει τους στόχους του». Στον κόσμο της τεχνολογίας και της επιστήμης είναι δύσκολο κανείς να δώσει ορισμούς σε διάφορες έννοιες. Το ίδιο και για την έννοια Πληροφοριακά Συστήματα δεν μπορεί να δοθεί ένας συγκεκριμένος ορισμός αλλά υπάρχουν πολλοί και διάφοροι ορισμοί από πολλούς γνωστούς επιστήμονες.

Κάποιοι υποστήριζαν πως τα Πληροφοριακά Συστήματα είναι ίδιο πράγμα με τα συστήματα επεξεργασίας δεδομένων βασισμένα σε ηλεκτρονικό υπολογιστή (computer based data processing systems). Αυτή όμως η δήλωση είναι λανθασμένη, διότι, όπως είχε προαναφερθεί, ένα Πληροφοριακό Σύστημα, είναι ένα σύστημα, το οποίο είναι είτε χειρογραφικό είτε μηχανογραφικό που σημαίνει ότι δε βασίζεται

απαραίτητα σε ηλεκτρονικό υπολογιστή. Επιπλέον, τα συστήματα επεξεργασίας δεδομένων βασισμένα σε ηλεκτρονικό υπολογιστή μερικές φορές δεν παρέχουν καμία πληροφόρηση στη διοίκηση της επιχείρησης. Πολλοί άλλοι επιστήμονες δε δίνουν σαφή εξήγηση για τον ορισμό των Πληροφοριακών Συστημάτων και με όσα αναφέρουν, αφήνουν τον αναγνώστη να συμπεραίνει μόνος του με την δική του κρίση το τι μπορεί να είναι ένα Πληροφοριακό Σύστημα.

Είναι πολύπλοκη η κατάσταση και ανάμεσα στα ανώτερα στελέχη μιας επιχείρησης ή ενός οργανισμού. Το μόνο κοινό που θα μπορούσε να ακούσει κανείς από τα στελέχη μίας είναι ότι δεν υπάρχει ένας και μοναδικός ορισμός για τα Πληροφοριακά Συστήματα αλλά πολλοί και διάφοροι. Υπάρχει έλλειψη γνώσης και κατανόησης ή και επιθυμία κατανόησης των ανώτερων στελεχών όπου είναι σαν να βάζει κανείς ένα άγνωστο προϊόν στην επιχείρησή του και να μην ξέρει το κόστος του και τι θα του αποφέρει. Η αδιαφορία αυτή όμως μπορεί να επηρεάσει τη διαδικασία παραγωγής και ανάπτυξης και την υλοποίηση ενός υψηλής ποιότητας Πληροφοριακού Συστήματος.

Άλλοι ορισμοί από διάφορους επιστήμονες, ώστε να γίνει περισσότερο κατανοητό τι ακριβώς είναι τα Πληροφοριακά Συστήματα:

- «Πληροφοριακό Σύστημα είναι κάποιες οργανωμένες διαδικασίες, οι οποίες όταν εφαρμόζονται υποστηρίζουν τη λήψη αποφάσεων και τον έλεγχο της επιχείρησης ή του οργανισμού» (Lucas, 1982).
- «Τα Πληροφοριακά Συστήματα είναι τυπικά συστήματα που βασίζονται σε ηλεκτρονικούς υπολογιστές οι οποίοι συνδέουν διάφορα δεδομένα από διάφορες πηγές μεταξύ τους έτσι ώστε να έχουν σαν αποτέλεσμα τις απαραίτητες πληροφορίες για την υποστήριξη αποφάσεων» (Hicks, 1987).

Ο εξοπλισμός που χρησιμοποιεί ένα Πληροφοριακό Σύστημα είναι μηχανολογικός και λογικός, χρησιμοποιεί χειρογραφικές διαδικασίες, υποδείγματα προς ανάλυση,

προγραμματισμό, έλεγχο, λήψη αποφάσεων και μία βάση δεδομένων, (Davis και Olson, 1985).

- Ένας άλλος ορισμός δείχνει ότι «Πληροφοριακό Σύστημα είναι ένα σύστημα το οποίο παρακολουθεί το περιβάλλον παίρνει δεδομένα από εκεί, μαζεύει δεδομένα από άλλες επιχειρήσεις, τα επεξεργάζεται, τα οργανώνει και στο τέλος επιλέγει τα δεδομένα που τα παρουσιάζει σαν πληροφορίες στα ανώτερα στελέχη της επιχείρησης ή του οργανισμού, επίσης παρέχει τα κατάλληλα μέσα στα ανώτερα στελέχη για να δημιουργούν την πληροφόρηση που απαιτείται» (Murdick 1986).
- Ακόμη ένας ορισμός αναφέρει πως «Πληροφοριακό Σύστημα είναι ένα σύστημα για τις επιχειρήσεις που επεξεργάζεται κάποια δεδομένα από το εσωτερικό και εξωτερικό περιβάλλον της επιχείρησης ή του οργανισμού και παρέχει πληροφορίες στη διοίκησή της με σκοπό να ληφθούν γρήγορα σωστές και έγκυρες αποφάσεις» (Γεωργόπουλος, Ν. και Οικονόμου, Γ. 2004).

Είναι εύκολο να κατανοηθούν οι βασικές λειτουργίες των Πληροφοριακών Συστημάτων:

- Αναγνωρίζουν και να καλύπτουν τις πληροφοριακές ανάγκες των χρηστών
- Έχουν τη δυνατότητα να επιλέγουν βασικά δεδομένα από μία μεγάλη ποικιλία από το εξωτερικό και εσωτερικό περιβάλλον της επιχείρησης
- Έχουν τη δυνατότητα να δημιουργούν πληροφορίες από τα δεδομένα που έχουν επιλέξει χρησιμοποιώντας τα κατάλληλα εργαλεία
- Δυνατότητα μεταφοράς της δημιουργημένης πληροφορίας στους χρήστες

Τα μέρη ενός πληροφοριακού συστήματος

Τα Πληροφοριακά Συστήματα υπήρχαν και πολύ πιο πριν από τους ηλεκτρονικούς υπολογιστές τα οποία επέτρεπαν στους χρήστες να ελέγχουν και να προγραμματίζουν τις δραστηριότητες τους. Στη συνέχεια με τη βοήθεια των υπολογιστών αναπτύχθηκαν σε μεγάλο βαθμό και είναι πλέον σημαντικά και αποτελούν μεγάλο κομμάτι στην επιχείρηση ή στον οργανισμό.

Ένα Πληροφοριακό Σύστημα αποτελείται από τρία συστατικά στοιχεία:

- **την πληροφορία**
- **το σύστημα**
- **τη διοίκηση**

Η πληροφορία

Ως γνωστό, οι οικονομικές μονάδες από την πρώτη ημέρα της λειτουργίας τους χρησιμοποιούν και επεξεργάζονται δεδομένα και πληροφορίες, έτσι ώστε να είναι πιο εύκολη η λήψη αποφάσεών τους. Επίσης η πληροφορία είναι σημαντική για τη σωστή και αποτελεσματική διοίκηση. Τα στελέχη της επιχείρησης θα πρέπει να έχουν πλήρη πληροφόρηση για τις άλλες επιχειρήσεις, τους πελάτες τους προμηθευτές, τα διάφορα προϊόντα που μπορούν να αντικατασταθούν με τα δικά τους και άλλους παράγοντες οι οποίοι είναι χρήσιμοι για να προχωρήσει η επιχείρηση. Έτσι, σύμφωνα με τα παραπάνω, οι επιχειρήσεις στηρίζονται στην πληροφορία για να πετύχουν τους στόχους τους και τα διάφορα προβλήματα τα οποία προκύπτουν σε μια επιχείρηση τις περισσότερες φορές είναι από την έλλειψη πληροφοριών ή από την έλλειψη ενός συστήματος. Το σύστημα αυτό θα έχει τη δυνατότητα παροχής σωστής και έγκυρης πληροφόρησης την κατάλληλη στιγμή στα στελέχη της επιχείρησης.

Έχοντας ένα τέτοιο σύστημα η επιχείρηση θα έχει τη δυνατότητα να συλλέγει, να αποθηκεύει και να επεξεργάζεται τα δεδομένα τα οποία έχει συλλέξει και στο τέλος να τα μετατρέπει σε πληροφορία.



Οι πληροφορίες αποτελούνται από έξι επίπεδα και χωρίζονται σε τρεις κατηγορίες.

Τα επίπεδα των πληροφοριών διακρίνονται σε:

- Διεθνή: Ένα σχετικό παράδειγμα για αυτό το επίπεδο είναι τα δημογραφικά στοιχεία κρατών
- Εθνικά: Ένα απλό παράδειγμα είναι οι εισαγωγές και εξαγωγές
- Κλάδου: Στο επίπεδο αυτό θα μπορούσαν να είναι τα μερίδια αγοράς, οι πληροφορίες που αφορούν τα ανταγωνιστικά προϊόντα
- Επιχείρησης: Αφορούν τις προβλέψεις των πωλήσεων, τις εκτιμήσεις κόστους
- Τμήματος: Πληροφορίες που έχουν σχέση με υποκαταστήματα
- Ατόμου: Αφορούν πωλήσεις ενός πωλητή ή μιας αντιπροσωπίας κατά την διάρκεια μιας περιόδου, επίσης μπορεί να είναι και ο μισθός ενός εργαζομένου

Οι πληροφορίες ανάλογα με τον σκοπό τους διακρίνονται σε τρεις κατηγορίες:

- **Στρατηγικές**
- **Τακτικές**
- **Λειτουργικές**

Οι κατηγορίες των πληροφοριών έχουν στενή σχέση και με τα επίπεδα των πληροφοριών που μόλις αναφέρθηκαν. Έτσι λοιπόν, οι στρατηγικές πληροφορίες έχουν να κάνουν με τα ανώτερα επίπεδα πληροφορίας, οι τακτικές με τα μεσαία επίπεδα και οι λειτουργικές με τα κατώτερα επίπεδα πληροφοριών. Οι Στρατηγικές Πληροφορίες διακρίνονται σε διεθνείς, εθνικές και πληροφορίες κλάδου. Οι στρατηγικές πληροφορίες πρέπει να είναι σε τέτοια μορφή έτσι ώστε να διευκολύνεται ο μακροπρόθεσμος προγραμματισμός όπως γίνεται στα Συστήματα Ανώτατης Διοίκησης (Executive Information Systems) αρκετές όμως από αυτές είναι από γεγονότα και πηγές οι οποίες δεν επεξεργάζονται με τη βοήθεια ηλεκτρονικού υπολογιστή.

Οι Τακτικές Πληροφορίες αφορούν τις επιχειρήσεις και οι Λειτουργικές Πληροφορίες διακρίνονται σε πληροφορίες τμήματος και πληροφορίες ατόμου. Η κατηγορία στρατηγικών αποφάσεων έχει να κάνει με τις μακροπρόθεσμες υποθέσεις της επιχείρησης και πιο πολύ για την λήψη αποφάσεων για αυτό και έχει να κάνει με τα ανώτερα επίπεδα.

Η κατηγορία των τακτικών πληροφοριών έχει να κάνει με τα μεσοπρόθεσμα προγράμματα μιας επιχείρησης των οποίων η χρονική περίοδος κυμαίνεται από μερικούς μήνες μέχρι και το πολύ δύο χρόνια. Οι τακτικές πληροφορίες είναι κυρίως δεδομένα τα οποία συλλέγονται από τρέχουσες δραστηριότητες της επιχείρησης που σημαίνει ότι θα πρέπει να επεξεργάζονται έγκαιρα. Μερικές φορές οι τακτικές πληροφορίες βασίζονται σε δεδομένα από το εξωτερικό περιβάλλον της επιχείρησης τα οποία είτε χρειάζονται λίγη είτε δεν χρειάζονται καθόλου επεξεργασία.

Στην κατηγορία των λειτουργικών πληροφοριών υποστηρίζονται οι αποφάσεις της καθημερινότητας και οι αποφάσεις του βραχυπρόθεσμου προγραμματισμού όπως για παράδειγμα ημιέτοιμα προϊόντα και ανεκτέλεστες παραγγελίες. Η πλειοψηφία των λειτουργικών πληροφοριών βασίζεται σε δεδομένα από καθημερινές και τρέχουσες δραστηριότητες της επιχείρησης για αυτό και πρέπει να επεξεργάζονται γρήγορα. Οι λειτουργικές πληροφορίες είναι πιο συγκεκριμένες και πιο λεπτομερές

από τις τακτικές και στρατηγικές πληροφορίες και πρέπει να παρουσιάζονται με κατάλληλο τρόπο έτσι ώστε να καλύπτουν τις απαιτήσεις των χρηστών.

Χαρακτηριστικά της πληροφορίας

Σύμφωνα με αυτά που ειπώθηκαν έως τώρα για την πληροφορία, βγαίνει συμπέρασμα πως όσο πιο πολλές πληροφορίες υπάρχουν, τόσο πιο πολύ αυξάνεται και η γνώση και γίνεται ευκολότερη η λήψη σωστών αποφάσεων. Οι πληροφορίες μπορούν να είναι μεμονωμένες πληροφορίες ή σύνολο πληροφοριών και έχουν κάποια χαρακτηριστικά έτσι ώστε να είναι αποτελεσματικά.

Τα ακόλουθα είναι τα χαρακτηριστικά της μεμονωμένης πληροφορίας:

- **Ακρίβεια:** Η ακρίβεια της πληροφορίας διακρίνεται σε αληθινή ή ψευδής και ακριβής ή ανακριβής. Ακριβής πληροφορία είναι αυτή που δείχνει την πληροφορία έτσι όπως είναι στην πραγματικότητα. Η ανακριβής πληροφορία είναι αυτή, που επιστρέφει λάθος αποτελέσματα.
- **Μορφή:** Έχει σχέση με τη δομή της πληροφορίας. Διακρίνεται σε ποσοτική και ποιοτική πληροφορία. Η ποσοτική πληροφορία έχει να κάνει με μετρήσεις ενώ οι ποιοτικές πληροφορίες έχουν να κάνουν με την περιγραφή καταστάσεων
- **Συχνότητα:** Η συχνότητα αφορά το πόσο συχνά χρειάζεται μία πληροφορία
- **Χρονικός Ορίζοντας:** Ο χρονικός ορίζοντας αναφέρεται στο παρελθόν, στο παρόν και στο μέλλον της πληροφορίας. Χωρίζεται σε δύο μέρη, την Ιστορική Πληροφορία, η οποία αναφέρεται στο παρελθόν και τη Μελλοντική Πληροφορία, η οποία αναφέρεται στο μέλλον
- **Έκταση:** Αναφέρεται στο πεδίο, όπου αναφέρεται η πληροφορία
- **Προέλευση:** Η προέλευση έχει σχέση με το από πού προέρχεται η πληροφορία, από το εσωτερικό ή από το εξωτερικό περιβάλλον της επιχείρησης.

Τα χαρακτηριστικά μιας ομάδας πληροφοριών είναι:

- **Σχετικότητα:** Σχετικές είναι οι πληροφορίες οι οποίες χρησιμοποιούνται για τη λύση κάποιου προβλήματος ή για τη λήψη μιας απόφασης. Η σχετικότητα κρίνεται σύμφωνα με το χρόνο χρησιμοποίησής της. Σχετικές θεωρούνται, επίσης, οι πληροφορίες οι οποίες χρησιμοποιούνται σε μια τρέχουσα κατάσταση.
- **Πληρότητα:** Ένας χρήστης λαμβάνει πληροφορίες τις οποίες χρειάζεται για να λύσει το πρόβλημα είτε σε μεγάλο είτε σε μικρότερο βαθμό. Η πληρότητα διακρίνεται σε πλήρες και ελλιπές. Όταν ο χρήστης παίρνει όλες τις πληροφορίες που χρειάζεται τότε λέγεται πως είναι πλήρες ενώ όταν έχει αναπάντητα ερωτήματα και απορίες τότε είναι ελλιπές.
- **Επικαιρότητα:** Οι πληροφορίες που δίνονται σε κάθε χρήστη θα πρέπει να είναι επίκαιρες να δίνονται ακριβώς τη χρονική στιγμή που τις χρειάζεται ο χρήστης

Η διαφορά της πληροφορίας και των δεδομένων

Τα δεδομένα (data) και οι πληροφορίες είναι δύο έννοιες που χρησιμοποιούνται καθημερινά στον κόσμο της τεχνολογίας και όχι μόνο. Λέγοντας δεδομένα με πολύ απλά λόγια μπορεί να αναλυθεί ως εξής: «είναι η πρώτη ύλη για τα Πληροφοριακά Συστήματα, για τη δημιουργία της πληροφορίας».

Τα δεδομένα λοιπόν, είναι τα στοιχεία τα οποία έχουν συλλεχθεί από διάφορες πηγές του εσωτερικού και εξωτερικού περιβάλλοντος της επιχείρησης τα οποία χρησιμοποιούνται για την περιγραφή των γεγονότων, των πραγμάτων, των προσώπων, των ιδεών κ.λπ. Στην αντίθετη περίπτωση με τον όρο πληροφορία (information), εννοούνται τα δεδομένα, τα οποία έχουν συλλεχθεί και τα έχουν επεξεργαστεί, έτσι ώστε να έχουν σαν αποτέλεσμα κάποια έννοια ή αξία για τη σωστή λήψη αποφάσεων στην επιχείρηση και για τη σωστή εκτέλεση των δραστηριοτήτων.



Μερικοί επιστήμονες γράφουν, πως η σχέση που υπάρχει ανάμεσα στα δεδομένα και την πληροφορία είναι ίδια με τη σχέση, που υπάρχει ανάμεσα στην πρώτη ύλη και το έτοιμο προϊόν. Όταν ληφθούν τα δεδομένα δεν είναι δυνατόν να χρησιμοποιηθούν κάπου. Αντίθετα, αν επεξεργαστούν και τα μετατραπούν σε μία χρήσιμη πληροφορία, είναι πολύ χρήσιμα στη λήψη αποφάσεων.

Υπάρχει και μία αλυσίδα ανάμεσα στα δεδομένα και την πληροφορία δηλαδή, μία πληροφορία κάποιας επιχείρησης μπορεί να είναι δεδομένα για κάποια άλλη επιχείρηση. Για παράδειγμα, με τα προϊόντα, ένα έτοιμο προϊόν, το οποίο παράγει μία επιχείρηση, μπορεί να είναι πρώτη ύλη για μία άλλη επιχείρηση, που το χρησιμοποιεί για να φτιάξει το τελικό προϊόν. Σύμφωνα λοιπόν με τα παραπάνω, τα δεδομένα χωρίς να έχουν μπει στην κατάλληλη μορφή δεν μπορούν να είναι πληροφορία, έτσι και η πληροφορία είναι δεδομένα τα οποία έχουν επεξεργαστεί κατάλληλα. Σε μία επιχείρηση συμβαίνει πολλές φορές τα άτομα να χρησιμοποιούν τα ίδια δεδομένα για να φτάσουν όμως στη λήψη διαφορετικών απόψεων.

Κύκλος ζωής δεδομένων

Όπως κάθε τι στον κόσμο έχει το δικό του κύκλο ζωής έτσι και τα δεδομένα έχουν το δικό τους κύκλο ζωής. Για να αναπτυχθεί, να σχεδιαστεί και να λειτουργήσει ένα Πληροφοριακό Σύστημα σωστά είναι πολύ σημαντικός ο κύκλος ζωής των δεδομένων. Τα πιο σημαντικά στοιχεία του κύκλου ζωής είναι:

- Να είναι γνωστός ο τρόπος με τον οποίο δημιουργούνται τα δεδομένα
- Να είναι γνωστός ο τρόπος με τον οποίο τα δεδομένα γίνονται πληροφορίες
- Να είναι γνωστός ο τρόπος επεξεργασίας των πληροφοριών, μεταφοράς των δεδομένων, μετάδοσης των πληροφοριών, η αποθήκευση και επανάκτησή τους

Ο κύκλος ζωής δεδομένων αποτελείται από 11 στάδια:

- **Δημιουργία:** Συλλογή δεδομένων μέσα και έξω από την επιχείρηση
- **Αποθήκευση:** Μετά από τη δημιουργία των δεδομένων, τα δεδομένα αποθηκεύονται σε αρχεία ή σε μία βάση δεδομένων
- **Καταστροφή:** Καταστρέφονται τα δεδομένα εκείνα που είναι άχρηστα
- **Μεταφορά:** Η μεταφορά των δεδομένων από μία διαδικασία σε κάποια άλλη
- **Επανάκτηση:** Η επανάκτηση των δεδομένων γίνεται από το σύστημα διαχείρισης βάσης δεδομένων
- **Αναπαραγωγή:** Το στάδιο της αναπαραγωγής των δεδομένων αφορά όλες τις κατηγορίες αναπαραγωγής δεδομένων τα οποία είναι αποθηκευμένα σε κάποια βάση ή σε κάποιον ηλεκτρονικό υπολογιστή ακόμα και αυτά που δεν είναι έτοιμα να χρησιμοποιηθούν.
- **Αξιολόγηση:** Μετά το στάδιο της επανάκτησης γίνεται η αξιολόγηση των δεδομένων. Στην αξιολόγηση αποφασίζεται αν τα δεδομένα πρέπει να επεξεργαστούν ή να επιστραφούν πίσω στη βάση δεδομένων, για να αποθηκευτούν ή για να καταστραφούν αν είναι άχρηστα. Όταν τα δεδομένα μετατραπούν σε

πληροφορία, ξαναγίνεται νέα αξιολόγηση για να διαπιστώσουν αν χρειάζονται κάποιες αλλαγές.

- **Ανάλυση:** Πριν τα δεδομένα ετοιμαστούν για παρουσίαση πρέπει πρώτα να αναλυθούν με τέτοιο τρόπο έτσι ώστε να καλύπτουν τις ανάγκες των χρηστών
- **Ταξινόμηση:** Μετά τη συλλογή των δεδομένων, τα δεδομένα συγκεντρώνονται με τυχαία σειρά για αυτό το λόγο υπάρχει και το στάδιο της ταξινόμησης έτσι ώστε να ταξινομηθούν με βάση κάποιο κριτήριο και να είναι εύκολα στη χρήση τους. Έχουν υπάρξει περιπτώσεις όπου τα ταξινομημένα δεδομένα ταξινομούνται ξανά από την αρχή με διαφορετικό όμως κριτήριο.
- **Σύνθεση:** Στο στάδιο αυτό ενοποιούνται τα δεδομένα για τη δημιουργία μιας ολοκληρωμένης αναφοράς.
- **Δημιουργία πληροφορίας:** Στο τελευταίο αυτό στάδιο τα δεδομένα μετατρέπονται σε πληροφορίες οι οποίες χρησιμοποιούνται από τους χρήστες για να πάρουν εύκολα και σωστά αποφάσεις

Το σύστημα

Ένα σύστημα είναι ένα σύνολο συστατικών μερών, τα οποία συνεργάζονται και είναι λειτουργικά συνδεδεμένα μεταξύ τους, με αποτέλεσμα να πετύχουν το στόχο τους. Τα συστατικά μέρη των συστημάτων είναι ενωμένα μεταξύ τους με τέτοιο τρόπο, ώστε να πετυχαίνουν ένα συνεργικό αποτέλεσμα και μπορούν να είναι ο άνθρωπος, το μηχάνημα, η πληροφορία, η επιχείρηση. Συνεργικό αποτέλεσμα υπάρχει, όταν το αποτέλεσμα, το οποίο προκύπτει από την ενοποίηση των συστατικών μερών του συστήματος, είναι μεγαλύτερο από το αποτέλεσμα, το οποίο προκύπτει από το άθροισμα των επιμέρους μερών.

Στη σημερινή εποχή, η οποία είναι και η εποχή της συνεχής εξελισσόμενης τεχνολογίας η έννοια του συστήματος είναι αρκετά διαδεδομένη. Καθημερινά, χρησιμοποιείται η λέξη σύστημα μαζί με διάφορες έννοιες, όπως για παράδειγμα συστήματα ηλεκτρονικού υπολογιστή, βιολογικά συστήματα, κοινωνικά συστήματα και πολλές άλλες έννοιες χρησιμοποιούν τη λέξη σύστημα. Ακόμα και οι απλοί καταναλωτές, όταν αγοράζουν κάποια πράγματα, τα ζητάνε σαν σύστημα. Για

παράδειγμα αντί για φωτογραφική μηχανή ζητάνε φωτογραφικά συστήματα ή ακόμα πιο απλά αντί για ηχεία ζητάνε ηχοσύστημα.

Το ίδιο πλέον ισχύει για όλες τις επιχειρήσεις και τους οργανισμούς όπου υπάρχουν πολλά και διάφορα συστήματα. Το πιο απλό σύστημα σε μία επιχείρηση είναι η κάρτα εργασίας του εργαζομένου και οι διαδικασίες, οι οποίες γίνονται για τον υπολογισμό του χρόνου εργασίας κάθε εργαζόμενου ανά εβδομάδα ή ανά μήνα. Ένα πιο πολύπλοκο σύστημα σε μία επιχείρηση είναι το λογιστικό σύστημα, όπου σε αυτό απαιτείται συνεργασία ανθρώπων, μηχανημάτων, διαδικασιών και κανόνων με σκοπό την εγγραφή των δεδομένων, τη μέτρηση των οικονομικών αποτελεσμάτων και την προετοιμασία της τελικής έκθεσης. Διαπιστώνεται λοιπόν, ότι ένα σύστημα δεν είναι απλά κάτι τυχαίο, το κάθε σύστημα αποτελείται από συστατικά μέρη τα οποία συνεργάζονται μεταξύ τους έτσι ώστε να πετύχουν το σκοπό τον οποίο έχουν θέσει. Γενικότερα, τα συστήματα αποτελούν μία ολότητα, τελειότητα και ενοποίηση των στοιχείων που το αποτελούν. Τα στοιχεία τα οποία είναι απαραίτητα για να υπάρξει ένα σύστημα είναι:

- το περιβάλλον
- τα όρια
- οι εισροές και εκροές
- τα συστατικά μέρη του συστήματος

Περιβάλλον του συστήματος:

Όλα τα συστήματα λειτουργούν μέσα σε κάποιο περιβάλλον από το οποίο επηρεάζονται και το οποίο επηρεάζουν σε μικρό ή και μεγάλο βαθμό ανάλογα. Ένα σχετικό παράδειγμα είναι ότι μέσα σε ένα περιβάλλον κάποιου συστήματος υπάρχουν διάφορες δραστηριότητες τις οποίες λειτουργούν άνθρωποι και μηχανήματα μαζί και μπορούν να φέρουν διάφορες επιπτώσεις στη κανονική λειτουργία της επιχείρησης. Έτσι ακριβώς το ίδιο ισχύει και για το περιβάλλον ενός

πληροφοριακού συστήματος όπου και αυτό περιλαμβάνει διάφορες δραστηριότητες και ανθρώπους ή οντότητες που επεξεργάζονται τα δεδομένα για το σύστημα.

Όρια του συστήματος:

Τα όρια του συστήματος ξεχωρίζουν το σύστημα από το περιβάλλον του. Ένα σύστημα βρίσκεται και λειτουργεί μέσα στα όριά του ενώ ότι υπάρχει έξω από τα όρια του συστήματος αποτελεί το περιβάλλον του συστήματος. Επίσης, τα όρια του συστήματος είναι ικανά να ελέγχουν τις εισροές και τις εκροές του συστήματος όπως επίσης να τις ρυθμίζουν και να παρέχουν ασφάλεια από τυχόν ζημιογόνες δραστηριότητες που έρχονται από το περιβάλλον. Για το λόγο αυτό, τα όρια μπορούν να χαρακτηριστούν και σαν φίλτρα των εισροών και των εκροών.

Η δραστηριότητα των ορίων σαν φίλτρα δεν είναι τόσο εύκολη όσο ακούγετε, είναι μία δύσκολη και υποκειμενική διαδικασία. Τα όρια του συστήματος ορίζονται από το άτομο το οποίο θα ερευνήσει το σύστημα, τα όρια τα ορίζει μόνος του, έτσι ώστε να του είναι πιο εύκολη η μελέτη του συστήματός του και θα πρέπει επίσης, να γνωρίζει πια όρια να επιλέξει. Ένα απλό παράδειγμα όσον αφορά τα παραπάνω είναι μία επιχείρηση και συγκεκριμένα το τμήμα παραγωγής της επιχείρησης. Το σύστημα παραγωγής της επιχείρησης θα έχει στα όριά του τον μηχανολογικό εξοπλισμό, τα ημιτελή προϊόντα, τους εργαζόμενους στον τομέα παραγωγής και τις διαδικασίες παραγωγής.

Άλλες λειτουργίες, που αφορούν την παραγωγή, όπως για παράδειγμα το μάρκετινγκ, η έρευνα και ανάπτυξη, είναι το περιβάλλον του συστήματος παραγωγής. Με αυτό το παράδειγμα, φαίνεται πως είναι δύσκολο κανείς να ξεχωρίσει τα όρια του συστήματος από το περιβάλλον.

Εισροές και εκροές του συστήματος:

Οι εισροές και οι εκροές είναι τα στοιχεία εκείνα τα οποία κάνουν το σύστημα να επικοινωνεί με το περιβάλλον του. Η εισροή είναι κάτι που εισέρχεται στο σύστημα από το περιβάλλον του συστήματος ενώ η εκροή είναι κάτι που έχει επεξεργαστεί μέσα στο σύστημα έχει περάσει τα όρια και εξέρχεται από το σύστημα στο περιβάλλον. Έτσι λοιπόν και ένα Πληροφοριακό Σύστημα είναι ένα σύστημα, στο οποίο εισέρχονται δεδομένα ως εισροές, επεξεργάζονται κατάλληλα, μετατρέπονται σε πληροφορία και εξέρχονται από το σύστημα ως εκροές.

Συστατικά μέρη του συστήματος:

Σύμφωνα με τα παραπάνω ένα σύστημα ορίζεται από τα όριά του, μπορεί να είναι μια απλή οντότητα ή και να αποτελείται από περισσότερα συστατικά μέρη. Μερικές φορές μπορεί ένα συστατικό μέρος του συστήματος να είναι από μόνο του ένα σύστημα τότε ονομάζεται υποσύστημα. Ένα συστατικό μέρος ενός συστήματος είναι μία μονάδα η οποία συνεργάζεται με άλλα συστατικά μέρη ή άλλα υποσυστήματα έτσι ώστε να μπορέσει να πετύχει το σκοπό του και να φτάσει στο επιθυμητό αποτέλεσμα. Όταν αναφέρεται ο σκοπός του συστήματος εννοείται μία εκροή η οποία μπορεί να είναι εισροή σε κάποιο άλλο σύστημα, που σημαίνει ότι μέσα σε ένα σύστημα πραγματοποιούνται συνδέσεις μεταξύ των υποσυστημάτων όπως επίσης και αλληλεπίδραση του περιβάλλοντος με το σύστημα.

Τύποι συστημάτων

Στον κόσμο των συστημάτων υπάρχουν δύο βασικοί τύποι συστημάτων, οι οποίοι είναι οι πιο χρήσιμοι και οι πιο σκόπιμοι να αναφερθούν. Υπάρχουν τα Προσδιοριστικά και Πιθανολογικά Συστήματα και τα Κλειστά και Ανοιχτά Συστήματα.

Προσδιοριστικά Συστήματα: Προσδιοριστικό σύστημα λέγεται το σύστημα το οποίο λειτουργεί σύμφωνα με κάποιους κανόνες. Όταν υπάρχουν τέτοιου είδους συστήματα μπορούν να γίνουν προβλέψεις για μελλοντικές συμπεριφορές του

συστήματος εάν όμως πρώτα είναι γνωστά τα χαρακτηριστικά του συστήματος και η τρέχουσα κατάστασή του, δηλαδή τη στιγμή που ελέγχεται. Απλά παραδείγματα προσδιοριστικών συστημάτων από την καθημερινή ζωή είναι η διαδικασία ανάληψης χρημάτων από την τράπεζα και διάφορα προγράμματα ηλεκτρονικών υπολογιστών όπου και τα δύο παραδείγματα για να φτάσουν στο επιθυμητό αποτέλεσμα ακολουθούν κάποιους κανόνες.

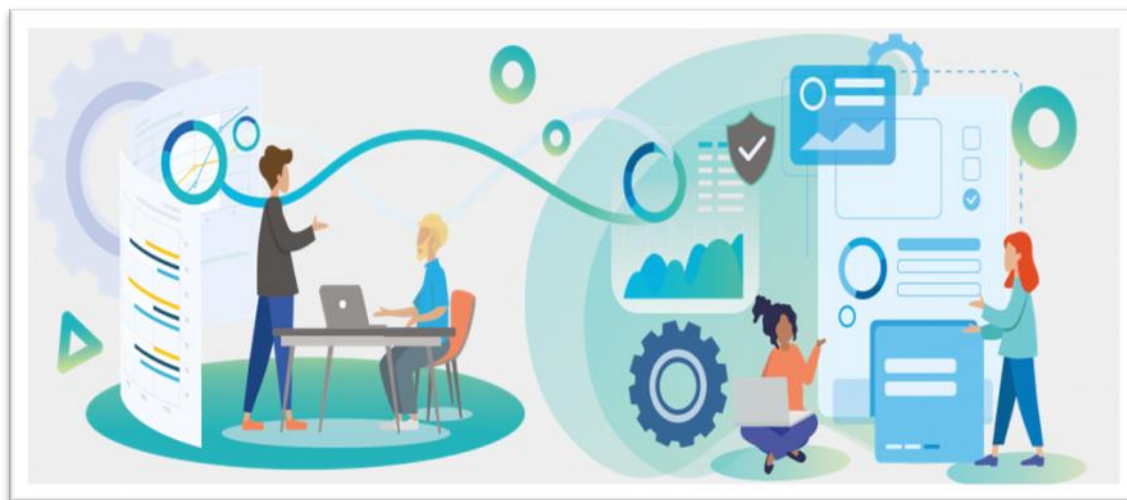
Πιθανολογικά Συστήματα: Με τον όρο πιθανολογικό σύστημα εννοείται ένα σύστημα το οποίο αλληλοεπιδρά με κάποιους εξωτερικούς παράγοντες. Άρα παθολογικό σύστημα είναι ένα σύστημα που επηρεάζεται εύκολα από μη βέβαια συμβάντα και δεν μπορεί να προβλέψει κανείς την συμπεριφορά του στο μέλλον διότι η κύρια ασχολία του είναι να εξετάζει πιθανότητες. Ένα απλό παράδειγμα πιθανολογικού συστήματος είναι το σύστημα διαχείρισης αποθεμάτων διότι τα αποθέματα επηρεάζονται από την ζήτηση των καταναλωτών.

Όσον αφορά τα πληροφοριακά συστήματα στις επιχειρήσεις και τους οργανισμούς τα πιο πολλά είναι προσδιοριστικά διότι τις περισσότερες φορές υπάρχουν προκαθορισμένοι κανόνες τους οποίους πρέπει να ακολουθήσει το σύστημα για να επεξεργαστεί τα δεδομένα και να επιστρέψει τις απαιτούμενες πληροφορίες. Τέτοιο σύστημα όμως θα μπορούσε να θεωρηθεί μερικές φορές και πιθανολογικό διότι υπάρχει μεγάλος αριθμός εισροών που μπορούν να επηρεάσουν το σύστημα και να μην είναι βέβαιη η μελλοντική συμπεριφορά του συστήματος.

Κλειστά συστήματα:

Με τον όρο κλειστό σύστημα εννοείται ένα σύστημα το οποίο δεν επικοινωνεί με το περιβάλλον του, που σημαίνει ότι δεν ανταλλάσσει πληροφορίες και ενέργειες με το περιβάλλον του. Ένα πολύ απλό παράδειγμα κλειστού συστήματος είναι μία κοινωνία η οποία είναι απομονωμένη, θα μπορούσε να είναι μία πόλη, η οποία από μόνη της μπορεί να παράγει τα απαραίτητα τρόφιμα και να μην έχει συνεργασίες με άλλες κοινωνίες. Τα κλειστά συστήματα είναι πιο εύκολο να τα συναντήσει κανείς

στα επιστημονικά συστήματα και όχι σε κοινωνικά. Για την ευκολότερη επίλυση ενός συστήματος μερικές φορές το λύνουν με βάση ενός κλειστού συστήματος, έτσι ώστε να δοθεί μία πρώτη εκτίμηση στο πρόβλημα.



στα επιστημονικά συστήματα και όχι σε κοινωνικά. Για την ευκολότερη επίλυση ενός συστήματος μερικές φορές το λύνουν με βάση ενός κλειστού συστήματος, έτσι ώστε να δοθεί μία πρώτη εκτίμηση στο πρόβλημα.

Ανοικτό σύστημα:

Μετά την ανάλυση του κλειστού συστήματος είναι εύκολο να αναλυθεί τι είναι το ανοικτό σύστημα. Το ανοικτό σύστημα είναι το αντίθετο από το κλειστό, δηλαδή, το ανοικτό σύστημα επικοινωνεί με το περιβάλλον του και ανταλλάσσει πληροφορίες. Οι εισροές σε ένα ανοικτό σύστημα μπορούν να είναι προβλέψιμες και μη. Ένα απλό παράδειγμα ανοικτού συστήματος είναι τα βιολογικά συστήματα. Τα συστήματα στις επιχειρήσεις λειτουργούν το ένα σύστημα μέσα στο άλλο, δηλαδή ένα μικρότερο σύστημα λειτουργεί μέσα σε ένα μεγαλύτερο.

Πιο απλά, η επιχείρηση από μόνη της είναι ένα μεγάλο σύστημα και μέσα της δουλεύουν άλλα τμήματα το καθένα σαν σύστημα έτσι λέμε ότι το ένα σύστημα δουλεύει μέσα στο άλλο. Κύριο χαρακτηριστικό των ανοικτών συστημάτων είναι η δυνατότητα προσαρμογής τους στις συνεχείς αλλαγές στο εξωτερικό περιβάλλον. Όταν ένα σύστημα δεν εκτελεί σωστά τη λειτουργία του και δεν είναι ικανό να ανταπεξέλθει στις εξωτερικές αλλαγές τότε το σύστημα αυτό φτάνει σε μία κατάσταση παράλυσης. Αντίθετα, όταν ένα ανοικτό σύστημα εκτελεί σωστά τις

λειτουργίες του και είναι ικανό να προβλέψει και να προσαρμοστεί στις αλλαγές του εξωτερικού περιβάλλοντος, τότε λέγεται ότι το σύστημα βρίσκεται σε κατάσταση δυναμικής ισορροπίας.

Τα πληροφοριακά συστήματα είναι συστήματα μεταξύ χρήστη και μηχανής τα οποία κάνουν τις κατάλληλες δραστηριότητες και επεξεργασίες δεδομένων έτσι ώστε να πετύχουν το στόχο τον οποίο έχουν θέσει. Και ο χρήστης αλλά και η μηχανή είναι το καθένα από μόνο του ένα σύστημα. Η μηχανή και το λογισμικό του ηλεκτρονικού υπολογιστή είναι κλειστά και προσδιοριστικά συστήματα ενώ ο χρήστης είναι ανοικτό και πιθανολογικό σύστημα. Για το λόγο αυτό υπάρχουν πολλοί συνδυασμοί στα πληροφοριακά συστήματα. Αυτό γίνεται πιο εύκολα κατανοητό λέγοντας ότι, σε ένα πληροφοριακό σύστημα αν δοθεί έμφαση στον ηλεκτρονικό υπολογιστή και όχι στο χρήστη, ο χρήστης το μόνο που έχει να κάνει είναι να παρακολουθεί τον ηλεκτρονικό υπολογιστή την ώρα που εκτελεί κάποιες λειτουργίες. Επιπλέον, αν ο ηλεκτρονικός υπολογιστής λειτουργεί σαν υποστηρικτής του χρήστη, όπου ο υπολογιστής κάνει όλους τους υπολογισμούς και ο χρήστης είναι διατεθειμένος να κάνει τις σημαντικότερες εργασίες.

Ο έλεγχος των συστημάτων

Ο έλεγχος των συστημάτων είναι η πιο σημαντική διαδικασία στην οργάνωση και τη διοίκηση των επιχειρήσεων. «Έλεγχος συστήματος εννοείται η διαδικασία στην οποία μετριέται η απόδοση του συστήματος και αν είναι ικανό να οδηγηθεί στο σκοπό που έχει θέσει».

Ο έλεγχος των συστημάτων έχει κάποια στοιχεία τα οποία είναι:

- ο σκοπός και ο στόχος
- το πρότυπο
- η μέτρηση της απόδοσής του
- η σύγκριση της απόδοσης με το στόχο τον οποίο έχει θέσει

- τα μέσα που χρησιμοποιούνται για να διορθωθεί
- η προσαρμογή στο σύστημα και η ανατροφοδότηση

Ο στόχος και ο σκοπός του συστήματος: η κάθε επιχείρηση που χρησιμοποιεί κάποιο σύστημα θα πρέπει να γνωρίζει το αποτέλεσμα που περιμένει να έχει από τη λειτουργία του συστήματος αυτού. Κάθε σύστημα μπορεί να έχει σαν αποτέλεσμα κάποια δραστηριότητα ή κάποια αριθμητική τιμή. Ένας στόχος μπορεί να είναι είτε απλός είτε πολλαπλός. Απλός στόχος υπάρχει όταν μία επιχείρηση έχει θέσει σαν στόχο να πετύχει κάποιο συγκεκριμένο πράγμα ενώ πολλαπλό στόχο έχει όταν η επιχείρηση θέτει ταυτόχρονα δύο ή παραπάνω στόχους.

Το στοιχείο της μέτρησης αφορά τη δυνατότητα μέτρησης της απόδοσης του συστήματος. Η μέτρηση της απόδοσης πρέπει να γίνεται με τις ίδιες μονάδες μέτρησης που αφορούν τον στόχο και όχι με διαφορετικές διότι τότε δεν θα έχουμε τα επιθυμητά αποτελέσματα (Γεωργόπουλος, Ν. και Οικονόμου, Γ. 2004). Όσον αφορά την σύγκριση υπάρχει η δυνατότητα να συγκριθεί η απόδοση του συστήματος με το στόχο τον οποίο έχει τεθεί και βοηθάει να γίνει κατανοητό αν το σύστημα λειτουργεί σωστά ή όχι. Όταν σχεδιάζεται το σύστημα θα πρέπει σχεδιάζεται κατάλληλα, έτσι ώστε να μπορούν να γίνουν διάφορες αλλαγές και προσαρμογές, που τυχόν θα χρειαστούν όταν η απόκλιση ανάμεσα στην απόδοση και στο αναμενόμενο αποτέλεσμα έχει μεγάλη διαφορά.

Το πιο σημαντικό στοιχείο του ελέγχου είναι η ανατροφοδότηση η οποία είναι κάποια πληροφορία που αφορά την απόδοση του συστήματος και την εισάγει σαν εισροή μέσα στο σύστημα με σκοπό να προσφέρει βοήθεια για το τελικό αποτέλεσμα. Υπάρχουν γνωστά δύο είδη ανατροφοδότησης, η θετική και η αρνητική. Η θετική ανατροφοδότηση βοηθάει τη λειτουργία του συστήματος, έτσι ώστε να μη σταματάει να εκτελεί τις δραστηριότητές του και χωρίς να κάνει αλλαγές. Η αρνητική ανατροφοδότηση μπορεί να χαρακτηριστεί και διορθωτική διότι πραγματοποιεί κάποιες διορθώσεις οι οποίες βοηθάνε το σύστημα να μειώσει τη μεγάλη διαφορά ανάμεσα στην απόκλιση και το στόχο και να συνεχίσει να εκτελεί τις δραστηριότητες του κανονικά.

Η Διοίκηση

Όπως και σε πολλές επιστημονικές έννοιες έτσι και στην έννοια διοίκηση δεν έχει δοθεί ένας συγκεκριμένος ορισμός, υπάρχουν πολλοί ορισμοί για τον όρο Διοίκηση. Μερικοί επιστήμονες υποστηρίζουν πως, η Διοίκηση είναι η διαδικασία που γίνεται



για να επιτευχθούν οι στόχοι μέσω των ανθρώπων. Αξίζει να αναφερθεί ένας ορισμός: *Η Διοίκηση είναι γνωστή ως, η διαδικασία ενοποίησης των πόρων και των ενεργειών, έχοντας σκοπό να πετύχει τους στόχους της επιχείρησης.* Η Διοίκηση είναι το πιο σημαντικό πεδίο μελέτης λόγω των συνεχών αλλαγών που συμβαίνουν, όπως για παράδειγμα η αλλαγή συνηθειών των καταναλωτών, αλλαγές στο εξωτερικό περιβάλλον κ.λπ.. Ο τομέας της Διοίκησης έχει κάποιες βασικές λειτουργίες οι οποίες είναι σκόπιμο να αναφερθούν.

Προγραμματισμός (planning).

Πρώτο στάδιο της Διοίκησης είναι να έχουν τη δυνατότητα τα διοικητικά στελέχη να αποφασίσουν το τι πρέπει να κάνει η επιχείρηση. Τα στελέχη αυτό που κάνουν είναι να θέτουν στόχους και να βρίσκουν εναλλακτικές στρατηγικές, έτσι ώστε να πετύχουν τους στόχους που έχουν θέσει. Το στάδιο του προγραμματισμού όμως επηρεάζεται από το εξωτερικό περιβάλλον της επιχείρησης, όχι μόνο από τις δυνάμεις και αδυναμίες αλλά τις απειλές και τις ευκαιρίες. Για τον λόγο αυτό υπάρχει η ανάγκη ενός Πληροφοριακού Συστήματος έτσι ώστε να μπορεί η επιχείρηση να προσαρμόζεται στις μεταβολές που γίνονται στο περιβάλλον.

Οργάνωση (organising).

Ύστερα από την απόφαση που πάρουν τα στελέχη στο τι πρέπει να κάνει η επιχείρηση θα πρέπει να αποφασίσουν πως θα πραγματοποιηθούν αυτά, αυτό συμβαίνει στο στάδιο της οργάνωσης. Αυτό που κάνουν σε αυτό το στάδιο είναι να οργανώνουν με τέτοιο τρόπο τις λειτουργίες έτσι ώστε να πετύχουν τους στόχους που έχουν θέσει.

Διεύθυνση (leading).

Μετά από τα δύο αυτά στάδια τα διοικητικά στελέχη πλέον βάζουν τους εργαζόμενους να δουλεύουν πάνω στο έργο τους και να τους διευθύνουν να δουλεύουν με τέτοιο τρόπο έτσι ώστε να επιτευχθούν οι στόχοι. Όσο τα διευθυντικά στελέχη είναι κοντά στους εργαζόμενους διευθύνοντας τους τόσο πιο αποτελεσματικοί είναι οι εργαζόμενοι στην εργασία τους.

Έλεγχος (control).

Η αμέσως επόμενη στη σειρά λειτουργία της Διοίκησης είναι ο έλεγχος. Στο στάδιο αυτό προσδιορίζονται τα πρότυπα, γίνεται επίβλεψη της προόδου που έχει πραγματοποιηθεί σε προηγούμενα στάδια και οι διορθωτικές αλλαγές που πρέπει να γίνουν για να πραγματοποιηθούν οι στόχοι της επιχείρησης. Στο στάδιο του ελέγχου τα στελέχη της επιχείρησης έχουν την δυνατότητα να μετράνε την επίδοση των εργαζομένων. Εξετάζουν την αποδοτικότητα και την αποτελεσματικότητά τους, όπως επίσης, ψάχνουν για προβλήματα επικοινωνίας μεταξύ των εργαζομένων και των στελεχών και την κατανομή των πόρων.

Συντονισμός (coordination).

Η τελευταία λειτουργία στη Διοίκηση είναι ο συντονισμός όπου τα ανώτερα στελέχη αξιολογούν όλες τις πληροφορίες που έχουν συλλέξει από την προηγούμενη λειτουργία του ελέγχου. Η λειτουργία του συντονισμού εκτός από αυτό ασχολείται

επίσης με τον καθορισμό των παραγόντων και των δραστηριοτήτων που πρέπει να τροποποιηθούν, έτσι ώστε να επιτευχθούν οι στόχοι οι οποίοι έχουν τεθεί. Ο συντονισμός όλων των τμημάτων και των δραστηριοτήτων της επιχείρησης βοηθάει στο τελικό αποτέλεσμα δηλαδή στην επίτευξη των στόχων.

Τα πληροφοριακά συστήματα από την άποψη των επιχειρήσεων

Ένα πληροφοριακό σύστημα για μία επιχείρηση σημαίνει οργανωτική και διοικητική λύση η οποία βασίζεται στην τεχνολογία της πληροφορικής. Για να καταλάβει κάποιο στέλεχος της επιχείρησης τον ορισμό αυτό θα πρέπει να γνωρίζει πρώτα τι είναι οργάνωση, μάνατζμεντ, τεχνολογία πληροφοριακών συστημάτων, τη δυνατότητα που έχουν να δίνουν λύσεις στα προβλήματα και τα προβλήματα του περιβάλλοντος της επιχείρησης.

Οργάνωση

Τα πληροφοριακά συστήματα είναι πολύ σημαντικό μέρος της οργάνωσης. Τα κυριότερα στοιχεία μιας οργάνωσης είναι ο άνθρωπος, η δομή, οι λειτουργικές διαδικασίες, η πολιτική και η κουλτούρα της. Μια επιχείρηση αποτελείται από διαφορετικά επίπεδα και οργανώσεις. Οι πιο σημαντικές επιχειρηματικές λειτουργίες της οργάνωσης είναι η πώληση, το μάρκετινγκ, η παραγωγή, οι χρηματοοικονομικές υπηρεσίες, το λογιστήριο και οι ανθρώπινοι πόροι.

Η οργάνωση συντονίζει το έργο με τη βοήθεια μιας ιεραρχικής δομής και κάποιων διαδικασιών λειτουργίας. Η ιεραρχική δομή κάνει την κατανομή των ανθρώπων ανάλογα με την εξουσία και την ευθύνη τους. Στα ανώτερα επίπεδα βρίσκονται τα διοικητικά, εξειδικευμένα και τεχνικά στελέχη, και στα κατώτερα βρίσκονται οι υπάλληλοι. Στην οργάνωση παίρνουν μέρος και οι πρότυπες διαδικασίες λειτουργιών, οι οποίες είναι μακροπρόθεσμοι επίσημοι κανόνες, που βοηθάνε το προσωπικό στις διαδικασίες, που κάνουν από τις πιο απλές μέχρι και τις πιο πολύπλοκες. Οι περισσότερες πρότυπες διαδικασίες λειτουργίας είναι ενσωματωμένες στα πληροφοριακά συστήματα που χρησιμοποιούν οι επιχειρήσεις.

Οι διαφορετικές δεξιότητες και οι άνθρωποι είναι τα πιο αναγκαία σε μια οργάνωση. Δε φτάνουν μόνο τα στελέχη, εκτός από αυτά υπάρχει το εξειδικευμένο προσωπικό, το προσωπικό γραφείου και το προσωπικό παραγωγής. Ο κάθε οργανισμός έχει μέσα του τη δική του μοναδική κουλτούρα. Η κουλτούρα είναι ένα σύνολο αντιλήψεων, αξιών και μεθόδων που ακολουθεί ο κάθε οργανισμός και έχουν γίνει δεχτά από όλα τα μέλη της επιχείρησης. Τα στοιχεία αυτά που αποτελούν την κουλτούρα είναι ενσωματωμένα και αυτά στα πληροφοριακά συστήματα.

Μάνατζμεντ

Τα στελέχη της επιχείρησης λαμβάνοντας τις καταστάσεις από την οργάνωση, καθορίζουν στρατηγικές και κατανέμουν τους ανθρώπους και τους οικονομικούς πόρους, έτσι ώστε να μπορέσουν να πετύχουν οι στρατηγικές αυτές για την αντιμετώπιση των προβλημάτων. Θα πρέπει τα στελέχη να αντιλαμβάνονται τις καταστάσεις που έχει να αντιμετωπίσει η οργάνωση και να μπορέσουν να ενεργήσουν κατάλληλα για τη γρήγορη και αποτελεσματική επίλυση των τυχόν προβλημάτων στην οργάνωση.

Εκτός από αυτό όμως τα στελέχη θα πρέπει να δημιουργούν καινούρια προϊόντα και υπηρεσίες όπως επίσης να αλλάζουν και την οργάνωση μερικές φορές. Ένα μέρος του μάνατζμεντ είναι να καθοδηγείται από καινούριες γνώσεις και πληροφορίες.

Οι ρόλοι και οι αποφάσεις των στελεχών διαφέρουν ανάλογα με τα επίπεδα. Τα ανώτερα στελέχη παίρνουν μακροπρόθεσμες αποφάσεις, τα μεσαία στελέχη εφαρμόζουν τα σχέδια των ανώτερων στελεχών και τα λειτουργικά στελέχη ασχολούνται με τις καθημερινές εργασίες της επιχείρησης. Και τα τρία επίπεδα του μάνατζμεντ είναι δημιουργικά και ικανά να βρίσκουν τις κατάλληλες λύσεις σε διάφορα προβλήματα. Το κάθε επίπεδο έχει διαφορετικές ανάγκες για πληροφόρηση και διαφορετικές απαιτήσεις από τα πληροφοριακά συστήματα.

Τεχνολογία

Η τεχνολογία της πληροφορίας είναι ένα από τα πιο σημαντικά εργαλεία που χρησιμοποιούν τα στελέχη για την επίλυση των προβλημάτων τους. Οι υπολογιστές είναι οι συσκευές οι οποίες χρησιμοποιούνται για δραστηριότητες εισόδου, επεξεργασίας και εξόδου ενός πληροφοριακού συστήματος. Οι υπολογιστές αυτοί αποτελούνται από τη μονάδα επεξεργασίας και από συσκευές εισόδου, εξόδου και αποθήκευσης. Το λογισμικό που χρησιμοποιούν οι υπολογιστές περιέχει διάφορες προκαθορισμένες εντολές προγραμμάτων που έχουν τη δυνατότητα να ελέγχουν, και να συντονίζουν τα στοιχεία στο πληροφοριακό σύστημα. Η τεχνολογία της αποθήκευσης αποτελείται από φυσικά μέσα όπως μαγνητικοί δίσκοι, οπτικοί δίσκοι και ταινίες όπου αποθηκεύει τα δεδομένα και το λογισμικό το οποίο δείχνει την κατεύθυνση την οποία πρέπει να ακολουθήσουν τα δεδομένα στα μέσα αποθήκευσης.

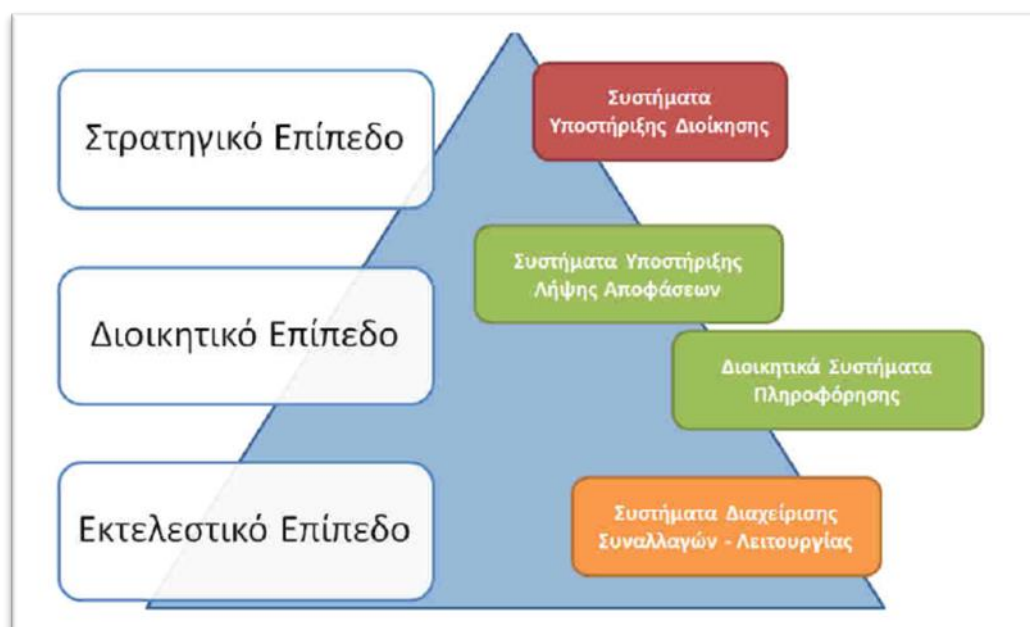
Η τεχνολογία της επικοινωνίας περιέχει και αυτή φυσικές συσκευές και λογισμικό. Συνδέει τα υλικά και κάνει μεταφορά των δεδομένων από τη μια θέση στην άλλη. Οι υπολογιστές και ο λοιπός εξοπλισμός έχουν τη δυνατότητα να συνδέονται στο δίκτυο, να επικοινωνούν μεταξύ τους και να μοιράζονται δεδομένα. Τέτοιου είδους τεχνολογίες, όπου γίνεται η κοινή χρήση δεδομένων, οδηγούν στην υποδομή της τεχνολογίας πληροφοριών (Information Technology). Η υποδομή αυτή είναι το θεμέλιο στο οποίο η επιχείρηση θα αναπτύξει το δικό της πληροφοριακό σύστημα. Κάθε επιχείρηση ή οργανισμός θα πρέπει να σχεδιάζει και να διαχειρίζεται με προσοχή την υποδομή αυτή με σκοπό να έχει τις υπηρεσίες που χρειάζεται για να γίνονται σωστά οι λειτουργίες στο πληροφοριακό σύστημα.

Πληροφοριακά Συστήματα Διοίκησης

Στις σύγχρονες επιχειρήσεις/οργανισμούς απαντώνται διάφορα είδη Πληροφοριακών Συστημάτων (ΠΣ) ή/και Συστημάτων Διαχείρισης Επιχειρησιακών Διεργασιών (ΣΔΕΔ). Κάθε είδος έχει ιδιαίτερα χαρακτηριστικά και αποσκοπεί στην

εξυπηρέτηση των ιδιαίτερων πληροφοριακών αναγκών των διαφόρων επιπέδων και τμημάτων μιας επιχείρησης ή ενός οργανισμού.

Τα Πληροφοριακά Συστήματα Διοίκησης ή ΠΣΔ (MIS) είναι μια κατηγορία λογισμικών ή Ολοκληρωμένων Πληροφοριακών Συστημάτων που εξυπηρετούν όλες εκείνες τις λειτουργίες που αφορούν τη διοίκηση μιας Επιχείρησης ή ενός Οργανισμού. Οι πιο συνήθεις λειτουργίες αυτών των συστημάτων είναι η οργάνωση, ο προγραμματισμός, ο συντονισμός και ο έλεγχος των διαφόρων τμημάτων, η λήψη αποφάσεων και η εποπτεία του προϋπολογισμού.



Πληροφορικά Συστήματα Διοίκησης σε σχέση με το επίπεδο του οργανισμού που εξυπηρετούν

Συστήματα Διαχείρισης Επιχειρησιακών Πόρων – ERPs

Τα Συστήματα Διαχείρισης Επιχειρησιακών Πόρων ή ΣΔΕΠ (Enterprise Resource Planning – ERPs) είναι λογισμικά διαχείρισης και δημιουργίας πλάνων αξιοποίησης των διαθέσιμων πόρων μιας επιχείρησης. Επιτρέπουν σε μια επιχείρηση ή έναν οργανισμό μέσα από ένα ολοκληρωμένο σύστημα να διαχειρίζεται τις δραστηριότητές του, να ελέγχει τις αποθήκες και τις παραγγελίες, να διαχειρίζεται θέματα ανθρώπινου δυναμικού και να αυτοματοποιεί όλες τις εργασίες γραφείου.

Επίσης, οι επιχειρήσεις μέσω τέτοιων συστημάτων μπορούν και παρακολουθούν το νομικό πλαίσιο, το λογιστήριο και γενικότερα τα οικονομικά. Επιπλέον, σε πολλές περιπτώσεις έχουν τη δυνατότητα να κάνουν και προβλέψεις.

Στο βασικό πυρήνα ενός ERP περιλαμβάνονται λειτουργίες για τα οικονομικά, τις αγορές, τις πωλήσεις, τις αποθήκες και τις αναφορές. Εκτός αυτών, στο βασικό πυρήνα περιλαμβάνονται και η διαχείριση αξιόγραφων και οι τιμοκατάλογοι με πρόβλεψη για πολιτική εκπτώσεων και ανατιμήσεων.



Σημαντικές λειτουργίες η οποία περιλαμβάνονται σε ένα ERP είναι η παραγωγή και η κοστολόγηση. Η **Παραγωγή** είναι ένα κομμάτι που αφορά και τα Συστήματα Διαχείρισης Εφοδιαστικής Αλυσίδας. Τα ΣΔΕΠ που εξυπηρετούν και αυτό το κομμάτι της επιχειρηματικής δραστηριότητας καλύπτουν πλήρως τις απαιτήσεις της παραγωγής ξεκινώντας από τις τεχνικές προδιαγραφές, την προμήθεια των πρώτων υλών. Επιπλέον διαχειρίζονται με ιδιαίτερη μέριμνα τον προγραμματισμό παραγωγής και τη σύνδεση του με τον προγραμματισμό πωλήσεων και τις παραγγελίες. Αντίστοιχα σημαντικό είναι το κομμάτι της **Κοστολόγησης** το οποίο συνδέεται τόσο με την Παραγωγή, όσο και με το δίκτυο διανομής ενώ πρέπει να επικοινωνεί και με το κομμάτι του Λογιστηρίου. Υπάρχουν και μεμονωμένα

λογισμικά τα οποία εξειδικεύονται στις επί μέρους εργασίες που περιλαμβάνει ένα ΣΔΕΠ. Η επιλογή είναι θέμα της ίδια της επιχείρησης και εξαρτάται από το μέγεθός της, από το οικονομικό της μέγεθος και από την έκταση των δραστηριοτήτων της.

Συστήματα Διαχείρισης Εφοδιαστικής Αλυσίδας – SCM

Με τον όρο **Εφοδιαστική Αλυσίδα** ή EA (Supply Chain) εννοείται η ροή υλικών, πληροφοριών και υπηρεσιών από τους προμηθευτές πρώτων υλών μέσα από τα εργοστάσια και τις αποθήκες, στους τελικούς πελάτες.

Εφοδιαστική Αλυσίδα Παραγωγής



Διαχείριση της Εφοδιαστικής Αλυσίδας (Supply Chain Management - SCM) καλείται ο σχεδιασμός, η οργάνωση, και ο συντονισμός όλων των δραστηριοτήτων της Εφοδιαστικής Αλυσίδας. Τα Συστήματα Διαχείρισης Εφοδιαστικής Αλυσίδας είναι λογισμικό, το οποίο περιλαμβάνει εργαλεία ή ενότητες που χρησιμοποιούνται για την εκτέλεση των συναλλαγών της εφοδιαστικής αλυσίδας, τη διαχείριση των σχέσεων με τους προμηθευτές και τον έλεγχο που συνδέεται με τις επιχειρηματικές διαδικασίες.



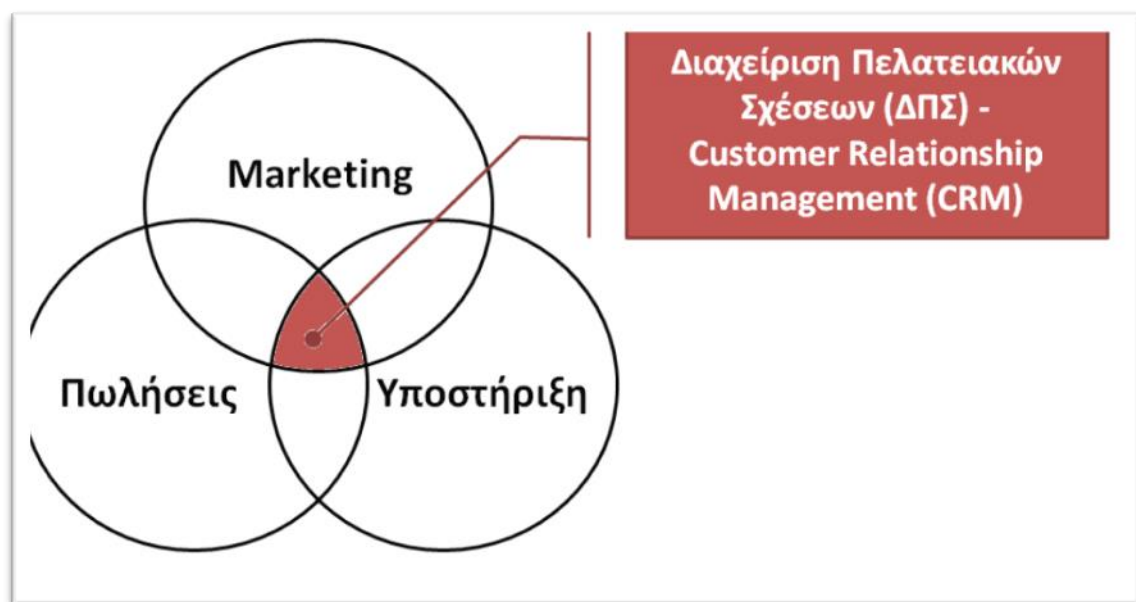
Ουσιαστικά, οι εφοδιαστικές αλυσίδες είναι στην πραγματικότητα δίκτυα. Τα δίκτυα αυτά αποτελούνται από συνδεδεμένες και αλληλεξαρτώμενες οργανώσεις που λειτουργούν από κοινού σε ένα κλίμα συνεργασίας, για να ελέγξουν, να διευθύνουν και να βελτιώσουν την ροή των υλικών και των πληροφοριών από τους προμηθευτές στους τελικούς χρήστες.

Τα δίκτυα αυτά έχουν συνήθως έναν αυθαίρετο αριθμό επίπεδων και συμμετέχουν σε αυτά μία ή περισσότερες επιχειρήσεις-κόμβοι. Ένα δίκτυο μπορεί να περιλαμβάνει προμηθευτές, μεσάζοντες αποθήκες, εταιρείες παροχής υπηρεσιών Third Party Logistics (3PL), κέντρα διανομής και πελάτες. Για παράδειγμα, στον κλάδο της βιομηχανίας κατασκευής Ηλεκτρονικών Υπολογιστών όταν γίνεται λόγος για κανάλια εφοδιαστικής αλυσίδας, νοείται η πορεία που ακολουθούν οι Ηλεκτρονικοί Υπολογιστές από τη στιγμή που δοθεί η παραγγελία και βγει από την παράγωγη μέχρι να φτάσει στον αντιπρόσωπο και από εκεί στον πελάτη.

Συστήματα Διαχείρισης σχέσεων με Πελάτες – CRM Systems

Η Διαχείριση Πελατειακών Σχέσεων (Customer Relationship Management - CRM) είναι μία ευρέως εφαρμοσμένη στρατηγική για την διαχείριση των

αλληλεπιδράσεων μιας επιχείρησης με τους υπάρχοντες αλλά και τους μελλοντικούς πελάτες της. Στόχος της στρατηγικής αυτής είναι η αύξηση των κερδών της επιχείρησης μέσω της αναζήτησης, του εντοπισμού, της κατανόησης, της πρόβλεψης και της διαχείρισης των αναγκών και προτιμήσεων των πελατών της, σημερινών και μελλοντικών. Οι τεχνολογίες που υποστηρίζουν τη φιλοσοφία CRM περιγράφονται από τον όρο Συστήματα Διαχείρισης Σχέσεων με Πελάτες - CRM Systems. Αποτελούν εξέλιξη του παραδοσιακού μάρκετινγκ και των πωλήσεων με σκοπό να βοηθήσουν τις επιχειρήσεις να αντεπεξέλθουν στις ραγδαίες αλλαγές που συμβαίνουν παγκόσμια σε οικονομικό και επιχειρησιακό επίπεδο.



Διαχείριση Πελατειακών Σχέσεων

Τα συστήματα CRM είναι αυτό που κυριολεκτικά δηλώνει το όνομα τους: Πληροφοριακά Συστήματα μέσω των οποίων μια επιχείρηση κατανοεί και εξυπηρετεί καλύτερα τις ανάγκες και τις προτιμήσεις ενός παλαιού ή ενός μελλοντικού πελάτη της.

Στόχοι ενός συστήματος ΔΠΣ (CRM) είναι:

- η πλήρης και σε βάθος κατανόηση των αναγκών και των επιθυμιών των πελατών

- η προσέλκυση νέων πελατών μέσω της προσαρμογής των υπηρεσιών της επιχείρησης στις ανάγκες του εκάστοτε πελάτη
- η βελτίωση του ποσοστού διατήρησης των πελατών μέσω της διαρκούς και αμφίδρομης επικοινωνίας και της αυξημένης αλληλεπίδρασης των πελατών με την επιχείρηση.
- η εμπλοκή των ήδη υπάρχοντων πελατών σε περισσότερο κερδοφόρες για την επιχείρηση δραστηριότητες

Οι παραπάνω στόχοι επιτυγχάνονται μέσα από:

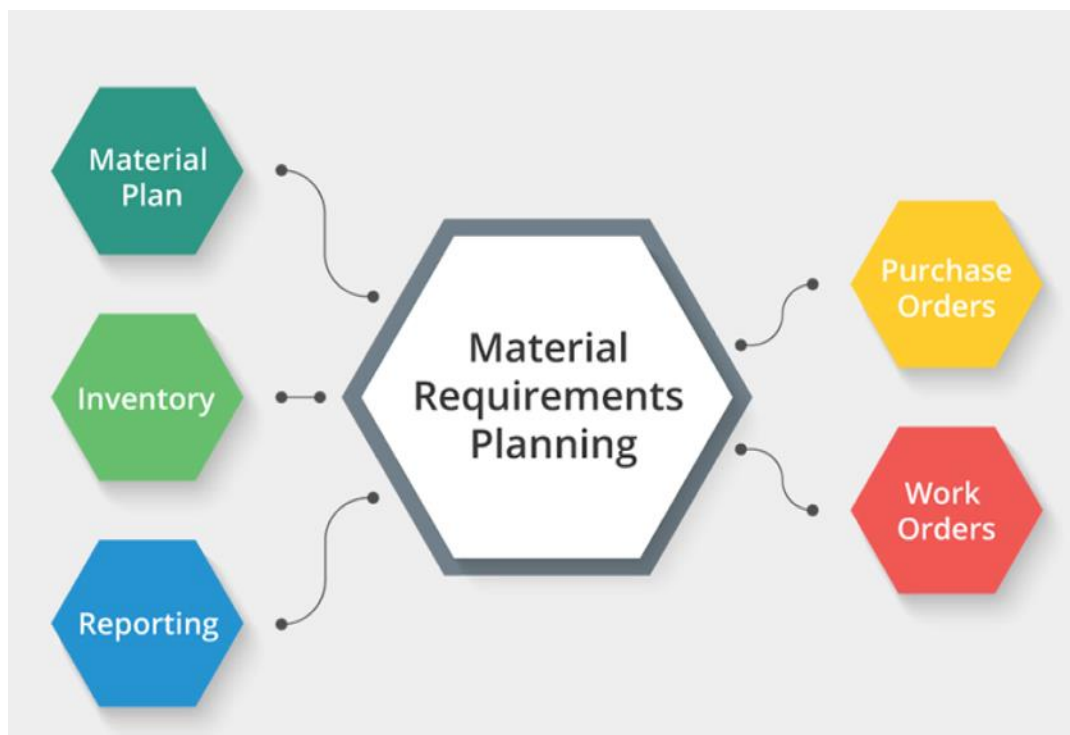
- τη συγκέντρωση, αποθήκευση και οργάνωση στοιχείων για τον πελάτη
- την επεξεργασία των στοιχείων αυτών (πληροφοριών) και τη μετατροπή τους σε γνώση
- τη διάχυση της γνώσης αυτής σε όλους τους εργαζόμενους της επιχείρησης
- την ένταξη των αναγκών, των προτιμήσεων και των επιθυμιών των πελατών στο σχεδιασμό νέων προϊόντων ή υπηρεσιών της επιχείρησης

Μέσα από αυτές τις διαδικασίες, τα Συστήματα ΔΠΣ (CRM) βοηθούν τις επιχειρήσεις να γνωρίσουν τους πελάτες τους καλύτερα και να διαχειριστούν αποτελεσματικότερα αυτή τη γνώση αξιοποιώντας τη συνολική εμπειρία των πελατών τους, αυξάνοντας τα έσοδα και τα κέρδη τους. Επιτυγχάνουν, ταυτόχρονα, την αύξηση της ικανοποίησης των πελατών τους. Πολλές φορές το Σύστημα Διαχείρισης Πελατειακών Σχέσεων (CRM System) αποτελεί κομμάτι του Συστήματος Διαχείρισης Επιχειρησιακών Πόρων (ERP) της επιχείρησης.

Συστήματα Διαχείρισης Παραγωγής - MRP

Τα **Συστήματα Διαχείρισης Παραγωγής** ή αλλιώς **Συστήματα Προγραμματισμού Απαιτήσεων Υλικών** (Material Requirements Planning - MRP) είναι επιφορτισμένα με τη διαχείριση των αποθεμάτων των υλικών, των εξαρτημάτων (και των

ενδιαμέσων προϊόντων της αλυσίδας παραγωγής) και κατ' επέκταση με ένα ιδιαίτερα σημαντικό κομμάτι του προγραμματισμού της παραγωγής (production planning).



Ένα σύστημα MRP είναι επιφορτισμένο να:

- Διασφαλίζει ότι οι απαιτούμενες πρώτες ύλες και εξαρτήματα (ενδιάμεσα προϊόντα) θα είναι διαθέσιμα για παραγωγή και ότι τα προϊόντα θα είναι διαθέσιμα προς παράδοση στους πελάτες
- Διατηρεί το χαμηλότερο δυνατό επίπεδο αποθέματος
- Προγραμματίζει τις δραστηριότητες της παραγωγής, τα χρονοδιαγράμματα αποστολών και τις διαδικασίες προμηθειών

Ένα σύστημα MRP εξασφαλίζει καλύτερο έλεγχο παραγωγής, ενώ επιτυγχάνει ακριβέστερη και εγκυρότερη πληροφόρηση. Ταυτόχρονα βοηθάει στη διατήρηση μικρότερων αποθεμάτων, μειώνει την απαξίωση τους, αυξάνει την αξιοπιστία της επιχείρησης, αυξάνει την ανταπόκριση στις απαιτήσεις της αγοράς και επιτυγχάνει σημαντική μείωση του κόστους παραγωγής. Επειδή, πολύ συχνά, συγχέονται τα

όρια των συστημάτων ERP και MRP θα πρέπει να αναφερθεί, ότι οι περισσότερες από τις λειτουργίες ενός Συστήματος Διαχείρισης Παραγωγής (MRP) είναι δυνατόν να υλοποιηθούν από ένα σύστημα Ενδοεπιχειρησιακού Σχεδιασμού - ERP. Αυτό που διαφοροποιεί ουσιαστικά ένα σύστημα MRP είναι η μεγαλύτερη ευελιξία και ο λεπτομερέστερος έλεγχος της παραγωγικής διαδικασίας.

Συστήματα διαχείρισης πωλήσεων και αγορών

Αποτελούν ίσως τα πιο παλιά συστήματα διαχείρισης των βασικών δραστηριοτήτων μιας επιχείρησης. Εμφανίζονται συνήθως με το σύστημα διαχείρισης της αποθήκης που προφανώς είναι άρρηκτα συνδεδεμένο με τις πωλήσεις και τις αγορές. Στα σύγχρονα ERP συστήματα, η διαχείριση πωλήσεων και αγορών είναι ένα μέρος του συνόλου και διασυνδέεται και με όλα τα υπόλοιπα υποσυστήματα.

Σε γενικές γραμμές ένα σύστημα διαχείρισης πωλήσεων και αγορών πρέπει να περιλαμβάνει κωδικοποίηση και κατηγοριοποίηση αγαθών, πελατών, πωλητών και προμηθευτών. Επίσης πρέπει να περιλαμβάνει ιστορικό παρακολούθησης παραγγελίας, υποστήριξη διαδικασιών επιλεκτικής ακύρωσης παραγγελίας, έγκριση αποστολής, ακόμη και διαδικασίες ποιοτικού ελέγχου και σύστημα πρόβλεψης πωλήσεων και αγορών. Σε μερικές περιπτώσεις υπάρχει και σύστημα παρακολούθησης προσφορών. Επίσης πρέπει να υποστηρίζεται η έκδοση παραστατικών. Τέτοια μεμονωμένα συστήματα είναι πολύ χρήσιμα σε μικρές εταιρίες οι οποίες έχουν σχετικά μικρό κύκλο εργασιών και μικρό αριθμό απασχολούμενων. Πολλές φορές τα συναντάμε με τις ονομασίες, «Σύστημα διαχείρισης εμπορικών συναλλαγών» ή «Σύστημα Λιανικών Πωλήσεων».

Εντούτοις μπορούν να παρουσιαστούν διαδικτυακά τμήματα ERP, τα οποία αφορούν δικτυακά σημεία αγορών (ή πωλήσεων αν το δούμε από την πλευρά του καταστήματος). Τέτοια διαδικτυακά λογισμικά είναι αυτά που αφορούν αγορές, είτε από καταστήματα e-shop, είτε εισιτηρίων (αεροπορικά, τρένων, γηπέδων,

συναυλιών) κλπ. Αυτά τα μεμονωμένα σημεία συνδέονται με ένα ευρύτερο σύστημα το οποίο υλοποιεί τις περισσότερες από τις λειτουργίες ενός ERP.

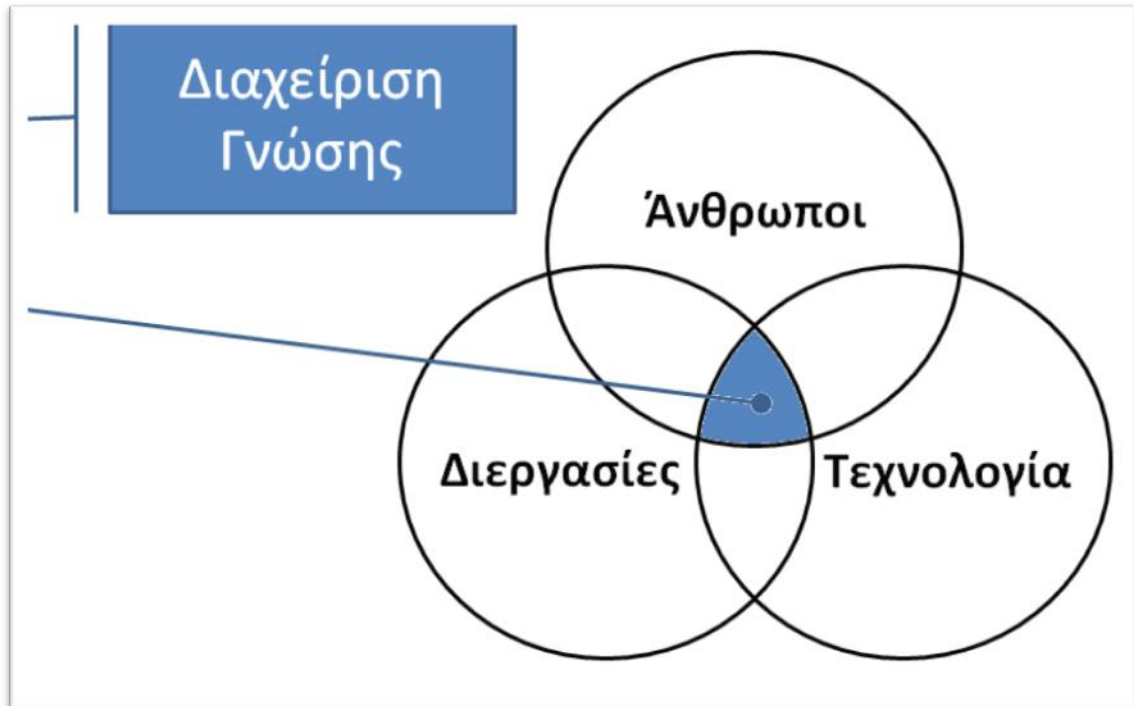
Συστήματα Διαχείρισης Γνώσης

Η Επιχειρηματική (εταιρική) Γνώση περιλαμβάνει τις ικανότητες και εμπειρίες των στελεχών της εταιρίας, τις δυνατότητες καινοτομίας και δημιουργικότητας, τις βέλτιστες επιχειρηματικές πρακτικές, τις πατέντες, τη γνώση που η επιχείρηση έχει συσσωρεύσει για την αγορά και τους πελάτες της, καθώς και τη γνώση για τους ανταγωνιστές της. Η Διαχείριση Γνώσης (ΔΓ) περιλαμβάνει μια σειρά από στρατηγικές και πρακτικές που χρησιμοποιούνται για τον εντοπισμό, την καταγραφή, την οργάνωση, την αξιολόγηση, την αξιοποίηση και τον διαμοιρασμό του «νοητικού κεφαλαίου» (γνώσης) μιας επιχείρησης ή ενός οργανισμού με στόχο τη βελτίωση της απόδοσης και της ανταγωνιστικότητας τους.

Η Διαχείριση της Γνώσης είναι αποτέλεσμα της συμπληρωματικής αλληλεπίδρασης μεταξύ τριών βασικών στοιχείων:

- **Ανθρώπινο Δυναμικό:** η γνώση βασίζεται στους ανθρώπους μιας και αυτοί αποτελούν βάση για τη δημιουργία γνώσης.
- **Τεχνολογία:** Αξιοπίστη τεχνολογική υποδομή για την υποστήριξη της Διαχείρισης Γνώσης
- **Διαδικασίες:** πρόκειται για τις διαδικασίες και λειτουργίες μέσω των οποίων πραγματοποιείται η συλλογή, επιλογή, μεταφορά και διαχείριση της γνώσης σε μια επιχείρηση / οργανισμό.

Μάλιστα τα τρία παραπάνω στοιχεία δεν είναι μόνο αναγκαία αλλά και συμπληρωματικά μεταξύ τους, όσον αφορά στη Διαχείριση Γνώσης, μιας και όπως φαίνεται στο σχήμα που ακολουθεί, επικαλύπτονται εν μέρει.



Η Διαχείριση Γνώσης πραγματοποιείται από Συστήματα Διαχείρισης Γνώσης (Knowledge Management Systems). Πρόκειται για κάθε είδος Πληροφοριακού Συστήματος που σαν στόχο του έχει τη διευκόλυνση των βασικών διεργασιών Διαχείρισης Γνώσης.

Παραδείγματα βασικών διεργασιών Διαχείρισης Γνώσης είναι η δημιουργία, η σύλληψη και η κωδικοποίηση, η αποθήκευση και η ανάκληση, η κοινή χρήση, η διανομή, ο εντοπισμός πηγών γνώσης, η εξόρυξη «κρυμμένης» γνώσης, κ.ά.

Οι Laudon & Laudon (2002) κατέταξαν τα σημαντικότερα πληροφοριακά συστήματα που διευκολύνουν τις βασικές διεργασίες διαχείρισης της γνώσης στις εξής κατηγορίες:

- Συστήματα Αυτοματισμού Γραφείου, τα συστήματα αυτά χρησιμοποιούνται για τη διανομή της γνώσης
- Συστήματα Γνωστικής Εργασίας που χρησιμοποιούνται για τη δημιουργία γνώσης

- Συστήματα Συνεργασίας Ομάδων, τα οποία χρησιμοποιούνται για τη κοινή χρήση της γνώσης και
- Εφαρμογές Τεχνητής Νοημοσύνης, όσα συστήματα χρησιμοποιούνται για τη σύλληψη και κωδικοποίηση της γνώσης.

Τα συστήματα διαχείρισης γνώσης επιτελούν τις παρακάτω λειτουργίες:

- συσσώρευση περιεχομένου τόσο από εσωτερικές όσο και από εξωτερικές πηγές
- ομαδοποίηση του περιεχομένου με τη χρήση ταξινομιών
- πραγματοποίηση αναζητήσεων
- εύκολο και αποτελεσματικό εντοπισμό γνώσης
- δυνατότητα δημιουργίας αναφορών

Σκοπός ενός Συστήματος Διαχείρισης Γνώσης είναι:

- η βελτίωση της απόδοσης της επιχείρησης
- η δημιουργία συγκριτικού πλεονεκτήματος για την επιχείρηση
- η προαγωγή της καινοτομίας
- ο διαμοιρασμός / διάχυση της γνώσης μέσα στην επιχείρηση
- η αποδοτικότερη ενσωμάτωση της γνώσης

Γεωγραφικά Συστήματα Πληροφοριών - GIS

Τα **Γεωγραφικά Συστήματα Πληροφορικών ή ΓΠΣ (GIS)** είναι ψηφιακά συστήματα, τα οποία μπορούν να ενσωματώσουν, να αναλύσουν και να παρουσιάσουν πληροφορίες γεωγραφικά συσχετισμένες. Σε τέτοια συστήματα αποθηκεύονται, συνδέονται και συσχετίζονται πληροφορίες, δεδομένα και γεωγραφικά στοιχεία.

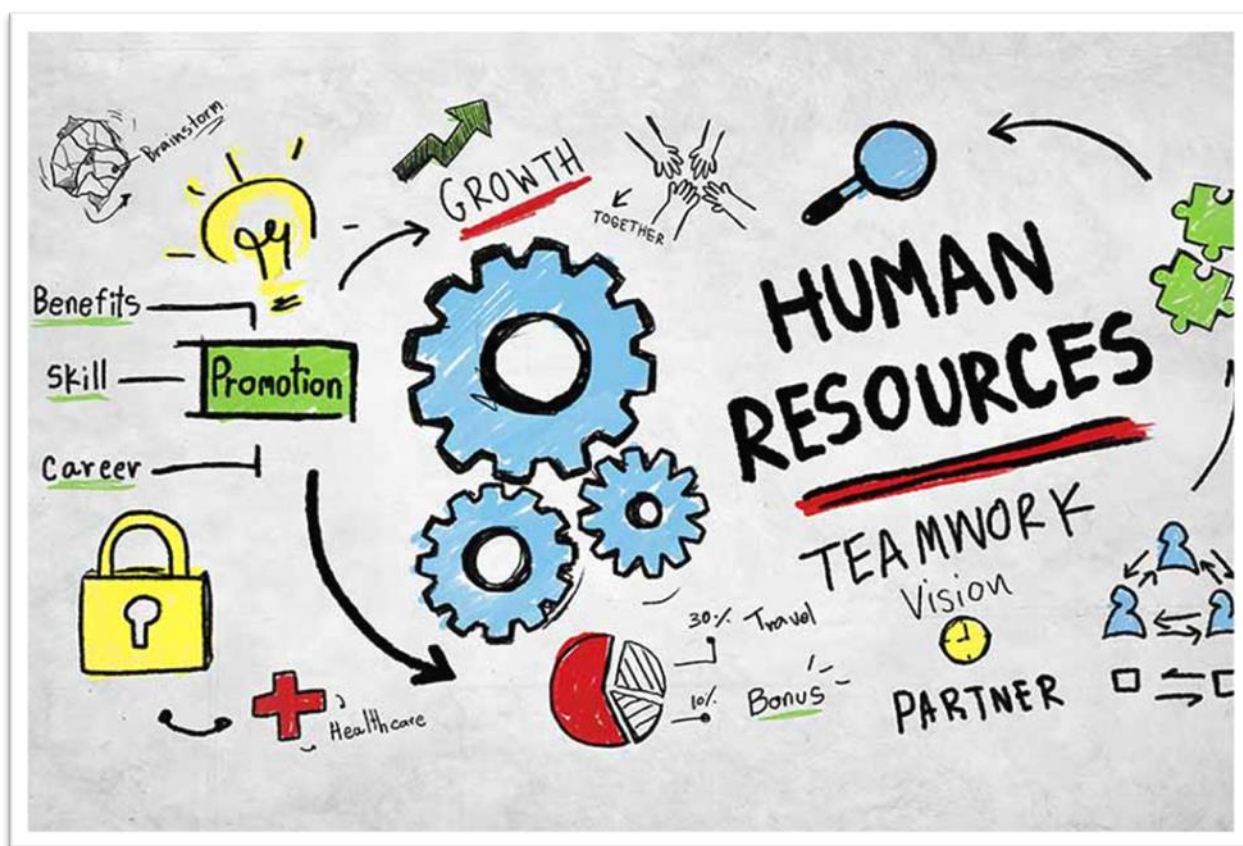
Στη συνέχεια αυτά τα συστήματα έχουν την ικανότητα να τα απεικονίζουν, να τα αναλύουν και να τα διαχειρίζονται. Χρησιμοποιούνται από τις επιχειρήσεις ποικιλοτρόπως. Για παράδειγμα μπορούν να έχουν πλήρη εικόνα των παραγγελιών τους, μπορούν να δίνουν τη δυνατότητα στους πελάτες τους να γνωρίζουν σε πιο σημείο βρίσκεται η παραγγελία τους. Επίσης μπορούν να έχουν εικόνα του δικτύου πωλήσεων, αποστάσεων, ειδικών συνθηκών κλπ.



Ένας οργανισμός μπορεί να έχει εικόνα των ιδιοτεροτήτων μιας γεωγραφικής περιοχής. Για παράδειγμα ένας οργανισμός που ασχολείται με την καταγραφή των κοιτασμάτων νερού μιας αγροτικής ή μιας αστικής περιοχής χρησιμοποιεί τέτοια συστήματα για να μπορεί να προτείνει με ακρίβεια τα σημεία γεώτρησης ή τα σημεία διασύνδεσης ποτιστικών συστημάτων. Μια εταιρία ή συνεταιρισμός ταξί μπορεί μέσω ενός τέτοιου συστήματος να διαχειρισθεί το στόλο της, να χρονοπρογραμματίσει και να εξυπηρετήσει τις απαιτήσεις μετακίνησης των πελατών της κ.λπ.

Συστήματα Διαχείρισης Ανθρώπινου Δυναμικού- HRMS

Ως Διαχείριση Ανθρώπινου Δυναμικού ορίζεται «το σύνολο των καλά σχεδιασμένων και πλήρως ελεγχόμενων δράσεων και ενεργειών μιας επιχείρησης για την οικοδόμηση και διατήρηση της σχέσης μεταξύ των εργαζόμενων και της επιχείρησης, που αποσκοπεί στην κάλυψη τόσο των στόχων της επιχείρησης όσο και προσδοκιών / αναγκών των εργαζομένων της». Όπως περιγράφεται στο σχήμα που ακολουθεί, οι HRM δράσεις που αναπτύσσει μια επιχείρηση / οργανισμός και συνθέτουν την HR



στρατηγική της / του περιλαμβάνουν μεταξύ άλλων, δράσεις επιλογής υποψηφίων, στρατολόγησης, ανάπτυξης και αξιοποίησης ταλέντων, εκπαίδευσης, αξιολόγησης, επιβράβευσης και αμοιβών, διαχείρισης της απώλειας εργαζομένων κ.λπ

Συνεπώς ο όρος «Διαχείριση Ανθρώπινου Δυναμικού» δεν περιγράφει μόνο το συντονισμό του ανθρώπινου δυναμικού, αλλά αφορά στην αποδοτική συνδιαλλαγή μεταξύ επιχείρησης και εργαζομένων στοχεύοντας στην επίτευξη των συμφερόντων

και των δύο παράλληλα. Η διαχείριση ανθρώπινου δυναμικού ορίζεται ως ένα σύνολο ενεργειών, στρατηγικών (π.χ. προγραμματισμός ανθρωπίνων πόρων, σχεδιασμός συστημάτων αμοιβής) αλλά και λειτουργιών (π.χ. επιλογή υποψηφίων, εκπαίδευση κ.α.), που πρέπει να γίνουν για να μπορέσει η επιχείρηση να αποκτήσει, διατηρήσει και αξιοποιήσει ικανούς εργαζόμενους που θα εκτελούν επιτυχώς και με παραγωγικό τρόπο το έργο τους.

Ένα Σύστημα Διαχείρισης Ανθρώπινου Δυναμικού ή ΣΔΑΔ (Human Resources Management System - HRMS ή Human Resources Information System - HRIS) υλοποιεί τη στρατηγική Διαχείρισης Ανθρώπινου Δυναμικού μιας εταιρίας / οργανισμού χρησιμοποιώντας Τεχνολογίες Πληροφορικής και Επικοινωνιών.



Πολλές φορές τα Συστήματα Διαχείρισης Ανθρώπινου Δυναμικού (ιδιαίτερα αυτά των μεγάλων επιχειρήσεων) είναι ενσωματωμένα στο ERP της επιχείρησης.

Συστήματα Υποστήριξης Αποφάσεων - DSS

Τα Συστήματα Υποστήριξης Αποφάσεων ή ΣΥΑ (στα αγγλικά Decision Support Systems ή DSS) υποστηρίζουν τα στελέχη, που έχουν ως αρμοδιότητα να λαμβάνουν

αποφάσεις. Ο ρόλος τους είναι καθαρά συμβουλευτικός και ένα άτομο που λαμβάνει αποφάσεις πρέπει να επιστρατεύσει και τη δική του γνώση και εμπειρία, για να λάβει μια απόφαση. Αυτό σημαίνει ότι τα Συστήματα Υποστήριξης Αποφάσεων δεν παρέχουν μια έτοιμη λύση σε ένα πρόβλημα λήψης απόφασης, αλλά εφοδιάζουν τους ενδιαφερόμενους να πάρουν αποφάσεις με ποικίλες πληροφορίες.

Ένα ΣΥΑ αποτελείται θεωρητικά από τρία υποσυστήματα για:

- Διαχείριση δεδομένων
- Διαχείριση μοντέλων ανάλυσης για τη λήψη αποφάσεων
- Εποπτεία της επικοινωνίας ανθρώπου- μηχανής

Το πρώτο υποσύστημα στηρίζεται σε διαθέσιμες για την επιχείρηση ή τον οργανισμό βάσεις δεδομένων, τα στοιχεία των οποίων συλλέγονται κυρίως από τα διάφορα συστήματα συναλλαγών της. Για να έχει επιτυχία ένα ΣΥΑ, θα πρέπει εκτός των άλλων να λαμβάνει στοιχεία από βάσεις δεδομένων που εξασφαλίζουν συνοχή και επικαιρότητα των δεδομένων. Το υποσύστημα διαχείρισης μοντέλων επιτρέπει στους χρήστες ενός ΣΥΑ να χρησιμοποιήσουν πρότυπα ανάλυσης, προκειμένου να εξερευνήσουν την επίδραση διαφόρων παραγόντων σε κάποια προσδοκώμενα αποτελέσματα. Η εργασία αυτή αναφέρεται και ως δημιουργία εναλλακτικών σεναρίων. Υπάρχουν πρότυπα για ποσοτική ανάλυση, στατιστική επεξεργασία, προσομοίωση καταστάσεων, διασπορά πιθανοτήτων, γραμμικό προγραμματισμό αλλά και διάφορα μοντέλα βελτιστοποίησης. Δύο δημοφιλή πρότυπα είναι η "what-if" ανάλυση και η αναζήτηση στόχου. Τα πρότυπα αυτά βρίσκονται πλέον ενσωματωμένα και στις διάφορες εφαρμογές υπολογιστικών φύλλων. Τα ΣΥΑ ασχολούνται εξ ορισμού με προβλήματα, για την λύση των οποίων ο επιλύων σε κάποια έκταση, όχι όμως πλήρως, έχει καθορίσει μια κατάλληλη μέθοδο ή στρατηγική.

Ανάλογα με τον τρόπο που προκύπτουν οι αποφάσεις, τα συστήματα αυτά κατατάσσονται σε κατηγορίες, από τα απλά για την ανάκτηση δεδομένων έως και τα πλέον εξελιγμένα, που βασίζονται σε πολύπλοκα μοντέλα ανάλυσης. Τυπικά παραδείγματα επιχειρηματικών δράσεων όπου χρησιμοποιούνται ΣΥΑ είναι οι:

- επενδυτικές αποφάσεις
- προβλέψεις πωλήσεων και αγορών νέων προϊόντων
- προγραμματισμός παραγωγής
- σχεδιασμός πολιτικής μάρκετινγκ
- σχεδιασμός συστημάτων διανομής προϊόντων
- προγραμματισμός προσωπικού

Όπως γίνεται κατανοητό, τα ΣΥΑ στηρίζονται αφενός στην εξόρυξη δεδομένων αφετέρου στη πολυδιάστατη ανάλυση δεδομένων προκειμένου να διευκολύνουν τα στελέχη μιας επιχείρησης ή οργανισμού να λάβουν αποφάσεις σε διάφορα θέματα. Αρκετοί ερευνητές αποκαλούν τη διεργασία αυτή ως παραγωγή επιχειρηματικής ευφυΐας (business intelligence creation).

Σύνοψη

Σε μια σύνοψη της ενότητας, εδώ οι σπουδαστές έρχονται για πρώτη φορά σε επαφή με το αντικείμενο της Διαχείρισης Ανθρώπινου Δυναμικού μαθαίνουν για τα Λειτουργικά και πληροφοριακά συστήματα. Αναφερθήκαμε στα συστήματα πολλών χρηστών και αναλύσαμε τα Πληροφοριακά συστήματα.

Ερωτήσεις ενότητας 4

A. Τι είναι Λειτουργικό σύστημα;

1. Λειτουργικό Σύστημα (Operating System), εκφράζει το σύνολο των προγραμμάτων ενός υπολογιστικού συστήματος το οποίο ελέγχει, διαχειρίζεται και συντονίζει τη λειτουργία των επιμέρους τμημάτων του.

2. Λειτουργικό Σύστημα είναι το υλικό μέρος του υπολογιστή
3. Λειτουργικό Σύστημα είναι το λογισμικό μέρος του υπολογιστή

B. Επιλέξτε τους στόχους ενός λογισμικού

1. Χρήση του υλικού και των περιφερειακών του ΥΣ με αποτελεσματικό / αποδοτικό τρόπο
2. Προστασία των προγραμμάτων και δεδομένων των διαφόρων χρηστών του ΥΣ
3. Όλα τα παραπάνω

Γ. Ποιες είναι βασικές αρμοδιότητες και λειτουργίες του Λειτουργικού Συστήματος;

1. Να λειτουργεί ως ενδιάμεσος (Διεπαφή ή Interface) ανάμεσα στον άνθρωπο και στη μηχανή.
2. Να μεταφέρει εντολές ή/και απαιτήσεις του χρήστη στον Η/Υ.
3. Όλα τα παραπάνω

Δ. Ποιες είναι οι κατηγορίες λειτουργικών συστημάτων;

1. Υποστήριξη ενός ή πολλών χρηστών- Ανοικτό η κλειστό λογισμικό
2. Μεταφέρει εντολές ή/και απαιτήσεις του χρήστη στον Η/Υ.
3. Προστασία των προγραμμάτων και δεδομένων των διαφόρων χρηστών

E. Τι είναι λειτουργικό σύστημα ενός χρήστη πολλών διαδικασιών (single user multitasking)

1. Αυτό το λειτουργικό σύστημα υποστηρίζει έναν χρήστη, με μια λειτουργική διαδικασία
2. Αυτό το λειτουργικό σύστημα υποστηρίζει έναν χρήστη, αλλά πολλές διαφορετικές διαδικασίες.

Z. Τι είναι λειτουργικό σύστημα πολλών χρηστών και πολλών διαδικασιών (multi-user multitasking);

1. Το λειτουργικό σύστημα αυτής της κατηγορίας μπορεί να διαχειρίζεται πολλούς χρήστες και πολλές διαφορετικές διαδικασίες.

2. Το λειτουργικό σύστημα αυτής της κατηγορίας μπορεί να διαχειρίζεται πολλούς χρήστες με ταυτόχρονη διαδικασία .

Η. Τι είναι το λειτουργικό σύστημα ομαδικής επεξεργασίας (batch operating system);

1. Στα λειτουργικά συστήματα αυτά κάθε χρήστης προετοιμάζει τα προγράμματά του και τα υποβάλλει όλα μαζί.

2. Στα λειτουργικά συστήματα αυτά κάθε χρήστης προετοιμάζει το πρόγραμμα του και το υποβάλλει

Θ. Ποιες βασικές αρμοδιότητες και λειτουργίες του Λειτουργικού Συστήματος ;

1. Να λειτουργεί ως ενδιάμεσος (Διεπαφή ή Interface) ανάμεσα στον άνθρωπο και στη μηχανή.

2. Να διαχειρίζεται την Κεντρική Μνήμη (RAM) του συστήματος.

3. Όλα τα παραπάνω

Ι. Ποιες είναι οι κατηγορίες λειτουργικών συστημάτων;

1. Δεν μπορεί να υποστήριξη ένα ή πολλούς χρήστες

2. Ανοικτό η κλειστό λογισμικό

Κ. Τι είναι καταναμημένο λειτουργικό σύστημα (distributed operating system);

1. Το λειτουργικό σύστημα αυτού του είδους εγκαθίσταται σε ένα υπολογιστή

2. Το λειτουργικό σύστημα αυτού του είδους εγκαθίσταται σε πολλούς υπολογιστές, στον καθένα με τη δική του μονάδα επεξεργασίας.

Άσκηση

Σε ένα ΤΕΙ, κατά την ανάλυση του υπάρχοντος Πληροφοριακού Συστήματος έγινε καταγραφή των οντοτήτων. Για τρεις οντότητες από αυτές ισχύουν τα εξής: Κάθε

μάθημα το διδάσκει ένας μόνο καθηγητής. Οι φοιτητές κάνουν εγγραφή σε μαθήματα για να αποκτήσουν το δικαίωμα να τα παρακολουθήσουν. Για τα μαθήματα που παρακολουθούν, στο τέλος του εξαμήνου βαθμολογούνται με γραπτές εξετάσεις.

1. Για τις οντότητες που αναφέρονται στην παραπάνω περιγραφή να σχεδιάσετε το διάγραμμα ER (οντοτήτων - συσχετίσεων).
2. Στο διάγραμμα να καταγράψετε τις ιδιότητες κάθε οντότητας.
3. Προσδιορίστε τον τύπο των σχέσεων μεταξύ των οντοτήτων (1:1, 1:N, N:M).

Βιβλιογραφία

Bryant E.R. ~O'Halloran Rd (2019) Συστήματα υπολογιστών - Μια προσέγγιση από την πλευρά του προγραμματιστή

MONDY R. WAYNE, MARTOCCHIO J. JOSEPH (2018) ΔΙΑΧΕΙΡΙΣΗ ΑΝΘΡΩΠΙΝΟΥ ΔΥΝΑΜΙΚΟΥ.

Grus Joel (2021) Επιστήμη Δεδομένων: Βασικές Αρχές και Εφαρμογές με Python, 2η έκδοση.

Avison David / Fitzgerald Guy(2006) Προηγμένα πληροφοριακά συστήματα: από τη θεωρία στην πράξη

Εκπαιδευτική ενότητα: 5.Θεωρία βάσεων δεδομένων

Σκοπός

Σκοπός αυτής της εκπαιδευτικής ενότητας είναι να μάθει ο σπουδαστής πως οργανώνονται τα δεδομένα μας, τα μοντέλα των βάσεων δεδομένων, καθώς και τη λειτουργία του Συστήματος Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ). Θα μάθει πως να σχεδιάζει μια βάση δεδομένων, για το μοντέλο οντοτήτων-συσχετίσεων. Τελος θα κάνουμε μια εισαγωγή στην Δομημένη Γλώσσα Ερωταπαντήσεων (SQL).

Προσδοκώμενα Αποτελέσματα

Όταν ολοκληρώσετε τη μελέτη του υλικού αυτής της ενότητας θα είστε σε θέση να γνωρίζεται πως να οργανώσετε τα δεδομένα σας, θα έχετε την απαραίτητη γνώση για τα μοντέλα βάσεων δεδομένων και τη λειτουργία του Συστήματος Διαχείρισης Βάσεων Δεδομένων. Επίσης, θα μπορείτε να σχεδιάζετε βάσεις δεδομένων για μοντέλα οντοτήτων-συσχετίσεων και θα γνωρίζετε πως να χρησιμοποιείτε τη Δομημένη Γλώσσα Ερωταπαντήσεων SQL.

Έννοιες-κλειδιά

οργάνωσης δεδομένων: Η οργάνωση δεδομένων είναι η πρακτική της κατηγοριοποίησης και ταξινόμησης δεδομένων για να γίνουν πιο χρήσιμα.

Παρόμοια με έναν φάκελο αρχείων όπου διατηρούμε σημαντικά έγγραφα, θα χρειαστεί να οργανώσετε τα δεδομένα σας με τον πιο λογικό και τακτοποιημένο τρόπο, ώστε εσείς και ο οποιοσδήποτε άλλος να έχει πρόσβαση σε αυτά και να μπορείτε να βρείτε εύκολα αυτό που θέλει ψάχνετε.

Βάσεις δεδομένων : Η βάση δεδομένων είναι η οργάνωση και αποθήκευση πληροφοριών σχετικά με έναν συγκεκριμένο τομέα. Με απλούστερο τρόπο, είναι η ομαδοποίηση δεδομένων που ασχολούνται με το ίδιο θέμα και που πρέπει να αποθηκευτούν για ασφάλεια ή μελλοντική διάσκεψη.

Συστήματα Διαχείρισης Βάσεων Δεδομένων: Τα συστήματα διαχείρισης βάσεων δεδομένων (DBMS) χρησιμοποιούνται για την πρόσβαση και το χειρισμό δεδομένων σε μια βάση δεδομένων. Ένα σύστημα διαχείρισης βάσεων δεδομένων είναι ένα πακέτο λογισμικού που επιτρέπει στους χρήστες να επεξεργάζονται και να ενημερώνουν πίνακες όπως απαιτείται. Τα συστήματα διαχείρισης βάσεων δεδομένων θα συζητηθούν σε μεγαλύτερο βάθος αργότερα.

τύποι βάσεων δεδομένων: Υπάρχουν πολλές επιλογές βάσεων δεδομένων αλλά πριν δουλέψουμε σε καθεμία από αυτές , πρέπει να κατανοήσουμε τη διαφορά μεταξύ σχεσιακών και μη σχεσιακών βάσεων δεδομένων.

NoSQL : Μια NoSQL , ή μη σχεσιακή βάση δεδομένων, επιτρέπει την αποθήκευση και τον χειρισμό μη δομημένων και ημιδομημένων δεδομένων (σε αντίθεση με μια

σχεσιακή βάση δεδομένων, η οποία καθορίζει τον τρόπο σύνθεσης όλων των δεδομένων που εισάγονται στη βάση δεδομένων). Οι βάσεις δεδομένων NoSQL έχουν γίνει δημοφιλείς καθώς οι εφαρμογές Ιστού έχουν γίνει πιο κοινές και πιο περίπλοκες. ΣΔΒΔ, Δομημένη Γλώσσα (SQL), τύποι δεδομένων.

Βάσεις δεδομένων OLTP: Μια βάση δεδομένων OLTP είναι μια γρήγορη, αναλυτική βάση δεδομένων που έχει σχεδιαστεί για μεγάλο αριθμό συναλλαγών που εκτελούνται από πολλούς χρήστες.

Εκπαιδευτική υποενότητα : 5.1 Μέθοδοι οργάνωσης δεδομένων



ΤΙ ΕΙΝΑΙ Η ΟΡΓΑΝΩΣΗ ΔΕΔΟΜΕΝΩΝ;

Η οργάνωση δεδομένων είναι ένα σύνολο αρχείων που συνομιλούν μεταξύ τους και αποθηκεύουν έναν τεράστιο όγκο διαφορετικών τύπων δεδομένων.

Για παράδειγμα: ονόματα, έγγραφα, πληρωμές, τιμολόγια, αποδείξεις, διευθύνσεις, υπηρεσίες, πελάτες, email, αριθμοί τηλεφώνου, μεταξύ χιλιάδων άλλων πραγμάτων και σημαντικά δεδομένα.

Αλλά φυσικά αυτά τα δεδομένα δεν χάνονται στον υπολογιστή ούτως ή άλλως. Διαμορφώνονται και διαχειρίζονται μαζί, μέσω μιας από τις γλώσσες προγραμματισμού, που μπορεί να είναι SQL, Javascript, PL/SQL ή κάποια άλλη.

Αυτό το σύνολο οργανωμένων δεδομένων ονομάζεται «βάση δεδομένων».

Με άλλα λόγια, μια βάση δεδομένων είναι η μορφή οργάνωσης δεδομένων ενός φυσικού ή νομικού προσώπου.

Είναι ένα μέρος για την ασφαλή αποθήκευση πληροφοριών της εταιρείας στο cloud.

ΣΕ ΤΙ ΧΡΗΣΙΜΕΥΕΙ Η ΟΡΓΑΝΩΣΗ ΔΕΔΟΜΕΝΩΝ;

Σε έναν όλο και πιο παγκοσμιοποιημένο και ψηφιοποιημένο κόσμο, είναι σημαντικό για την εταιρεία να έχει τον πλήρη έλεγχο και διαχείριση των πληροφοριών της.

Η οργάνωση δεδομένων χρησιμεύει για την επιτάχυνση των διαδικασιών εντός και εκτός της εταιρείας.

Αυτό μπορεί να είναι μια από τις διαφορές σας για να επιτύχετε μεγαλύτερη επιτυχία και προβολή στην επιχειρηματική αγορά που είναι εξαιρετικά ανταγωνιστική.

Έχοντας αυτές τις πληροφορίες καλή διαχείριση και οργάνωση, είναι δυνατό να αναπτυχθούν καλύτερα σχεδιασμένες και πιο αποτελεσματικές στρατηγικές για να διασφαλιστεί η καλύτερη λειτουργία της εταιρείας.

Είναι επίσης σημαντικό να φιλτράρετε τις πληροφορίες που θα βρίσκονται μέσα στην οργάνωση της βάσης δεδομένων, επειδή η καλύτερη ποιότητα πληροφοριών βελτιώνει επίσης την απόδοση των δραστηριοτήτων.

Από την οργάνωση των δεδομένων είναι επίσης δυνατό να αναλυθεί η κατεύθυνση προς την οποία κινείται η επιχείρησή σας.

Ο διαχειριστής, κατά την ανάλυση αυτών των δεδομένων, θα είναι σε θέση να διορθώσει την κατάσταση ή να εντείνει τα μέτρα για να συνεχίσει την πορεία που ακολουθεί η εταιρεία.

Εκτός από την ανάλυση των κατευθύνσεων που ακολουθεί η εταιρεία, είναι επίσης δυνατή η ανάλυση άλλων τύπων πληροφοριών που μπορούν να βοηθήσουν την εταιρεία.

Όπως για παράδειγμα:

Ο αντίκτυπος που προκαλούν οι ενέργειες Μάρκετινγκ της εταιρείας και αν είναι πραγματικά αποτελεσματικές ή όχι.

Ανάλυση εισοδήματος εργαζομένων.

Ανάλυση ποιότητας υπηρεσιών και πελατειακής σχέσης.

Πίνακας κύριων πελατών

Ανάλυση αγοράς

Επαλήθευση αποτελεσματικότερων και λιγότερο αποτελεσματικών στρατηγικών

Αυτά είναι μερικά παραδείγματα των αμέτρητων δυνατοτήτων, μεταξύ άλλων, που μπορεί να προσφέρει στην εταιρεία μια καλή οργάνωση δεδομένων.

ΠΟΣΟ ΣΗΜΑΝΤΙΚΗ ΕΙΝΑΙ Η ΟΡΓΑΝΩΣΗ ΔΕΔΟΜΕΝΩΝ ΓΙΑ ΤΗΝ ΕΤΑΙΡΕΙΑ;

Όπως αναφέρθηκε προηγουμένως, η βάση δεδομένων (οργάνωση δεδομένων) αποθηκεύει και διαχειρίζεται όλες τις πληροφορίες μιας εταιρείας.

Η σημασία της ύπαρξης αυτού του τύπου διαχείρισης σε μια εταιρεία δίνεται από τον παράγοντα της αγοράς και του χρόνου.

Με την αγορά να επιταχύνεται όλο και περισσότερο και να είναι πάντα γεμάτη ειδήσεις, απαιτούνται ταχύτερες και πιο αποτελεσματικές απαντήσεις από τις εταιρείες, με στρατηγικές με καλύτερο σχεδιασμό και εκτέλεση.

Εάν η εταιρεία δεν δώσει αυτές τις απαντήσεις στην αγορά, καταλήγει να χανθεί και να ξεχαστεί, γεγονός που φέρνει χρεοκοπία σε πολλές επιχειρήσεις.

Σε αυτήν την προσπάθεια να είστε πάντα πρώτοι και να έχετε τις καλύτερες ιδέες, η πληροφόρηση είναι δύναμη και η καλή διαχείριση αυτών των πληροφοριών είναι η διαφορά.

Δεν ωφελεί να έχετε τις πληροφορίες εάν δεν ξέρετε πώς να τις χρησιμοποιήσετε ή πώς να τις εφαρμόσετε καλύτερα στη δική σας επιχείρηση.

Ένα άλλο σημείο που καθιστά σημαντική τη χρήση της οργάνωσης δεδομένων είναι η καλύτερη σχέση μεταξύ εταιρείας και πελάτη.

Με μια οργανωμένη και βελτιστοποιημένη βάση δεδομένων, η απάντηση του πελάτη γίνεται πολύ πιο ξεκάθαρη και η ανατροφοδότηση χρησιμοποιείται με τον καλύτερο δυνατό τρόπο.

Όταν η σχέση πελάτη-εταιρείας βελτιώνεται, ο πελάτης αισθάνεται ότι υποστηρίζεται από την εταιρεία, καθώς αισθάνεται ότι οι ανάγκες του ικανοποιούνται σωστά.

Και ένα άλλο πολύ σημαντικό σημείο που φέρνει η οργάνωση δεδομένων είναι η βελτίωση του εσωτερικού ελέγχου της εταιρείας και του εξωτερικού ελέγχου.

Πράγμα που θα ήταν, αντίστοιχα, όταν η εταιρεία γνωρίζει τι πρέπει να βελτιωθεί εντός της εταιρείας και τι πρέπει να βελτιωθεί εκτός αυτής για να προσελκύσει περισσότερους πελάτες και να εδραιωθεί στην αγορά.



ΚΑΘΕ ΤΥΠΟΣ ΕΠΙΧΕΙΡΗΣΗΣ ΧΡΕΙΑΖΕΤΑΙ ΝΑ ΕΧΕΙ ΟΡΓΑΝΩΣΗ ΔΕΔΟΜΕΝΩΝ.

Κάθε τύπος και μέγεθος εταιρείας πρέπει να έχει καλή και αποτελεσματική οργάνωση δεδομένων.

Όπως αναφέραμε σε προηγούμενα θέματα, όταν η εταιρεία έχει μια καλά οργανωμένη βάση δεδομένων και ξέρει πώς να τη χρησιμοποιεί, το αποτέλεσμα που φέρνει στην εταιρεία είναι η καινοτομία σε σχέση με άλλες εταιρείες.

Όλοι οι τύποι επιχειρήσεων χρησιμοποιούν κάποια βάση για την οργάνωση δεδομένων, είτε είναι πιο ρουστίκ είτε πιο ανεπτυγμένη.

Συνήθως μικρές ή πρόσφατες εταιρείες επιλέγουν απλοποιημένο και αυτοματοποιημένο λογισμικό, το οποίο είναι κατασκευασμένο για διάφορες εταιρείες και όχι ειδικά για τη δική σας.

Κάτι που δεν είναι λάθος, φυσικά. Είναι όμως μια αξιόλογη επένδυση που μπορεί να συμβάλει πολύ στην ανάπτυξη της εταιρείας.

Ωστόσο, για μια εταιρεία που μόλις ξεκινά και δεν έχει τόσα πολλά κεφάλαια για να επενδύσει στη δική της βάση δεδομένων, μια εξαιρετική διέξοδος είναι αυτά τα «προκατασκευασμένα».

Άλλες εταιρείες επιλέγουν να επενδύσουν στο δικό τους λογισμικό, το οποίο θα καλύψει ειδικά τις ανάγκες της εταιρείας τους, για το είδος του κοινού και της ζήτησης τους.

Γενικά, αυτές οι εταιρείες με το δικό τους λογισμικό επιτυγχάνουν καλύτερες επιδόσεις και ανάπτυξη από άλλες.

Αυτή η εξέλιξη είναι ξεκάθαρη με τις στρατηγικές της για εξατομικευμένη εξυπηρέτηση πελατών ή όταν ο ίδιος ο πελάτης αποκαλύπτει την εταιρεία που του πρόσφερε μια υπηρεσία.

Αυτό που πραγματικά έχει σημασία είναι ότι η εταιρεία διαθέτει μια βάση οργάνωσης δεδομένων που λειτουργεί και καλύπτει τουλάχιστον τις βασικές της ανάγκες.

Και το συντομότερο δυνατό, επενδύστε στο δικό τους λογισμικό για να βελτιώσετε την απόδοσή τους στην ανάπτυξη των πληροφοριών τους.

ΠΟΙΟΙ ΕΙΝΑΙ ΟΙ ΤΡΟΠΟΙ ΟΡΓΑΝΩΣΗΣ ΤΩΝ ΔΕΔΟΜΕΝΩΝ

Όταν αναφερόμαστε σε μια βάση δεδομένων, μιλάμε για το πώς είναι οργανωμένα τα δεδομένα, πώς αλληλοεπιδρούν οι πληροφορίες και τι είδους μοντέλο δεδομένων χρησιμοποιείται.

Υπάρχουν πολλά μοντέλα και μορφές βάσεων δεδομένων που χρησιμοποιούνται σε εταιρείες, για παράδειγμα: αντικειμενοστραφές, αντικειμενοσχεσιακές, επίπεδες, ιεραρχικές και σχεσιακές.

Το σχεσιακό είναι το πιο χρησιμοποιούμενο μεταξύ αυτών των μοντέλων επειδή είναι πιο πρακτικό και πιο ευέλικτο.

Η κατανόηση της δομής της βάσης δεδομένων είναι πολύ σημαντική, ώστε να μπορείτε να ορίσετε ποια δομή και γλώσσα ταιριάζει καλύτερα στο προφίλ της εταιρείας σας.

Το σχεσιακό μοντέλο, όπως ήδη αναφέρθηκε, είναι το πιο χρησιμοποιούμενο για την κατασκευή αρκετών μοντέλων λογισμικού οργάνωσης δεδομένων.

Ωστόσο, ακόμα κι έτσι, είναι εξαιρετικά σημαντικό να γνωρίζουμε τουλάχιστον τα κύρια Συστήματα Διαχείρισης Βάσεων Δεδομένων (DBMS) που κυκλοφορούν στην αγορά, τις εφαρμογές τους και σε ποιους τύπους εταιρειών θα χρησιμοποιηθούν περισσότερο.

Ας δούμε τώρα μερικά παραδείγματα των τριών κύριων μοντέλων DBMS και ποια θα ήταν τα καταλληλότερα για τον τύπο και το μέγεθος της εταιρείας σας:

ΠΑΡΑΔΕΙΓΜΑΤΑ ΜΟΝΤΕΛΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ:

Τα πιο δημοφιλή μοντέλα DBMS στην αγορά είναι:

Βάση δεδομένων Oracle

Είναι ίσως το πιο διάσημο και αναγνωρισμένο μοντέλο DBMS στην αγορά.

Το κόστος της άδειας χρήσης του είναι ένα από τα υψηλότερα, αλλά αντισταθμίζεται από την ασφάλειά του και τον όγκο των πληροφοριών που αποθηκεύει.

Ταιριάζει καλύτερα σε μεγάλες εταιρείες.

SQL Server

Είναι προϊόν της Microsoft και χρησιμοποιείται επίσης ευρέως από την αγορά.

Δυστυχώς, το πρόβλημά σας είναι ότι είναι συμβατό μόνο με Windows. Ωστόσο, είναι ένα πολύ αξιόπιστο μοντέλο, το οποίο είχε τεράστιες επενδύσεις από την ίδια τη Microsoft.

Ένα άλλο πλεονέκτημα είναι ότι το κόστος του είναι χαμηλότερο από το Oracle Database. Και είναι πολύ κατάλληλο για μικρές και μεσαίες επιχειρήσεις.

MySQL

Είναι ένα προϊόν της Oracle και ένα από τα πιο χρησιμοποιούμενα από τους νέους προγραμματιστές.

Η διαφορά του είναι ότι έχει ανοιχτό κωδικό άδειας και ότι το σύστημά του είναι περισσότερο εστιασμένο στις διαδικτυακές λειτουργίες, καθιστώντας τη διαδικασία πιο δυναμική.

Είναι ιδανικό για εταιρείες που δραστηριοποιούνται περισσότερο στο διαδίκτυο παρά για εταιρείες που εστιάζουν εκτός σύνδεσης.

ΠΟΙΟ ΛΟΓΙΣΜΙΚΟ ΝΑ ΧΡΗΣΙΜΟΠΟΙΗΣΩ ΓΙΑ ΤΗΝ ΟΡΓΑΝΩΣΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΤΗΣ ΕΠΙΧΕΙΡΗΣΗΣ ΜΟΥ;

Είναι πολύ εύκολο να χαθείς ανάμεσα στα διάφορα μοντέλα DBMS της αγοράς.

Πριν αγοράσετε λοιπόν μια άδεια προγράμματος, βάλτε τις προτεραιότητές σας και ψάξτε αυτές που ταιριάζουν καλύτερα στο προφίλ σας.

Επιλέξτε ένα κατάλληλο DBMS

Αυτή η επιλογή θα πρέπει να βασίζεται στην τιμή του λογισμικού και στο πόσα μπορείτε να επενδύσετε σε αυτό.

Είναι επίσης σημαντικό να ελέγξετε τον τύπο της άδειας χρήσης που προσφέρει το πρόγραμμα και ποιες δυνατότητες επιτρέπει την ανάπτυξη.

Αποκτήστε έναν επαγγελματία Διαχειριστή βάσης δεδομένων

Είναι αυτός που θα διασφαλίσει τη σωστή λειτουργία και χρήση του λογισμικού σας.

ασφάλεια δεδομένων

Τα δεδομένα και οι πληροφορίες της εταιρείας σας πρέπει να διατηρούνται ασφαλή, ώστε καμία άλλη εταιρεία να μην μπορεί να τα χρησιμοποιήσει χωρίς την εξουσιοδότησή σας.

Είναι σημαντικό να φροντίζετε για τη σωστή λειτουργία του υλικού που φιλοξενεί τα προγράμματα, όπως είναι σημαντικό να φροντίζετε το ίδιο το λογισμικό.

Επενδύστε σε προγράμματα προστασίας από ιούς.

ποιότητα πληροφοριών

Πρέπει να γνωρίζετε ακριβώς τι είδους πληροφορίες χρειάζονται και τι όχι. Μην αποθηκεύετε κανενός είδους πληροφορίες.

Συσσωρεύοντας πολλά δεδομένα, αναμειγνύοντας τα σχετικά και τα μη, δημιουργείτε ένα κουβάρι πληροφοριών που είναι πιο δύσκολο να φιλτράρετε και να χρησιμοποιήσετε.

Η ΣΥΝΟΠΤΙΚΗ ΟΡΓΑΝΩΣΗ ΔΕΔΟΜΕΝΩΝ ΕΙΝΑΙ:

Συμπερασματικά, η οργάνωση δεδομένων είναι ένα σημαντικό εργαλείο για να διατηρείται η εταιρεία οργανωμένη και να ρέει με σαφή και αποτελεσματικό τρόπο για όλους.

Μέσω ενός προκαθορισμένου τυποποιημένου συστήματος, θα αποθηκεύονται οι απαραίτητες πληροφορίες για την εκτέλεση των δραστηριοτήτων της εταιρείας, καθώς και για την ανάλυση και τη διεξαγωγή των εργασιών.

Για να μπορούν όλοι οι εργαζόμενοι να έχουν πρόσβαση σε αυτά και να τα χρησιμοποιούν στις δραστηριότητές τους εντός της εταιρείας εύκολα και αποτελεσματικά, συμβάλλοντας στην ανάπτυξή της.

Με τη χρήση βάσεων δεδομένων θα βελτιστοποιήσετε τη χρήση των πληροφοριών σας, εφαρμόζοντάς τις στα σωστά σημεία, με τον σωστό τρόπο και τη σωστή στιγμή.

Εκτός από τη βελτίωση των σχέσεων με τους πελάτες, την αύξηση της προβολής τους μεταξύ τους και τη βελτίωση της θέσης τους στην αγορά.

Κάνοντας τη χρήση των πληροφοριών σημαντικό σύμμαχο για την επιτυχία και την ανάπτυξη στον τομέα σας.

Ελπίζουμε ότι σας βοηθήσαμε να κατανοήσετε καλύτερα την οργάνωση δεδομένων, τη σημασία της και τον καλύτερο τρόπο για την επιχείρησή σας.

Μέσω αυτών των συμβουλών, ελπίζουμε ότι μπορείτε να βελτιώσετε την επιχείρησή σας και να την κάνετε ακόμα πιο αποτελεσματική. Οποιοσδήποτε ερωτήσεις προκύψουν, μη διστάσετε να επικοινωνήσετε μαζί μας και θα απαντήσουμε.

Ποιοι είναι οι τρεις τρόποι οργάνωσης δεδομένων;

Ακολουθούν τρεις αποτελεσματικοί τρόποι για να οργανώσετε τις ψηφιακές σας πληροφορίες για παρουσίαση και αποθήκευση.

Εντοπισμός σφαλμάτων δεδομένων. Το σκούπισμα δεδομένων, το σκούπισμα δεδομένων ή το σκούπισμα δεδομένων είναι ακριβώς αυτό που ακούγεται. ...

Πίνακες και Γραφήματα. ...

Οργάνωση ανά κατηγορία και χαρακτηριστικά.

Ποιοι είναι οι τρόποι οργάνωσης των δεδομένων;

Οργάνωση και Παρουσίαση Δεδομένων

Κωδικοποίηση δεδομένων.

Διερευνητική ανάλυση δεδομένων.

Διάγραμμα στελέχους και φύλλου.

Διαγράμματα μετρητών.

Πίνακες συχνότητας: Τυπική, Σχετική και Αθροιστική.

γραφικά δεδομένων,

Ιστογράμματα/Γραφήματα ράβδων, Διαγράμματα

Διαγράμματα πίτας.

Ποιοι είναι οι 3 τρόποι οργάνωσης και ανάλυσης δεδομένων;

Υπάρχουν τρία βασικά βήματα στην ανάλυση δεδομένων:

Βήμα 1 – Οργάνωση και προετοιμασία των δεδομένων για ανάλυση.

Βήμα 2 – Ανάλυση των δεδομένων.

Βήμα 3 – Ερμηνεία των αποτελεσμάτων.

Τι είναι η οργάνωση δεδομένων;

Η οργάνωση δεδομένων είναι η πρακτική της κατηγοριοποίησης και ταξινόμησης δεδομένων για να γίνουν πιο χρήσιμα . Παρόμοια με έναν φάκελο αρχείων όπου διατηρούμε σημαντικά έγγραφα, θα χρειαστεί να οργανώσετε τα δεδομένα σας με τον πιο λογικό και τακτοποιημένο τρόπο, ώστε εσείς –και οποιοσδήποτε άλλος έχει πρόσβαση σε αυτά– να μπορείτε να βρείτε εύκολα αυτό που ψάχνετε.

Διαχείριση Βάσεων Δεδομένων: Συγκριτική Ανάλυση DBMS'S

DBMS: Διαχείριση βάσεων δεδομένων

Οι βάσεις δεδομένων ή οι βάσεις δεδομένων είναι οργανωμένες συλλογές δεδομένων που σχετίζονται με σκοπό να δημιουργήσουν κάποιο νόημα (Information) και να δώσουν μεγαλύτερη αποτελεσματικότητα κατά τη διάρκεια μιας έρευνας ή μελέτης. Είναι ζωτικής σημασίας για τις εταιρείες και εδώ και δύο δεκαετίες έχουν γίνει το κύριο μέρος των πληροφοριακών συστημάτων. Συνήθως υπάρχουν για αρκετά χρόνια χωρίς αλλαγές στη δομή τους.

Αυτή η εργασία στοχεύει να δείξει τη σχέση και τις χρήσεις των συστημάτων διαχείρισης βάσεων δεδομένων (DBMS) .

Τι είναι το SGDB; Πως δουλεύει?

Μια βάση δεδομένων είναι ένα οργανωμένο σύνολο δεδομένων (σύμφωνα με ένα μοντέλο δεδομένων) που υπάρχει σε ένα σύστημα υπολογιστή, διαθέσιμο σε όλους τους χρήστες ή τις διαδικασίες του οργανισμού στο οποίο η πρόσβαση και η ενημέρωση πραγματοποιούνται μέσω συγκεκριμένου λογισμικού. Το Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS) ή Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS) και το λογισμικό που χρησιμοποιείται για τη διαχείριση βάσεων δεδομένων, που σας επιτρέπει να δημιουργείτε βάσεις δεδομένων, να τροποποιείτε βάσεις δεδομένων, να διαγράφετε βάσεις δεδομένων, να εισάγετε δεδομένα στη βάση δεδομένων και να διαγράφετε δεδομένα από τη βάση δεδομένων.

Τα DBMS (Συστήματα Διαχείρισης Βάσεων Δεδομένων) έχουν ένα σύνολο λειτουργικών απαιτήσεων : Ασφάλεια, Ακεραιότητα (περιλαμβάνει μόνο έγκυρα δεδομένα σε σχέση με την πραγματικότητα), Έλεγχο Ανταγωνισμού (Κλείδωμα, Ετικέτα ή Αισιόδοξο) και ανάκτηση και ανοχή σε αστοχίες (Δημιουργία αντιγράφων ασφαλείας και καταγραφή συναλλαγών).

Ένα DBMS (Συστήματα Διαχείρισης Βάσεων Δεδομένων) έχει συνήθως ως στοιχεία, τη μηχανή βάσης δεδομένων, το υποσύστημα ορισμού δεδομένων, το υποσύστημα χειρισμού δεδομένων, το υποσύστημα διαχείρισης δεδομένων και το υποσύστημα δημιουργίας εφαρμογών.

Τα DBMS (Συστήματα Διαχείρισης Βάσεων Δεδομένων) διαθέτουν τα συστήματα διαχείρισης που χρησιμοποιούνται και τη γεωγραφική κατανομή των δεδομένων .

Στο εννοιολογικό επίπεδο της ανάλυσης, το βασικό μέλημα έγκειται στη σημασιολογία των δεδομένων που αντιπροσωπεύει το υψηλότερο επίπεδο αμετάβλητου.

Σε αυτό το πλαίσιο, μπορούν να χρησιμοποιηθούν πολλά εννοιολογικά εργαλεία, συγκεκριμένα η Ένωση οντοτήτων (ονομάζεται επίσης Σχέση οντοτήτων) ή το διάγραμμα κλάσης UML .

Ένα λογικό σχήμα είναι ένα μοντέλο δεδομένων συγκεκριμένου τομέα προβλήματος που εκφράζεται με όρους συγκεκριμένης τεχνολογίας βάσης δεδομένων. Το λογικό σχήμα δεν πρέπει να είναι συγκεκριμένο για ένα συγκεκριμένο προϊόν (ή συγκεκριμένο DBMS). Για παράδειγμα, θα πρέπει να υποδεικνύει ποιους πίνακες, στήλες και σχέσεις (αν το σχήμα ακολουθεί το σχεσιακό μοντέλο), κλάσεις (αν το μοντέλο αντικειμένου) ή XML.

Στη σύλληψη του λογικού σχήματος, το σχεσιακό μοντέλο είναι ένα από τα πιο χρησιμοποιούμενα.

Το φυσικό σχήμα περιγράφει τα φυσικά μέσα μέσω των οποίων αποθηκεύονται τα δεδομένα.

Ένα μοντέλο βάσης δεδομένων είναι μια θεωρία ή μια προδιαγραφή που περιγράφει τον τρόπο δομής και χρήσης μιας βάσης δεδομένων.

Λειτουργία ενός DBMS

Ασφάλεια: Το DBMS δημιουργεί ένα σύστημα ασφαλείας που προστατεύει τη βάση δεδομένων από μη εξουσιοδοτημένη πρόσβαση. Επιβάλλονται κανόνες που καθορίζουν ποιοι χρήστες μπορούν να έχουν πρόσβαση στη βάση δεδομένων και εντός των εξουσιοδοτημένων χρηστών σε ποια αρχεία μπορούν να έχουν πρόσβαση και ποιους τύπους λειτουργιών μπορούν να εκτελέσουν (ανάγνωση, προσθήκη, ενημέρωση, διαγραφή κ.λπ.). Υπάρχουν επίσης διαδικασίες που καθιστούν δυνατή τη δημιουργία αντιγράφων ασφαλείας και την ανάκτηση δεδομένων σε περίπτωση αστοχίας, προκειμένου να διασφαλιστεί η ασφάλεια και η ακεραιότητα των δεδομένων.

Φυσική ασφάλεια: Οι πληροφορίες δεν μπορούν να είναι προσβάσιμες σε μη εξουσιοδοτημένα άτομα, δηλαδή πρέπει να φυλάσσονται καλά.

Λογική ασφάλεια: Οι πληροφορίες μπορούν να προστατευθούν χρησιμοποιώντας λογικές μεθόδους ασφαλείας, π.χ. κωδικούς πρόσβασης.

Ακεραιότητα: Το DBMS διασφαλίζει την επαλήθευση των περιορισμών ακεραιότητας, προκειμένου τα δεδομένα να διατηρούνται πάντα έγκυρα, μειώνοντας τον πλεονασμό και μεγιστοποιώντας τη συνέπεια των δεδομένων. Η διαχείριση συναλλαγών είναι μια πολύ σημαντική πτυχή της διατήρησης της ακεραιότητας των δεδομένων. Μια συναλλαγή αποτελείται από ένα σύνολο ενεργειών που εκτελούνται από έναν χρήστη ή μια εφαρμογή. Ένα παράδειγμα συναλλαγής μπορεί να είναι μια συναλλαγή μεταφοράς χρημάτων μεταξύ δύο λογαριασμών. Εάν η συναλλαγή διακοπεί πριν από το τέλος (διακοπή ρεύματος, προβλήματα δίσκου κ.λπ.) το σύστημα θα πρέπει να αποφύγει μια κατάσταση ασυνέπειας, ενεργοποιώντας την επαναφορά, που είναι ένας μηχανισμός που αναιρεί ό,τι είχε γίνει μέχρι τη στιγμή του προβλήματος και επιστρέφει τη βάση δεδομένων στην κατάσταση συνέπειας.

Έλεγχος Ανταγωνισμού

Σε βάσεις δεδομένων πολλών χρηστών (οι οποίες θα είναι η πλειοψηφία), μπορεί να γίνει προσπάθεια πρόσβασης στα ίδια δεδομένα από περισσότερους από έναν χρήστες ταυτόχρονα. Το DBMS παρέχει μηχανισμούς που διασφαλίζουν ότι η βάση δεδομένων ενημερώνεται σωστά, αυτός ο μηχανισμός ονομάζεται μηχανισμός ελέγχου ταυτόχρονης χρήσης.

•**Αντίγραφο ασφαλείας:** Αντίγραφο ασφαλείας, που συνήθως φυλάσσεται σε δισκέτες, μαγνητικές ταινίες ή CD-R, το οποίο επιτρέπει την ανάκτηση σημαντικών πληροφοριών ή προγραμμάτων σε περίπτωση βλάβης του σκληρού δίσκου.

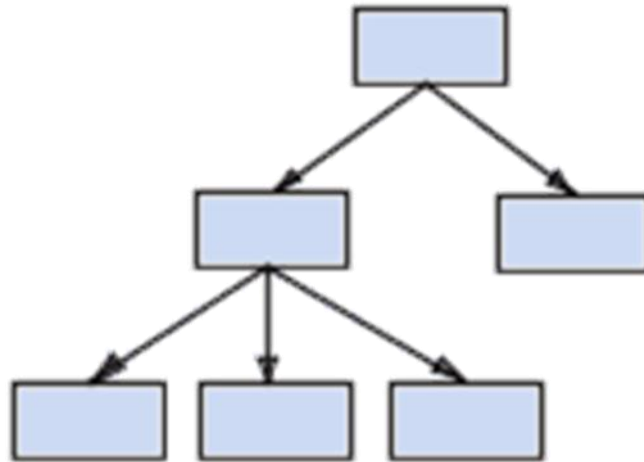
•**Οι πληροφορίες Inside Backup** αποθηκεύονται στην εταιρεία.

•**Outside Backup** οι πληροφορίες αποθηκεύονται εκτός της εταιρείας

Μοντέλα Βάσεων Δεδομένων

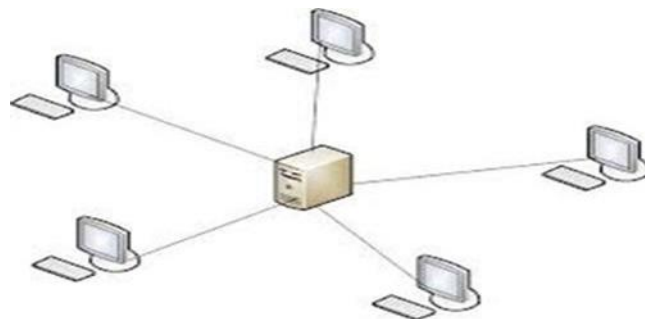
Έχουν προταθεί αρκετά μοντέλα βάσεων δεδομένων, και συγκεκριμένα:

Ιεραρχικό μοντέλο: τα δεδομένα ταξινομούνται ιεραρχικά, σύμφωνα με μια φθίνουσα δενδρόμηση. Αυτό το μοντέλο χρησιμοποιεί δείκτες μεταξύ των διαφορετικών εγγραφών. Αυτό είναι το πρώτο μοντέλο DBMS , όπως μπορούμε να δούμε στο Σχήμα.



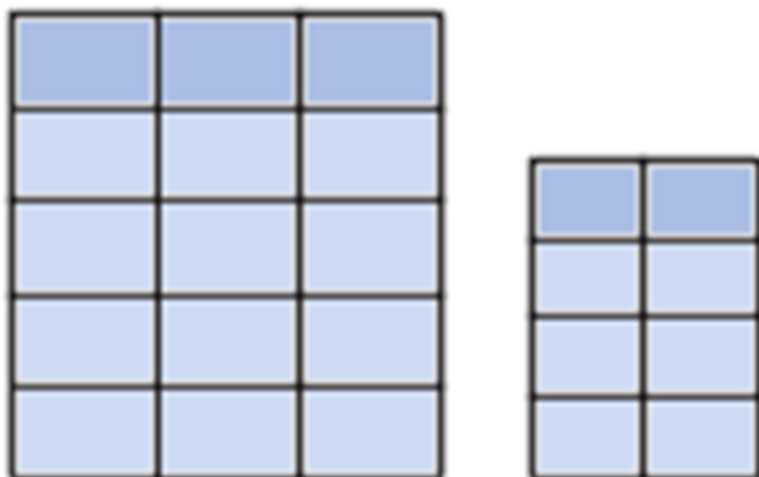
Αναπαράσταση ιεραρχικού μοντέλου

- Μοντέλο δικτύου: Η οργάνωσή του είναι παρόμοια με αυτή των ιεραρχικών βάσεων δεδομένων, με τη διαφορά ότι κάθε θυγατρική εγγραφή μπορεί να συνδεθεί με περισσότερες από μία γονικές εγγραφές, δημιουργώντας πολύ περίπλοκες συνδέσεις και χρησιμοποιούνται ευρέως σε συστήματα για μεγάλους υπολογιστές (mainframe) . Δεδομένου ότι αυτό το μοντέλο αποτελείται από μια πιο ολοκληρωμένη δομή, έχει τις βασικές ιδιότητες των εγγραφών, των συνόλων και των εμφανίσεων και χρησιμοποιεί τη γλώσσα ορισμού (DDL) και τη γλώσσα χειρισμού δεδομένων (DML), επιπλέον του ότι επιτρέπει την πιο αποτελεσματική εξέλιξη του μοντέλου . Δείτε ένα παράδειγμα στο Σχήμα.



Αναπαράσταση μοντέλου Δικτύου

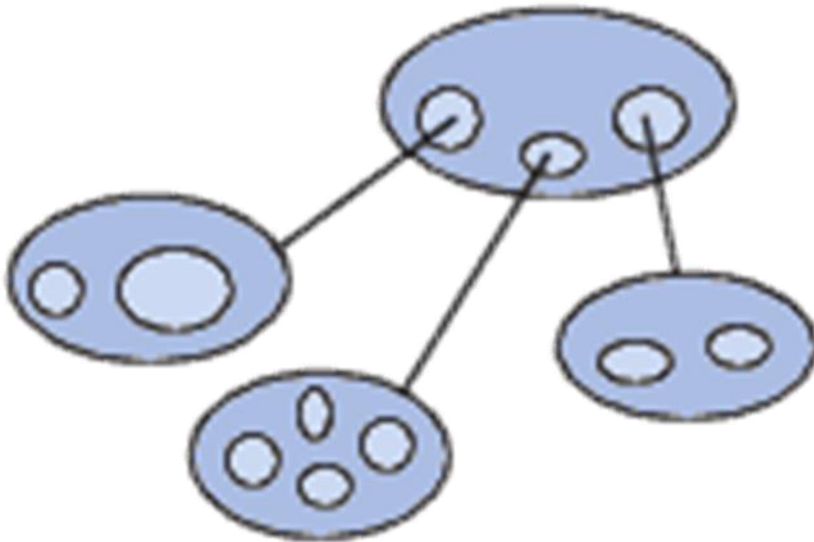
•Σχεσιακό μοντέλο: Το σχεσιακό μοντέλο είναι ένα μοντέλο δεδομένων, κατάλληλο για να είναι το υποκείμενο μοντέλο ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων (DBMS) , το οποίο βασίζεται στην αρχή ότι όλα τα δεδομένα αποθηκεύονται σε πίνακες (ή, μαθηματικά μιλώντας, σχέσεις). Ολόκληρος ο ορισμός του είναι θεωρητικός και βασίζεται στη λογική των κατηγορημάτων και στη θεωρία συνόλων. Δείτε ένα παράδειγμα στο Σχήμα.



Αναπαράσταση Σχεσιακού Μοντέλου

•Η μοντελοποίηση διαστάσεων είναι μια λογική τεχνική σχεδίασης που χρησιμοποιείται συνήθως για αποθήκες δεδομένων που έρχεται σε αντίθεση με τη μοντελοποίηση σχέσεων οντοτήτων. Σύμφωνα με τον καθηγητή Kimball, η μοντελοποίηση διαστάσεων είναι η μόνη βιώσιμη τεχνική για βάσεις δεδομένων που πρέπει να απαντούν σε ερωτήματα σε μια αποθήκη δεδομένων. Επίσης, σύμφωνα με τον ίδιο, η μοντελοποίηση σχέσεων οντοτήτων είναι πολύ χρήσιμη για την καταγραφή συναλλαγών και για τη φάση διαχείρισης της κατασκευής μιας αποθήκης δεδομένων, αλλά θα πρέπει να αποφεύγεται κατά την παράδοση του συστήματος στον τελικό χρήστη.

•Μοντέλο που σχετίζεται με αντικείμενα (DBMS, Σύστημα Διαχείρισης Βάσεων Δεδομένων αντικειμένων): τα δεδομένα αποθηκεύονται με τη μορφή αντικειμένων, δηλαδή δομών που ονομάζονται κλάσεις και παρουσιάζουν δεδομένα μέλους. Τα πεδία είναι περιπτώσεις αυτών των κλάσεων.



Ενότητες στοιχείων ενός ΣΔΒΔ

Οι κύριες λειτουργικές μονάδες που ενσωματώνουν ένα DBMS είναι:

- Προμεταγλωττιστές;
- Μεταγλωττιστές;
- διερμηνείς?
- μηχανισμός διαβούλευσης·
- Διαχειριστής buffer;
- Υπεύθυνος Συναλλαγών;
- Διαχείριση αρχείων.

Παραδείγματα ορισμένων DBMS

- IBM Informix
- PostgreSQL
- Firebird
- HSQLDB
- DB2

- mSQL
- MySQL
- Μαντείο
- SQL-Server
- TinySQL
- ZODB
- ΕΞΟΔΟΣ
- Sybase
- Microsoft Access (Μερικοί το θεωρούν DBMS αλλά είναι DBMS)
- Microsoft Visual Foxpro
- Παράδοξο

Χρήση και σύγκριση μεταξύ DBMS (SQL-SERVER και ORACLE)

SQL Server

Το OMS SQL Server είναι ένα DBMS που αναπτύχθηκε από τη Microsoft.

Δημιουργήθηκε σε συνεργασία με τη Sybase το 1988 αρχικά για την πλατφόρμα OS/2. Αυτή η συνεργασία κράτησε μέχρι το 1994, με την κυκλοφορία της έκδοσης των Windows NT, και έκτοτε η Microsoft διατηρεί το προϊόν. Ως βάση δεδομένων, είναι ένα προϊόν λογισμικού του οποίου η κύρια λειτουργία είναι να αποθηκεύει και να ανακτά δεδομένα που ζητούνται από άλλες εφαρμογές λογισμικού, είτε αυτές που βρίσκονται στον ίδιο υπολογιστή είτε αυτές που εκτελούνται σε άλλον υπολογιστή μέσω δικτύου.

- DBMS που διατίθεται επί του παρόντος από τη Microsoft.
- Κόστος μιας άδειας Enterprise ανά CPU: περίπου. 28.000 \$ ανά CPU.
- Μπορεί να εγκατασταθεί μόνο σε λειτουργικό σύστημα Windows.

- Διαθέτει εργαλεία διαχείρισης με εξαιρετική γραφική διεπαφή, που επιτρέπουν μια ευκολότερη και πιο παραγωγική διαχείριση. Π.χ.: SQL Server Management Studio 2008;
- Όπως κάθε εργαλείο της Microsoft, η βάση δεδομένων του SQL Server είναι γενικά πιο εύκολη στη διαχείριση και στον προγραμματισμό από την Oracle.
- Λογική διαίρεση ενός στιγμιότυπου που περιέχει πολλαπλά DB. Ο SQL Server έχει ένα επιπλέον επίπεδο όπου κάθε DB μπορεί να περιέχει πολλούς χρήστες ή σχήματα. Με αυτόν τον τρόπο, μπορούμε να οργανώσουμε και να διαχειριστούμε καλύτερα τα συσχετισμένα σχήματα, που ανήκουν στην ίδια εφαρμογή, σε ένα DB που απομονώνεται από τα άλλα σχήματα DB.
- Έχει ορισμένα σχήματα συστήματος που ονομάζονται MODEL και χρησιμεύουν ως πρότυπο για σχήματα χρηστών. Εάν, για παράδειγμα, το σχήμα MODEL έχει 2 πίνακες και μια προβολή, κατά τη δημιουργία ενός νέου σχήματος χρήστη, θα δημιουργηθεί με αυτά τα ίδια αντικείμενα.

Το Oracle είναι ένα DBMS που εμφανίστηκε στα τέλη της δεκαετίας του 1970 όταν ο Larry Ellison είδε μια ευκαιρία που είχαν χάσει άλλες εταιρείες, όταν βρήκε μια περιγραφή ενός λειτουργικού πρωτοτύπου μιας σχεσιακής βάσης δεδομένων και διαπίστωσε ότι καμία εταιρεία δεν είχε κάνει καμία προσπάθεια να εμπορευματοποιήσει αυτήν την τεχνολογία.

- Επί του παρόντος διατίθεται στην αγορά από την Oracle, η οποία γεννήθηκε το 1979 και ήταν η πρώτη σχεσιακή DB που διατίθεται στο εμπόριο στον κόσμο.
- Κόστος μιας άδειας Enterprise ανά CPU: 47.500 € (έως 2 πυρήνες).
- Μπορεί να εγκατασταθεί σε πολλαπλές πλατφόρμες από το 1985. Μεταξύ των κυριότερων, μπορούμε να αναφέρουμε: Unix, Linux, HP/UX, IBM AIX, IBM VMS και Windows.
- Διαθέτει εργαλεία διαχείρισης με λιγότερο φιλικό γραφικό περιβάλλον από αυτά του SQL Server. Αυτό αλλάζει και βελτιώνεται με κάθε νέα έκδοση της Oracle. Στο 10G, το Enterprise Manager έχει πολλές δυνατότητες και η γραφική του

διεπαφή έχει εξελιχθεί πολύ, αλλά εξακολουθώ να πιστεύω ότι τα εργαλεία της Microsoft είναι πιο διαισθητικά και πιο παραγωγικά.

- Έχει περισσότερες δυνατότητες ασφάλειας και απόδοσης από τον SQL Server.
- Από προεπιλογή, η Oracle δεν δεσμεύει συναλλαγές. Αυτό σας επιτρέπει να αναιρέσετε τις αλλαγές σε μια δήλωση SQL εάν υποβλήθηκε κατά λάθος.
- Από προεπιλογή, η Oracle επιτρέπει τη συνεπή ανάγνωση δεδομένων. Αυτή η δυνατότητα επιτρέπει στον χρήστη "B" να διαβάζει δεδομένα από μια σειρά ενός πίνακα, την ίδια στιγμή που αλλάζουν από τον χρήστη "A", χωρίς ο χρήστης "B" να βλέπει τα δεδομένα να αλλάζουν από το "THE".
- Πιο ευέλικτη αρχιτεκτονική με περισσότερες δυνατότητες για βελτιστοποίηση απόδοσης. Στην Oracle είναι δυνατή η δημιουργία και η διαχείριση πολλών δομών μνήμης στο DB. Είναι δυνατό, για παράδειγμα, να οριστούν δομές αποθήκευσης με μεγέθη μπλοκ που μπορεί να ποικίλλουν από 2k έως 32k. Στον SQL Server είναι δυνατή μόνο η δημιουργία δομών αποθήκευσης 8k. Στην Oracle, τα συστήματα OLAP και τα ευρετήρια γενικά βελτιστοποιούνται με μεγαλύτερα μεγέθη μπλοκ (32k).
- Διαθέτει πακέτα, τα οποία είναι αντικείμενα που επιτρέπουν (μεταξύ πολλών άλλων πλεονεκτημάτων) να ομαδοποιούν και να ενσωματώνουν αποθηκευμένες διαδικασίες και κώδικα συναρτήσεων.
- Η Oracle έχει περισσότερους τύπους ευρετηρίων από τον SQL Server.

Και τα δύο είναι εξαιρετικά BD και το καθένα έχει τα πλεονεκτήματα και τα μειονεκτήματά του. Ο SQL Server έχει το κύριο πλεονέκτημα ότι έχει χαμηλότερο κόστος, περίπου το ήμισυ της τιμής ενός Oracle. Στο SQL Server DBMS, όλες οι λειτουργίες (π.χ.: πίνακες διαμερισμάτων) περιλαμβάνονται στο αρχικό τέλος άδειας χρήσης, ενώ στο Oracle DBMS είναι απαραίτητο να αποκτήσετε πρόσθετες άδειες χρήσης (που ονομάζονται Options), οι οποίες κάνουν το προϊόν Oracle έως και 10 φορές περισσότερο ακριβό από το προϊόν της Microsoft. Ένα άλλο πλεονέκτημα του SQL Server είναι η ευκολία χρήσης και διαχείρισης. Εγώ, ιδιαίτερα,

όταν άρχισα να δουλεύω με την Oracle, τα βρήκα όλα πολύ δύσκολα. Τα εργαλεία του SQL Server ήταν πιο παραγωγικά και πιο διαισθητικά.

Το Oracle έχει υψηλότερο κόστος από τον SQL Server και είναι πιο δύσκολο στη διαχείριση, αλλά είναι ένα προϊόν με περισσότερες δυνατότητες ασφάλειας και απόδοσης, οι οποίες μπορεί να είναι πολύ σημαντικές και κρίσιμες για εταιρείες που έχουν κρίσιμες εφαρμογές και έχουν πολλά δεδομένα και πολλά ανταγωνιστικούς χρήστες γενικά

Από τεχνική άποψη, τα DBMS μπορεί να διαφέρουν πολύ. Οι όροι σχεσιακός, δικτυακός, επίπεδος και ιεραρχικός αναφέρονται στον τρόπο με τον οποίο ένα ΣΔΒΔ οργανώνει τις πληροφορίες εσωτερικά. Η εσωτερική οργάνωση μπορεί να επηρεάσει το πόσο γρήγορα και ευέλικτα μπορείτε να εξάγετε πληροφορίες.

Εκπαιδευτική υποενότητα : 5.2 Μοντέλα Βάσεων Δεδομένων

Τι είναι η βάση δεδομένων;

Η βάση δεδομένων είναι η οργάνωση και αποθήκευση πληροφοριών σχετικά με έναν συγκεκριμένο τομέα. Με απλούστερο τρόπο, είναι η ομαδοποίηση δεδομένων που ασχολούνται με το ίδιο θέμα και που πρέπει να αποθηκευτούν για ασφάλεια ή μελλοντική διάσκεψη.

Είναι σύνηθες για τις εταιρείες να έχουν πολλές πληροφορίες που πρέπει να οργανωθούν και να διατεθούν εντός της επιχείρησης, έτσι ώστε να μπορεί να συμβουλευτεί αργότερα η ομάδα και η διοίκηση.

Επομένως, είναι ενδιαφέρον να υπάρχει ένα σύστημα διαχείρισης βάσεων δεδομένων, το SGBD, για να μπορεί να χειριστεί τις πληροφορίες και να κάνει τη ρουτίνα της εταιρείας πολύ πιο απλή.

Σήμερα, υπάρχουν διάφοροι τύποι DBMS και ο καθένας είναι κατάλληλος για τις ανάγκες του πελάτη. Τα πιο συνηθισμένα είναι: Oracle, DB2, MySQL, SQL Server, PostgreSQL και άλλα.

Εάν η εταιρεία σας διαθέτει ιστότοπο στο WordPress ή σε κάποια άλλη πλατφόρμα, η βάση δεδομένων είναι απαραίτητη για τη διατήρηση της ομαλής λειτουργίας και της πρακτικότητας των καθημερινών εργασιών.

Πόσο σημαντική είναι μια βάση δεδομένων;

Αν αναρωτιέστε γιατί το WordPress χρειάζεται μια βάση δεδομένων, η απάντηση είναι απλή: να σας επιτρέψει να αποθηκεύσετε όλες τις πληροφορίες που περιέχονται στον ιστότοπο και το ιστολόγιο σας στο WordPress αποτελεσματικά, επιτρέποντας στον χρήστη να έχει την καλύτερη δυνατή εμπειρία και να περιηγηθεί και να βρει όλες τις πληροφορίες που ανάγκη κατά την αναζήτησή σας.

Αυτά τα σωστά οργανωμένα δεδομένα θα βοηθήσουν στη βελτίωση όχι μόνο της εμπειρίας αυτών των επισκεπτών, αλλά και της ρουτίνας της περιοχής πληροφορικής σας. Εξάλλου, οι επαγγελματίες σας θα μπορούν να έχουν πρόσβαση, να αποθηκεύουν και να αποκαθιστούν δεδομένα όποτε είναι απαραίτητο με τη χρήση ενός καλού εργαλείου διαχείρισης βάσεων δεδομένων.

Ποια είναι τα πλεονεκτήματα μιας βάσης δεδομένων;

Η επένδυση σε μια καλή βάση δεδομένων μπορεί να αποφέρει πολλά οφέλη στην εταιρεία, επηρεάζοντας θετικά όχι μόνο την παραγωγικότητα της ομάδας, αλλά και την ευθυγράμμιση μεταξύ των ομάδων και τα αποτελέσματα που επιτυγχάνονται κάθε μήνα.

Βελτιωμένη σχέση και παραγωγικότητα στην εταιρεία

Όταν μιλάμε για βελτίωση των σχέσεων στην εταιρεία, μιλάμε για τον τρόπο με τον οποίο επικοινωνούν οι πιο διαφορετικοί τομείς.

Μια αποτελεσματική βάση δεδομένων θα επιτρέψει σε πολλούς διαφορετικούς τομείς να επικοινωνήσουν και να διατηρήσουν σαφείς τις προσδοκίες, τους στόχους και τους στόχους τους μέσω της διαχείρισης δεδομένων.

Οι τομείς θα μπορούν να λειτουργήσουν, κατανοώντας ακριβώς ποιες είναι οι πραγματικές πληροφορίες για την εταιρεία και τι πρέπει να επιτευχθεί. Αυτή η διαφάνεια μπορεί να είναι ένας τρόπος ευθυγράμμισης των προσδοκιών των διαφορετικών ομάδων, εκτός από τη βελτίωση της παραγωγικότητας αυτών των

ομάδων, την αποφυγή συγκρούσεων και την αύξηση των δυνατοτήτων για καλά αποτελέσματα.

Μείωση λειτουργικών κινδύνων

Με τη διαφάνεια των πληροφοριών, είναι δυνατό να μειωθούν οι λειτουργικοί κίνδυνοι εντός της εταιρείας. Αυτό συμβαίνει επειδή όλες οι ομάδες θα είναι ευθυγραμμισμένες με τις ανάγκες και τις απαιτήσεις της επιχείρησης και θα μπορούν να συντονίζονται και να διοικούν τις ενέργειές τους ως μια ενιαία μεγάλη ομάδα.

Οι τομείς θα έχουν περισσότερη διεκδίκηση στα καθήκοντά τους, κάτι που θα επιτρέψει σε άλλες ομάδες να ξεχωρίζουν και να βελτιώνουν την απόδοσή τους σε καθημερινή βάση.

αυξημένη ασφάλεια

Επιπλέον, έχουμε αυξημένη ασφάλεια εντός της εταιρείας. Με ένα σύστημα διαχείρισης πληροφοριών, θα είναι δυνατό να έχετε περισσότερη ησυχία με την αποθήκευση αυτών των δεδομένων, επιτρέποντας τον περιορισμό της πρόσβασης και μόνο συγκεκριμένα άτομα να μπορούν να προβάλλουν τα έγγραφα.

Βελτιωμένη λήψη αποφάσεων

Μια εταιρεία που διαθέτει καλή βάση δεδομένων έχει επίσης την ηρεμία και τη διαφάνεια που απαιτούνται για να λαμβάνει αποφάσεις πολύ πιο αποτελεσματικά.

Πολλές εταιρείες εξακολουθούν να αποφασίζουν σημαντικά ζητήματα μέσω εικασιών. Με τη βάση δεδομένων, θα έχετε πολύτιμες πληροφορίες στις οποίες θα βασιστείτε και θα αξιολογήσετε το σενάριο ως σύνολο, με σαφή και διαφανή τρόπο.

Σύντομα θα μπορείτε επίσης να εντοπίσετε τα κύρια σημεία συμφόρησης και βελτιώσεις που πρέπει να γίνουν στην εταιρεία σας.

Ποιοι είναι οι κύριοι τύποι βάσεων δεδομένων;

Υπάρχουν πολλές επιλογές βάσης δεδομένων διαθέσιμες στην αγορά. Αλλά πριν δουλέψουμε σε καθένα από αυτά, πρέπει να κατανοήσουμε τη διαφορά μεταξύ σχεσιακών και μη σχεσιακών βάσεων δεδομένων.

Οι σχεσιακές βάσεις δεδομένων είναι χτισμένες στο πρότυπο που προσανατολίζεται στο σύνολο. Με αυτόν τον τρόπο, τα δεδομένα που είναι διαθέσιμα εκεί θα αποθηκεύονται σε πίνακες. Κάθε πίνακας θα έχει χαρακτηριστικά και σειρές ή εγγραφές υπεύθυνες για την οργάνωση αυτών των πληροφοριών.

Χρησιμοποιούνται συνήθως για δεδομένα σε πίνακα, τα οποία είναι πολύ πιο απλά στην εισαγωγή και επιτρέπουν επίσης πιο πρακτική καθημερινή ανάκτηση. Η γλώσσα που χρησιμοποιείται σε αυτήν τη μορφή είναι η SQL, Structured Query Language. Επομένως, εάν θέλετε να χρησιμοποιήσετε μια σχεσιακή βάση δεδομένων, πρέπει να δώσετε προσοχή σε αυτή τη λεπτομέρεια.

Είναι ιδανικά για CRM, ERP ή ακόμα και οικονομική διαχείριση εταιρειών.

Από την άλλη πλευρά, οι μη σχεσιακές βάσεις δεδομένων είναι υπεύθυνες για την ικανοποίηση απαιτήσεων που δεν μπορούν να ικανοποιήσουν οι σχεσιακές βάσεις δεδομένων. Ένα παράδειγμα απαιτήσεων είναι αυτά τα μικτά δεδομένα, όπου πίνακες, εικόνες και χάρτες αναμειγνύονται, για παράδειγμα, τα οποία δεν μπορούν να ταξινομηθούν σε στήλες και σειρές πίνακα.

Οι λύσεις τους βασίζονται στην αποθήκευση cloud. Η γλώσσα που χρησιμοποιείται σε αυτή τη μορφή είναι NoSQL, όχι μόνο SQL.

Εάν σκέφτεστε να προγραμματίσετε ένα blog στο WordPress ή να δημιουργήσετε έναν ισότοπο για την εταιρεία σας, πρέπει να γνωρίζετε τους κύριους τύπους βάσεων δεδομένων.

Το Oracle Database είναι το πιο χρησιμοποιούμενο σύστημα διαχείρισης βάσεων δεδομένων στον κόσμο. Λειτουργεί με τη γλώσσα SQL και εγγυάται ασφάλεια και αρκετούς πόρους για τους πελάτες και τους χρήστες του.

Ένα από τα πλεονεκτήματα αυτού του μοντέλου είναι η ευκολία εγκατάστασης στις πιο διαφορετικές πλατφόρμες, καθώς είναι συμβατό με IBM AIX, IBM VMS, Windows, Linux, Unix και HP/UX.

Ωστόσο, είναι ενδιαφέρον να επενδύσουμε σε καλό υλικό για να μην υποφέρουν οι επιδόσεις.

Ένα άλλο πλεονέκτημα της Oracle είναι η τεκμηρίωσή της.

Είναι εξαιρετικά λεπτομερές, επομένως οι προγραμματιστές θα έχουν πολύ περισσότερες γνώσεις για τις δυνατότητες που είναι διαθέσιμες στην πλατφόρμα.

Επιπλέον, η Oracle προσφέρει χαρακτηριστικά ασφάλειας και απόδοσης που εγγυώνται την ποιότητα της εργασίας και την ηρεμία των χρηστών, καθιστώντας την την καλύτερη εναλλακτική λύση για μεγάλες εταιρείες ή επιχειρήσεις που έχουν πιο σύνθετες απαιτήσεις.

SQL Server

Ο SQL Server , που δημιουργήθηκε από τη Microsoft, είναι πολύ γνωστός και χρησιμοποιείται στην αγορά.

Η γλώσσα που χρησιμοποιείται σε αυτό το εργαλείο είναι η T-SQL και προσφέρει προηγμένες και διαφοροποιημένες λειτουργίες για τη διευκόλυνση της ενημέρωσης δεδομένων και της αποθήκευσης πληροφοριών με ασφαλή και αξιόπιστο τρόπο.

Ο SQL Server λειτουργεί με ενσωματωμένα συστήματα κρυπτογράφησης, επιτρέποντας την προβολή ή την αλλαγή πληροφοριών να γίνεται μόνο από τους υπεύθυνους, γεγονός που εγγυάται ακόμη περισσότερη ασφάλεια και ηρεμία για χρήστες και επιχειρηματίες.

Είναι μια εναλλακτική που χρησιμοποιείται συνήθως σε ηλεκτρονικά καταστήματα, κυβερνητικά ιδρύματα, τράπεζες και βιομηχανίες όλων των μεγεθών.

MySQL

Η MySQL είναι μια σχεσιακή βάση δεδομένων που ανήκει στην Oracle . Ένα από τα πιο εντυπωσιακά χαρακτηριστικά αυτού του μοντέλου είναι το γεγονός ότι είναι ανοιχτού κώδικα. Χρησιμοποιεί τη γλώσσα SQL και λειτουργεί με εμπορικές άδειες χρήσης και άδειες ελεύθερου λογισμικού.

Η MySQL ξεχωρίζει για την ευκολία χρήσης και το πλαίσιο ασφάλειας και αξιοπιστίας που επέτρεπε σε εταιρείες και εφαρμογές που βασίζονται στο διαδίκτυο να χρησιμοποιούν τους πόρους της. Μεταξύ των κύριων χρηστών είναι η Google, το Facebook , το Youtube , το Twitter και η NASA.

Ένα άλλο πλεονέκτημα είναι η εύκολη ενσωμάτωση PHP των πακέτων φιλοξενίας που προσφέρονται συνήθως.

PostgreSQL

Η PostgreSQL είναι επίσης ένας διαχειριστής σχεσιακών βάσεων δεδομένων ανοιχτού κώδικα, που χρησιμοποιείται συνήθως για ηλεκτρονικά συστήματα όπως το Skype, η Apple και το μετρό του Σάο Πάολο.

Θεωρείται μια από τις πιο προηγμένες εναλλακτικές λύσεις της αγοράς, με διαφοροποιημένους και πολύπλοκους πόρους, που επιτρέπει στους χρήστες να έχουν μεγαλύτερη ευκολία πρόσβασης και ακεραιότητα συναλλαγών. Αυτή η εναλλακτική απαιτεί μια ισχυρή λύση υλικού, ώστε να μην παρεμποδίζεται η ανάπτυξη.

Η PostgreSQL έχει τη δυνατότητα να υποστηρίζει μεγάλη ροή δεδομένων, διασφαλίζοντας ασφάλεια και σταθερότητα, καθώς και υψηλή απόδοση σε ακόμη πιο προσιτή τιμή.

NoSQL

Το NoSQL είναι ένα μη σχεσιακό σύστημα βάσης δεδομένων , όπως εξηγήθηκε προηγουμένως. Σήμερα, αυτός ο όρος χρησιμοποιείται συνήθως από άτομα που παράγουν περιεχόμενο μέσω συσκευών, κοινωνικών δικτύων και άλλων τύπων λειτουργιών ιστού, που απαιτούν τη διαχείριση δεδομένων σε διαφορετικές μορφές.

Η χρήση βάσεων δεδομένων NoSQL θα σας επιτρέψει να έχετε μεγαλύτερη επεκτασιμότητα και μεγαλύτερη εξοικονόμηση σε καθημερινή βάση, επειδή, σε αντίθεση με κάποιες που έχουμε ήδη παρουσιάσει σε αυτό το άρθρο, δεν απαιτεί πολύ ισχυρό υλικό.

Το NoSQL έχει επίσης τη συντήρησή του πολύ πιο απλή και πρακτική, γεγονός που μπορεί να επιτρέψει τη μείωση της ομάδας και τον έλεγχο των δαπανών, καθιστώντας μια πολύ ενδιαφέρουσα εναλλακτική για μεγάλες εταιρείες.

MongoDB

Το MongoDB είναι ένα από τα μεγαλύτερα highlights της αγοράς. Αυτή η βάση δεδομένων είναι ανοιχτού κώδικα και είναι μια από τις πιο χρησιμοποιούμενες από πολλές εταιρείες. Το σύστημά του τρέχει σε Windows, Linux και OSX, με γλώσσα προγραμματισμού C++.

Η χρήση του προσανατολίζεται σε έγγραφα JSON, επιτρέποντας τη δημιουργία πινάκων και στηλών εκ των προτέρων. Η απόδοση του συστήματος είναι εξαιρετική, λόγω της γλώσσας προγραμματισμού του, που υπόσχεται καλύτερη απόδοση και ηρεμία σε καθημερινή βάση.

Το MongoDB δημιουργήθηκε για να εξασφαλίσει μια ορισμένη δυναμική. Ως εκ τούτου, θεωρήθηκε ότι τα Big Data ήταν σε θέση να υποστηρίξουν την κλιμάκωσή τους τόσο κάθετα όσο και οριζόντια. Επιπλέον, χρησιμοποιεί σερβιεράκια, που του επιτρέπουν να μπορεί να χειρίζεται μεγάλο όγκο πληροφοριών.

Redis

Το Redis έχει γίνει μια δημοφιλής βάση δεδομένων στην αγορά και λειτουργεί επίσης ως ανοιχτού κώδικα. Μέσω αυτού του συστήματος, οι πληροφορίες αποθηκεύονται σε μορφή κλειδιού-τιμής.

Είναι μια εναλλακτική λύση που ξεχωρίζει για την ταχύτητα, τη λειτουργικότητα και την ευκολία χρήσης της, καθώς χρησιμοποιείται συνήθως σε διαδικτυακές και κινητές εφαρμογές, όπως η τεχνολογία διαφημίσεων και τα παιχνίδια, για παράδειγμα.

Αυτό το σύστημα έχει ένα πιο ευέλικτο σύνολο δομών, υπεύθυνο για την ευκολία στη δημιουργία εφαρμογών και την ευελιξία για εγγραφή και ανάγνωση δεδομένων.

Αν όλα αυτά φαίνονται πολύ περίπλοκα, μην ανησυχείτε. Υπάρχουν εταιρείες που βοηθούν τους πελάτες να επιλέξουν τη βάση δεδομένων και να δημιουργήσουν τον ιστότοπο και το blog. Ένα από αυτά είναι το Stage , μια υπηρεσία της Rock Content.

Πώς να δημιουργήσετε βάση δεδομένων;

Για να δημιουργήσετε το σύστημα βάσης δεδομένων σας, χρειάζεστε ένα έργο που περιλαμβάνει την οργάνωση των πληροφοριών και αξιολογεί τη χρήση τεχνικών που θα επιτρέψουν την απόδοση του συστήματος.

Επιπλέον, είναι επίσης σημαντικό να ληφθεί υπόψη ένα σύστημα που μπορεί εύκολα να συντηρηθεί όποτε είναι απαραίτητο.

Εν ολίγοις, ένα έργο για το σύστημα βάσης δεδομένων χωρίζεται σε δύο φάσεις, οι οποίες αφορούν ένα σύστημα βάσης δεδομένων που δεν έχει ακόμη υλοποιηθεί:

δημιουργία εννοιολογικής μοντελοποίησης.

λογική δημιουργία σχεδίου.

Σε περίπτωση που πρέπει να βελτιστοποιήσετε και να βελτιώσετε ένα σύστημα βάσης δεδομένων που υπάρχει ήδη, είναι πολύ παλιό, δεν έχει τεκμηρίωση ή χρειάζεται ενημέρωση, θα χρειαστεί να ακολουθήσετε μια τεχνική που ονομάζεται Αντίστροφη Μηχανική.

Πώς να διαχειριστείτε τη βάση δεδομένων;

Για να γίνει μια αποτελεσματική διαχείριση της βάσης δεδομένων, πρέπει να ληφθούν υπόψη μερικά πράγματα. Το πρώτο είναι η αξιολόγηση της υποδομής του συστήματος που χρησιμοποιείται. Βεβαιωθείτε ότι αντιμετωπίζετε όλες τις ανάγκες σας με την τρέχουσα δομή του συστήματος.

Μόλις γίνει αυτό, είναι καιρός να σκεφτούμε τη μεθοδολογία οργάνωσης.

Δημιουργήστε μια μεθοδολογία που ταιριάζει στις απαιτήσεις σας και σας επιτρέπει

να οργανώνετε και να βρίσκετε πληροφορίες σύμφωνα με τις προτιμήσεις της εταιρείας σας.

Ήρθε η ώρα να σκεφτούμε την πρόσβαση στα δεδομένα: πρέπει να είναι εύκολη, απλή και πρακτική, για να κάνει τη ρουτίνα της εταιρείας και την καθημερινότητα των εργαζομένων πολύ πιο αποτελεσματική.

Για να κάνετε αυτή τη ρουτίνα ακόμα πιο απλή, η συμβουλή είναι να αυτοματοποιήσετε τη διαχείριση της βάσης δεδομένων. Τώρα που ορίστηκε η μεθοδολογία οργάνωσης, οραματιστείτε με ποιους τρόπους θα είναι δυνατό να γίνουν αυτές οι εργασίες πολύ πιο αυτοματοποιημένες.

Είναι ενδιαφέρον να διατηρείται η αυτόματη δημιουργία αντιγράφων ασφαλείας και, σε ορισμένες περιπτώσεις, να επιτρέπεται στο επόμενο σύστημα να κάνει μια έξυπνη διασταύρωση των δεδομένων, για να εντοπίζει εύκολα μετρήσεις και πιθανά προβλήματα.

Τελευταίο αλλά εξίσου σημαντικό, είναι σημαντικό να παρακολουθείτε το σύστημα διαχείρισης δεδομένων σας, να προσδιορίζετε όλα όσα συμβαίνουν, εάν απαιτούνται προσαρμογές ή εάν χρειάζεται να πραγματοποιήσετε κάποιες ενημερώσεις.

Η βάση δεδομένων είναι ένα θεμελιώδες μέρος για την εύρυθμη λειτουργία μιας εταιρείας, από τη διαχείριση εσωτερικών πληροφοριών έως τη διαχείριση περιεχομένου ιστοσελίδων και επιχειρηματικών ιστολογίων.

Ένα μοντέλο βάσης δεδομένων δείχνει τη λογική δομή μιας βάσης δεδομένων, συμπεριλαμβανομένων των σχέσεων και των περιορισμών που καθορίζουν τον τρόπο αποθήκευσης και πρόσβασης στα δεδομένα. Τα μεμονωμένα μοντέλα βάσεων δεδομένων σχεδιάζονται με βάση τους κανόνες και τις έννοιες του πιο ολοκληρωμένου μοντέλου δεδομένων που υιοθετούν οι σχεδιαστές. Τα περισσότερα μοντέλα δεδομένων μπορούν να αναπαρασταθούν από ένα συνοδευτικό διάγραμμα βάσης δεδομένων.

Τύποι μοντέλων βάσεων δεδομένων

Υπάρχουν διάφοροι τύποι μοντέλων δεδομένων. Μερικά από τα πιο κοινά είναι:

- Μοντέλο ιεραρχικής βάσης δεδομένων
- σχεσιακό μοντέλο
- μοντέλο δικτύου
- Αντικειμενοστραφή μοντέλο βάσης δεδομένων
- μοντέλο σχέσης οντότητας
- Μοντέλο ντοκιμαντέρ
- Μοντέλο οντότητας-ιδιότητας-τιμής
- σχέδιο αστεριών
- Το αντικειμενο-σχεσιακό μοντέλο, που συνδυάζει τα δύο που συνθέτουν το όνομά του

Μπορείτε να επιλέξετε να περιγράψετε μια βάση δεδομένων με οποιοδήποτε από αυτά τα μοντέλα, ανάλογα με διάφορους παράγοντες. Ο πιο σημαντικός παράγοντας είναι εάν το σύστημα διαχείρισης βάσης δεδομένων που χρησιμοποιείτε υποστηρίζει ένα συγκεκριμένο μοντέλο. Τα περισσότερα συστήματα διαχείρισης βάσεων δεδομένων έχουν κατασκευαστεί με ένα συγκεκριμένο μοντέλο δεδομένων και απαιτούν από τους χρήστες τους να υιοθετήσουν αυτό το μοντέλο, αν και ορισμένα υποστηρίζουν πολλά μοντέλα.

Επίσης, διαφορετικά μοντέλα εφαρμόζονται σε διαφορετικά στάδια της διαδικασίας δημιουργίας βάσης δεδομένων. Τα εννοιολογικά μοντέλα δεδομένων υψηλού επιπέδου είναι τα καλύτερα για τη χαρτογράφηση των σχέσεων μεταξύ των δεδομένων με τρόπο που οι άνθρωποι αντιλαμβάνονται αυτά τα δεδομένα. Τα λογικά μοντέλα που βασίζονται σε αρχεία καταγραφής, από την άλλη πλευρά, αντικατοπτρίζουν καλύτερα τους τρόπους με τους οποίους αποθηκεύονται τα δεδομένα στον διακομιστή.

Η επιλογή ενός μοντέλου δεδομένων είναι επίσης θέμα ευθυγράμμισης των προτεραιοτήτων σας για τη βάση δεδομένων με τα δυνατά σημεία ενός συγκεκριμένου μοντέλου, είτε αυτές οι προτεραιότητες περιλαμβάνουν ταχύτητα, εξοικονόμηση κόστους, χρηστικότητα ή οτιδήποτε άλλο.

Ας ρίξουμε μια ματιά σε μερικά από τα πιο κοινά μοντέλα βάσεων δεδομένων.

Σχεσιακό μοντέλο

Το πιο κοινό μοντέλο, το σχεσιακό μοντέλο, ταξινομεί τα δεδομένα σε πίνακες, γνωστούς και ως σχέσεις, καθένας από τους οποίους αποτελείται από στήλες και γραμμές. Κάθε στήλη παραθέτει ένα χαρακτηριστικό της εν λόγω οντότητας, όπως τιμή, ταχυδρομικός κώδικας ή ημερομηνία γέννησης. Μαζί, τα χαρακτηριστικά σε μια σχέση ονομάζονται τομέας. Ένα συγκεκριμένο χαρακτηριστικό ή συνδυασμός χαρακτηριστικών επιλέγεται ως πρωτεύον κλειδί που μπορεί να αναζητηθεί σε άλλους πίνακες, όταν ονομάζεται ξένο κλειδί.

Κάθε σειρά επίσης, περιλαμβάνει δεδομένα σχετικά με μια συγκεκριμένη παρουσία της εν λόγω οντότητας.

Το μοντέλο εξηγεί επίσης τους τύπους σχέσεων μεταξύ αυτών των πινάκων, συμπεριλαμβανομένων των σχέσεων ένα προς ένα, ένα προς πολλά και πολλά προς πολλά.

Εντός της βάσης δεδομένων, οι πίνακες μπορούν να κανονικοποιηθούν ή να διαμορφωθούν ώστε να συμμορφώνονται με κανόνες κανονικοποίησης που καθιστούν τη βάση δεδομένων ευέλικτη, προσαρμόσιμη και επεκτάσιμη. Όταν κανονικοποιείται, κάθε τμήμα δεδομένων είναι ατομικό ή αναλύεται σε μικρά, χρήσιμα κομμάτια.

Οι σχεσιακές βάσεις δεδομένων γράφονται συνήθως σε SQL (Structured Query Language). Το μοντέλο εισήχθη από τον EF Codd το 1970.

Ιεραρχικό μοντέλο

Το ιεραρχικό μοντέλο οργανώνει τα δεδομένα σε μια δομή που μοιάζει με δέντρο, όπου κάθε εγγραφή έχει έναν μόνο «γονέα» ή ρίζα.

Οι εγγραφές "αδελφών" ταξινομούνται με συγκεκριμένη σειρά. Αυτή η σειρά χρησιμοποιείται ως φυσική παραγγελία για την αποθήκευση της βάσης δεδομένων. Αυτό το μοντέλο είναι καλό για την περιγραφή πολλών σχέσεων στον πραγματικό κόσμο.

Αυτό το μοντέλο χρησιμοποιήθηκε κυρίως από τα Συστήματα Διαχείρισης Πληροφοριών της IBM στις δεκαετίες του 1960 και του 1970, αλλά σπάνια εμφανίζεται σήμερα λόγω ορισμένων λειτουργικών αναποτελεσματικών.

Μοντέλο δικτύου

Το μοντέλο δικτύου βασίζεται στο ιεραρχικό μοντέλο, επιτρέποντας σχέσεις πολλά-προς-πολλά μεταξύ συνδεδεμένων εγγραφών, υπονοώντας πολλαπλές εγγραφές "γονέα". Με βάση τη μαθηματική θεωρία συνόλων, το μοντέλο κατασκευάζεται με σύνολα σχετικών εγγραφών. Κάθε σετ αποτελείται από μια εγγραφή κατόχου ή "γονέα" και ένα ή περισσότερα αρχεία μελών ή "παιδιού". Ένας δίσκος μπορεί να είναι μέλος, ή "παιδί", σε πολλαπλά σετ, επιτρέποντας σε αυτό το μοντέλο να μεταφέρει σύνθετες σχέσεις.

Ήταν πιο δημοφιλές στη δεκαετία του 1970 αφού ορίστηκε επίσημα από το Conference on Data Systems Languages (CODASYL).

Αντικειμενοστραφή μοντέλο βάσης δεδομένων

Αυτό το μοντέλο ορίζει τη βάση δεδομένων ως μια συλλογή αντικειμένων ή επαναχρησιμοποιήσιμων στοιχείων λογισμικού, με σχετικούς πόρους και μεθόδους. Υπάρχουν διάφοροι τύποι αντικειμενοστρεφών βάσεων δεδομένων

Μια βάση δεδομένων πολυμέσων ενσωματώνει μέσα, όπως εικόνες, που δεν μπορούν να αποθηκευτούν σε μια σχεσιακή βάση δεδομένων.

Μια βάση δεδομένων υπερκειμένου επιτρέπει σε οποιοδήποτε αντικείμενο να συνδεθεί με οποιοδήποτε άλλο αντικείμενο. Είναι χρήσιμο για την οργάνωση παρτίδων διαφορετικών δεδομένων, αλλά δεν είναι ιδανικό για αριθμητική ανάλυση.

Το αντικειμενοστραφή μοντέλο βάσης δεδομένων είναι το πιο γνωστό μετασχεσιακό μοντέλο βάσης δεδομένων καθώς ενσωματώνει, αλλά δεν περιορίζεται σε, πίνακες.

Τέτοια μοντέλα είναι επίσης γνωστά ως μοντέλα υβριδικών βάσεων δεδομένων.

Αντικείμενο-σχεσιακό μοντέλο

Αυτό το μοντέλο υβριδικής βάσης δεδομένων συνδυάζει την απλότητα του σχεσιακού μοντέλου με ορισμένες από τις προηγμένες λειτουργίες του αντικειμενοστρεφούς μοντέλου βάσης δεδομένων. Στην ουσία, επιτρέπει στους σχεδιαστές να ενσωματώσουν αντικείμενα στη γνωστή δομή του πίνακα.

Οι διεπαφές γλωσσών και οι κλήσεις περιλαμβάνουν SQL3, γλώσσες προμηθευτή, ODBC, JDBC και ιδιόκτητες διεπαφές κλήσεων που είναι επεκτάσεις των γλωσσών και των διεπαφών που χρησιμοποιούνται από το σχεσιακό μοντέλο.

Μοντέλο σχέσης οντότητας

Αυτό το μοντέλο καταγράφει τις σχέσεις μεταξύ οντοτήτων του πραγματικού κόσμου, παρόμοιες με το μοντέλο δικτύου, αλλά δεν συνδέεται άμεσα με τη φυσική δομή της βάσης δεδομένων. Αντίθετα, χρησιμοποιείται συχνά για τον εννοιολογικό σχεδιασμό μιας βάσης δεδομένων.

Εδώ, άτομα, μέρη και πράγματα για τα οποία αποθηκεύονται σημεία δεδομένων και αναφέρονται ως οντότητες, καθένα από τα οποία έχει ορισμένα χαρακτηριστικά που μαζί αποτελούν τον τομέα του. Οι σχέσεις μεταξύ οντοτήτων χαρτογραφούνται επίσης.

Άλλα μοντέλα βάσεων δεδομένων

Μια μεγάλη ποικιλία από άλλα μοντέλα βάσεων δεδομένων χρησιμοποιήθηκαν ή εξακολουθούν να χρησιμοποιούνται σήμερα.

Πρότυπο ανεστραμμένου αρχείου

Μια βάση δεδομένων που έχει κατασκευαστεί με τη δομή του ανεστραμμένου αρχείου έχει σχεδιαστεί για να διευκολύνει τις γρήγορες αναζητήσεις πλήρους κειμένου. Σε αυτό το μοντέλο, το περιεχόμενο δεδομένων καταχωρείται ως μια σειρά κλειδιών σε έναν πίνακα αναζήτησης, με τις τιμές να δείχνουν τη θέση των συσχετισμένων αρχείων. Αυτό το πλαίσιο μπορεί να παρέχει σχεδόν στιγμιαία αναφορές για μεγάλα δεδομένα και αναλυτικά στοιχεία, για παράδειγμα.

Αυτό το μοντέλο χρησιμοποιείται από το σύστημα διαχείρισης βάσεων δεδομένων ADABAS της Software AG από το 1970 και εξακολουθεί να υποστηρίζεται σήμερα.

Επίπεδο μοντέλο

Το επίπεδο μοντέλο είναι το παλαιότερο και απλούστερο μοντέλο δεδομένων. Απλώς παραθέτει όλα τα δεδομένα σε έναν ενιαίο πίνακα, ο οποίος αποτελείται από στήλες και γραμμές. Για να έχει πρόσβαση ή να χειριστεί τα δεδομένα, ο υπολογιστής πρέπει να διαβάσει ολόκληρο το επίπεδο αρχείο στη μνήμη, γεγονός που καθιστά αυτό το μοντέλο αναποτελεσματικό για όλα εκτός από τα μικρότερα σύνολα δεδομένων.

Πολυδιάστατο μοντέλο

Αυτή είναι μια παραλλαγή του σχεσιακού μοντέλου που έχει σχεδιαστεί για να διευκολύνει τη βελτιωμένη αναλυτική επεξεργασία. Ενώ το σχεσιακό μοντέλο έχει βελτιστοποιηθεί για την ηλεκτρονική επεξεργασία συναλλαγών (OLTP), αυτό το μοντέλο έχει σχεδιαστεί για ηλεκτρονική αναλυτική επεξεργασία (OLAP).

Κάθε κελί σε μια βάση δεδομένων διαστάσεων περιέχει δεδομένα σχετικά με τις διαστάσεις που παρακολουθούνται από τη βάση δεδομένων. Οπτικά, είναι σαν μια συλλογή από κύβους και όχι σαν δισδιάστατους πίνακες.

Ημιδομημένο μοντέλο

Σε αυτό το μοντέλο, τα δομικά δεδομένα που συνήθως περιέχονται στο σχήμα της βάσης δεδομένων είναι ενσωματωμένα στα ίδια τα δεδομένα. Εδώ η διάκριση μεταξύ δεδομένων και σχήματος είναι στην καλύτερη περίπτωση ασαφής. Αυτό το μοντέλο είναι χρήσιμο για την περιγραφή συστημάτων, όπως ορισμένες πηγές δεδομένων που βασίζονται στον ιστό, που αντιμετωπίζουμε ως βάσεις δεδομένων, αλλά δεν μπορούμε να περιορίσουμε με ένα σχήμα. Είναι επίσης χρήσιμο για την περιγραφή αλληλεπιδράσεων μεταξύ βάσεων δεδομένων που δεν τηρούν το ίδιο σχήμα.

Μοντέλο πλαισίου

Αυτό το μοντέλο μπορεί να ενσωματώσει στοιχεία από άλλα μοντέλα βάσης δεδομένων, όπως απαιτείται.

Συγκεντρώνει στοιχεία από αντικειμενοστραφή, ημιδομημένα και δικτυακά μοντέλα.

Συνειρμικό μοντέλο

Αυτό το μοντέλο αναλύει όλα τα σημεία δεδομένων με βάση το αν περιγράφουν μια οντότητα ή μια συσχέτιση. Σε αυτό το μοντέλο, μια οντότητα είναι οτιδήποτε υπάρχει ανεξάρτητα, ενώ μια συσχέτιση είναι κάτι που υπάρχει μόνο σε σχέση με κάτι άλλο.

Το συσχετιστικό μοντέλο δομεί τα δεδομένα σε δύο σύνολα:

- Ένα σύνολο στοιχείων, το καθένα με ένα μοναδικό αναγνωριστικό, όνομα και τύπο
- Ένα σύνολο συνδέσμων, ο καθένας με ένα μοναδικό αναγνωριστικό και τα μοναδικά αναγνωριστικά μιας πηγής, ενός ρήματος και ενός προορισμού. Το γεγονός που αποθηκεύεται έχει να κάνει με την πηγή και καθένα από τα τρία αναγνωριστικά μπορεί να αναφέρεται σε έναν σύνδεσμο ή ένα στοιχείο.

Άλλα λιγότερο κοινά μοντέλα βάσεων δεδομένων περιλαμβάνουν:

- Σημασιολογικό μοντέλο, το οποίο περιλαμβάνει πληροφορίες σχετικά με το πώς τα αποθηκευμένα δεδομένα σχετίζονται με τον πραγματικό κόσμο
- Βάση δεδομένων XML, η οποία επιτρέπει τον καθορισμό και την αποθήκευση δεδομένων σε μορφή XML
- με όνομα γράφημα
- Τρία πατώματα

Μοντέλα βάσης δεδομένων χωρίς SQL

Εκτός από το μοντέλο της βάσης δεδομένων αντικειμένων, άλλα μοντέλα που δεν είναι SQL έχουν προκύψει σε σύγκριση με το σχεσιακό μοντέλο:

Το μοντέλο γραφικής βάσης δεδομένων, το οποίο είναι ακόμη πιο ευέλικτο από ένα μοντέλο δικτύου, επιτρέπει σε κάθε κόμβο να συνδεθεί με οποιονδήποτε άλλο.

Το μοντέλο πολλαπλών τιμών, το οποίο διαφέρει από το σχεσιακό μοντέλο επιτρέποντας στα χαρακτηριστικά να περιέχουν μια λίστα δεδομένων και όχι ένα μεμονωμένο σημείο δεδομένων.

Το μοντέλο εγγράφου , το οποίο έχει σχεδιαστεί για αποθήκευση και διαχείριση ημιδομημένων εγγράφων ή δεδομένων, αντί για ατομικά δεδομένα.

Βάσεις δεδομένων στο Διαδίκτυο

Οι περισσότεροι ιστότοποι βασίζονται σε κάποιο είδος βάσης δεδομένων για την οργάνωση και την παρουσίαση δεδομένων στους χρήστες. Κάθε φορά που κάποιος χρησιμοποιεί τις λειτουργίες αναζήτησης σε αυτούς τους ιστότοπους, οι όροι αναζήτησής του μετατρέπονται σε ερωτήματα για επεξεργασία από έναν διακομιστή βάσης δεδομένων. Συνήθως, το ενδιαμέσο λογισμικό συνδέει τον διακομιστή web με τη βάση δεδομένων.

Η ευρεία παρουσία των βάσεων δεδομένων επιτρέπει τη χρήση τους σχεδόν σε κάθε τομέα, από τις ηλεκτρονικές αγορές έως τη μικροστόχευση ενός τμήματος ψηφοφόρων ως μέρος μιας πολιτικής εκστρατείας. Διάφορες βιομηχανίες έχουν αναπτύξει τα δικά τους πρότυπα για τη δημιουργία βάσεων δεδομένων, από τις αεροπορικές μεταφορές έως την κατασκευή οχημάτων.

Όταν είστε έτοιμοι να ξεκινήσετε τη μοντελοποίηση της βάσης δεδομένων σας, δοκιμάστε το Lucidchart. Είναι το τέλειο εργαλείο για τη σχεδίαση συστημάτων βάσεων δεδομένων γρήγορα και εύκολα. Μπορείτε ακόμη να κάνετε εισαγωγές και εξαγωγές SQL.

Μοντέλο βάσης δεδομένων

Ένα μοντέλο βάσης δεδομένων είναι ένας τύπος μοντέλου δεδομένων που καθορίζει τη λογική δομή μιας βάσης δεδομένων και , ουσιαστικά, καθορίζει τον τρόπο αποθήκευσης , οργάνωσης και επεξεργασίας δεδομένων. Το πιο δημοφιλές παράδειγμα μοντέλου βάσης δεδομένων είναι το σχεσιακό μοντέλο , το οποίο χρησιμοποιεί μια μορφή που βασίζεται σε πίνακα.

Παραδείγματα

Τα κοινά λογικά μοντέλα δεδομένων για βάσεις δεδομένων περιλαμβάνουν:

Μοντέλο ιεραρχικής βάσης δεδομένων

Είναι η παλαιότερη μορφή μοντέλου βάσης δεδομένων. Αναπτύχθηκε από την IBM για το IMS (Information Management System). Είναι ένα σύνολο δεδομένων οργανωμένων σε δομή δέντρου. Η εγγραφή της βάσης δεδομένων είναι ένα δέντρο που αποτελείται από πολλές ομάδες που ονομάζονται τμήματα. Χρησιμοποιεί ένα για πολλές σχέσεις. Η πρόσβαση στα δεδομένα είναι επίσης προβλέψιμη.

μοντέλο δικτύου

σχεσιακό μοντέλο

Μοντέλο σχέσης οντότητας

Βελτιωμένο μοντέλο σχέσης οντότητας

μοντέλο αντικειμένου

Μοντέλο εγγράφου

Μοντέλο οντότητας-ιδιότητας-τιμής

σχέδιο αστεριών

Μια αντικείμενο-σχεσιακή βάση δεδομένων συνδυάζει τις δύο σχετικές δομές.

Τα μοντέλα φυσικών δεδομένων περιλαμβάνουν:

- ανεστραμμένος δείκτης
- επίπεδο αρχείο

Άλλα μοντέλα περιλαμβάνουν:

- συνειρμικό μοντέλο
- Συσχετιστικό μοντέλο
- πολυδιάστατο μοντέλο
- Μοντέλο πολλαπλών τιμών
- σημασιολογικό μοντέλο
- Βάση δεδομένων XML

- με όνομα γράφημα
- Triplestore

Σχέσεις και ρόλους

Ένα δεδομένο σύστημα διαχείρισης βάσεων δεδομένων μπορεί να παρέχει ένα ή περισσότερα μοντέλα. Η βέλτιστη δομή εξαρτάται από τη φυσική οργάνωση των δεδομένων της εφαρμογής και τις απαιτήσεις εφαρμογής, οι οποίες περιλαμβάνουν ρυθμό συναλλαγής (ταχύτητα), αξιοπιστία, δυνατότητα συντήρησης, επεκτασιμότητα και κόστος. Τα περισσότερα συστήματα διαχείρισης βάσεων δεδομένων είναι χτισμένα γύρω από ένα συγκεκριμένο μοντέλο δεδομένων, αν και είναι δυνατό τα προϊόντα να υποστηρίζουν περισσότερα από ένα μοντέλα.

Πολλαπλά μοντέλα φυσικών δεδομένων μπορούν να εφαρμόσουν οποιοδήποτε δεδομένο λογικό μοντέλο. Τα περισσότερα λογισμικά βάσεων δεδομένων παρέχουν στο χρήστη κάποιο επίπεδο ελέγχου στη ρύθμιση της φυσικής υλοποίησης, καθώς οι επιλογές που γίνονται έχουν σημαντική επίδραση στην απόδοση.

Ένα μοντέλο δεν είναι απλώς ένας τρόπος δόμησης δεδομένων: ορίζει επίσης ένα σύνολο λειτουργιών που μπορούν να εκτελεστούν στα δεδομένα. [1] Το σχεσιακό μοντέλο, για παράδειγμα, ορίζει λειτουργίες όπως επιλογή (έργο) και join . Αν και αυτές οι λειτουργίες μπορεί να μην είναι σαφείς σε μια συγκεκριμένη γλώσσα ερωτήματος, παρέχουν τη βάση πάνω στην οποία έχει χτιστεί μια γλώσσα ερωτήματος.

Επίπεδο μοντέλο

Το απλό μοντέλο (ή πίνακας) αποτελείται από έναν ενιαίο δισδιάστατο πίνακα στοιχείων δεδομένων, όπου όλα τα μέλη μιας δεδομένης στήλης θεωρούνται παρόμοιες τιμές και όλα τα μέλη μιας σειράς θεωρούνται σχετιζόμενα μεταξύ τους. Για παράδειγμα, στήλες για όνομα και κωδικό πρόσβασης που μπορούν να χρησιμοποιηθούν ως μέρος μιας βάσης δεδομένων ασφαλείας συστήματος. Κάθε γραμμή θα έχει τον συγκεκριμένο κωδικό πρόσβασης που σχετίζεται με έναν μεμονωμένο χρήστη. Οι στήλες του πίνακα συνήθως έχουν έναν τύπο που σχετίζεται με αυτές, ορίζοντας τους ως δεδομένα χαρακτήρων, πληροφορίες

ημερομηνίας ή ώρας, ακέραιους αριθμούς ή αριθμούς κινητής υποδιαστολής. Αυτή η μορφή πίνακα είναι ένας πρόδρομος του σχεσιακού μοντέλου.

Αρχικά μοντέλα δεδομένων

Αυτά τα μοντέλα ήταν δημοφιλή στις δεκαετίες του 1960 και του 1970, αλλά σήμερα μπορούν να βρεθούν κυρίως σε παλαιότερα συστήματα παλαιού τύπου. Χαρακτηρίζονται κυρίως από την πλοήγηση με ισχυρές συνδέσεις μεταξύ των λογικών και φυσικών αναπαραστάσεων τους και τις ελλείψεις στην ανεξαρτησία δεδομένων.

Ιεραρχικό μοντέλο

Σε ένα ιεραρχικό μοντέλο, τα δεδομένα οργανώνονται σε μια δομή που μοιάζει με δέντρο, υπονοώντας έναν μόνο γονέα για κάθε εγγραφή. Ένα πεδίο ταξινόμησης διατηρεί αρχεία αδελφών με συγκεκριμένη σειρά. Οι ιεραρχικές δομές χρησιμοποιήθηκαν ευρέως σε πρώιμα συστήματα διαχείρισης βάσεων δεδομένων mainframe, όπως το σύστημα διαχείρισης πληροφοριών της IBM (IMS) και τώρα περιγράφουν τη δομή των εγγράφων XML. Αυτή η δομή επιτρέπει μια σχέση ένα προς πολλά μεταξύ δύο τύπων δεδομένων. Αυτό το πλαίσιο είναι πολύ αποτελεσματικό για την περιγραφή πολλών σχέσεων στον πραγματικό κόσμο. συνταγές, πίνακας περιεχομένων, διάταξη παραγράφων/στίχων, τυχόν ένθετες και ταξινομημένες πληροφορίες.

Αυτή η ιεραρχία χρησιμοποιείται ως η φυσική σειρά των εγγραφών στο κατάστημα. Η πρόσβαση στην εγγραφή γίνεται με πλοήγηση προς τα κάτω στη δομή δεδομένων χρησιμοποιώντας δείκτες σε συνδυασμό με διαδοχική πρόσβαση. Εξαιτίας αυτού, η ιεραρχική δομή είναι αναποτελεσματική για ορισμένες λειτουργίες βάσης δεδομένων όταν δεν περιλαμβάνεται επίσης μια πλήρης διαδρομή (σε αντίθεση με το πεδίο urlink και ταξινόμησης) για κάθε εγγραφή. Αυτοί οι περιορισμοί αντισταθμίστηκαν σε μεταγενέστερες εκδόσεις του IMS από πρόσθετες λογικές ιεραρχίες που επιβλήθηκαν στη βασική φυσική ιεραρχία.

Μοντέλο δικτύου

Το μοντέλο δικτύου επεκτείνει την ιεραρχική δομή επιτρέποντας σχέσεις πολλά-προς-πολλά σε μια δομή που μοιάζει με δέντρο που επιτρέπει πολλούς γονείς. Ήταν

πιο δημοφιλές πριν αντικατασταθεί από το σχεσιακό μοντέλο και ορίζεται από την προδιαγραφή CODASYL.

Το μοντέλο δικτύου οργανώνει δεδομένα χρησιμοποιώντας δύο θεμελιώδεις έννοιες, που ονομάζονται εγγραφές και σύνολα . Οι εγγραφές περιέχουν πεδία (τα οποία μπορούν να ταξινομηθούν ιεραρχικά, όπως στη γλώσσα προγραμματισμού COBOL). Τα σύνολα (δεν πρέπει να συγχέονται με τα μαθηματικά σύνολα) ορίζουν σχέσεις ένα προς πολλά μεταξύ των εγγραφών: ένας ιδιοκτήτης, πολλά μέλη. Μια εγγραφή μπορεί να είναι κάτοχος σε οποιονδήποτε αριθμό σετ και μέλος σε οποιονδήποτε αριθμό σετ.

Ένα σύνολο αποτελείται από συνδεδεμένες κυκλικές λίστες όπου ένας τύπος εγγραφής, ο κάτοχος ή ο γονέας του συνόλου, εμφανίζεται μία φορά σε κάθε κύκλο και ένας δεύτερος τύπος εγγραφής, ο υφιστάμενος ή το παιδί, μπορεί να εμφανιστεί πολλές φορές σε κάθε κύκλο. Με αυτόν τον τρόπο, μπορεί να δημιουργηθεί μια ιεραρχία μεταξύ οποιωνδήποτε δύο τύπων εγγραφών, για παράδειγμα, ο τύπος A είναι ο ιδιοκτήτης του B. Ταυτόχρονα, μπορεί να οριστεί ένα άλλο σύνολο όπου ο B είναι ο ιδιοκτήτης του A. Έτσι, όλα τα σύνολα περιλαμβάνουν ένα γενικό κατευθυνόμενο γράφημα (η ιδιότητα ορίζει μια κατεύθυνση) ή κατασκευή δικτύου . Η πρόσβαση στις εγγραφές είναι διαδοχική (γενικά σε κάθε τύπο εγγραφής) ή με περιήγηση σε συνδεδεμένες κυκλικές λίστες.

Το μοντέλο δικτύου είναι σε θέση να αναπαραστήσει τον πλεονασμό σε δεδομένα πιο αποτελεσματικά από το ιεραρχικό μοντέλο και μπορεί να υπάρχουν περισσότερα από ένα μονοπάτια από έναν προγονικό κόμβο σε έναν απόγονο. Οι λειτουργίες του μοντέλου δικτύου έχουν στυλ πλοήγησης: ένα πρόγραμμα διατηρεί μια τρέχουσα θέση και πλοηγείται από εγγραφή σε εγγραφή ακολουθώντας τις σχέσεις στις οποίες συμμετέχει η εγγραφή. Οι εγγραφές μπορούν επίσης να εντοπιστούν παρέχοντας βασικές τιμές.

Αν και δεν είναι ουσιαστικό χαρακτηριστικό του μοντέλου, οι βάσεις δεδομένων δικτύου συχνά υλοποιούν σχέσεις συνόλου μέσω δεικτών που κατευθύνουν άμεσα τη θέση μιας εγγραφής στο δίσκο. Αυτό παρέχει εξαιρετική απόδοση ανάκτησης, σε βάρος λειτουργιών όπως η φόρτωση και η αναδιοργάνωση της βάσης δεδομένων.

Τα δημοφιλή προϊόντα DBMS που το χρησιμοποιούσαν ήταν το Total της Cincom Systems και το IDMS της Cullinet. Η IDMS έχει αποκτήσει σημαντική πελατειακή βάση. τη δεκαετία του 1980, υιοθέτησε το σχεσιακό μοντέλο και την SQL, εκτός από τα αρχικά εργαλεία και τις γλώσσες της.

Οι περισσότερες βάσεις δεδομένων αντικειμένων (που εφευρέθηκαν τη δεκαετία του 1990) χρησιμοποιούν την έννοια της πλοήγησης για να παρέχουν γρήγορη πλοήγηση μέσω δικτύων αντικειμένων, χρησιμοποιώντας συχνά αναγνωριστικά αντικειμένων ως «έξυπνους» δείκτες σε σχετικά αντικείμενα. Το Objectivity/DB, για παράδειγμα, υλοποιεί σχέσεις με το όνομα one-to-one, one-to-many, many-to-one και many-to-many που μπορούν να διασχίσουν βάσεις δεδομένων. Πολλές βάσεις δεδομένων αντικειμένων υποστηρίζουν επίσης SQL, συνδυάζοντας τα δυνατά σημεία και των δύο μοντέλων.

Πρότυπο ανεστραμμένου αρχείου

Σε ένα ανεστραμμένο αρχείο ή ένα ανεστραμμένο ευρετήριο, το περιεχόμενο δεδομένων χρησιμοποιείται ως κλειδί σε έναν πίνακα αναζήτησης και οι τιμές στον πίνακα είναι δείκτες στη θέση κάθε παρουσίας ενός δεδομένου στοιχείου περιεχομένου. Αυτή είναι επίσης η λογική δομή των σύγχρονων ευρετηρίων βάσεων δεδομένων, τα οποία μπορούν να χρησιμοποιήσουν μόνο τα περιεχόμενα μιας συγκεκριμένης στήλης στον πίνακα αναζήτησης. Το μοντέλο δεδομένων ανεστραμμένου αρχείου μπορεί να τοποθετήσει ευρετήρια σε ένα σύνολο αρχείων μαζί με υπάρχοντα επίπεδα αρχεία βάσης δεδομένων, προκειμένου να έχει αποτελεσματική πρόσβαση στις απαραίτητες εγγραφές σε αυτά τα αρχεία.

Αξιοσημείωτο για τη χρήση αυτού του μοντέλου δεδομένων είναι το ADABAS DBMS της Software AG, που παρουσιάστηκε το 1970. Η ADABAS απέκτησε σημαντική πελατειακή βάση και υπάρχει και υποστηρίζεται μέχρι σήμερα. Στη δεκαετία του 1980, υιοθέτησε το σχεσιακό και SQL μοντέλο, εκτός από τα αρχικά εργαλεία και τις γλώσσες του.

Η βάση δεδομένων προσανατολισμένη στα έγγραφα του Clusterpoint χρησιμοποιεί μοντέλο ανεστραμμένου ευρετηρίου για την παροχή γρήγορης αναζήτησης πλήρους κειμένου για αντικείμενα δεδομένων XML ή JSON, για παράδειγμα.

Σχεσιακό μοντέλο

Το σχεσιακό μοντέλο εισήχθη από τον EF Codd το 1970 ως ένας τρόπος να γίνουν τα συστήματα διαχείρισης βάσεων δεδομένων πιο ανεξάρτητα από οποιαδήποτε συγκεκριμένη εφαρμογή. Είναι ένα μαθηματικό μοντέλο που ορίζεται από την άποψη της λογικής κατηγορημάτων και της θεωρίας συνόλων και οι υλοποιήσεις του έχουν χρησιμοποιηθεί από συστήματα mainframe, μεσαίου μεγέθους και μικροϋπολογιστών.

Τα προϊόντα που συνήθως ονομάζονται σχεσιακές βάσεις δεδομένων εφαρμόζουν στην πραγματικότητα ένα μοντέλο που είναι απλώς μια προσέγγιση του μαθηματικού μοντέλου που ορίζεται από τον Codd. Τρεις βασικοί όροι χρησιμοποιούνται εκτενώς σε μοντέλα σχεσιακών βάσεων δεδομένων, *σχέσεις*, *ιδιότητες και τομείς*. *Μια σχέση είναι ένας πίνακας με στήλες και γραμμές. Οι ονομασμένες στήλες της σχέσης ονομάζονται χαρακτηριστικά και ο τομέας είναι το σύνολο τιμών που μπορούν να λάβουν τα χαρακτηριστικά.*

Η βασική δομή δεδομένων του σχεσιακού μοντέλου είναι ο πίνακας, όπου οι πληροφορίες για μια συγκεκριμένη οντότητα (για παράδειγμα, έναν υπάλληλο) αναπαρίστανται σε σειρές (ονομάζονται επίσης πλειάδες) και στήλες. Έτσι, η "σχέση" στη "σχεσιακή βάση δεδομένων" αναφέρεται στους διάφορους πίνακες της βάσης δεδομένων. μια σχέση είναι ένα σύνολο πλειάδων. Οι στήλες απαριθμούν τα διάφορα χαρακτηριστικά της οντότητας (για παράδειγμα το όνομα, τη διεύθυνση ή τον αριθμό τηλεφώνου του υπαλλήλου) και μια σειρά είναι μια πραγματική παρουσία της οντότητας (ένας συγκεκριμένος υπάλληλος) που αντιπροσωπεύεται από τη σχέση. Ως αποτέλεσμα, κάθε πλειάδα στον πίνακα υπαλλήλων αντιπροσωπεύει πολλαπλά χαρακτηριστικά ενός μεμονωμένου υπαλλήλου.

Όλες οι σχέσεις (και επομένως οι πίνακες) σε μια σχεσιακή βάση δεδομένων πρέπει να συμμορφώνονται με ορισμένους βασικούς κανόνες για να χαρακτηριστούν ως σχέσεις. Πρώτον, η σειρά των στηλών είναι άσχετη σε έναν πίνακα. Δεύτερον, δεν μπορούν να υπάρχουν πανομοιότυπες πλειάδες ή σειρές σε έναν πίνακα. Και τρίτον, κάθε πλειάδα θα περιέχει μια ενιαία τιμή για κάθε χαρακτηριστικό της.

Μια σχεσιακή βάση δεδομένων περιέχει πολλούς πίνακες, ο καθένας παρόμοιος με το μοντέλο της «απλής» βάσης δεδομένων. Ένα από τα δυνατά σημεία του σχεσιακού μοντέλου είναι ότι, καταρχήν, οποιαδήποτε τιμή εμφανίζεται σε δύο διαφορετικές εγγραφές (που ανήκουν στον ίδιο πίνακα ή σε διαφορετικούς πίνακες), συνεπάγεται μια σχέση μεταξύ αυτών των δύο εγγραφών. Ωστόσο, προκειμένου να επιβληθούν σαφείς περιορισμοί ακεραιότητας, οι σχέσεις μεταξύ των εγγραφών σε πίνακες μπορούν επίσης να οριστούν ρητά, προσδιορίζοντας ή μη προσδιορίζοντας τις σχέσεις γονέα-παιδιού που χαρακτηρίζονται από εκχώρηση (1:1, (0)1:M, M: M) Πίνακες μπορεί επίσης να έχει ένα μεμονωμένο εκχωρημένο χαρακτηριστικό ή ένα σύνολο χαρακτηριστικών που μπορεί να λειτουργήσει ως "κλειδί", το οποίο μπορεί να χρησιμοποιηθεί για τον μοναδικό προσδιορισμό κάθε πλειάδας στον πίνακα.

Ένα κλειδί που μπορεί να χρησιμοποιηθεί για τον μοναδικό προσδιορισμό μιας σειράς σε έναν πίνακα ονομάζεται πρωτεύον κλειδί. Τα κλειδιά χρησιμοποιούνται συνήθως για τη σύνδεση ή το συνδυασμό δεδομένων από δύο ή περισσότερους πίνακες. Για παράδειγμα, ένας πίνακας Υπαλλήλων μπορεί να περιέχει μια στήλη που ονομάζεται Θέση που περιέχει μια τιμή που αντιστοιχεί στο κλειδί ενός πίνακα Τοποθεσίας. Τα κλειδιά είναι επίσης κρίσιμα για τη δημιουργία ευρετηρίων, τα οποία διευκολύνουν τη γρήγορη ανάκτηση δεδομένων από μεγάλους πίνακες. Οποιαδήποτε στήλη μπορεί να είναι κλειδί ή πολλές στήλες μπορούν να ομαδοποιηθούν σε ένα σύνθετο κλειδί. Δεν είναι απαραίτητο να ρυθμίσετε όλα τα πλήκτρα εκ των προτέρων. μια στήλη μπορεί να χρησιμοποιηθεί ως κλειδί, ακόμα κι αν δεν προοριζόταν αρχικά να είναι μια.

Ένα κλειδί που έχει μια εξωτερική, πραγματική σημασία (όπως το όνομα ενός ατόμου, το ISBN ενός βιβλίου ή ο σειριακός αριθμός ενός αυτοκινήτου) ονομάζεται μερικές φορές "φυσικό" κλειδί. Εάν κανένα φυσικό κλειδί δεν είναι κατάλληλο (σκεφτείτε τους πολλούς ανθρώπους που ονομάζονται Brown), μπορεί να εκχωρηθεί ένα αυθαίρετο ή υποκατάστατο κλειδί (π.χ. παροχή αριθμών ταυτότητας στους υπαλλήλους). Στην πράξη, οι περισσότερες βάσεις δεδομένων διαθέτουν και δημιουργημένα και φυσικά κλειδιά, επειδή τα δημιουργημένα κλειδιά μπορούν να χρησιμοποιηθούν εσωτερικά για τη δημιουργία συνδέσμων μεταξύ σειρών που δεν

μπορούν να σπάσουν, ενώ τα φυσικά κλειδιά μπορούν να χρησιμοποιηθούν, λιγότερο αξιόπιστα, για αναζητήσεις και ενοποίηση με άλλες βάσεις δεδομένων. (Για παράδειγμα, οι εγγραφές σε δύο βάσεις δεδομένων που έχουν αναπτυχθεί ανεξάρτητα μπορούν να συγκριθούν με αριθμό κοινωνικής ασφάλισης, εκτός εάν οι αριθμοί κοινωνικής ασφάλισης είναι λανθασμένοι, λείπουν ή έχουν αλλάξει.)

Η πιο κοινή γλώσσα ερωτημάτων που χρησιμοποιείται με το σχεσιακό μοντέλο είναι η δομημένη γλώσσα ερωτημάτων (SQL).

Μοντέλο διαστάσεων

Το μοντέλο διαστάσεων είναι μια εξειδικευμένη προσαρμογή του σχεσιακού μοντέλου που χρησιμοποιείται για την αναπαράσταση δεδομένων σε αποθήκες δεδομένων, έτσι ώστε τα δεδομένα να μπορούν εύκολα να συνοψιστούν χρησιμοποιώντας διαδικτυακή αναλυτική επεξεργασία ή ερωτήματα OLAP. Στο μοντέλο διαστάσεων, ένα σχήμα βάσης δεδομένων αποτελείται από έναν ενιαίο μεγάλο πίνακα γεγονότων που περιγράφονται χρησιμοποιώντας διαστάσεις και μέτρα. Μια διάσταση παρέχει το πλαίσιο ενός συμβάντος (όπως ποιος συμμετείχε, πότε και πού συνέβη και τον τύπο του) και χρησιμοποιείται σε ερωτήματα για την ομαδοποίηση σχετικών γεγονότων. Οι διαστάσεις τείνουν να είναι διακριτές και συχνά ιεραρχικές. για παράδειγμα, η τοποθεσία μπορεί να περιλαμβάνει το κτίριο, την πολιτεία και τη χώρα. Ένα μέτρο είναι μια ποσότητα που περιγράφει το γεγονός, όπως τα έσοδα. Είναι σημαντικό τα μέτρα να μπορούν να συγκεντρωθούν με ουσιαστικό τρόπο - για παράδειγμα, μπορούν να προστεθούν έσοδα από διαφορετικές τοποθεσίες.

Σε ένα ερώτημα OLAP, επιλέγονται οι διαστάσεις και τα γεγονότα ομαδοποιούνται και συγκεντρώνονται για να δημιουργηθεί μια σύνοψη.

Το μοντέλο διαστάσεων εφαρμόζεται συχνά πάνω από το σχεσιακό μοντέλο χρησιμοποιώντας ένα σχήμα αστεριού, που αποτελείται από έναν εξαιρετικά κανονικοποιημένο πίνακα που περιέχει τα γεγονότα και γύρω από κανονικοποιημένους πίνακες που περιέχουν κάθε διάσταση. Μια εναλλακτική

φυσική υλοποίηση, που ονομάζεται σχήμα νιφάδας χιονιού, κανονικοποιεί τις ιεραρχίες πολλαπλών επιπέδων μέσα σε μια διάσταση σε πολλούς πίνακες.

Μια αποθήκη δεδομένων μπορεί να περιέχει σχήματα πολλαπλών διαστάσεων που μοιράζονται πίνακες διαστάσεων, επιτρέποντάς τους να χρησιμοποιούνται μαζί. Η δημιουργία ενός τυπικού συνόλου διαστάσεων είναι ένα σημαντικό μέρος της μοντελοποίησης διαστάσεων.

Η υψηλή του απόδοση έχει κάνει το μοντέλο διαστάσεων την πιο δημοφιλή δομή βάσης δεδομένων για το OLAP.

Μετασχεσιακά μοντέλα βάσεων δεδομένων

Τα προϊόντα που προσφέρουν ένα πιο γενικό μοντέλο δεδομένων από το σχεσιακό μοντέλο ταξινομούνται μερικές φορές ως μετασχεσιακά. Οι εναλλακτικοί όροι περιλαμβάνουν "υβριδική βάση δεδομένων", "Βελτιωμένο με αντικείμενο RDBMS" και άλλους. Το μοντέλο δεδομένων σε τέτοια προϊόντα ενσωματώνει σχέσεις, αλλά δεν περιορίζεται από την Αρχή Πληροφοριών του EF Codd, η οποία απαιτεί όλες οι πληροφορίες της βάσης δεδομένων πρέπει να εκφράζονται ρητά με όρους τιμών σε σχέσεις και όχι διαφορετικά

Μερικές από αυτές τις επεκτάσεις του σχεσιακού μοντέλου ενσωματώνουν έννοιες από τεχνολογίες πριν από το σχεσιακό μοντέλο. Για παράδειγμα, επιτρέπουν την αναπαράσταση ενός κατευθυνόμενου γραφήματος με δέντρα στους κόμβους. Η γερμανική εταιρεία sones εφαρμόζει αυτή την ιδέα στο GraphDB της.

Ορισμένα μετασχεσιακά προϊόντα επεκτείνουν τα σχεσιακά συστήματα με μη σχεσιακά χαρακτηριστικά.

Άλλοι έχουν φτάσει σχεδόν στο ίδιο μέρος, προσθέτοντας σχεσιακές δυνατότητες στα προ σχεσιακά συστήματα.

Παραδόξως, αυτό επιτρέπει σε προϊόντα που είναι ιστορικά προ-σχεσιακά, όπως το PICK και το MUMPS, να κάνουν έναν εύλογο ισχυρισμό ότι είναι μετασχεσιακά.

Το μοντέλο χώρου πόρων (RSM) είναι ένα μη σχεσιακό μοντέλο δεδομένων που βασίζεται σε πολυδιάστατη ταξινόμηση.

Γραφικό μοντέλο

Οι γραφικές βάσεις δεδομένων επιτρέπουν μια ακόμη πιο γενική δομή από μια βάση δεδομένων δικτύου. οποιοσδήποτε κόμβος μπορεί να συνδεθεί με οποιονδήποτε άλλο κόμβο.

Μοντέλο πολλαπλών τιμών

Οι βάσεις δεδομένων πολλαπλών τιμών είναι «αθροιστικά» δεδομένα καθώς μπορούν να αποθηκεύσουν με τον ίδιο ακριβώς τρόπο όπως οι σχεσιακές βάσεις δεδομένων, αλλά επιτρέπουν επίσης ένα επίπεδο βάθους που το σχεσιακό μοντέλο μπορεί να προσεγγίσει μόνο με τη χρήση υποπίνακων. Αυτό είναι σχεδόν πανομοιότυπο με τον τρόπο που η XML εκφράζει δεδομένα, όπου ένα δεδομένο πεδίο/χαρακτηριστικό μπορεί να έχει πολλές σωστές απαντήσεις ταυτόχρονα. Πολλαπλές τιμές μπορούν να θεωρηθούν ως συμπιεσμένη μορφή XML.

Ένα παράδειγμα είναι ένα τιμολόγιο, το οποίο σε δεδομένα πολλών τιμών ή σχεσιακά μπορεί να φανεί ως (Α) Πίνακας κεφαλίδας τιμολογίου - μία καταχώρηση ανά τιμολόγιο και (Β) πίνακας λεπτομερειών τιμολογίου - μία καταχώρηση ανά στοιχείο γραμμής. Στο μοντέλο πολλαπλών τιμών, έχουμε την επιλογή αποθήκευσης των δεδομένων όπως στον πίνακα, με έναν ενσωματωμένο πίνακα που αντιπροσωπεύει τις λεπτομέρειες: (Α) Πίνακας τιμολογίων - μία καταχώρηση ανά τιμολόγιο, δεν χρειάζεται άλλος πίνακας.

Το πλεονέκτημα είναι ότι η ατομικότητα του Τιμολογίου (εννοιολογική) και του Τιμολογίου (αναπαράσταση δεδομένων) είναι ένα προς ένα. Αυτό έχει επίσης ως αποτέλεσμα λιγότερες αναγνώσεις, λιγότερα ζητήματα ακεραιότητας αναφοράς και δραστική μείωση του υλικού που απαιτείται για την υποστήριξη ενός δεδομένου όγκου συναλλαγών.

Αντικειμενοστραφή μοντέλα βάσεων δεδομένων

Στη δεκαετία του 1990, το πρότυπο αντικειμενοστρεφούς προγραμματισμού εφαρμόστηκε στην τεχνολογία βάσεων δεδομένων, δημιουργώντας ένα νέο μοντέλο βάσης δεδομένων γνωστό ως βάσεις δεδομένων αντικειμένων. Αυτό γίνεται για να αποφευχθεί η αναντιστοιχία αντικειμενικής σχέσης σύνθετης αντίστασης - η επιβάρυνση της μετατροπής πληροφοριών μεταξύ της

αναπαράστασής τους στη βάση δεδομένων (π.χ. ως σειρές σε πίνακες) και της αναπαράστασής τους στο πρόγραμμα εφαρμογής (συνήθως ως αντικείμενα). Επιπλέον, το σύστημα τύπου που χρησιμοποιείται σε μια συγκεκριμένη εφαρμογή μπορεί να οριστεί απευθείας στη βάση δεδομένων, επιτρέποντας στη βάση δεδομένων να επιβάλλει τις ίδιες αμετάβλητες ακεραιότητα δεδομένων. Οι βάσεις δεδομένων αντικειμένων εισάγουν επίσης τις βασικές ιδέες του προγραμματισμού αντικειμένων, όπως η ενθυλάκωση και ο πολυμορφισμός, στον κόσμο των βάσεων δεδομένων.

Αρκετοί από αυτούς τους τρόπους έχουν δοκιμαστεί [από ποιον;] για την αποθήκευση αντικειμένων σε μια βάση δεδομένων. Μερικά [ποια;] προϊόντα έχουν αντιμετωπίσει το πρόβλημα από το τέλος του προγραμματισμού εφαρμογών κάνοντας τα αντικείμενα που χειρίζονται το πρόγραμμα να είναι μόνιμα. Αυτό τυπικά απαιτεί την προσθήκη κάποιου είδους γλώσσας ερωτημάτων, καθώς οι συμβατικές γλώσσες προγραμματισμού δεν έχουν τη δυνατότητα να εντοπίζουν αντικείμενα με βάση το πληροφοριακό τους περιεχόμενο. Άλλοι [ποια;] επιτέθηκε στο πρόβλημα από την πλευρά της βάσης δεδομένων ορίζοντας ένα αντικειμενοστραφή μοντέλο δεδομένων για τη βάση δεδομένων και ορίζοντας μια γλώσσα προγραμματισμού βάσης δεδομένων που επιτρέπει πλήρεις δυνατότητες προγραμματισμού καθώς και παραδοσιακές δυνατότητες ερωτημάτων.

Οι βάσεις δεδομένων αντικειμένων υπέφεραν από έλλειψη τυποποίησης: αν και τα πρότυπα καθορίστηκαν από την ODMG, δεν εφαρμόστηκαν ποτέ αρκετά καλά ώστε να διασφαλιστεί η διαλειτουργικότητα μεταξύ των προϊόντων. Ωστόσο, οι βάσεις δεδομένων αντικειμένων έχουν χρησιμοποιηθεί με επιτυχία σε πολλές εφαρμογές: συνήθως εξειδικευμένες εφαρμογές όπως βάσεις δεδομένων μηχανικής ή βάσεις δεδομένων μοριακής βιολογίας, αντί για συμβατική εμπορική επεξεργασία δεδομένων. Ωστόσο, οι ιδέες της βάσης δεδομένων αντικειμένων ελήφθησαν από σχεσιακούς προμηθευτές και επηρέασαν τις επεκτάσεις που έγιναν σε αυτά τα προϊόντα και μάλιστα στη γλώσσα SQL.

Μια εναλλακτική λύση στη μετάφραση μεταξύ αντικειμένων και σχεσιακών βάσεων δεδομένων είναι η χρήση μιας βιβλιοθήκης αντικειμενικής σχέσης αντιστοίχισης (ORM).

Εκπαιδευτική υποενότητα : 5.3 Σχεδιασμός Μοντέλου Οντοτήτων- Συσχετίσεων

Εννοιολογικά, λογικά και φυσικά μοντέλα δεδομένων

Τα μοντέλα ER και τα μοντέλα δεδομένων σχεδιάζονται συνήθως σε έως και τρία επίπεδα λεπτομέρειας:

Εννοιολογικό μοντέλο δεδομένων: η προβολή υψηλότερου επιπέδου που περιέχει τις λιγότερες λεπτομέρειες. Η αξία του είναι να δείξει ένα γενικό πεδίο εφαρμογής του μοντέλου και να απεικονίσει την αρχιτεκτονική του συστήματος. Για ένα σύστημα μικρότερης εμβέλειας, ο σχεδιασμός μπορεί να μην είναι απαραίτητος. Αντίθετα, ξεκινήστε με ένα λογικό μοντέλο.

Μοντέλο Λογικών Δεδομένων: Περιέχει περισσότερες λεπτομέρειες από ένα εννοιολογικό μοντέλο και τώρα ορίζονται πιο λεπτομερείς λειτουργικές και συναλλακτικές οντότητες. Το λογικό μοντέλο είναι ανεξάρτητο από την τεχνολογία στην οποία θα εφαρμοστεί.

Μοντέλο φυσικών δεδομένων: Ένα ή περισσότερα φυσικά μοντέλα μπορούν να αναπτυχθούν από κάθε λογικό μοντέλο.

Τα φυσικά μοντέλα πρέπει να παρουσιάζουν αρκετές τεχνολογικές λεπτομέρειες για την παραγωγή και την υλοποίηση της βάσης δεδομένων.

Σημειώστε ότι υπάρχουν παρόμοια επίπεδα λεπτομέρειας και εμβέλειας σε άλλους τύπους διαγραμμάτων, όπως τα διαγράμματα ροής δεδομένων, αλλά αυτό έρχεται σε αντίθεση με την προσέγγιση τριών σχημάτων της μηχανικής λογισμικού, η οποία με τη σειρά της διαχωρίζει τις πληροφορίες με άλλο τρόπο.

Οι μηχανικοί μερικές φορές διακλαδίζουν διαγράμματα ER με πρόσθετες ιεραρχίες για να προσθέσουν επίπεδα πληροφοριών που απαιτούνται για τη δημιουργία της βάσης δεδομένων.

Για παράδειγμα, μπορούν να προσθέσουν ομαδοποιήσεις που εκτείνονται προς τα πάνω με υπερκλάσεις και προς τα κάτω με υποκλάσεις.

Περιορισμοί διαγραμμάτων και μοντέλων ER

Μόνο για σχεσιακά δεδομένα: κατανοήστε ότι ο σκοπός είναι να δείξετε τις σχέσεις. Τα διαγράμματα ER δείχνουν μόνο αυτή τη σχεσιακή δομή.

Όχι για μη δομημένα δεδομένα: εκτός εάν τα δεδομένα οριοθετούνται σαφώς σε διαφορετικά πεδία, σειρές ή στήλες, τα διαγράμματα ER είναι πιθανό να έχουν περιορισμένη χρήση. Το ίδιο ισχύει και για τα ημιδομημένα δεδομένα, καθώς μόνο μερικά από τα δεδομένα θα είναι χρήσιμα.

Δυσκολία κατά την ενοποίηση με μια υπάρχουσα βάση δεδομένων: Λόγω διαφορετικών αρχιτεκτονικών, η χρήση μοντέλων ER για την ενοποίηση με μια βάση δεδομένων μπορεί να είναι δύσκολη.

Πώς να σχεδιάσετε ένα βασικό διάγραμμα ER

Σκοπός και πεδίο εφαρμογής: Καθορίστε το σκοπό και το εύρος αυτού που αναλύετε ή διαμορφώνετε.

Οντότητες: Προσδιορίστε τις οντότητες που εμπλέκονται. Όταν είστε έτοιμοι, αρχίστε να τα σχεδιάζετε σε ορθογώνια (ή την προτίμηση σχήματος του συστήματός σας) και να τα χαρακτηρίζετε ως ουσιαστικά.

Σχέσεις: Προσδιορίστε πώς σχετίζονται όλες οι οντότητες. Σχεδιάστε γραμμές ανάμεσά τους για να δείξετε τις σχέσεις και να τις χαρακτηρίσετε. Ορισμένες οντότητες ενδέχεται να μην σχετίζονται και αυτό δεν αποτελεί πρόβλημα. Σε διαφορετικά συστήματα σημειογραφίας, η σχέση μπορεί να επισημανθεί σε ένα διαμάντι, ένα άλλο ορθογώνιο ή απευθείας στην κορυφή της γραμμής σύνδεσης.

Χαρακτηριστικά: Προσθέστε περισσότερα επίπεδα λεπτομέρειας προσθέτοντας βασικά χαρακτηριστικά οντοτήτων. Τα χαρακτηριστικά εμφανίζονται συχνά ως οβάλ.

Περισσότερες συμβουλές για διαγράμματα ER

Δείξτε το επίπεδο λεπτομέρειας που απαιτείται για το σκοπό σας. Μπορείτε να σχεδιάσετε ένα εννοιολογικό, λογικό ή φυσικό μοντέλο, ανάλογα με τις λεπτομέρειες που απαιτούνται. (Δείτε παραπάνω για περιγραφές αυτών των επιπέδων.)

Προσέξτε για περιττές οντότητες ή σχέσεις.

Εάν αντιμετωπίζετε προβλήματα με ένα πρόβλημα βάσης δεδομένων, να είστε σε επιφυλακή για τρύπες στις σχέσεις ή λείπουν οντότητες ή χαρακτηριστικά.

Βεβαιωθείτε ότι όλες οι οντότητές σας και οι σχέσεις τους έχουν επισημανθεί.

Μπορείτε να μεταφράζετε σχεσιακούς πίνακες και διαγράμματα ER ξανά και ξανά, εάν σας βοηθά να πετύχετε τον στόχο σας.

Βεβαιωθείτε ότι το διάγραμμα ER υποστηρίζει όλα τα δεδομένα που χρειάζεστε για αποθήκευση.

Μπορεί να υπάρχουν διαφορετικές προσεγγίσεις σε ένα διάγραμμα ER. Εφόσον παρέχει τις απαραίτητες πληροφορίες για το εύρος και το σκοπό του, είναι έγκυρο.

Μοντέλο οντοτήτων και σχέσεων

Το μοντέλο οντοτήτων-σχέσεων (σχεσιακό μοντέλο) ορίστηκε από τον Peter Chen το 1976 και βασίστηκε στη σχεσιακή θεωρία που δημιουργήθηκε από τον EFCood (1970). Σύμφωνα με τον Chen, το όραμα μιας δεδομένης πραγματικότητας βασίζεται στη σχέση μεταξύ των εννοιών αυτής της πραγματικότητας, οι οποίες απεικονίζουν τα γεγονότα που διέπουν την ίδια πραγματικότητα και ότι κάθε έννοια (οντότητα ή σχέση) μπορεί να έχει ιδιότητες (προσόντα αυτής της πραγματικότητας).

Η έννοια της αφαίρεσης επιτρέπει στον αναλυτή να διαχωρίσει από την υπό μελέτη πραγματικότητα τα μέρη που είναι πραγματικά σχετικά με την ανάπτυξη του πληροφοριακού συστήματος και να αποκλείσει από τη μοντελοποίηση όλες τις πτυχές που δεν επηρεάζουν το περιβάλλον που πρόκειται να μοντελοποιηθεί.

Ο σκοπός της μοντελοποίησης δεδομένων είναι να επιτρέψει την παρουσίαση μιας ενιαίας, μη περιττής, συνοπτικής προβολής των δεδομένων ενός προβλήματος. Μας βοηθά επίσης να κατανοήσουμε τη δομή και το νόημα αυτών των δεδομένων.

Στην ανάπτυξη εφαρμογών βάσης δεδομένων, το μοντέλο σχέσης οντοτήτων (ER) είναι το πιο ευρέως χρησιμοποιούμενο για την αναπαράσταση και την κατανόηση των δεδομένων που συνθέτουν την ουσία ενός προβλήματος.

Η μοντελοποίηση δεδομένων χρησιμοποιείται ευρέως ως μέσο κατανόησης οργανωτικών προβλημάτων και σχεδιασμού λύσεων.

ΕΝΝΟΙΟΛΟΓΙΚΟ ΜΟΝΤΕΛΟ

Κατά τη χρήση της Εννοιολογικής Μοντελοποίησης δεδομένων με την τεχνική Οντότητα και Σχέσεις, θα λάβουμε αποτελέσματα και καθαρά εννοιολογικά σχήματα σχετικά με την ουσία ενός προβλήματος ή επιχείρησης για την οποία αναπτύσσουμε ένα έργο, χωρίς να αντιπροσωπεύουμε υπάρχουσες διαδικασίες ή ροή δεδομένων.

Ο Τσεν αφιερώθηκε στην ανάδειξη της σημασίας της αναγνώρισης των αντικειμένων που συνθέτουν ένα πρόβλημα ή μια επιχείρηση, ανεξάρτητα από το αν ασχολείται με τρόπους χειρισμού πληροφοριών, διαδικασιών, προγραμμάτων κ.λπ.

Αυτά τα αντικείμενα που θέλουμε να γνωρίζουμε και να μοντελοποιούμε για ένα σύστημα, ο Chen τα ταξινόμησε σε δύο ομάδες: Οντότητες και Σχέσεις.

Η μοντελοποίηση γίνεται σε 3 φάσεις:

1. Καταχωρίστε τις Απαιτήσεις Πληροφοριών – συλλογή πληροφοριών από τον πελάτη.
2. Κατασκευάστε μοντέλα για κάθε στοιχείο
3. Επιλογή της τεχνολογίας που θα χρησιμοποιηθεί – μετατροπή μοντέλων σε συστήματα

Με άλλα λόγια, η Εννοιολογική Μοντελοποίηση Δεδομένων χαρακτηρίζεται από το ότι είναι Ανεξάρτητη από τον τύπο του ΣΔΒΔ και των Προγραμμάτων που πρόκειται να αναπτυχθούν. Δηλαδή, σε αυτό το στάδιο, δεν ανησυχούμε πολύ για το πώς θα υλοποιηθούν τα προγράμματα, οι υπηρεσίες web, τα στοιχεία ή οι σελίδες του συστήματος.

Το μοντέλο σχέσεων οντοτήτων προτείνει η πραγματικότητα να εξεταστεί από τρεις οπτικές γωνίες:

- 1 – Τα αντικείμενα που συνθέτουν την πραγματικότητα

2 - τα είδη των πληροφοριών ή των χαρακτηριστικών που θέλετε να μάθετε για τα αντικείμενα που το αποτελούν πραγματικότητα και

3 - ο τρόπος που αυτά τα αντικείμενα σχετίζονται μεταξύ τους.

Έτσι, το Μοντέλο Σχέσεων Οντότητας αποτελείται από τρεις έννοιες: Οντότητα, Ιδιότητα και Σχέση. Για να αναπαραστήσουμε αυτές τις έννοιες, υπάρχει το Διάγραμμα Σχέσεων Οντοτήτων.

ΔΙΑΓΡΑΜΜΑ ΣΧΕΣΗΣ ΟΝΤΟΤΗΤΩΝ

- Η πιο διαδεδομένη τεχνική εννοιολογικής μοντελοποίησης

– Προσέγγιση σχέσης οντοτήτων (ER).

Είναι ένα διάγραμμα που περιγράφει τις οντότητες και τις σχέσεις μεταξύ οντοτήτων σε ένα μοντέλο δεδομένων.

- Το εννοιολογικό μοντέλο αναπαρίσταται μέσω ενός διαγράμματος σχέσης οντότητας

Οντότητα:

Ως οντότητα ορίζεται ένα αντικείμενο που υπάρχει στον πραγματικό κόσμο με διακριτή ταύτιση και νόημα.

- Εκπροσώπηση οντότητας.

Ιδιότητες οντότητας

- Η οντότητα από μόνη της δεν αναφέρει τίποτα
- Είναι απαραίτητο να εκχωρήσετε ιδιότητες σε οντότητες
- Ιδιότητες που προσδιορίζονται με τη μορφή: σχέσεων, ιδιοτήτων και γενικεύσεων/εξειδικεύσεων

ΧΑΡΑΚΤΗΡΙΣΤΙΚΟ

Είναι ένα χαρακτηριστικό ή ποιότητα μιας οντότητας που έχει αξία για την επιχείρηση, η οποία πρέπει να τηρείται από τον χρήστη.

Παράδειγμα: Στο αρχείο πελατών μιας εταιρείας, μπορούμε να βρούμε τα ακόλουθα χαρακτηριστικά πελατών:

crf, όνομα, διεύθυνση, πόλη, γειτονιά, πολιτεία, ταχυδρομικός κώδικας.

Ο καθορισμός των χαρακτηριστικών γίνεται συνήθως κατά τη διάρκεια πολλών συζητήσεων με τον πελάτη. Κάθε χαρακτηριστικό που μπορεί να είναι απαραίτητο σε κάποια λήψη αποφάσεων, στο παρόν ή στο μέλλον, πρέπει να περιλαμβάνεται.

Προσοχή όμως...αποφύγετε την αποθήκευση περιττών δεδομένων .

Τα χαρακτηριστικά μπορούν να ταξινομηθούν με διάφορους τρόπους ανάλογα με τον τύπο των πληροφοριών που αντιπροσωπεύουν, την υποχρεωτική συμπλήρωση χαρακτηριστικών και τη σύνθεση χαρακτηριστικών.

Μονοτιμημένο Χαρακτηριστικό : υποθέτει μια ενιαία τιμή για κάθε στοιχείο του συνόλου x οντότητα

Πρώην. : Ονομα

Σύνθετο χαρακτηριστικό : σχηματίζεται από ένα ή περισσότερα υπο-χαρακτηριστικά. Όταν βρισκόμαστε αντιμέτωποι με μια τέτοια κατάσταση, πρέπει να αναλύσουμε εάν πρέπει να κατακερματίσουμε αυτά τα χαρακτηριστικά σε υπο-χαρακτηριστικά.

Πρώην. : Διεύθυνση (σύνθεση οδού, αριθμού, συμπλήρωμα, γειτονιά, ταχυδρομικός κώδικας)

Multivalued Attribute : μια μεμονωμένη οντότητα έχει πολλές τιμές για αυτό το χαρακτηριστικό (το όνομά της αναπαρίσταται πάντα στον πληθυντικό)

Πρώην. : Εξαρτήματα

Προσδιορισμός Χαρακτηριστικού : προσδιορίζει κάθε οντότητα ενός συνόλου οντοτήτων (γνωστό και ως χαρακτηριστικό κλειδιού)

Πρώην. : Cod_Func

Τομέας ενός χαρακτηριστικού : σύνολο επιτρεπόμενων τιμών για το χαρακτηριστικό.

Πρώην. : Φύλο {M, F}, Οικογενειακή Κατάσταση (C, S, D)

Υποχρεωτική και Προαιρετική Ιδιότητα

Ορισμένα χαρακτηριστικά μιας οντότητας είναι υποχρεωτικά, άλλα είναι προαιρετικά.

Το όνομα είναι υποχρεωτικό καθώς κάθε άτομο πρέπει να έχει όνομα.

Το τηλέφωνο είναι προαιρετικό, καθώς δεν έχουν όλοι τηλέφωνο.

Για να καταλήξουμε σε αυτό το συμπέρασμα, πρέπει να κάνουμε μια λεπτομερή ανάλυση.

Από πεδίο σε πεδίο της(των) οντότητας(ών) που δημιουργούμε.

Στη συνέχεια μπορούμε να κάνουμε τις παρακάτω ρυθμίσεις:

Υποχρεωτικό χαρακτηριστικό - είναι αυτό που για παράδειγμα μιας οντότητας ή μιας σχέσης πρέπει να έχει μια τιμή. (OXI KENO)

Προαιρετικό χαρακτηριστικό - Είναι ένα χαρακτηριστικό που για παράδειγμα της οντότητας ή της σχέσης μπορεί να έχει μια τιμή. Αυτό είναι αυτό που ονομάζουμε μηδενική τιμή. (ΜΗΔΕΝΙΚΟ)

Πώς να επιλέξετε το όνομα, τον τύπο δεδομένων και το μέγεθος ενός χαρακτηριστικού;

Τα ονόματα των χαρακτηριστικών πρέπει πάντα να έχουν νόημα και να περιγράφουν με σαφήνεια τον τύπο των πληροφοριών που αντιπροσωπεύει το χαρακτηριστικό.

Ο τύπος δεδομένων ενός χαρακτηριστικού εξαρτάται από τις πληροφορίες που θέλουν να αποθηκευτούν στο χαρακτηριστικό. Ο τύπος δεδομένων εάν αποθηκεύονται δεδομένα που δεν θα αποτελούν μέρος των υπολογισμών, ένα όνομα για παράδειγμα, ο τύπος μπορεί να είναι κείμενο, εάν αποθηκεύονται αριθμητικές τιμές που θα αποτελούν μέρος των υπολογισμών, ο τύπος θα είναι αριθμητικός (ακέραιος, απλός...) , εάν αποθηκεύονται ημερομηνίες, ο τύπος θα είναι ημερομηνία και ούτω καθεξής.

Το χαρακτηριστικό μέγεθος καθορίζει τον όγκο των πληροφοριών που μπορείτε να αποθηκεύσετε, ως εξής: για ένα πεδίο κειμένου με μέγεθος 30 μπορείτε να αποθηκεύσετε κείμενα με μέγιστο αριθμό 30 χαρακτήρων, ένα αριθμητικό πεδίο μεγέθους 2 αποθηκεύει αριθμούς με έως και δύο ψηφία.

Τύποι δεδομένων:

Αριθμός

Smallint - Υπογεγραμμένη ακέραια τιμή 15-bit

ακέραιος - Υπογεγραμμένη ακέραια τιμή 31-bit

float(p) - επιστημονική μορφή με ακρίβεια

δεκαδικό(p,q) - δεκαδική μορφή

Κείμενο

char(n) - συμβολοσειρά σταθερού μήκους n χαρακτήρων

varchar(n) - συμβολοσειρά μεταβλητού μήκους n χαρακτήρων

μακρύ varchar - χορδή μεταβλητού μήκους

Ημερομηνία ώρα

Σου δίνει

ομάδα

χρονική σφραγίδα

Συμβουλές για τον προσδιορισμό Οντοτήτων, Σχέσεων και Ιδιοτήτων.

Δίνεται ένα κείμενο που περιγράφει τη βάση δεδομένων που πρόκειται να σχεδιαστεί:

Η παρουσία ενός ουσιαστικού συνήθως υποδηλώνει μια οντότητα,

Η παρουσία ενός ρήματος είναι μια ισχυρή ένδειξη μιας σχέσης,

Ένα επίθετο, που είναι μια ποιότητα, είναι μια ισχυρή ένδειξη μιας ιδιότητας,

Ένα χρονικό επίρρημα, που χαρακτηρίζει το ρήμα, είναι ένδειξη μιας ιδιότητας της σχέσης.

Εκπαιδευτική υποενότητα : 5.4 Συστήματα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ)

Τι είναι ένα σύστημα διαχείρισης βάσεων δεδομένων DBMS και ποια πλεονεκτήματα προσφέρει;

Το DBMS είναι υπεύθυνο για τα πάντα, αποθήκευση των δεδομένων στο HD, διατήρηση των δεδομένων με τη μεγαλύτερη πρόσβαση στη μνήμη , σύνδεση δεδομένων και μεταδεδομένων, παροχή διεπαφής για προγράμματα και εξωτερικούς χρήστες για πρόσβαση στη βάση δεδομένων (για σχεσιακές βάσεις δεδομένων , τη γλώσσα SQL), κρυπτογράφηση δεδομένα , έλεγχος πρόσβασης σε...

Πώς λειτουργεί ένας διαχειριστής βάσης δεδομένων;

Τι είναι το SGDB ; Πώς λειτουργεί ; ... Το Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS) ή Σύστημα Διαχείρισης Βάσεων Δεδομένων (DBMS) και το λογισμικό που χρησιμοποιείται για τη διαχείριση βάσεων δεδομένων, που σας επιτρέπει να δημιουργείτε βάσεις δεδομένων, να τροποποιείτε βάσεις δεδομένων, να διαγράψετε βάσεις δεδομένων, να εισάγετε δεδομένα στη βάση δεδομένων και να διαγράψετε δεδομένα από τη βάση δεδομένων.

Τι είναι ένα DBMS και ποια είναι η λειτουργία του;

Το DBMS είναι το ακρωνύμιο για το Σύστημα Διαχείρισης Βάσεων Δεδομένων. Μέσω αυτού του πόρου, που λειτουργεί με τη μορφή λογισμικού, η εταιρεία σας είναι σε θέση να εκτελεί μια σειρά από εργασίες και δραστηριότητες, λαμβάνοντας τα δεδομένα σας ως αναφορά.

Ποια είναι τα κύρια οφέλη από τη χρήση μιας βάσης δεδομένων DBMS);

Πλεονεκτήματα ενός DBMS

- Γρήγορος χειρισμός και πρόσβαση σε πληροφορίες.
- Κοινή χρήση δεδομένων .
- Μείωση της Ανθρώπινης Προσπάθειας.

- Μεγαλύτερος έλεγχος δεδομένων .
- Αποφύγετε προβλήματα ακεραιότητας δεδομένων .
- Μεγαλύτερη Διαθεσιμότητα Πληροφοριών.
- Μεγαλύτερη Ασφάλεια.
- Αποφύγετε τον πλεονασμό και την ασυνέπεια δεδομένων .

Πώς να διαχειριστείτε μια βάση δεδομένων;

5 συμβουλές για τη διαχείριση μιας βάσης δεδομένων πελατών

- 1- Διατηρήστε τις πληροφορίες ενημερωμένες.
- 2- Μάθετε ποιες πληροφορίες χρειάζονται για την επιχείρησή σας.
- 3- Η βάση δεδομένων είναι χρυσή.
- 4- Αποτελεσματικό σύστημα διαχείρισης.
- 5- Κατανοήστε την ανάγκη του πελάτη.

Τι είναι τα συστήματα διαχείρισης δεδομένων;

Τα συστήματα διαχείρισης δεδομένων είναι χτισμένα σε πλατφόρμες διαχείρισης δεδομένων και μπορούν να περιλαμβάνουν βάσεις δεδομένων, λίμνες και αποθήκες δεδομένων, συστήματα διαχείρισης μεγάλων δεδομένων, προηγμένες αναλύσεις δεδομένων και άλλα.

Τι είναι η διαχείριση μιας βάσης δεδομένων;

Η δημιουργία αντιγράφων ασφαλείας είναι ένας τρόπος για να διασφαλιστεί ότι οι πληροφορίες δεν θα χαθούν.

Τέλος πάντων, μπορείτε ήδη να δείτε ότι η διαχείριση μιας βάσης δεδομένων δεν είναι κάτι που πρέπει να αφηθεί στην άκρη, γιατί μια εταιρεία μπορεί να εξαρτάται από αυτήν, δηλαδή μπορεί να βοηθήσει την εταιρεία να πετύχει, αλλά μπορεί επίσης να την οδηγήσει σε αποτυχία.

Τι είναι τα συστήματα διαχείρισης δεδομένων;

Για τη διασφάλιση της συνέπειας των δεδομένων, τον έλεγχο της πρόσβασης, τη διατήρηση των δεδομένων ασφαλή, την παροχή μέσων πρόσβασης σε δεδομένα, ... δημιουργήθηκαν τα Συστήματα Διαχείρισης Βάσεων Δεδομένων ή DBMS (DBMS στα Αγγλικά Σύστημα Διαχείρισης Βάσεων Δεδομένων).

Τι είναι το σύστημα βάσης δεδομένων ανοιχτού κώδικα;

Ένα σύστημα βάσης δεδομένων ανοιχτού κώδικα είναι αυτό του οποίου ο πηγαίος κώδικας είναι ανοιχτού κώδικα.

Αυτές οι βάσεις δεδομένων μπορεί να είναι βάσεις δεδομένων SQL ή NoSQL. Βάσεις δεδομένων cloud.

DBMS: ΤΙ ΕΙΝΑΙ, ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ ΚΑΙ ΚΥΡΙΑ ΠΛΕΟΝΕΚΤΗΜΑΤΑ

Ξέρετε τι είναι το DBMS; Μπορούμε ήδη να πούμε ότι το ακρωνύμιο έχει να κάνει με τη συλλογή και τη χρήση πληροφοριών που πληρούν τις προϋποθέσεις για τη λήψη αποφάσεων σε εταιρείες.

Στο πλαίσιο των επιχειρήσεων που εργάζονται με την τεχνολογία ή που εξαρτώνται από αυτήν, το Σύστημα Διαχείρισης Βάσεων Δεδομένων αναδεικνύεται ως ένας από τους σημαντικότερους πόρους.

Δείτε σε αυτό το περιεχόμενο πώς ωφελείται η εταιρεία σας όταν διαθέτει ένα σύστημα ελέγχου δεδομένων στις ρουτίνες της.

ΤΙ ΕΙΝΑΙ ΤΟ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ (DBMS);

Το DBMS είναι το ακρωνύμιο για το Σύστημα Διαχείρισης Βάσεων Δεδομένων .

Μέσω αυτού του πόρου, που λειτουργεί με τη μορφή λογισμικού, η εταιρεία σας είναι σε θέση να εκτελεί μια σειρά από εργασίες και δραστηριότητες, λαμβάνοντας τα δεδομένα σας ως αναφορά.

Είναι, όπως υποδηλώνει το όνομα, ένα σύστημα υποστήριξης διαχείρισης δεδομένων, κάτι που εμφανίζεται όλο και περισσότερο σε εταιρείες όλων των μεγεθών και τμημάτων.

ΔΗΜΙΟΥΡΓΗΣΤΕ ΣΧΕΣΕΙΣ ΜΕΤΑΞΥ ΠΙΝΑΚΩΝ

Μέσω ενός κατάλληλου συστήματος διαχείρισης, είναι δυνατό να δημιουργηθούν σχέσεις μεταξύ των πινάκων και, κατά συνέπεια, να ληφθούν σχετικές πληροφορίες ή που καθοδηγούν μια απόφαση.

Θα ήταν, για παράδειγμα, η περίπτωση ενός καταστήματος που ήθελε να συγκρίνει τα αποτελέσματα πωλήσεων μεταξύ δύο διαδοχικών μηνών από τα εμπορικά αρχεία στις βάσεις δεδομένων του.

ΔΙΑΓΡΑΦΗ ΚΑΙ ΑΝΤΙΓΡΑΦΗ ΑΡΧΕΙΩΝ

Όσοι ασχολούνται με δεδομένα πρέπει συνεχώς να τα ενημερώνουν .

Με αυτόν τον τρόπο, το DBMS γίνεται η λύση όταν είναι απαραίτητο να διαγραφούν διπλότυπα αρχεία ή αρχεία που δεν εξυπηρετούν πλέον τους αρχικούς τους σκοπούς.

ΑΛΛΑΞΤΕ ΤΗ ΔΟΜΗ ΤΟΥ ΠΕΔΙΟΥ

Τα έντυπα υπόκεινται επίσης σε αλλαγές στα πεδία τους.

Ανάλογα με το πλαίσιο, μπορεί να είναι καιρός να συμπεριλάβετε ή να εξαιρέσετε ορισμένες από τις γραμμές σας.

ΕΚΤΕΛΕΣΤΕ ΕΡΩΤΗΜΑΤΑ ΣΕ ΠΙΝΑΚΕΣ

Όταν μια αναζήτηση πραγματοποιείται σε ένα λογισμικό, εφαρμογή ή πλατφόρμα, το αρχείο ή το αναμενόμενο αποτέλεσμα βρίσκεται μέσω του συστήματος διαχείρισης δεδομένων.

Με άλλα λόγια, είναι σαν το DBMS να ήταν ένα είδος «mini Google» στην υπηρεσία ανθρώπων και εταιρειών.

ΔΗΜΙΟΥΡΓΙΑ ΚΑΙ ΔΙΑΧΕΙΡΙΣΗ ΧΡΗΣΤΩΝ

Φανταστείτε ότι η ομάδα των συνεργατών σας έχει μεγαλώσει και τώρα έχει δύο νέα ταλέντα.

Μέσω του ΣΔΒΔ θα μπορείτε να τους καταχωρείτε ως χρήστες στο ERP ή στο λογισμικό σας για καθημερινή χρήση.

ΕΙΣΑΓΩΓΗ ΚΑΙ ΕΞΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ

Μια άλλη λειτουργία που εξαρτάται από το DBMS είναι η εισαγωγή και εξαγωγή δεδομένων από αρχεία όπως υπολογιστικά φύλλα και έγγραφα.

Ο τρόπος για να γίνει αυτό ποικίλλει, αλλά σε όλες τις περιπτώσεις, θα χρειαστεί ένα κατάλληλο σύστημα για αυτόν τον τύπο λειτουργίας.

ΠΟΙΟΙ ΕΙΝΑΙ ΟΙ ΤΥΠΟΙ DBMS;

Ένα σύστημα διαχείρισης βάσεων δεδομένων μπορεί να μορφοποιηθεί με διαφορετικούς τρόπους, ανάλογα με τους στόχους και τα δεδομένα που πρόκειται να χειριστούν.

Παρακάτω επισημαίνουμε πέντε από αυτά:

Relational (SQL): η λειτουργία του οποίου είναι να συνδέει δεδομένα από διαφορετικές πηγές

Μη σχεσιακή (NoSQL): η διαφορά από τις βάσεις δεδομένων SQL είναι ότι η δομή δεδομένων εδώ δεν χρειάζεται να καθοριστεί εκ των προτέρων

Ιεραρχική: παλαιότερη μορφή, στην οποία τα δεδομένα είναι οργανωμένα σε μια πυραμιδική διάταξη, όπου τα δεδομένα "παιδιού" συνδέονται με δεδομένα "γονέα"

Δίκτυο: σε αυτό, κάθε «παιδί» μπορεί να έχει περισσότερους από έναν «γονείς» και η δομή δεδομένων μοιάζει περισσότερο με ιστό αράχνης παρά με πυραμίδα

Αντικειμενοστρεφές: πιο προηγμένο μοντέλο στο οποίο συγχωνεύονται διαφορετικές μορφές δεδομένων και επομένως πιο ακριβό στην εφαρμογή.

ΔΕΙΤΕ ΤΑ 4 ΒΑΣΙΚΑ ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΟΥ DBMS

Οι λειτουργίες του DBMS ήδη μας δίνουν ενδείξεις ότι είναι απαραίτητο και, ως εκ τούτου, η χρήση του φέρνει μαζί του αρκετά λειτουργικά και στρατηγικά πλεονεκτήματα.

Δηλαδή, για εταιρείες που είναι χρήστες λογισμικού, ανεξάρτητα από το σκοπό, η ύπαρξη ενός επαρκούς συστήματος για τη διαχείριση δεδομένων είναι απαραίτητη .

Δείτε τα οφέλη που προσθέτετε βασιζόμενοι σε αυτήν τη λύση παρακάτω.

1. ΛΙΓΟΤΕΡΗ ΑΝΘΡΩΠΙΝΗ ΠΡΟΣΠΑΘΕΙΑ

Ίσως ο κύριος σκοπός ενός συστήματος ελέγχου δεδομένων είναι να θέσει σε λειτουργία εργασίες που είναι πολύ απαιτητικές για να γίνουν χωρίς αυτοματισμό .

Θα ήταν σαν να αντικαθιστούσαμε μια μη αυτόματη αναζήτηση με ένα βιβλίο σε μια βιβλιοθήκη, δίνοντας τη θέση σας σε έναν αλγόριθμο της Google που μπορεί να εντοπίσει ένα αρχείο σε χιλιοστά του δευτερολέπτου.

2. ΠΡΟΣΑΡΜΟΓΗ

Υπάρχουν λογισμικά αυτού του τύπου που, επειδή είναι ανοιχτού κώδικα, επιτρέπουν την προσαρμογή των λειτουργιών τους, καλύπτοντας πιο συγκεκριμένες ανάγκες .

Επομένως, αυτό είναι ένα ακόμη πλεονέκτημα ενός ΣΔΒΔ που μπορεί να γίνει υπέρ σας.

3. ΑΠΟΦΥΓΕΤΕ ΤΙΣ ΑΠΟΛΥΣΕΙΣ

Όταν ασχολούμαστε με δεδομένα, είναι φυσιολογικό να υπάρχει ο λεγόμενος πλεονασμός, όταν καταλήγουν να αποθηκεύονται πανομοιότυπα (ή πολύ παρόμοια) δεδομένα.

Με ένα DBMS, είναι δυνατός ο εντοπισμός τους για τον καθαρισμό και την προσαρμογή της βάσης δεδομένων.

4. ΕΓΚΑΤΑΣΤΑΣΕΙΣ ΚΟΙΝΗΣ ΧΡΗΣΗΣ ΔΕΔΟΜΕΝΩΝ

Σε μικρές εταιρείες, η κοινή χρήση πληροφοριών δεν αποτελεί πρόβλημα, καθώς υπάρχουν λίγοι χρήστες στα συστήματά τους.

Αλλά όταν πρόκειται για μεγαλύτερους οργανισμούς με 100 υπαλλήλους ή περισσότερους, το ζήτημα απαιτεί πιο κατάλληλο χειρισμό.

Σε αυτήν την περίπτωση, ένα DBMS είναι ο πόρος που θα διασφαλίσει ότι όλοι οι άνθρωποι, ανεξάρτητα από το πόσοι, έχουν πρόσβαση στο σύστημα.

Έτσι μπορούν να τα μοιραστούν σύμφωνα με τα αντίστοιχα διαπιστευτήριά τους.

Ο ΡΟΛΟΣ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΣΤΗΝ ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

Για τη δημιουργία αποτελεσμάτων, τα δεδομένα πρέπει να συλλέγονται μαζικά, σύμφωνα με την αρχή των Μεγάλων Δεδομένων.

Από την άλλη, δεν αρκεί απλώς να τα συσσωρεύουμε, αλλά κυρίως να τα αντιμετωπίζουμε ώστε να δίνουν αποτελεσματικές απαντήσεις.

Αυτό είναι ένα έργο που μπορεί να γίνει μόνο με τις κατάλληλες τεχνολογίες, μεταξύ των οποίων το DBMS είναι ένα από τα κύρια.

Τα εργαλεία που χρησιμοποιούνται για την ανάλυση και την επεξεργασία δεδομένων υπάρχουν σε μεγάλο αριθμό στην αγορά.

Και αν δεν αισθάνεστε σίγουροι για τη λήψη αυτής της απόφασης ή για την εφαρμογή της δυνατότητας στο έργο σας, μπορείτε να βασιστείτε στην υπηρεσία παροχής συμβουλών και εκπαίδευσης - επισκεφτείτε τον ιστότοπο για να μάθετε περισσότερα.

ΠΟΙΟΣ ΧΡΕΙΑΖΕΤΑΙ ΕΝΑ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ;

Στην πραγματικότητα, όχι μόνο οι εταιρείες αλλά γενικά τα άτομα χρειάζονται ένα σύστημα που τους επιτρέπει να ελέγχουν και να εγγυώνται την πρόσβαση στα δεδομένα όταν είναι απαραίτητο.

Υπάρχουν πολλά από αυτά στην αγορά, το καθένα με διαφορετικές λειτουργίες και προσανατολισμένα, σε ορισμένες περιπτώσεις, σε διαφορετικά επιχειρηματικά προφίλ.

Ορισμένα είναι ακόμη και ανοιχτού κώδικα, κάτι που είναι ένα πλεονέκτημα που πρέπει να λαμβάνεται υπόψη όταν χρειάζονται πιο συγκεκριμένες λύσεις.

Ένα σύστημα διαχείρισης βάσης δεδομένων είναι ένα σύνολο προγραμμάτων λογισμικού που επιτρέπει στους χρήστες να δημιουργούν, να επεξεργάζονται, να ενημερώνουν, να αποθηκεύουν και να ανακτούν δεδομένα σε πίνακες βάσεων δεδομένων. Τα δεδομένα σε μια βάση δεδομένων μπορούν να προστεθούν, να διαγραφούν, να αλλάξουν, να ταξινομηθούν χρησιμοποιώντας ένα DBMS. Εάν

ήσασταν υπάλληλος σε έναν μεγάλο οργανισμό, οι πληροφορίες για εσάς πιθανότατα θα αποθηκευτούν σε διαφορετικούς πίνακες που συνδέονται μεταξύ τους. Με τη διασταύρωση αυτών των πινάκων, θα μπορούσε κανείς να αλλάξει τη διεύθυνση ενός ατόμου σε έναν πίνακα και αυτό το άτομο θα αντικατοπτρίζεται αυτόματα σε όλους τους άλλους πίνακες.

Επιθυμητά χαρακτηριστικά σε μια βάση δεδομένων

- Έλεγχος πλεονασμού.
- Κοινή χρήση δεδομένων.
- Έλεγχος πρόσβασης δεδομένων.
- Πολλαπλές διεπαφές.
- Εκπροσώπηση πολύπλοκων ενώσεων.
- Διασφάλιση περιορισμών ακεραιότητας.
- Αποκατάσταση καταστροφών.

Τα DBMS χρησιμοποιούνται συνήθως για τη διαχείριση:

- Λίστες αλληλογραφίας μελών και συνδρομών
- Λογιστικές και λογιστικές πληροφορίες
- Δεδομένα που προέρχονται από επιστημονική έρευνα
- πληροφορίες πελατών
- πληροφορίες αποθέματος
- προσωπικά αρχεία
- πληροφορίες βιβλιοθήκης

Τα πλεονεκτήματα ενός DBMS

Μεγαλύτερη διαθεσιμότητα: Ένα από τα κύρια πλεονεκτήματα ενός DBMS είναι ότι οι ίδιες πληροφορίες μπορούν να διατίθενται σε διαφορετικούς χρήστες, δηλαδή η κοινή χρήση δεδομένων.

Ελαχιστοποιημένος πλεονασμός: Τα δεδομένα σε ένα DBMS είναι πιο συνοπτικά, επειδή, κατά γενικό κανόνα, οι πληροφορίες σε αυτό εμφανίζονται μόνο μία φορά . Αυτό μειώνει τον πλεονασμό δεδομένων, ή με άλλα λόγια, την ανάγκη επανάληψης των ίδιων δεδομένων ξανά και ξανά. Η ελαχιστοποίηση του πλεονασμού μπορεί επομένως να μειώσει σημαντικά το κόστος αποθήκευσης πληροφοριών σε σκληρούς δίσκους και άλλες συσκευές αποθήκευσης.

Ακρίβεια: Τα ακριβή, συνεπή δεδομένα είναι σημάδι ακεραιότητας δεδομένων . Τα DBMS ενισχύουν την ακεραιότητα των δεδομένων, επειδή οι ενημερώσεις και οι αλλαγές δεδομένων πρέπει να γίνονται μόνο σε ένα μέρος . Οι πιθανότητες να κάνετε ένα λάθος είναι υψηλότερες εάν αναγκαστείτε να αλλάξετε τα ίδια δεδομένα σε πολλά διαφορετικά μέρη από ό,τι αν χρειαστεί να κάνετε την αλλαγή μόνο σε ένα μέρος.

Συνοχή προγράμματος και αρχείων: Χρησιμοποιώντας ένα σύστημα διαχείρισης βάσης δεδομένων, οι μορφές πινάκων και τα προγράμματα συστήματος είναι τυποποιημένα . Αυτό διευκολύνει τη διατήρηση των πινάκων δεδομένων, επειδή οι ίδιοι κανόνες και οδηγίες ισχύουν για όλους τους τύπους δεδομένων. Το επίπεδο συνοχής μεταξύ πινάκων και προγραμμάτων διευκολύνει επίσης τη διαχείριση δεδομένων όταν εμπλέκονται πολλοί προγραμματιστές.

Φιλικό προς τον χρήστη: Η πρόσβαση και ο χειρισμός των δεδομένων είναι ευκολότερη με ένα DBMS παρά χωρίς αυτό. Στις περισσότερες περιπτώσεις, τα DBMS μειώνουν επίσης την εξάρτηση των μεμονωμένων χρηστών από ειδικούς υπολογιστών για την κάλυψη των αναγκών τους σε δεδομένα.

Βελτιωμένη ασφάλεια: Όπως αναφέρθηκε προηγουμένως, τα DBMS επιτρέπουν σε πολλούς χρήστες να έχουν πρόσβαση στους ίδιους πόρους δεδομένων. Αυτή η ικανότητα θεωρείται γενικά ως όφελος, αλλά υπάρχουν πιθανοί κίνδυνοι για τον οργανισμό . Ορισμένες πηγές πληροφοριών πρέπει να προστατεύονται ή να προστατεύονται και να προβάλλονται μόνο από επιλεγμένα άτομα. Μέσω της χρήσης κωδικών πρόσβασης , τα συστήματα διαχείρισης βάσεων δεδομένων μπορούν να χρησιμοποιηθούν για τον περιορισμό της πρόσβασης στα δεδομένα μόνο σε εκείνους που πρέπει να τα δουν.

Άλλα: Μειώνεται ο χρόνος ανάπτυξης εφαρμογών , μεγαλύτερη ευελιξία για αλλαγές (ανεξαρτησία δεδομένων) και μεγαλύτερη εξοικονόμηση, ενημερωμένες πληροφορίες, μικρότερος όγκος χαρτιού.

Ποια χαρακτηριστικά διακρίνουν ένα DBMS;

Κατάλογος

- Ένα ΣΔΒΔ διατηρεί όχι μόνο τη Βάση Δεδομένων, αλλά και ορισμό και περιγραφή δομών και περιορισμών (κατάλογος - μεταδεδομένα)
- Η ύπαρξη του καταλόγου επιτρέπει τη χρήση του ίδιου DBMS για διαφορετικές εφαρμογές (ο κατάλογος υποδεικνύει μια φυσική δομή που χρησιμοποιείται)

Ανεξαρτησία δεδομένων

- Ένα DBMS παρέχει στους χρήστες μια αφηρημένη άποψη των δεδομένων , καλύπτοντας μη σχετικές λεπτομέρειες (ο χρήστης-προγραμματιστής δεν χρειάζεται να γνωρίζει πώς αποθηκεύονται φυσικά τα δεδομένα).

Πολλαπλές προβολές δεδομένων

- Κάθε χρήστης μπορεί να απαιτεί διαφορετική προβολή της βάσης δεδομένων

Κοινή χρήση και συναλλαγές

- Έλεγχος ανταγωνισμού

Τα μειονεκτήματα ενός DBMS

Υπάρχουν βασικά δύο βασικά μειονεκτήματα στα DBMS. Το ένα είναι το κόστος και το άλλο ο κίνδυνος για την ασφάλεια των δεδομένων.

Κόστος: Η εφαρμογή ενός συστήματος DBMS μπορεί να είναι δαπανηρή και χρονοβόρα , ειδικά σε μεγάλους οργανισμούς. Οι απαιτήσεις εκπαίδευσης μπορεί να είναι αρκετά επαχθείς.

Ασφάλεια: Ακόμη και με τις διασφαλίσεις, ενδέχεται να είναι δυνατό για ορισμένους μη εξουσιοδοτημένους χρήστες να έχουν πρόσβαση στη βάση δεδομένων. Γενικά, η πρόσβαση στη βάση δεδομένων είναι μια πρόταση όλα ή τίποτα. Μόλις ένας μη εξουσιοδοτημένος χρήστης βρίσκεται στη βάση δεδομένων, έχει πρόσβαση σε όλους τους πίνακες, όχι μόνο σε λίγους. Ανάλογα με τη φύση των δεδομένων που εμπλέκονται, αυτές οι παραβιάσεις ασφαλείας μπορούν επίσης να αποτελέσουν απειλή για το απόρρητο του ατόμου. Πρέπει επίσης να λαμβάνεται τακτικά μέριμνα για τη δημιουργία αντιγράφων ασφαλείας των πινάκων και την αποθήκευση τους λόγω της πιθανότητας πυρκαγιών και σεισμών που θα μπορούσαν να καταστρέψουν το σύστημα.

Κάθε σύμπλεγμα βάσεων δεδομένων έχει ένα σύνολο χρηστών βάσης δεδομένων. Αυτοί οι χρήστες διαφέρουν από τους χρήστες που διαχειρίζεται το λειτουργικό σύστημα όπου εκτελείται ο διακομιστής. Οι χρήστες έχουν στην κατοχή τους αντικείμενα βάσης δεδομένων (π.χ. πίνακες) και μπορούν να εκχωρήσουν δικαιώματα σε αυτά τα αντικείμενα σε άλλους χρήστες, ελέγχοντας έτσι ποιος μπορεί να έχει πρόσβαση σε ποιο αντικείμενο.

Διαχειριστής βάσης δεδομένων (DBA)

Σε ένα περιβάλλον βάσης δεδομένων, ο πρωτεύων πόρος είναι η ίδια η βάση δεδομένων και ο δευτερεύων πόρος είναι το DBMS και το σχετικό λογισμικό. Η διαχείριση αυτών των πόρων είναι ευθύνη του Διαχειριστή της Βάσης Δεδομένων, ο οποίος είναι υπεύθυνος για την εξουσιοδότηση πρόσβασης στη βάση δεδομένων και για τον συντονισμό και την παρακολούθηση της χρήσης της. Με άλλα λόγια, συντονίζει όλες τις δραστηριότητες του συστήματος βάσης δεδομένων. Έχει καλή κατανόηση των πόρων και των αναγκών πληροφοριών της εταιρείας.

Οι λειτουργίες του περιλαμβάνουν:

- ορισμός σχήματος
- Δομή αποθήκευσης και ορισμός πρόσβασης δεδομένων
- Φυσική διάταξη και οργάνωση

- Παραχώρηση πρόσβασης στους χρήστες
- Φροντίστε την ακεραιότητα των δεδομένων
- Λειτουργεί ως σύνδεσμος με τους χρήστες
- Παρακολουθεί την απόδοση και ανταποκρίνεται στις απαιτούμενες αλλαγές
- Δραστηριότητες συντήρησης (Δημιουργία αντιγράφων ασφαλείας)

Σχεδιαστής Βάσεων Δεδομένων

Ο Σχεδιαστής Βάσης Δεδομένων είναι υπεύθυνος για την αναγνώριση των δεδομένων που πρέπει να αποθηκευτούν στη βάση δεδομένων, επιλέγοντας τη σωστή δομή για την αναπαράσταση και την αποθήκευση δεδομένων. Συχνά, οι σχεδιαστές βάσεων δεδομένων ενεργούν ως «προσωπικό» για το DBA, αναλαμβάνοντας άλλες ευθύνες μετά την κατασκευή της βάσης δεδομένων. Είναι επίσης ο ρόλος του σχεδιαστή να αξιολογήσει τις ανάγκες κάθε ομάδας χρηστών για να καθορίσει τις απόψεις που θα χρειαστούν, ενσωματώνοντάς τις, καθιστώντας τη βάση δεδομένων ικανή να καλύψει όλες τις ανάγκες των χρηστών.

Τελικούς χρήστες

Υπάρχουν βασικά τρεις κατηγορίες τελικών χρηστών που είναι οι τελικοί χρήστες της βάσης δεδομένων, κάνοντας ερωτήματα, ενημερώσεις και δημιουργώντας έγγραφα:

- Περιστασιακοί χρήστες : πρόσβαση στη βάση δεδομένων περιστασιακά, αλλά μπορεί να χρειάζονται διαφορετικές πληροφορίες για κάθε πρόσβαση. χρησιμοποιούν εξελιγμένες γλώσσες ερωτημάτων για να προσδιορίσουν τις ανάγκες τους.
- Αρχάριοι ή παραμετρικοί χρήστες: χρησιμοποιήστε προκαθορισμένα τμήματα της βάσης δεδομένων, χρησιμοποιώντας προκαθορισμένα ερωτήματα που έχουν ήδη δοκιμαστεί εξαντλητικά.
- Προηγμένοι χρήστες: είναι χρήστες που είναι εξοικειωμένοι με το DBMS και εκτελούν πολύπλοκα ερωτήματα.

Αναλυτές Συστημάτων και Προγραμματιστές Εφαρμογών

Οι αναλυτές καθορίζουν τις απαιτήσεις του τελικού χρήστη και αναπτύσσουν προδιαγραφές για συναλλαγές που πληρούν αυτές τις απαιτήσεις και οι προγραμματιστές εφαρμόζουν αυτές τις προδιαγραφές ως προγράμματα, δοκιμές, εντοπισμό σφαλμάτων, τεκμηρίωση και διατήρησή τους. Είναι σημαντικό τόσο οι αναλυτές όσο και οι προγραμματιστές να γνωρίζουν τους πόρους που προσφέρει το DBMS.

3. Τύποι DBMS (DBMS)

Τα DBMS διατίθενται σε πολλά σχήματα και μεγέθη. Για μερικές εκατοντάδες δολάρια ή ακόμα και δωρεάν, μπορείτε να αγοράσετε ένα DBMS για τον επιτραπέζιο υπολογιστή σας. Για μεγαλύτερα συστήματα, τα DBMS μπορεί να είναι πολύ πιο ακριβά. Πολλά DBMS βασίζονται σε mainframe και μισθώνονται από οργανισμούς. Τα DBMS αυτής της κλίμακας είναι πολύ εξελιγμένα και θα ήταν εξαιρετικά ακριβό να αναπτυχθούν από την αρχή. Επομένως, είναι φθηνότερο για έναν οργανισμό να νοικιάσει ένα πρόγραμμα παρά να το αναπτύξει. Δεδομένου ότι υπάρχει μια ποικιλία διαθέσιμων DBMS, θα πρέπει να γνωρίζετε ορισμένα από τα βασικά χαρακτηριστικά, καθώς και τα δυνατά και τα αδύνατα σημεία των κύριων τύπων.

Δομικοί τύποι των πιο συχνά χρησιμοποιούμενων συστημάτων διαχείρισης βάσεων δεδομένων:

Σχετικός

Τα συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) υποστηρίζουν το σχεσιακό (= προσανατολισμένο σε πίνακα) μοντέλο δεδομένων. Το σχήμα ενός πίνακα (=σχήμα σχέσης) ορίζεται από το όνομα του πίνακα και έναν σταθερό αριθμό χαρακτηριστικών με σταθερούς τύπους δεδομένων. Μια εγγραφή (= οντότητα) αντιστοιχεί σε μια γραμμή πίνακα και αποτελείται από τις τιμές κάθε χαρακτηριστικού. Μια σχέση, επομένως, αποτελείται από ένα σύνολο ομοιόμορφων εγγραφών.

Τα σχήματα πινάκων δημιουργούνται με κανονικοποίηση στη διαδικασία μοντελοποίησης δεδομένων.

Ορισμένες βασικές λειτουργίες ορίζονται σε σχέσεις:

- κλασικές πράξεις συνόλου (ένωση, τομή και διαφορά)
- Επιλογή (επιλογή ενός υποσυνόλου εγγραφών σύμφωνα με ορισμένα κριτήρια φίλτρου για τιμές χαρακτηριστικών)
- Προβολή (επιλογή ενός υποσυνόλου χαρακτηριστικών/στηλών από τον πίνακα)
- Join: ειδικός συνδυασμός πολλαπλών πινάκων ως συνδυασμός καρτεσιανού προϊόντος με επιλογή και προβολή.

Αυτές οι βασικές λειτουργίες, καθώς και οι λειτουργίες δημιουργίας, τροποποίησης και διαγραφής σχημάτων πίνακα, λειτουργίες ελέγχου συναλλαγών και διαχείριση χρηστών εκτελούνται χρησιμοποιώντας γλώσσες βάσης δεδομένων, ενώ η SQL είναι ένα καθιερωμένο πρότυπο για τέτοιες γλώσσες.

Τα πρώτα σχεσιακά συστήματα διαχείρισης βάσεων δεδομένων εμφανίστηκαν στην αγορά στις αρχές της δεκαετίας του 1980 και από τότε είναι ο πιο συχνά χρησιμοποιούμενος τύπος.

Στις σχεσιακές βάσεις δεδομένων, η σχέση μεταξύ των πινάκων δεδομένων είναι σχεσιακή. Οι σχεσιακές βάσεις δεδομένων συνδέουν δεδομένα σε διαφορετικούς πίνακες χρησιμοποιώντας κοινά στοιχεία δεδομένων ή ένα πεδίο κλειδιού. Τα δεδομένα σε σχεσιακές βάσεις δεδομένων αποθηκεύονται σε διαφορετικούς πίνακες, ο καθένας με ένα βασικό πεδίο που προσδιορίζει κάθε γραμμή ή εγγραφή. Οι σχεσιακές βάσεις δεδομένων είναι πολύ πιο ευέλικτες από τις ίδιες τις ιεραρχικές ή δικτυακές δομές δεδομένων. Στις σχεσιακές βάσεις δεδομένων οι σύνδεσμοι μεταξύ των πινάκων ονομάζονται σχέσεις, οι πλειάδες ορίζουν μια γραμμή ή μια εγγραφή και οι στήλες αναφέρονται ως χαρακτηριστικά ή πεδία.

Οι σχεσιακές βάσεις δεδομένων λειτουργούν με βάση την αρχή ότι κάθε πίνακας έχει ένα βασικό πεδίο που προσδιορίζει μοναδικά κάθε γραμμή, και ότι αυτά τα

βασικά πεδία μπορούν να χρησιμοποιηθούν για τη σύνδεση ενός πίνακα δεδομένων με έναν άλλο.

Με αυτόν τον τρόπο, ένας πίνακας μπορεί να έχει μια σειρά που σχηματίζεται από έναν αριθμό λογαριασμού πελάτη, όπως το πεδίο κλειδιού, μαζί με τη διεύθυνση και τον αριθμό τηλεφώνου.

Ο αριθμός λογαριασμού πελάτη σε αυτόν τον πίνακα μπορεί να συνδέεται με έναν άλλο πίνακα δεδομένων που περιλαμβάνει επίσης τον αριθμό λογαριασμού του πελάτη (ένα βασικό πεδίο), αλλά σε αυτήν την περίπτωση περιέχει πληροφορίες επιστροφής προϊόντος, συμπεριλαμβανομένης μιας παραγγελίας (άλλο βασικό πεδίο).

Αυτό το πεδίο κλειδιού μπορεί να συνδεθεί με έναν άλλο πίνακα που περιέχει αριθμούς στοιχείων και άλλες πληροφορίες προϊόντος, όπως τοποθεσία παραγωγής, χρώμα και άλλα δεδομένα.

Επομένως, χρησιμοποιώντας αυτήν τη βάση δεδομένων, οι πληροφορίες πελατών μπορούν να συνδεθούν με συγκεκριμένες πληροφορίες προϊόντος.

Η σχεσιακή βάση δεδομένων έχει γίνει αρκετά δημοφιλής για δύο βασικούς λόγους. Πρώτον, οι σχεσιακές βάσεις δεδομένων μπορούν να χρησιμοποιηθούν με λίγη ή καθόλου εκπαίδευση .

Δεύτερον, οι καταχωρήσεις της βάσης δεδομένων μπορούν να τροποποιηθούν χωρίς να επαναπροσδιοριστεί ολόκληρη η δομή τους .

Το μειονέκτημα της χρήσης μιας σχεσιακής βάσης δεδομένων είναι ότι η ανάκτηση δεδομένων μπορεί να διαρκέσει περισσότερο από ό,τι εάν χρησιμοποιούνται άλλες μέθοδοι.

Με τα χρόνια, πολλά DBMS έχουν επεκταθεί με μη σχεσιακές έννοιες, όπως τύπους δεδομένων που καθορίζονται από το χρήστη, μη ατομικά χαρακτηριστικά, κληρονομικότητα και ιεραρχίες, γι' αυτό μερικές φορές ονομάζονται αντικειμενοσχεσιακά DBMS.

Βάση δεδομένων προσανατολισμένη στα έγγραφα

Οι χώροι αποθήκευσης εγγράφων, που ονομάζονται επίσης συστήματα βάσεων δεδομένων προσανατολισμένων σε έγγραφα, χαρακτηρίζονται από την οργάνωση δεδομένων χωρίς σχήματα.

Αυτό σημαίνει:

- Οι εγγραφές δεν χρειάζεται να έχουν ομοιόμορφη δομή, δηλαδή διαφορετικές εγγραφές μπορούν να έχουν διαφορετικές στήλες.
- Οι τύποι μεμονωμένων τιμών στηλών μπορεί να είναι διαφορετικοί για κάθε εγγραφή.
- Οι στήλες μπορεί να έχουν περισσότερες από μία τιμές (πίνακες).
- Οι εγγραφές μπορούν να έχουν μια ένθετη δομή.

Τα καταστήματα εγγράφων χρησιμοποιούν συχνά ενσωματωμένες σημειώσεις, οι οποίες μπορούν να υποβληθούν σε επεξεργασία απευθείας σε εφαρμογές, ιδιαίτερα σε JSON .

Τα έγγραφα JSON μπορούν επίσης να αποθηκευτούν ως απλό κείμενο σε καταστήματα βασικών τιμών ή σε συστήματα σχεσιακών βάσεων δεδομένων. Αυτό, ωστόσο, θα απαιτούσε την επεξεργασία των δομών από την πλευρά του πελάτη, η οποία έχει το μειονέκτημα ότι οι δυνατότητες που προσφέρονται από καταστήματα εγγράφων (όπως δευτερεύοντα ευρετήρια) δεν είναι διαθέσιμες.

Αποθήκευση βασικής αξίας

Τα καταστήματα βασικών τιμών είναι πιθανώς η απλούστερη μορφή συστημάτων διαχείρισης βάσεων δεδομένων.

Μπορούν να αποθηκεύουν μόνο ζεύγη κλειδιών και τιμών, καθώς και να ανακτούν τιμές όταν ένα κλειδί είναι γνωστό.

Αυτά τα απλά συστήματα συνήθως δεν είναι κατάλληλα για πολύπλοκες εφαρμογές. Από την άλλη πλευρά, αυτή ακριβώς η απλότητα είναι που κάνει αυτά τα συστήματα ελκυστικά σε ορισμένες περιπτώσεις. Για παράδειγμα, οι αποθηκευτικοί χώροι κλειδιών-τιμών με αποδοτικότητα πόρων εφαρμόζονται

συχνά σε ενσωματωμένα συστήματα ή ως βάσεις δεδομένων υψηλής απόδοσης κατά τη διαδικασία.

Προηγμένες φόρμες

Μια εκτεταμένη μορφή αποθήκευσης κλειδιών-τιμών είναι ικανή να ταξινομεί κλειδιά και επομένως επιτρέπει ερωτήματα εύρους καθώς και τακτική επεξεργασία των κλειδιών.

Πολλά συστήματα παρέχουν πρόσθετες επεκτάσεις, ώστε να μπορούμε να δούμε μια εύλογα απρόσκοπτη μετάβαση σε καταστήματα εγγράφων και μεγάλες στήλες.

Σχεσιακό μοντέλο

Σήμερα τα περισσότερα συστήματα βάσεων δεδομένων είναι σχεσιακού τύπου.

Οι σχεσιακές βάσεις δεδομένων έχουν πολύτιμα χαρακτηριστικά που τις διακρίνουν ως ανώτερες.

Πιθανώς το πιο σημαντικό πράγμα είναι ότι μπορείτε να αλλάξετε τη δομή δεδομένων χωρίς να αλλάξετε τις εφαρμογές.

Ας υποθέσουμε, για παράδειγμα, ότι προσθέτετε μία ή περισσότερες στήλες σε έναν πίνακα.

Δεν χρειάζεται να αλλάξετε καμία εφαρμογή καθώς το σύστημα θα συνεχίσει να επεξεργάζεται.

Φυσικά, εάν αφαιρέσετε μια στήλη που χρησιμοποιεί μια υπάρχουσα εφαρμογή, θα αντιμετωπίσετε προβλήματα.

Σχεσιακά αντικείμενα βάσης δεδομένων

Θα δούμε τώρα μια σύντομη περιγραφή των αντικειμένων που συνθέτουν μια σχεσιακή βάση δεδομένων.

Θα φανούν με περισσότερες λεπτομέρειες καθώς προχωράμε βαθύτερα στην πορεία.

Τραπέζια	Αυτά είναι τα αντικείμενα που περιέχουν τους τύπους δεδομένων και τα πραγματικά δεδομένα
Στήλες ή Πεδία	Αυτά είναι τα μέρη των πινάκων που αποθηκεύουν τα δεδομένα. Πρέπει να λάβουν έναν τύπο δεδομένων και να έχουν ένα μοναδικό όνομα
Τύπους δεδομένων	Υπάρχουν διάφοροι τύποι δεδομένων που πρέπει να χρησιμοποιηθούν όπως: χαρακτήρας, αριθμός, ημερομηνία. Ένας μεμονωμένος τύπος δεδομένων εκχωρείται σε μια στήλη μέσα σε έναν πίνακα
Αποθηκευμένες διαδικασίες	Είναι σαν μακροεντολές όπου ο κώδικας Transact-SQL μπορεί να γραφτεί και να αποθηκευτεί με ένα όνομα.

Ενεργοποιητές	Είναι σαν αποθηκευμένες διαδικασίες που ενεργοποιούνται αυτόματα όταν εισάγονται, αλλάζουν ή διαγράφονται δεδομένα. Διασφαλίζουν ότι οι κανόνες επιχειρηματικότητας και ακεραιότητας επιβάλλονται στη βάση δεδομένων.
Κανόνες	Αντιστοιχίζονται σε στήλες έτσι ώστε τα δεδομένα που εισάγονται πρέπει να συμμορφώνονται με τα καθορισμένα πρότυπα. Για παράδειγμα, μπορείτε να χρησιμοποιήσετε κανόνες για να επιτρέψετε σε ένα πεδίο που θα αποθηκεύει το UF να περιέχει μόνο έγκυρες καταστάσεις.
Κύρια κλειδιά (PK)	Αν και δεν είναι αντικείμενα από μόνα τους, τα κλειδιά είναι απαραίτητα για τις σχεσιακές βάσεις δεδομένων. Προωθεί τη μοναδικότητα των γραμμών, παρέχοντας έναν τρόπο να αναγνωρίζετε μοναδικά κάθε στοιχείο που θέλετε να αποθηκεύσετε.
Ξένα κλειδιά (FK)	Και πάλι, δεν είναι αντικείμενα τα ίδια, τα ξένα κλειδιά είναι στήλες που αναφέρονται στα πρωτεύοντα κλειδιά άλλων πινάκων.
Προεπιλογές	Τα πεδία μπορούν να διαμορφωθούν έτσι ώστε αν δεν εισαχθούν δεδομένα κατά τη διάρκεια μιας λειτουργίας Εισαγωγής, θα χρησιμοποιηθούν οι προεπιλεγμένες τιμές.

Προβολές	Βασικά αποτελούνται από ερωτήματα αποθηκευμένα σε βάσεις δεδομένων που μπορούν να αναφέρονται σε έναν ή πολλούς πίνακες. Μπορείτε να δημιουργήσετε και να αποθηκεύσετε προβολές και να τις χρησιμοποιήσετε στο μέλλον. Συνήθως αποκλείουν ορισμένες στήλες από έναν πίνακα και συνδέουν δύο ή περισσότερους πίνακες μεταξύ τους. Μπορούν επίσης να χρησιμοποιηθούν ως μηχανισμός ασφαλείας.
Ευρετήρια	Μπορεί να βοηθήσει τα δεδομένα, ώστε τα ερωτήματα να εκτελούνται πιο γρήγορα

Μια βάση δεδομένων (DB) είναι, βασικά, μια λογικά συνεκτική συλλογή αλληλένδετων δεδομένων, η οποία αντικατοπτρίζει κάποια πτυχή του πραγματικού κόσμου, έχοντας έναν συγκεκριμένο σκοπό, που επιτρέπει την εξαγωγή πληροφοριών.

Με την τεχνολογία όλο και πιο προηγμένη και παρούσα στην καθημερινή μας ζωή, οι βάσεις δεδομένων γίνονται όλο και πιο σημαντικές στην κοινωνία, όπως στους επιχειρηματικούς, ιατρικούς, οικονομικούς, διαγωνισμούς, μεταξύ άλλων.

Μερικά από τα παραδείγματα της καθημερινής μας αλληλεπίδρασης με τη βάση δεδομένων είναι μεταξύ άλλων οι διαδικτυακές μας αγορές, οι τραπεζικές εφαρμογές, η πρόσβαση στα μαθήματα Strategy Contest.

Ωστόσο, μην νομίζετε ότι οι βάσεις δεδομένων υπάρχουν μόνο σε συστήματα υπολογιστών. Μπορούν επίσης να βρεθούν με άλλους τρόπους, χειρόγραφα για παράδειγμα ως τηλεφωνικός κατάλογος.

Μπορούν επίσης να έχουν τα πιο ποικίλα μεγέθη, απλής ή πολύπλοκης δομής.

Λοιπόν, ένα DBMS είναι μια συλλογή προγραμμάτων που επιτρέπουν στους χρήστες να ορίζουν, να δημιουργούν, να χειρίζονται, να μοιράζονται, να συντηρούν και να ασφαλίζουν μια βάση δεδομένων.

Υπάρχουν πολλά πλεονεκτήματα από τη χρήση ενός DBMS, καθώς μπορεί να αποφευχθούν ασυνέπειες σε ένα DB, να διατηρηθεί η ακεραιότητα των δεδομένων, να αυξηθεί η ασφάλεια των δεδομένων, να επιτρέπεται η χρήση προτύπων, ο έλεγχος των συναλλαγών και η πρόσβαση των χρηστών, μεταξύ άλλων λειτουργιών.

Η ένωση αυτού του συστήματος διαχείρισης με μια βάση δεδομένων είναι γνωστή ως Σύστημα Βάσεων Δεδομένων.

Χαρακτηριστικά

Υπάρχουν ορισμένα σημαντικά χαρακτηριστικά των διαχειριζόμενων βάσεων δεδομένων, όπως μπορούμε να δούμε παρακάτω:

Αυτοπεριγραφή: οι τράπεζες, εκτός από το ότι έχουν τα ίδια τα δεδομένα, διαμορφώνονται και από τους ορισμούς και τις περιγραφές των δομών αυτών των δεδομένων.

Τι θα λέγατε για τις περιγραφές;

Λοιπόν, για παράδειγμα, σε έναν πίνακα σχετικά με τους γιατρούς σε ένα νοσοκομείο, εκτός από τα ανεπεξέργαστα δεδομένα, υπάρχουν ορισμοί της δομής του, όπως ονόματα στηλών, περιορισμοί χαρακτηριστικών (όπως να επιτρέπεται ο μέγιστος αριθμός γραμμάτων σε ένα όνομα να είναι 50), μεταξύ άλλων σημαντικών πληροφοριών από έναν πίνακα.

Υποστήριξη για πολλαπλές προβολές: είναι πιθανό διαφορετικές εφαρμογές να έχουν πρόσβαση στη βάση δεδομένων και να ζητηθούν διαφορετικές πληροφορίες από κάθε μία από αυτές. Έτσι, μπορούν να δημιουργηθούν διαφορετικές προβολές, οι οποίες είναι υποσύνολα της βάσης δεδομένων.

Κοινή χρήση δεδομένων: Ένα DBMS μπορεί να είναι πολλαπλών χρηστών, επιτρέποντας την ταυτόχρονη πρόσβαση στη βάση δεδομένων από πολλούς χρήστες.

Συναλλαγές Βάσεων Δεδομένων

Παιδιά, θα πάμε τώρα σε ένα θέμα που είναι πολύ δημοφιλές σε δοκιμές, συναλλαγές και τις ιδιότητές τους σε ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων.

Τι είναι μια συναλλαγή;

Φανταστείτε χιλιάδες διαγωνιζόμενους να προσπαθούν να αγοράσουν την απεριόριστη συνδρομή στο Strategy Contests την ημέρα της κυκλοφορίας του. Εκατοντάδες αγορές μπορούν να πραγματοποιηθούν ανά λεπτό στον ιστότοπο, με μεγάλο όγκο δεδομένων να εισάγεται σχεδόν ταυτόχρονα στη βάση δεδομένων Strategy. Αλλά πώς να εγγυηθούμε ότι όλες αυτές οι αλλαγές στο DB πραγματοποιούνται σωστά, χωρίς να χάνονται ή να μπερδεύονται δεδομένα από διαφορετικούς χρήστες;

Για αυτό χρησιμοποιούνται συναλλαγές, οι οποίες επιτρέπουν τη σωστή αλλαγή της βάσης δεδομένων, διασφαλίζοντας την ακεραιότητα του DB όταν συμβαίνουν πολλές ταυτόχρονες προσβάσεις.

Έχει 4 σημαντικές ιδιότητες που το βοηθούν στο έργο του, όπως φαίνεται παρακάτω.

Ατομικότητα

Παρόμοια με την ιδέα του αδιαίρετου του ατόμου, οι συναλλαγές πρέπει επίσης να είναι αδιαίρετες, δηλαδή είτε γίνονται όλα είτε δεν γίνεται τίποτα. Δηλαδή, επιστρέφοντας στο παράδειγμα των συνδρομών Concurros Strategy, εάν ένας πελάτης κάνει την αγορά ενός μαθήματος, είναι απαραίτητο, στη βάση δεδομένων της εταιρείας, τα δεδομένα αγοράς να εισαχθούν στον πίνακα πωλήσεων, ώστε να δημιουργηθεί το δελτίο πληρωμής και μόλις ληφθεί η πληρωμή, απελευθερώνεται η πρόσβαση στον ιστότοπο.

Με αυτόν τον τρόπο, δεν είναι δυνατόν, για παράδειγμα, να απελευθερωθεί η πρόσβαση πριν από τη λήψη των δεδομένων πληρωμής, δηλαδή είτε να εκτελεστεί ολόκληρη η λειτουργία είτε να μην εκτελεστεί τίποτα, να εγγυηθεί την ακεραιότητα της συναλλαγής.

Συνοχή

Αυτή η ιδιότητα δηλώνει ότι μια εκτέλεση συναλλαγής πρέπει να μεταφέρει τη βάση δεδομένων από μια συνεπή κατάσταση σε μια άλλη συνεπή κατάσταση,

δηλαδή, όλοι οι περιορισμοί που επιβάλλονται σε μια βάση δεδομένων σε μια συναλλαγή πρέπει να τηρούνται.

Φανταστείτε έναν γιατρό που έχει τον μοναδικό του αριθμό CRM. Είναι δυνατόν περισσότεροι από ένας γιατροί να έχουν τον ίδιο αριθμό CRM; Όχι, γιατί είναι μοναδικό για τον καθένα. Επομένως, είναι σημαντικό να διατηρούνται όλες οι ιδιότητες μιας βάσης δεδομένων, πριν και μετά την πραγματοποίηση συναλλαγών.

Απομόνωση

Η απομόνωση είναι η ιδιότητα που επιτρέπει τις συναλλαγές να εκτελούνται παράλληλα, ταυτόχρονα, χωρίς παρεμβολές μεταξύ τους, επιτρέποντας σε κάθε συναλλαγή να εκτελείται μεμονωμένα, ανεξάρτητα.

Αντοχή

Η ανθεκτικότητα επιτρέπει στα αποτελέσματα των συναλλαγών να είναι μακροχρόνια, να μην παρεμβάλλονται ή να χάνονται όταν, για παράδειγμα, υπάρχει διακοπή ρεύματος.

Με αυτόν τον τρόπο τα αποτελέσματα θα είναι μόνιμα και θα είναι μόνιμα διαθέσιμα.

Για παράδειγμα, φανταστείτε ότι χρησιμοποιείτε ένα ATM και όταν ολοκληρώσετε τη μεταφορά χρημάτων, υπάρχει διακοπή ρεύματος.

Η ιδιότητα ανθεκτικότητας διασφαλίζει ότι η μεταφορά, εάν ολοκληρωθεί, δεν θα χαθεί.

Ορισμένες ερωτήσεις απαιτούν μόνο το όνομα αυτών των ιδιοτήτων και είναι σημαντικό να τις θυμάστε. Ένα καλό μνημονικό για να μην τα ξεχνάμε είναι το ACID (Tomicity, Consistency, Insulation, Durability).

Αρχιτεκτονική Βάσεων Δεδομένων

Ονομάζεται επίσης ANSI/SPARC Architecture ή Three Schema Architecture, χρησιμοποιείται για τη δημιουργία μιας βάσης δεδομένων σε ένα DBMS.

Χωρίζεται σε τρία ανεξάρτητα επίπεδα αφαίρεσης: το Εσωτερικό, το Εννοιολογικό και το Εξωτερικό.

Εσωτερικός

Αυτό το επίπεδο, που ονομάζεται επίσης φυσικό ή επίπεδο αποθήκευσης, είναι υπεύθυνο για την ανησυχία σχετικά με το πώς αποθηκεύονται φυσικά τα δεδομένα, όπως στο υλικό ενός υπολογιστή. Επιπλέον, παρέχει τις απαραίτητες διαδρομές για πρόσβαση σε καθένα από αυτά τα δεδομένα.

Ονομάζεται επίσης λογικό επίπεδο, είναι ένα επίπεδο μεταξύ του εσωτερικού και του εξωτερικού, που είναι υπεύθυνο για τον καθορισμό της δομής της εν λόγω βάσης δεδομένων, για τον καθορισμό των δεδομένων που θα αποθηκευτούν και της σχέσης μεταξύ τους. Σχετίζεται με το λογικό επίπεδο, σε επίπεδο οντότητας, κρύβοντας τη δομή της φυσικής αποθήκευσης.

Με άλλα λόγια, αυτό το επίπεδο ασχολείται με το πώς περιγράφονται οι πίνακες δεδομένων και σχετίζονται μεταξύ τους μέσω λογικού προγραμματισμού.

Εξωτερικός

Τέλος, το εξωτερικό επίπεδο είναι το επίπεδο που βρίσκεται πλησιέστερα στους χρήστες, επιτρέποντας την παράδοση διαφορετικών προβολών σε αυτούς, προκειμένου να διευθετηθεί πώς θα βλέπουν τα δεδομένα από τους χρήστες, μεμονωμένα, δηλαδή αυτό είναι το επίπεδο που είναι υπεύθυνο για τη διεπαφή του στην βάση δεδομένων με τον τελικό χρήστη.

Επιτρέπει την εμφάνιση δεδομένων που σχετίζονται με τους χρήστες, ώστε να αποκρύπτονται αυτά που δεν είναι, ξεχωριστά και ανεξάρτητα για τον καθένα.

Ας δώσουμε ένα παράδειγμα και για τα τρία επίπεδα αρχιτεκτονικής της βάσης δεδομένων.

Φανταστείτε, πάλι, μια βάση δεδομένων διαθέσιμη σε ένα ΑΤΜ. Περιέχει πληροφορίες από χιλιάδες χρήστες, έτσι δεν είναι; Αυτά τα δεδομένα, εσωτερικά, αποθηκεύονται φυσικά κάπου, όπως στον σκληρό δίσκο ενός υπολογιστή.

Το εννοιολογικό επίπεδο, από την άλλη πλευρά, παρέχει πώς θα συσχετιστούν αυτά τα δεδομένα, παρέχοντας, για παράδειγμα, ποια θα είναι τα κύρια κλειδιά, τα χαρακτηριστικά και οι σχέσεις των πληροφοριών.

Το εξωτερικό επίπεδο, από την άλλη πλευρά, ορίζει τη διεπαφή του DB με τους χρήστες, επιτρέποντας στους κατόχους λογαριασμών να έχουν ατομική πρόσβαση στα δεδομένα που ζητούν, με κάθε χρήστη να έχει πρόσβαση μόνο στα τραπεζικά δεδομένα του ενδιαφέροντος του, τα οποία θα καλύπτουν τις ανάγκες τους. Είναι διαθέσιμα δεδομένα από άλλους κατόχους λογαριασμού σε άλλους που δεν είναι ο κάτοχος.

Είναι σημαντικό να συνειδητοποιήσουμε ότι αυτά τα τρία επίπεδα επικοινωνούν ανά πάσα στιγμή, επιτρέποντας την ακεραιότητα των δεδομένων και αυτή η επικοινωνία ονομάζεται χαρτογράφηση. Το επίπεδο που θεωρείται υψηλότερο θα αντιστοιχιστεί σε ένα αμέσως χαμηλότερο επίπεδο.

Ποια είναι όμως τα ανώτερα και τα κατώτερα επίπεδα;

Η σειρά που εξετάζεται, από το χαμηλότερο επίπεδο στο υψηλότερο, είναι Εσωτερική, Εννοιολογική και Εξωτερική.

Με αυτόν τον τρόπο, υπάρχει η Εξωτερική/Εννοιολογική χαρτογράφηση, η οποία καθορίζει την αντιστοιχία μεταξύ της εξωτερικής και της εννοιολογικής άποψης. Η Εννοιολογική/Εσωτερική χαρτογράφηση καθορίζει τον τρόπο με τον οποίο η εννοιολογική δομή αποθηκεύεται φυσικά.

Αυτές οι αντιστοιχίσεις επιτρέπουν την ανεξαρτησία μεταξύ των δεδομένων σε κάθε επίπεδο. Με αυτόν τον τρόπο, η λογική ανεξαρτησία επιτρέπει στο εννοιολογικό σχήμα να αλλάξει χωρίς να τροποποιηθούν τα εξωτερικά σχήματα. Ενώ η φυσική ανεξαρτησία των δεδομένων επιτρέπει την τροποποίηση των φυσικών σχημάτων χωρίς αλλαγή των εννοιολογικών σχημάτων.

Έργα Βάσεων Δεδομένων

Παιδιά, αυτή είναι μια άλλη ταξινόμηση που σας επιτρέπει να αναπαραστήσετε τα διαφορετικά επίπεδα αφαίρεσης μιας βάσης δεδομένων. Εδώ χρειάζεται μεγάλη προσοχή, καθώς η ονοματολογία των επιπέδων μοιάζει αρκετά με αυτές που φαίνονται παραπάνω. Τα μοντέλα που υπάρχουν σε αυτή τη δομή είναι το Εννοιολογικό Μοντέλο, το Λογικό Μοντέλο και το Φυσικό Μοντέλο.

Το φυσικό μοντέλο χαμηλού επιπέδου περιγράφει πώς τα δεδομένα θα αποθηκευτούν φυσικά στον υπολογιστή, χωρίς να είναι εύκολα προσβάσιμα στους τελικούς χρήστες.

Το εννοιολογικό μοντέλο θεωρείται μοντέλο υψηλού επιπέδου, το οποίο χρησιμοποιεί τις έννοιες των οντοτήτων, των χαρακτηριστικών και των σχέσεων. Χρησιμοποιείται για τον καθορισμό των προτεραιοτήτων των πελατών και των επιχειρηματικών πτυχών. Εδώ, δεν υπάρχει ακόμη χρήση καμίας τεχνολογίας, αφού υπάρχει μόνο ένας προηγούμενος σχεδιασμός για το πώς θα αναπτυχθεί η σχέση στη βάση δεδομένων.

Τέλος, το λογικό μοντέλο σχετίζεται με τον ορισμό του τρόπου με τον οποίο θα συσχετιστούν λογικά τα δεδομένα, μέσω πινάκων, πρωτεύοντων κλειδιών, περιορισμών. Εξαρτάται από την τεχνολογία και το μοντέλο πρέπει να σχεδιαστεί σύμφωνα με το πρόγραμμα που θα χρησιμοποιηθεί.

Φυσικό μοντέλο:

Πώς αποθηκεύονται φυσικά τα δεδομένα.

Εννοιολογικό μοντέλο: είναι ο εννοιολογικός σχεδιασμός του τι και πώς θα τακτοποιηθούν και θα συσχετιστούν τα δεδομένα. Δεν εξαρτάται από την τεχνολογία που χρησιμοποιείται.

Λογικό μοντέλο: υλοποιείται αφού σχεδιαστεί το εννοιολογικό μοντέλο, προκειμένου να δημιουργηθούν σχέσεις μέσω λογικού προγραμματισμού, όπως πρωτεύοντα κλειδιά, ξένα κλειδιά, περιορισμοί, μεταξύ άλλων ορισμών. Εξαρτάται από την τεχνολογία που χρησιμοποιείται.

Ποιοι είναι οι 12 κύριοι τύποι βάσεων δεδομένων!

Πρώτα απ' όλα, για να εργαστείτε με μια βάση δεδομένων είναι απαραίτητο να κατανοήσετε το είδος της ζήτησης της επιχείρησής σας, ώστε να εντοπίσετε αυτό που σχετίζεται περισσότερο με την εταιρεία σας.

Εκτός από το ότι υπάρχουν διάφοροι τύποι τραπεζών που διατίθενται στην αγορά, μπορούν επίσης να χωριστούν σε δύο κατηγορίες, οι οποίες είναι:

1. Σχεσιακές βάσεις δεδομένων

Σήμερα, οι σχεσιακές βάσεις δεδομένων είναι οι πιο κοινές στην αγορά, όπως για παράδειγμα στα συστήματα ERP και CRM. Αυτή η δημοτικότητα επιτυγχάνεται με την ευκολία αποθήκευσης και την αξιοπιστία των πληροφοριών.

Λειτουργεί ως εξής: τα δεδομένα αποθηκεύονται σε σωληνοειδείς μορφές, δηλαδή τα δεδομένα βρίσκονται στη στήλη, ενώ η περιγραφή είναι σε γραμμές και χαρακτηριστικά. Η σημασία των σχεσιακών βάσεων δεδομένων προκύπτει από τους πυλώνες ACID, που είναι το ακρωνύμιο που αναφέρεται στην ατομικότητα, τη συνέπεια, την απομόνωση και την ανθεκτικότητα.

Πόσο σημαντική είναι η SQL για μη σχεσιακές βάσεις δεδομένων;

Τέλος, ένα άλλο σημαντικό χαρακτηριστικό είναι η γλώσσα του, η οποία βασίζεται σε SQL (Structured Query Language). Αυτός ακριβώς είναι ο λόγος που η σχεσιακή κατηγορία θεωρείται εύκολη, καθώς η εισαγωγή δεδομένων είναι εύκολη και ανακτήσιμη. Αξίζει να αναφέρουμε ότι οι περισσότεροι τύποι βάσεων δεδομένων χρησιμοποιούν SQL.

Μη σχεσιακές βάσεις δεδομένων

Χρησιμοποιούμε το μη σχεσιακό όποτε χρειάζεται να εργαστούμε με δεδομένα που δεν μπορούν να εισαχθούν σε μορφή πίνακα, όπως εικόνες, βίντεο και γραφήματα. Έχει υψηλές επιδόσεις και για αυτό το λόγο εκτιμάται ιδιαίτερα στην αγορά, όλοι οι δίσκοι γίνονται στον ίδιο χώρο.

Ως αποτέλεσμα, οι μη σχεσιακές βάσεις δεδομένων δεν απαιτούν τη χρήση ενός συστήματος σχέσεων, σε αντίθεση με μια σχεσιακή βάση δεδομένων. Η επίσημη γλώσσα του είναι η NoSQL (Όχι μόνο SQL).

Καθώς είναι ένα σύστημα που απορροφά κάποιο πιο περίπλοκο περιεχόμενο, το μη σχετιζόμενο απαιτεί ένα σύστημα έγκρισης, το οποίο επιτρέπει τον προσδιορισμό του είδους των πληροφοριών που σχετίζονται περισσότερο με την επιχείρηση.

Έχοντας αυτό υπόψη, ας δούμε τώρα ποιοι τύποι βάσεων δεδομένων υπάρχουν και για ποιες περιπτώσεις είναι οι πιο κατάλληλες:

Αντικειμενοστρεφείς βάσεις δεδομένων

Σε αυτόν τον τύπο βάσης δεδομένων, χρησιμοποιείται η αντικειμενοστραφής δομή. Αυτό σημαίνει ότι οι πληροφορίες είναι οργανωμένες σε μπλοκ πληροφοριών με αναγνωριστικά. Σε αυτό το μοντέλο δεν υπάρχει προκαθορισμένη λογική, η οποία το διαφοροποιεί από τη σχεσιακή βάση δεδομένων, στην οποία υπάρχει ένας πίνακας με γραμμές και στήλες.

Για παράδειγμα, ας υποθέσουμε ότι θέλουμε να αποθηκεύσουμε πληροφορίες για εταιρείες. Έτσι, δημιουργούμε ένα αναγνωριστικό που ονομάζεται "εταιρεία" και σε αυτό περνάμε χαρακτηριστικά με τις αντίστοιχες τιμές, για παράδειγμα, όταν δημιουργούμε ένα χαρακτηριστικό που ονομάζεται "name_fantasy", μπορούμε στη συνέχεια να περάσουμε την τιμή που αναφέρεται στο φανταχτερό όνομα της εταιρείας που θέλουμε να αποθηκεύσουμε.

Αυτόνομες Βάσεις Δεδομένων

Αυτός είναι ένας τύπος βάσης δεδομένων που διαχειρίζεται η μηχανική εκμάθηση, στο cloud. Με αυτόν τον τρόπο, τα αντίγραφα ασφαλείας, η ασφάλεια, τα ερωτήματα και άλλες εργασίες ρουτίνας είναι υπό την ευθύνη της τεχνητής νοημοσύνης.

Αυτός ο τύπος βάσης δεδομένων είναι ιδανικός για καταστάσεις όπου υπάρχει μεγάλη ροή δεδομένων και οι εμπλεκόμενοι μπορούν εύκολα να χαθούν όταν εκτελούν εργασίες με μη αυτόματο τρόπο.

Με αυτόν τον τρόπο, η ανάθεση αυτών των λειτουργιών στο AI σημαίνει κέρδος σε χρόνο και ταχύτητα.

Ποια είναι η σχέση μεταξύ Αυτόνομων Βάσεων Δεδομένων και Μηχανικής

Μάθησης:

Οι αυτόνομες βάσεις δεδομένων και η μηχανική εκμάθηση σχετίζονται άμεσα. Η μηχανική μάθηση είναι μια τεχνητή νοημοσύνη που εκπαιδεύεται για την εκτέλεση συγκεκριμένων εργασιών.

Με αυτόν τον τρόπο, η αυτόνομη τράπεζα κάνει χρήση αυτής της δυνατότητας για να χειριστεί τις συνήθεις εργασίες της, όπως η ρουτίνα δημιουργίας αντιγράφων ασφαλείας, για παράδειγμα.

Βάσεις δεδομένων OLTP

Η βάση δεδομένων OLTP εκτελεί επεξεργασία δεδομένων σε ηλεκτρονικές συναλλαγές. Αυτός ο τύπος αποθήκευσης είναι ιδανικός για το χειρισμό πληροφοριών οργανισμού όπως οι πληρωμές και η διαχείριση παραγγελιών.

Αυτό οφείλεται στην ανάγκη αυτών των οργανισμών να εκτελούν ατομικές και συνεπείς διαδικασίες. Αυτό σημαίνει ότι κάθε συναλλαγή πρέπει να πετύχει ή να αποτύχει. Σε περίπτωση αποτυχίας, όλα τα βήματα που έχουν περάσει πρέπει να αναιρεθούν.

Κατανεμημένες βάσεις δεδομένων

Μια κατανεμημένη βάση δεδομένων αποτελείται από ένα δίκτυο κόμβων, όπου κάθε κόμβος είναι ένας υπολογιστής σε διαφορετική τοποθεσία. Για την επικοινωνία, τα μηχανήματα που είναι συνδεδεμένα στο σύστημα χρησιμοποιούν το δίκτυο.

Ο κύριος διαφοροποιητής αυτού του τύπου βάσης δεδομένων είναι η διανομή των πληροφοριών του. Ενώ τα κεντρικά μοντέλα έχουν ολόκληρη τη βάση τους σε ένα μέρος και είναι ευάλωτα σε μη αναστρέψιμες επιθέσεις και απώλειες, σε μια κατανεμημένη τράπεζα οι πληροφορίες μπορούν να ανακτηθούν από έναν άλλο κόμβο.

Βάση δεδομένων JSON

Είναι μια βάση δεδομένων που αποθηκεύει και υποβάλλει ερωτήματα σε έγγραφα JSON. Το JSON είναι ένα στοιχείο κειμένου, αλλά μπορεί επίσης να λάβει άλλες τιμές, όπως ένα BLOB, που χρησιμοποιείται για την αποθήκευση εικόνων. Οι πίνακες μπορούν να χρησιμοποιηθούν για τη διαχείριση και τον χειρισμό αυτής της βάσης δεδομένων. Η επιλογή του πίνακα βασίζεται στο μέγεθος του JSON.

Αποθήκες δεδομένων

Οι αποθήκες δεδομένων επιτρέπουν την ενοποίηση δεδομένων από διαφορετικές πηγές. Εκτός από τη διατήρηση των τρεχόντων δεδομένων, είναι επίσης δυνατό να

συμβουλευτείτε ένα ιστορικό. Γενικά, η αποθήκη δεδομένων είναι κατάλληλη για τη δημιουργία αναφορών ανάλυσης.

Με αυτόν τον τύπο συστήματος, είναι δυνατή η συγκέντρωση μεγάλων ποσοτήτων δεδομένων. Επιπλέον, λόγω της ευελιξίας του, είναι δυνατή η ονομασία στηλών και η τροποποίηση της σχέσης μεταξύ των πινάκων, ώστε η αναφορά που θα δημιουργηθεί να είναι όσο το δυνατόν εξατομικευμένη για το άτομο που θα την αναλύσει.

Γραφικές βάσεις δεδομένων

Οι γραφικές βάσεις δεδομένων αποθηκεύουν πολύπλοκες δομές δεδομένων, οι οποίες θα ήταν ασύμβατες σε μια παραδοσιακή βάση δεδομένων. Αυτός ο τύπος συστήματος χρησιμοποιείται συχνά στα κοινωνικά δίκτυα καθώς είναι ιδανικός για την αντιμετώπιση δεδομένων υψηλής διασύνδεσης. Έτσι, σε αυτόν τον τύπο βάσης δεδομένων, οι πληροφορίες διασυνδέονται με γραφήματα σύνδεσης, όπου οι πληροφορίες έχουν μεγαλύτερη σημασία από τη δομή.

Βάσεις δεδομένων ανοιχτού κώδικα

Αυτός ο τύπος βάσης δεδομένων είναι οποιαδήποτε βάση δεδομένων που μπορεί να διανεμηθεί και να επαναχρησιμοποιηθεί. Αυτή η δυνατότητα παρέχει στους προγραμματιστές τις απαραίτητες προϋποθέσεις για τη δημιουργία νέων τεχνολογιών, οι οποίες χρησιμοποιούν ως πρότυπο τον ανοιχτό κώδικα του συστήματος.

Οι βάσεις δεδομένων ανοιχτού κώδικα μπορούν να ταξινομηθούν σε σχεσιακές και μη σχεσιακές. Μερικά παραδείγματα που εμπίπτουν σε αυτόν τον τύπο βάσης δεδομένων είναι οι MySQL, PostgreSQL και MongoDB.

Βάσεις δεδομένων cloud

Μια βάση δεδομένων cloud είναι μια βάση δεδομένων στην οποία μπορείτε να έχετε πρόσβαση μέσω μιας διαδικτυακής πλατφόρμας. Αυτό σημαίνει ότι οι πληροφορίες δεν αποθηκεύονται σε τοπικό διακομιστή, αλλά σε απομακρυσμένους διακομιστές.

Όλο το σύστημα λειτουργεί ως υπηρεσία. Με αυτόν τον τρόπο, ο συμβαλλόμενος μπορεί να πληρώσει τέλη χρήσης ή να αποκτήσει ένα πρόγραμμα που παρέχει συγκεκριμένο χώρο αποθήκευσης, ταχύτητα κυκλοφορίας, μεταξύ άλλων χαρακτηριστικών.

Επιπλέον, αυτός που διαχειρίζεται και χειρίζεται τα αποθηκευμένα δεδομένα είναι ο πάροχος, δηλαδή η συμβατική πλατφόρμα. Γενικά, αυτού του είδους η υπηρεσία μειώνει σημαντικά το κόστος υλικού και συντήρησης.

Βάση δεδομένων πολλαπλών μοντέλων

Η βάση δεδομένων πολλαπλών μοντέλων πραγματοποιεί έναν συνδυασμό μιας ποικιλίας βάσεων δεδομένων σε ένα μόνο backend. Κάτι που το καθιστά εξαιρετικά ευέλικτο όσον αφορά τα δεδομένα που μπορούν να αποθηκευτούν και τις δομές που μπορούν να συναρμολογηθούν.

Τα 10 κορυφαία DBMS στην παγκόσμια αγορά

Oracle



Η Oracle, που δημιουργήθηκε τη δεκαετία του 1980, δημιουργήθηκε από τον Larry Ellison, είναι πλέον μία από τις μεγαλύτερες εταιρείες τεχνολογίας στον κόσμο και από το 2009 κατέχει επίσης τις γραμμές λογισμικού Java.

Το DBMS είναι το κύριο προϊόν της και από την κυκλοφορία του στην αγορά έχει τελειοποιηθεί και αναπτυχθεί για να ανταποκρίνεται στις ανάγκες των εταιρειών και του συνδεδεμένου κόσμου στον οποίο ζούμε. Υπάρχουν πολλές εκδόσεις του λογισμικού, καθεμία από τις οποίες έχει χαρακτηριστικά που το καθιστούν ιδανικό

για διαφορετικά επιχειρηματικά μοντέλα. Είναι ένα λογισμικό που εστιάζει στις ανάγκες των μεσαίων και μεγάλων επιχειρήσεων.

Αξίζει να θυμηθούμε ότι, μέχρι στιγμής, το Oracle DBMS είναι σχεσιακό. Για χειρισμό και διαχείριση συστήματος, χρησιμοποιείται η γλώσσα PL/SQL. Γλώσσα που αναπτύχθηκε από την Oracle, από την SQL (ANSI), επεκτείνοντας την αρχική της χωρητικότητα.

Κυριαρχία στην αγορά

Με ευρεία κυριαρχία στην αγορά, η γνώση της γλώσσας που χρησιμοποιείται στο DBMS της Oracle είναι απαραίτητη για τον επαγγελματία πληροφορικής που θέλει να ξεκινήσει μια καριέρα ως DBA. Μπορείτε να λάβετε περισσότερες πληροφορίες επισκεπτόμενοι τον επίσημο ιστότοπο της Oracle.



MySQL

Μία από τις πιο χρησιμοποιούμενες και δημοφιλείς βάσεις δεδομένων είναι μια τεχνολογία ανοιχτού κώδικα, δηλαδή ανοιχτού κώδικα. Αυτό επιτρέπει την ανάπτυξη του ίδιου να πραγματοποιηθεί σύμφωνα με τις ανάγκες ενός οργανισμού.

Επιπλέον, η ευκολία χρήσης και η ικανότητά του να εκτελείται σε διαφορετικά λειτουργικά συστήματα διασφαλίζουν ότι είναι το αγαπημένο των μεγάλων εταιρειών σε όλο τον κόσμο. Θεωρείται το πιο δημοφιλές DBMS στον κόσμο.

Oracle + MySQL = ?

Επί του παρόντος, ανήκει στην Oracle, η οποία, εκτός από την προώθηση της χρήσης της τεχνολογίας Ανοιχτού Κώδικα, παρέχει και μια πιο στιβαρή έκδοση (με πληρωμή, φυσικά), με εταιρικά χαρακτηριστικά όπως: παρακολούθηση υψηλής

διαθεσιμότητας, τείχη προστασίας, προηγμένα αντίγραφα ασφαλείας, έλεγχος και μεταξύ άλλων.

Για τον επαγγελματία πληροφορικής, η γνώση του τρόπου χρήσης της MySQL, που είναι ένα σχεσιακό DBMS, μπορεί να θεωρηθεί εξίσου σημαντική με τη γνώση του Oracle DBMS. Με άλλα λόγια, είναι πρακτικά υποχρεωτικό για όσους θέλουν να εκμεταλλευτούν όλες τις ευκαιρίες της αγοράς (μικρές, μεσαίες και μεγάλες επιχειρήσεις).

SQL Server



Το ισχυρό σχεσιακό DBMS της Microsoft , που κυκλοφόρησε το 1988 ως μέρος των Windows NT και αργότερα κυκλοφόρησε ως ξεχωριστό προϊόν και έκτοτε βρίσκεται σε συνεχή ανάπτυξη.

Η μεγάλη του διαφορά σε σχέση με τις άλλες επιλογές που έχουν ήδη αναφερθεί είναι η δυνατότητα του προγραμματιστή να χρησιμοποιεί διαχειριζόμενες γλώσσες προγραμματισμού, όπως C# και Visual Basic .NET, αντί να χρησιμοποιεί εντολές SQL. Φυσικά, μπορείτε επίσης να χρησιμοποιήσετε SQL ή μάλλον επέκταση T-SQL – SQL (ANSI) για MS SQL Server. Ο MS SQL Server επιτρέπει επίσης διαφανή ερωτήματα.

Ο Microsoft SQL Server είναι μια εξαιρετικά ισχυρή επιλογή και, παρόλο που είναι μια πλήρως πληρωμένη λύση, βρίσκεται πάντα μεταξύ των TOP 5 των κύριων κατατάξεων στην περιοχή.

Μπορείτε να λάβετε περισσότερες πληροφορίες επισκεπτόμενοι τον ιστότοπο του προϊόντος.

PostgreSQL



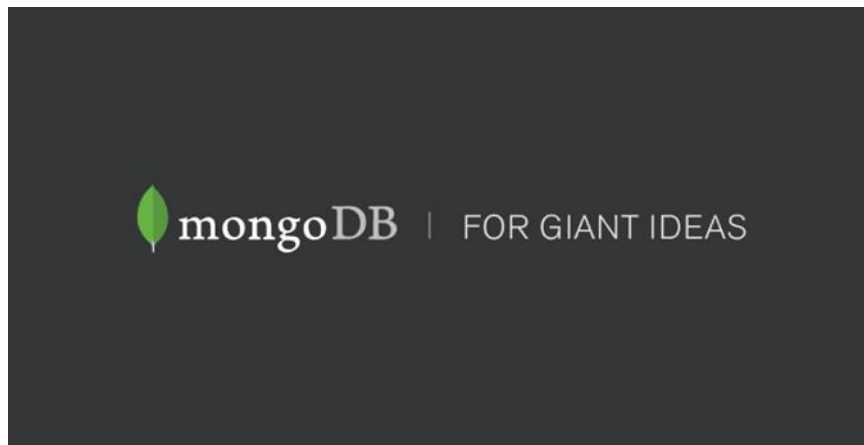
Σχεσιακή βάση δεδομένων και βάση δεδομένων ανοιχτού κώδικα, που αναπτύχθηκε από την PostgreSQL Global Development Group. Ξεκίνησε το 1989 και κατατάσσεται συχνά μεταξύ των πέντε κορυφαίων συστημάτων διαχείρισης βάσεων δεδομένων που χρησιμοποιούνται από οργανισμούς σε όλο τον κόσμο.

Όπως και η MySQL, χρησιμοποιείται ευρέως για διαδικτυακά συστήματα, επιτρέποντας την ανάπτυξη λύσεων να πραγματοποιηθεί με τρόπο που να ανταποκρίνεται στις συγκεκριμένες ανάγκες μιας επιχείρησης, επιτυγχάνοντας έτσι καλύτερη απόδοση.

Η πιο προηγμένη βάση δεδομένων ανοιχτού κώδικα στον κόσμο.

[easy-tweet tweet="Η πιο προηγμένη βάση δεδομένων ανοιχτού κώδικα στον κόσμο #PostgreSQL" user="DGROU" hashtags="DGROU" template="light"]

5.MongoDB



Ένα από τα ταχύτερα αναπτυσσόμενα DBMS τα τελευταία χρόνια. Είναι ένα σύστημα NoSQL που κυκλοφόρησε το 2009 από την εταιρεία που φέρει το ίδιο όνομα.

Το DBMS επιδιώκει να ενώσει τα καλύτερα σχεσιακά συστήματα και τις καινοτομίες του NoSQL, διατηρώντας πολλά χαρακτηριστικά του πρώτου, όπως ευρετήρια και

δυναμικά ερωτήματα, αλλά και με το μοντέλο δεδομένων προσανατολισμένο στα έγγραφα.

Με αυτόν τον τρόπο, τα κέρδη ευελιξίας μέσω ευέλικτων σχημάτων και η μεγαλύτερη ευκολία οριζόντιας επεκτασιμότητας είναι σημαντικά σε σύγκριση με άλλες λύσεις. Το λογισμικό είναι ανοιχτού κώδικα, όπως και οι MySQL και PostgreSQL. Αυτό επιτρέπει την ανάπτυξη του συστήματος σύμφωνα με τις ανάγκες της επιχείρησης.

Με την αύξηση της προτίμησης στη χρήση του Mongo DB, ο επαγγελματίας πληροφορικής που κατέχει τη JavaScript (τη γλώσσα που χρησιμοποιείται από το σύστημα), έχει μεγάλες πιθανότητες να ξεχωρίσει στην τρέχουσα αγορά.



DB2

Βάση δεδομένων από τον γίγαντα IBM. Κυκλοφόρησε τη δεκαετία του 1980, αν και η ανάπτυξή του έλαβε χώρα κατά τη δεκαετία του 70. Χρησιμοποιώντας τη θεωρία των σχεσιακών βάσεων δεδομένων, που αναπτύχθηκε από τον Edgar Frank Codd, ο οποίος εκείνη την εποχή εργαζόταν για την εταιρεία.

Γρήγορες λεπτομέρειες:

Η IBM δεν πίστευε στις δυνατότητες του Codd και στη γλώσσα που είχε αναπτύξει, αφήνοντάς τον έξω από τα έργα της και υποτιμώντας αρκετές από τις ιδέες του μαθηματικού. Με άλλα λόγια, ο πατέρας της θεωρίας των σχεσιακών βάσεων δεδομένων ήταν ένα απλό πρόσθετο στην ανάπτυξη του τελικού προϊόντος.

Επιστρέφοντας στα DB2 και IBM, το σύστημα χαρακτηρίζεται από το πρωτοποριακό του πνεύμα και επιπλέον, από την υποστήριξή του για λειτουργικά συστήματα.

Ιστορικά, αμφισβήτησε τη θέση της στην αγορά με την Oracle για πολλά χρόνια.

Ωστόσο, η εμφάνιση και η ανάπτυξη νέων χαρακτηριστικών σε άλλα DBMS

συνέβαλε σε σημαντική απώλεια χώρου για το DB2. Εξακολουθεί να είναι ένα εξαιρετικά σχετικό σύστημα. Μπορείτε να λάβετε περισσότερες πληροφορίες μεταβαίνοντας στον επίσημο ιστότοπο της IBM.



Cassandra

Μη σχεσιακή βάση δεδομένων (NoSQL), που αναπτύχθηκε αρχικά από το Facebook το οποίο, το 2008, έκανε το DBMS ανοιχτού κώδικα. Άνοιξε δηλαδή τον κωδικό στην αγορά. Επί του παρόντος, διατηρείται από προγραμματιστές του ιδρύματος Apache (μη κερδοσκοπικός οργανισμός που δημιουργήθηκε με στόχο την παροχή υποστήριξης για έργα ανοιχτού κώδικα).

Όπως και με τις εναλλακτικές λύσεις NoSQL, το μεγάλο πλεονέκτημά της είναι η ικανότητά της να κλιμακώνεται χωρίς απώλεια απόδοσης. Και όπως και το MongoDB, χρησιμοποιεί JavaScript ως προεπιλεγμένη γλώσσα προγραμματισμού.

Προσοχή στις λεπτομέρειες!

Το 2008, το Facebook, μια εταιρεία εξαιρετικά μπροστά από την εποχή της, χρειάστηκε να αναπτύξει το δικό της DBMS, καθώς τα DBMS της εποχής δεν κάλυπταν τις ανάγκες του. Ναι, αυτό είναι μια ένδειξη ότι το NoSQL είναι μια τάση και είναι εξαιρετικά σχετική γνώση. Και ενδεχομένως, στο μέλλον, θα μπορούσε να απειλήσει την ηγεμονία των σχεσιακών συστημάτων. Έτσι, για όσους σκέφτονται

μακροπρόθεσμα, η γνώση της γλώσσας JavaScript θα είναι επίσης απαραίτητη για τους διαχειριστές βάσεων δεδομένων (DBA).

Μπορείτε να λάβετε περισσότερες πληροφορίες για το σύστημα εδώ μεταβαίνοντας στη διεύθυνση cassandra.apache.org



Microsoft Access

Ναι, η Access θεωρείται σχεσιακό DBMS. Είναι μια τεχνολογία που αναπτύχθηκε από τη Microsoft και επί του παρόντος μέρος της σουίτας του Office.

Η γλώσσα ανάπτυξης που χρησιμοποιείται στο DBMS είναι η Visual Basic (VBA), αν και είναι επίσης συμβατή με SQL. Αυτό επιτρέπει στους χρήστες να συνδυάζουν και τα δύο στο χειρισμό δεδομένων.

Η Access ήταν κάποτε μια τεχνολογία μπροστά από την εποχή της. Σήμερα, ενώ είναι χρήσιμο για την ανάπτυξη μικρότερων έργων και για τη διδασκαλία σε κολέγια, η Microsoft Access αντιμετωπίζει προβλήματα κλιμάκωσης σε μεγαλύτερα έργα. Λόγω αυτού του παράγοντα, αγνοείται στις μεσαίες και μεγάλες εταιρείες. Παρά όλα τα προβλήματά του, το σύστημα εξακολουθεί να έχει καλό αριθμό χρηστών.

Μπορείτε να λάβετε περισσότερες πληροφορίες μεταβαίνοντας στον επίσημο ιστότοπο της Microsoft.



Redis

Η βάση δεδομένων NoSQL με βάση το κλειδί-τιμή, μπορεί να αποθηκεύσει διαφορετικούς τύπους δεδομένων όπως: συμβολοσειρές, λίστες, σύνολα, κατακερματισμοί και ταξινομημένα σύνολα.

Πολύ γρήγορο και ελαφρύ, είναι ικανό να αντέξει το C10k και την αναπαραγωγή. Και όπως παρόμοιες εναλλακτικές λύσεις, έχει αυξηθεί σημαντικά η προτίμηση για χρήση μεταξύ των εταιρειών. Η απλή χρήση του είναι ένα από τα μεγάλα πλεονεκτήματα.

Σημαντικό: ένα ακόμη NoSQL DBMS μεταξύ των 10 κορυφαίων θέσεων. Είναι ο Μαρκ (Facebook) οραματιστής ή όχι;

```
[easy-tweet tweet="Άλλο ένα NoSQL DBMS ανάμεσα στα κορυφαία 10. Είναι ο Μαρκ (Facebook) οραματιστής ή δεν είναι;" user="DGROUP" usehashtags="no" template="light"]
```

Μπορείτε να λάβετε περισσότερες πληροφορίες επισκεπτόμενοι τον ιστότοπο redis.io.



SQLite

Αναπτύχθηκε από τον Richard Hipp, η SQLite είναι μια βιβλιοθήκη σε γλώσσα C, η οποία επιτρέπει τη χρήση μιας βάσης δεδομένων SQL. Έτσι, αντί να συνδεθείτε σε έναν μεγάλο διακομιστή βάσης δεδομένων, είναι ο ίδιος ο διακομιστής.

Η χρήση του συνιστάται όταν η διαχείριση είναι εύκολη και η υλοποίηση και η συντήρηση δεδομένων είναι πιο σημαντικοί παράγοντες από τους αμέτρητους πόρους που μπορούν να προσφέρουν άλλα DBMS. Είναι ιδανικό για ιστότοπους με λίγες προσβάσεις, εφαρμογές επιτραπέζιου υπολογιστή, εργαλεία στατικής ανάλυσης και πολλά άλλα.

Επιπλέον, είναι δωρεάν λογισμικό δημόσιου τομέα, υποστηρίζει δεδομένα κάτω των 2 terabyte και δεν έχει εξωτερικές εξαρτήσεις, γεγονός που διευκολύνει τη χρήση του από μικρομεσαίες επιχειρήσεις. Μπορείτε να λάβετε περισσότερες πληροφορίες μεταβαίνοντας στο sqlite.org



Μπόνους: MariaDB

Το MariaDB αξίζει να αναφερθεί λόγω της ιστορίας του, καθώς προέρχεται από το MySQL, ένα από τα πιο δημοφιλή συστήματα στην παγκόσμια αγορά.

Μετά την αγορά της MySQL από την Oracle, ο ιδρυτής της Michael “Monty” Widenius υιοθέτησε τον ακόλουθο επαγγελματικό στόχο: να αναπτύξει ένα νέο DBMS που διατηρεί και βελτιώνει τα χαρακτηριστικά του προηγούμενου συστήματος. Το αποτέλεσμα αυτού ήταν το MariaDB, που κυκλοφόρησε το 2009.

Με την ίδια λειτουργία με την MySQL, η μεγάλη διαφορά της MariaDB (σχεσιακό DBMS) είναι η έντονη ανησυχία της για την ασφάλεια των δεδομένων και νέες ενημερώσεις κώδικα σχετικά με αυτό διατίθενται συνεχώς στους χρήστες.

Διαφημίζεται ως αντικατάσταση του συστήματος που αναπτύχθηκε προηγουμένως από τη Widenius, MySQL. Έχει καλές προοπτικές ανάπτυξης και χρήσης εντός των οργανισμών. Αξίζει να το μάθετε και να ακολουθήσετε τα επόμενα βήματά σας. Περισσότερες πληροφορίες στο mariadb.org

Εκπαιδευτική υποενότητα : 5.5 Εισαγωγή στη Δομημένη Γλώσσα Ερωταπαντήσεων (SQL)

Η γλώσσα SQL είναι ο πιο δημοφιλής πόρος για DBA και προγραμματιστές για την εκτέλεση εντολών σε σχεσιακές βάσεις δεδομένων . Μέσω αυτού δημιουργούμε πίνακες, στήλες, ευρετήρια, εκχωρούμε δικαιώματα σε χρήστες, καθώς και δεδομένα ερωτημάτων.

Τέλος, με τη χρήση της SQL «μιλάμε» στη βάση δεδομένων.

Αυτή η σύντομη περιγραφή κάνει ήδη ξεκάθαρη τη σημασία της κατανόησης και της γνώσης ενός τέτοιου πόρου.

Γνωρίστε, επίσης, ότι διευκολύνει όχι μόνο την εκτέλεση εργασιών σε DBMS, αλλά και τον διάλογο μεταξύ επαγγελματιών βάσεων δεδομένων και προγραμματιστών.

Με βάση αυτό, η εκμάθηση της SQL, της δομημένης γλώσσας ερωτημάτων, είναι απαραίτητη για όποιον θέλει να εργαστεί σε αυτούς τους τομείς.

Πρώτα βήματα

Αφού γνωρίζετε την SQL, είναι φυσικό να σας ενδιαφέρει να τη δείτε στην πράξη, εκτελώντας τις πρώτες εντολές, τα πρώτα ερωτήματα, ακόμη και σε ένα παράδειγμα βάσης δεδομένων που παρέχεται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων.

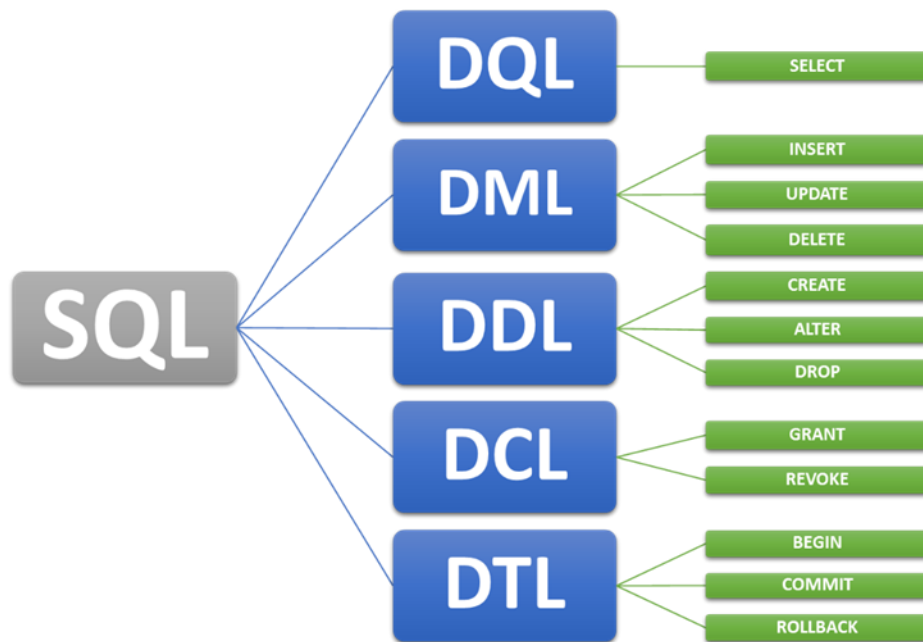
Αλλά, από πού να ξεκινήσετε ακριβώς; Πού να εκτελέσετε αυτές τις εντολές;

Για να απαντήσουμε σε αυτές τις ερωτήσεις, θα δούμε παρακάτω πως: να διαμορφώσουμε το περιβάλλον, να δημιουργήσουμε τον πρώτο πίνακα, να εισαγάγουμε δεδομένα, καθώς και να εκτελέσουμε κάποια ερωτήματα.

Οργάνωση SQL

Στο παραπάνω μάθημα ίσως έχετε παρατηρήσει ότι η γλώσσα SQL είναι οργανωμένη σε υποσύνολα , το καθένα με καλά καθορισμένους σκοπούς όπως την εικόνα.

- DQL - Data Query Language - Καθορίζει την εντολή που χρησιμοποιείται έτσι ώστε να μπορούμε να κάνουμε ερώτημα (SELECT) στα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων.
- DML - Γλώσσα χειρισμού δεδομένων - Καθορίζει τις εντολές που χρησιμοποιούνται για τον χειρισμό δεδομένων στη βάση δεδομένων (INSERT, UPDATE και DELETE).
- DDL - Data Definition Language - Καθορίζει τις εντολές που χρησιμοποιούνται για τη δημιουργία (ΔΗΜΙΟΥΡΓΙΑ) πινάκων, προβολών, ευρετηρίων, ενημέρωσης αυτών των δομών (ALTER), καθώς και κατάργησής τους (DROP).
- DCL - Data Control Language - Καθορίζει τις εντολές που χρησιμοποιούνται για τον έλεγχο της πρόσβασης στα δεδομένα της βάσης δεδομένων, προσθέτοντας (GRANT) και αφαιρώντας (REVOKE) δικαιώματα πρόσβασης.
- DTL - Data Transaction Language - Καθορίζει τις εντολές που χρησιμοποιούνται για τη διαχείριση των συναλλαγών που εκτελούνται στη βάση δεδομένων, όπως η έναρξη (BEGIN) μιας συναλλαγής, η δέσμευσή της (COMMIT) ή η επαναφορά της (ROLLBACK).



Υποδιαιρέσεις γλώσσας SQL

SQL DQL - Γλώσσα ερωτημάτων δεδομένων

Όπως μπορείτε ήδη να δείτε και θα επαληθεύσετε σε όλη τη διάρκεια της καριέρας σας, μια από τις πιο κοινές εργασίες στις βάσεις δεδομένων είναι η εκτέλεση ερωτημάτων. Επομένως, το να γνωρίζουμε πώς να τα δημιουργήσουμε με τον καλύτερο τρόπο είναι πολύ σημαντικό για την απόδοση της εταιρίας και των εφαρμογών που εξαρτώνται από αυτήν.

Σε αυτό το σημείο, ως πρώτο βήμα, δεν χρειάζεται να ανησυχείτε πολύ για αυτό, αλλά σας συνιστούμε να το σκέφτεστε με κάθε νέα δυνατότητα που σχετίζεται με το ερώτημα που μελετάτε.

Παρακάτω θα δούμε πώς να δημιουργείτε ερωτήματα SQL, χρησιμοποιώντας την εντολή SELECT. Έτσι, θα γνωρίσουμε μερικά από τα κύρια χαρακτηριστικά, για παράδειγμα: ταξινόμηση εγγραφών, συναρτήσεις συνάθροισης, συνδέσεις, μεταξύ άλλων.

Η δήλωση SELECT είναι ένα αναπόσπαστο και αρχέγονο μέρος της γλώσσας SQL που γεννήθηκε μαζί με τη θεωρία των σχεσιακών βάσεων δεδομένων. Για πολλούς επαγγελματίες που κυκλοφορούν αυτή τη στιγμή στην αγορά, πρέπει να είναι ο

μόνος γνωστός τρόπος πρόσβασης στις πληροφορίες που είναι αποθηκευμένες σε ένα DBMS.

Αρχικά ορίστηκε στα τέλη της δεκαετίας του '70 και από τότε, η χρήση του έχει αυξηθεί δραματικά και έχει γίνει σχεδόν ένας μοναδικός τρόπος πρόσβασης σε δεδομένα που είναι αποθηκευμένα στο DBMS.

Η δήλωση SELECT είναι εξαιρετικά λειτουργική, εάν χρησιμοποιείται σωστά, φέρνοντας ομαδοποιημένα, ταξινομημένα, λεπτομερή ή αθροιστικά αποτελέσματα από διάφορες πηγές δεδομένων ή πίνακες, σε πολλές διαφορετικές μορφές, χωρίς ο προγραμματιστής να χρειάζεται να γράψει ούτε μια γραμμή κώδικα.

Αλλά αυτή η ίδια οδηγία μπορεί να είναι ο κακός των σοβαρών προβλημάτων απόδοσης στην εφαρμογή σας, εάν δεν είναι γραμμένη σωστά ή εάν η βάση δεδομένων σας δεν είναι έτοιμη να την εκτελέσει με τον τρόπο που την γράψατε.

Παρακάτω παρουσιάζουμε τη δήλωση SELECT. Θα βρείτε τον τρόπο να συγκεντρώσετε αυτήν την οδηγία, με πρόσβαση σε έναν μόνο πίνακα μιας βάσης δεδομένων, αλλά με λεπτομέρειες όπως περιορισμούς δεδομένων, ταξινόμηση, ομαδοποίηση, σύνοψη, μέσος υπολογισμός, μεταξύ άλλων, ώστε να ξεκινήσετε να γράφετε πρώτες οδηγίες με ασφάλεια και απλά.

Θα παρουσιαστεί μια σειρά παραδειγμάτων για να διευκρινιστεί και να γίνει πιο κατανοητό. Θα δείξουμε τη χρήση της δήλωσης SELECT, ώστε να μπορείτε να τη χρησιμοποιήσετε στην ανάπτυξη των εφαρμογών σας, αποφέροντας το καλύτερο όφελος στην απλότητα στην ανάπτυξή της και στην απόδοση στη χρήση της.

ΕΠΙΛΟΓΗ ΑΠΟ table

Η αρχή όλων είναι η πιο βασική μορφή της πρότασης SELECT, χρησιμοποιώντας την πρόταση FROM. Η απλούστερη σύνταξη της δήλωσης είναι:

```
SELECT <list of column> FROM <name of table>
```

Δεν υπάρχει συντομότερη έκδοση αυτής της δήλωσης όταν ο στόχος είναι η ανάκτηση δεδομένων. Υπάρχουν DBMS από ορισμένους κατασκευαστές, όπου η

πρόταση SELECT υπάρχει χωρίς την πρόταση FROM, αλλά μόνο για να γίνει ένας υπολογισμός ή να επιστρέψει μια σταθερά σε ένα μπλοκ προγραμματιζόμενων δηλώσεων. Αλλά αυτό δεν είναι το επίκεντρό μας.

Ας χρησιμοποιήσουμε ως παράδειγμα έναν πίνακα υπαλλήλων μιας υποθετικής εταιρείας που αποθηκεύει τις ακόλουθες πληροφορίες: εγγραφή, όνομα, εποπτεία, τμήμα και μισθός.

Για να ανακτήσουμε τον αριθμό εγγραφής και το όνομα όλων των υπαλλήλων, θα πρέπει να γράψουμε: «ΕΠΙΛΟΓΗ εγγραφής, όνομα».

Ή ακόμα πιο απλό, για να ανακτήσετε όλες τις στήλες του πίνακα, θα αρκούσε να βάλετε ένα «*» στη θέση των ονομάτων των στηλών, με αυτόν τον τρόπο, το DBMS θα καταλάβαινε ότι θέλουμε όλες τις στήλες. Αλλά η πρόταση SELECT δεν μπορεί να ζήσει χωρίς την πρόταση FROM.

Στην ρήτρα FROM καθορίζετε από πού θα προέρχονται τα δεδομένα σας, που θα είναι το όνομα του πίνακα ή των πινάκων. Άρα η εντολή θα ήταν πλήρης ως εξής:

```
SELECT υπάλληλοι, name FROM εργαζόμενοι
```

Στη Λίστα 1 θα βρείτε τα δείγματα δεδομένων από τον πίνακα Υπαλλήλων.

Για την επεξεργασία πιο συγκεκριμένων ερωτημάτων, το φιλτράρισμα των επιστρεφόμενων δεδομένων, καθώς και την εκτέλεση υπολογισμών ή ελέγχων μεγαλύτερης ή μικρότερης αξίας, για παράδειγμα, η SQL μας παρέχει επίσης καλές επιλογές.

Εκτός από τα εργαλεία που αναφέρθηκαν μέχρι τώρα, έχουμε και άλλα πολύ σημαντικά που επηρεάζουν άμεσα τον χρόνο διαβούλευσης. Επομένως, η γνώση του τρόπου χρήσης τους έχει μεγάλη αξία για έναν καλό επαγγελματία.

Ένας πίνακας μπορεί να περιέχει εκατομμύρια εγγραφές. Ωστόσο, η μεταφορά τόσο μεγάλης ποσότητας δεδομένων στη μνήμη μπορεί να είναι ανέφικτη. Επίσης, στις περισσότερες περιπτώσεις χρειάζεται να τροποποιήσουμε ή να έχουμε πρόσβαση σε λίγες εγγραφές. Η γλώσσα SQL μας επιτρέπει να παραμετροποιήσουμε τα ερωτήματα για να φιλτράρουμε την απόδοσή τους.

Χρησιμοποιούμε WHERE για να φιλτράρουμε δεδομένα σε μια εντολή SQL . Αυτή η ρήτρα πρέπει να ακολουθείται από μια λογική έκφραση.

Το παρακάτω είναι ένα παράδειγμα της σύνταξης αυτής της πρότασης.

```
WHERE [product]
```

Λογικές εκφράσεις

Ονομάζουμε λογική έκφραση το σύνολο μιας ή περισσότερων συνθηκών που πρέπει να ελέγχονται από τον όρο WHERE. Αυτές οι εκφράσεις δημιουργούνται χρησιμοποιώντας τελεστές.

Ισότητα

Αυτός ο τελεστής εκτελεί τη λειτουργία ισότητας μεταξύ δύο τιμών. Με αυτό, ελέγχουμε αν η τιμή στα δεξιά του συμβόλου = είναι ίση με την τιμή που περιέχεται στη στήλη που αναφέρεται στα αριστερά.

Στο παρακάτω παράδειγμα αναζητούμε ένα προϊόν με βάση την περιγραφή του:

```
SELECT
```

```
    description,
```

```
    premature
```

```
FROM
```

```
    products
```

```
ONDE
```

```
    description = "School Notebook"
```

Θα λάβουμε την πρώτη σειρά στον πίνακα προϊόντων ως επιστροφή.

Διαφορά

Μπορούμε επίσης να επιλέξουμε δεδομένα σε έναν πίνακα χρησιμοποιώντας τον τελεστή <> , ο οποίος εφαρμόζει την πράξη ανισότητας μεταξύ δύο τιμών.

Στο παρακάτω παράδειγμα αναζητούμε όλα τα προϊόντα με κατηγορία διαφορετική από 2.

```
SELECT  
  
    description,  
  
    premature  
  
FROM  
  
    products  
  
WHERE  
  
    category <> 2
```

Στο τέλος αυτού του ερωτήματος, θα επιστραφεί μόνο η τελευταία εγγραφή του πίνακα προϊόντων, καθώς είναι η μόνη που πληροί τα καθορισμένα κριτήρια.

Μικρότερη από

Ο τελεστής < ελέγχει εάν η τιμή που εισάγεται στα αριστερά είναι μικρότερη από αυτή στα δεξιά. Μπορούμε να χρησιμοποιήσουμε αυτόν τον τελεστή σε συνδυασμό με το = , σχηματίζοντας έναν μόνο τελεστή <= , που ονομάζεται μικρότερος ή ίσος, για να ελέγξουμε αν η τιμή είναι μικρότερη ή ίση με την αναμενόμενη.

Στο παρακάτω παράδειγμα αναζητούμε όλα τα προϊόντα με τιμή πώλησης μικρότερη ή ίση με 10:

```
SELECT  
  
    description,  
  
    premature  
  
FROM  
  
    products  
  
WHERE  
  
    pre-sale <= 10
```

Δεδομένου ότι ο πίνακας προϊόντων έχει δύο εγγραφές με τιμή πώλησης μικρότερη ή ίση με 10, θα επιστραφεί μόνο το "Blue Pen".

Μεγαλύτερο τότε

Ο τελεστής > ελέγχει εάν η τιμή που εισάγεται στα αριστερά είναι μεγαλύτερη από αυτή στα δεξιά. Μπορούμε επίσης να συνδυάσουμε τα > και = καταλήγοντας σε >=, το οποίο ελέγχει αν η τιμή είναι μεγαλύτερη ή ίση.

Στο παρακάτω παράδειγμα, αναζητούμε προϊόντα με τιμή πώλησης μεγαλύτερη ή ίση με 40:

```
SELECT
    description,
    premature
FROM
    products
WHERE
    pre-sale >= 40
```

Δεδομένου ότι ο πίνακας προϊόντων έχει μια εγγραφή με τιμή πώλησης μεγαλύτερη ή ίση με 40, "Φορητός φορτιστής", μόνο αυτή η εγγραφή θα επιστραφεί.

METAΞΥ

Μερικές φορές χρειάζεται να αναζητήσουμε εγγραφές σύμφωνα με μια σειρά τιμών. Για αυτό, βασιζόμαστε στον τελεστή BETWEEN, ο οποίος λαμβάνει μια ελάχιστη και μια μέγιστη τιμή και επιστρέφει τα δεδομένα στηλών που πληρούν αυτό το κριτήριο.

Στο παρακάτω παράδειγμα, πραγματοποιούμε αναζήτηση στον πίνακα προϊόντων για εγγραφές με τιμή κόστους μεταξύ 1 και 5.

```
SELECT
    description,
```

premature

FROM

products

WHERE

pre-sale BETWEEN 1 AND 5

Για αυτό το ερώτημα, θα επιστραφεί μόνο το προϊόν "Blue Pen".

Σημείωση: Μπορούμε να χρησιμοποιήσουμε τον τελεστή NOT πριν από το BETWEEN - με τη μορφή NOT BETWEEN - για να επιστρέψουμε εγγραφές που δεν πληρούν την καθορισμένη προϋπόθεση. Εκτελώντας το παραπάνω ερώτημα με το NOT BETWEEN θα είχαμε την αντίθετη συμπεριφορά επιστρέφοντας όλες τις εγγραφές εκτός από το "Blue Pen".

Εκτός από τα εργαλεία που αναφέρθηκαν μέχρι τώρα, έχουμε και άλλα πολύ σημαντικά που επηρεάζουν άμεσα τον χρόνο διαβούλευσης. Επομένως, η γνώση του τρόπου χρήσης τους έχει μεγάλη αξία για έναν καλό επαγγελματία.

Υποερωτήματα

Ένα υποερώτημα (γνωστότερο ως SUBQUERY ή SUBSELECT) είναι μια πρόταση SELECT μέσα σε μια άλλη πρόταση SQL. Επειδή χρησιμοποιούνται ευρέως για την αντικατάσταση του JOIN, όταν δεν λειτουργεί.

Είναι σύνηθες οι προγραμματιστές να επιλέγουν να χρησιμοποιήσουν την ρήτρα JOIN για να ενώσουν περισσότερους από έναν πίνακες στο ίδιο ερώτημα. Ωστόσο, υπάρχουν σενάρια όπου το JOIN δεν θα επιστρέψει το αναμενόμενο αποτέλεσμα. Αυτό το βλέπουμε όταν πρέπει να φέρουμε ως νέα στήλη το αποτέλεσμα των συναρτήσεων συγκέντρωσης, όπως SUM, COUNT ή AVG. Για να λύσετε σενάρια παρόμοια με αυτό, είναι απαραίτητο να γνωρίζετε υποερωτήματα.

Παρακάτω είναι ένα παράδειγμα ΥΠΟΕΡΩΤΗΣΗΣ:

SELECT

```

*
FROM
    table1 AS T
WHERE
    column1 IN
    (
        SELECT
            column2
        FROM
            table2 AS T2
        WHERE
            T.id = T2.id
    )

```

Στο παραπάνω παράδειγμα έχουμε μια πρώτη πρόταση SELECT που εκτελεί ένα φίλτρο μέσω της ρήτρας IN μέσα σε ένα άλλο SELECT, δηλαδή ένα ερώτημα εντός του αποτελέσματος ενός άλλου ερωτήματος.

Πώς να χρησιμοποιήσετε;

Υπάρχουν μερικοί τρόποι χρήσης υποερωτημάτων. Σε αυτό το άρθρο θα καλύψουμε τα ακόλουθα μέσα:

- Υποερώτημα ως νέα στήλη ερωτήματος (ΕΠΙΛΟΓΗ ΩΣ ΠΕΔΙΟ).
- Υποερώτημα ως φίλτρο ερωτήματος (με χρήση τελεστών IN, EXISTS ή σύγκρισης).
- Υποερώτημα ως η πηγή δεδομένων ενός κύριου ερωτήματος (ΕΠΙΛΟΓΗ ΑΠΟ ΕΠΙΛΟΓΗ)

Για χειρισμό δεδομένων στη βάση δεδομένων, έχουμε τις εντολές **INSERT**, **UPDATE** και **DELETE** στη γλώσσα SQL, οι οποίες εισάγουν, ενημερώνουν και αφαιρούν δεδομένα, αντίστοιχα. Ας δούμε, στις παρακάτω αναρτήσεις, πώς να τα χρησιμοποιήσετε:

Εισαγωγή δεδομένων: INSERT SQL

Η εντολή για συμπερίληψη στη βάση δεδομένων είναι INSERT , η οποία έχει την ακόλουθη δομή που εμφανίζεται στη Λίστα 1 .

```
INSERT INTO table_name (field-list)
```

```
VALUES (data_list)
```

--OR

```
INSERT INTO table_name VALUES (data_list)
```

Table_name: όνομα του πίνακα στον οποίο θα εισαχθούν τα δεδομένα.

- Πεδίο-λίστα: όνομα των στηλών που θα λάβουν τις τιμές.
- Λίστα-δεδομένα: τιμές που θα εισαχθούν στον πίνακα. Αυτά τα πεδία πρέπει να είναι με την ίδια σειρά που περιγράφεται στη λίστα πεδίων, χωρισμένα με κόμματα.
- Εάν χρησιμοποιείται μια εντολή SELECT , πρέπει να επιστρέψει τον ίδιο αριθμό στηλών με τους ίδιους τύπους δεδομένων που καθορίζονται στη λίστα πεδίων.

Ενημέρωση δεδομένων: UPDATE SQL

Η εντολή για την ενημέρωση των εγγραφών είναι UPDATE , η οποία έχει την ακόλουθη σύνταξη:

```
UPDATE table_name
```

```
SET FIELD = "new_value"
```

```
WHERE CONDITION
```

- table_name: όνομα του πίνακα που πρόκειται να τροποποιηθεί

- Πεδίο: πεδίο που θα αλλάξει η τιμή του
- New_value: τιμή που θα αντικαταστήσει τα παλιά δεδομένα που έχουν καταχωρηθεί στο πεδίο
- Πού: Εάν δεν ενημερωθεί, ολόκληρος ο πίνακας θα ενημερωθεί
- Συνθήκη: κανόνας που επιβάλλει προϋποθέσεις για την εκτέλεση εντολής

Αφαίρεση δεδομένων: DELETE SQL

Η εντολή που χρησιμοποιείται για τη διαγραφή δεδομένων είναι DELETE.

DELETE FROM table_name

WHERE condition

- table_name: όνομα του πίνακα που πρόκειται να τροποποιηθεί
- Where: ρήτρα που επιβάλλει έναν όρο για την εκτέλεση της εντολής

Μια βάση δεδομένων μπορεί να αποθηκεύσει τιμές, ημερομηνίες και πολλές άλλες πληροφορίες σχετικές με την επιχείρηση. Εφόσον το μικρότερο κομμάτι δεδομένων μπορεί να επηρεάσει μια απόφαση, γιατί να το αποκλείσετε.

Μια απαραίτητη αλλά λιγότερο συχνά εκτελούμενη εργασία είναι η δημιουργία της βάσης δεδομένων. Στη συνέχεια, είναι φυσικό να δημιουργηθούν οι πίνακες που σχετίζονται με αυτό. Για αυτό, ασχολούμαστε με τις εντολές SQL CREATE DATABASE και CREATE TABLE, οι οποίες θα παρουσιαστούν στις ακόλουθες αναρτήσεις, οργανωμένες σύμφωνα με το DBMS:

Η SQL Server Management Studio προσφέρει πολλούς τρόπους εργασίας με πίνακες. Μπορούμε να δημιουργήσουμε νέους πίνακες χρησιμοποιώντας τη δυνατότητα Νέος πίνακας στο SQL Server Management Studio ή την εντολή Δημιουργία πίνακα .

Δημιουργία πινάκων στο SQL Server Management Studio

Πριν δημιουργήσουμε έναν πίνακα, πρέπει να σκεφτούμε προσεκτικά το όνομα του πίνακα, καθώς μπορεί να έχει μήκος έως 128 χαρακτήρες. Τα ονόματα των πινάκων

πρέπει να ξεκινούν με αλφαβητικό χαρακτήρα, αλλά μπορούν επίσης να περιέχουν σύμβολα κάτω παύλες (_), σύμβολα @, σύμβολα λιβρών (#) και αριθμούς. Τα ονόματα των πινάκων πρέπει να είναι μοναδικά για κάθε σχήμα σε μια βάση δεδομένων. Διαφορετικά σχήματα μπορεί να περιέχουν πίνακες με το ίδιο όνομα και κάθε πίνακας μπορεί να έχει έως 1024 στήλες. Τα ονόματα στηλών ακολουθούν τους ίδιους κανόνες ονομασίας με τους πίνακες και πρέπει να είναι μοναδικά.

Στην SQL Server Management Studio δημιουργούμε έναν πίνακα ακολουθώντας τα εξής βήματα:

1. Στο SQL Server Management Studio, συνδεθείτε στην παρουσία διακομιστή που περιέχει τη βάση δεδομένων. Βεβαιωθείτε ότι έχετε δικαίωμα δημιουργίας πίνακα στη βάση δεδομένων και δικαίωμα Αλλαγής στο σχήμα όπου δημιουργείται ο πίνακας.
2. Στην Εξερεύνηση αντικειμένων, αναπτύξτε τον κόμβο Βάσεις δεδομένων και, στη συνέχεια, επιλέξτε μια βάση δεδομένων. Επεκτείνετε ξανά την προβολή για να εμφανίσετε τους κόμβους πόρων σας.
3. Για να δημιουργήσετε έναν νέο πίνακα, κάντε δεξί κλικ στον κόμβο Πίνακες και, στη συνέχεια, επιλέξτε Νέοι πίνακες από το μενού συντόμευσης. Στη συνέχεια, αποκτήστε πρόσβαση στο Table Designer στο SQL Server Management Studio.
4. Ή στο Νέο ερώτημα, εκτελέστε τις ακόλουθες εντολές:

```
Create table city
```

```
(  
    city_id int primary key,  
    city varchar(30),  
    int state  
)
```

```
Create unique index ix_city on city (city)
```

```
create table state
(
    state_id int primary key,
    varchar(2) state
)
```

Create unique index ix_state on state (state)

Create table product

```
(
    product_id int primary key,
    product_name varchar(40),
    numeric product_value (16,2),
    date_purchase datetime
)
```

Alter table product add manufacturer_name int

Create unique index ix_product_name on product(product_name)

Create table manufacturer

```
(
    manufacturer_id int primary key,
    manufacturer_name varchar(40)
)
```

Create unique index ix_manufacturer_name on manufacturer (manufacturer_name)

Εδώ είδαμε πόσο απλό είναι να δημιουργήσετε έναν πίνακα χρησιμοποιώντας το SQL Server Management Studio και εξετάσαμε τις εντολές Create Table και Create Index.

Κατά τη δημιουργία του πρώτου πίνακα, σύντομα θα είμαστε σε επαφή με την έννοια του πρωτεύοντος κλειδιού. Αυτό είναι ένα υποχρεωτικό πεδίο και αντιπροσωπεύει το μοναδικό αναγνωριστικό κάθε εγγραφής που υπάρχει σε έναν πίνακα. Και όταν έχουμε περισσότερα από ένα τραπέζια και πρέπει να δημιουργήσουμε μια σχέση μεταξύ τους, θα συναντήσουμε την έννοια του ξένου κλειδιού.

Πρωτεύων κλειδί

Το πρωτεύον κλειδί ή πρωτεύον κλειδί είναι η πιο βασική έννοια που σχετίζεται με την οργάνωση σε μια βάση δεδομένων. Κάθε τραπέζι θα έχει ένα και μόνο ένα πρωτεύον κλειδί. Αυτό το κλειδί χρησιμοποιείται ως μοναδικό αναγνωριστικό για τον πίνακα, το οποίο αντιπροσωπεύεται από αυτό το πεδίο (ή πεδία) που δεν θα λαμβάνουν επαναλαμβανόμενες τιμές.

Εξαιτίας αυτού, υπάρχει μια λίστα χαρακτηριστικών που πρέπει να ληφθούν υπόψη κατά τον ορισμό ενός πρωτεύοντος κλειδιού:

Τα κύρια κλειδιά δεν μπορούν να είναι μηδενικά.

Κάθε εγγραφή στον πίνακα πρέπει να έχει ένα και μόνο ένα πρωτεύον κλειδί.

Κανονικά, τα πρωτεύοντα κλειδιά προσαυξάνονται αυτόματα από τη βάση δεδομένων, δηλαδή, δεν χρειάζεται να μεταβιβαστεί αυτή η τιμή σε ένα INSERT . Ωστόσο, αυτή είναι μια επιλογή που έχει ρυθμιστεί κατά τη δημιουργία της βάσης δεδομένων που δεν είναι υποχρεωτική. Σε περιπτώσεις που δεν έχει οριστεί (αυτόματη αύξηση), πρέπει να βεβαιωθείτε ότι δεν υπάρχουν επαναλαμβανόμενες τιμές σε αυτήν τη στήλη.

Είναι τα κλειδιά για τη σχέση μεταξύ οντοτήτων ή πινάκων στη βάση δεδομένων. Έτσι, στον σχετικό πίνακα θα υπάρχει μια αναφορά σε αυτό το πρωτεύον κλειδί (το οποίο θα είναι, στον σχετικό πίνακα, το ξένο κλειδί).

Για να δημιουργήσουμε ένα πρωτεύον κλειδί χρειαζόμαστε έναν κωδικό όπως αυτός που εμφανίζεται στην Λίστα 1.

```
CREATE TABLE Person  
  
(  
  
    Person_ID integer PRIMARY KEY AUTOINCREMENT,  
  
    Name varchar(255),  
  
    Address varchar(255),  
  
    City varchar(255)  
  
);
```

Γραμμή 01 : Εντολή SQL για τη δημιουργία του πίνακα Person στη βάση δεδομένων. Όταν εκτελεστεί, θα δημιουργήσει τον πίνακα με τα πεδία που ορίζονται στις γραμμές 03 έως 06.

Γραμμή 03 : ID_Person πεδίο του πίνακα. Δημιουργεί ένα ακέραιο πεδίο (ακέραιος), με περιορισμό ΚΥΡΙΟ ΚΛΕΙΔΙ που υποδεικνύει ότι αυτό το πεδίο είναι το πρωτεύον κλειδί του πίνακα. Σημειώστε επίσης ότι το ορίζουμε ως autoincrement (AUTOINCREMENT).

Γραμμές 04 έως 06: ορισμός των άλλων πεδίων του πίνακα Πρόσωπο.

Σημείωση: οι περιορισμοί εντός της SQL είναι κανόνες/περιορισμοί που ορίζονται για μια στήλη ενός πίνακα βάσης δεδομένων. Μπορούμε επίσης να πούμε ότι αντιπροσωπεύουν ιδιότητες στις οποίες πρέπει να υπακούουν τα δεδομένα σε μια συγκεκριμένη στήλη. Τα κύρια κλειδιά και τα ξένα κλειδιά είναι παραδείγματα περιορισμών.

Αξίζει να σημειωθεί ότι το πρωτεύον κλειδί είναι απαραίτητο για τη λειτουργία της βάσης δεδομένων, αντιπροσωπεύοντας μια ενιαία εγγραφή που διευκολύνει τις

αναζητήσεις και εγγυάται ότι κάθε τιμή εντός του πίνακα θα είναι διαφορετική από την άλλη.

Ξένο κλειδί

Το ξένο κλειδί, είναι μια ελαφρώς διαφορετική έννοια. Δεν πρόκειται συγκεκριμένα για έναν πίνακα, αλλά για μια σχέση μεταξύ πινάκων. Εν συντομία, ένα ξένο κλειδί είναι μια αναφορά σε έναν πίνακα σε ένα πρωτεύον κλειδί σε έναν άλλο πίνακα. Για να γίνει πιο κατανοητό, ας πάρουμε ως παράδειγμα δύο πίνακες: Πρόσωπο και Αυτοκίνητο. Για να δημιουργήσουμε μια σχέση μεταξύ τους, θα μπορούσαμε να έχουμε το πεδίο ID_Person στον πίνακα Car που αναφέρεται στο πρωτεύον κλειδί του πίνακα Person.

Σε αντίθεση με το πρωτεύον κλειδί, το ξένο κλειδί:

Μπορεί να είναι null (NOT NULL);

Είναι ένα πεδίο σε έναν πίνακα που αναφέρεται σε ένα πεδίο που είναι πρωτεύον κλειδί σε έναν άλλο πίνακα.

Είναι δυνατό να υπάρχουν περισσότερα από ένα (ή κανένα) σε έναν πίνακα.

Μια προειδοποίηση: αν και ουσιαστικά δεν υπάρχει πρόβλημα με τα ξένα κλειδιά να δέχονται τη μηδενική τιμή, ένα τέτοιο χαρακτηριστικό μπορεί να δημιουργήσει αυτό που ονομάζεται ορφανή εγγραφή, δηλαδή μια εγγραφή χωρίς δεδομένα για μια δεδομένη σχέση. Για παράδειγμα, ένα άτομο που δεν είναι κάτοχος δικού του αυτοκινήτου. Αν και συνηθισμένο στην πραγματικότητα, είναι απαραίτητο να ληφθεί υπόψη αυτός ο επιχειρηματικός κανόνας στην εφαρμογή για την αποφυγή προβλημάτων.

Η δημιουργία ξένων κλειδιών σε έναν πίνακα γίνεται με δύο τρόπους: Η Λίστα 2 δείχνει την προσθήκη του ξένου κλειδιού απευθείας όταν δημιουργούμε τον πίνακα, ενώ στην Λίστα 3 βλέπουμε τη χρήση της εντολής ALTER TABLE για την εισαγωγή αυτού του περιορισμού σε έναν υπάρχοντα πίνακα.

```
CREATE TABLE Car
```

```
(
```

```

Car_ID integer PRIMARY KEY AUTOINCREMENT,

Name varchar(255),

Tag varchar(255),

ID_Person integer,

CONSTRAINT fk_newCar FOREIGN KEY (ID_Person) REFERENCES Person
(ID_Person)

);

```

- Γραμμή 01 : Εντολή SQL για τη δημιουργία του πίνακα Car στη βάση δεδομένων. Όταν εκτελεστεί, θα δημιουργήσει τον πίνακα με τα πεδία που ορίζονται στις γραμμές 03 έως 07 .
- Γραμμή 03 : δημιουργεί το πρωτεύον κλειδί του πίνακα Car, συμπεριλαμβανομένης της αυτόματης αύξησης, με τον ίδιο τρόπο που είδαμε για τον πίνακα Person.
- Γραμμές 04 και 05 : εισαγωγή των πεδίων Όνομα και Μάρκα αυτοκινήτου στον πίνακα.
- Γραμμή 06 : δημιουργία του πεδίου ID_Person , τύπου integer (integer). Αυτό το πεδίο αντιπροσωπεύει το ξένο κλειδί και, ως εκ τούτου, θα λάβει την τιμή του πεδίου ID_Person (το πρωτεύον κλειδί) του ατόμου «ιδιοκτήτη» του αυτοκινήτου, στη μοντελοποίηση της βάσης δεδομένων του παραδείγματος μας.
- Γραμμή 07 : ορισμός του ίδιου του ξένου κλειδιού. Για να το κάνετε αυτό, παρατηρήστε ότι προσθέσαμε έναν περιορισμό που ονομάζεται fk_newCar (προεπιλεγμένο όνομα: μικτά ονόματα σχετικών πινάκων με το πρόθεμα fk) ως ΞΕΝΟ ΚΛΕΙΔΙ (ξένο κλειδί) και τον συσχετίσαμε στο πεδίο ID_Person του πίνακα Car. Ακόμα στην ίδια γραμμή, ορίζουμε την ίδια την αναφορά (λέξη-κλειδί REFERENCES) στον πίνακα Person, συγκεκριμένα στο πεδίο Person_ID του πίνακα Person.

```
01 CREATE TABLE Car
```

```
02 (
```

```
03 Car_ID integer PRIMARY KEY AUTOINCREMENT,  
04 Name varchar(255),  
05 Mark varchar(255),  
06 Person_ID integer,  
07 );  
08  
09 ALTER TABLE Car  
10 ADD CONSTRAINT fk_newCar FOREIGN KEY (ID_Person) REFERENCES Person  
(ID_Person)
```

Γραμμές 01 έως 07 : εντολή για τη δημιουργία του πίνακα αυτοκινήτου χωρίς να ορίσετε το ξένο κλειδί.

Γραμμές 09 και 10 : εντολή για αλλαγή του πίνακα αυτοκινήτου για προσθήκη περιορισμού ξένου κλειδιού .

Γραμμή 10 : εντολή για προσθήκη του περιορισμού με το όνομα “ fk_newCar , τύπου ΞΕΝΟ ΚΛΕΙΔΙ , στο πεδίο ID_Person του πίνακα Car.

Σημειώστε επίσης την αναφορά (λέξη-κλειδί REFERENCES) στο πεδίο Person_ID του πίνακα Person.

Οι εντολές στις Καταχωρίσεις 2 και 3 κάνουν το ίδιο πράγμα, αλλά με διαφορετικούς τρόπους.

Το ξένο κλειδί που δημιουργείται είναι το ίδιο, όπως και η σχέση μεταξύ των πινάκων Person και Car.

Ένα αξιοσημείωτο σημείο είναι ότι τόσο τα ξένα κλειδιά όσο και τα πρωτεύοντα κλειδιά μπορούν να είναι σύνθετα, δηλαδή να περιλαμβάνουν περισσότερα από ένα πεδία στον πίνακα.

Αυτός ο τύπος καταστάσεων, ωστόσο, δεν είναι πολύ συνηθισμένος, καθώς μειώνει κυρίως την απόδοση της πρόσβασης στα δεδομένα μέσω των εντολών SELECTs.

Ευρετήρια

Ένα ευρετήριο είναι ένας πολύ σημαντικός πόρος όταν πρόκειται για βάσεις δεδομένων. Με έναν απλό τρόπο, μπορούμε να το περιγράψουμε ως μια δομή που στοχεύει στη μείωση του χρόνου που αφιερώνεται σε τραπεζικά ερωτήματα, διευκολύνοντας την αναζήτηση των επιθυμητών εγγραφών.

Για να απλοποιήσουμε την κατανόηση του τι είναι δείκτης, ας κάνουμε έναν παραλληλισμό. Σκεφτείτε δύο βιβλία: ένα με ευρετήριο και ένα χωρίς. Σε ποια θα το βρείτε πιο εύκολα και θα αφιερώσετε λιγότερο χρόνο για να βρείτε το θέμα που θέλετε. Ομοίως, συμβαίνει με πίνακες σε μια βάση δεδομένων.

Βελτίωση απόδοσης με χρήση στατιστικών και δεικτών

Η ευρετηρίαση έχει γίνει βασικό ζήτημα στο τοπίο της βάσης δεδομένων. Ένα υπάρχον πρόβλημα στη βάση δεδομένων αναφέρεται στην ανάγκη να μην ανησυχείτε πολύ για το πώς εκτελούνται τα ερωτήματα SQL και αναφέρονται στο πρόγραμμα βελτιστοποίησης ερωτημάτων, το οποίο πρώτα ελέγχει εάν υπάρχει διαθέσιμο σχέδιο για την εκτέλεσή του.

Εάν όχι, καταρτίζει ένα σχέδιο αυτόματα. Για την αποτελεσματική απόδοση, είναι απαραίτητο να μπορούμε να εκτιμήσουμε τις ενδιάμεσες γραμμές που θα δημιουργηθούν από τις διάφορες στρατηγικές και εναλλακτικές λύσεις για την απόδοση ενός αποτελέσματος.

Η Μηχανή Βάσης Δεδομένων διατηρεί στατιστικά στοιχεία σχετικά με την κατανομή των βασικών τιμών σε κάθε ευρετήριο στον πίνακα και χρησιμοποιεί αυτά τα στατιστικά στοιχεία για να προσδιορίσει ποιους δείκτες θα χρησιμοποιήσει κατά τη σύνταξη του σχεδίου ερωτήματος. Εάν, ωστόσο, υπάρχουν προβλήματα με αυτά τα στατιστικά στοιχεία, η απόδοση του ερωτήματος θα επηρεαστεί.

Εάν δεν υπάρχουν στατιστικά στοιχεία, ο βελτιστοποιητής θα πρέπει να εκτιμήσει τον αριθμό των σειρών αντί να τις εκτιμήσει, και πιστέψτε με: αυτό δεν είναι αυτό που θέλετε και αυτό μπορεί να είναι το πρόβλημα της βραδύτητας στην εκτέλεση ορισμένων ερωτημάτων.

Υπάρχουν διάφοροι τρόποι για να διαπιστώσετε ότι τόσο το εκτιμώμενο όσο και το πραγματικό σχέδιο εκτέλεσης έχουν προβλήματα όταν ο βελτιστοποιητής αντιμετωπίζει έλλειψη στατιστικών στοιχείων.

Σε αυτήν την περίπτωση, θα υπάρχουν προειδοποιήσεις στο σχέδιο, όπως ένα θαυμαστικό ή ένα σφάλμα ένδειξης "x", στο σχέδιο εκτέλεσης και μια προειδοποίηση στις πληροφορίες.

Εάν χρειάζεται να προβάλετε τα σχέδια εκτέλεσης προηγούμενων ερωτημάτων, μπορείτε να ζητήσετε αυτά τα σχέδια χρησιμοποιώντας την εντολή `sys.dm_exec_cached_plans` DMV . Υπάρχουν πολλές περιπτώσεις που χρειάζεται να προβάλετε στατιστικά στοιχεία στο ιστορικό, είτε για να γνωρίζετε τυχόν αλλαγές που έχουν γίνει είτε για παρακολούθηση και παρακολούθηση των διαδικασιών. Πριν προχωρήσετε σε μια εις βάθος συζήτηση, είναι απαραίτητο να έχετε μια ιδέα σχετικά με ορισμένους όρους που χρησιμοποιούνται στην ευρετηρίαση:

- Σάρωση πίνακα: Κάθε φορά που γίνεται επίκληση ενός ερωτήματος σε μια βάση δεδομένων, το εργαλείο βελτιστοποίησης ερωτημάτων SQL προσπαθεί να βρει τον καλύτερο δυνατό τρόπο εκτέλεσης του ερωτήματος προκειμένου να επιτύχει καλύτερη απόδοση.

Σε αυτήν τη διαδικασία, όταν το εργαλείο βελτιστοποίησης ερωτημάτων καθορίσει ότι δεν υπάρχουν διαθέσιμα χρήσιμα ευρετήρια για την παραγωγή του συνόλου αποτελεσμάτων εξόδου, τότε σαρώνει ολόκληρο τον πίνακα. Έτσι, το Table Scan πρέπει να επιθεωρήσει ολόκληρο τον πίνακα, σειρά προς σειρά, προκειμένου να λάβει τα αναμενόμενα αποτελέσματα.

Αυτό είναι πολύ αργό και συνιστάται να αποφεύγεται, καθώς υπάρχουν ορισμένες περιπτώσεις όπου η σάρωση πίνακα είναι ταχύτερη από τη σάρωση ευρετηρίου, για παράδειγμα, σε έναν πίνακα με λίγα δεδομένα.

- Σάρωση ευρετηρίου: βοηθητική δομή δεδομένων που χρησιμοποιείται για την επιτάχυνση της πρόσβασης σε δεδομένα εντός της βάσης δεδομένων. Εδώ, ολόκληρος ο πίνακας δεν θα ανιχνευθεί, αλλά οι σειρές που πληρούν τις προϋποθέσεις για το ερώτημα του αιτούντος θα αναζητηθούν απευθείας.

Για να γίνει αυτό, η δομή δεδομένων διατηρείται χωριστή για να υποστηρίζονται πληροφορίες σχετικά με τις βασικές τιμές του πίνακα.

Η κύρια ιδιότητα του ευρετηρίου είναι ότι είναι διατεταγμένο. Για παράδειγμα, στην περίπτωση ενός λεξικού, όλες οι λέξεις είναι σε σειρά, έτσι ώστε όταν αναζητάτε οποιαδήποτε λέξη, μεταπηδάτε απευθείας στην αντίστοιχη σελίδα που σχετίζεται με τη λέξη. Ομοίως, η μηχανή βάσης δεδομένων αναζητά τις απαιτούμενες σειρές όταν οι αντίστοιχες στήλες είναι ευρετηριασμένες.

Ρύθμιση αυτόματης δημιουργίας στατιστικών ΑΠΕΝΕΡΓΟΠΟΙΗΣΗ

Πρόβλημα

Πώς αυξάνουν τα στατιστικά στοιχεία την απόδοση εκτέλεσης ερωτημάτων του SQL Server;

Λύση

Τα στατιστικά ιστογράμματα χρησιμοποιούνται από το πρόγραμμα βελτιστοποίησης ερωτημάτων για την επιλογή του βέλτιστου σχεδίου εκτέλεσης ερωτημάτων. Εάν ένα ερώτημα περιέχει μια στήλη με στατιστικά στοιχεία, το εργαλείο βελτιστοποίησης μπορεί να εκτιμήσει με ακρίβεια τον αριθμό των σειρών που επηρεάζονται από αυτό το ερώτημα. Έτσι, ο βελτιστοποιητής έχει αρκετές πληροφορίες για να δημιουργήσει το σχέδιο εκτέλεσης. Ο SQL Server δημιουργεί στατιστικά στοιχεία με διάφορους τρόπους:

- Οι στατιστικές δημιουργούνται αυτόματα για κάθε νέο ευρετήριο.
- Εάν η ρύθμιση αυτόματης δημιουργίας στατιστικών στοιχείων της βάσης δεδομένων έχει οριστεί σε ON, τότε ο SQL Server θα δημιουργήσει αυτόματα στατιστικά στοιχεία για στήλες χωρίς ευρετήριο που χρησιμοποιούνται στα ερωτήματά σας.

Χρήση μεταβλητών πίνακα

Πρόβλημα

Για μεταβλητές πίνακα SQL Server, δεν θα διατηρηθούν ποτέ στατιστικά στοιχεία. Να το έχετε υπόψη σας. Κατά την επιλογή από μεταβλητές πίνακα, ο εκτιμώμενος αριθμός σειρών είναι πάντα 1, εκτός εάν ένα κατηγορημα αξιολογείται ως ψευδές και δεν έχει σχέση με τη μεταβλητή του πίνακα.

Σε αυτήν την περίπτωση εφαρμόζεται (όπου WHERE 1 = 0) όταν ο αριθμός σειρών της εκτιμώμενης εκτίμησης είναι 0.

Λύση

Μην υπολογίζετε στη χρήση μεταβλητών πίνακα για προσωρινούς πίνακες, εάν είναι πιθανό να περιέχουν περισσότερες από μερικές σειρές. Κατά γενικό κανόνα, χρησιμοποιήστε προσωρινούς πίνακες (πίνακες με "#" ως πρώτο χαρακτήρα στο όνομά τους), αντί για μεταβλητές πίνακα, για πίνακες temp με περισσότερες από 100 σειρές, καθώς αυτό οδηγεί σε λιγότερη ή καθόλου ανασύνταξη.

Σύνοψη

Σε αυτή την ενότητα ο εκπαιδευόμενος γνωρίζει την θεωρία των βάσεων δεδομένων, ποια βάση δεδομένων είναι πιο σωστή, διαθέτοντας φυσικά συγκεκριμένα κριτήρια αξιολόγησης για βάσεις, ενώ γνωρίζει τα πλεονεκτήματα και τα μειονεκτήματα των πιο δημοφιλών επιλογών, για τις μεθόδους οργάνωσης δεδομένων και συστήματα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ).

Ερωτήσεις ενότητας 5

A. Τι είναι οι βάσεις δεδομένων;

1 Οι Βάσεις Δεδομένων είναι ένα μηχανισμός αποθήκευσης στοιχείων – Δεδομένων (Data) που έχουν σχέση μεταξύ τους και μπορούν να καταγραφούν.

2 Οι Βάσεις Δεδομένων τα σημεία αναζητήσεις πληροφοριών ,συνηθώς είναι ο χώρος του διαδικτύου.

B. Που είναι αποθηκευμένα συνηθώς τα δεδομένα που χρησιμοποιεί ο υπολογιστής στις διάφορες επεξεργασίες;

1. Τα δεδομένα που χρησιμοποιεί ο υπολογιστής στις διάφορες επεξεργασίες είναι συνήθως αποθηκευμένα κεντρική μνήμης.

2. Τα δεδομένα που χρησιμοποιεί ο υπολογιστής στις διάφορες επεξεργασίες είναι συνήθως αποθηκευμένα στις μονάδες βοηθητικής μνήμης.

Γ. Τι είναι ένας χαρακτήρας .

1. Ένα γράμμα
2. Ένας αριθμός
3. Όλα τα παραπάνω

Δ . Τι είναι το Μήκος εγγραφής;

1. Είναι το άθροισμα των μηκών των πεδίων της εγγραφής
2. Είναι ένα ομαδοποιημένο σύνολο πεδίων.

Ε . Ποιος είναι ο ορισμός του αρχείου;

1. Είναι ένα σύνολο ομαδοποιημένων χαρακτήρων.
2. Είναι μια σειρά από ομοειδείς εγγραφές, κάθε εγγραφή, στο ίδιο αρχείο, έχει γενικά την ίδια δομή.

Ζ. Ποιες οι μέθοδοι επεξεργασίας ;

1. Συνηθισμένες μέθοδοι επεξεργασίας είναι η μαζική επεξεργασία (batch processing) ή επεξεργασία κατά ομάδες και η επεξεργασία κινήσεων (transaction processing).
2. Συνηθισμένες μέθοδοι επεξεργασίας είναι η συνολική ή επεξεργασία κατά τμήματα και η επεξεργασία κινήσεων

Η Ποια είναι τα μειονέκτημα που παρουσιάζουν οι βάσεις Δεδομένων;

1. Το λογισμικό είναι συνήθως πολύπλοκο και απαιτείται εκπαίδευση του προσωπικού για να το χρησιμοποιεί αποτελεσματικά.
2. Κίνδυνος πρόσβασης από μη εξουσιοδοτημένα άτομα. Επειδή όλα τα δεδομένα είναι συγκεντρωμένα, κάποια αθέλητη ή σκόπιμη βλάβη στα δεδομένα της Βάσης έχει σοβαρές συνέπειες.

3. Όλα τα παραπάνω

Θ. ποιες περισσότερο γνωστές δομές των Βάσεων Δεδομένων είναι;

1. Η ιεραρχική (Hierarchical)- Η δικτυωτή

2. Η απόλυτη

Ι. Τι θυμίζει η δομή του Ιεραρχικό μοντέλο ((Hierarchical Model));

1. Θυμίζει πυραμίδα

2. Η δομή του θυμίζει δέντρο

Κ. Ποτέ υλοποιήθηκε το πρώτο Δικτυωτό μοντέλο

1. 19980-85

2. 1964-65

Άσκηση

Να φτιάξετε μια βάση δεδομένων για ένα ξενοδοχείο με το όνομα HOTEL και να καταχωρίσετε τα στοιχεία των πελάτων :

1. Όνομα
2. Επώνυμο
3. Τηλέφωνο
4. ΑΡΘ. Ταυτότητας
5. e-mail
6. Ημερομηνία άφιξης
7. Ημερομηνία αναχωρήσεις

Να δημιουργηθεί η αντίστοιχη φόρμα δεδομένων.

Βιβλιογραφία

Ramez Elmasri(2007) Θεμελιώδεις αρχές συστημάτων βάσεων δεδομένων.

Αθανάσιος Σταυρακούδης (2010) Βάσεις δεδομένων και SQL, Μια πρακτική προσέγγιση.

Γκλαβά, Μαρία (2019) Συστήματα βάσεων δεδομένων Θεωρία και πρακτικές εφαρμογές με: MS Access, MS SQL Server, MySQL.

Κωνσταντίνος Θρασύβουλος Τσιάτσος (2015) Εκπαιδευτικά περιβάλλοντα διαδικτύου: Σχεδίαση, ανάπτυξη και αξιολόγηση.

Fowler Adam (2015) NoSQL For Dummies

Εκπαιδευτική Υποενότητα : 6 Σχεδιασμός και ανάπτυξη ιστοτόπων

Σκοπός

Σκοπός της ενότητας αυτής είναι να μάθει ο εκπαιδευόμενος να αναλύει τα θέματα γύρω από τι είναι ο παγκόσμιος ιστός, αλλά και με ποιο τρόπο μπορεί να αναπτύξει και να δημιουργήσει ιστοσελίδες. Μαθαίνει για HTML, CSS και JavaScript τα απαραίτητα εργαλεία για τη δημιουργία δυναμικών, διαδραστικών ιστοσελίδων. Καταλαβαίνει την τριάδα των γλωσσών κωδικοποίησης, έτσι ώστε να μπορεί να δημιουργήσει και να ενημερώσει ιστότοπους

Προσδοκώμενα αποτελέσματα

Στο τέλος αυτής της ενότητας θα είναι σε θέση ο εκπαιδευόμενος να αναπτύσσει ιστοσελίδες, να γνωρίζει για η HTML και η CSS, θα γνωρίσει δια δραστικές ιστοσελίδες Javascript και πιο δυναμικές ιστοσελίδες .

Έννοιες-κλειδιά

Παγκόσμιος Ιστός: World Wide Web σημαίνει στα πορτογαλικά παγκόσμιο δίκτυο, γνωστό και ως Web ή WWW. Ο Παγκόσμιος Ιστός είναι ένα σύστημα εγγράφων υπερμέσων που συνδέονται και εκτελούνται στο Διαδίκτυο. Τα έγγραφα του παγκόσμιου ιστού μπορούν να έχουν τη μορφή βίντεο, ήχων, υπερκειμένων και εικόνων και για την οπτικοποίηση των πληροφοριών, χρησιμοποιείται ένα πρόγραμμα υπολογιστή που ονομάζεται πρόγραμμα περιήγησης για τη λήψη αυτών των πληροφοριών και την εμφάνιση τους στην οθόνη του χρήστη.

web design: Το Web Design περιλαμβάνει εργασίες που σχετίζονται με τη διάταξη και το σχεδιασμό διαδικτυακών σελίδων καθώς και την παραγωγή περιεχομένου, αν και τείνει να εφαρμόζεται περισσότερο στη δημιουργία ιστοσελίδων

HTML : Για να ορίσετε τι είναι HTML, πρέπει πρώτα να κατανοήσετε το ακρωνύμιο. Αντιπροσωπεύει Hyper Text Markup Language, δηλαδή γλώσσα σήμανσης υπερκειμένου.

Είναι μια γλώσσα σήμανσης για την ανάπτυξη ηλεκτρονικών σελίδων και εγγράφων για το Διαδίκτυο, παροχή πληροφοριών σε χρήστες, προγράμματα περιήγησης και μηχανές αναζήτησης.

CSS: Η CSS είναι μια γλώσσα που χρησιμοποιείται για να ορίσει πώς θα πρέπει να παρουσιάζονται έγγραφα γραμμένα σε γλώσσα σήμανσης (HTML ή XML) όσον αφορά τη μορφοποίηση και τη διάταξη. Σε ένα ιδανικό σενάριο, ενώ η HTML χρησιμοποιείται για τη δομή του περιεχομένου, το CSS χρησιμοποιείται για τη μορφοποίησή του. Με αυτόν τον τρόπο υπάρχει μια τεράστια ενοποίηση μεταξύ HTML και CSS .

Διαδραστικές Ιστοσελίδες: Ένας διαδραστικός ισότοπος είναι μια ιστοσελίδα που χρησιμοποιεί διάφορα λογισμικά για να δημιουργήσει μια διαδραστική εμπειρία που επιτρέπει στο άτομο που βλέπει τη σελίδα να αλληλοεπιδρά ενεργά με τον ισότοπο. Αυτό μπορεί να γίνει για διάφορους λόγους και χρησιμοποιώντας διάφορες μεθόδους και λογισμικό για να επιτευχθεί αυτή η διαδραστικότητα. Ένας διαδραστικός ισότοπος μπορεί να είναι οποιοσδήποτε ισότοπος που επιτρέπει στους χρήστες να υπερβαίνουν την απλή ανάγνωση κειμένου και την προβολή εικόνων.

Υπερσύνδεσμοι: Οι υπερσύνδεσμοι, ή απλά οι σύνδεσμοι, είναι η ψυχή μιας ιστοσελίδας. Είναι αυτά που επιτρέπουν την εύκολη πρόσβαση μεταξύ διαφορετικών σελίδων (πλοήγηση) και ακόμη και τη γρήγορη μετακίνηση μέσα σε ένα μεγάλο κείμενο. Μπορούμε να πούμε ότι ο Παγκόσμιος Ιστός (WWW) , το πιο γνωστό κομμάτι του διαδικτύου, δεν θα υπήρχε χωρίς αυτά. Στην πραγματικότητα, ο υπερσύνδεσμος δεν είναι τίποτα άλλο από μια εντολή όπως το εικονίδιο που πατιέται για να ξεκινήσει ένα πρόγραμμα ή να ανοίξει ένα αρχείο. Αλλά εδώ ανοίγει μια ιστοσελίδα, ή σας στέλνει το άλλο μέρος του κειμένου, ή ακόμα και σας επιτρέπει να αντιγράψετε ένα αρχείο, όλα αυτά σε άλλο φάκελο, υπολογιστή ή ακόμα και σε άλλο μέρος του κόσμου.

Javascript: Η JavaScript είναι μια δημοφιλής γλώσσα δέσμης ενεργειών για την προσθήκη διαδραστικών λειτουργιών και άλλου δυναμικού περιεχομένου Ιστού σε

ιστοσελίδες. Γνωστά παραδείγματα περιεχομένου JavaScript περιλαμβάνουν φόρμες με δυνατότητα συμπλήρωσης, προβολές διαφανειών γκαλερί φωτογραφιών και κινούμενα γραφικά.

jQuery: Δημιουργήθηκε από τον John Resig το 2006 με σκοπό να διευκολύνει τη ζωή των προγραμματιστών που χρησιμοποιούν JavaScript στους ιστότοπούς τους. Δεν είναι ξεχωριστή γλώσσα προγραμματισμού, λειτουργεί μαζί με JavaScript. Με το jQuery, θα μπορείτε να κάνετε πολλά περισσότερα με λιγότερα – θα σας εξηγήσω αργότερα.

PHP: είναι μια γλώσσα αντικειμενοστρεφής προγραμματισμού που μπορείς να φτιάχνει ιστοσελίδες και συνεργάζεται με την βάση δεδομένων MySQL

MySQL: μια γλώσσα προγραμματισμού για συστήματα διαχειρίσεις βάσεων δεδομένων (ΣΔΒΔ- DBMS) συνεργάζεται με την γλώσσα PHP

Εκπαιδευτική υποενότητα : 6.1 Παγκόσμιος Ιστός **Ο ορισμός**

Ο παγκόσμιος ιστός είναι ήδη τόσο ριζωμένος στη ζωή μας που συχνά δεν αμφισβητείται καν την προέλευσή του. Ωστόσο, κάθε φορά που γίνεται πρόσβαση σε έναν ιστότοπο προκειμένου να διαβασθούν ειδήσεις, να αναζητηθούν δεδομένα για εργασία ή να πραγματοποιηθεί αγορά από το ηλεκτρονικό εμπόριο, χρησιμοποιείται ο Παγκόσμιος Ιστός. Το «www» που είναι μέρος της καθημερινότητάς μας προέκυψε για να εκδημοκρατίσει την πρόσβαση σε πληροφορίες. Ακόμη και στην άλλη άκρη του κόσμου, είναι δυνατόν να υπάρχει πρόσβαση σε μια σελίδα εδώ στη Βραζιλία σε λίγα χιλιοστά του δευτερολέπτου. Ωστόσο, παρά το γεγονός ότι έχει συντομεύσει τις αποστάσεις και επιταχύνει τις πληροφορίες, ο Ιστός άνοιξε επίσης την πόρτα στην παραπληροφόρηση, τον εξτρεμισμό και την κλοπή δεδομένων .

Το WorldWideWeb ή WWW ήταν το πρώτο web browser, ήταν το μοναδικό πρόγραμμα για την περιήγηση στον παγκόσμιο ιστό.

Τι είναι ο Παγκόσμιος Ιστός; (World Wide Web)

Ο Παγκόσμιος Ιστός είναι η διασύνδεση ενός μεγάλου συνόλου δεδομένων και εγγράφων που είναι προσβάσιμα μέσω του Διαδικτύου οπουδήποτε στον κόσμο. Αυτά τα έγγραφα μπορεί να έχουν κείμενα, εικόνες, ήχους και βίντεο που, μαζί, αποτελούν τις ιστοσελίδες. Ωστόσο, εάν ήταν απομονωμένες, αυτές οι σελίδες δεν θα σχημάτιζαν δίκτυο. Επομένως, ένα από τα κύρια στοιχεία του ιστού είναι ο υπερσύνδεσμος, ή απλώς ένας σύνδεσμος, που κάνει τη σύνδεση μεταξύ σελίδων στον ίδιο ιστότοπο και σε άλλους ιστότοπους . Είναι μέσω συνδέσμων που οι άνθρωποι πλοηγούνται από τη μια σελίδα στην άλλη και έχουν πρόσβαση σε πληροφορίες. Για την απόκτηση πρόσβασης στις σελίδες και την περιήγηση στους συνδέσμους Ιστού, πρέπει να χρησιμοποιηθεί ένα πρόγραμμα περιήγησης, όπως το τρέχον Google Chrome και το Microsoft Edge. Κατανοούν τα πρωτόκολλα του Παγκόσμιου Ιστού και μετατρέπουν τα δεδομένα σε πληροφορίες οργανωμένες στις σελίδες για τον χρήστη.

Σε αυτή την βάση πάνω , δημιουργήθηκε ο Παγκόσμιος Ιστός (world wide web), που είναι πιο γνωστό με τη συντομογραφία WWW , το οποίο προσφέρει ένα εύχρηστο



και περιβάλλον στον χρήστη, αλλά και φιλικό προς αυτόν προκειμένου να διαχειριστεί την επιθυμητή πληροφορία αναζήτησης του . Το WWW, χρησιμοποιεί τα πολύ δημοφιλή προγράμματα πλοήγησης στο διαδίκτυο, όπως το Netscape και το Explorer.

Η ιστορία πίσω από τον Παγκόσμιο Ιστό

Το 1989 δημιουργήθηκε η τεχνολογία του ιστού. Ο Βρετανός Τιμ Μπέρνερς Λη, ο οποίος εκείνη την εποχή εργαζόταν στον Ευρωπαϊκό Οργανισμό Πυρηνικών Ερευνών (CERN) στην Γενεύη της Ελβετίας έδωσε στην εφεύρεσή του ο ίδιος ο Λη το όνομα **World Wide Web**, όρος γνωστός στους περισσότερους από το "www". Αυτό που οδήγησε τον Λη στην εφεύρεση του Παγκόσμιου ιστού ήταν το όνειρο του για ένα κόσμο όπου ο καθένας θα μπορούσε να ανταλλάσσει πληροφορίες και ιδέες άμεσα προσβάσιμες όλους .Ο βασικός του σκοπός ήταν να προσφέρει στους επιστήμονες ένα μέσο, προκειμένου να ανταλλάσσουν στοιχεία, επιστημονικά δεδομένα και μελέτες, μέσω του διαδικτύου. Η προσπάθεια του αυτή στηρίχτηκε

στην υλοποίηση ενός μέσου που θα επιτρέπει την σύνδεση μεταξύ διαφορετικών εγγράφων. Το μέσο αυτό ονομάστηκε, υπερκείμενο(Hyper text). Το 1994 κατασκευάστηκε το πρόγραμμα Mosaic, το οποίο έδινε την δυνατότητα στους χρήστες να χρησιμοποιούν , να διαβάζουν και να βλέπουν ,τις πληροφορίες που ήταν διαθέσιμες στο WWW με γραφικό τρόπο. Το WWW στα επόμενα χρόνια, έγινε το απόλυτο μέσο αναζήτησης πληροφοριών για εκατομμύρια χρήστες του διαδικτύου.

Ποιες είναι οι διαφορές μεταξύ του Διαδικτύου και του Ιστού;

Διαδίκτυο και Ιστός είναι όροι που συγχέονται και χρησιμοποιούνται συχνά ως συνώνυμα. Ωστόσο, είναι διαφορετικά. Όπως θα γίνει αναφορά αργότερα, το διαδίκτυο εμφανίστηκε πολύ νωρίτερα ως τρόπος διασύνδεσης υπολογιστών, αλλά ήταν ο ιστός που έκανε δημοφιλή τη χρήση του. Το Διαδίκτυο είναι ένα παγκόσμιο σύστημα πληροφοριών που συνδέει υπολογιστές και περιλαμβάνει πολλούς πόρους και εφαρμογές επικοινωνίας. Ο Ιστός είναι ένα σύνολο πρωτοκόλλων για την πρόσβαση σε πληροφορίες σε σελίδες και έγγραφα μέσω του Διαδικτύου . Είναι ακριβώς η συνδεσιμότητα του διαδικτύου που επιτρέπει τη συμμετοχή ανθρώπων και διακομιστών σε αυτό το μεγάλο δίκτυο σε όλο τον κόσμο.

Το WWW, το οποίο αποτελεί μέρος των ιστοσελίδων που επισκέπτεται ο κόσμος καθημερινά, είναι μόνο ένας από τους τρόπους αλληλεπίδρασης στο διαδίκτυο. Αλλά γίνεται επίσης να υπάρχει πρόσβαση σε email και εφαρμογές χωρίς την χρήση αυτού του πρωτόκολλου, για παράδειγμα.

Πώς λειτουργεί η αρχιτεκτονική του Διαδικτύου;

Το Διαδίκτυο συνδέει πολλούς υπολογιστές σε ένα δίκτυο. Για αυτό, η αρχιτεκτονική του διαδικτύου είναι οργανωμένη σε τέσσερα επίπεδα πρωτοκόλλων που κάνουν την επικοινωνία μεταξύ των υπολογιστών. Παρακάτω θα γίνει η αναφορά τους.

Στρώμα εφαρμογής

Το υψηλότερο είναι το επίπεδο εφαρμογής, μέσω του οποίου ο χρήστης εισέρχεται στο διαδίκτυο. Εάν το άτομο θέλει να χρησιμοποιήσει τον Ιστό, για παράδειγμα, χρησιμοποιεί το πρωτόκολλο HTTP. Εάν πρέπει να χρησιμοποιηθεί email, πρέπει να γίνει χρήση του πρωτόκολλου Simple Mail Transfer Protocol. Εάν πρέπει να μεταφερθούν αρχεία, χρησιμοποιείται το FTP

Στρώμα μεταφοράς

Στη συνέχεια έρχεται το επίπεδο μεταφοράς, το οποίο μεταδίδει τα δεδομένα στο δίκτυο. Για αυτό, το κύριο πρωτόκολλο που χρησιμοποιείται είναι το TCP (Protocol Control Control). Αυτό το πρωτόκολλο αποσυνθέτει τα μηνύματα σε πακέτα και τα αποστέλλει σε επίπεδο δικτύου.

Επίπεδο δικτύου

Στο επίπεδο δικτύου, το πρωτόκολλο Internet Protocol (IP) επικοινωνεί μεταξύ των διακομιστών για τη μετάδοση πακέτων δεδομένων. Εάν ο υπολογιστής δεν βρίσκεται στο ίδιο δίκτυο, είναι οι δρομολογητές που μεταφέρουν τα δεδομένα. Για την αναγνώριση των μηχανημάτων και των δικτύων στα οποία βρίσκονται, χρησιμοποιείται ένα αναγνωριστικό, που ονομάζεται διεύθυνση IP.

Φυσική στρώση

Τέλος, το φυσικό επίπεδο είναι υπεύθυνο για την αναγνώριση της φυσικής σύνδεσης πάνω από την οποία ταξιδεύουν τα πακέτα δεδομένων. Στη συνέχεια, μεταφέρονται μεταξύ υπολογιστών ή δρομολογητών που είναι συνδεδεμένοι μέσω ενός τοπικού δικτύου, όπως ένα δίκτυο Ethernet ή ασύρματο (Wi-Fi).

Με αυτόν τον τρόπο, λοιπόν, τα δεδομένα κυκλοφορούν στο διαδίκτυο σε όλο τον κόσμο.

Ποια είναι τα 10 κύρια στοιχεία του Παγκόσμιου Ιστού;

Ο παγκόσμιος ιστός λειτουργεί μέσω μιας σειράς στοιχείων και πρωτοκόλλων. Πολλά από αυτά αποτελούν μέρος της καθημερινότητάς — και τώρα θα γίνει αντιληπτό τι είναι και πώς λειτουργούν:

1. HTTP

Το HTTP σημαίνει Πρωτόκολλο μεταφοράς υπερκειμένου . Είναι το βασικό πρωτόκολλο του Παγκόσμιου Ιστού , που βασίζεται σε έγγραφα που συνδέονται με υπερκείμενο . Όχι τυχαία, υπάρχει σε όλες τις διευθύνσεις ιστού. Όταν κάποιος πληκτρολογεί HTTP, το πρόγραμμα περιήγησης προσδιορίζει ήδη ποιο πρωτόκολλο πρέπει να ακολουθήσει.

Επί του παρόντος, υπάρχει επίσης το HTTPS (HyperText Transfer Protocol Secure), το οποίο είναι το ίδιο με το HTTP, αλλά προσθέτει ένα επίπεδο ασφάλειας στο πρωτόκολλο.

2. URL

Το URL είναι το ακρωνύμιο του Uniform Resource Locator . Είναι μια διεύθυνση που προσδιορίζει πού βρίσκονται σελίδες, έγγραφα και αρχεία στον παγκόσμιο ιστό . Στην καρτέλα του προγράμματος περιήγησης, ενώ το HTTP υποδεικνύει το πρωτόκολλο που πρέπει να ακολουθηθεί, το URL υποδεικνύει τη διαδρομή.

3. Τομέας

Ο τομέας είναι το κύριο URL ενός ιστότοπου, το οποίο χρησιμεύει για την αναγνώρισή του στον Ιστό και τη διευκόλυνση της απομνημόνευσής του.

Ο τομέας χωρίζεται με κουκκίδες, που συνήθως σχηματίζονται από το όνομα του ιστότοπου, ακολουθούμενο από τον τύπο του ιστότοπου (Τομέας ανώτατου επιπέδου ή TLD) και η χώρα προέλευσής του (Τομέας επιπέδου κωδικού χώρας ή ccTLD). Στο `www.example.com.br`, "παράδειγμα" είναι το όνομα τομέα, ".com" είναι το TLD και ".br" είναι το ccTLD.

Οι καταλήξεις τομέα είναι πλέον πιο ευέλικτες. Τα παραδοσιακά TLD (π.χ. .com, .net, .org, .gov) επιτρέπουν πλέον μια ποικιλία επιλογών (π.χ. .cafe, .design, .club).

4. HTML

Το HTML είναι το ακρωνύμιο του όρου HyperText Markup Language , το οποίο σημαίνει γλώσσα σήμανσης υπερκειμένου. Είναι μια από τις κύριες γλώσσες του

ιστού , η οποία χρησιμεύει στον καθορισμό της σημασίας και της δομής μιας σελίδας, έτσι ώστε τα προγράμματα περιήγησης να μπορούν να κατανοήσουν και να αποκωδικοποιήσουν το περιεχόμενό της .

5. Πρωτόκολλο TCP/IP

Το πρωτόκολλο TCP/IP είναι ένα σύνολο βασικών πρωτοκόλλων Διαδικτύου, το οποίο διασφαλίζει την επικοινωνία δεδομένων μεταξύ ενός υπολογιστή και ενός άλλου . Το TCP (Transport Control Protocol) βρίσκεται στο επίπεδο μεταφοράς δεδομένων, ενώ το IP (Internet Protocol) βρίσκεται στο επίπεδο δικτύου και προσδιορίζει τη διεύθυνση των υπολογιστών.

6. Διακομιστής

Ένας διακομιστής είναι ένας υπολογιστής που περιέχει έναν ή περισσότερους ισχυρούς επεξεργαστές και παρέχει υπηρεσίες σε ένα δίκτυο υπολογιστών , που ονομάζεται πελάτες. Ο διακομιστής μπορεί να αποθηκεύει δεδομένα , να επεξεργάζεται εφαρμογές και να εκτελεί προγράμματα σε φυσικά ή εικονικά συστήματα. Όταν ένα άτομο αποκτά πρόσβαση σε έναν ιστότοπο, όλα τα δεδομένα και τα αιτήματα περνούν από τους διακομιστές.

7. Φιλοξενία

Η φιλοξενία είναι ο χώρος στους διακομιστές που αποθηκεύει τα δεδομένα και τα έγγραφα που συνθέτουν έναν ιστότοπο . Μια υπηρεσία φιλοξενίας είναι υπεύθυνη για αυτήν την αποθήκευση και τη διαθεσιμότητα του ιστότοπου 24 ώρες την ημέρα, οπουδήποτε στον κόσμο.

8. DNS

Το DNS είναι το ακρωνύμιο του Domain Name System ή Domain Name System. Είναι ένα σύστημα διακομιστών που είναι υπεύθυνο για τον εντοπισμό και τη μετάφραση των διευθύνσεων URL που πληκτρολογούνται στα προγράμματα περιήγησης σε αντίστοιχες διευθύνσεις IP . Με αυτόν τον τρόπο, ο χρήστης δεν χρειάζεται να εισάγει έναν κωδικό αριθμών και κουκκίδων για να ενεργοποιήσει μια ιστοσελίδα.

9. Υπερσύνδεσμος

Ένας υπερσύνδεσμος, που ονομάζεται επίσης απλά σύνδεσμος, είναι μια αναφορά στον κώδικα μιας σελίδας που συνδέεται με άλλες ιστοσελίδες και έγγραφα . Είναι μέσω των υπερσυνδέσμων που ο Ιστός συγκροτείται ως ένα μεγάλο δίκτυο μέσω του οποίου οι άνθρωποι πλοηγούνται για πρόσβαση σε πληροφορίες.

10. W3C

Το W3C (World Wide Web Consortium) είναι μια διεθνής μη κερδοσκοπική κοινοπραξία που ρυθμίζει πρότυπα ιστού , όπως οδηγίες σχεδίασης, αρχιτεκτονικής και σημασιολογίας. Η αποστολή του είναι να αξιοποιήσει το διαδίκτυο στο μέγιστο των δυνατοτήτων του.

Η κοινοπραξία ιδρύθηκε και επί του παρόντος διευθύνεται από τον Tim Berners-Lee, τον δημιουργό του Ιστού. Υποστηρίζεται από μεγάλες εταιρείες, οργανισμούς και κυβερνητικά ιδρύματα σε όλο τον κόσμο, συμπεριλαμβανομένων: Google , Amazon , Apple, Microsoft, BBC και Adobe.

Εκλαϊκευση και εξέλιξη του παγκόσμιου ιστού.

Σε όλη αυτή την εξέλιξη , ο Παγκόσμιος Ιστός έχει περάσει από διάφορες φάσεις καθώς η τεχνολογία και η συμπεριφορά της κοινωνίας έχουν αλλάξει. Στη συνέχεια, ας δούμε τι σημείωσε τα βήματα ιστού:

Web 1.0

Το Web 1.0 σηματοδοτεί την πρώτη φάση του World Wide Web , τα πρώτα χρόνια μετά τη δημιουργία του το 1989.

Στην αρχή, οι σελίδες ήταν απλές και στατικές, χωρίς καμία αλληλεπίδραση με τους χρήστες . Η πρόθεση ήταν απλώς να γίνει διαθέσιμο περιεχόμενο, συνήθως σε ιστότοπους εταιρειών, πύλες ειδήσεων και επιχειρηματικούς καταλόγους, κάτι που βοήθησε στη διάδοση πληροφοριών.

Web 2.0

Το Web 2.0 αρχίζει να προσεγγίζει το σενάριο που έχουμε σήμερα. Αυτή η φάση χαρακτηρίζεται από συνεργασία και διαδραστικότητα — από εκεί, οι άνθρωποι

μπορούν να παράγουν το δικό τους περιεχόμενο , σελίδες προγραμμάτων και ιστολόγια, να σχολιάζουν και να συμμετέχουν.

Ένα από τα ορόσημα του Web 2.0 είναι η δημιουργία της Wikipedia , μιας μεγάλης συλλογικής εγκυκλοπαίδειας στον Ιστό. Τα ιστολόγια και τα κοινωνικά δίκτυα ενίσχυσαν επίσης την αλληλεπίδραση των χρηστών. Για τις εταιρείες και τις επωνυμίες, αυτή η φάση σηματοδότησε μια μεταμόρφωση στη σχέση με τους καταναλωτές, καθώς μπορούσαν να βασίζονται στα σχόλιά τους (τόσο θετικά όσο και αρνητικά).

Web 3.0

Το Web 3.0 ονομάζεται Σημασιολογικός Ιστός. Εκτός από τη συνεργασία και τη διαδραστικότητα της προηγούμενης φάσης, ο Ιστός βασίζεται πλέον και στην τεχνητή νοημοσύνη . Από εκεί, η αλληλεπίδραση μεταξύ ανθρώπων και μηχανών μεταμορφώνει επίσης τη σχέση με τον Ιστό .

Μέσω της τεχνολογίας επεξεργασίας φυσικής γλώσσας (NLP), για παράδειγμα, η Google κατανοεί τι γράφουν οι άνθρωποι και μπορεί να κάνει την εμπειρία αναζήτησης πολύ πιο αποτελεσματική. Μέσω της μηχανικής μάθησης (Machine Learning), τα εικονικά καταστήματα μπορούν να μάθουν για τα ενδιαφέροντα και τις συμπεριφορές των πελατών και να εξατομικεύσουν τις προτάσεις προϊόντων.

Ένα από τα θεμέλια του Web 3.0 είναι τα Big Data και η χρήση δεδομένων χρήστη . Επομένως, αυτή η φάση φέρνει επίσης ζητήματα ελέγχου της ιδιωτικής ζωής των προσωπικών δεδομένων και της ασφάλειας στον κυβερνοχώρο .

Web 4.0

Το Web 4.0, η πιο πρόσφατη φάση που βρίσκεται ακόμη σε μετάβαση, είναι το Symbiotic Web. Εκτός από το να φέρει τα χαρακτηριστικά των προηγούμενων φάσεων, τώρα ο Ιστός ενσωματώνεται και στην καθημερινότητά μας, στα αντικείμενα, στα πράγματα . Όχι τυχαία, η κύρια ιδέα αυτής της φάσης είναι το Διαδίκτυο των Πραγμάτων .

Σε αυτό το σημείο, δεν υπάρχει πλέον διαχωρισμός μεταξύ online και offline. Οι συσκευές συνδέονται στο διαδίκτυο και εκτελούν εργασίες χωρίς ανθρώπινη

παρέμβαση, για να κάνουν την καθημερινή ζωή πιο έξυπνη — ή έξυπνη . Υπό αυτή την έννοια, προέρχονται από έξυπνες τηλεοράσεις, έξυπνα σπίτια έως έξυπνες πόλεις, συνδεδεμένες και έξυπνες πόλεις, με σκοπό τη βελτίωση της ζωής των ανθρώπων.

Σε ποιον ανήκει ο παγκόσμιος ιστός;

Αλλά, τελικά, σε ποιον ανήκουν όλα; Ποιος φροντίζει το διαδίκτυο; Ποιος πληρώνεται για τη χρήση του ιστού παγκοσμίως;

Ο Tim Berners-Lee δεν είναι ιδιοκτήτης του ιστού — είναι απλώς ο δημιουργός του και διευθυντής της κοινοπραξίας που παρουσιάσαμε νωρίτερα . Παρεμπιπτόντως, το W3C και κανένα από τα μέλη του δεν κατέχουν ούτε το διαδίκτυο. Απλώς ρυθμίζουν πρότυπα για να δημιουργήσουν μια πλουσιότερη και πιο αποτελεσματική πλατφόρμα για τους ανθρώπους. Από την αρχή, ο Παγκόσμιος Ιστός δημιουργήθηκε για να είναι ένα ανοιχτό δίκτυο . Έτσι, μετά την αρχική του πρόταση, ο παγκόσμιος ιστός μέχρι σήμερα δεν έχει ιδιοκτήτη . Οποιοσδήποτε μπορεί να έχει πρόσβαση σε πληροφορίες, να δημιουργήσει σελίδες και ακόμη και να αναπτύξει γλώσσες για τον Ιστό. Ωστόσο, για αυτό, είναι απαραίτητη η πρόσβαση στο Διαδίκτυο και στις συνδεδεμένες συσκευές. Τότε είναι που ο Ιστός γίνεται απροσπέλαστος σε μεγάλο αριθμό ατόμων, που δεν μπορούν να αντέξουν οικονομικά προγράμματα δεδομένων, υπολογιστές ή κινητά τηλέφωνα.

Έτσι, όπως προτείνει ο Tim Berners-Lee, αυτή είναι μια από τις κύριες προκλήσεις που αντιμετωπίζει ο ιστός σήμερα. Η προσέγγιση ολόκληρου του παγκόσμιου πληθυσμού, προκειμένου να εκδημοκρατιστεί η πρόσβαση στις πληροφορίες, ήταν —και συνεχίζει να είναι— ο στόχος του δημιουργού του Παγκόσμιου Ιστού τον περασμένο αιώνα.

Τώρα, εάν θέλει επίσης ο οποιοσδήποτε να συμβάλει στη βελτίωση του Παγκόσμιου Ιστού , ίσως θέλει να ακολουθήσει μια καριέρα ως προγραμματιστής ιστού.

Παγκόσμιος Ιστός και Διαδίκτυο

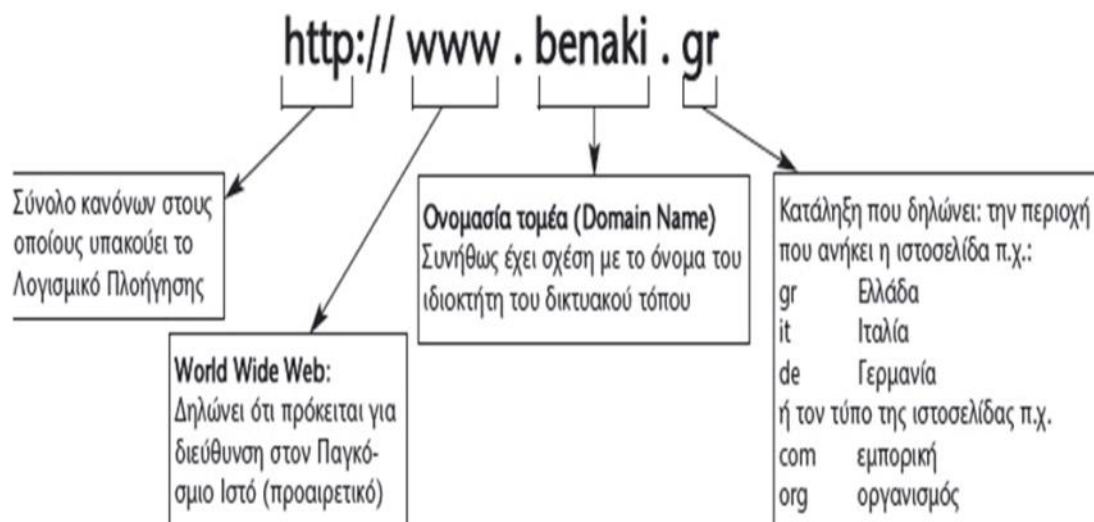
Ένα λάθος που γίνεται λόγω της μεγάλης απήχησης του Παγκόσμιου Ιστού, είναι ότι πολλές φορές ταυτίζεται ο Παγκόσμιος Ιστός με το Διαδίκτυο. Αυτό είναι λάθος γιατί ο Παγκόσμιος Ιστός, είναι μία τεράστια βιβλιοθήκη μια συλλογή από από ψηφιακά έγγραφα τις ιστοσελίδες. Όλα αυτά τα έγγραφα είναι αποθηκευμένα σε υπολογιστές του Διαδικτύου. Το Διαδίκτυο, είναι ένα Παγκόσμιο Δίκτυο Υπολογιστών συνδεδεμένων μεταξύ τους.

Μία ιστοσελίδα (web page) περιλαμβάνει κείμενο, εικόνες, ήχο ή βίντεο κ.τ.λ. Για παράδειγμα , ένα φορτίο μεταφέρεται με κάποιο φορτηγό σε έναν αυτοκινητόδρομο, έτσι ακριβώς και μία ιστοσελίδα του Παγκόσμιου Ιστού, μεταφέρεται στο Διαδίκτυο. Φεύγει από κάποιον κεντρικό υπολογιστή, που είναι αποθηκευμένη, και έρχεται στον υπολογιστή μας. Φυσικά οι περισσότερες πληροφορίες στον Παγκόσμιο Ιστό είναι δωρεάν.



Για παράδειγμα πολλές δημόσιες υπηρεσίες, πανεπιστήμια, βιβλιοθήκες, ιδιωτικοί οργανισμοί διατηρούν σε έναν υπολογιστή του Διαδικτύου μια συλλογή από ιστοσελίδες (έναν δικτυακό τόπο – web site), που μπορεί να επισκεφθεί κάποιος, οποιαδήποτε στιγμή .

Κάθε ιστοσελίδα που περιέχεται σε δικτυακό τόπο, έχει τη δική της διεύθυνση στον Παγκόσμιο Ιστό, όπως έχει ο κάθε άνθρωπος τη δική του διεύθυνση κατοικίας. Όταν καλεί κάποιος ένα φίλο στο σπίτι του για πρώτη φορά, του δίνει τη διεύθυνσή , ώστε να μπορέσει να βρει τη κατοικία. Όμοια, αν θέλει κάποιος να «επισκεφτεί» μία ιστοσελίδα, πρέπει να ξέρει τη διεύθυνσή της. Η διεύθυνση αυτή καλείται URL



(Uniform Resource Locator) – Ενιαίος Προσδιοριστής Πόρου) ή διεύθυνση ιστοσελίδας. Μία διεύθυνση ιστοσελίδας είναι μοναδική και έχει συνήθως την εξής μορφή:

Πλοήγηση στον Παγκόσμιο Ιστό

Η πρόσβαση στην υπηρεσία WWW γίνεται μέσω ενός προγράμματος-πελάτη, που ονομάζεται πρόγραμμα περιήγησης ιστού, επειδή επιτρέπει στους χρήστες να περιφέρονται σύμφωνα με συνδέσμους HyperText χωρίς να χρειάζεται να κάνουν σκόπιμα ερωτήματα.

Το πρόγραμμα περιήγησης είναι ένα λογισμικό που μπορεί να εμφανίσει τα περιεχόμενα των αρχείων HTML σε έναν διακομιστή web ή σύστημα αρχείων και να επιτρέπει στους χρήστες να αλληλοεπιδρούν με αυτά τα αρχεία.

Το πρόγραμμα περιήγησης ιστού αλληλοεπιδρά κυρίως με τον διακομιστή ιστού μέσω του πρωτοκόλλου HTTP και αποκτά ιστοσελίδες. Αυτές οι ιστοσελίδες καθορίζονται από διευθύνσεις URL και η μορφή αρχείου είναι συνήθως HTML και καθορίζεται από το MIME στο πρωτόκολλο HTTP.

Μια ιστοσελίδα μπορεί να περιλαμβάνει πολλά έγγραφα και κάθε έγγραφο λαμβάνεται από τον διακομιστή ξεχωριστά. Τα περισσότερα προγράμματα περιήγησης υποστηρίζουν εγγενώς ένα ευρύ φάσμα μορφών εκτός της HTML, όπως JPEG, PNG, GIF και άλλες μορφές εικόνας, και μπορούν να επεκταθούν για να υποστηρίξουν πολλά πρόσθετα (plug-ins).

Επιπλέον, πολλά προγράμματα περιήγησης υποστηρίζουν επίσης άλλους τύπους URL και τα αντίστοιχα πρωτόκολλα τους, όπως FTP, Gopher, HTTPS (μια κρυπτογραφημένη έκδοση του πρωτοκόλλου HTTP).

Οι προδιαγραφές τύπου περιεχομένου HTTP και πρωτοκόλλου URL επιτρέπουν στους σχεδιαστές ιστού να ενσωματώνουν εικόνες, κινούμενα σχέδια, βίντεο, ήχους, πολυμέσα ροής κ.λπ. σε ιστοσελίδες.

Εκτός από τα δύο παραπάνω προγράμματα περιήγησης, ορισμένοι προγραμματιστές λογισμικού, συμπεριλαμβανομένης της Apple, έχουν κυκλοφορήσει επίσης ορισμένα προϊόντα προγράμματος περιήγησης.

Οι ιστοσελίδες στον Παγκόσμιο Ιστό είναι συνδεδεμένες μεταξύ τους, έτσι ώστε με τη βοήθεια του ποντικιού να μπορεί κάποιος εύκολα να περιηγηθεί ανάμεσα τους.

Στη γλώσσα της Πληροφορικής, όταν κάποιος επισκέπτεται τη μία ιστοσελίδα μετά την άλλη, λέγεται ότι «πλοηγείτε» στον Παγκόσμιο Ιστό.

Για να υπάρχει η δυνατότητα να δει κάνεις το περιεχόμενο των ιστοσελίδων στην οθόνη του υπολογιστή και να προηγηθεί ανάμεσα τους, πρέπει να έχει εγκατασταθεί στον υπολογιστή ένα Λογισμικό Πλοήγησης ή Φυλλομετρητή (Browser).

Οι πιο δημοφιλείς φυλλομετρητές είναι: Microsoft Internet Explorer, Netscape Navigator, Mozilla, Firefox, Opera κ.α.



Αν επιθυμεί κάποιος να επισκεφτεί μια ιστοσελίδα, αρκεί να πληκτρολογήσει τη διεύθυνσή της στο Λογισμικό Πλοήγησης που διαθέτει. Μέσα σε λίγα δευτερόλεπτα θα εμφανιστεί η σελίδα στην οθόνη του υπολογιστή και θα αρχίσει την εξερεύνηση.

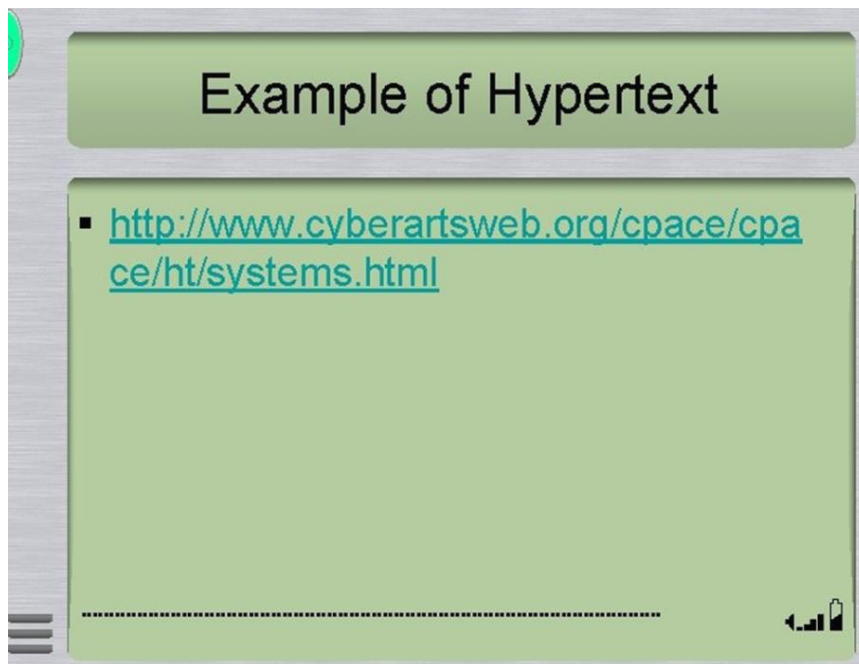
Η ανάγνωση των κειμένων των ιστοσελίδων είναι διαφορετική από αυτή ενός παραδοσιακού κειμένου. Κάθε φορά γίνεται να επιλεγεί διαφορετικός δρόμος ανάγνωσης, ανάλογα με τις πληροφορίες που ψάχνει ο καθένας. Αντίθετα, ένα παραδοσιακό βιβλίο το διαβάζει κάποιος με τη σειρά ξεκινώντας από την αρχή και φτάνοντας μέχρι το τέλος. Τα κείμενα των ιστοσελίδων είναι συνδεδεμένα μεταξύ τους σαν ένα δίκτυο (ιστός). Ανάλογα με τις πληροφορίες που ψάχνει ο ενδιαφερόμενος, μπορεί να περιηγηθεί με ευκολία από τη μία ιστοσελίδα στην άλλη.

Το κείμενο που αποτελείται από πολλά στοιχειώδη αυτόνομα κείμενα συνδεδεμένα μεταξύ τους, ώστε να υπάρχει η δυνατότητα επιλογής διαφορετικών διαδρομών ανάγνωσης και αναζήτησης πληροφοριών, ονομάζεται **Υπερκείμενο (Hypertext)**.

Υπερκείμενο και πως το κατασκευάζουμε (Hypertext)

Το υπερκείμενο, είναι σαν μια φανταστική βιβλιοθήκη στα ράφια της οποίας τα βιβλία ταξινομούνται κάθε φορά με διαφορετικό τρόπο, ανάλογα με τα θέματα που πρόκειται να διαβαστούν.

Για την περιγραφή ενός υπερκείμενου χρησιμοποιείται μια ειδική γλώσσα την Hyper Text Markup Language (HTML). Η γλώσσα **σήμανσης HTML είναι μια γλώσσα που χρησιμοποιείται για την δημιουργία όλου του περιεχομένου που θα εμφανιστεί και είναι ένας τρόπος για να δημιουργηθούν δεδομένα εγγράφων με στοιχεία κειμενικής σήμανσης. Η γλώσσα σήμανσης HTML,** επιτρέπει στον



συγγραφέα να διαμορφώσει ένα κείμενο, το οποίο θα παρέχει συνδέσμους με άλλα έγγραφα και διάφορα αρχεία. Τα υπερκείμενα που βρίσκονται στο WWW ονομάζονται **ιστοσελίδες (web pages)** και το σύνολο των ιστοσελίδων αποτελούν το **site (δικτυακός τόπος)**.

Μετάδοση των πληροφοριών ενός υπερκειμένου στο διαδίκτυο

Η μετάδοση των πληροφοριών ενός υπερκειμένου στο διαδίκτυο γίνεται σε πακέτα:

- Τα δεδομένα τεμαχίζονται σε μικρότερα τμήματα (πακέτα).

- Κάθε πακέτο ακολουθεί διαφορετικές διαδρομές προκειμένου να φτάσει στον προορισμό του.
- Τα πακέτα, αφού φτάσουν στον προορισμό τους επανενώνονται.

Στη συνέχεια, η υπηρεσία WWW, ακολουθεί την αρχιτεκτονική πελάτη-εξυπηρετητή ((Client: Browser - Server: Ένας Η/Υ με ειδικό λογισμικό που επιτρέπει την ανάκτηση πληροφοριών από clients). Τέλος, η μεταφορά των πληροφοριών από τον Server γίνεται με βάση ένα ειδικό πρωτόκολλο επικοινωνίας, το Hyper Text Transport Protocol (HTTP).

Εκπαιδευτική Ενότητα: 6.2 Ανάπτυξη ιστοσελίδων (HTML, CSS)

Το Διαδίκτυο, είναι μια ομάδα υπολογιστών οι οποίοι απαρτίζεται από χιλιάδες ετερογενή δίκτυα ανά τον κόσμο. Δεν γίνεται να υπολογιστεί επακριβώς πόσοι υπολογιστές είναι συνδεδεμένοι στο Διαδίκτυο. Είναι σίγουρο, ωστόσο, ότι ο αριθμός αυτός ανέρχεται σε εκατομμύρια.

Σημαντικό χαρακτηριστικό του Διαδικτύου είναι, ότι κανείς δεν είναι υπεύθυνος για τη διαχείρισή του. Υπάρχουν φυσικά οργανισμοί, οι οποίοι ασχολούνται με τα τεχνικά ζητήματα του δικτύου δημιουργώντας διαδικτυακών εφαρμογών, αλλά δεν υπάρχει κάποιος οργανισμός για τον έλεγχο και την εποπτεία όλου του Διαδικτύου.

Οι χρήστες έχουν πρόσβαση σε μια μεγάλη ποικιλία υπηρεσιών. Οι ιστοσελίδες του Διαδικτύου χωρίζονται σε κατηγορίες ανάλογα με το σκοπό, τις υπηρεσίες και τον στόχο που επιδιώκουν.

Τρόπος εργασίας

Ο τρόπος εργασίας είναι ο εξής: Συντάσσεται ο κώδικας της ιστοσελίδας με το συντάκτη κειμένου, αποθηκεύεται το αρχείο του κώδικα με κατάληξη .html ή όπως αλλιώς ορίζεται σε κάθε εφαρμογή και μετά γίνεται το άνοιγμα του αρχείου του κώδικα με το φυλλομετρητή για να γίνει ορατό το αποτέλεσμα.

Σχεδιασμός σελίδων ιστού

Μια πρώτη ιστοσελίδα, με κάπως μινιμαλιστική άποψη ξεκινά με τον ελάχιστο απαραίτητο κώδικα που πρέπει να γραφτεί για να φτιαχτεί μια κενή ουσιαστικά ιστοσελίδα. Ο κώδικας αυτός καλείται συχνά και σκελετός της ιστοσελίδας.

1	<code><!DOCTYPE html></code>
2	<code><!-- Εφαρμογή 1: Ο ελάχιστος κώδικας HTML μιας ιστοσελίδας</code>
3	<code><u>Αρχείο p01.html</code>
4	<code>--></code>
5	<code><html></code>
6	<code><head></code>
7	<code><title>Τίτλος ιστοσελίδας</title></code>
8	<code><meta charset="utf-8"></code>
9	<code></head></code>
10	<code><body></code>
11	<code>Περιεχόμενο ιστοσελίδας</code>
12	<code></body></code>
13	<code></html></code>

Ο ελάχιστος απαραίτητος κώδικας για τη δημιουργία ιστοσελίδας

Στο πλαίσιο το έγγραφο που εμφανίζεται είναι γραμμένο σε HTML5, (δηλαδή στην έκδοση 5 της γλώσσας σήμανσης υπερκειμένου HTML).

Σε αυτή την γλωσσά είναι γραμμένες οι περισσότερες στατικές σελίδες στον παγκόσμιο ιστό. Οι σελίδες δηλαδή τις οποίες το περιεχόμενο

μεταφέρεται όπως είναι αποθηκευμένο στον εξυπηρετητή ιστοσελίδων, άρα δεν αλλάζει, αν δεν το αλλάξει ο δημιουργός τους.

Αυτό δε ισχύει στις δυναμικές ιστοσελίδες, που το περιεχόμενο δημιουργείται από μια εφαρμογή που εκτελείται στον εξυπηρετητή ιστοσελίδων και άρα μπορεί να αλλάξει, μεταξύ δύο ανανεώσεων της ιστοσελίδας στο παράθυρο του φυλλομετρητή.

Στατική ιστοσελίδα HTML δεν είναι πρόγραμμα που εκτελείται, αλλά οδηγίες για τη διαμόρφωση ενός εγγράφου, σε αντίθεση η δυναμική ιστοσελίδα είναι ένα πρόγραμμα που εκτελείται στο διακομιστή ιστοσελίδων και μπορεί να φτιάχνει διαφορετικό περιεχόμενο σε κάθε κλήση της.

Στον πιο πάνω πίνακα παράδειγμα, η πρώτη γραμμή του κώδικα, αποτελεί μια οδηγία προς το φυλλομετρητή για το είδος του εγγράφου που ακολουθεί.

Γιατί ο φυλλομετρητής πρέπει να γνωρίζει τον τύπο και την έκδοση του κώδικα της ιστοσελίδας για να την εμφανίσει σωστά. Εδώ λοιπόν δήλωση ότι το έγγραφο που ακολουθεί είναι γραμμένο σε κώδικα HTML5.

Ο κώδικας HTML δεν είναι ευαίσθητος σε κεφαλαία και πεζά γράμματα (case sensitive).

Συνεπάγεται, ότι έστω αν γράψουμε DOCTYPE, ή doctype, ή DocType, κ.ο.κ. Σε αυτό που επιβάλλεται να δοθεί προσοχή είναι η εισαγωγή διαστήματος, καθώς για το σύστημα είναι διαφορετικό το DOCTYPE από το DOC TYPE.

Οι 2-4 γραμμές είναι ένα σχόλιο. Πάντα τα σχόλια ξεκινάνε με `<!--` και τελειώνει με `-->`.

Τα σχόλια δεν απασχολούν δεν τα λαμβάνει υπόψιν από το φυλλομετρητή, αλλά μας βοηθούν στην τεκμηρίωση του κώδικα που γράφεται.

Πρέπει να γράφουμε σχόλια στον κώδικα που αναπτύσσουμε, έτσι ώστε αν χρειαστεί στο μέλλον να μπορούμε να ανακαλέσουμε τι κάνουμε σε κάθε σημείο του.

Στη 5 υπάρχει μια ετικέτα (tag). Στην γλώσσα HTML την ονομάζουμε ετικέτα και κάποια σημασία για το φυλλομετρητή. Κάθε ετικέτα οριοθετείται από δύο τριγωνικές αγκύλες (`< >`).

Αν δούμε την γραμμή 5 θα παρατηρήσουμε ότι και στη γραμμή 13 υπάρχει μια ετικέτα που μοιάζει με αυτή της γραμμής 5, αλλά το όνομα της ετικέτας της γραμμής 13 ξεκινά με το χαρακτήρα slash (`/`).

Στην γραμμή 5 λοιπόν η ετικέτα επισημαίνει την έναρξη ενός στοιχείου (element) ενώ η ετικέτα της γραμμής 13 επισημαίνει τη λήξη του.

Αρά αρχικά για τη δημιουργία μιας ιστοσελίδας συντάσσεται :

- ο κώδικα της ιστοσελίδας με το συντάκτη κειμένου,
- αποθηκεύεται το αρχείο του κώδικα με κατάληξη .html ή όπως αλλιώς ορίζεται σε κάθε εφαρμογή και τέλος,
- ανοίγεται το αρχείο του κώδικα με το φυλλομετρητή για να γίνει ορατό το αποτέλεσμα.

Όμως πέρα από το τεχνικό μέρος, που θα αναλυθεί στις πιο κάτω ενότητες, ένα σημαντικό ρολό παίζει και το αισθητικό κομμάτι της.

Ένας ιδιοκτήτης μιας εταιρείας ή ένα στέλεχος σε μια επιχείρηση, θα πρέπει να είναι εξοικειωμένος με τη χρήση μιας ιστοσελίδας ως μέσο προώθησης και αναγνωσιμότητας της επωνυμίας του, επειδή η ιστοσελίδα αποτελεί ήδη σημαντικό μέρος της σύγχρονης επιχειρηματικής αγοράς.

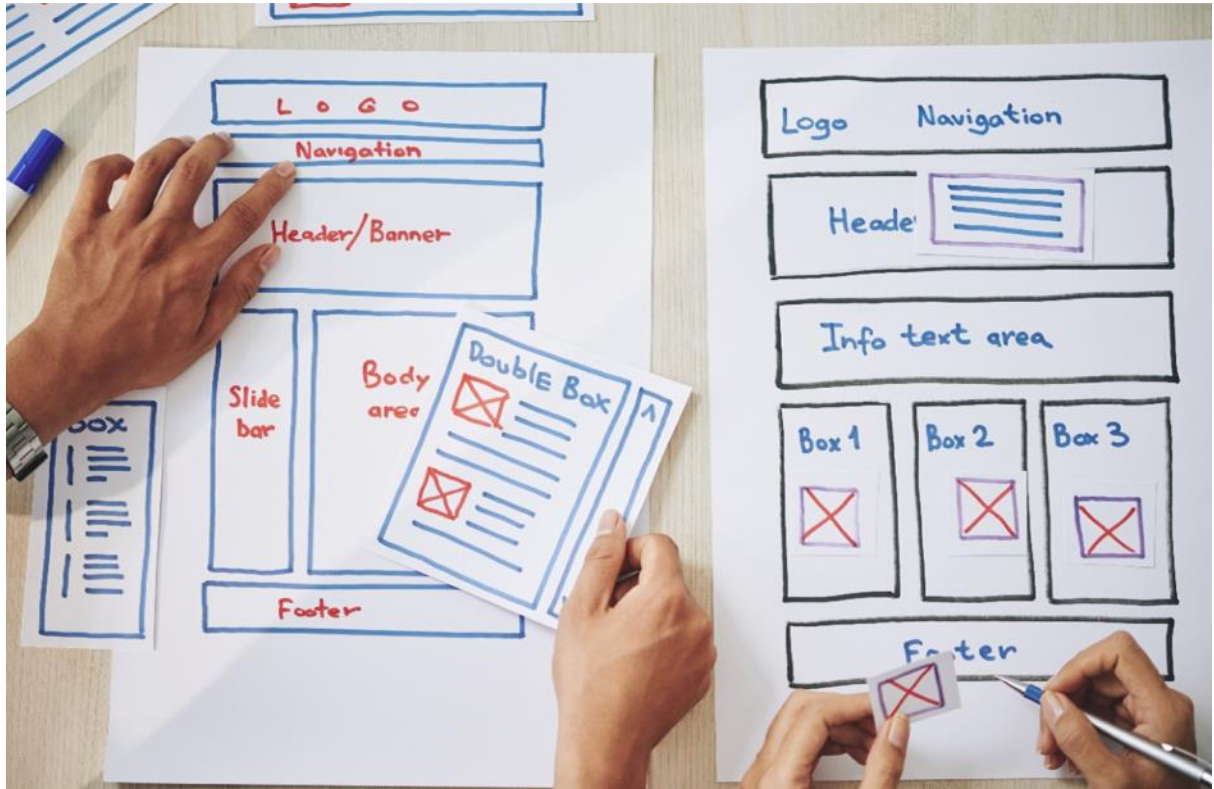
Μέσω του ισότοπου , είναι δυνατόν να βοηθηθούν οι πελάτες να μάθουν περισσότερα για τον ισότοπό και να πεισθούν ότι η εταιρεία είναι αξιόπιστη . Εάν κάποιος ιδιοκτήτης επιχείρησης σκοπεύει να δημιουργήσει έναν ισότοπο, τότε μπορεί να προσλάβει έναν σχεδιαστή ή/και έναν προγραμματιστή ιστού για να τον βοηθήσει να δημιουργήσει τον ισότοπο όπως θέλει. Το πρόβλημα είναι ότι για μερικούς ανθρώπους, οι γραμμές μεταξύ των σχεδιαστών ιστοσελίδων και των προγραμματιστών ιστού είναι ασαφείς. Αυτό κάνει πολλούς ανθρώπους να πιστεύουν ότι οι δύο τους είναι πραγματικά εναλλάξιμοι.

Φυσικά, οι σχεδιαστές ιστοσελίδων μπορεί να χρειαστεί να γνωρίζουν πώς να επεξεργάζονται βασικά προγράμματα και οι προγραμματιστές ιστού πρέπει να έχουν μια βασική κατανόηση του σχεδιασμού.

Ωστόσο, η δουλειά τους δεν είναι στην πραγματικότητα η ίδια. Πιο κάτω θα μαθευτούν περισσότερα για τις διαφορές μεταξύ των δύο.

Web Design

Ο ευκολότερος τρόπος για να γίνει κατανοητός ένας σχεδιαστή Ιστού είναι η εικόνα κάποιου που μεταφράζει μια γνώμη, ή μια ιστορία σε ένα οπτικά ελκυστικό σχέδιο και χρησιμοποιεί τη διάταξή του για να δημιουργήσει μια εμπειρία χρήστη σε



ολόκληρο τον ιστότοπο.

Δεν πρόκειται μόνο για τη δημιουργία σπουδαίων σχεδίων, αλλά για το πώς να δημιουργηθεί η καλύτερη εμπειρία χρήστη και πώς να παραχθεί ένα φιλόξενο περιβάλλον για τους χρήστες.

Μια όμορφη και ομαλή σχεδίαση διάταξης προσφέρει μια αναντικατάστατη εμπειρία στους τελικούς καταναλωτές, ίσως πατώντας το κουμπί αγοράς σε μια παρόρμηση.

Αλλά πολλοί βλέπουν τους σχεδιαστές ιστοσελίδων και τους προγραμματιστές ιστού ως εναλλάξιμους, επειδή οι χρήστες δεν μπορούν πραγματικά να καταλάβουν τι κάνει η ανάπτυξη.

Υπάρχει δυνατότητα να κατανοηθεί τι κάνει ένας **web designer**:

Χρησιμοποιεί εργαλεία λογισμικού όπως το Adobe Photoshop, το Illustrator ή το Sketch για τη δημιουργία της σχεδίασης του ιστότοπου και κατέχει δεξιότητες γραφιστικής και σχεδίασης λογότυπου.

Να γίνει κατανοητό πώς να πραγματοποιηθεί ο σχεδιασμός της εμπειρίας χρήστη και να γίνει προσδιορισμός του ευκολότερου τρόπου επίτευξης της επιθυμητής λειτουργικότητας. Αυτό περιλαμβάνει τη διάταξη του ιστότοπου, τα κουμπιά, τις εικόνες και τη γενική μορφοποίηση.

Η παρακολούθηση των τελευταίων τάσεων σχεδιασμού είναι επίσης μέρος της δουλειάς ενός σχεδιαστή ιστοσελίδων.

Επιπλέον, οι σχεδιαστές ιστοσελίδων πρέπει να διατηρούν συνέπεια στη σχεδίαση των πιο πρόσφατων ιστότοπων, όπως η Google και το Facebook, επειδή η σχεδίαση ιστοσελίδων με την οποία είναι ήδη εξοικειωμένοι οι χρήστες θα κάνει το περιβάλλον και τη διεπαφή του ιστότοπου πιο εύκολη στην πλοήγηση και χρήση στα μάτια του χρήστη .

Να ληφθεί υπόψιν η επωνυμία του ιστότοπου, η χρωματική παλέτα που χρησιμοποιείται και η τυπογραφία και η αναγνωσιμότητα του ιστότοπου.

Υπεύθυνος για την κατασκευή πρωτοτύπων, εκτέλεση γραφιστικής σχεδίασης καταλόγου προϊόντων, εκτέλεση διαφόρων δραστηριοτήτων διαφημιστικού σχεδιασμού.

Ένας ώριμος σχεδιαστής πρέπει να είναι σε θέση να κατέχει διάφορα εργαλεία λογισμικού για να δημιουργήσει όμορφες και ακριβείς διατάξεις εικόνων για πελάτες.

Αυτά που αναφέρθηκαν είναι οι υπηρεσίες που μπορούν να παρέχουν οι επαγγελματίες σχεδιαστές ιστοσελίδων.



Εδώ θα αφευθούμε στους προγραμματιστές Ιστού

Οι προγραμματιστές Ιστού μετατρέπουν το σχέδιο σε ζωντανό ισότοπο αναπτύσσοντας τη σχεδίαση και τη λειτουργικότητα του ισότοπου χρησιμοποιώντας γλώσσες ιστού και εργαλεία λογισμικού. Οι προγραμματιστές ιστού χωρίζονται περαιτέρω σε δύο υποκατηγορίες: προγραμματιστές front-end και προγραμματιστές back-end. Οι προγραμματιστές διεπαφής θα πρέπει να κατέχουν τουλάχιστον τρεις κύριες γλώσσες, HTML, CSS και JavaScript (JS). Αυτές οι γλώσσες θα επιτρέψουν στους προγραμματιστές:

- να δημιουργήσουν έναν πλήρη ιστότοπο, από την κύρια διάταξη σχεδίασης έως την εισαγωγή εικόνων,
- την εφαρμογή διαφορετικών τύπων οικογενειών τυπογραφίας και γραμματοσειρών, την υλοποίηση κινούμενων εικόνων,
- τη ροή διαφορετικών σελίδων,
- διεπαφές φορμών και πολλά άλλα.

Από την άλλη πλευρά, το back-end engineering φέρει την πιο σημαντική επιχειρηματική λογική μιας επιχείρησης, όπως ο έλεγχος μελών, η υπογραφή πόρων και η ταμειακή ροή των καταναλωτών. Μπορεί να ειπωθεί ότι ο προηγούμενος

σχεδιασμός είναι να προσελκύει επισκέπτες, αλλά στο τέλος η πραγματική κερδοφόρα συμπεριφορά συναλλαγής πραγματοποιείται στην ανάπτυξη back-end. Αν και αόρατο και αφηρημένο, είναι το πιο κρίσιμο κομμάτι.

Οι προγραμματιστές front-end/back-end είναι συνήθως υπεύθυνοι για πολλές κουραστικές εργασίες ενοποίησης, όπως:

Δημιουργία της πραγματικής διεπαφής όπου ο χρήστης αλληλοεπιδρά με τον ισότοπο. Η διεπαφή έχει κατασκευαστεί από προγραμματιστές front-end που χρησιμοποιούν γλώσσες HTML, CSS και JS. Για να επιταχύνουν τη διαδικασία ανάπτυξης, οι προγραμματιστές front-end μπορούν να χρησιμοποιήσουν προεπεξεργαστές στυλ, Javascript και πλαίσια.

Προκειμένου να υλοποιηθεί ένας δυναμικός ισότοπος και να υποβληθούν όλα τα απαιτούμενα δεδομένα στον διακομιστή και τη βάση δεδομένων, οι προγραμματιστές front-end θα πρέπει να παρέχουν σχέδια σήμανσης για προγραμματιστές back-end. Δημιουργήστε τη ραχοκοκαλιά του ισότοπου σας χρησιμοποιώντας γλώσσες όπως PHP και MySQL. Χρήση του IDE :

- για την δημιουργία της δομής ιστοτόπου για τους πελάτες να συνδέουν εργαλεία συναλλαγών όπως ERP/CRM ή ηλεκτρονικό σύστημα πιστωτικών καρτών για να διατηρήσουν τη λειτουργία του κεντρικού υπολογιστή, να ενημερώνονται τακτικά για να διατηρείται ομαλή λειτουργία,
- συντήρηση ασφάλειας πληροφοριών,
- έλεγχος και δημιουργία αντιγράφων ασφαλείας για την αποφυγή απώλειας δεδομένων

Συμπερασματικά

Προφανώς, οι σχεδιαστές και οι προγραμματιστές παίζουν πάντα έναν αναπόσπαστο ρόλο όταν πρόκειται για ρόλους εργασίας. Ωστόσο, υπάρχουν ορισμένες προφανείς διαφορές μεταξύ των δύο, όπως οι σχεδιαστές είναι κυρίως υπεύθυνοι για το σχεδιασμό μιας όμορφης σελίδας χρησιμοποιώντας τη δική τους αισθητική γνώση διαφόρων εργαλείων. Είναι επίσης υπεύθυνοι να δίνουν στους πελάτες μια πρώτη εντύπωση.

Από την άλλη πλευρά, οι προγραμματιστές ιστού επικεντρώνονται στην πραγματική λογική των συναλλαγών, επικεντρώνονται στη δημιουργία καθαρού κώδικα που ζωντανεύει το σχέδιο.

Για επιχειρήσεις που αναζητούν ενσωμάτωση εικόνας, αυτό που χρειάζεται είναι μια ομάδα που μπορεί να σχεδιάσει αλλά και να προγραμματίσει την σελίδα της στο διαδίκτυο.

Οι σχεδιαστές ιστού μπορεί να μην γνωρίζουν πώς λειτουργεί η επιχειρηματική λογική όμως μπορούν και γνωρίζουν πως να σχεδιάζουν

Κατασκευή μιας ιστοσελίδας

Ο σχεδιασμός ιστοσελίδων (από τα αγγλικά Web design) είναι ένας κλάδος της ανάπτυξης ιστοσελίδων και ένας τύπος σχεδίασης του οποίου οι εργασίες περιλαμβάνουν τη σχεδίαση διεπαφών χρήστη Ιστού για τοποθεσίες ή εφαρμογές Ιστού .

Οι σχεδιαστές ιστοσελίδων σχεδιάζουν τη λογική δομή των ιστοσελίδων , σκέφτονται καλύτερα βολικές λύσεις για την παρουσίαση πληροφοριών , καθώς και τη διακόσμηση του έργου web . Ως αποτέλεσμα της διασταύρωσης δύο κλάδων της ανθρώπινης δραστηριότητας, ένας ικανός σχεδιαστής ιστοσελίδων πρέπει να είναι εξοικειωμένος με τις τελευταίες τεχνολογίες Ιστού και να έχει τις κατάλληλες καλλιτεχνικές ιδιότητες .

ΑΙΣΘΗΤΙΚΗ WEB DESIGN

Οι βασικές αρχές - Νόμοι - κανόνες για την σωστή και αποτελεσματική σχεδίαση

Πολλοί ισότοποι φαίνεται να πιστεύουν ότι κάτι δεν πάει καλά, η διάταξη δεν είναι πολύ κακή, οι εικόνες είναι όμορφες και το κείμενο είναι καλογραμμένο. Ποιο είναι όμως το πρόβλημα; Τα περισσότερα από αυτά προκαλούνται από τη μη τήρηση των βασικών αρχών σχεδίασης και το υλικό του ισότοπου δεν έχει διαμορφωθεί σωστά, γεγονός που καθιστά δύσκολο για τους χρήστες να δουν πού βρίσκεται το επίκεντρο του σχεδιασμού του ισότοπου. Έχουν διευθετηθεί αρκετές βασικές αρχές

σχεδιασμού για την παραγωγή ισότοπου, έτσι ώστε οι σχεδιαστές να μην εργάζονται πλέον σκληρά, αλλά να είναι αναποτελεσματικοί, επιτρέποντάς τους να ξεπεράσουν γρήγορα τα τυφλά σημεία του σχεδιασμού.

Το πιο συνηθισμένο λάθος που κάνει ένας σχεδιαστής ιστοσελίδων;

Το πιο συνηθισμένο λάθος που κάνει ένας ,είναι να ξεκινήσει να εργάζεται σε ένα έργο όταν βιάζεται και να αρχίσει να σχεδιάζει έναν ισότοπο χωρίς να γνωρίζει τους στόχους, τα αντικείμενα, τα χαρακτηριστικά του κλάδου και τα χαρακτηριστικά του ισότοπου.

Όλα αυτά προκαλούνται από την αποτυχία να καθιερωθούν κοινές αξίες με τους πελάτες, έτσι ώστε οι ιδέες του άλλου να φαίνονται παράλληλες σε χρόνο και χώρο, η επικοινωνία χειροτερεύει και κάτω από έναν φαύλο κύκλο, η εμπιστοσύνη και των δύο μερών γίνεται σιγά σιγά.

Πώς να συλλεχθούν πληροφορίες πριν το σχεδιασμό της ιστοσελίδας;

Το πρώτο αντικείμενο συλλογής δεδομένων είναι ο "πελάτης". Η γνώμη κανενός δεν είναι πιο σημαντική από τη γνώμη του πελάτη. Αυτή είναι επίσης μια σημαντική διαδικασία για τη διόρθωση των τιμών και των δύο μερών. Γίνεται να ληφθούν διάφορες πληροφορίες ισότοπου από τον πελάτη, συμπεριλαμβανομένης της εταιρικής ταυτότητας (CI), φωτογραφιών προϊόντων, εταιρικών προφίλ, καταλόγων, κοινού ισότοπου, τύπων κλάδου... Όσο περισσότερα γνωρίζει κάποιος για τον κλάδο, τόσο πιο ευκολά μπορεί να σχεδιάσει έναν κατάλληλο ισότοπο και, στη συνέχεια, μπορεί να συλλέξει πληροφορίες ισότοπου από τον ίδιο κλάδο και να συγκεντρώσει τις πληροφορίες. Μόνο τότε ξεκινάει επίσημα ο σχεδιασμός.

Δεν μπορεί να γίνει σχεδιασμός σύμφωνα με τις προσωπικές προτιμήσεις;

Αυτό που αρέσει σε έναν σχεδιαστή δεν σημαίνει ότι είναι "καλό" ή "κατάλληλο". Κάθε σχέδιο πρέπει να είναι μηδενικό. Ο σχεδιασμός είναι για να λύνει διάφορα προβλήματα. Εάν η ιστοσελίδα που σχεδιάζεται είναι απαράδεκτη, θα είναι αποτυχία. Πολλές εταιρείες σχεδιασμού περιορίζουν τον αριθμό των αναθεωρήσεων, δεν δέχονται κριτική, δεν δέχονται τις απόψεις των πελατών και επιμένουν στις δικές τους ιδέες σχεδίασης. Αυτοί είναι όλοι λάθος τρόποι για να

φτάσει το σχέδιο στα άκρα. Ο καλύτερος σχεδιασμός είναι σαν το νερό, που μπορεί να είναι διαφορετικά δοχεία ζήτησης.

Αρχές σχεδιασμού ιστοσελίδων

Μετά τη συλλογή των πληροφοριών, οι εργασίες σχεδίασης ιστότοπου θα πραγματοποιηθούν επίσημα. Υπάρχουν πολλές βασικές αρχές για το σχεδιασμό ιστοσελίδων. Εάν υπάρχει δυνατότητα να σχεδιαστεί σύμφωνα με αυτές τις αρχές σχεδιασμού, ο ιστότοπος δεν θα είναι πολύ κακός.

Για να γίνει κατανοητό τι φταίει με τον σχεδιασμένο ιστότοπο, πρέπει να χρησιμοποιηθούν οι ακόλουθες αρχές σχεδίασης και να ελεγχθούν τα σχέδιά. Οι σχεδιαστές δεν ξέρουν ποια είναι τα προβλήματά τους, δηλαδή δεν έχουν μια «αίσθηση σχεδιαστικής ασθένειας», αισθάνονται ότι «όλα είναι υποκειμενικές προτιμήσεις των άλλων» και δεν αισθάνονται ότι υπάρχει κάποιο λάθος στο δικό τους σχέδιο.

Η λέξη Σχέδιο προέρχεται από τα λατινικά σημαίνει «ολοκλήρωσε το σχέδιο» Αν το σχέδιο δεν μπορεί να ολοκληρωθεί είναι αποτυχημένο σχέδιο.

Ποιες είναι οι βασικές αρχές του σχεδιασμού;

Κοινοποιήστε ένα μόνο, σαφές θέμα

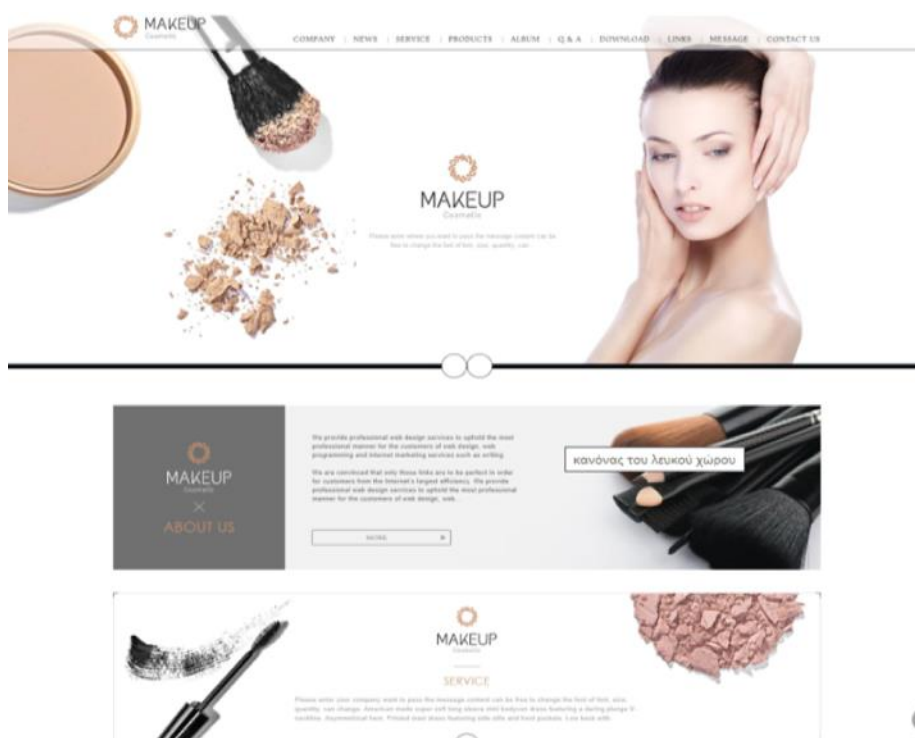
Υπάρχουν πολλές πληροφορίες που είναι να παρουσιαστούν ταυτόχρονα, αλλά οι υπερβολικές πληροφορίες θα κάνουν τους ανθρώπους να τρομάζουν. Εάν υπάρχει μόνο ένα θέμα, χρειάζεται μόνο μερικά απλά διαγράμματα, έναν τίτλο και μια περιγραφή. Ο ενιαίος οπτικός σχεδιασμός βοηθά τον θεατή να εστιάσει σε αυτό το θέμα. Εάν η ιστοσελίδα χρησιμοποιεί πολλά θέματα και στοιχεία σχεδίασης, συνιστάται να κάνετε συμβιβασμούς. Να γίνει τοποθέτηση του περιεχόμενου για το οποίο πρόκειται να μιλήσει κάποιος σε άλλες ιστοσελίδες και συνδέσει αυτές τις ιστοσελίδες.

Ρυθμίστε την ατμόσφαιρα του ιστότοπου από την αρχή.

Είναι πολύ σημαντικό να ορισθεί η ατμόσφαιρα που θα δημιουργήσει ο ιστότοπος από την αρχή. Είναι επίσης αυτό που οι εταιρείες σχεδιασμού αποκαλούν συχνά

"στυλ σχεδίασης". Η ρύθμιση του στυλ εξαρτάται από πολλές συνθήκες, οι πιο σημαντικές είναι η εταιρική ταυτότητα (CI), τα χαρακτηριστικά του κλάδου , και προτιμήσεις. Εάν πρόκειται για ισότοπο αγορών, είναι κατάλληλο να χρησιμοποιηθούν πλούσια χρώματα για να γίνει αντικείμενο προσοχής. Εάν είναι μια παραδοσιακή βιομηχανία κατασκευής, είναι κατάλληλη για μια ήρεμη και συγκρατημένη ατμόσφαιρα. Εάν είναι ισότοπος επίπλων, χρειάζεται να δημιουργηθεί μια αρμονική και ζεστή ατμόσφαιρα. Η ατμόσφαιρα του ισότοπου είναι το πιο δύσκολο πράγμα να οριστεί στο σχεδιασμό ιστοσελίδων, επειδή η αίσθηση της ατμόσφαιρας του καθενός είναι υποκειμενική και η αίσθηση χρησιμοποιείται συχνά για να ορίσει το στυλ τεχνολογίας, το απλό στυλ, το στυλ Zen, το τρισδιάστατο στυλ, το χαρούμενο στυλ. Τα στοιχεία που δημιουργούν στυλ είναι: διαμόρφωση διάταξης, σχήμα, χρώμα, μοτίβο, γραμματοσειρά, ισορροπία, υφή. Οι έμπειροι σχεδιαστές μπορούν να κατανοήσουν τις ιδέες των πελατών πιο γρήγορα και θα προτείνουν διάφορες περιπτώσεις σχεδίασης για να διευκρινίσουν τις ανάγκες των πελατών . Επίσης σχεδιάζουν χωρίς να γνωρίζουν τι λέει ο πελάτης, έχουν τις δικές τους ιδέες και επιμονή και διατηρούν την αρχική πρόθεση του σχεδίου που συγκινεί τις καρδιές των ανθρώπων.

Κανονίστε προσεκτικά τη σειρά του περιεχομένου του ιστότοπου.



Προκειμένου να ανταποκριθεί στις κινητές συσκευές, ο ισότοπος έχει αναπτύξει μια σχεδίαση ισότοπου RWD. Η περιήγηση στον ισότοπο είναι από πάνω προς τα κάτω, επομένως η πρώτη εικόνα παραπάνω γίνεται πολύ σημαντική. Αυτή είναι η πρώτη οπτική σχεδίαση που προκαλεί το ενδιαφέρον. Ο εγκέφαλος επεξεργάζεται τις εικόνες πολύ πιο γρήγορα από τις λέξεις, η χρήση εικόνων μπορεί να τραβήξει την προσοχή του θεατή πιο γρήγορα. Το τακτοποιημένο περιεχόμενο του ισότοπου προκαλεί το μεγάλο ενδιαφέρον του θεατή και το διαβάζει σε όλη τη διαδρομή, κάτι που είναι ένα επιτυχημένο σχέδιο.

Σωστή αξιολόγηση του λευκού χώρου.

Στο βιβλίο « The Design Aesthetics of White Space », αναφέρεται ότι το 90% του καλού design εξαρτάται από τον «λευκό χώρο». Ο λευκός χώρος στην πραγματικότητα δεν αφήνει κενό χώρο. Το λευκό διάστημα σημαίνει ότι υπάρχει χώρος. Δεν πρέπει να γεμίζεται η κάθε θέση με πληροφορίες, έτσι ώστε οι εικόνες και τα κείμενα του θέματος να είναι πιο εμφανείς και πιο ευανάγνωστα. Πολλοί σχεδιαστές ισότοπων βλέπουν ότι υπάρχει χώρος στην οθόνη. Εάν γεμίσουν με πληροφορίες, όχι μόνο δεν θα είναι σε θέση να παρουσιάσει την υφή, αλλά θα εμποδίσει επίσης την ανάγνωση. Υπάρχουν πολλά σημεία για σχεδιασμό κενού εφαρμογών, όπως: εταιρική ταυτότητα, γραφικά κενά, πίνακες, διάστιχο, διάστιχο λέξεων, διαμόρφωση εικόνας. Η σχεδίαση λευκού χώρου επιτρέπει στον θεατή να αναπνέει, να χαλαρώνει τα μάτια και να εστιάζεται περισσότερο, επιτρέποντας στο θέμα να εστιάσει. Πόσος χώρος πρέπει να μείνει κενός; Πώς να διαμορφωθεί το λευκό διάστημα; Είναι μια σχετικά δύσκολη σχεδιαστική δεξιότητα που απαιτεί εμπειρία. Ακόμα κι αν είναι δύσκολη, δεν πρέπει να υπάρχουν κενά. Εάν υπάρξουν κενά, πρέπει να αναβαθμιστεί η σχεδιαστική ικανότητα και να δημιουργηθούν πιο τυπικά έργα ισότοπου.

Ο συνδυασμός χρωμάτων είναι η πρώτη εντύπωση του ισότοπου

Γιατί οι επιγραφές καταστημάτων στην Ταϊβάν είναι ακατάστατες και αποδιοργανωμένες, ενώ οι πινακίδες καταστημάτων στην Ιαπωνία, την Ευρώπη και τις Ηνωμένες Πολιτείες έχουν μια συνολική αισθητική; Ένα από τα προβλήματα είναι η αντιστοίχιση χρωμάτων, η ακατάστατη και παράλογη αντιστοίχιση

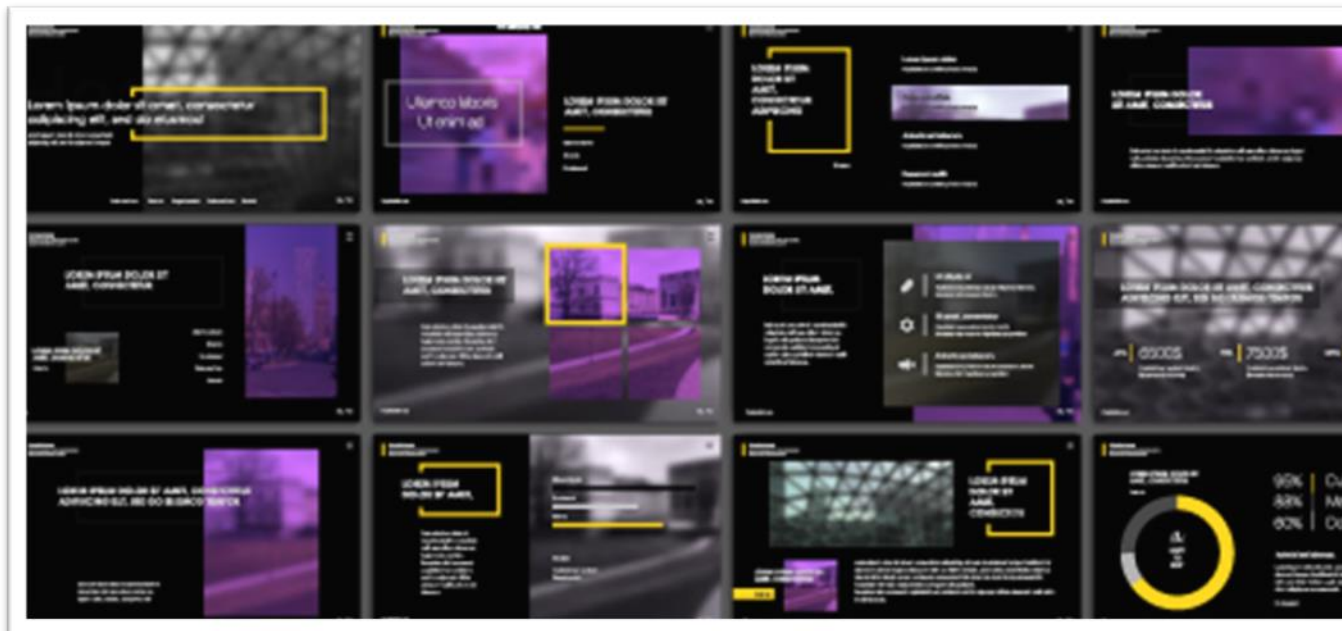
χρωμάτων είναι ο πρώτος δολοφόνος του σχεδιασμού. Το ίδιο ισχύει και για τους ισότοπους. Η αντιστοίχιση χρωμάτων είναι αυτό που αισθάνεται ο ισότοπος με την πρώτη ματιά. Η ασυνέπεια της αντιστοίχισης χρωμάτων δημιουργεί αμέσως μια αίσθηση παραβίασης. Η αντιστοίχιση χρωμάτων ισότοπου συνήθως συνεχίζει τη στρατηγική αντιστοίχισης χρωμάτων της εταιρικής ταυτότητας (CI). Εάν δεν υπάρχει σχέδιο αναγνώρισης, η βοήθεια αντιστοίχισης χρωμάτων από τους σχεδιαστές ιστοσελίδων είναι πολύ απαραίτητη. Η επιστήμη των χρωμάτων είναι η ικανότητα που πρέπει να αναπτύξουν οι σχεδιαστές. Το χρώμα δεν αφορά μόνο τη διάταξη, γραμματοσειρές, πίνακες, γραμμές κ.λπ. Οι φωτογραφίες, οι εικονογραφήσεις και τα βίντεο που έχουν επιλεγεί να χρησιμοποιηθούν έχουν επίσης ζητήματα αντιστοίχισης χρωμάτων. Το χρώμα επηρεάζει την ατμόσφαιρα του ισότοπου. Οι σχεδιαστές που μπορούν να κατακτήσουν την αντιστοίχιση χρωμάτων και τα γραφικά είναι η μισή μάχη. Εάν είναι καλοί σε αυτό, συνιστάται η χρήση εργαλείων αντιστοίχισης χρωμάτων ή τσιπ χρωμάτων για να εξασκηθούν στην αντιστοίχιση χρωμάτων και να εξετάσουν την αντιστοίχιση χρωμάτων ορισμένων ξένων ιστοσελίδων.

Διατήρηση συνεπή στυλ σχεδίασης ισότοπου

Αφού ο ισότοπος ορίσει το θέμα, το στυλ, τη σειρά περιεχομένου, την κεφαλίδα και το υποσέλιδο σελίδας, το επόμενο σχέδιο πρέπει να ακολουθεί αυτούς τους ορισμούς, συμπεριλαμβανομένων του χρώματος, του σχήματος, της υφής, των εικόνων και άλλων υλικών. Έτσι ώστε κάθε σχέδιο ή σελίδα να μοιάζει με μια οικογένεια. Αυτή είναι η ίδια λογική με το σύστημα εταιρικής ταυτότητας (CIS). Πρέπει να παρουσιάζει μια αίσθηση ακεραιότητας με συστηματικό τρόπο. Η αίσθηση της ακεραιότητας δεν απαιτεί να γίνει συνδυασμός όλα τα παρόμοια πράγματα μαζί. Υπάρχει ένας ισότοπος επίπλων, σχεδιασμένος από σχεδιαστές που είναι γεμάτος φωτογραφίες επίπλων. Αυτή τη στιγμή, θα πρέπει να εισαχθούν μερικές ενδιαφέρουσες εικόνες ή μερικές μικρές φωτογραφίες προνομιακού μάρκετινγκ ή διαφημιστικά γραφικά του άρθρου, τα οποία μπορούν να αμβλύνουν τα προβλήματα που προκαλούνται από παρόμοιο περιεχόμενο.

Όχι στην υποτίμηση των επιλογών γραμματοσειράς του ισότοπου

Η χρήση γραμματοσειρών ισότοπου θα επηρεάσει την "υφή του ισότοπου". Για ισότοπους που δίνουν έμφαση στην αναγνωσιμότητα, να μην γίνεται χρήση γραμματοσειρών με περιγράμματα, σκιές, διακοσμήσεις, αστείες, φανταχτερές,



χαριτωμένες και υπερβολικά διακοσμημένες γραμματοσειρές. Εάν δεν συντίθενται απευθείας με εικόνες γραμματοσειρές, να δίνεται ιδιαίτερη προσοχή στη σειρά ρύθμισης των γραμματοσειρών CSS. Οι αγγλικές γραμματοσειρές θα πρέπει να τοποθετούνται μπροστά από τις κινεζικές γραμματοσειρές, επειδή τα κινέζικα συχνά περιέχουν ήδη αγγλικές γραμματοσειρές.

Για την βελτιστοποίηση της ταχύτητα και της απόδοση του ισότοπου, συνιστάται η χρήση του κατασκευασμένου ισότοπου σε γραμματοσειρές, αν και διαφορετικά λειτουργικά συστήματα και προγράμματα περιήγησης θα έχουν διαφορετικές αποδόσεις και η εμπειρία δεν μπορεί να ενοποιηθεί.

Ωστόσο, εφόσον η σειρά γραμματοσειράς, το διάστημα κειμένου και το λευκό διάστημα έχουν ρυθμιστεί σωστά, μπορεί να επιτευχθεί η υφή που απαιτείται από τον ισότοπο. Τέλος, να μην γίνεται χρήση πολλών γραμματοσειρών στον ισότοπο, το πολύ δύο γραμματοσειρές μπορούν να χρησιμοποιηθούν μαζί και οι γραμματοσειρές μπορούν να διαβαστούν στο σύνολό τους. Χρησιμοποιώντας σωστά τις αλλαγές μεγέθους γραμματοσειράς και τις διαμορφώσεις, τα μπλοκ μπορούν επίσης να χωριστούν σε περιοχές χωρίς επιπλέον σχεδιασμό.

Οι εικονογραφήσεις εξελίσσουν οπτικά τον ιστότοπο

Μερικές φορές, όταν το κείμενο δεν μπορεί να εκφραστεί με ακρίβεια, οι πληροφορίες πρέπει να οπτικοποιηθούν. Αυτή τη στιγμή, μπορείτε να σχεδιάσετε εικόνες για να οπτικοποιήσετε τις πληροφορίες. Οι καλές εικονογραφήσεις κάνουν τον ιστότοπο πιο ενδιαφέρον. Η εισαγωγή φωτογραφιών ή γραφημάτων μεταξύ των παραγράφων μπορεί να βελτιώσει την ανάγνωση στους αναγνώστες. Η φαντασία και η κατανόηση του άρθρου θα κάνει επίσης το άρθρο πιο ζωντανό και ενδιαφέρον, αλλά δεν πρέπει να υπάρχουν ένα σωρό φωτογραφίες ή εικόνες που δεν έχουν νόημα. Η χρήση παράλογων υλικών είναι ένα ταμπού στο σχεδιασμό. Ανάλογα με το περιεχόμενο του ιστότοπου, μερικές φορές μπορούν να χρησιμοποιηθούν χειροποίητα προσχέδια για τη δημιουργία ατμόσφαιρας ιστότοπου. Ο σκοπός της χρήσης φωτογραφιών, εικόνων και εικονογραφήσεων είναι να γίνει η ιστοσελίδα πιο ευανάγνωστη και κατανοητή.

Σημαντικές φωτογραφίες του ιστότοπου

Η αρχή του σχεδιασμού είναι η καλή χρήση φωτογραφιών και εικονογραφήσεων, αλλά τι γίνεται αν δεν υπάρχουν φωτογραφίες; Εκτός από την αγορά εικόνων από τη γκαλερί, για να παρουσιαστεί καλύτερο οπτικό αποτέλεσμα, συνιστάται η λήψη αποκλειστικών φωτογραφιών. Ολόκληρος ο ιστότοπος χρησιμοποιεί πάρα πολλές φωτογραφίες γκαλερί, κάτι που θα δώσει μια εξωπραγματική αίσθηση εκτός από την έλλειψη γενικής αίσθησης. Οι φωτογραφίες (εικόνες) είναι τα πιο σημαντικά στοιχεία στο υλικό του ιστότοπου. Όσο κακός κι αν είναι ο ιστότοπος, με καλές φωτογραφίες, μπορεί να φτάσει στο βασικό επίπεδο. Χωρίς καλές φωτογραφίες, όσο καλή διάταξη, αντιστοίχιση χρωμάτων, copywriting, κενός χώρος, σειρά περιεχομένου, κ.λπ., θα οφείλεται στο ότι οι φωτογραφίες χαμηλής ποιότητας καθιστούν τον ιστότοπο μη κατάλληλο, ειδικά την εικόνα της αρχικής σελίδας. Οι καλές φωτογραφίες μπορούν να δημιουργήσουν γρήγορα μια ατμόσφαιρα, να τονίσουν το θέμα και να έχουν προγραμματισμένες φωτογραφίες με πλήρες περιεχόμενο, επαρκή φωτισμό, ισορροπημένη σύνθεση και ακριβείς γωνίες θέασης. Η απλή μετα-επεξεργασία μπορεί να εφαρμοστεί σε ιστότοπους. Η κακή ποιότητα

φωτογραφίας απαιτεί περισσότερη μετα-επεξεργασία προσαρμογή, συμπεριλαμβανομένης της διόρθωσης του χρώματος, της προσαρμογής των τόνων, της αφαίρεσης θορύβου, της ενίσχυσης της αντίθεσης, της αύξησης της έκθεσης.

Εφαρμογή του βίντεο στον ισότοπο

Μερικές φορές οι λέξεις, οι φωτογραφίες, οι εικόνες και οι πίνακες δεν μπορούν να εκφράσουν πλήρως το περιεχόμενο που πρέπει να εξηγηθεί, επομένως είναι πολύ κατάλληλο να τραβηχτεί ένα βίντεο , όπως: διαδικασία μηχανικής λειτουργίας, μαγείρεμα συνταγής, μέθοδος εγκατάστασης επίπλων. Ο ισότοπος μπορεί να γράψει ένα σύστημα διαχείρισης ήχου και βίντεο ή να χρησιμοποιήσει απευθείας το Youtube και το Vimeo για να ενσωματώσει τον ήχο και το βίντεο. Η καλή χρήση του βίντεο είναι επίσης η βασική αρχή του σχεδιασμού της ιστοσελίδας.

Η διαφορά στον σχεδιασμό της ιστοσελίδας βρίσκεται στις λεπτομέρειες

Εκτός από τις προκαταρκτικές και βασικές αρχές σχεδιασμού που αναφέρθηκαν παραπάνω, το επόμενο βήμα είναι η εισαγωγή πιο κουραστικών σχεδιαστικών λεπτομερειών. Για την μετάβαση από έναν συνηθισμένο σχεδιαστή σε έναν σχεδιαστή υψηλού επιπέδου, πρέπει να δοθεί προσοχή στις λεπτομέρειες του σχεδιασμού. Η εργασία σχεδιασμού ιστοσελίδων έχει φτάσει σε ένα επίπεδο όπου οι λεπτομέρειες είναι η μεγαλύτερη διαφορά από άλλες.

Αποκλειστικό στυλ σχεδίασης

Μια επαγγελματική εταιρεία σχεδίασης ιστοσελίδων θα εντοπίσει σίγουρα το στυλ σχεδίασης της εταιρείας και οι διορισμένοι σχεδιαστές θα ακολουθήσουν επίσης αυτήν την κατεύθυνση. Προϊόντα όπως η APPLE έχουν το δικό τους στυλ σχεδίασης, το οποίο είναι απλό, κατανοητό, εύκολο στη χρήση, ομαλή και απλή χρωματική αντιστοίχιση. Τα προϊόντα μπορούν να βρουν κοινά χαρακτηριστικά μεταξύ τους, και το ίδιο ισχύει για τους ισότοπους, ανεξάρτητα από την ατμόσφαιρα που θα δημιουργηθεί; Τι υλικό να σχεδιαστεί; Τι φωτογραφία να χρησιμοποιηθεί; Ποια έκδοση να ανοιχτεί; Θα υπάρχουν κάποιες βασικές αρχές που δεν θα αλλάξουν και

αυτή η τοποθέτηση του στυλ σχεδίασης θα επηρεάσει την κατεύθυνση σχεδίασης ολόκληρου του ισότοπου, επομένως όταν επιλεγθεί μια εταιρεία σχεδιασμού ισότοπου, πρέπει πρώτα να εξετασθεί το στυλ σχεδίασης της εταιρείας σχεδιασμού.

Σε περίπτωση που τα έργα σχεδιασμού στον ισότοπο δεν είναι αρεστά τότε δεν είναι σωστός ο εξαναγκασμός της επιτροπής σχεδιασμού. Επιπρόσθετα, επιλέγονται έργα σχεδιασμού που είναι επιθυμητά. Έτσι είναι εφικτό η επιτροπή σχεδιασμού να γίνει ομαλότερη.

Πρακτική εφαρμογή του κενού

Αναφέρθηκε νωρίτερα ότι πρέπει να χρησιμοποιείται λευκός χώρος. Ποια είναι η πρακτική εφαρμογή του λευκού χώρου; Για παράδειγμα, το φόντο και ένα συγκεκριμένο εύρος του εταιρικού σήματος (LOGO) παραμένουν κενά, το οποίο προστατεύει την παρουσίαση του LOGO. Πολλές κενές στήλες αφήνονται σκόπιμα μεταξύ των παραγράφων κειμένου για να δημιουργηθεί ένα αόρατο όριο παραγράφου. Υπάρχουν επίσης ισότοποι αγορών που συχνά πηγαίνουν στο παρασκήνιο και αφήνουν μόνο τα προϊόντα να επισημαίνουν τα προϊόντα. Τα παραπάνω τρία είναι κοινές περιπτώσεις κενού χώρου. Φυσικά, υπάρχουν πολλές έννοιες του κενού χώρου, αλλά είναι σχεδόν αχώριστες από αυτές τις τρεις αρχές:

- ❖ δημιουργία αίσθησης χώρου,
- ❖ άνετη και εύκολη ανάγνωση και
- ❖ βελτίωση της εστίασης.

Η τυπογραφία είναι σημαντική

Ο ισότοπος είναι είτε μια εικόνα είτε ένα κείμενο. Η εικόνα είναι μια εικόνα, αλλά το κείμενο δεν είναι τόσο απλό. Το κείμενο πρέπει να είναι τακτοποιημένο και κάποια πράγματα που πρέπει να τους δοθούν ιδιαίτερη προσοχή είναι: ο πυρήνας, το διάστιχο, το μέγεθος γραμματοσειράς, ο αριθμός λέξεων, το χρώμα τίτλου, από αριστερά προς τα δεξιά, το κέντρο, τα σημεία στίξης, ο αριθμός των νέων γραμμών, ο αριθμός των κενών παραγράφων και η συνέπεια της διάταξης. Εάν το κείμενο είναι σωστά διατεταγμένο, θα υπάρχει μια καλή σχεδιαστική παρουσίαση. Όταν κάποιοι ισότοποι είναι με πολλά του κείμενα θα πρέπει να δίνεται μεγαλύτερη

προσοχή στην πλαϊνή σειρά του κειμένου. Μπορείτε να δείτε ορισμένους ισότοπους με επεξεργασία κειμένου. Όπως το Commonwealth Magazine, το Kangjian, το Business Weekly. Αυτοί οι ισότοποι έχουν μια τυποποιημένη και ολιστική αίσθηση διάταξης κειμένου, κάτι που αξίζει πολύ αναφορά. Η διάταξη κειμένου είναι πολύ σημαντική. Για να δώσουμε ένα παράδειγμα μη φιλικής ανάγνωσης κειμένου, πολλά άρθρα ισότοπου έχουν πάρα πολλές λέξεις σε κάθε γραμμή, γεγονός που τις καθιστά υπερβολικά πλατιές. Ο αριθμός των κινεζικών χαρακτήρων είναι πάνω από 50 λέξεις και ο αριθμός των λέξεων υπερβαίνει το πλάτος ανθρώπινων όντων. Το πλάτος του αριθμού των λέξεων που είναι κατάλληλες για ανάγνωση είναι μεταξύ 45 και 55 κινεζικών χαρακτήρων σε μία γραμμή. Ένας τέτοιος αριθμός των λέξεων είναι ένα πλάτος που διαβάζεται πιο εύκολα. Επίσης, η απόσταση μεταξύ των γραμμών γραμματοσειράς του άρθρου είναι επίσης πολύ σημαντική. Η απόσταση μεταξύ των γραμμών είναι πολύ γεμάτη και δυσανάγνωστη. Η καλύτερη απόσταση μεταξύ των γραμμών είναι μεταξύ 30% και 50% του ύψους της λέξης. Προσοχή επίσης στο μέγεθος της γραμματοσειράς. Όσο μεγαλύτερη είναι η λέξη, τόσο το καλύτερο. Εάν η λέξη είναι πολύ μεγάλη, θα είναι αρκετά τραχιά. Στον ισότοπο Wenqing χρησιμοποιούνται μικρές λέξεις για να εκφραστεί η υφή, αλλά είναι επίσης δύσκολο για τους μεγαλύτερους με πρεσβυωπία. Το μέγεθος της γραμματοσειράς θα πρέπει να προσαρμόζεται ανάλογα με τα χαρακτηριστικά του ισότοπου. Οι γραμματοσειρές του ισότοπου υγείας έχουν σκόπιμα μεγεθυμένες γραμματοσειρές, επειδή ο ισότοπος έχει περισσότερους μεσήλικες και ηλικιωμένους θεατές.

Το περιεχόμενο είναι το επίκεντρο του καλού σχεδιασμού

Ο σκοπός της σχεδίασης ιστοσελίδων είναι σίγουρα η έκθεση. Είναι η παρουσίαση "χρήσιμων πληροφοριών" στο Διαδίκτυο. Μέσω του σχεδιασμού, ο ισότοπος μπορεί να γίνει πιο εύκολα αποδεκτός και αγαπητός, βελτιώνοντας έτσι την αποτελεσματικότητα της έκθεσης του ισότοπου. Αυτές οι λογικές τυχαίνει να είναι παρόμοιες με αυτές της Google. Η λογική κατάταξης είναι συνεπής, με σαφές θέμα, καλή σχεδίαση εμπειρίας χρήστη (UX), καθομιλουμένου και ευανάγνωστο κείμενο, σαφείς πίνακες, βίντεο που ταιριάζουν στο θέμα, καλή δομή SEO και εύκολη. Όλα τα παραπάνω πρέπει να επιτυγχάνονται με το σχεδιασμό. Η σχεδίαση ιστοτόπου

είναι μια εργασία τεχνικής ολοκλήρωσης, η οποία απαιτεί σταθερή εσωτερική αντοχή αντί για κίνημα με λάστιχο. Εάν η ουσία ενός ισότοπου είναι το περιεχόμενο, τότε θα πρέπει να ληφθεί σοβαρά σε τι μπορεί να βοηθήσει ο ισότοπος; Ποιο είναι το κοινό; Ποιες πληροφορίες πρέπει να προετοιμαστούν για να ικανοποιήσουν τον θεατή; Μετά την εκκαθάριση των αναγκών των χρηστών, γίνεται επιστροφή στη σχεδίαση. Η χρήση της σχεδίασης για την επίλυση διαφόρων προβλημάτων είναι το πραγματικό νόημα της κατασκευής μιας ιστοσελίδας.

Συμπερασματικά

Γιατί είναι βασικό το web design;

Πως και γιατί λειτουργεί το web design;

Σίγουρα δεν βασίζεται μόνο στο συναίσθημα και την υποκειμενική κρίση. Για τις επαγγελματικές εταιρείες σχεδιασμού, γίνεται αναφορά στους λογούς για τον ακατάλληλο σχεδιασμό. Φυσικά, ο σχεδιασμός δεν είναι τόσο απλός, αλλά εφόσον βασίζεται στις βασικές σχεδιαστικές αρχές, ο ισότοπος σίγουρα δεν θα είναι κακός. Θα παρουσιαστούν τα στοιχεία που είναι όλα "βασικές δεξιότητες σχεδιασμού". Εάν δεν είναι σωστές αυτές οι λεπτομέρειες σχεδίασης, πρέπει να γίνει προσπάθεια να κατανοηθούν αυτές τις βασικές αρχές, ελπίζοντας ότι θα βοηθήσουν στο δρόμο του σχεδιασμού ιστοσελίδων.

Για να ξεκινήσει ο σχεδιασμό αρχικά, θα πρέπει να γίνει ταξινόμηση των στοιχείων του ισότοπου σύμφωνα με τους στόχους μάρκετινγκ. Χωρίς σαφή στόχο, δεν ξέρετε τι να δώσετε προτεραιότητα.

❶ Εικόνα προϊόντος Βαλτέ την εικόνας στην πιο εντυπωσιακή θέση. Η θέση αυτή αριστερή πλευρά (στον εγκέφαλο αρέσει ιδιαίτερα ο τρόπος οπτικής κίνησης από αριστερά προς τα δεξιά).

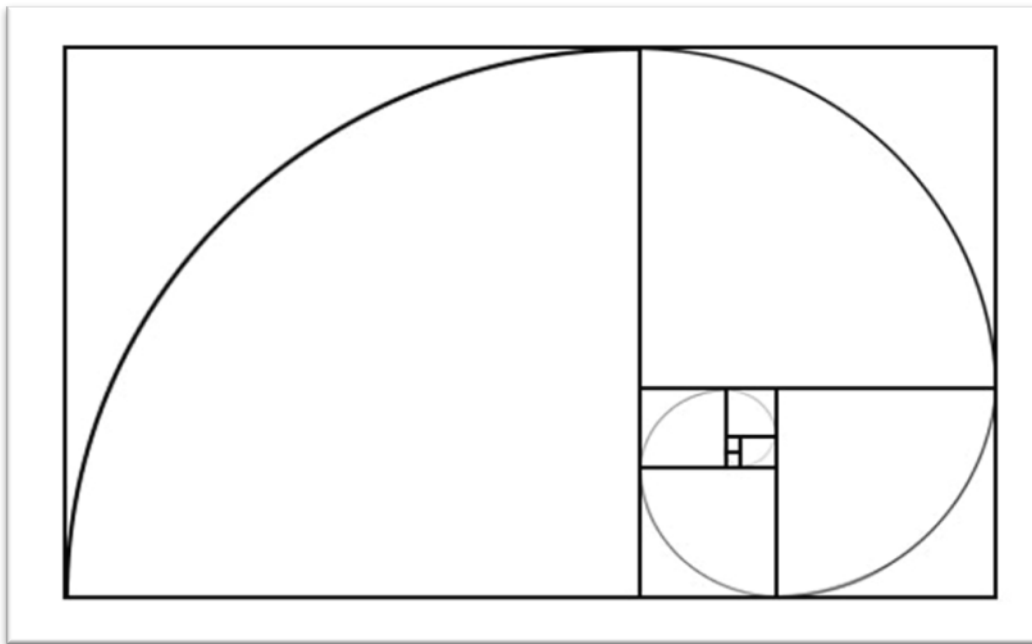
❷ Στη συνέχεια, θα αναφερθείτε στον σκοπό, αν είναι έκπτωση θα πρέπει να γράψετε πχ υπάρχουν εντυπωσιακές προσφορές.

❸ Εποχική απήχηση, σύμφωνα με την εποχή υπάρχουν προϊόντα ,ή δραστηριότητες που πρέπει να αναφέρουμε

4 Περίοδος προώθησης : αγοράστε το γρηγορά!

Πρακτική Αρχή Σχεδιασμού Ιστού : Η Χρυσή Αναλογία

Το 1,618 είναι ο αντιπροσωπευτικός αριθμός της χρυσής αναλογίας, που κάνει τα πάντα τέλεια επίδειξη δύναμης και ομορφιάς (τουλάχιστον οι περισσότεροι από αυτούς πιστεύουν έτσι). Επίσης, υπάρχουν οι ακολουθίες Fibonacci, όπου κάθε αριθμός είναι το άθροισμα των δύο προηγούμενων αριθμών—0, 1, 1, 2, 3, 5, 8, 13, 21... και ούτω καθεξής. Αξίζει να σημειωθεί ότι δύο φαινομενικά άσχετοι αριθμοί



μπορούν στην πραγματικότητα να συναχθούν από τους σχετικούς αριθμούς τους.

Έτσι φαίνεται η χρυσή τομή:

Πολλοί καλλιτέχνες και αρχιτέκτονες ολοκληρώνουν τα έργα τους σύμφωνα με τη χρυσή τομή, το πιο γνωστό από τα οποία είναι ο Παρθενώνας που χτίστηκε στην αρχαία Ελλάδα.

Μπορεί να χρησιμοποιηθεί για σχεδιασμό ιστοσελίδων; Το Twitter χρησιμοποιεί τη χρυσή τομή στο layout του.

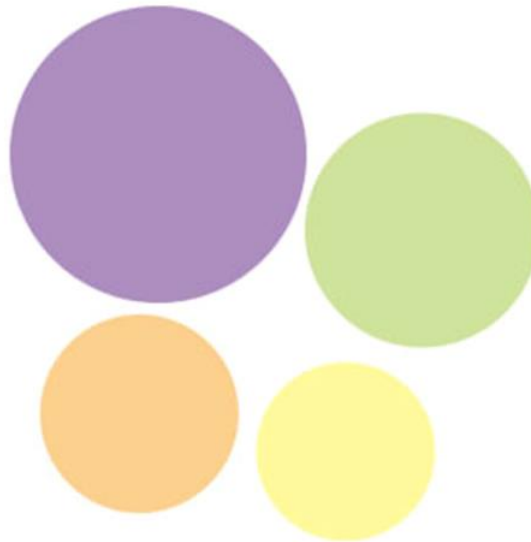
Εάν η οθόνη του προγράμματος περιήγησής είναι ευρύτερη, θα εμφανιστούν περισσότερα παράθυρα που παρέχουν πληροφορίες για τη βελτίωση της χρηστικότητάς της, αλλά αυτό το στενότερο πλάτος αντιπροσωπεύει επίσης την

αρχή της χρυσής αναλογίας. Έτσι, εάν το πλάτος είναι 960 εικονοστοιχεία, διαιρείται το με 1,618 (=593 εικονοστοιχεία). Το περιεχόμενο του ιστότοπου μπορεί να είναι 593 px και το περίγραμμα είναι 367 px. Εάν το ύψος της ιστοσελίδας είναι 760px, μπορεί να χωριστεί σε δύο κομμάτια, 479px και 299px αντίστοιχα ($760/1.618 \approx 470$).

Αποτελεσματική Αρχή Σχεδίασης Ιστού: Οπτική Ιεραρχία

Μία από τις σημαντικές αρχές του καλού σχεδιασμού ιστοσελίδων είναι η οπτική ιεραρχία, η οποία είναι η σειρά με την οποία το ανθρώπινο μάτι παρατηρεί αυτό που βλέπει.

Εξάσκηση: Κοιτάξτε τους κύκλους με σειρά σπουδαιότητας:



Οι άνθρωποι μπορούν εύκολα να τους ταξινομήσουν χωρίς να τους γνωρίζουν. Αυτό είναι το αποτέλεσμα της οπτικής ιεραρχίας. Αυτός ο νόμος μέσα σε εισαγωγικά είναι μια αρμονική βάση για την σχεδίαση και τοποθέτηση στον σχεδιασμό ιστοσελίδων

Αποτελεσματική Αρχή Σχεδιασμού Ιστού: Ο κανόνας των τρίτων

Είναι καλή ιδέα να χρησιμοποιηθούν περισσότερο οι εικόνες στη σχεδίαση ιστοσελίδων. Η οπτική επικοινωνία των ιδεών είναι πιο αποτελεσματική και ταχύτερη από οποιαδήποτε περιγραφή κειμένου.

Είναι καλύτερο να ακολουθεί ο κανόνας των τρίτων για τις φωτογραφίες, τοποθέτηση του θέματος στη διαχωριστική γραμμή , ή αποκοπή της εικόνας σε τρία ίσα μέρη, έτσι ώστε το θέμα να ξεχωρίζει, αλλά και να διατηρεί την κατάλληλη αίσθηση του χώρου.



Αποτελεσματική Αρχή Σχεδιασμού Ιστού: **Σχεδιασμός Gestalt**

Η ψυχολογία Gestalt είναι μια θεωρία του νου και του εγκεφάλου. Η αρχή του είναι ότι το ανθρώπινο μάτι βλέπει το σύνολο του αντικειμένου πριν αντιληφθεί τα διάφορα μέρη του αντικειμένου.



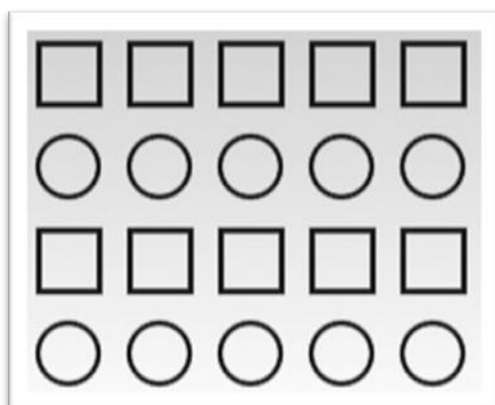
Παράδειγμα:

Έχετε παρατηρήσει ότι βλέπετε ένα σκύλο χωρίς να γίνει εστίαση σε κάθε μαύρο σημείο του σκύλου.

Συμπεραίνεται ότι το κλειδί είναι ότι οι άνθρωποι βλέπουν το σύνολο πριν από τα μέρη. Δηλαδή οι χρήστες βλέπουν πάντα την πλήρη εικόνα του ιστότοπού , πριν μεταβούν στην κεφαλίδα, τα μενού, το υποσέλιδο κ.λπ.

Για αποτελεσματικό σχεδιασμό ιστοσελίδων, πρέπει να διασφαλισθεί ότι τα πράγματα που δεν είναι μαζί δεν θεωρούνται ως ένα. Ομοίως, γίνεται ομαδοποίηση ορισμένων στοιχείων σχεδίασης (μενού πλοήγησης, υποσέλιδα, κ.λπ.) για να φανεί ότι αποτελούν ένα σύνολο.

Οι άνθρωποι έχουν την τάση να βλέπουν τα αντικείμενα στο σύνολό τους. Για παράδειγμα, ομαδοποιούν παρόμοια πράγματα μαζί. Αυτή η ομοιότητα μπορεί να



έχει τη μορφή σχημάτων, χρωμάτων, αποχρώσεων ή άλλων χαρακτηριστικών.

Υπάρχει η τάση να σκέφτονται οι άνθρωποι παρόμοια πράγματα σαν ένα, όπως φαίνεται στο παρακάτω σχήμα, όταν μια σειρά από τετράγωνα και μια σειρά κύκλων είναι δίπλα-δίπλα, δύο σειρές τετραγώνων και δύο σειρές κύκλων διατεταγμένες οριζόντια, όχι κατακόρυφα

Οι άνθρωποι έχουν την τάση να κλείνουν τα σχήματα ή τις φόρμες . Τείνουν και επιδιώκουν να ολοκληρώνουν το σχήμα χωρίς κλειστά σχήματα, όπως όταν λείπουν τμήματα ολόκληρης της εικόνας, η αντίληψή καλύπτει τα οπτικά κενά. Για

παράδειγμα, στην παρακάτω εικόνα, βλέπουμε έναν κύκλο και ένα τετράγωνο, παρόλο που αυτά τα σχήματα δεν υπάρχουν στην πραγματικότητα.

Χωρίς τον νόμο του κλεισίματος, θα ήταν αντιληπτές μόνο διαφορετικές γραμμές διαφορετικού μήκους, αλλά σύμφωνα με το νόμο του κλεισίματος, γίνεται συνδυασμός αυτών των γραμμών σε ένα συνολικό σχήμα.

Η χρήση του νόμου του κλεισίματος κάνει ένα λογότυπο ή ένα στοιχείο σχεδίασης πιο ενδιαφέρον. Ένα καλό παράδειγμα είναι το λογότυπο της WWF που



σχεδιάστηκε από τον Sir Peter Scott το 1961



Ο νόμος της συνέχειας

Οι άνθρωποι τείνουν να πιστεύουν ότι μια γραμμή αναπτύσσεται προς μια δεδομένη κατεύθυνση. Όπου υπάρχουν τομές μεταξύ αντικειμένων (όπως

γραμμές), τείνουν να θεωρούν τις δύο γραμμές ως δύο αδιάσπαστες οντότητες. Ακόμη και με επικάλυψη, τα ερεθίσματα εξακολουθούν να είναι ξεκάθαρα.

Σε αυτό το κλασικό παράδειγμα διαφήμισης παρακάτω, ο εγκέφαλος βλέπει τις διακεκομμένες γραμμές ως ένα συνεχές σύνολο, γεγονός που κάνει το λογότυπο της IBM πιο συμβολικό και ενοποιημένο από τις προηγούμενες εκδόσεις.

Χρησιμοποιώντας τον νόμο της συνέχειας,



ένα λογότυπο από ρίγες του ίδιου χρώματος δημιουργεί μια μόνιμη εντύπωση.

Υπάρχουν και άλλες αρχές Gestalt όπως το Figure-ground ή οι Αρχές Gestalt (αν σχηματίζει ένα κανονικό, απλό και διατεταγμένο μοτίβο, όπως οι Ολυμπιακοί δακτύλιοι, τα αντικείμενα τείνουν να ομαδοποιούνται σύμφωνα με την αντίληψη των ανθρώπων).

Ένας νόμος:

Απλό, μινιμαλιστικό σχέδιο δεν σημαίνει χωρίς σχέδιο. Το υπερβολικό περιεχόμενο στη σελίδα θα προκαλέσει πίεση στους χρήστες, θα χρησιμοποιήσει έξυπνα τον λευκό χώρο και θα εργαστεί σκληρά προς την κατεύθυνση του μινιμαλιστικού σχεδιασμού!

Συμπερασματικά

Για τον σωστό σχεδιασμό ιστοσελίδων πρέπει να είναι γνωστές και να ακολουθούνται οι βασικές αρχές της σχεδίασης. Αρά η τέχνη και η τεχνική δεν είναι το ίδιο.

Θα πρέπει η σκέψη να βρίσκεται στον χρήστη και να λαμβάνονται υπόψιν οι στόχοι μάρκετινγκ. Χρησιμοποιώντας αυτές τις αρχές σχεδιασμού ιστοσελίδων,

βοηθούνται οι πελάτες να δημιουργήσουν όμορφα σχέδια ιστοσελίδων που αποδίδουν.

Μια ιστοσελίδα, όταν πληροί τις παραπάνω προδιαγραφές από θέμα σχεδιασμού, αισθητικής αλλά έχει και εύκολη και σωστή χρήση, θα είναι ένα σωστό εργαλείο για την άνοδο της εκάστοτε εταιρίας.

Η χρυσή τομή φυσικά είναι πάντοτε η δημιουργία ενός ιστότοπου με τις αρχές της χρηστικότητας, ο οποίος την ίδια στιγμή ενσωματώνει όμορφο αισθητικά περιεχόμενο και παρουσιάζει μια βελτιωμένη εικόνα, προσφέροντας μια καλή εμπειρία χρήσης.

Το μεγαλύτερο και συνηθέστερο λάθος που γίνεται στο web design, είναι ότι ο σχεδιαστής ή ο αναπτυκτής πολλές φορές αδυνατεί να καταλάβει ότι ο ίδιος δεν αντιπροσωπεύει σε καμία περίπτωση το μέσο χρήστη. Αυτό είναι απολύτως λογικό - το απαιτεί η δουλειά και, επομένως, ο αναπτυκτής θα έχει περισσότερες γνώσεις, ενώ θα χαρακτηρίζεται από μια μεγαλύτερη ικανότητα καλής πλοήγησης μέσα σε ένα δύσκολο ισότοπο. Για αυτό ακριβώς το λόγο, πολλές ιστοσελίδες κατασκευάζονται με γνώμονα τα προσωπικά θέλω του αναπτυκτή, τα οποία, μπορεί από τη μια να θέτουν υψηλά τον πήχη, αλλά ίσως αγνοούν από την άλλη ότι ο μέσος χρήστης αναζητά κάτι απλό στην πλοήγηση. Είναι απαραίτητο να δίνεται έναυσμα στον επισκέπτη να χρησιμοποιεί τον ιστότοπο. Η ευχρηστία υιοθετείται για τον σχεδιασμό ιστοσελίδων - όχι για τον ίδιο τον ιστότοπο, αλλά για τους τυπικούς τελικούς χρήστες. Αυτός είναι ο τρόπος με τον οποίο η χρηστικότητα γίνεται όλο και πιο δημοφιλής στους αναπτυκτές ιστού επειδή μόνο έτσι μπορεί να είναι σίγουρο ότι ο χρήστης θα είναι ικανοποιημένος.

Ένα άλλο λάθος είναι το γεγονός ότι πολλές ιστοσελίδες έχουν ήχο, ο οποίος ξεκινά να παίζει μόλις ο επισκέπτης φτάσει στη σελίδα. Πρέπει πάντοτε να υπάρχει η επιλογή ο ήχος να απενεργοποιείται - υπάρχει μεγάλη πιθανότητα ο χρήστης να ενοχληθεί από την παρουσία μουσικής. Από την άλλη πλευρά, θα πρέπει επίσης να αποφεύγονται διαφημίσεις που αναβοσβήνουν, όπως και banners.

Σχεδόν πάντα, τα banners είναι στην ιστοσελίδα ό,τι είναι σε ένα τσίρκο τα μεγάλα πανιά που προσπαθούν να πείσουν κάποιον να μπει στο χώρο - περιττά και αχρείαστα. Αν το περιεχόμενο είναι ποιοτικό, ο επισκέπτης θα γυρίσει ξανά. Οι υπερβολικά κυριολεκτικές μηχανές αναζήτησης μειώνουν την ευχρηστία επειδή δεν μπορούν να χειριστούν τα πληκτρολόγια, τους πλουραλισμούς, τις παύλες και άλλες παραλλαγές των όρων ερωτήματος.

Τέτοιες μηχανές αναζήτησης είναι μη παραγωγικές και το μόνο που καταφέρνουν είναι να ενοχλήσουν το χρήστη. Ένα σχετικό πρόβλημα είναι όταν οι μηχανές αναζήτησης δίνουν προτεραιότητα στα αποτελέσματα καθαρά βάσει του αριθμού των όρων ερωτήματος που περιέχουν, αντί για τη σημασία κάθε εγγράφου.

Πολύ καλύτερα θα ήταν, αν η μηχανή αναζήτησης του site παρουσιάζει τα πιο σχετικά με τους αναζητηθέντες όρους ευρήματα στην κορυφή της λίστας - ειδικά για σημαντικά ερωτήματα, όπως ονόματα των προϊόντων.

Η μηχανή αναζήτησης θα πρέπει να λαμβάνεται πολύ σοβαρά - είναι το σημείο της σελίδας στο οποίο στρέφεται ο επισκέπτης όταν δε μπορεί να βρει αυτό που θέλει. Αν και η προχωρημένη αναζήτηση πολλές φορές δύναται να βοηθήσει, συνήθως η απλή αναζήτηση αρκεί, αν ο αλγόριθμος είναι σωστά κωδικοποιημένος.

Σε κάθε περίπτωση, το κουμπί της αναζήτησης θα πρέπει να είναι προσβάσιμο καθ' όλη τη διάρκεια της επίσκεψης στον ισότοπο. Η αναζήτηση είναι ο πιο συνηθισμένος τρόπος για τους χρήστες να ανακαλύπτουν ισότοπους που αναζητούν.

Η αναζήτηση είναι επίσης ένας από τους πιο σημαντικούς τρόπους με τους οποίους οι χρήστες βρίσκουν το δρόμο τους μέσα στη σελίδα.

Κάτι άλλο που μπορεί να βοηθήσει τον επισκέπτη, είναι η αλλαγή του χρώματος των τοποθεσιών που ο ίδιος έχει επισκεφθεί μέσα στη σελίδα. Οι σύνδεσμοι είναι ο βασικός παράγοντας αυτής της διαδικασίας.

Οι χρήστες μπορούν να αποκλείσουν συνδέσμους που αποδείχτηκαν μη βοηθητικοί στις προηγούμενες επισκέψεις τους. Αντίθετα, θα μπορούσαν να επανεξετάσουν συνδέσμους που κατά τη γνώμη τους ήταν χρήσιμοι στο παρελθόν.

Επιπλέον, γνωρίζοντας ποιες σελίδες έχουν ήδη επισκεφθεί, απαλλάσσονται από την επανειλημμένη επανάληψη των ίδιων σελίδων ξανά και ξανά.

Τι κάνει ένας web designer

Τέτοιοι ειδικοί είναι υπεύθυνοι για την εμφάνιση και τη διάταξη του ισότοπου. Μπορεί να είναι είτε μια εντελώς νέα σελίδα που δημιουργείται από την αρχή, είτε επανασχεδιασμός μιας προηγούμενης πύλης. Οι σχεδιαστές αποφασίζουν πώς θα ταυριάζουν διαφορετικά μέρη του ισότοπου και ποια λογική θα λειτουργήσει μεταξύ τους. Σε αντίθεση με τους προγραμματιστές Ιστού που ασχολούνται με τον προγραμματισμό και τη σύνταξη κωδίκων, ένας σχεδιαστής ιστοσελίδων δίνει μεγαλύτερη προσοχή στο οπτικό. Σε ορισμένες περιπτώσεις, αυτά τα επαγγέλματα μπορεί να αλληλεπικαλύπτονται.

Ένας σχεδιαστής Ιστού σχεδιάζει διατάξεις για σελίδες προορισμού, ισότοπους για εταιρείες, ηλεκτρονικά καταστήματα και επίσης σχεδιάζει διεπαφές για διαδικτυακές υπηρεσίες και εφαρμογές. Τι άλλο μπορεί να κάνει: να δημιουργήσει πρότυπα για ενημερωτικά δελτία ηλεκτρονικού ταχυδρομείου, να εφεύρει και να σχεδιάζει πανό στο Διαδίκτυο.

Η ροή εργασίας του σχεδιαστή μοιάζει με αυτό:

- ❖ Ο σχεδιαστής λαμβάνει δεδομένα από τον πελάτη , γιατί χρειάζεται ο ισότοπος, πώς θα λειτουργήσει, ποιοι χρήστες θα τον επισκεφτούν. Μια καλή επιλογή εάν ο πελάτης γνωρίζει ακριβώς τι χρειάζεται. Αλλά αυτό δεν συμβαίνει πάντα, επομένως ο σχεδιαστής πρέπει να κάνει ερωτήσεις και να παίρνει απαντήσεις ο ίδιος. Ο σχεδιαστής καταγράφει την εργασία, τις απαιτήσεις για τον ισότοπο και τις απαντήσεις σε ερωτήσεις στη σύντομη περιγραφή.
- ❖ Ο σχεδιαστής ιστοσελίδων δημιουργεί σκίτσα ή πρωτότυπα του μελλοντικού ισότοπου, σε χαρτί ή σε ειδικά προγράμματα. Τα πρωτότυπα μοιάζουν περισσότερο με διαγράμματα, τα οποία υποδεικνύουν τη θέση των κύριων στοιχείων στις σελίδες του ισότοπου.

- ❖ Με βάση τα πρωτότυπα, ο σχεδιαστής σχεδιάζει μια χρωματική διάταξη. Εδώ μπορείτε ήδη να εργαστείτε σε συνδυασμούς χρωμάτων, να προσθέσετε κινούμενα σχέδια.

Ο σχεδιαστής δίνει την ολοκληρωμένη διάταξη στον πελάτη ή στους προγραμματιστές που θα τη μεταφέρουν στον ιστότοπο.

Σημαντικές δεξιότητες σχεδιαστή Ιστού πέρα από το σχεδιασμό

- ❖ Internet marketing και analytics για την ανάλυση δεδομένων σχετικά με τις ανάγκες και τις επιθυμίες των χρηστών, για να μάθει πώς λειτουργεί η διαφήμιση και γιατί να δημιουργήσει έναν ιστότοπο.
- ❖ Διαπραγματεύσεις για επικοινωνία με πελάτες και συναδέλφους.
- ❖ Συγγραφή και επεξεργασία, γιατί κείμενο και σχέδιο είναι άρρηκτα συνδεδεμένα.
- ❖ Ψυχολογία να μπορεί να μπει στη θέση του επισκέπτη του site και να καταλάβει τι είναι σημαντικό για αυτόν.
- ❖ Ανάπτυξη διάταξης και frontend, για να μάθει πώς η διάταξη που σχεδιάζεται στο πρόγραμμα μετατρέπεται σε χώρο εργασίας.
- ❖ Άλλοι τομείς σχεδιασμού: εικονογράφηση, γραφιστική, animation και δημιουργία διαδραστικών banner.

Σε ποια κατεύθυνση πρέπει να αναπτυχθεί ένας web designer;

Καθώς ένας σχεδιαστής εξελίσσεται, είτε εμβαθύνει σε έναν από τους τομείς του web design, είτε μαθαίνει να εργάζεται σε ολόκληρο το προϊόν, εφαρμόζοντας γνώσεις από διαφορετικούς τομείς.

Εδώ είναι σε τι μπορεί να ειδικευτεί ένας σχεδιαστής ιστοσελίδων:

- ❖ Ο σχεδιασμός UX / UI σχεδιάζει την εμφάνιση ισότοπων και εφαρμογών, λαμβάνοντας υπόψη την ευκολία για διαφορετικούς χρήστες.
- ❖ Το UX, συντομογραφία της εμπειρίας χρήστη, σημαίνει κυριολεκτικά "εμπειρία χρήστη". Εδώ, δίνεται προσοχή στον χρήστη και τι εντύπωση παίρνει από την εργασία με τη διεπαφή, πώς περιηγείται στις σελίδες, αν πετυχαίνει τον στόχο του και πόσο δύσκολο είναι για αυτόν να το κάνει.

- ❖ UI - διεπαφή χρήστη - αναφέρεται στην εμφάνιση της διεπαφής, τα χαρακτηριστικά και το σχεδιασμό της. Η τοποθεσία, το μέγεθος των στοιχείων και οι χρωματικοί τόνοι είναι σημαντικές. Γιατί πρώτα απ' όλα, πρέπει να γίνει το σχέδιο φιλικό προς το χρήστη.

Οι σχεδιαστές διεπαφής δεν σχεδιάζουν μόνο κανονικούς ισότοπους. εργάζονται στην εμφάνιση υπηρεσιών Διαδικτύου, εφαρμογών για κινητά, προγραμμάτων υπολογιστών.

Δημιουργία μικροαλληλεπιδράσεων και σχεδίασης κίνησης. Οι μικροαλληλεπιδράσεις είναι σύντομες κινούμενες εικόνες σε έναν ισότοπο ή μια εφαρμογή που εμφανίζονται ως απόκριση στις ενέργειες των χρηστών.

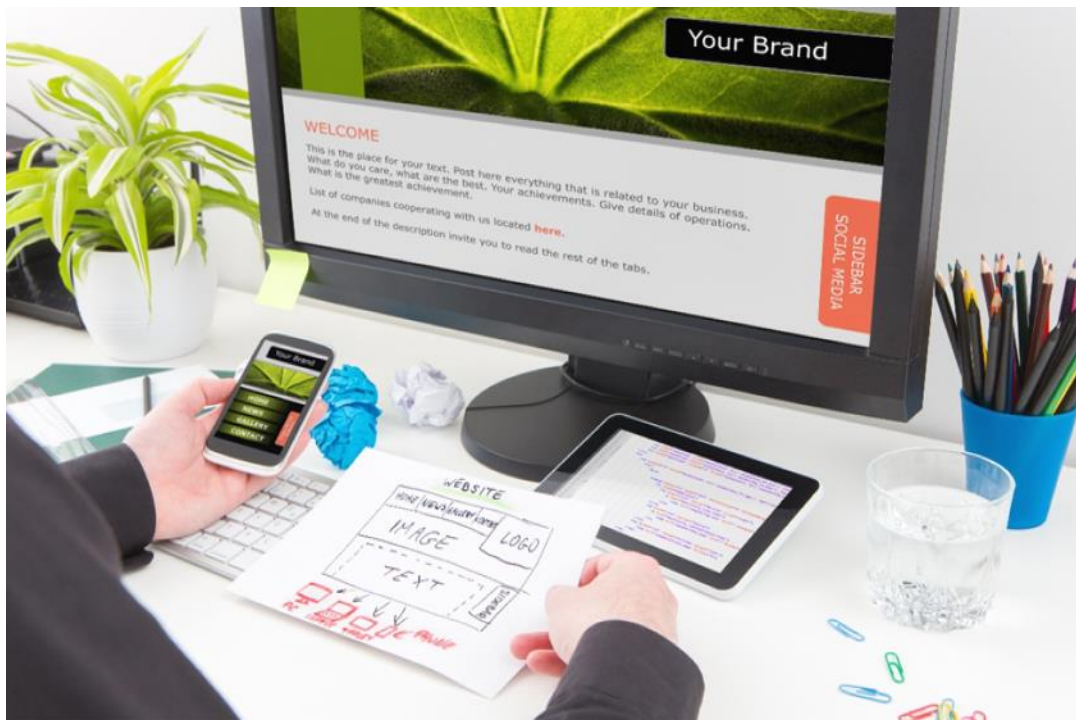
Branding και δημιουργία λογότυπου. Εκτός από μια καλή ιστοσελίδα, οι εταιρείες καλούνται να δημιουργήσουν ένα λογότυπο, εταιρική ταυτότητα. Για την ασχολία προς αυτή την κατεύθυνση, χρειάζεται μια δημιουργική προσέγγιση και την ικανότητα σχεδίασης.

Μερικοί σχεδιαστές γίνονται αναλυτές, πραγματοποιούν δοκιμές A/B σε τελικά προϊόντα, κανονίζουν συνεντεύξεις χρηστών, κάνουν ερωτήσεις στους πελάτες για να κάνουν το προϊόν καλύτερο.

Οι σχεδιαστές προϊόντων δεν ειδικεύονται σε έναν συγκεκριμένο τομέα, αλλά προσπαθούν να μάθουν για τα πάντα. Εργάζονται στο προϊόν ως σύνολο, δοκιμάζοντας ιδέες και υποθέσεις, δημιουργώντας πρωτότυπα, σχεδιάζοντας και συλλέγοντας απόψεις.

Ο σχεδιασμός ιστοσελίδων αλλάζει συνεχώς, με νέες τάσεις και τεχνολογίες να εμφανίζονται κάθε χρόνο.

Ο σχεδιαστής πρέπει να ενημερώνει συνεχώς τις γνώσεις του - να κοιτάξει τι είναι τώρα πιο σε ζήτηση, ποιες τεχνολογίες έχουν εμφανιστεί και επίσης να κατανοήσει αρκετούς σχετικούς τομείς προκειμένου να δημιουργήσει σύγχρονα προϊόντα. Επομένως, ένας σχεδιαστής δεν είναι μόνο καλλιτέχνης, αλλά και αναλυτής, σχεδιαστής, λεπτός ψυχολόγος.



Βασικές ευθύνες ενός Web Designer:

- ❖ Δημιουργία ή ενημέρωση της εμφάνισης της πύλης (συμπεριλαμβανομένων μεμονωμένων σελίδων).
- ❖ Σχεδιάζοντας τη λογική του.
- ❖ Δημιουργία διατάξεων ιστού του μελλοντικού ισότοπου.
- ❖ Επεξεργασία της εμφάνισης σελίδων για διαφορετικές συσκευές.
- ❖ Εργασία με βασικές γλώσσες προγραμματισμού.
- ❖ Χρήση βασικών επεξεργαστών γραφικών.
- ❖ Συναντήσεις με πελάτες για συζητήσεις.

Ένας ισότοπος μπορεί να οριστεί ως μια συλλογή βασικών στοιχείων. Ο σχεδιασμός και η παραγωγή των στοιχείων αυτών είναι η βασική φάση της κατασκευής ιστότοπων. Τα περισσότερα από τα στοιχεία μπορούν να επεξεργαστούν από τον ίδιο το πρόγραμμα περιήγησης χωρίς εξωτερικά προγράμματα ή πρόσθετα. Για παράδειγμα, τα περισσότερα προγράμματα περιήγησης μπορούν να εμφανίσουν τις μορφές PNG, JPG και GIF. Τα αρχεία βίντεο όπως το FLV και το SWF μπορούν επίσης να αναπαραχθούν από προγράμματα περιήγησης. Το μουσικό αρχείο όμως όπως το mp3 πρέπει να μεταφορτωθεί στον τοπικό δίσκο και να εκτελεστεί με το κατάλληλο εξωτερικό πρόγραμμα.

Το plug-in του προγράμματος περιήγησης μπορεί να αναπαράγει περισσότερες μορφές αρχείων πολυμέσων. Το πρόγραμμα περιήγησης kernel της IE της Microsoft παρέχει τεχνολογία βασισμένη σε OLE-ActiveX στη σελίδα πολυμέσων. Το ActiveX έχει πλέον γίνει μια δημοφιλής τεχνολογία. Μια άλλη τεχνολογία πολυμέσων είναι JavaApplet. Είναι γραμμένο σε γλώσσα Java που χρησιμοποιείται σε εφαρμογές ιστού, η οποία είναι μικρότερη σε σύγκριση με plug-ins και ActiveX, το JavaApplet



έχει μεγαλύτερη ευελιξία και ικανοποιητικές δυνατότητες πολλαπλών πλατφορμών.

Διαφορετικά layout

Τα layout στον σχεδιασμό μιας σελίδας είναι πολύ βασικό αφού καθορίζει την τοποθέτηση των στοιχείων μέσα στον ισότοπο. Η ανάλυση 1024x768 που κάποτε ήταν η συνηθισμένη έχει, πλέον, εκθρονιστεί από αναλύσεις high definition. Υπάρχει, λοιπόν, μια συνεχώς αυξανόμενη ροπή στη χρήση σωστού user experience design.

Είναι ανεπίτρεπτο μια σελίδα να μην μπορεί να προσαρμοστεί σε όλες τις αναλύσεις και να παρουσιάζει προβλήματα προβολής.

Fixed Layout

Fixed Layout ή στατικό layout είναι η σχεδίαση στην οποία το μήκος του περιεχόμενου έχει οριστεί σύμφωνα με μια προορισμένη ανάλυση και δεν αλλάζει σύμφωνα με την οθόνη και την ανάλυση που ο χρήστης χρησιμοποιεί.

Όμως Fixed Layout αποτελεί πλέον παρελθόν γιατί είναι διαμορφούμενο για οθόνες ανάλυσης 800x600 και 1024x768 pixels.

Fluid Layout

Fluid Layout ,το μήκος στο οποίο το περιεχόμενο θα εμφανιστεί εξαρτάται άμεσα από το μέγεθος του παραθύρου του φυλλομετρητή που χρησιμοποιείται, λόγω της χρήσης δομικών στοιχείων που καθορίζονται σε σχετικούς δείκτες, π.χ. σε ποσοστιαία κλίμακα αντί για στατικά εικονοστοιχεία. Το συγκεκριμένο layout είναι απαρχαιωμένο, επειδή λαμβάνεται υπόψη του συγκεκριμένος αριθμός συσκευών (και άρα αναλύσεων οθόνης) και δεν υπάρχουν γραμμές που να προβλέπουν μια καλή εμφάνιση σε όλο το μήκος των οθονών.

Τα κυριότερα μειονεκτήματα που έρχονται με τη χρήση fluid layout είναι:

- i) Το γεγονός ότι είναι αδύνατο να προβλεφθεί η ακριβής εμφάνιση της σελίδας σε διαφορετικές αναλύσεις οθονών.
- ii) Το γεγονός ότι οθόνες μεγαλύτερης ανάλυσης, όταν χρησιμοποιείται fluid layout, τείνουν να δημιουργούν μεγάλες περιόδους λόγου, δυσανάγνωστες, οι οποίες απέχουν πολύ μεταξύ τους.
- iii) Οι αυξημένες πιθανότητες να παρουσιαστεί το περιεχόμενο της ιστοσελίδας με διαφορετικές προδιαγραφές από αυτές που αναμενότουσαν.

Προσαρμοστικό (adaptive) layout

Προσαρμοστικό (adaptive) web design (AWD) προωθεί τη δημιουργία πολλαπλών εκδόσεων μιας ιστοσελίδας για να ταιριάζει καλύτερα στη συσκευή του χρήστη, σε αντίθεση με μια ενιαία έκδοση που χρησιμοποιούν οι παραπάνω τεχνικές σχεδιασμού ιστοσελίδων. Ο προσαρμοστικός σχεδιασμός ιστοσελίδων

περιλαμβάνει μια σειρά από άλλες στρατηγικές που μπορούν να συνδυαστούν με το responsive web design (περισσότερα παρακάτω). Με το adaptive web design, κυρίως κάνουμε αναφορά σε μια διαδικασία ανίχνευσης από την πλευρά του διακομιστή, η οποία επιλέγει μια διάταξη σχεδίασης και μέγεθος για εμφάνιση. Όλα τα τυποποιημένα σχέδια διαμόρφωσης ιστοσελίδων έχουν ενσωματωθεί στη διάταξή τους. Ο προσαρμοστικός σχεδιασμός θα εξυπηρετεί διαφορετικές εκδόσεις του ισότοπου σε διαφορετικές συσκευές βασισμένες σε κοινά μεγέθη οθόνης και αναλύσεις (Gustafson, 2011). Αυτός ο προσαρμοστικός σχεδιασμός είναι μια ευρεία προσέγγιση στο σχεδιασμό ιστοσελίδων που επικεντρώνεται στην καταλληλότητα για μια ποικιλία διεπαφών, αντί να περιορίζεται στην μορφή που προορίζεται για μια συγκεκριμένη ανάλυση οθόνης. Αυτό είναι ιδιαίτερα σημαντικό, τώρα που η χρήση *smartphone* τηλεφώνων έχει αυξηθεί, με τα *tablets* να ακολουθούν. Νέες τεχνολογίες έχουν προκύψει, φέρνοντας την αλλαγή στο πρόσωπο του σχεδιασμού ιστοσελίδων στο σύνολό του και δίνοντας το προβάδισμα στο *adaptive layout*, ως μια πολύτιμη ιδέα. Αυτές οι πρόσφατες αλλαγές κάνουν το συγκεκριμένο τρόπο σχεδιασμού ιστοσελίδας μια πρακτική που χρησιμοποιείται συνήθως στον κυβερνοχώρο, με περίπου 70.000 νέες ιστοσελίδες που δημιουργούνται κάθε μέρα.

Στο προσαρμοστικό layout, χρησιμοποιούνται editors του περιεχομένου, που επιτρέπουν την επεξεργασία της σελίδας. Οι πιο διαδεδομένοι editors είναι αυτοί του fluid motion. Βασίζονται σε drag-and-drop διαδικασία HTML5.

Επιτρέπουν την τοποθέτηση του περιεχομένου οπουδήποτε μέσα στη σελίδα ενώ τα πλεονεκτήματα περιλαμβάνουν:

- i) Κίνηση χωρίς μείωση framerates
- ii) 3D layout
- iii) Layout πολλαπλών σελίδων
- iv) Ξεχωριστό σχεδιασμό για φορητές συσκευές
- v) Εκτύπωση
- vi) Προσαρμογή του layout

vii) Διατήρηση του layout

viii) Zoom.

Responsive layout

Responsive layout χρησιμοποιεί λιγότερες διατάξεις σελίδας από το συνηθισμένο προσαρμοστικό layout. Η συγκεκριμένη προσέγγιση θεωρείται λιγότερο «ασφαλής» όσον αφορά το μέλλον, κυρίως επειδή το μέγεθος και η ανάλυση των οθονών των φορητών συσκευών συνεχώς αλλάζει και βελτιώνεται. Στο responsive layout οι σελίδες περιλαμβάνουν διάφορες διατάξεις και εφαρμόζεται κάθε φορά αυτή που δύναται να προβληθεί καλύτερα στην οθόνη του χρήστη.

Από την άλλη πλευρά, το responsive layout είναι πιο ασφαλές όσον αφορά τον εντοπισμό της σωστής διάταξης, ανάλογα με την οθόνη. Το responsive web design αποτελεί, κυρίως, μια προσέγγιση δημιουργίας ιστοσελίδων στην

οποία αναπτύσσεται ένας ισότοπος με την προσδοκία να παρέχει εύκολη και φιλική προς το χρήστη σχεδίαση. Επίσης συμπεριλαμβάνεται η εύκολη προβολή της ιστοσελίδας, με ελάχιστη πάντοτε αλλαγή μεγέθους και παροχή επιπλέον δυνατοτήτων, σε ένα ευρύ φάσμα συσκευών, φορητών και μη (Maximiliano, 2010). Για αυτό το λόγο, το responsive web design θα πρέπει να διέπεται από τις ακόλουθες αρχές (Gasston, 2013):

- ❖ Στην ανάπτυξη ιστοσελίδων με responsive web design, θα πρέπει να χρησιμοποιείται αυστηρά HTML 4/5, ή CSS3, χωρίς κάποια αναφορά σε JavaScript ή οποιοδήποτε JavaScript framework, στο οποίο θα στηρίζεται η διαδικασία αναγνώρισης των στοιχείων.
- ❖ Το responsive layout είναι αυτό που καθορίζει πώς τα στοιχεία της ιστοσελίδας θα εμφανίζονται στην κάθε οθόνη - παρ' όλα αυτά, αυτά τα ίδια στοιχεία δε θα πρέπει να κρύβονται ή να αποκαθίστανται από άλλα, διαφορετικά, ενώ η λειτουργία τους θα πρέπει να παραμένει ως έχει.
- ❖ Το responsive web design θα πρέπει να βρίσκεται σε θέση να αλλάζει, ανάλογα με το μέγεθος και την ανάλυση της οθόνης.
- ❖ Όταν τα στοιχεία θα πρέπει να εμφανίζονται χωρίς να χάνεται ποιότητα.

Το responsive web design, είναι αυτό που συνιστά η Google. Ο κολοσσός συνεχίζει να αλλάξει τον αλγόριθμο της μηχανής αναζήτησης, ώστε να συμπεριλάβει τον αυξανόμενο αριθμό των χρηστών smartphones και να λάβει υπόψη το πώς μια ιστοσελίδα που έχει έκδοση φορητής συσκευής είναι φιλική στο χρήστη, έναντι μιας σελίδας που δε παρέχει αυτή τη δυνατότητα.

Ένα responsive web design μπορεί να βοηθήσει το σχεδιαστή να διασφαλίσει ότι ο ισότοπός παρέχει τόσο καλή εμπειρία χρήστη όσο και επαρκή χρόνο φόρτωσης όταν προβάλλεται σε τηλέφωνο, tablet ή άλλες κινητές συσκευές.

Responsive layout και viewports

Το CSS3 αποτελείται από διάφορες ενότητες (modules). Μία από αυτές τις ενότητες είναι τα ερωτήματα που εξετάζουν τις δυνατότητες της οθόνης στο μέσον στο οποίο προβάλλονται (media queries). Με τη βοήθεια των media queries, το responsive web design καθορίζει ποια έκδοση θα εμφανιστεί στην οθόνη του χρήστη. *Με την χρήση του CSS ο σχεδιαστής μπορεί, μέσα από λίγες γραμμές κώδικα να αλλάξει τον τρόπο που προβάλλεται το περιεχόμενο στην οθόνη βασιζόμενος στο πλάτος της ωφέλιμης οθόνης της συσκευής (viewport width).* Στην αναλογία της οθόνης (aspect ratio) ή στον προσανατολισμό της οθόνης (orientation, portrait or landscape).

Το βασικό πλεονέκτημα των media queries είναι η δυνατότητα που έχουν να προσαρμόζουν το στυλ, ανάλογα με τη συσκευή που ο χρήστης χρησιμοποιεί. Για να γίνει δυνατό αυτό, τα media queries χρησιμοποιούν τις εντολές που έχουν να κάνουν με τις διαστάσεις συσκευής.

Οι εντολές που χρησιμοποιούν τα media queries έχουν να κάνουν με:

- i) το μήκος του viewport
- ii) το ύψος του viewport
- iii) το μήκος της συσκευής
- iv) το πλάτος της συσκευής
- v) τον προσανατολισμό
- vi) το aspect ratio

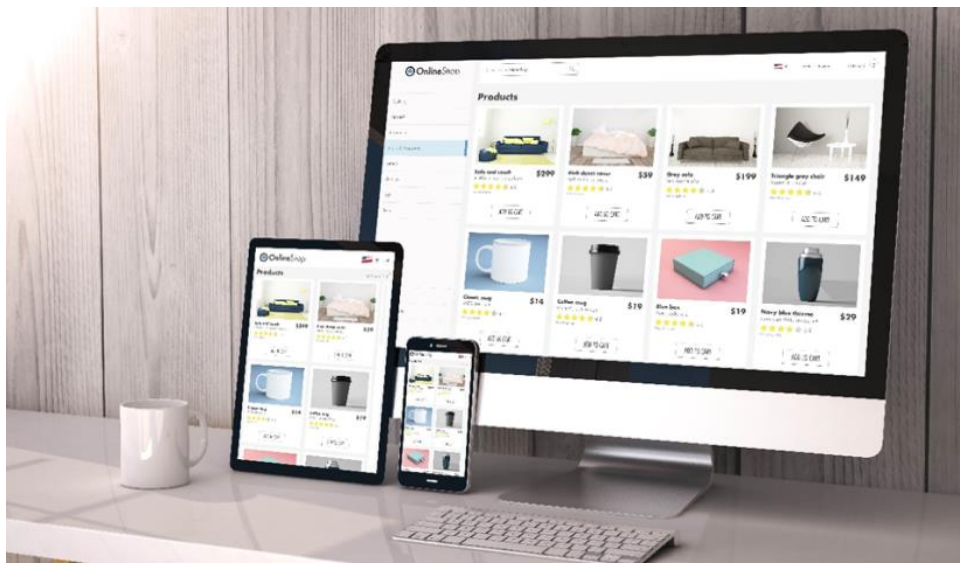
vii) το device-aspect ration, δηλαδή το μήκος και το ύψος της οθόνης της συσκευής

viii) το χρώμα

ix) το μονόχρωμα

x) την ανάλυση της οθόνης

Με τα παραπάνω χαρακτηριστικά καθορίζονται οι ελάχιστες και μέγιστες τιμές των media queries. Τα media queries είναι σημαντικά στο responsive web design, καθώς με αυτά καθορίζεται ο τρόπος εμφάνισης της πληροφορίας στην οθόνη του χρήστη.



ΔΗΜΙΟΥΡΓΙΑ LAYOUT

Υπάρχουν αρκετοί τρόποι να δημιουργήσει ο σχεδιαστής layouts, αλλά η πιο συνηθισμένη είναι αυτή του CSS (Robbins, 2007).

Υπάρχουν τρεις τρόποι να επιτευχθεί κάτι τέτοιο:

- i) liquid (προσαρμογή μεγέθους ανάλογα με το μέγεθος της οθόνης)
- ii) fixed (το περιεχόμενο είναι συγκεκριμένου μεγέθους και δεν αλλάζει)
- iii) elastic (μερικές περιοχές του περιεχομένου αλλάζουν όταν το κείμενο αλλάζει).

Αν και οι τρεις ανωτέρω δίοδοι λειτουργούν αποδοτικά, ο σχεδιαστής διαλέγει αυτή που ανταποκρίνεται καλύτερα στο περιεχόμενο της ιστοσελίδας. Ένα σχεδιάγραμμα της ιστοσελίδας βοηθά, επίσης, σε καλύτερη οργάνωση του ιστότοπου,

δημιουργώντας ένα σύστημα στηλών, ορίων και τμημάτων, μέσα στα οποία το περιεχόμενο μπορεί να τοποθετηθεί με διαφορετικούς τρόπους.

Ένας τρόπος εξισορρόπησης του design είναι να λάβει υπόψη του ο σχεδιαστής το ύψος των στοιχείων. Τα σημαντικότερα είναι τα μεγαλύτερα στοιχεία με το κυρίως περιεχόμενο της σελίδας. Σε ένα design με συμμετρική ισορροπία, το βάρος αυτό μοιράζεται ισότοπα στη σελίδα. Αντίθετα σε ένα ασυμμετρικό design το αποτέλεσμα δεν είναι αισθητικά ικανοποιητικό, επειδή δημιουργούνται whitespaces ανάμεσα στα στοιχεία της σελίδας. Αυτή η τεχνική θα πρέπει να χρησιμοποιείται μόνο

σε περιπτώσεις που ο σχεδιαστής θέλει να τραβήξει την προσοχή του επισκέπτη σε ένα συγκεκριμένο στοιχείο στο website.

Κείμενο

Η κύρια λειτουργία του κειμένου στην οπτική επικοινωνία είναι η μετάδοση των πληροφοριών και η επίτευξη του στόχου. Ο σχεδιασμός του κειμένου πρέπει να αποφεύγει την πολυπλοκότητα, ακόμη κι αν οι άνθρωποι μπορούν να κατανοήσουν.

Ο θεμελιώδης σκοπός της γραφής είναι να σχεδιαστεί καλύτερα και πιο αποτελεσματικά να μεταδοθεί η πρόθεση του συγγραφέα να εκφράσει το θέμα και τις ιδέες. Για παράδειγμα, κανείς δεν είναι διατεθειμένος να διαβάσει μια ιστοσελίδα με τεράστιο περίπλοκο κείμενο.

Το μέγεθος γραμματοσειράς μπορεί να υπολογιστεί με διάφορους τρόπους, όπως το σημείο και το πίξελ - για το λόγο ότι η τεχνολογία των πίξελ που βασίζεται σε υπολογιστή πρέπει να μετατρέψει το σημείο για εκτύπωση, συνιστάται να χρησιμοποιηθεί το σημείο.

Το πιο κατάλληλο μέγεθος γραμματοσειράς για το σώμα της σελίδας είναι δώδεκα (12) πόντοι, ενώ για τις σελίδες που πρέπει να εμφανίζονται περισσότερο συνήθως χρησιμοποιείται το μέγεθος της γραμματοσειράς των εννέα (9) σημείων.

Μεγαλύτερη γραμματοσειρά μπορεί να χρησιμοποιηθεί για να τονίσει τον τίτλο ή άλλα μέρη, μικρότερες γραμματοσειρές μπορούν να χρησιμοποιηθούν για το

υποσέλιδο και υποστηρικτικές πληροφορίες. Το μικρό μέγεθος γραμματοσειράς μπορεί εύκολα να παράγει αίσθηση εξαιρετικής σχεδίασης, αλλά η αναγνωσιμότητα μπορεί να είναι κακή.

Οι αλλαγές στη διαχωριστική γραμμή θα έχουν επίσης μεγάλη επίδραση στην αναγνωσιμότητα του κειμένου. Γενικά, είναι κατάλληλο να ορίσει ο σχεδιαστής την απόσταση γραμμών κοντά στο μέγεθος της γραμματοσειράς. Ο κανονικός λόγος διαχωρισμού είναι 10:12, δηλαδή δέκα (10) λέξεις με απόσταση γραμμής δώδεκα (12) μονάδων. Η σωστή απόσταση θα σχηματίσει ένα σημαντικό επίπεδο κενών χώρων για να καθοδηγήσει την προσοχή του θεατή, αλλά σε πολύ μεγάλη απόσταση θα χάσει τη συνέχεια.

Το κείμενο πρέπει να έχει το δικό του χαρακτήρα: ο σχεδιασμός του κειμένου πρέπει να εξαρτάται από το κύριο ύφος. Στο σχεδιασμό ιστοσελίδων, οι σχεδιαστές μπορούν να ορίσουν κείμενο, συνδέσμους κειμένου, συνδέσμους επισκεπτών και συνδέσμους που έχουν επιλεγεί με διάφορα χρώματα. Για παράδειγμα, αν ο σχεδιαστής χρησιμοποιεί τον επεξεργαστή Frontpage, η προεπιλεγμένη ρύθμιση έχει ως εξής: η κανονική γραμματοσειρά είναι μαύρη, ο προεπιλεγμένος

σύνδεσμος είναι μπλε και το κείμενο γίνεται μωβ μετά το κλικ. Ωστόσο, θα πρέπει να σημειωθεί ότι, η υπερβολική χρήση των χρωμάτων για το κείμενο πιθανότατα να ενοχλήσει τον επισκέπτη και να τον αποτρέψει από το να διαβάσει τα περιεχόμενα, εκτός αν υπάρχει ειδικός σκοπός σχεδιασμού, ο οποίος θα πρέπει να γίνει γνωστός.

Η αντίθεση χρώματος κειμένου, είναι επίσης σημαντική και περιλαμβάνει τη φωτεινότητα, την καθαρότητα και τη σύγκριση των κρύων και ζεστών χρωμάτων. Αυτά όχι μόνο παίζουν ρόλο στην αναγνωσιμότητα του κειμένου, αλλά υπάρχει η δυνατότητα ο σχεδιαστής να χρησιμοποιήσει χρώμα για να επιτύχει το επιθυμητό αποτέλεσμα του σχεδιασμού.

Φόντο

Το φόντο του σχεδιασμού ιστοσελίδας είναι πολύ σημαντικό, ειδικά για την προσωπική αρχική σελίδα. Το φόντο ενός σπιτιού είναι ισοδύναμο με τον τοίχο ενός δωματίου - όπως ένα καλό φόντο μπορεί όχι μόνο να αφήνει στους επισκέπτες

καλές εντυπώσεις αλλά και να επηρεάσει την ιδέα που θα έχει ο επισκέπτης για το σπίτι, έτσι και το φόντο της αρχικής σελίδας του ιστότοπου αφήνει θετικές ή αρνητικές εντυπώσεις, ανάλογα με το πώς έχει σχεδιαστεί.

Ακόμη και ένας ιστότοπος με διαφορετικές σελίδες, έχει διαφορετικό σχεδιασμό φόντου.

Συμπαγές χρώμα ως φόντο

Το συμπαγές έγχρωμο υπόβαθρο είναι απλό, αλλά είναι το πιο συνηθισμένο, επειδή σε σύγκριση με την εικόνα φόντου, έχει πλεονεκτήματα από την ταχύτητα εμφάνισης. Το αρχείο σελίδας, συνήθως χρησιμοποιώντας την ετικέτα <body> για να καθορίσετε το χρώμα φόντου της

σελίδας, η σύνταξη HTML είναι:

```
<body bgcolor = "χρώμα">
```

Το «χρώμα» σημαίνει διαφορετικά χρώματα και μπορεί κανείς να χρησιμοποιήσει μια ποικιλία διαφορετικών χρωμάτων. Ο ευκολότερος τρόπος είναι απλά να πληκτρολογήσει ο σχεδιαστής κοινά ονόματα αγγλικών, όπως μπλε, κίτρινο, μαύρο κ.λπ., Ωστόσο μπορεί επίσης να χρησιμοποιήσει δεκαεξαδικούς κωδικούς χρώματος, όπως #0000FF, #FFFF00, #000000, κλπ..

Οποιαδήποτε μέθοδος δίνει το ίδιο αποτέλεσμα.

Φόντο σε μορφή κύματος

Στην πραγματικότητα, το κυματιστό φόντο είναι απλώς ο συνδυασμός και η επανάληψη των ίδιων εικόνων. Αυτός είναι ο τρόπος με τον οποίο το πρόγραμμα περιήγησης επεξεργάζεται την εικόνα φόντου και λόγω αυτού γίνεται να χρησιμοποιηθεί μια εικόνα μικρού μεγέθους ως σελίδα παρασκηνίου. Η ίδια διαδικασία επαναλαμβάνεται αυτόματα και καλύπτει ολόκληρη τη σελίδα, έτσι ώστε να μειώνεται σημαντικά ο χρόνος λήψης των ιστοσελίδων.

Κουμπιά

Ο σχεδιαστής συνηθώς δεν δίνει σημασία στον σχεδιασμό των κουμπιών, όμως ένας σωστός σχεδιασμός τους μπορεί να αλλάξει εντελώς την εικόνα και την εμπειρία χρήσης της ιστοσελίδας.

Πλοήγηση

Μια σωστή πλοήγηση στον ιστότοπο είναι το πιο σημαντικό έργο που πρέπει να επιτύχει ένας σχεδιαστής ιστοσελίδων στον σχεδιασμό τους.

Στον ιστότοπο η πλοήγηση είναι η διαδρομή που ακολουθούν οι χρήστες για να περιηγηθεί σε αυτόν και στα υπομενού. Είναι σημαντικό, επομένως, το web site να είναι καλά κατασκευασμένο, εύκολο στη χρήση και διαισθητικό. Αυτό επιτυγχάνεται με μια σωστή σχεδιαστική γραμμή πλοήγησης και η κακή πλοήγηση δεν βοηθά τους χρήστες, αφού συχνά ο ιστότοπός μπορεί να αποδειχθεί λιγότερο προσβάσιμος από άλλους, απομακρύνοντας τον επισκέπτη (McNeil, 2014).

Η καλή πλοήγηση είναι θεμελιώδης για τον καλό σχεδιασμό ιστοσελίδων - τόσο σε επιχειρηματικούς όσο και σε ενημερωτικούς χώρους. Οι χρήστες θα πρέπει να μπορούν να βρίσκουν εύκολα πληροφορίες. Εάν η πλοήγηση δεν είναι εύκολη στη χρήση, οι χρήστες θα πάνε γρήγορα αλλού, προκειμένου να κάνουν αναζήτηση πληροφοριών. Η πλοήγηση είναι τόσο καλά κρυμμένη ή συγκαλυμμένη, ώστε ορισμένοι χρήστες απλά δεν ξέρουν πώς να την χρησιμοποιήσουν. Η πλοήγηση είναι το πιο σημαντικό στοιχείο στη δημιουργία προσβάσιμων και χρησιμοποιήσιμων ιστότοπων (McNeil, 2014).

Συχνά, ο σχεδιαστής δημιουργεί έναν ιστότοπο γεμάτο πληροφορίες και το παρουσιάζει με τέτοιο τρόπο ώστε να πρέπει ο χρήστης να ξοδέψει αρκετή ώρα μαθαίνοντας τον ιστότοπο, πριν μπορέσει να αλληλεπιδράσει μαζί του. Είναι απαραίτητο οι πληροφορίες να οργανώνονται σε απλές δομές, ακριβώς επειδή οι περισσότεροι χρήστες κατανοούν ακριβώς αυτό.

Μουσική Υπάρχουν δύο τρόποι εισαγωγής μουσικής στην ιστοσελίδα - με την ετικέτα <bgsound> ή με την ετικέτα <embed> (Robbins, 2014).

Η ετικέτα «bgsound» περιέχει πέντε (5) ιδιότητες και του οποίου η «ισορροπία» αντιπροσωπεύει την ισορροπία του αριστερού και του δεξιού καναλιού. Η

«καθυστέρηση» σημαίνει καθυστέρηση αναπαραγωγής του ήχου, με το «loop» γίνεται αναφορά στον αριθμό των κύκλων που το κομμάτι προς αναπαραγωγή θα κάνει, ενώ το «volume src» είναι η διαδρομή του αρχείου.

Αυτός είναι ο πιο συνηθισμένος τρόπος προσθήκης μουσικής υπόκρουσης και υποστηρίζει διάφορους τύπους αρχείων μουσικής όπως MP3, WAV και MID.

Ένας άλλος τρόπος ενσωμάτωσης της μουσικής στην ιστοσελίδα χρησιμοποιεί την ετικέτα <embed>. Η ετικέτα αυτή έχει περισσότερες δυνατότητες από την ετικέτα <bgound> και, σε συνδυασμό με κάποια πρόσθετα μουσικής, μπορεί να εμφανιστεί ως web player.

Τύπος και Μέγεθος Αρχείων Όλα τα στοιχεία μιας ιστοσελίδας θα μεταφορτωθούν από τα προγράμματα περιήγησης του επισκέπτη, επομένως το μέγεθος του αρχείου αυτών των εικόνων και των εικονιδίων επηρεάζει άμεσα την ταχύτητα λήψης.

Διάφορες μορφές εικόνας χρησιμοποιούνται σήμερα όπως JPEG (JPG), BMP, PNG, GIF κλπ.. Όλοι αυτοί οι τύποι μπορούν να προβληθούν με προγράμματα περιήγησης. Για τους ιστοτόπους χρησιμοποιούνται συνήθως οι εικόνες JPG, GIF και PNG. Η μορφή JPG και GIF έχει μεγάλο βαθμό συμπίεσης που θα είναι πολύ μικρότερη στο ίδιο επίπεδο οπτικής επίδρασης σε σύγκριση με άλλες μορφές. Η χρήση του GIF ή του JPG εξαρτάται από το μέγεθος των χρωμάτων όπου το GIF είναι κατάλληλο για εικόνες με λιγότερα χρώματα. Η μορφή PNG μπορεί να δώσει ένα διαφανές υπόβαθρο και αυτός είναι ο βασικός λόγος για τη χρήση της.

Χρώματα Το χρώμα στο είναι ένας εξαιρετικός τρόπος για να αποκτήσει μια ιστοσελίδα άλλο πρόσωπο. Μια οθόνη υπολογιστή αποτελείται από χιλιάδες μικροσκοπικά μικρά κόκκινα, πράσινα και μπλε πίξελ, που ομαδοποιούνται ανά τρία. Αυτές οι μικρές κουκίδες είναι κοντά μεταξύ τους που με γυμνό μάτι και, λαμβάνοντας πλέον υπόψη την ανάλυση των συσκευών δύσκολα παρατηρούνται. Τα πίξελ μπορούν να απεικονίσουν όλα τα χρώματα. Για παράδειγμα, αν οι κόκκινες και πράσινες κουκίδες λάμπουν στο 100% της δύναμής τους, ενώ το μπλε δεν ανάβει καθόλου, θα δούμε το αποτέλεσμα ως ένα καθαρό κίτρινο χρώμα.

Τα HTML χρώματα είναι ένας συνδυασμός κόκκινου, πράσινου και μπλε φωτός. Ο σχεδιαστής μπορεί να χρησιμοποιήσει οποιοδήποτε αριθμό από το 0 έως το 255 και να πάρει σαν αποτέλεσμα έναν κωδικό RGB που μπορεί να χρησιμοποιηθεί για χρώμα. Αυτό σημαίνει ότι υπάρχουν δεκαέξι (16) εκατομμύρια χρώματα για να διαλέξει κανείς, ενώ όλες οι σύγχρονες οθόνες έχουν τη δυνατότητα αναπαράστασης τουλάχιστον 16 χιλιάδων χρωμάτων.

Ορισμένοι σχεδιαστές χρησιμοποιούν δεκαεξαδικούς κωδικούς χρωμάτων. Οι κωδικοί αυτοί είναι ομάδες τριών δεκαεξαδικών αριθμών. Ο πρώτος αριθμός είναι κόκκινου χρώματος, ο δεύτερος αντικατοπτρίζει το πράσινο και ο τρίτος το μπλε. Κάθε αριθμός είναι μια τριπλέτα τύπου RGB. Η χαμηλότερη τιμή είναι το 00 και το FF είναι η υψηλότερη.

Ο κωδικός για το κόκκινο είναι 0000FF, ο κωδικός για το πράσινο είναι 00FF00 και ο κωδικός για το μωβ είναι 990099. Μια άλλη μέθοδος προσθήκης χρωμάτων στην ιστοσελίδα είναι η χρήση κώδικα CSS. Η πιο συνηθισμένη μέθοδος είναι η χρήση του tag <style>.

Για παράδειγμα, με τη χρήση CSS, το χρώμα τοποθετείται με την ακόλουθη γραμμή:

```
<span style="color : #ff0000">red</span>
```

Το βασικότερο στοιχείο που ο σχεδιαστής θα πρέπει να έχει κατά νου, είναι το γεγονός ότι θα πρέπει η ιστοσελίδα να μην έχει περισσότερα από πέντε (5) χρώματα. Ακόμη κι αν όλα είναι της ίδιας απόχρωσης σε διαφορετικούς τόνους, είναι περισσότερο πιθανό η πλοήγηση να είναι κουραστική.

Σχεδιασμός πρώτης ιστοσελίδας

Ο σχεδιασμός του περιεχομένου είναι ένα κρίσιμο σημείο το οποίο πρέπει να προσεχτεί από την αρχή της δημιουργίας μιας ιστοσελίδας, διότι η οποιαδήποτε απόφαση θα είναι καθοριστικής σημασίας. Η οποιαδήποτε διαφοροποίηση στο μέλλον θα πολλαπλασιάσει το κόστος, καθώς θα απαιτηθούν σημαντικά περισσότερες ανθρωπόωρες.

Τα βήματα που είναι τα παρακάτω

Σχεδιάζουμε μια σαφή ιεράρχηση του περιεχομένου: Μια από τις βασικές αρχές του σωστού σχεδιασμού, είναι ότι η ιεράρχηση δεν θα πρέπει να έχει μεγάλο βάθος. Αυτό κάνει την ιστοσελίδα σας πολύ πιο εύχρηστη και δίνει τη δυνατότητα στις μηχανές να την ανιχνεύσουν ταχύτερα και ευκολότερα.

Δημιουργούμε έναν χάρτη της ιστοσελίδας: Η δημιουργία και ύπαρξη ενός χάρτη δίνει την δυνατότητα στους επισκέπτες της ιστοσελίδας να περιηγηθούν ευκολότερα σε αυτήν και επιπλέον διευκολύνει τη σάρωσή της από τις μηχανές αναζήτησης. Χάρτης ιστοσελίδας, είναι μία λίστα με όλες τις σελίδες που υπάρχουν σε μία ιστοσελίδα για να μπορεί ένας χρήστης να προηγηθεί πιο εύκολα σε αυτή.

Προσδιορίζουμε τις κατάλληλες λέξεις κλειδιά (Keywords): Επιλέγουμε από την αρχή τις κατάλληλες λέξεις-κλειδιά (Keywords), με τις οποίες θέλουμε να εντοπίζεται η σελίδα μας από τις μηχανές αναζήτησης και τις οποίες θα πρέπει να χρησιμοποιούμε συχνά στο περιεχόμενό μας. Οι λέξεις-κλειδιά θα πρέπει να αποτελούν το 6 % των συνολικών λέξεων του περιεχομένου της ιστοσελίδας μας.

Τελικός στόχος είναι η υψηλή κατάταξη του ιστοχώρου και η αύξηση της επισκεψιμότητας της ιστοσελίδας. Ένα από τα σημαντικότερα στοιχεία που πρέπει να συμπεριλαμβάνεται στην ιστοσελίδα, ώστε να βοηθά στην καλύτερη κατάταξή της στις μηχανές αναζήτησης, είναι η συνάφεια (relevancy) μεταξύ του τίτλου, της περιγραφής, των λέξεων κλειδιών και του περιεχομένου της κάθε σελίδας.

Η χρήση των λέξεων-κλειδιών είναι ζωτικής σημασίας για την επισήμανση ενός ιστοτόπου από τις μηχανές αναζήτησης. Αν στον ιστοτόπο περιλαμβάνονται περισσότερες από μια σελίδες, οι λέξεις-κλειδιά πρέπει να εμφανίζονται σε όλες. Βασικός παράγοντας, επίσης, είναι οι λέξεις-κλειδιά να συμπεριλαμβάνονται όσο το δυνατό περισσότερες φορές στον κορμό του κυρίως κειμένου της σελίδας (body text). Αυτό συμβαίνει επειδή οι περισσότερες μηχανές αναζήτησης τις "ψάχνουν" και στο κυρίως κείμενο. Αν δε συναντήσουν μια λέξη-κλειδί στο κυρίως κείμενο, απλά την αγνοούν.

Δίνουμε έμφαση στο κείμενο: Οι μηχανές αναζήτησης είναι ιδιαίτερα ευφυείς και μπορούν να διαβάσουν διάφορες μορφές από το κείμενο του περιεχομένου μας. Το κείμενο είναι αυτό που τελικά θα καταχωρήσουν.

Μπορεί να έχουμε εκατοντάδες εντυπωσιακές φωτογραφίες ή flash animations, αλλά αν δεν έχουμε κείμενο για να περιγράψει τα παραπάνω, τότε δυστυχώς όλα θα θεωρούνται ανύπαρκτα. Ίσως στο μέλλον να ξεπεραστεί αυτή η αδυναμία και να υπάρχει τρόπος ανίχνευσης και ταυτόχρονης αξιολόγησής τους.

Άρα, το μόνο χρήσιμο για την ώρα είναι το περιεχόμενο με κείμενο. Συνεπώς, θα πρέπει να δώσουμε μεγαλύτερη βαρύτητα σε πληροφοριακό περιεχόμενο που είναι κείμενο και λιγότερη σε ό,τι δεν είναι κείμενο. Θα πρέπει να προσέξουμε επίσης τις ετικέτες ALT tags των περιγραφών των φωτογραφιών αλλά και των συνδέσμων (links), οι οποίες ανιχνεύονται και διαβάζονται από τις μηχανές αναζήτησης, παρέχοντας σημαντικές πληροφορίες για τη φωτογραφία ή το σύνδεσμο (link).

Ελέγχουμε την ορθότητα του περιεχομένου: Κάνουμε έλεγχο και διασφαλίζουμε ότι δεν έχουμε σπασμένους συνδέσμους (broken links), συνδέσμους (links) δηλαδή που δεν οδηγούν πουθενά. Όταν υπάρχουν σπασμένοι σύνδεσμοι (broken links), σημαίνει ότι η ιστοσελίδα είναι σε εγκατάλειψη με αποτέλεσμα οι μηχανές αναζήτησης να την κατατάσσουν σε χαμηλότερη θέση σε σχέση με κάποιες που οι σύνδεσμοί τους λειτουργούν. Στη χειρότερη περίπτωση η ιστοσελίδα δεν θα μπορεί καθόλου να ανιχνευτεί από τις μηχανές αναζήτησης.

Εργαλεία δημιουργίας ιστοσελίδας

Κατασκευή ιστοσελίδων (αγγλικά: web development), είναι η διαδικασία δημιουργίας παρουσιάσεων περιεχομένου (συνήθως υπερκειμένου ή πολυμέσων), οι οποίες προβάλλονται στον τελικό χρήστη του Διαδικτύου μέσω ενός προγράμματος περιήγησης (browser) ή άλλων υπηρεσιών, όπως διαδικτυακή τηλεόραση, ιστολόγια (blogs) και RSS Feeds.

Μπορούν να χρησιμοποιηθούν διάφορες τεχνικές προβολής του περιεχομένου μιας σελίδας, μεταξύ αυτών: κινούμενα σχέδια, γραφιστική, αλληλεπίδραση ανθρώπου-υπολογιστή, μάρκετινγκ, φωτογραφία, βελτιστοποίηση μηχανών αναζήτησης και τυπογραφία.

Οι τεχνολογίες που χρησιμοποιούνται στην ανάπτυξή τους είναι:

- **γλώσσες σελίδας:** HTML και XHTML
- **φύλλα στυλ CSS**
- **γλώσσες περιγραφής δεδομένων:** XML, JSON
- **γλώσσες προγραμματισμού σεναρίων:** Perl, PHP, Python, Ruby
- **πλατφόρμες:** ASP και ASP.NET της Microsoft, Java Enterprise της Sun, agile
- **frameworks:** όπως το Django και το Ruby on Rails
- **συγγραφή κώδικα στην πλευρά του πελάτη:** Javascript

Η γλώσσα HTML

ΒΑΣΙΚΗ ΟΡΟΛΟΓΙΑ HTML

Τα βασικά δομικά στοιχεία που αποτελείται η HTML είναι οι ετικέτες (tags) και οι ιδιότητες (attributes).

ΕΤΙΚΕΤΕΣ(TAGS)

Είναι ο κώδικας της HTML που περικλείεται ανάμεσα στα σύμβολα <>.Είναι ένα είδος εντολής που καθορίζουν την εμφάνιση και τις ιδιότητες του κειμένου. Οι ετικέτες χωρίζονται επίσης σε δύο είδη, στις ετικέτες πλαίσια και στις κενές ετικέτες. Σε αντίθεση με τις κενές ετικέτες, οι οποίες δεν έχουν ετικέτα τέλους, οι ετικέτες πλαίσια περιέχουν κάποιο κείμενο ή άλλα στοιχεία. Οι ετικέτες πλαίσια αποτελούνται από δύο ετικέτες, την ετικέτα έναρξης και την ετικέτα τερματισμού.

ΙΔΙΟΤΗΤΕΣ (ATTRIBUTES)

Οι ιδιότητες είναι κώδικας που προσδιορίζει κάποια επιπλέον χαρακτηριστικά στην ετικέτα. Οι ιδιότητες παίρνουν τιμές οι οποίες εμφανίζονται μέσα στα σύμβολα " "

ΣΥΝΤΑΞΗ ΕΤΟΛΗΣ HTML

Για να συντάξουμε μία εντολή HTML ακολουθούμε έναν συγκεκριμένο τύπο:

<ετικέτα έναρξης ιδιότητα= "τιμή"> Περιεχόμενο ετικέτας <ετικέτα τερματισμού>

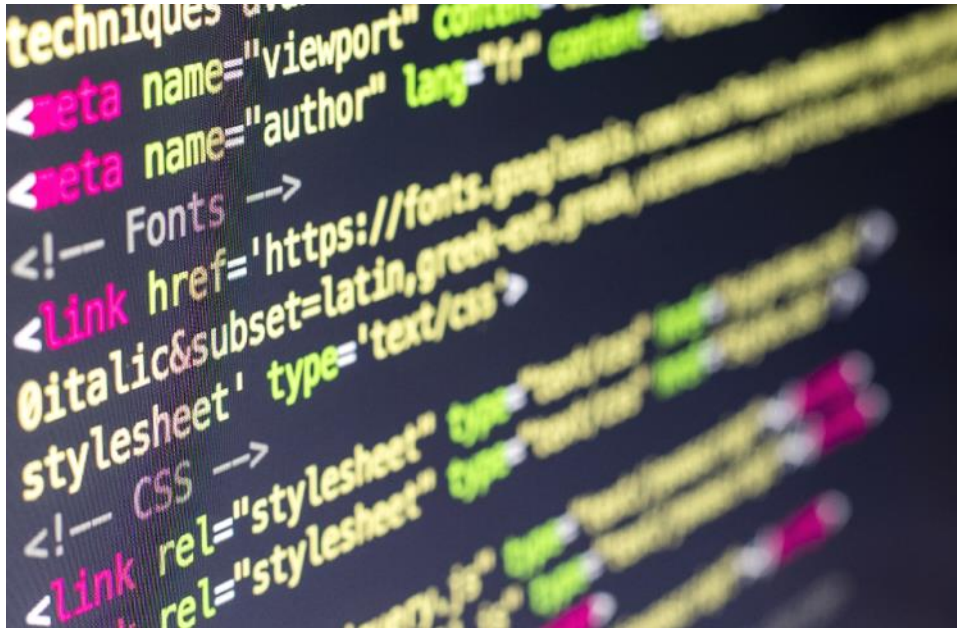
Η HTML (HyperText Markup Language, Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, τα στοιχεία της οποίας είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.

Η HTML γράφεται υπό μορφή στοιχείων που αποτελούνται από ετικέτες (tags), οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας.

Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ο σκοπός ενός web browser είναι να διαβάσει τα έγγραφα HTML και να τα συνθέσει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει.

Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να παρουσιάσει το περιεχόμενο της σελίδας. Στο παρόν κεφάλαιο, θα μελετήσουμε τη γλώσσα σήμανσης HTML και πιο συγκεκριμένα οι κύριοι κανόνες σύνταξής της, τα βασικά τμήματα των HTML



εγγράφων και οι τρόποι δημιουργίας αυτών. Επίσης, μελετώνται τα διαδοχικά φύλλα στυλ, που παρέχουν κανόνες μορφοποίησης των HTML εγγράφων, καθώς και οι κανόνες σύνταξης αυτών και παρουσιάζονται ορισμένα παραδείγματα σύνταξης κανόνων στυλ.

Η πρώτη δημόσια διαθέσιμη περιγραφή της HTML ήταν ένα έγγραφο με το όνομα Ετικέτες HTML, το οποίο πρωτοαναφέρθηκε στο Διαδίκτυο από τον Μπέρνερς Λι στα τέλη του 1991.

Περιέγραφε τα 20 στοιχεία τα οποία αποτελούσαν τον αρχικό, σχετικά απλό σχεδιασμό της HTML. Εκτός από την ετικέτα υπερσυνδέσμου, οι υπόλοιπες ήταν έντονα επηρεασμένες από την SGMLguid, μια μορφή δημιουργίας τεκμηρίωσης, φτιαγμένη στο CERN και βασισμένη στην SGML. Δεκατρία από εκείνα τα αρχικά στοιχεία υπάρχουν ακόμα σήμερα στην HTML 4.

Χρονικό εκδόσεων HTML

24 Νοεμβρίου 1995. Η HTML 2.0 δημοσιεύτηκε ως IETF RFC 1866.[12] Επιπλέον συμπληρωματικά RFC πρόσθεσαν δυνατότητες:

- 25 Νοεμβρίου 1995: RFC 1867 (ανέβασμα αρχείο από φόρμα)
- Μάιος 1996: RFC 1942 (πίνακες)
- Αύγουστος 1996: RFC 1980 (πελατοκεντικοί χάρτες εικόνων (client-side))

- Ιανουάριος 1997: RFC 2070 (διεθνοποίηση και τοπικοποίηση)
- Τον Ιούνιο του 2000, όλα τα ανωτέρω χαρακτηρίστηκαν ως ξεπερασμένα ή ιστορικά, από το RFC 2854.

Η γλώσσα σήμανσης HTML (Hypertext Markup Language) έχει σχεδιασθεί με σκοπό:

- Την στοιχειοθέτηση (μορφοποίηση-formatting) ηλεκτρονικών κειμένων – Την αναμετάδοσή τους στο Διαδίκτυο
- Την αναπαράστασή τους σε διαφορετικού είδους οθόνες Η περιγραφή της στοιχειοθέτησης του κειμένου γίνεται με την εισαγωγή ετικετών της HTML στο σώμα του εγγράφου. Οι περισσότερες ετικέτες της HTML χρησιμοποιούνται για την οργάνωση του περιεχομένου σε μια λογική ιεραρχική δομή. Υπάρχουν ωστόσο και καθαρά μορφολογικοί χαρακτήρες (π.χ. , <I>).

```
1
2
3 <!DOCTYPE html> <html lang="en-GB" <head> <meta content="text/html; charset=utf-8" http-equiv="content-type" <title>@mail</title> <meta
4 http-equiv="X-UA-Compatible" content="IE=edge" <meta name="application-name" content="@mail" <meta name="description" content="Google
5 3> approach to email" <meta name="application-url" content="https://mail.google.com/mail/u/0/" <meta name="google
6 content="notranslate" <link rel="canonical" href="https://mail.google.com/mail/" > <link rel="shortcut icon
7 href="mailto:0/images/favicon.ico" type="image/x-icon" <link rel="alternate" type="application/atom+xml" title="Mail Atom Feed"
8 href="feed/atom" </script type="text/javascript"
9
10 </script type="text/javascript"
11
12 </script type="text/javascript"
13
14 </script type="text/javascript"
15
16
17 <script> <style
18 var ssp=0;
19 </script> <body
20 </script> </body> </html>
```

Η HTML επιτρέπει την ενσωμάτωση υπερσυνδέσμων (links, δεικτών) προς άλλα κείμενα αλλά και αρχεία ήχου, εικόνων, κινουμένων σχεδίων κλπ. Έτσι, μπορεί να χρησιμοποιηθεί για το σχεδιασμό και την ανάπτυξη εφαρμογών υπερμέσων.

Ένα άλλο χαρακτηριστικό της HTML είναι ότι η γλώσσα αυτή είναι επεκτάσιμη, υπό την έννοια, ότι σε αυτή μπορούν να προστεθούν νέες ιδιότητες και λειτουργίες, χωρίς να ακυρώνονται υπερκείμενα αναπτυγμένα σε παλαιότερες εκδόσεις της. Παρακάτω, μπορείτε να δείτε τον κώδικα μια ιστοσελίδας σε HTML.

Αρχεία HTML

Τα αρχεία, τα οποία περιέχουν κώδικα HTML έχουν συνήθως κατάληξη «.html ή .htm». Τα αρχεία αυτά περιέχουν χαρακτήρες ASCII και μπορούν να δημιουργηθούν με έναν οποιονδήποτε επεξεργαστή κειμένου ASCII (editor), όπως ο vi και ο emacs, ο notepad κλπ. Υπάρχουν επίσης πληθώρα WYSIWYG εργαλείων συγγραφής HTML σε περιβάλλοντα Windows, UNIX και Apple, όπως FrontPage, HotMetal, NetObject Fusion, Net, DreamWeaver, κ.ά.

Προκειμένου να δημιουργήσει κάποιος μια ιστοσελίδα, είναι απαραίτητη η καλή γνώση των προδιαγραφών HTML, παράλληλα με τη χρήση κάποιου ισχυρού εργαλείου συγγραφής εφαρμογών HTML. Ο έλεγχος ενός αρχείου HTML μπορεί να γίνει «τοπικά» με τη χρήση φυλλομετρητή ιστού (web browser).

Σήμανση

Η σήμανση HTML, αποτελείται από μερικά βασικά συστατικά, συμπεριλαμβανομένων των στοιχείων (και των ιδιοτήτων τους) τους, βασισμένους σε χαρακτήρες τύπους δεδομένων, τις αναφορές χαρακτήρων και τις αναφορές οντοτήτων. Ένα ξεχωριστό σημαντικό συστατικό είναι η δήλωση τύπου εγγράφου (document type declaration), η οποία ορίζει στον browser τον τρόπο εμφάνισης της σελίδας.

Στην HTML, το πρόγραμμα Hello world, ένα συνηθισμένο πρόγραμμα υπολογιστή που χρησιμεύει για τη σύγκριση γλωσσών προγραμματισμού, γλωσσών σεναρίων και γλωσσών σήμανσης, φτιάχνεται με 9 γραμμές κώδικα, παρότι οι νέες γραμμές είναι προαιρετικές στην HTML:

- `<!DOCTYPE html>`
- `<html>`

- `<head>`
- `<title>Hello HTML</title>`
- `</head>`
- `<body>`
- `<p>Hello world</p>`
- `</body>`
- `</html>`

(Το κείμενο ανάμεσα στο `<html>` και το `</html>` περιγράφει την ιστοσελίδα, και το κείμενο μεταξύ του `<body>` και του `</body>` είναι το ορατό μέρος της. Το σημασμένο κείμενο '`<title>Hello HTML</title>`' καθορίζει τον τίτλο που θα εμφανίζεται στην μπάρα τίτλου του browser).

Στοιχεία

Το κύριο λήμα: Στοιχεία HTML

Τα έγγραφα HTML αποτελούνται από στοιχεία HTML τα οποία στην πιο γενική μορφή τους έχουν τρία συστατικά:

- ένα ζεύγος από ετικέτες (την «ετικέτα εκκίνησης» και την «ετικέτα τερματισμού»),
- μερικές ιδιότητες μέσα στην ετικέτα εκκίνησης και, τέλος,
- το κείμενο ή το γραφικό περιεχόμενο μεταξύ των ετικετών, το οποίο μπορεί να περιλαμβάνει και άλλα στοιχεία που παρεμβάλλονται μέσα .

Το στοιχείο HTML μπορεί να είναι οτιδήποτε ανάμεσα στις ετικέτες εκκίνησης και τερματισμού. Τέλος, κάθε ετικέτα περικλείεται σε σύμβολα «μεγαλύτερο από» και «μικρότερο από», δηλαδή `<` και `>`.

Μερικά στοιχεία HTML περιγράφονται ως άδεια στοιχεία, έχουν τη μορφή `<tag attribute1="value1" attribute2="value2" >`, και δεν έχουν καθόλου περιεχόμενο. Το όνομα κάθε στοιχείου HTML είναι το ίδιο όνομα, που χρησιμοποιείται στις

αντίστοιχες ετικέτες. Το όνομα της ετικέτας τερματισμού ξεκινά με μια κάθετο «/», η οποία παραλείπεται στα άδεια στοιχεία. Τέλος, αν δεν αναφέρονται ρητά οι ιδιότητες ενός στοιχείου, τότε χρησιμοποιούνται οι προεπιλογές σε κάθε περίπτωση.

Παραδείγματα στοιχείων (Επεξεργασία)

Κεφαλίδα του εγγράφου HTML: <head>...</head>. Συνήθως περιέχει τον τίτλο, π.χ:

```
<head>
```

```
<title>The title</title>
```

```
</head>
```

Επικεφαλίδες: οι επικεφαλίδες στην HTML ορίζονται με τις ετικέτες <h1> έως <h6>:

```
<h1>Επικεφαλίδα1</h1>
```

```
<h2>Επικεφαλίδα2</h2>
```

```
<h3>Επικεφαλίδα3</h3>
```

```
<h4>Επικεφαλίδα4</h4>
```

```
<h5>Επικεφαλίδα5</h5>
```

```
<h6>Επικεφαλίδα6</h6>
```

Παράγραφοι: <p>Paragraph 1</p> <p>Paragraph 2</p>

Αλλαγή γραμμής:
. Η διαφορά ανάμεσα στο
 και το <p> είναι ότι το «br» αλλάζει γραμμή, χωρίς να αλλάζει τη σημαντική δομή της σελίδας, ενώ το «p» τεμαχίζει τη σελίδα σε παραγράφους. Το «br» είναι ένα άδειο στοιχείο, δηλαδή δεν έχει περιεχόμενο, ούτε και χρειάζεται ετικέτα τερματισμού.

Σχόλια: Τα σχόλια μπορούν να βοηθήσουν στην κατανόηση της σήμανσης. Δεν εμφανίζονται στην ιστοσελίδα.

Υπάρχουν διάφοροι τύπου στοιχείων στην HTML.

Η δομική σήμανση, περιγράφει το σκοπό του κειμένου. Για παράδειγμα, το `<h2>Golf</h2>`, σημαίνει ότι η λέξη «Golf» θα είναι μια επικεφαλίδα δεύτερου επιπέδου. Η δομική σήμανση δεν εμπεριέχει συγκεκριμένο τρόπο εμφάνισης, αλλά οι περισσότεροι browser έχουν δικό τους προκαθορισμένο τρόπο για τη μορφοποίηση όλων των στοιχείων. Το περιεχόμενο μπορεί να μορφοποιηθεί οπτικά με τη χρήση των CSS. Η σήμανση οπτικής μορφοποίησης περιγράφει την εμφάνιση του κειμένου, άσχετα από το σκοπό του.

Για παράδειγμα, το `έντονο κείμενο` υποδεικνύει ότι οι συσκευές που θα εμφανίσουν το κείμενο θα το κάνουν έντονο, αλλά δεν λέει τι θα κάνουν με το κείμενο οι συσκευές που δεν μπορούν να το εμφανίσουν, όπως για παράδειγμα οι συσκευές φωνητικής ανάγνωσης σελίδων. Τόσο στην περίπτωση του `έντονο` όσο και του `<i>πλάγιο</i>`, υπάρχουν άλλα στοιχεία τα οποία μπορεί οπτικά να εμφανίζονται ίδια, αλλά είναι πιο σημαντικά, όπως τα `σημαντικό κείμενο` και `κείμενο με έμφαση` αντίστοιχα.

Είναι ευκολότερο να ερμηνεύσει ένας φωνητικός browser τα τελευταία δύο στοιχεία. Ωστόσο, δεν είναι ισότιμα με τα αντίστοιχα στοιχεία οπτικής μορφοποίησης. Για παράδειγμα, ένα πρόγραμμα φωνητικής ανάγνωσης της σελίδας δεν θα πρέπει να προφέρει με έμφαση τον τίτλο ενός βιβλίου, αλλά όταν το ίδιο περιεχόμενο εμφανίζεται σε μια οθόνη τότε ο τίτλος θα είναι με πλάγια γράμματα.

Τα περισσότερα στοιχεία οπτικής μορφοποίησης θεωρούνται ξεπερασμένα μετά την προδιαγραφή HTML 4.0 και έχουν αντικατασταθεί από τα CSS. Η σήμανση υπερκειμένου κάνει μερικά τμήματα ενός εγγράφου να συνδέονται με άλλα έγγραφα. Το στοιχείο `anchor` δημιουργεί έναν υπερσύνδεσμο στο έγγραφο, και η ιδιότητα `href` ορίζει τον στόχο του συνδέσμου.

Για παράδειγμα: η σήμανση HTML `Βικιπαίδεια` εμφανίζει το κείμενο «Βικιπαίδεια» ως υπερσύνδεσμο. Για την εμφάνιση μιας εικόνας ως συνδέσμου, μπορεί να εισαχθεί ένα στοιχείο «`img`» ως περιεχόμενο του στοιχείου «`a`». Όπως και το «`br`», το «`img`» είναι άδειο στοιχείο.

Ιδιότητες των στοιχείων

Οι περισσότερες ιδιότητες των στοιχείων είναι ζεύγη ονομάτων και τιμών, τα οποία διαχωρίζονται με ένα «=» και γράφονται μέσα στην ετικέτα εκκίνησης ενός στοιχείου, μετά το όνομα του στοιχείου. Η τιμή μπορεί να περικλείεται σε μονά ή διπλά εισαγωγικά, παρότι τιμές που αποτελούνται από συγκεκριμένους χαρακτήρες μπορούν να γράφονται χωρίς εισαγωγικά στην HTML, αλλά όχι στην XHTML.

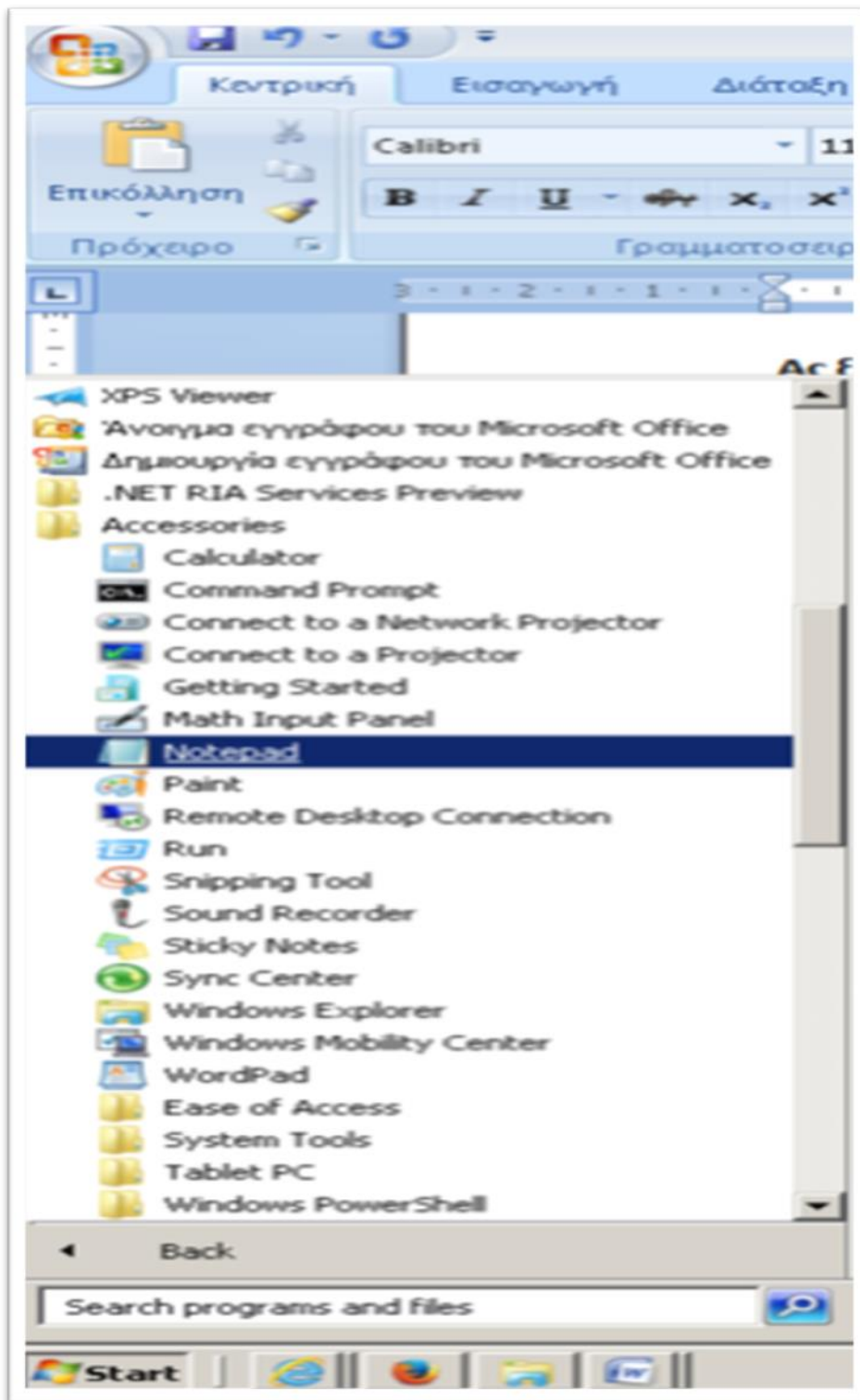
Το να μένουν οι τιμές των ιδιοτήτων χωρίς εισαγωγικά θεωρείται ανασφαλές. Εκτός από τις ιδιότητες που γράφονται ως ζεύγη ονομάτων και τιμών, υπάρχουν και μερικές, οι οποίες επηρεάζουν το στοιχείο απλά με την παρουσία τους μέσα στην ετικέτα εκκίνησης, όπως η ιδιότητα `ismap` του στοιχείου `img`.

Υπάρχουν και μερικές κοινές ιδιότητες, οι οποίες εμφανίζονται σε πολλά στοιχεία:

Η ιδιότητα `id`: παρέχει ένα αναγνωριστικό για ένα στοιχείο, το οποίο είναι μοναδικό σε ολόκληρο το έγγραφο. Χρησιμοποιείται για να ταυτοποιεί το στοιχείο, ώστε τα CSS να μπορούν να αλλάξουν τον τρόπο που αυτό εμφανίζεται, καθώς και τα σενάρια μπορούν να αλλάξουν, να μετακινήσουν ή να διαγράψουν τα περιεχόμενα ή την εμφάνισή του. Ακόμα, αν ένα `id` προστεθεί στο URL μιας σελίδας, παρέχει ένα μοναδικό αναγνωριστικό για ένα τμήμα της σελίδας.

Η ιδιότητα `class`: παρέχει τη δυνατότητα ταξινόμησης παρόμοιων αντικειμένων στην ίδια κλάση. Μπορεί να χρησιμοποιηθεί, για να αποδώσει κάποια σημασία στο στοιχείο ή για σκοπούς εμφάνισης. Για παράδειγμα, ένα έγγραφο HTML μπορεί να χρησιμοποιεί την επισήμανση `class="notation"` σε μερικά στοιχεία για να ξεχωρίσει από το υπόλοιπο κείμενο του εγγράφου.

Κατά την εμφάνιση του εγγράφου, αυτά τα στοιχεία μπορεί -για παράδειγμα- να εμφανίζονται όλα μαζί στο τέλος της σελίδας ως υποσημειώσεις, άσχετα με τη θέση που εμφανίζονται μέσα στον κώδικα. Επίσης, οι ιδιότητες `class` χρησιμοποιούνται σημασιολογικά στα `microformat`. Ένα στοιχείο μπορεί να έχει πολλαπλές κλάσεις.



Για παράδειγμα το `class="notation important"` βάζει το στοιχείο τόσο στην κλάση «notation» όσο και στην «important».

Η ιδιότητα **style**: εφαρμόζει στυλ εμφάνισης σε συγκεκριμένα στοιχεία. Θεωρείται καλύτερη τακτική να χρησιμοποιούνται οι ιδιότητες `id` ή `class`, ώστε να επιλέγεται το

στοιχείο μέσα σε ένα CSS, αλλά μερικές φορές μπορεί να είναι πιο απλό να ανατεθούν style κατευθείαν στο στοιχείο.

Η ιδιότητα title: προσθέτει μια εξήγηση στο στοιχείο στο οποίο εφαρμόζεται. Στους περισσότερους browser αυτή η ιδιότητα εμφανίζεται ως αναδυόμενο παράθυρο βοήθειας.

Η ιδιότητα lang: ταυτοποιεί τη φυσική γλώσσα των περιεχομένων του στοιχείου, η οποία μπορεί να είναι διαφορετική από το υπόλοιπο έγγραφο.

Το στοιχείο **«abbr»** μπορεί να χρησιμοποιηθεί για να επιδείξει μερικές από τις πιο πάνω ιδιότητες: `<abbr id="anId" class="jargon" style="color:purple;" title="Hypertext Markup`

`Language">HTML</abbr>`.

Αυτό το παράδειγμα εμφανίζεται ως HTML.

Στους περισσότερους browser, όταν σταθεί ο δείκτης του ποντικιού πάνω από αυτό το στοιχείο, τότε θα εμφανιστεί ένα αναδυόμενο παράθυρο με το περιεχόμενο **«Hypertext Markup Language»**.

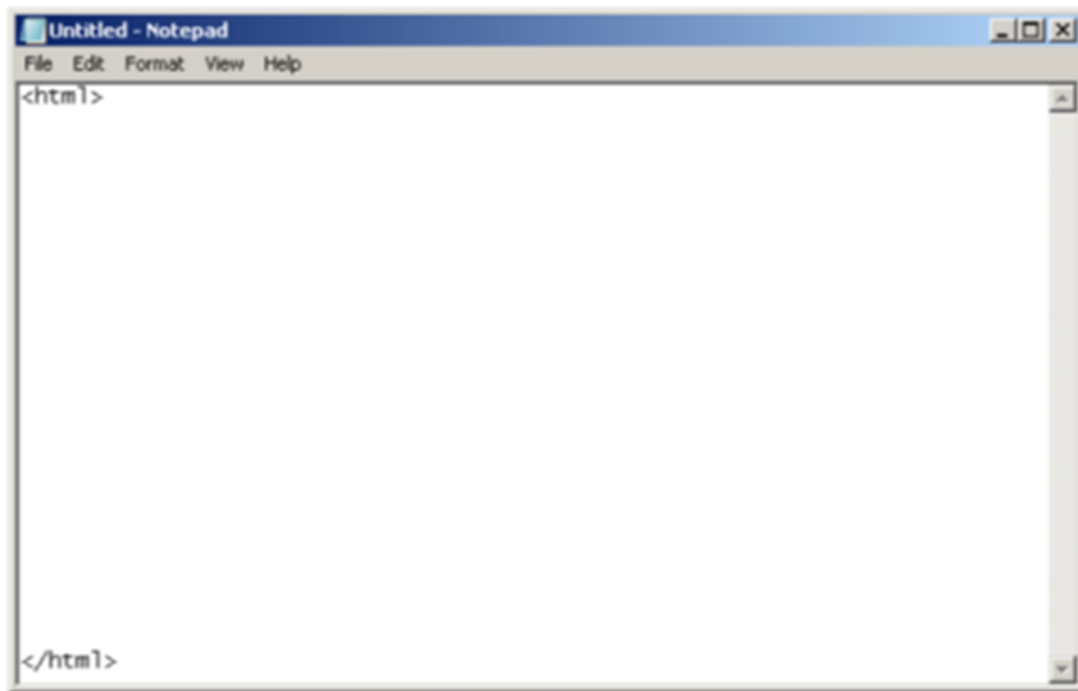
Πως γράφουμε έναν HTML κώδικα

Για να ξεκινήσουμε να γράφουμε HTML κώδικα ο πιο απλός τρόπος είναι να ανοίξουμε ένα Notepad.

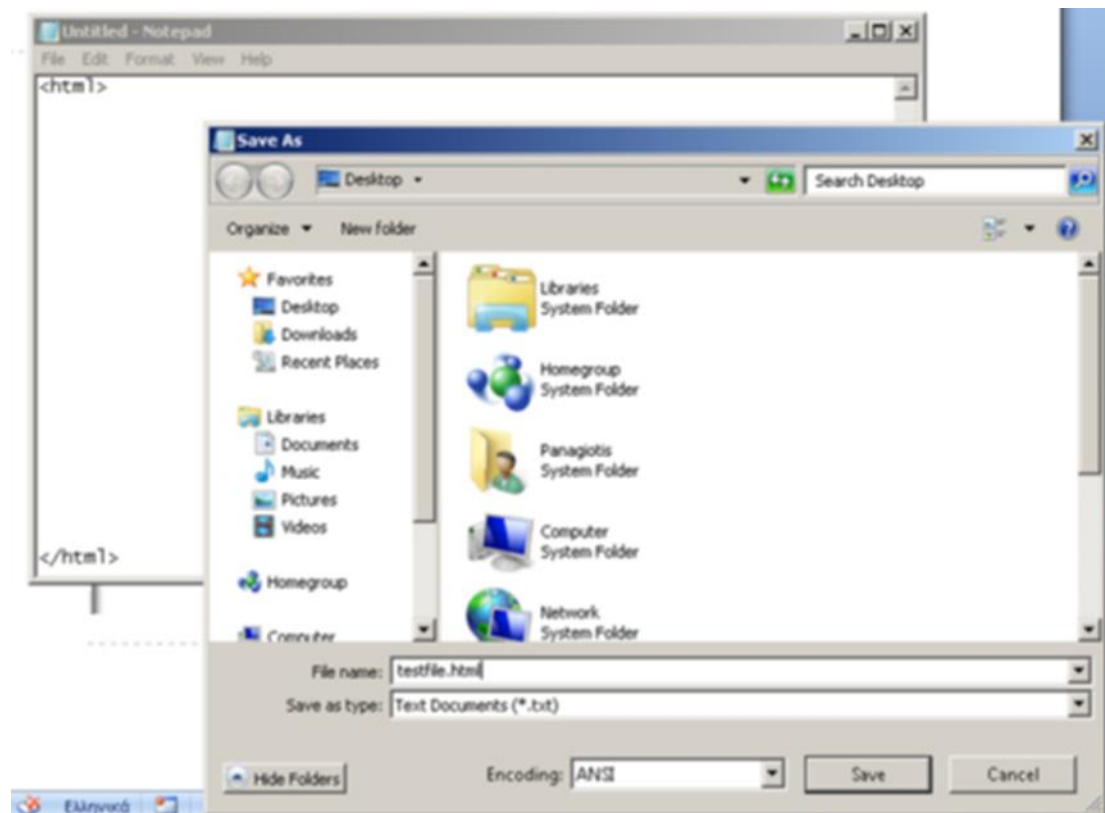
Για να το πετύχουμε αυτό πάμε στην Έναρξη (Start) μετά στα Προγράμματα (Programs) και στη συνέχεια στα Βοηθήματα (Accessories).

Τέλος, επιλέγουμε Σημειωματάριο (Notepad).

Για να καταλάβει ο φυλλομετρητής σας, ποια γλώσσα χρησιμοποιείτε γράψτε `<html>` στην αρχή και κλείστε γράφοντας `</html>`.



Για να σώσετε την ιστοσελίδα σας πηγαίνετε στο "save as", επιλέγετε μια ονομασία με κατάληξη .html (όπως "testfile.html") και επιλέγετε "all files" or "text".



Παράδειγμα

Έντονα, Πλάγια, Υπογραμμισμένα

Για να γράψουμε μια λέξη με έντονα γράματα (bold), αρκεί να τη βάλουμε μέσα σε tags της μορφής: `εδώ γράφουμε το κείμενο`.

Το αποτέλεσμα θα είναι: **εδώ γράφουμε το κείμενό μας.**

Παράδειγμα:

Καλώς ήρθατε στο site του `BookCrossing`! - Καλώς ήρθατε στο site του **BookCrossing!**

Για να γράψουμε μια λέξη με πλάγια γράματα (italic) θα χρησιμοποιήσουμε tags της μορφής: `<i>εδώ γράφουμε το κείμενο</i>`.

Το αποτέλεσμα θα είναι: *εδώ γράφουμε το κείμενό μας.*

Παράδειγμα:

Καλώς ήρθατε στο site του `<i>BookCrossing</i>`! - Καλώς ήρθατε στο site του *BookCrossing!*

Για να γράψουμε μια λέξη υπογραμμισμένη (underlined) θα χρησιμοποιήσουμε tags της μορφής: `<u>εδώ γράφουμε το κείμενο</u>`.

Το αποτέλεσμα θα είναι: εδώ γράφουμε το κείμενό μας.

Παράδειγμα:

Καλώς ήρθατε στο site του `<u>BookCrossing</u>`! - Καλώς ήρθατε στο site του BookCrossing!

Τέλος, μπορούμε να χρησιμοποιήσουμε ταυτόχρονα δύο ή και τα τρία tags:

`<u>εδώ γράφουμε το κείμενο</u>`

Το αποτέλεσμα θα είναι: **εδώ γράφουμε το κείμενό μας.**

Χρώμα γραμμάτων

Για να γράψουμε μια λέξη με κόκκινα γράματα αρκεί να τη βάλουμε μέσα σε tags της μορφής: `εδώ γράφουμε το κείμενο`.

Το αποτέλεσμα θα είναι: εδώ γράφουμε το κείμενο μας.

aqua (#00FFFF)	gray (#808080)	navy (#000080)	silver (#C0C0C0)
black (#000000)	green (#008000)	olive (#808000)	teal (#008080)
blue (#0000FF)	lime (#00FF00)	purple (#800080)	yellow (FFFF00)
fuchsia (FF00FF)	maroon (#800000)	red (FF0000)	white (FFFFFF)

Υπάρχει μια τεράστια ποικιλία χρωμάτων! Κάποια από τα βασικά χρώματα αρκεί να τα γραφτούν με το όνομά τους, για τα υπόλοιπα θα χρειαστεί ο κωδικός τους.

Χρώματα και κωδικούς θα βρείτε άπειρα στο internet... Ενδεικτικά: «html colors»

Παράδειγμα:

Καλώς ήρθατε στο site του `BookCrossing!` - Καλώς ήρθατε στο site του BookCrossing!

Φυσικά μπορείτε να συνδυάσετε περισσότερα από ένα tags:

Παράδειγμα:

Καλώς ήρθατε στο site του `BookCrossing!`

Καλώς ήρθατε στο site του BookCrossing!

Μέγεθος γραμμάτων

Το tag που θα χρησιμοποιήσουμε εδώ είναι και πάλι το font, αυτή τη φορά μαζί με το size.

Το default size είναι 3. Εμείς μπορούμε να το αλλάξουμε είτε δίνοντας τιμές από 1 έως 5 είτε αυξομειώνοντας το με την εντολή +1, +2, -1 κτλ...

Για να γράψουμε κάτι με γράματα νούμερο 5 αρκεί να το βάλουμε μέσα σε tags της μορφής: `εδώ γράφουμε το κείμενο`

Το αποτέλεσμα θα είναι: εδώ γράφουμε το κείμενό μας.

Παραδείγματα:

Καλώς ήρθατε στο site του BookCrossing!

Καλώς ήρθατε στο site του BookCrossing!

Καλώς ήρθατε στο site του BookCrossing!

Καλώς ήρθατε στο site του BookCrossing!

Και συνδυάζοντας το με κάποια από τα tags που έχουμε ήδη μάθει: Καλώς ήρθατε στο site του BookCrossing!

Καλώς ήρθατε στο site του BookCrossing!

Λίστες και πίνακες

Για να φτιάξουμε μια απλή λίστα με κουκίδες θα πρέπει όλη η λίστα να μπει μέσα σε tags της μορφής και κάθε στοιχείο της λίστας να μπει σε tags της μορφής . Έτσι έχουμε:

```
<ul>
<li>στοιχείο 1</li>
<li>στοιχείο 2</li>
<li>στοιχείο 3</li>
</ul>
```

Το αποτέλεσμα θα είναι:

- στοιχείο 1
- στοιχείο 2
- στοιχείο 3

Για να φτιάξουμε μια απλή λίστα με αρίθμηση θα πρέπει όλη η λίστα να μπει μέσα σε tags της μορφής και κάθε στοιχείο της λίστας να μπει σε tags της μορφής . Έτσι:

```
< ol>
```

```
<li>στοιχείο 1</li>
<li>στοιχείο 2</li>
<li>στοιχείο 3</li>
</ol>
```

Το αποτέλεσμα θα είναι:

1. στοιχείο 1
2. στοιχείο 2
3. στοιχείο 3

Για να δημιουργήσουμε ένα πίνακα χρειαζόμαστε τρία βασικά tags:

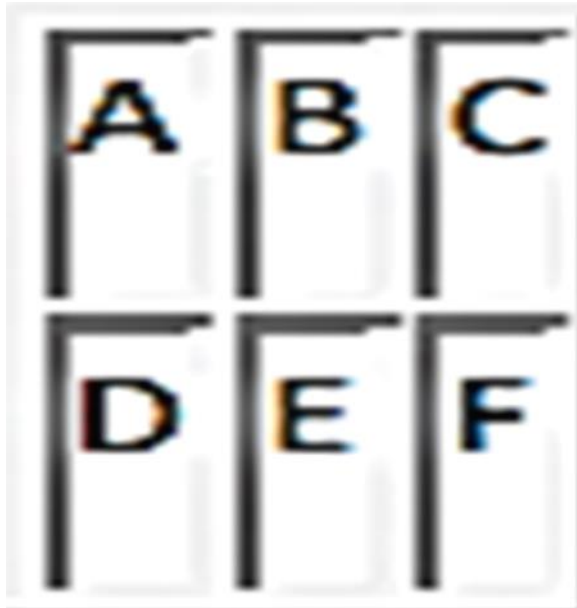
- Το tag που δημιουργεί τον πίνακα: <table> </table>
- Το tag που δημιουργεί τις γραμμές του πίνακα: <tr> </tr>
- Το tag που δημιουργεί τα κελία του πίνακα: <td> </td>

Ένας πίνακας λοιπόν 2x3 φτιάχνεται έτσι:

```
<table>
<tr>
<td> A </td><td> B </td><td> C </td>
</tr>
<tr>
<td> D </td><td> E </td><td> F </td>
</tr>
</table>
```

Το αποτέλεσμα θα είναι: A B C, D E F

Για να βάλουμε περίγραμμα στον πίνακά μας, αρκεί μέσα στο tag table να προσθέσουμε το: border="1": <table border="1">. Οπότε, θα έχουμε:



Για να το δηλώσουμε ότι το έγγραφο είναι γραμμένο στην Νέο-Ελληνική γλώσσα, πρέπει να προσθέσουμε στην περιοχή του <head> την εξής δήλωση <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-7">. Δηλαδή το παράδειγμά μας θα πρέπει να μετατραπεί στη παρακάτω μορφή:

```
<HTML>
```

```
<head>
```

```
<title>Το πρώτο μου HTML έγγραφο</title>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-7">
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</HTML>
```

Έχουμε 2 ειδών link που μπορούμε να χρησιμοποιήσουμε μέσα στα κείμενα του site.

Εσωτερικός σύνδεσμος (σύνδεσμος που οδηγεί σε άλλη σελίδα του site)

Για να προσθέσετε ένα link θα πρέπει πρώτα να επιλέξετε τη λέξη, τη φράση ή την εικόνα που θα θέλατε να δημιουργήσετε το link και πατάτε το εικονίδιο «Insert / Edit link».

Εξωτερικός Σύνδεσμος (σύνδεσμος που οδηγεί σε άλλο site)

Για να προσθέσετε ένα εξωτερικό link θα πρέπει πρώτα να επιλέξετε τη λέξη, τη φράση ή την εικόνα που θα θέλατε να δημιουργήσετε το link και πατάτε το εικονίδιο «Insert / Edit link».

Στο νέο παράθυρο που ανοίγει θα γράψετε το URL του site που θέλετε (π.χ. <http://www.clickmedia.gr>) στο πεδίο «URL».

Εισαγωγή υπερσυνδέσμων (links)

Οι υπερσύνδεσμοι υπάρχουν παντού , τους συναντάμε σε όλες τις σελίδες του διαδικτύου καθώς είναι ο τρόπος μετακίνησης και πλοήγησης μεταξύ των ιστοσελίδων. Ένας υπερσύνδεσμος μπορεί να είναι κείμενο, εικόνα, κουμπί κ.α.

Τα links κατασκευάζονται μέσω του tag <a> που χρησιμοποιεί το χρήσιμο attribute <href> όπως φαίνεται: link text.

Όπως παρατηρούμε και από την σύνταξη του tag <a> το σημαντικό είναι να δώσουμε τη σωστή διεύθυνση – url μέσω του attribute href, ώστε να είναι δυνατή η μετάβαση από την μια σελίδα στην άλλη ή από ένα σημείο της σελίδας σε ένα άλλο (bookmarks), δηλαδή δημιουργία σελιδοδεικτών.

Θα πρέπει να επισημάνουμε τη διαφορά μεταξύ απόλυτης διεύθυνσης και σχετικής.

Χρησιμοποιούμε **απόλυτη διεύθυνση**, όταν θέλουμε να μεταβούμε μέσω του υπερσυνδέσμου σε μια τελείως διαφορετική σελίδα σε κάποιον άλλο server. Στην περίπτωση αυτή, το απόλυτο url συντάσσεται ως εξής: «http://www.*.com/html/».

Διαφορετικά, αν θέλουμε να μετακινηθούμε σε τοπικό επίπεδο, δηλαδή σε μια διαφορετική σελίδα μέσα στον φάκελο, στον οποίο έχουμε αποθηκεύσει τα αρχεία Html και τα οποία έχουμε ανεβάσει σε κάποιον server, αρκεί να γράψουμε στην

διεύθυνση το όνομα του αρχείου html της σελίδας, στην οποία θέλουμε να μετακινηθούμε.

Μορφοποίηση υπερσυνδέσμων

Όταν ο δείκτης του ποντικιού περνάει πάνω από έναν υπερσύνδεσμο δύο πράγματα συμβαίνουν:

- Ο δείκτης μετατρέπεται σε «χεράκι» που σου δίνει την δυνατότητα να πατήσεις πάνω στο link
- και το ίδιο το link αλλάζει χρώμα ανάλογα με την κατάσταση στην οποία βρίσκεται.

Εξ' ορισμού οι υπερσύνδεσμοι στην περίπτωση που ο χρήστης δεν έχει κάνει κλικ πάνω τους, θα εμφανιστούν με υπογράμμιση και χρώμα γραμματοσειράς μπλε ενώ στην αντίθετη περίπτωση παραμένουν υπογεγραμμένοι, αλλά το χρώμα γραμματοσειράς μεταβάλλεται σε μωβ.

Επίσης υπάρχει και η περίπτωση του ενεργού υπερσυνδέσμου κατά την οποία το χρώμα γραμματοσειράς γίνεται κόκκινο.

```
<html> <head> <title> style tag </title>
<style>
a:link {
    color:#000000;
    background-color:transparent;
    text-decoration:none;
}
a:visited {
    color:#000000;
    background-color:transparent;
    text-decoration:none;
}
a:hover {
    color:#ff0000;
    background-color:transparent;
    text-decoration:underline;
}
a:active {
    color:#ff0000;
    background-color:transparent;
    text-decoration:underline;
}
</style>
</head>
<body>
<p>You can change the default colors of links</p>
<a href="html_images.asp">HTML Images</a>
</body> </html>
```

Παρόλα αυτά ένας

σχεδιαστής μπορεί να

αλλάξει τις προεπιλογές εμφάνισης των υπερσυνδέσμων χρησιμοποιώντας το tag style όπως φαίνεται παρακάτω:

Παρατηρούμε ότι το tag <style> γράφεται ανάμεσα στο tag <head> και όχι στο <body>, γιατί πολύ απλά είναι tag μορφοποίησης και όχι attribute που θα πρέπει να εισάγεται μέσα σε άλλα tags του <body>.

Επιπλέον, παρατηρούμε πώς συντάσσονται οι διαφορετικές περιπτώσεις του tag <a> των υπερσυνδέσμων. Για παράδειγμα, για να μορφοποιήσουμε τον υπερσύνδεσμο στην περίπτωση που κάνει κλικ ο χρήστης τότε θα γράψουμε:

```
a:visited { color:#000000; background-color:transparent; text-decoration:none; }
```

Άρα, το tag <a> μετατρέπεται στην ουσία σε attribute του tag <style> παίρνοντας τις παρακάτω τιμές:

- **visited**, είναι η περίπτωση που αλλάζει η μορφοποίηση γιατί ο χρήστης έχει επισκεφτεί προηγουμένως την σελίδα,
- **active**, όπου είναι η περίπτωση που έχει γίνει κλικ στο link από τον χρήστη και είναι ενεργός ο υπερσύνδεσμος,
- **link**, είναι η αρχική μορφοποίηση του υπερσυνδέσμου
- **hover** που είναι η περίπτωση, στην οποία περνά απλά ο δείκτης του ποντικιού πάνω από το link και η μορφοποίηση του υπερσυνδέσμου αλλάζει. Στις παραπάνω λοιπόν περιπτώσεις, όπου αλλάζει η κατάσταση στην οποία βρίσκεται ο υπερσύνδεσμος, μας δίνεται το δικαίωμα να αλλάξουμε και την εμφάνιση του link.

To attribute target

Το attribute target είναι αυτό που καθορίζει, πού θα ανοίξει για παράδειγμα η νέα σελίδα, στην οποία θα οδηγηθεί κάποιος αν πατήσει τον υπερσύνδεσμο, αν δηλαδή η νέα σελίδα θα ανοίξει σε νέα καρτέλα ή σε νέο παράθυρο του browser.

Οι τιμές που μπορεί να πάρει είναι:

- `_blank`: ανοίγει τον υπερσύνδεσμο σε νέα καρτέλα ή παράθυρο

- `_self`: ανοίγει τον υπερσύνδεσμο στην frame που ανήκει δηλαδή αν το link είναι γραμμένο μέσα σε ένα πλαίσιο τότε και η σελίδα θα ανοίξει μέσα στο ίδιο πλαίσιο (αυτό είναι προεπιλογή).
- `_parent`: ανοίγει τον υπερσύνδεσμο στο ανώτερο πλαίσιο της σελίδας. Αν υπάρχουν κρυμμένα πλαίσια το ένα μέσα στο άλλο, τότε η σελίδα θα ανοίξει στο ανώτερο πλαίσιο. Για τα πλαίσια όμως θα γίνει αναφορά αναλυτικότερα σε επόμενες παραγράφους.
- `_top`: ανοίγει τον υπερσύνδεσμο σε ολόκληρο το παράθυρο της σελίδας.
- `_frameName`: ανοίγει τον υπερσύνδεσμο στο πλαίσιο που έχει δημιουργηθεί και που καλείται με το όνομα που του έχει δοθεί

Ειδικοί χαρακτήρες

Υπάρχουν ειδικοί χαρακτήρες που δεν μπορούν να απεικονισθούν με απλές πληκτρολογήσεις (πατώντας ένα πλήκτρο ή ένα πλήκτρο και το Shift), αλλά μόνο με πιο περίπλοκους τρόπους (π.χ. πατώντας το Alt και έναν αριθμό).

Πρόκειται, για χαρακτήρες που δεν ανήκουν στο κλασικό 7-bit ASCII (character set) αλλά στο επεκταμένο 8-bit ASCII. Οι περισσότεροι browsers αναγνωρίζουν αυτούς τους ISO-Latin-1 (ISO-8859-1) χαρακτήρες, αλλά με ειδικό τρόπο. Για να αναγνωρίσουν τους χαρακτήρες αυτούς οι browsers όσων βλέπουν τις σελίδες, θα πρέπει να γραφτούν με ειδικό τρόπο (με έναν κωδικό που αρχίζει με το & και τελειώνει με το ;).

Στην κατηγορία των ειδικών χαρακτήρων συμπεριλαμβάνονται και οι χαρακτήρες άλλων γλωσσών. Αν στο PC γραφτεί μια HTML σελίδα στα Γερμανικά και τοποθετηθεί αυτή τη σελίδα στο Internet θα μπορούν να τη διαβάσουν σίγουρα μόνο όσοι έχουν εγκατεστημένο γερμανικό πληκτρολόγιο ή όσοι browsers υποστηρίζουν Unicode.

Αν όμως δεν είναι εγκατεστημένο γερμανικό πληκτρολόγιο, αλλά πρέπει απλώς να γραφτεί ένα όνομα ή λίγες λέξεις, τις οποίες θα διαβάσουν χρήστες χωρίς γερμανικό πληκτρολόγιο ή Unicode, θα πρέπει να χρησιμοποιηθεί το ISO-Latin

ισοδύναμο αυτών των χαρακτήρων. Έτσι, ο Δανός φιλόσοφος Søren Kierkegaard θα πρέπει να γραφτεί ως Søren Kierkegaard.

Πίνακες με όλους τους ειδικούς χαρακτήρες και τα ISO-Latin ισοδύναμά τους υπάρχουν σε διάφορα σημεία του Internet.

Χαρακτήρες με ειδικές λειτουργίες

Ας υποθέσουμε, πως θέλετε να βάλετε μέσα σε μια οδηγία (tag) τον χαρακτήρα < ή > θα δείτε πως θα υπάρξει πρόβλημα. Πχ. την εντολή 15 το Netscape 3 την αναγνωρίζει κανονικά, ενώ ο Internet Explorer 2 βλέπει μόνο το 15 και όποιο άλλο κείμενο υπάρχει πέρα από αυτό σαν bold (δηλαδή, διαβάζει το <7 σαν μια οδηγία που δεν καταλαβαίνει και την προσπερνά χωρίς να την εμφανίζει ή να την ερμηνεύει).

Για τον λόγο αυτό οι ειδικοί χαρακτήρες που περιέχονται στις οδηγίες (tags), όταν θέλουμε να τα δούμε μέσα σε μια σελίδα σαν απλοί χαρακτήρες, πρέπει να απεικονιστούν ως εξής:

Χαρακτήρας και Συμβολισμός :

< <

> >

& &

" "

Ορισμός γραμματοσειρών ...

Η καθορίζει ρυθμίσεις σχετικές με τις γραμματοσειρές. Τέτοιες είναι:

...

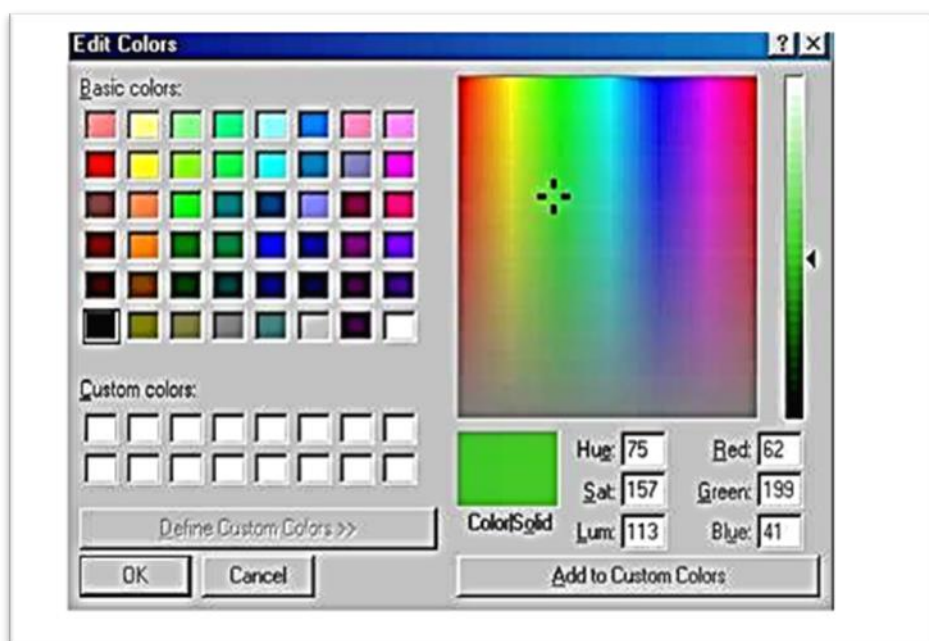
Καθορίζει το μέγεθος των γραμμάτων. Σε παλαιότερους browsers, το x μπορεί να πάρει τιμές από 1 (η μικρότερη) μέχρι 7 (η μεγαλύτερη). Default x=3. Η μπορεί να πάρει και σχετικές τιμές (από -3 έως +4), που καθορίζουν το μέγεθός της σε σχέση με την default γραμματοσειρά που έχει οριστεί στον browser. Σε

νεότερους browsers, το x μπορεί να πάρει και μεγαλύτερες τιμές:

Υπάρχει και η δυνατότητα να δοθούν ξεχωριστά χρώματα σε μεμονωμένες λέξεις, φράσεις ή και χαρακτήρες. Αυτό γίνεται με την οδηγία **FONT**. Π.χ. κείμενο σε κίτρινο χρώμα.

Δυστυχώς τα χρώματα που ορίζονται με το όνομα τους (π.χ. color=aqua για το ανοιχτό γαλάζιο) είναι σχετικά λίγα. Για να χρησιμοποιηθούν όλες τις δυνατές αποχρώσεις θα πρέπει να είναι γνωστό το RGB του χρώματος, που πρόκειται να χρησιμοποιηθεί (όλα τα χρώματα στον Η/Υ απεικονίζονται ως ένας συνδυασμός των Red Green Blue).

Για να γίνει αυτό, αρκεί να διαθέτουμε ένα πρόγραμμα γραφικών. Για παράδειγμα, ανοίγουμε το Paint των Windows 98 και επιλέγουμε Colors -Edit Colors - Define Custom Colors. Στο δεξιό μέρος του παραθύρου που βρίσκεται μπροστά μας, υπάρχει μια μπάρα και αριστερά της ένα μεγάλο τετράγωνο με διάφορα χρώματα.



Κάντε κλικ μέσα στο μεγάλο τετράγωνο με τα πολλά χρώματα στο σημείο που υπάρχει χρώμα της αρεσκείας σας. Το χρώμα εμφανίζεται κάτω δεξιά από το μεγάλο τετράγωνο. Μετακινώντας το δείκτη της μπάρας, που βρίσκεται δεξιά από

το μεγάλο τετράγωνο σε κάποιο άλλο σημείο εκτός από το τέρμα κάτω (το default) ορίζουμε τη φωτεινότητα αυτού του χρώματος.

Στο κάτω δεξιά μέρος του παραθύρου, αναγράφονται οι τιμές R (Red) G (Green) και B (Blue) του χρώματος που διαλέξατε. Τώρα που τις έχετε, το μόνο που σας μένει είναι να βρείτε τον δεκαεξαδικό κωδικό του χρώματος, γιατί αυτός πρέπει να οριστεί μέσα στον HTML κώδικα.

Για παράδειγμα: Το RGB 62,199,41, του παραπάνω παραδείγματος είναι το 3ec729 σε δεκαεξαδικό κωδικό και το χρώμα που μας δίνει είναι το: ανοικτό πράσινο

Καθορίζει την γραμματοσειρά. Καλό είναι να χρησιμοποιείται με φειδώ, γιατί αν η γραμματοσειρά δεν υπάρχει στον Η/Υ του αναγνώστη των σελίδων, μπορεί να υπάρξουν προβλήματα (π.χ. να μην φαίνονται τα κείμενα με Ελληνικούς χαρακτήρες).

ΠΡΟΣΟΧΗ: Καλύτερα να αποφεύγεται η υπερβολική χρήση της (ορισμός πολλών γραμματοσειρών ή/και πολλών χρωμάτων γραμμάτων στο ίδιο κείμενο), γιατί δίνει πολύ άσχημη εικόνα.

Εισαγωγή Εικόνων και Γραφικών

Είδη αρχείων

Υπάρχουν πολλοί τρόποι, με τους οποίους μπορεί να αποθηκευτεί μια εικόνα ή ένα γραφικό στον Η/Υ. Κάθε ένας από αυτούς, βασίζεται σε ένα ξεχωριστό πρότυπο και αναγνωρίζεται από την επέκταση του ονόματος του αρχείου που περιέχει την εικόνα.

Οι Web browsers αναγνωρίζουν μόνο δύο τέτοια πρότυπα. Το «GIF» και το .JPG (δηλαδή το jpeg).

Αν έχετε μια εικόνα που θέλετε να συμπεριλάβετε σε μια σελίδα και δεν είναι αποθηκευμένη σε μια από τις δύο αυτές μορφές, τότε πρέπει να την μετατρέψετε σε gif ή jpg χρησιμοποιώντας ένα από τα ειδικά προγράμματα μετατροπής, που παρέχουν αυτή την δυνατότητα(π.χ. PAINTSHOP PRO).

Προσθέτοντας μια εικόνα σε μια web σελίδα

Αν θέλουμε να προσθέσουμε μία εικόνα στην ιστοσελίδα μας πρέπει να χρησιμοποιήσουμε το tag . Σημειώνεται πως το tag αυτό δεν κλείνει, δηλαδή δεν υπάρχει το .

Το tag , παίρνει τις εξής παραμέτρους:

src="images/square.jpg" : src η παράμετρος αυτή παίρνει ως τιμή την απόλυτη διαδρομή και το όνομα της εικόνας που θα εμφανιστεί. Σημειώνεται πως οι εικόνες που υποστηρίζει ένας web browser είναι της μορφής jpg, gif και png.

width="400" height="267": width η παράμετρος αυτή παίρνει ως τιμή το πλάτος της εικόνας σε pixels, ενώ η παράμετρος height το ύψος της εικόνας σε pixels.

Δεν είναι απαραίτητο αλλά είναι καλό να χρησιμοποιούνται , επίσης πως αν αυξηθούν ή μειωθούν σε σχέση με το πραγματικό μέγεθος της εικόνας τότε η εικόνα θα μεγαλώσει ή μικρύνει αντίστοιχα. Υπάρχουν περιπτώσεις όπου για σχεδιαστικούς λόγους το πλάτος και ύψος μιας εικόνας πρέπει να δηλωθεί ως ποσοστό π.χ. 100%.

Αν θέλαμε να εισάγουμε μία εικόνα που ονομάζεται square.jpg στην ιστοσελίδα μας ο κώδικας θα ήταν. με δεδομένο ότι η εικόνα βρίσκεται σε ένα φάκελο images και το αρχείο html βρίσκεται εκτός του φακέλου.

*Αν τώρα θέλουμε να εισάγουμε μια εικόνα σε ένα σημείο, αυτό μπορεί να γίνει με την οδηγία *

Με την οδηγία αυτή η εικόνα θα μπει στην σελίδα στο σημείο, όπου υπάρχει η οδηγία, αλλά στην αριστερή πλευρά της οθόνης.

Ο δεξιά της χώρος θα παραμείνει κενός. Το κείμενο που υπάρχει πριν από αυτήν θα βρίσκεται από πάνω της και το κείμενο που υπάρχει μετά από αυτήν θα βρίσκεται από κάτω της.

Συμβουλή: Προσπαθήστε να ανακυκλώνετε τις εικόνες σας (π.χ. να χρησιμοποιείτε την ίδια εικόνα σε διαφορετικές περιπτώσεις αν αυτό είναι εφικτό). Με τον τρόπο αυτό δίνετε μεγαλύτερη ομοιομορφία στις σελίδες σας και διευκολύνετε την κίνηση του επισκέπτη (είναι πιο εύκολο να βρει το δρόμο του κινούμενος μμέσα σε σελίδες που έχουν την ίδια βασική σχεδίαση). Επίσης το εικονίδιο αυτό χρειάζεται να έρθει στον browser μόνο μια φορά (την πρώτη). Τις επόμενες φορτώνεται από την cache και έτσι ο επισκέπτης, κερδίζει χρόνο.

Δημιουργία links σε μία εικόνα

Αν θέλουμε σε μία εικόνα να δημιουργήσουμε κάποιο Link, ώστε όταν ένας χρήστης πατάει πάνω της να ανοίγει μία ιστοσελίδα ή να στέλνεται ένα e-mail, τότε χρησιμοποιούμε το tag <a> όπως περιγράψαμε παραπάνω για το κείμενο.

Η μόνη διαφορά είναι ότι περιεχόμενο του tag <a> δεν θα είναι κείμενο αλλά ο κώδικας που εμφανίζει μία εικόνα.

Εικόνες και γραμμή κειμένου

Μια εικόνα μαζί με μια σειρά κείμενο μπορεί να συμπεριληφθεί σε μια επικεφαλίδα με το ακόλουθο τρόπο: <H6>ANY MODEM</H6>.

Το ίδιο θα συμβεί, αν η εικόνα τοποθετηθεί μέσα σε μια παράγραφο κειμένου <P> (αν η οδηγία IMG της εικόνας δεν βρίσκεται μέσα σε παράγραφο ή heading τότε θα λάβει θέση μόνη της μέσα στη σελίδα χωρίς τίποτε άλλο δίπλα της).

Μέσα στις επικεφαλίδες και μέσα στις παραγράφους, μπορεί να υπάρχει κείμενο τόσο δεξιά όσο και αριστερά της εικόνας (αν φυσικά χωράει) και οι αποστάσεις μεταξύ των γραμμών θα επηρεαστούν από το μέγεθος της εικόνας. Πάντα όμως δεξιά και αριστερά της εικόνας θα υπάρχει ΜΟΝΟ ΜΙΑ γραμμή κειμένου.

Φυσικά, μπορεί να τύχει η εικόνα να βρεθεί στην αρχή ή στο τέλος της γραμμής. Στην περίπτωση αυτή δεν θα υπάρχει κείμενο αριστερά ή δεξιά της αντίστοιχα.

Η εικόνα μπορεί να τοποθετηθεί διάφορες θέσεις σχετικά με το επίπεδο της γραμμής κειμένου, που υπάρχει δεξιά και αριστερά της. Αυτό θα γίνει με την οδηγία align.

Οι δυνατότητες που υπάρχουν είναι: . Θα βάλει το πάνω μέρος της εικόνας στο ίδιο επίπεδο με την γραμμή του κειμένου.

Align=Bottom

Θα βάλει το κάτω μέρος της εικόνας στο ίδιο επίπεδο με την γραμμή του κειμένου (η default επιλογή).

Align=Middle

Θα βάλει το μεσαίο επίπεδο της εικόνας στο ίδιο επίπεδο με την γραμμή του κειμένου.

Εισαγωγή μιας εικόνας σαν φόντο της σελίδας

Αν θέλουμε το φόντο της σελίδας να μην είναι κάποιο χρώμα αλλά ένα γραφικό, χρησιμοποιούμε την παράμετρο BACKGROUND. Η σύνταξή της είναι: <BODY BACKGROUND="όνομα αρχείου">...</BODY> "όνομα αρχείου" είναι φυσικά το όνομα (ή/και το path) του αρχείου εικόνας που θέλουμε να βάλουμε σαν φόντο στην σελίδα.

Τα αρχείο εικόνας που θα χρησιμοποιήσουμε σαν φόντο, περιλαμβάνουν συνήθως κάποιο γραφικό ή γραφικά που (αν η εικόνα έχει μέγεθος μικρότερο από αυτό της σελίδας) επαναλαμβάνονται (tiled) καθ' ύψος και πλάτος μέχρι να καλύψουν ολόκληρο το φόντο.

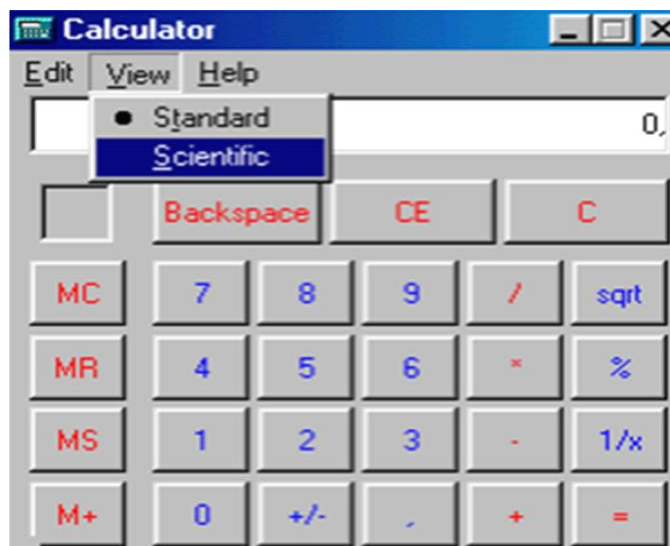
Σε ορισμένες περιπτώσεις το αρχείο του φόντου μπορεί να αποτελείται και από ένα μόνο χρώμα (αν θέλουμε μονόχρωμο φόντο). Στην περίπτωση αυτή, το αρχείο



είναι σκόπιμο να έχει όσο το δυνατόν μικρότερο μέγεθος (κατά προτίμηση ένα μόνο pixel). Το pixel αυτό θα εκταθεί για να καταλάβει ολόκληρο το φόντο.

Πώς βρίσκουμε τον 16εξαδικό κωδικό ενός χρώματος

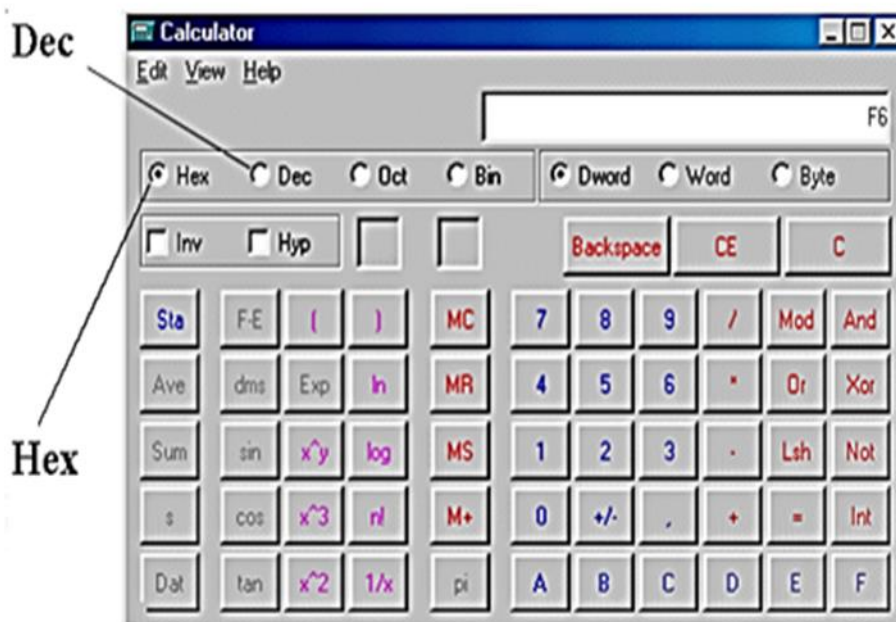
Κάθε χρώμα, το οποίο θα θελήσουμε να χρησιμοποιήσουμε, προέρχεται συνήθως από 2 πηγές: Είτε το έχουμε δημιουργήσει εμείς (με κάποιο πρόγραμμα γραφικών), είτε το έχουμε δει κάπου αλλού (π.χ. σε ένα άλλο site). Και στις 2 περιπτώσεις, αποθηκεύουμε το γραφικό που μας ενδιαφέρει, το ανοίγουμε με ένα σχεδιαστικό πρόγραμμα και βλέπουμε ποιο είναι το RGB του (η δυνατότητα αυτή υπάρχει σε όλα



σχεδόν τα προγράμματα γραφικών). Το RGB αποτελείται από έναν αριθμό για το κόκκινο, έναν για το πράσινο και έναν για το μπλε χρώμα (ο συνδυασμός αυτών των τριών χρωμάτων σε διαφορετικές αναλογίες μας δίνει όλα τα χρώματα που εμφανίζονται στην οθόνη του Η/Υ μας). Οι τιμές των αριθμών αυτών εκτείνονται από το 0 έως το 255. Π.χ. το μαύρο είναι 0,0,0 και το λευκό είναι 255,255,255 Αφού μάθουμε το RGB, μετατρέπουμε την τιμή του σε δεκαεξαδικό κωδικό χρησιμοποιώντας ένα από τα διάφορα προγράμματα που υπάρχουν για το σκοπό αυτό (Agile ColorWeb , Crayon κ.α.) ή τον calculator των windows.

Για να βρούμε τον δεκαεξάρικό κωδικό με τον calculator των windows ανοίγουμε το πρόγραμμα (αν δεν υπάρχει στα μενού ψάχνουμε για το αρχείο calc.exe και το εκτελούμε) και επιλέγουμε View – Scientific.

Αν υποθέσουμε πως θέλουμε μια απόχρωση του μουσταρδί με Red=246 Green=192 Blue=128 τότε γράφουμε 246 και αλλάζουμε την προβολή από δεκαδική - Dec - σε δεκαεξαδική - Hex - (κάνουμε κλικ στην επιλογή Hex).



Παράδειγμα HTML LAYOUT

```
<!DOCTYPE html>
```

```
<html>
```

```

<head>

<title>HTML Layout using Tables</title>

</head>

<body>

<table width = "100%" border = "0">

<tr>

<td colspan = "2" bgcolor = "#b5dcb3">

<h1>This is Web Page Main title</h1>

</td>

</tr>

<tr valign = "top">

<td bgcolor = "#aaa" width = "50">

<b>Main Menu</b><br />

HTML<br />

PHP<br />

PERL...

</td>

<td bgcolor = "#eee" width = "100" height = "200">

Technical and Managerial Tutorials

</td>

</tr>

<tr>

<td colspan = "2" bgcolor = "#b5dcb3">

```

```
<center>
```

```
Copyright © 2007 Tutorialspoint.com
```

```
</center>
```

```
</td>
```

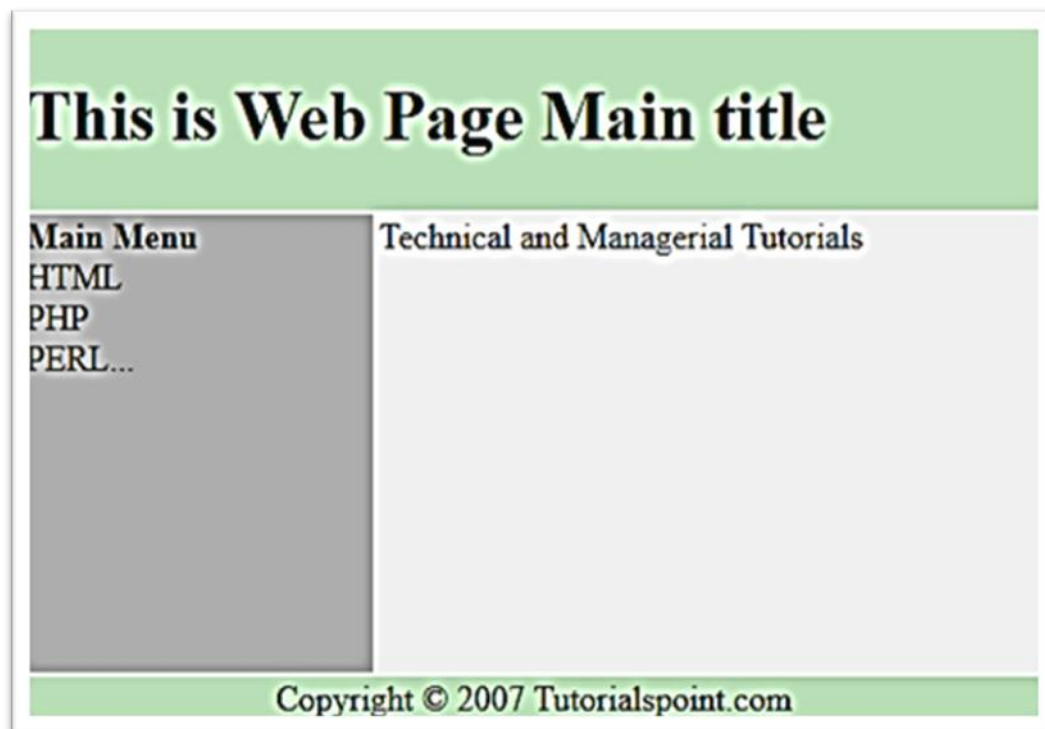
```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

Το αποτέλεσμα θα είναι



HTML FONTS

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Setting Font Size</title>

</head>

<body>

  <font size = "1">Font size = "1"</font><br />

  <font size = "2">Font size = "2"</font><br />

  <font size = "3">Font size = "3"</font><br />

  <font size = "4">Font size = "4"</font><br />

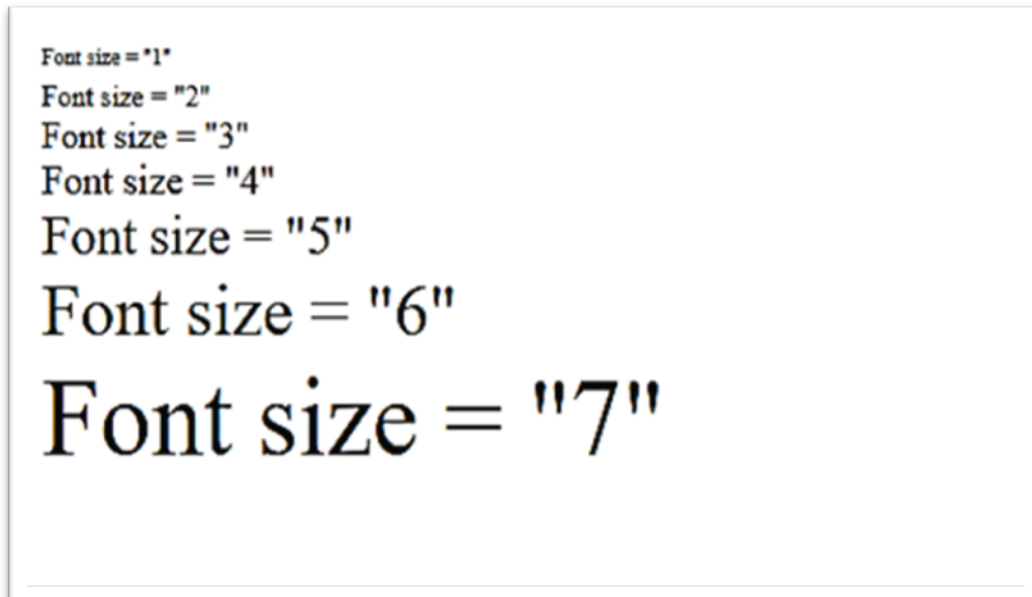
  <font size = "5">Font size = "5"</font><br />

  <font size = "6">Font size = "6"</font><br />

  <font size = "7">Font size = "7"</font>

</body>

</html>
```



HTML FORMS

```
<!DOCTYPE html>

<html>

<head>

<title>Text Input Control</title>

</head>

<body>

<form >

First name: <input type = "text" name = "first_name" />

<br>

Last name: <input type = "text" name = "last_name" />

</form>

</body>

</html>
```

A screenshot of a web form rendered from the HTML code above. The form is contained within a rectangular border. It features two text input fields. The first field is preceded by the label "First name:" and the second by "Last name:". Both labels are in a blue, serif font. The input fields are simple rectangular boxes with thin borders.

Τι είναι το CSS (Cascading Style Sheets)

Cascading Style Sheets – CSS μας βοηθά σαν γλώσσα να ορισθεί με σαφήνεια και ιδιαίτερη ευελιξία ο τρόπος, με τον οποίο θα εμφανίζονται τα διάφορα στοιχεία στην ιστοσελίδα μας. Είναι δηλαδή ένα σύνολο από ορίσματα, διαφορετικά από αυτά στην HTML, τα οποία αναπτύχθηκαν με στόχο την καλύτερη διαχείριση της εμφάνισης των ιστοσελίδων.

Τα πλεονεκτήματα της χρήση CSS έναντι της μορφοποίησης μέσω HTML attributes

1) Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασική HTML.

2) Ευκολότερη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός ολόκληρου site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στο site.

3) Μικρότερο μέγεθος αρχείου, δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.

4) Καλύτερο SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο περιεχόμενο σκέτο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexing).

5) Γρηγορότερες σελίδες. Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS, ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του site μας.

Βασικοί κανόνες σύνταξης

Σχόλια

Αν και η ανάγκη για σχόλια στο CSS δεν είναι τόσο επιτακτική όσο σε γλώσσες προγραμματισμού, μπορούμε να εισάγουμε σχόλια μέσα σε `/*...*/`

Παράδειγμα: `/* Αυτό είναι ένα σχόλιο */`

CSS rules

Ένας «κανόνας» CSS αποτελείται από 2 μέρη: Τον επιλογέα (CSS selector), ο οποίος αφορά το TI θα μορφοποιηθεί και τις ιδιότητες (CSS properties), οι οποίες αφορούν το ΠΩΣ αυτό θα μορφοποιηθεί. Η σύνταξη είναι η εξής:

{

ιδιότητα1: τιμή1;

ιδιότητα2: τιμή2;

ιδιότητα3: τιμή3;

...

}

Απλοί επιλογείς CSS (CSS selectors)

Επιλέγουμε ποια στοιχεία θα μορφοποιήσουμε μέσω των επιλογέων CSS (CSS selectors), οι οποίοι ουσιαστικά αποτελούν κάποιους «κανόνες», ώστε να καταλάβει ο browser, πού θέλουμε να εφαρμοστούν τα όσα γράφουμε κάθε φορά. Υπάρχουν αρκετά περίπλοκοι CSS selectors στο specification του CSS, οι οποίοι προσφέρουν τεράστια ευελιξία, ωστόσο εδώ θα εξετάσουμε τους πιο βασικούς, και για λόγους απλούστευσης, και επειδή αρκετοί από τους πιο περίπλοκους δεν υποστηρίζονται από όλους τους browsers.

Όταν ο επιλογέας είναι ένας χαρακτήρας αστερίσκου (*), τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε στοιχείο της σελίδας μας. Όπως είναι κατανοητό, συνήθως δεν είναι και πολύ χρήσιμος επιλογέας από μόνος του, και χρησιμοποιείται κυρίως σε συνδυασμό με άλλους.

Στοιχείο

Όταν ο επιλογέας αποτελείται απλά από το όνομα ενός html tag, τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε τέτοιο στοιχείο html. Για παράδειγμα, ο επιλογέας p θα εφαρμοστεί σε οτιδήποτε στη σελίδα μας περιλαμβάνεται εντός των tags <p>...</p>, ο επιλογέας table θα εφαρμοστεί σε όλους τους πίνακες στη σελίδα μας, ο επιλογέας img, θα αφορά όλες τις εικόνες στη σελίδα κοκ. Προφανώς όταν θέλουμε να εφαρμόσουμε κάποιες ιδιότητες CSS σε ολόκληρη τη σελίδα, χρησιμοποιούμε ως επιλογέα body μιας και όλο το ορατό τμήμα της σελίδας περιέχεται εντός των tags <body>...</body>.

Όνομα κλάσης

Όταν ο επιλογέας περιλαμβάνει μια τελεία (.) στην αρχή του, τότε ο browser ψάχνει όσα στοιχεία στη σελίδα μας περιλαμβάνουν την ιδιότητα class και εφαρμόζει τις ιδιότητες που θα γράψουμε στον κανόνα CSS αυτό σε οποιοδήποτε στοιχείο περιλαμβάνει την κλάση «όνομα_κλάσης» στην ιδιότητα class του. Φυσικά ως όνομα_κλάσης μπορούμε να γράψουμε οτιδήποτε αποτελείται από γράματα, αριθμούς, παύλες και χαρακτήρες underscore και να ξεκινάει με γράμα. Αξίζει να σημειωθεί ότι μπορεί το ίδιο στοιχείο να ανήκει σε περισσότερες από μια κλάσεις, διαχωρισμένες με κενά μέσα στην class html attribute του. Πχ `<p class="emphasis bodytext">...</p>`.

Για παράδειγμα ο παρακάτω κανόνας CSS: `.emphasis { color: red; }`. Θα κάνει κόκκινα τα γράματα και στο στοιχείο `<p class="emphasis">blah blah</p>`, και στο στοιχείο `<div class="emphasis otherclass">blah blah</div>` αλλά όχι στο στοιχείο `<h1 class="otherclass">blah blah</h1>`.

Οι κλάσεις γενικά χρησιμοποιούνται όταν θέλουμε να ομαδοποιήσουμε κάποια στοιχεία html, για τα οποία δεν μπορούμε να βρούμε κάποιον άλλο επιλογέα που να αφορά όλα αυτά και μόνον αυτά. Οπότε τους προσδίδουμε μια συγκεκριμένη κλάση, ώστε να μπορούμε στο CSS μας να αναφερθούμε μόνο σε αυτά και να τα μορφοποιήσουμε.

Στοιχείο- όνομα κλάσης

Αποτελεί ουσιαστικά συνδυασμό των δύο παραπάνω επιλογέων. Εφαρμόζεται σε όσα στοιχεία αποτελούνται από το html tag `<στοιχείο>` και ανήκουν στην κλάση `όνομα_κλάσης`. Πχ ο επιλογέας `p.emphasis` εφαρμόζεται σε ό,τι περιέχεται σε tags της μορφής `<p class="emphasis">...</p>`. Ο επιλογέας αυτός είναι χρήσιμος όταν υπάρχουν πολλά διαφορετικού τύπου στοιχεία με την ίδια κλάση και είναι επιθυμητό να εφαρμοστεί διαφορετική μορφοποίηση ανάλογα με τον τύπο του στοιχείου.

#όνομα_id

Όταν ο επιλογέας μας περιλαμβάνει ένα χαρακτήρα δέησης (#) στην αρχή του, τότε ο browser εφαρμόζει τις ιδιότητες που θα γράψουμε στο στοιχείο, το οποίο

περιλαμβάνει την ιδιότητα `id="όνομα_id"`. Δεν πρέπει να υπάρχουν δύο (ή περισσότερα) στοιχεία στη σελίδα με το ίδιο `id`. Τα `ids` διέπονται από τους ίδιους κανόνες ονοματολογίας με τις κλάσεις. Ουσιαστικά, ο,τι μπορεί να γίνει με τα `ids` μπορεί να γίνει και με τη χρήση κλάσεων, αλλά όταν το στοιχείο που προκειται να μορφοποιηθεί είναι μοναδικό, είναι γενικά καλύτερο να γίνει χρήση `ids`.

Στοιχείο[attribute="value"]

Αποτελεί ουσιαστικά μια «επέκταση» του επιλογέα, στοιχείο που αναλύθηκε πρώτος. Ο εν λόγω επιλογέας, κάνει τον browser να εφαρμόζει τις ιδιότητες που θα γραφτούν σε αυτόν σε κάθε στοιχείο με tag <στοιχείο>, το οποίο επιπροσθέτως έχει την τιμή `value` στην html ιδιότητα `attribute`. Π.χ , ο επιλογέας `input[type="submit"]`, αφορά όλα τα κουμπιά υποβολής φόρμας που υπάρχουν στη σελίδα μας, χωρίς ωστόσο να εφαρμόζεται σε άλλα στοιχεία φορμών, όπως τα πεδία κειμένου (στα οποία η ιδιότητα `type` είναι `text`).

Άλλο ένα παράδειγμα: Έστω ότι είναι να μορφοποιηθούν μόνο όσοι πίνακες στη σελίδα είναι κεντραρισμένοι. Αν γινόταν χρήση ως επιλογέα `table`, τότε οι ιδιότητες που θα γραφόταν σε αυτόν τον επιλογέα θα εφαρμόζονταν σε όλους τους πίνακες ανεξαιρέτως. Ενώ αν γινόταν χρήση του επιλογέα `table[align="center"]` τότε ο κανόνας CSS που θα γραφτεί θα εφαρμοστεί μόνο σε όσους πίνακες έχουν την ιδιότητα `align="center"`.

Ψευδό-κλάσεις και ψευδό-στοιχεία

Ορισμένες φορές χρησιμοποιούνται οι λεγόμενες ψευδό-κλάσεις (`pseudo-classes`) ή ψευδό-στοιχεία (`pseudo-elements`), τα οποία επιτρέπουν να επιλεχθούν στοιχεία τα οποία δεν αποτελούν `html elements`, αλλά κομμάτια τους ή συγκεκριμένες καταστάσεις τους. Ουσιαστικά αποτελούν κάποιες λέξεις-κλειδιά που γράφονται μετά από έναν επιλογέα του τύπου στοιχείο και ξεκινούν με . Οι πιο συνηθισμένες ψευδο-κλάσεις/στοιχεία που χρησιμοποιούνται είναι:

a:link: Χρησιμοποιείται αποκλειστικά για στοιχεία `a` (δηλαδή ως εξής: `a:link`) και αφορά τους συνδέσμους που ο χρήστης δεν έχει ακόμη επισκεφθεί.

a:visited: Χρησιμοποιείται αποκλειστικά για στοιχεία a (δηλαδή ως εξής: a:visited) και αφορά τους συνδέσμους που ο χρήστης έχει επισκεφθεί.

στοιχείο:active: Αφορά τα στοιχεία τύπου <στοιχείο> τη στιγμή που ο χρήστης έχει πατημένο το ποντίκι πάνω σε αυτά. Πχ ο επιλογέας a:active εφαρμόζεται σε συνδέσμους την ώρα που ο χρήστης έχει πατημένο το ποντίκι πάνω τους.

στοιχείο:hover: Από τις πιο συχνά χρησιμοποιούμενες ψευδό - κλάσεις. Αφορά τα στοιχεία τύπου <στοιχείο> τη στιγμή που ο χρήστης έχει το δείκτη του ποντικιού πάνω σε κάποιο από αυτά (χωρίς να πατάει κάποιο πλήκτρο). Μπορεί να βοηθήσει να δημιουργηθούν διάφορα όμορφα εφέ, τα οποία παλιότερα ήταν εφικτά μόνο με JavaScript.

στοιχείο:focus: Χρησιμοποιείται κυρίως για στοιχεία φορμών και εφαρμόζεται στα στοιχεία τύπου <στοιχείο> που εκείνη τη στιγμή έχουν «focus», παραδείγματος χάριν, ένα πεδίο κειμένου, στο οποίο ο χρήστης έκανε κλικ για να εισάγει κείμενο.

στοιχείο:first - letter: Αφορά το πρώτο γράμα του κειμένου εντός κάποιου στοιχείου τύπου <στοιχείο>. Το ψευδό - στοιχείο αυτό μπορεί να μας βοηθήσει να δημιουργήσουμε αρχιγράμματα. Πχ ο επιλογέας p:first - letter αφορά το πρώτο γράμα κάθε παραγράφου.

στοιχείο:first - line: Ψευδό - στοιχείο παρόμοιο με το παραπάνω, μόνο που αντί να αφορά μόνο το πρώτο γράμα του κειμένου μέσα στο στοιχείο τύπου <στοιχείο>, αφορά ολόκληρη την πρώτη γραμμή.

Αν και δεν έχει ιδιαίτερη πρακτική χρησιμότητα, αξίζει να σημειωθεί ότι τα 5 πρώτα θεωρούνται ψευδό - κλάσεις (pseudo - classes) ενώ τα 2 τελευταία ψευδό - στοιχεία (pseudo - elements).

Σύνθετοι επιλογείς CSS (CSS selectors)

Πολλές φορές γίνεται να συνδυαστεί σε έναν επιλογέα περισσότεροι από έναν υπό - επιλογείς, βάσει συγκεκριμένων κανόνων σύνταξης, κάτι που προσφέρει μεγαλύτερη ευελιξία και εξοικονόμηση χρόνου. Οι πιο βασικοί και συχνά χρησιμοποιούμενοι τρόποι συνδυασμού επιλογέων παρουσιάζονται παρακάτω:

A. επιλογέας1, επιλογέας2, επιλογέας3, ...

Οι ιδιότητες που θα γραφτούν σε αυτόν τον κανόνα CSS, θα εφαρμοστούν σε κάθε στοιχείο που πληροί τις προϋποθέσεις επιλογής είτε για τον επιλογέα1, είτε για τον επιλογέα2, είτε για τον επιλογέα3 κ.ο.κ. Για παράδειγμα, ο επιλογέας `input[type="text"], textarea` θα εφαρμοστεί σε κάθε πεδίο κειμένου που υπάρχει στη σελίδα μας, είτε είναι για εισαγωγή κειμένου μιας γραμμής είτε είναι πολλών γραμμών

B. επιλογέας1 επιλογέας2 επιλογέας3 ... επιλογέας n

Οι ιδιότητες που θα γραφτούν σε αυτόν τον κανόνα CSS θα εφαρμοστούν σε στοιχεία που πληρούν τις προϋποθέσεις του επιλογέα n και επιπροσθέτως περιέχονται μέσα σε κάποιο στοιχείο που πληροί τις προϋποθέσεις του επιλογέα n - 1 το οποίο βρίσκεται μέσα σε κάποιο στοιχείο που πληροί τις προϋποθέσεις του επιλογέα n - 2 κ.ο.κ. Για παράδειγμα, ο επιλογέας `p img` θα εφαρμοστεί σε όσες εικόνες περιέχονται μέσα σε tags `<p>...</p>`.

Πως εισάγουμε CSS στη σελίδα μας;

Για να εισαχθεί CSS στη σελίδα μας με τρεις διαφορετικούς τρόπους, αναλόγως την περίπτωση. Οι τρόποι αυτοί, κατά σειρά φθίνουσα προτεραιότητας, είναι:

A. Inline CSS: Αν είναι να μορφοποιηθεί ένα στοιχείο μόνο, και δεν πρόκειται να χρειαστεί αυτό το είδος μορφοποίησης για τίποτε άλλο στο site, γίνεται να γραφτεί η λέξη «χύμα» (δηλ. χωρίς το κομάτι του CSS selector και χωρίς αγκύλες) CSS properties μέσα στο attribute style, το οποίο το διαθέτει σχεδόν κάθε στοιχείο html. Για παράδειγμα, για να γίνει μια συγκεκριμένη παράγραφο κόκκινη, θα προστεθεί `style="color:red;"` στο `<p>` tag. Αξίζει να σημειωθεί, ότι αν κάποιες από τις ιδιότητες που θα γράφτουν εντός της attribute style ενός στοιχείου «συγκρούονται» με

κανόνες CSS που έχουν οριστεί για αυτό αλλού και το αφορούν, τότε ο browser θα επιλέξει να εφαρμόσει αυτά που γράφτηκαν εντός του attribute style, ως πιο συγκεκριμένα για το στοιχείο αυτό.

B. CSS για μια συγκεκριμένη σελίδα: Πολλές φορές, μπορεί να χρειαστεί να εφαρμοσθεί κάποιους κανόνες CSS μόνο για τη συγκεκριμένη σελίδα και όχι για όλο το site (ή να διαφοροποιηθούν κάποιες ιδιότητες τους υπάρχοντες). Ένας τρόπος να γίνει αυτό, για να μην δημιουργηθεί ένα ξεχωριστό αρχείο CSS είναι να εισαχθεί εντός των tags <head>...</head> της σελίδας τους κανόνες CSS μας μέσα σε <style> tags (με την html attribute type τους σε text/css).

Αξίζει να σημειωθεί, ότι συνήθως όταν ξεκινάει η ανάπτυξη ενός site, είναι πιο βολικό να χρησιμοποιηθεί αυτός ο τρόπος αρχικά, μιας και πρέπει να επεξεργαστεί μόνο ένα αρχείο αντί για δύο. Όταν τελειώσει η πρώτη σελίδα, συνήθως μεταφέρεται το CSS που έχει γραφτεί σε εξωτερικό αρχείο, ώστε να είναι εύκολο να χρησιμοποιηθούν οι ίδιοι κανόνες CSS και στις άλλες σελίδες του site, χωρίς να πρέπει φυσικά να χρειαστεί copy - paste σε κάθε σελίδα.

Γ. Εξωτερικό αρχείο CSS: Η πιο «σωστή» χρήση του CSS και αυτή που θα έπρεπε να είναι η πρώτη λύση, είναι η χρήση εξωτερικού αρχείου CSS. Για να τη χρησιμοποιήσουμε, γράφουμε τους κανόνες CSS μας σε ένα αρχείο με επέκταση css (πχ main.css) και στο head της κάθε σελίδας του site μας

Αξίζει να σημειωθεί ότι μπορούμε να έχουμε πολλά αρχεία css στην ίδια σελίδα, και μεγαλύτερη προτεραιότητα έχει πάντα αυτό που έχει γραφτεί τελευταίο.

Βασικές ιδιότητες CSS

Ιδιότητες μορφοποίησης κειμένου

Color

Αφορά το χρώμα του κειμένου, αλλά αν δεν οριστεί χρώμα περιγράματος (μέσω της ιδιότητας `border - color`), ο browser χρησιμοποιεί αυτό που ορίστηκε στην ιδιότητα `color`. Τα χρώματα μπορούν να εισαχθούν είτε σε μορφή RGB (πχ `color: rgb(255,128,30);`), είτε σε μορφή hex (πχ `color: #ff801e;`) είτε με τη μορφή κάποιου keyword (πχ `color:orange;`).

Font - size

Αφορά το μέγεθος της γραματοσειράς. Οι τιμές που δέχεται μπορούν να είναι εκφρασμένες σε ένα μεγάλο πλήθος μονάδων μεγέθους, από τις οποίες οι πιο ευρέως διαδεδομένες είναι τα pixels (πχ `font-size: 12px;`) και οι στιγμές (πχ `font-size:10pt`).

Font family

Η ιδιότητα αυτή επιτρέπει να οριστεί ένα πλήθος γραματοσειρών που θα χρησιμοποιηθούν για το κείμενο, κατά σειρά προτίμησης. Ουσιαστικά μέσω αυτής ορίζεται η γραματοσειρά του κειμένου, απλά επιτρέπεται να ορίζονται και εναλλακτικές επιλογές, ώστε αν η γραματοσειρά που ορίστηκε δεν υπάρχει στον υπολογιστή του χρήστη, να μην καταστρέψει ο browser την εμφάνιση της σελίδας επιλέγοντας όποια αυτός νομίζει, απλά την επόμενη επιλογή .

Αν καμία από όσες γραματοσειρές ορίστηκαν δεν υπάρχει στον υπολογιστή του χρήστη, τότε ο browser επιλέγει κάποια που να ανήκει στη γενική οικογένεια γραματοσειρών (`generic font - family`) που θα οριστούν στο τέλος με το κατάλληλο keyword. Τέτοια keywords είναι:

serif: Γραματοσειρές με «πατούρες» όπως η Georgia

sans - serif: Γραματοσειρές χωρίς «πατούρες» όπως η Trebuchet MS.

monospace: Γραματοσειρές όπου το κάθε γράμμα καταλαμβάνει ίσο πλάτος με τα υπόλοιπα, όπως η Courier New.

cursive: Καλλιγραφικές γραματοσειρές όπως η Monotype Corsiva.

fantasy: «Διακοσμητικές» γραματοσειρές.

Font - style

Σε αντίθεση με αυτό που θα περίμενε κανείς από μια ιδιότητα με αυτό το όνομα, δηλαδή την εφαρμογή πολλών και διαφόρων «εφέ» στο κείμενο, στην πραγματικότητα αφορά μόνο την περίπτωση όπου το κείμενο θα είναι πλάγιο. Για άλλα εφέ, χρησιμοποιούνται άλλες ιδιότητες, οι οποίες θα εξεταστούν παρακάτω. Οι πιθανές τιμές της είναι normal, italic και oblique. Οι δύο τελευταίες συνήθως κάνουν το ίδιο πράγμα (και είναι προτιμότερη η χρήση της italic μιας και υποστηρίζεται και από Internet Explorer 3, ενώ η υποστήριξη για την oblique ξεκινά από Internet Explorer 4), ενώ είναι προφανής η λειτουργία της τιμής normal.

Font-weight

Αφορά το «βάρος» της γραμματοσειράς και στην πράξη χρησιμοποιείται για να ορίσει αν το κείμενο μας θα είναι έντονο ή όχι, μιας και οι περισσότερες γραμματοσειρές που χρησιμοποιούνται στο web διατίθενται μόνο σε δύο βάρη: Κανονικό και έντονο, σε αντίθεση με πιο εξειδικευμένες γραμματοσειρές που χρησιμοποιούνται από γραφίστες, οι οποίες πολλές φορές διατίθενται σε διάφορα βάρη. Οπότε οι τιμές που συνήθως χρησιμοποιούνται σε αυτή την ιδιότητα είναι οι normal και bold, η λειτουργία των οποίων είναι προφανής.

Text - decoration

Επιτρέπει να εφαρμοστεί στο κείμενο διάφορα εφέ, συμπεριλαμβανόμενης και της υπογράμμισης. Οι τιμές που δέχεται είναι οι εξής:

none: Καμία διακόσμηση

underline: Υπογράμμιση

overline: Γραμμή πάνω από το κείμενο (ουσιαστικά το αντίθετο της υπογράμμισης)

line - through: Διαγράμμιση

blink: Κάνει το κείμενο να αναβοσβήνει. Δεν πρέπει να χρησιμοποιείται, παρά μόνο σε εξαιρετικά σπάνιες περιπτώσεις, μιας και η εμπειρία έχει δείξει ότι είναι εξαιρετικά κουραστικό για τον αναγνώστη και αφαιρεί επαγγελματικότητα από το design της ιστοσελίδας.

Text - align

Επιτρέπει να καθοριστεί η στοίχιση του κειμένου. Όπως θα περίμενε κανείς, οι πιθανές τιμές είναι left, center, right και justify.

Font

Επιτρέπει να καθοριστούν αρκετές ιδιότητες κειμένου με μια μόνο ιδιότητα (συγκεκριμένα τις font - size, font - family, font - weight και font - style από όσες αναφέρθηκαν παραπάνω, καθώς και κάποιες άλλες που δεν αναφέρθηκαν). Αν κάποια ιδιότητα δεν συμπεριληφθεί, ο browser θα «υποθέσει» την προεπιλεγμένη τιμή της.

Προσοχή: Αν δεν οριστεί κάποια επιμέρους ιδιότητα όταν χρησιμοποιείται η συγκεντρωτική ιδιότητα font, ο browser «υποθέτει» την προεπιλεγμένη τιμή για αυτή, όχι ότι δεν ορίστηκε. Αυτό σημαίνει ότι αν γραφτεί για παράδειγμα: font-weight:bold; font: 12px Calibri, Trebuchet MS, Verdana, sans-serif; τότε το κείμενο δεν θα είναι έντονο, μιας και αφού δεν ορίστηκε στη font και την font - weight. Τότε ο browser «υποθέτει» την προεπιλεγμένη τιμή για αυτή την ιδιότητα, η οποία είναι η normal, και μιας και η font βρίσκεται κάτω από την font - weight, έχει μεγαλύτερη προτεραιότητα, οπότε επικρατεί για τη font - weight η τιμή normal.

Ιδιότητες που αφορούν το φόντο

Background - color

Χρησιμοποιείται για να ορίσει χρώμα φόντου στα στοιχεία που αφορά ο επιλογέας. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιεγράφηκαν για την ιδιότητα color.

Background - image

Χρησιμοποιείται για να ορίσει μια εικόνα φόντου. Η τιμή που δέχεται είναι της μορφής url (διεύθυνση εικόνας), για παράδειγμα background-image:url (http://www.e-steki.gr/logo.gif);. Γενικά όταν εισάγεται μια διεύθυνση URL ως τιμή (ή τμήμα τιμής) κάποιας ιδιότητας CSS, πάντα γράφεται εντός των παρενθέσεων του url(). Φυσικά είναι δυνατόν να εισαχθούν και σχετικά URLs, πχ background-image:url(logo.gif).

Background - repeat

Αφορά τον τρόπο που θα επαναλαμβάνεται η εικόνα που ορίστηκε στην παραπάνω ιδιότητα. Οι τιμές που δέχεται είναι:

no - repeat: Καμία επανάληψη

repeat - x: Η εικόνα επαναλαμβάνεται στον οριζόντιο άξονα

repeat - y: Η εικόνα επαναλαμβάνεται στον κάθετο άξονα

repeat: Η εικόνα επαναλαμβάνεται και οριζόντια και κάθετα (η τιμή repeat είναι και η προεπιλεγμένη τιμή της ιδιότητας background - repeat).

Background - position

Η ιδιότητα αυτή επιτρέπει να οριστεί η θέση που θα τοποθετηθεί η εικόνα φόντου που έχει οριστεί με την ιδιότητα background - image τόσο κάθετα, όσο και οριζόντια. Συνήθως ως τιμές της χρησιμοποιούνται κάποια keywords, τα οποία είναι τα εξής:

- left, center, right όσον αφορά την οριζόντια θέση της εικόνας
- top, center, bottom όσον αφορά την κάθετη θέση της εικόνας

Τα παραπάνω μπορούν να χρησιμοποιηθούν και συνδυαστικά (και συνήθως αυτό γίνεται), όπως στα παρακάτω παραδείγματα:

`background-position: left bottom;`

`background-position: top right;`

`background-position: center top;`

ή μπορεί να οριστεί η τοποθέτηση μόνο κατά τον ένα άξονα (για τον άλλο υποτίθεται η προεπιλεγμένη τιμή, η οποία είναι συνήθως top για τον κάθετο άξονα και left για τον οριζόντιο), όπως στα παρακάτω παραδείγματα:

`background-position: left;`

`background-position: bottom;`

Αντί για τα παραπάνω keywords, γίνεται να οριστεί η θέση της εικόνας φόντου και αριθμητικά, είτε με ποσοστό ή με κάποια μονάδα μεγέθους, όπως στα παρακάτω παραδείγματα:

`background-position: 10% 25%;`

`background-position: 60px 100px;`

Αυτές οι τιμές χρησιμοποιούνται πολύ πιο σπάνια. Η ιδιότητα αυτή χρησιμοποιείται συνήθως όταν έχει οριστεί να μην επαναλαμβάνεται η εικόνα φόντου (μέσω της ιδιότητας `background - repeat`). Κάποιοι browsers υποστηρίζουν τη χωριστή δήλωση θέσης της εικόνας στο v κάθετο και τον οριζόντιο άξονα, μέσω δύο ιδιοτήτων με όνομα `background - position - x` και `background - position - y`, αλλά καλύτερα να αποφεύγεται η χρήση αυτών των ιδιοτήτων, μιας και δεν υποστηρίζονται από όλους τους browsers.

Background

Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να οριστούν με μία μόνο ιδιότητα όλες οι παραπάνω (όπως είναι η ιδιότητα `font` για τα χαρακτηριστικά κειμένου).

Ιδιότητες που αφορούν το περίγραμμα

Border - color

Ρυθμίζει το χρώμα περιγράματος. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιγράφηκαν για την ιδιότητα color. Αν δεν οριστεί αυτή η ιδιότητα, χρησιμοποιείται το χρώμα που ορίστηκε στην ιδιότητα color.

Border - width

Ρυθμίζει το πάχος του περιγράματος σε κάποια από τις μονάδες μέτρησης που μπορούν να χρησιμοποιηθούν στο CSS, συνηθέστερα σε pixels. Παράδειγμα:

```
border - width: 10px;
```

Border - style

Ορίζει το στυλ του περιγράματος. Οι τιμές που χρησιμοποιούνται συνήθως για την ιδιότητα αυτή είναι οι εξής:

solid: «Συμπαγές» περίγραμμα, δηλαδή χωρίς κάποια διακόσμηση, μια ενιαία γραμμή.

dashed: Περίγραμμα που αποτελείται από παύλες.

dotted: Περίγραμμα που αποτελείται από τελείες. Αξίζει να σημειωθεί ότι τα αποτελέσματα της χρήσης της τιμής αυτής διαφέρουν ελαφρώς ανά τους browsers. Για παράδειγμα ο Internet Explorer εμφανίζει τα dotted περιγράματα ως αποτελούμενα από μικρούς κύκλους, ενώ ο Mozilla Firefox τα εμφανίζει ως αποτελούμενα από μικρά τετράγωνα.

double: Σαν την τιμή solid, μόνο που δημιουργεί δύο περιγράματα, η απόσταση και το πάχος των οποίων δεν μπορούν να ρυθμιστούν χωριστά αλλά ρυθμίζονται αυτόματα ώστε το πάχος τους να είναι συνολικά όσο η τιμή που ορίσαμε στην ιδιότητα border - width. Συνήθως, χρησιμοποιείται σε συνδυασμό με πάχος περιγράματος 3px, για το οποίο δίνει ένα αρκετά καλαίσθητο αποτέλεσμα.

Υπάρχουν και οι τιμές groove, ridge, inset, outset οι οποίες δημιουργούν διαφόρων ειδών «ψευδό - 3D» περιγράματα, οι οποίες καλύτερα να αποφεύγονται για λόγους αισθητικής.

Border

Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να οριστεί με μία μόνο ιδιότητα όλες τις παραπάνω (όπως είναι η ιδιότητα font για τα χαρακτηριστικά κειμένου). Παράδειγμα: border: 10px solid #999999;

Κάθε μια από τις ιδιότητες που αναφέρθηκαν παραπάνω, αποτελείται από τέσσερις επιμέρους ιδιότητες, οι οποίες επιτρέπουν να οριστούν διαφορετικές τιμές για την ιδιότητα αυτή όσον αφορά το πάνω/δεξί/κάτω/αριστερό περίγραμμα. Κάθε ιδιότητα της μορφής border - κ ά τ ι αντιστοιχεί σε τέσσερις ιδιότητες της μορφής border - top - κ ά τ ι , border - right - κ ά τ ι , border - bottom - κ ά τ ι , border - left - κ ά τ ι .

Επίσης, κάθε ιδιότητα από αυτές που αναφέρθηκαν παραπάνω δέχεται και τιμές της μορφής [top] [right] [bottom] [left] (ένας εύκολος μνημονικός κανόνας για τη σειρά των τιμών είναι ότι η σειρά αυτή είναι σύμφωνη με την πορεία των δεικτών του ρολογιού) ή της μορφής [top,bottom] [right,left] αντί για μόνο μια τιμή που θα εφαρμοστεί και για τις τέσσερις πλευρές του περιγράμματος. Για παράδειγμα, για να οριστεί ένα γκρι συμπαγές περίγραμμά με πάχος αριστερά 1px, δεξιά 2px, πάνω 3px και κάτω 4px γίνεται να γραφτεί είτε:

```
border-left-width: 1px;
```

```
border-right-width: 2px;
```

```
border-top-width: 3px;
```

```
border-bottom-width: 4px;
```

```
border-style: gray;
```

```
border-style: solid; είτε
```

```
border-width: 3px 2px 4px 1px;
```

```
border-style: gray;
```

```
border-style: solid;
```

Άλλες χρήσιμες ιδιότητες

Padding

Ορίζει το κενό που θα υπάρχει μεταξύ των ορίων ενός στοιχείου και των περιεχομένων του. Είναι πολύ σημαντικό να ορίζουμε padding σε στοιχεία στα οποία έχουμε ορίσει κάποιο περίγραμμα, ώστε να μην «κολλάνε» τα περιεχόμενα τους με το περίγραμμα, κάτι που φαίνεται ιδιαίτερα άσχημο και ερασιτεχνικό.

Περισσότερο padding δίνει πιο επαγγελματική εμφάνιση (μέχρι κάποιων ορίων φυσικά). Για να οριστεί διαφορετικό padding ανά πλευρά, γίνεται ομοίως με το border, είτε να χρησιμοποιηθούν οι ιδιότητες padding - top, padding - right, padding - bottom, padding - left, είτε να εισαχθούν και οι τέσσερις τιμές στην ιδιότητα padding με τη σειρά [top] [right] [bottom] [left], είτε [top,bottom] [right,left]. Π.χ

```
padding: 8px;
```

```
padding: 2px 6px;
```

```
padding: 0px 6px 6px 6px;
```

Margin

Το αντίθετο ουσιαστικά του padding. Ορίζει τον χώρο μεταξύ των ορίων ενός στοιχείου και όσων το περιβάλλουν. Είναι ιδιαίτερα σημαντικό να οριστεί margin σε εικόνες, ώστε να έχουν απόσταση από τα περιεχόμενα τους, μιας και είναι ιδιαίτερα αντιαισθητικό να «κολλάνε» με το κείμενο. Φυσικά και εδώ γίνεται να χρησιμοποιηθούν είτε οι ιδιότητες margin - top, margin - right, margin - bottom, margin - left για να οριστούν διαφορετικές τιμές margin ανά πλευρά, είτε να εισαχθούν οι τέσσερις διαφορετικές τιμές margin με τη σειρά [top] [right] [bottom] [left], ή [top,bottom] [right,left]. Παραδείγματα:

```
margin: 4px;
```

```
margin: 2px 4px;
```

```
margin: 0px 8px 2px 2px;
```

```
width, height
```

Float

Η ιδιότητα αυτή είναι ανεκτίμητης χρησιμότητας, όταν χρησιμοποιείται CSS και για το layout της σελίδας, αν και είναι η δυσκολότερη στην κατανόηση από όσες αναφέρθηκαν. Δέχεται τις τιμές right, left και none και επιτρέπει τα στοιχεία που περιβάλλουν το στοιχείο εκείνο στο οποίο εφαρμόζεται η ιδιότητα αυτή να «ρέουν» τριγύρω του. Μια συνηθισμένη χρήση της ιδιότητας είναι στις εικόνες που συνοδεύουν ένα άρθρο. Πολλές φορές χρησιμοποιείται μετά από floated στοιχεία κάποιο στοιχείο (συνήθως ένα άδειο div) με την ιδιότητα clear: both; για να «καθαρίσει» τα floats για το μετέπειτα περιεχόμενο.

Προτεραιότητα

Πολλές φορές για το ίδιο στοιχείο μπορεί να ταιριάζουν αρκετοί κανόνες CSS, ορισμένοι εκ των οποίων να περιέχουν ιδιότητες που «συγκρούονται». Για παράδειγμα, έστω ότι έχουμε στο CSS τους παρακάτω κανόνες:

```
.testclass { color: red; }
```

```
div { color: gray; }
```

και στην html μας υπάρχει κάποιο στοιχείο `<div class="testclass">test text</div>`. Το κείμενο του θα είναι κόκκινο ή γκρι;

Απάντηση: Το κείμενο θα είναι κόκκινο, διότι ο επιλογέας .testclass είναι πιο «συγκεκριμένος» απ'ότι ο επιλογέας div. Ωστόσο, αν στο CSS υπήρχαν οι κανόνες:

```
.testclass { color:red; }
```

```
.testclass2 { color: gray; }
```

και στην html μας είχαμε το στοιχείο `<div class="testclass testclass2">test text</div>`, τότε το κείμενο θα ήταν γκρι, γιατί η δήλωση της κλάσης testclass2 διαβάστηκε τελευταία από τον browser, οπότε έχει μεγαλύτερη προτεραιότητα.

Οι βασικότεροι κανόνες στο CSS είναι οι εξής:

- Οι ιδιότητες CSS που βρίσκονται μέσα στην ιδιότητα style ενός στοιχείου έχουν πάντα μεγαλύτερη προτεραιότητα από οποιεσδήποτε άλλες.
- Οι κανόνες CSS που βρίσκονται μέσα στα tags <style>...</style> έχουν μεγαλύτερη προτεραιότητα από αυτούς που βρίσκονται σε ένα εξωτερικό αρχείο CSS.
- Όταν δύο ή περισσότερα αρχεία CSS συμπεριλαμβάνονται στην ίδια ιστοσελίδα, οι κανόνες αυτού που γράφτηκε τελευταίο στη σελίδα έχουν μεγαλύτερη προτεραιότητα από αυτούς του πρώτου.
- Ένας κανόνας CSS έχει μεγαλύτερη προτεραιότητα από όσους κανόνες CSS βρίσκονται γραμμένοι πιο πάνω από αυτόν αν δεν συντρέχει άλλος λόγος για να μετρήσει παραπάνω κάποιος από αυτούς.
- Οι κανόνες με επιλογή τύπου .όνομα κλάσης έχουν μεγαλύτερη προτεραιότητα από κανόνες με επιλογή τύπου στοιχείο, μιας και είναι πιο συγκεκριμένοι.
- Οι επιλογές του τύπου #όνομα_id έχουν μεγαλύτερη προτεραιότητα από επιλογές του τύπου ,όνομα κλάσης, μιας και είναι πιο συγκεκριμένοι.

Σε κάθε περίπτωση, αν είναι επιθυμητό μια ιδιότητα ενός κανόνα να έχει μεγαλύτερη προτεραιότητα από αυτή που της αντιστοιχεί, γίνεται πριν το ; να γραφτεί !important.

Κληρονομικότητα

Όταν εφαρμόζεται μια ιδιότητα CSS σε ένα στοιχείο, μπορεί να κληρονομηθεί και στα στοιχεία - απογόνους του ή μπορεί και να μην κληρονομηθεί, αναλόγως την ιδιότητα. Όσον αφορά τις ιδιότητες που αναφέρθηκαν παραπάνω, αυτές που κληρονομούνται και στα στοιχεία απογόνους είναι οι: color, font - size, font - family ,font - style, font - weight, text - align, text - decoration.

Παραδείγματα

Σελίδα με διαμορφωμένη εικόνα

Για να δημιουργηθεί το αποτέλεσμα της επόμενης εικόνας, χρησιμοποιείται συνδυασμός κώδικα HTML και CSS, όπως παρακάτω.

Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet, consectetur...



Εικόνα 12.3 Παράδειγμα συνδυασμού CSS και HTML για μορφοποίηση σελίδας.

```
CSS
.floatright
{
float: right;
margin: 0 0 10px 10px;
border: 1px solid #666;
padding: 2px;
}

HTML
<p>
 Lorem ipsum dolor sit amet, consectetur...
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet,
consectetur... Lorem ipsum dolor sit amet, consectetur... Lorem ipsum
dolor sit amet, consectetur... Lorem ipsum dolor sit amet,
consectetur... Lorem ipsum dolor sit amet, consectetur... Lorem ipsum
dolor sit amet, consectetur... Lorem ipsum dolor sit amet,
consectetur... Lorem ipsum dolor sit amet, consectetur... </p>
```

Παρατηρείται ότι μια εμφανίσιμη σελίδα δεν χρειάζεται μεγάλους μήκους κώδικα! Επίσης, ο σχεδιαστής ιστοσελίδων μπορεί να χρειαστεί εικόνες ελεύθερης διανομής, για να εμπλουτίσει τις σελίδες που κατασκευάζει. Στο διαδίκτυο υπάρχουν πολλές πηγές τέτοιων εικόνων, όπου οι καλλιτέχνες διαθέτουν ελεύθερα τις φωτογραφίες τους.

Δημιουργία μενού

Όμοια, για να δημιουργηθεί η Εικόνα 12.4 πρέπει να γραφτεί ο παρακάτω κώδικας. Παρατηρείται ότι ο κώδικας HTML χρησιμοποιεί λίστες. Η διαμόρφωση μέσω CSS μετατρέπει την απλή λίστα σε μενού.


```

<!DOCTYPE html>
<HTML>
<HEAD>
<TITLE> Αυτό είναι το πρώτο μου CSS μενού! </TITLE>
<STYLE>

#navlist
{
padding: 0;
margin: 0;
list-style-type: none;
float: left;
width: 100%;
color: #fff;
background-color: #036;
}

#navlist li { display: inline; }

#navlist li a
{
float: left;
width: 5em;
color: #fff;
background-color: #036;
padding: 0.2em 1em;
text-decoration: none;
border-right: 1px solid #fff;
}

#navlist li a:hover
{
background-color: #369;color: #fff;
}

</STYLE>
</HEAD>

<BODY>

<ul id="navlist">
<li><a href="#">Item one</a></li>
<li><a href="#">Item two</a></li>
<li><a href="#">Item three</a></li>
</ul>

<p> Στο παραπάνω παράδειγμα, κάνουμε τα στοιχεία ul και a να "επιπλέουν" (float) προς τα αριστερά. Τα στοιχεία li παρουσιάζονται εμβόλιμα (inline) χωρίς δηλαδή να υπάρχει αλλαγή γραμμής πριν ή μετά από αυτά. Με τον τρόπο αυτό επιβάλλεται, η λίστα να βρίσκεται όλη σε μια γραμμή. Το στοιχείο ul έχει πλάτος 100% και κάθε υπερσύνδεσμος στη λίστα έχει πλάτος 6em (που σημαίνει 6 φορές το μέγεθος της τρέχουσας γραμματοσειράς). Επίσης, προσθέτουμε κάποια χρώματα για να γίνει πιο όμορφο.</p>
<p> Τυπικά, στη θέση των # στα href θα μπου οι υπερσύνδεσμοι προς τις αντίστοιχες ιστοσελίδες του site σας και το κείμενο "Item one" (και τα υπόλοιπα) θα έχουν τους τίτλους των μενού σας. Για παράδειγμα, Αρχική σελίδα, Εικόνες, Επικοινωνία, κ.λπ. </p>
<p> Αλλάξτε το μέγεθος του παραθύρου παρουσίασης για να δείτε πως συμπεριφέρεται το μενού.</p>

</BODY>
</HTML>

```

Εκπαιδευτική Ενότητα: 6.3 Ανάπτυξη αλληλεπιδραστικών ιστοσελίδων (Javascript)

Η JavaScript (JS), είναι διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται.

Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία.

Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστραφές προστακτικό και συναρτησιακό στυλ προγραμματισμού.

Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων — τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side).

Μοντέλο εκτέλεσης

Η αρχική έκδοση της Javascript βασίστηκε στη σύνταξη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα

Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού).

Παρά την ευρεία χρήση της Javascript για συγγραφή προγραμμάτων σε περιβάλλον φυλλομετρητή, από την αρχή χρησιμοποιήθηκε και για τη συγγραφή κώδικα από την πλευρά του διακομιστή, από την ίδια τη Netscape στο προϊόν LiveWire, με μικρή επιτυχία. Η χρήση της Javascript στο διακομιστή εμφανίζεται πάλι σήμερα, με τη διάδοση του Node.js, ενός μοντέλου προγραμματισμού βασισμένο στα γεγονότα (events).

Διαφορά Javascript και Java

Η Javascript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης "Java" στο όνομα της γλώσσας έχει περισσότερη σχέση με το προφίλ του προϊόντος που έπρεπε να έχει και λιγότερο με κάποια πιθανή συμβατότητα ή άλλη στενή σχέση με τη Java. Ρόλο σε αυτήν τη σύγχυση έπαιξε και ότι η Java και η Javascript έχουν δεχτεί σημαντικές επιρροές από τη γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δύο αντικειμενοστρεφείς γλώσσες. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι "Javascript" και όχι 'Java script' σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές.

Η JavaScript είναι μια γλώσσα συγγραφής σεναρίων (scripting language), που χρησιμοποιείται για να προσθέσει εφέ και διαλογικότητα (αλληλεπίδραση, διαδραστικότητα, interactivity) στις ιστοσελίδες μας και είναι ανταγωνιστική της γλώσσας προγραμματισμού VBScript. Δημιουργήθηκε από την εταιρεία Netscape και το αρχικό της όνομα ήταν LiveScript.

Ο κώδικας της JavaScript γράφεται σε καθαρό κείμενο (ASCII μορφή) και ενσωματώνεται μέσα στον κώδικα της HTML, μπορεί δε να εκτελεσθεί αμέσως ή όταν λαμβάνει χώρα ένα συμβάν (event). Δεν γίνεται μεταγλώττιση (compilation) του κώδικα της JavaScript, αρκεί μόνο ο φυλλομετρητής (browser) να υποστηρίζει την JavaScript.

Αν και ακούγονται ίδιες, η Java και η JavaScript δεν έχουν καμία απολύτως σχέση μεταξύ τους, ούτε στη σύνταξή τους σαν γλώσσες προγραμματισμού ούτε και στις εφαρμογές που χρησιμοποιούνται.

Ανάλυση της JavaScript

Η Javascript αναλύεται , όπως θα δείτε παρακάτω:

A. Η Εντολή document.write()

Ένα πρώτο απλό παράδειγμα σε JavaScript είναι το εξής :

```
<html> <body> <script type="text/javascript"> document.write("Hello from  
JavaScript") </script> </body> </html>.
```

Ο κώδικας της JavaScript περιέχεται ανάμεσα στα tags `<script>` και `</script>` και σαν χαρακτηριστικό (attribute) μπορεί να χρησιμοποιηθεί το `type="text/javascript"` ή το `language="JavaScript"`. Μέσα σ' ένα αρχείο HTML μπορούν να υπάρχουν όσα σύνολα tags `<script>` και `</script>` χρειαστούν, είτε στο τμήμα head ή στο τμήμα body του εγγράφου.

Η εντολή `document.write` χρησιμοποιείται για να μπορούμε να εμφανίσουμε κείμενο στην οθόνη του φυλλομετρητή μέσα από εντολές της JavaScript. Το κείμενο `Hello from JavaScript` θα εμφανισθεί μαζί με το υπόλοιπο κείμενο του HTML κώδικα.

Αν και εδώ δεν φαίνεται καθαρά η αξία της JavaScript, θα φανεί όταν χρησιμοποιήσουμε εντολές ελέγχου, όπως είναι η `if`, για να μπορούμε να εμφανίσουμε διαφορετικό κείμενο ανάλογα με την τιμή κάποιων μεταβλητών ή ανάλογα με τις επιλογές που έχει κάνει ο χρήστης.

B. Η Εντολή document.last.Modified

Για να εμφανιστεί η ημερομηνία τελευταίας τροποποίησης μιας ιστοσελίδας, γράφεται ο εξής κώδικας της JavaScript μέσα στον HTML κώδικα :

```
<html> <body> <script type="text/javascript"> document.write("Ημερομηνία  
τελευταίας τροποποίησης της σελίδας : ") document.write(document.last.Modified)  
</script> </body> </html>.
```

Άνοιγμα Δευτερογενών Παραθύρων

Για να εμφανιστεί ένα άλλο παράθυρο (δευτερογενές ή εστιασμένο) εκτός από το βασικό παράθυρο του φυλλομετρητή και να τοποθετηθεί μέσα σ' αυτό τα περιεχόμενα ενός άλλου HTML εγγράφου, γράφεται ο εξής κώδικας :

```
<html> <head> <script type="text/javascript"> function WinOpen()  
{window.open("page01.html", "Window1", "toolbar=yes"); } </script> </head>  
</html>.
```

Δημιουργείται μια συνάρτηση (function) με όνομα WinOpen, η οποία περιέχει την εντολή window.open, με την οποία γίνεται να εμφανιστεί ένα δευτερογενές παράθυρο, και η οποία έχει τις εξής παραμέτρους (ορίσματα) :

- page01.html, που είναι το όνομα της ιστοσελίδας (εγγράφου), που θα εμφανισθεί μέσα στο δευτερογενές παράθυρο,
- Window1, που είναι το όνομα του νέου παραθύρου και
- toolbar=yes, δηλ. το νέο παράθυρο θα έχει γραμμή εργαλείων (toolbar).

Για να κληθεί τώρα η παραπάνω συνάρτηση και να εμφανισθεί το δευτερογενές παράθυρο, πρέπει μέσα στο ίδιο HTML έγγραφο και στο τμήμα <body> να γραφτεί ο εξής κώδικας για τη δημιουργία μιας φόρμας (form) :

```
<form> <input type="button" name="WindowButton" value="κλικ εδώ για να  
εμφανισθεί ένα νέο παράθυρο" onclick="WinOpen()"> </form>.
```

Δημιουργήθηκε μια φόρμα και τοποθετήθηκε μέσα της ένα πλήκτρο εντολής (button) με όνομα WindowButton και με μια ετικέτα (τίτλο) που προτρέπει τον χρήστη να κάνει κλικ. Η ιδιότητα onclick της φόρμας είναι ένας από τους χειριστές συμβάντων (event handlers) της JavaScript και εδώ καθορίζει ότι αν κάνουμε κλικ

στο πλήκτρο εντολής θα κληθεί η συνάρτηση WinOpen, οπότε και θα εμφανισθεί το παράθυρο με όνομα Window1 και μέσα του το έγγραφο (ιστοσελίδα) page01.html.

Προσθήκη Πλήκτρων Πλοήγησης

Στην JavaScript γίνεται να δημιουργηθούν πλήκτρα πλοήγησης (navigation buttons), τα οποία εκτελούν τις ίδιες λειτουργίες με τα γνωστά πλήκτρα Back (Πίσω) και Εμπρός (Forward) που υπάρχουν στη γραμμή εργαλείων ενός φυλλομετρητή.

```
<form> <input type="button" value="Πίσω δύο σελίδες" onclick="history.go(-2)">
<input type="button" value="Προηγούμενη σελίδα" onclick="history.go(- 1)"> <input
type="button" value="Επόμενη σελίδα" onclick="history.go(1)"> <input
type="button" value="Μπροστά δύο σελίδες" onclick="history.go(2)"> </form>.
```

Ο χειριστής συμβάντων onclick καλεί τη συνάρτηση history.go() με όρισμα τη σχετική σελίδα στην οποία θέλει να πάει κάνεις.

Δημιουργία Μηνυμάτων σε Πλαίσια Διαλόγου

Για να εμφανιστούν τα μηνύματα σε πλαίσια διαλόγου σε μια ιστοσελίδα, χρησιμοποιείται η συνάρτηση alert, η οποία εμφανίζει το κείμενο που της περνιέται σαν όρισμα σ' ένα πλαίσιο (παράθυρο) διαλόγου. Για να γίνει αλλαγή γραμμής στο κείμενο αυτό, πρέπει να συμπεριληφθεί τον ειδικό χαρακτήρα \n.

Το παρακάτω παράδειγμα εμφανίζει ένα μήνυμα μέσα σε μια ιστοσελίδα :

```
<script type="text/javascript"> alert("Καλωσήρθατε στη σελίδα της Φλώρινας")
</script>.
```

Γίνεται να χρησιμοποιηθούν και οι χειριστές συμβάντων onload και onunload, οι οποίοι καλούνται όταν έχει ολοκληρωθεί η φόρτωση μιας ιστοσελίδας ή όταν έχει φύγει κάποιος από μια ιστοσελίδα, αντίστοιχα.

Οι χειριστές αυτοί τοποθετούνται στο tag <body> ενός εγγράφου.

```
<body onload = alert("Καλωσήρθατε στη σελίδα της Φλώρινας")>
```

```
<body onload = alert("Σας άρεσε η περιήγηση στη Φλώρινα;")>
```

Μηνύματα στη Μπάρα Κατάστασης

Για να εμφανίσει ένα μήνυμα στη γραμμή κατάστασης (status bar) του παραθύρου του φυλλομετρητή, καταχωρείται μια τιμή στη μεταβλητή status του αντικειμένου window, ως εξής :

```
<a href="florina.html" onmouseover="window.status='Αν κάνετε κλικ θα πάτε στη σελίδα της Φλώρινας';return true"> Φλώρινα </a>.
```

Ο χειριστής συμβάντων onmouseover εκτελεί μια ενέργεια όταν ο δείκτης του ποντικιού βρεθεί πάνω από τον σύνδεσμο και στη συγκεκριμένη περίπτωση καταχωρεί μια τιμή στη μεταβλητή window.status, που έχει σαν αποτέλεσμα να εμφανισθεί ένα μήνυμα στη γραμμή κατάστασης.

Προσθήκη Μηνυμάτων σε Φόρμες

Γίνεται να προστεθούν μηνύματα στις φόρμες μιας HTML σελίδας τα οποία θα εμφανίζονται όταν κάνουμε κλικ σ' ένα στοιχείο της φόρμας ή όταν γίνεται εστίαση σ' ένα άλλο στοιχείο της φόρμας. Για να δημιουργηθεί ένα τέτοιο μήνυμα, γράφεται πρώτα το παρακάτω script στο τμήμα head :

```
<script type="text/javascript"> function AlertBox() { alert("Καλημέρα από τη Φλώρινα") } </script>.
```

Στο τμήμα head δηλώνονται οι συναρτήσεις που θα χρησιμοποιήθουν και μετά στον κώδικα για τη δημιουργία της φόρμας, γράφονται τα εξής :

```
<form> <input type="checkbox" name="check1" onclick="AlertBox"> Υπογράψτε </form>.
```

Όταν κάνει κάποιος κλικ στο πλαίσιο ελέγχου, καλείται η συνάρτηση AlertBox η οποία με τη σειρά της καλεί την ενσωματωμένη συνάρτηση alert και εμφανίζεται το μήνυμα.

Αυτόματη Προώθηση σε Άλλη Σελίδα

Για να προωθήθουν αυτόματα οι επισκέπτες μιας ιστοσελίδας σε μια άλλη ιστοσελίδα, γράφονται τα εξής :

```
<head> <script type="text/javascript"> alert("Η διεύθυνση της σελίδας έχει αλλάξει") </script> </head> <body onload = "location= 'new.html' ">.
```

Μόλις ο φυλλομετρητής αρχίζει να διαβάζει το HTML έγγραφο, εμφανίζεται το πλαίσιο διαλόγου alert με το προειδοποιητικό μήνυμα και μόλις κάνουμε κλικ στο OK για να το κλείσουμε, φορτώνεται η ιστοσελίδα, εκτελείται ο χειριστής συμβάντων onload και φορτώνεται η ιστοσελίδα new.html.

Η Συνάρτηση Confirm

Η συνάρτηση confirm() εμφανίζει ένα πλαίσιο επιβεβαίωσης σ' ένα πλαίσιο διαλόγου με τα πλήκτρα OK και Cancel, όπου αν κάνουμε κλικ στο OK θα επιστραφεί η τιμή true, ενώ αν κάνουμε κλικ στο Cancel θα επιστραφεί η τιμή false.

Παράδειγμα:

```
<head> <script type="text/javascript"> function redirect() { if (confirm("Η διεύθυνση της σελίδας έχει αλλάξει")) { location = 'new.html' } } </script> </head> <body onload = "redirect()">.
```

Αφού ολοκληρωθεί το φόρτωμα της σελίδας θα κληθεί η συνάρτηση redirect() και θα εμφανισθεί ένα πλαίσιο διαλόγου επιβεβαίωσης με το παραπάνω επεξηγηματικό κείμενο και τα πλήκτρα OK και Cancel. Αν κάνουμε κλικ στο πλήκτρο OK, θα εκτελεσθεί η εντολή if και θα πάμε στην καινούργια ιστοσελίδα, σύμφωνα με την τιμή της μεταβλητής location.

Τα Αρχεία Σεναρίων

Γίνεται να γραφτούν σενάρια (scripts) της JavaScript σ' ένα ξεχωριστό αρχείο κειμένου, διαφορετικό από το HTML έγγραφο, και μετά να συνδεθούν χρησιμοποιώντας την ιδιότητα SRC της ετικέτας (tag) <script>. Όταν ο φυλλομετρητής διαβάσει τη σελίδα, θα δει την ιδιότητα SRC και θα διαβάσει το αρχείο σεναρίου σαν αυτό να βρισκόταν μέσα στο HTML έγγραφο.

Αυτή η ιδιότητα είναι ιδιαίτερα χρήσιμη, όταν δημιουργούνται πολύπλοκα σενάρια και χρησιμοποιούνται σε πολλές ιστοσελίδες. Για παράδειγμα, γίνεται να δημιουργηθεί ένα απλό αρχείο κειμένου με το εξής κείμενο :

```
<!-- document.write("Ημερομηνία τελευταίας τροποποίησης της σελίδας :  
")document.write(document.lastModified) //-->
```

Αποθηκεύουμε το αρχείο αυτό μ' ένα όνομα με επέκταση .js, π.χ. doc01.js, και μέσα στον κώδικα HTML της ιστοσελίδας γράφουμε τα εξής :

```
<script type="text/javascript" SRC="doc01.js"> </script>.
```

Τα Σχόλια στην JavaScript

Για να εισάγουμε σχόλια (comments) στην JavaScript, μπορούμε να γράψουμε τα εξής :

```
// αυτό είναι ένα σχόλιο ή /* αυτή είναι η αρχή ενός μεγαλύτερου σχολίου, αυτή  
είναι η δεύτερη γραμμή του και αυτή είναι η τελευταία γραμμή του */.
```

Για να βάλουμε ένα απλό σχόλιο σ' ένα script, χρησιμοποιούμε τους χαρακτήρες //, οπότε ο φυλλομετρητής αγνοεί οτιδήποτε βρίσκεται μετά και πηγαίνει στην επόμενη γραμμή. Τους χαρακτήρες // μπορούμε να τους τοποθετήσουμε και στη μέση μιας γραμμής εντολής ενός script.

Αν θέλουμε να γράψουμε ένα μεγαλύτερο σχόλιο που να επεκτείνεται σε πολλές γραμμές, μπορούμε να ξεκινήσουμε την πρώτη γραμμή με τους χαρακτήρες /* και να κλείσουμε το σχόλιο τελειώνοντας την τελευταία γραμμή με τους χαρακτήρες */.

Οι Μεταβλητές στην JavaScript

Για τη δήλωση των μεταβλητών στην JavaScript χρησιμοποιούμε τη λέξη κλειδί `var` και μπορούμε να καταχωρήσουμε και τιμές σε μια μεταβλητή μαζί με τη δήλωσή της, ως εξής:

```
var firstname
```

```
var country="Greece"
```

```
var salary=300000
```

Μπορούμε να δηλώσουμε μεταβλητές και χωρίς τη λέξη κλειδί `var` απλά και μόνο με την καταχώρηση τιμής.

Οι κανόνες για την ονομασία των μεταβλητών της JavaScript είναι οι εξής :

- Ο πρώτος χαρακτήρας πρέπει να είναι γράμα (a-z ή A-Z) ή ο χαρακτήρας `_`.
- Το υπόλοιπο του ονόματος μπορεί να περιέχει γράματα, ψηφία ή τον χαρακτήρα `_`.
- Δεν πρέπει να υπάρχουν κενά ανάμεσα στα ονόματα.
- Δεν πρέπει να χρησιμοποιούμε σαν ονόματα μεταβλητών τις λέξεις κλειδιά της JavaScript.

Δεν πρέπει να ξεχνάμε ότι η JavaScript ξεχωρίζει τα πεζά από τα κεφαλαία γράματα (case sensitive). Δηλαδή, η μεταβλητή `Florina` είναι διαφορετική από τις μεταβλητές `FLORINA` και `florina`.

Η JavaScript χρησιμοποιεί αλφαριθμητικές, αριθμητικές και λογικές μεταβλητές και υπάρχει μια ειδική λέξη κλειδί, η `null`, η οποία επέχει τη θέση κενής μεταβλητής, δηλ. μεταβλητής χωρίς περιεχόμενο.

Ο τύπος δεδομένων μιας μεταβλητής καθορίζεται από την τοποθέτηση δεδομένων σ' αυτήν αλλά μπορεί να αλλάξει στην πορεία. Για παράδειγμα, μπορούμε να έχουμε τα εξής :

```
var Member=5 και var Member=False
```

Οι μεταβλητές μπορούν να δηλωθούν εσωτερικά ή εξωτερικά μιας συνάρτησης :

Οι Τελεστές της JavaScript

Οι αριθμητικοί τελεστές που χρησιμοποιεί η JavaScript είναι οι εξής:

- = εκχώρηση τιμών σε μεταβλητές.
- +, πρόσθεση τιμών.
- -, αφαίρεση τιμών.
- *, πολλαπλασιασμός τιμών.
- /, διαίρεση τιμών.
- %, τελεστής υπολοίπου ακέραιας διαίρεσης.
- ++, τελεστής μοναδιαίας αύξησης.
- --, τελεστής μοναδιαίας μείωσης.

Ο αλφαριθμητικός τελεστής που χρησιμοποιεί η JavaScript είναι ο εξής ένας :

- +, ένωση στοιχείων σειρών (strings).

Η χρήση του τελεστή + για την πρόσθεση στοιχείο-σειρών με αριθμούς μετατρέπει τις αριθμητικές τιμές σε στοιχείο-σειρές. Ακολουθούν παραδείγματα :

```
var sMonth="Αύγουστος"
```

```
var nYear=2001
```

```
var nDay=1
```

```
var sDate=sMonth + " " + nDay + ", " + nYear
```

Το sDate θα γίνει ίσο με Αύγουστος 1, 2001.

a="8" + 7 + 2001, όπου το a θα γίνει ίσο με 872001 και όχι με 2016.

Οι λογικοί τελεστές που χρησιμοποιεί η JavaScript είναι οι εξής :

- !, λογικό Not.
- &&, λογικό And.

- `||`, λογικό Or

Η JavaScript χρησιμοποιεί κι άλλους τελεστές εκχώρησης :

- `+=`, το `x+=y` είναι ισοδύναμο με το `x=x+y`.
- `-=`, το `x-=y` είναι ισοδύναμο με το `x=x-y`.
- `*=`, το `x*=y` είναι ισοδύναμο με το `x=x*y`.
- `/=`, το `x/=y` είναι ισοδύναμο με το `x=x/y`.
- `%=`, το `x%=y` είναι ισοδύναμο με το `x=x%y`.

Η Εντολή If

Η σύνταξη της εντολής if είναι η εξής : if (συνθήκη) { ... εντολές ... } else { ... εντολές ... }.

Για τη σύγκριση τιμών, η JavaScript χρησιμοποιεί τον τελεστή `==`. Ακολουθεί ένα παράδειγμα :

```
var bStatus=true function CheckStatus() { if (bStatus) { alert("Η bStatus είναι αληθής") } else { alert("Η bStatus είναι ψευδής") } }.
```

Δημιουργούμε και δύο πλήκτρα εντολής μέσα σε μια φόρμα :

```
<form> <input type="button" value="Αληθής"
onclick="bStatus=true;CheckStatus()">
<input type="button" value="Ψευδής" onclick="bStatus=false;CheckStatus()">
</form>.
```

Ανάλογα σε ποιο πλήκτρο κάνουμε κλικ, στη μεταβλητή `bStatus` εκχωρείται η τιμή `true` ή η τιμή `false`. Και στις δύο περιπτώσεις, αμέσως μετά την εκχώρηση, ο χειριστής συμβάντων `onclick` καλεί τη συνάρτηση `CheckStatus`, η οποία ελέγχει την τιμή της λογικής μεταβλητής `bStatus` και εμφανίζει ένα ανάλογο μήνυμα.

Οι Τελεστές Σύγκρισης της JavaScript

Οι τελεστές σύγκρισης που χρησιμοποιεί η JavaScript είναι οι εξής :

- `=`, ελέγχει αν δύο τιμές είναι ίσες.
- `!=`, ελέγχει αν δύο τιμές είναι άνισες.
- `>`, ελέγχει αν μια τιμή είναι μεγαλύτερη από μια άλλη.
- `<`, ελέγχει αν μια τιμή είναι μικρότερη από μια άλλη.
- `>=`, ελέγχει αν μια τιμή είναι μεγαλύτερη ή ίση από μια άλλη.
- `<=`, ελέγχει αν μια τιμή είναι μικρότερη ή ίση από μια άλλη.

Οι Λογικοί Τελεστές της JavaScript

Οι λογικοί (boolean) τελεστές χρησιμοποιούνται για να συνδυάζουν εκφράσεις που περιέχουν συγκρίσεις. Παράδειγμα:

```
function function1() { var dToday=new Date() var nHours=dToday.getHours() var nDay=dToday.getDay() if ((nDay= =5) && (nHours>=12)) { alert("Είναι Παρασκευή απόγευμα") } else { if ((nDay= =6) || (nDay= =0)) { alert("Ηρθε το Σαββατοκύριακο") } else { alert("Άλλη μια βαρετή μέρα") } // end else } // end if } // end function
```

Ακολουθεί κι ένα παράδειγμα με τον τελεστή άρνησης (!) : `var bStatus=true function CheckStatus() { if (!bStatus) { alert("Η bStatus είναι ψευδής") } else { alert("Η bStatus είναι αληθής") } .}`

Ο Τριαδικός Τελεστής Σύγκρισης

Ο τριαδικός τελεστής σύγκρισης κάνει ταυτόχρονα σύγκριση και καταχώρηση τιμής σε μεταβλητή και η σύνταξή του είναι η εξής : `μεταβλητή = (συνθήκη) ? τιμή1 : τιμή2.`

Αν η συνθήκη είναι αληθής, τότε η μεταβλητή γίνεται ίση με την τιμή1, αλλιώς γίνεται ίση με την τιμή2. Ακολουθεί ένα παράδειγμα:

```
function function1() { var dToday=new Date() var nSeconds=dToday.getSeconds() var sText = (nSeconds > 30) ? "ΠΕΡΙΣΣΟΤΕΡΟ" : "ΛΙΓΟΤΕΡΟ" alert("Πέρασε " + sText + " από το μισό αυτού του λεπτού") }.
```

Η Εντολή For

Η σύνταξη της εντολής for είναι η εξής : for (αρχική τιμή; συνθήκη; εντολή) { ... εντολές ... }.

Παράδειγμα:

```
function1() { var i for (i=1; i<4; i++) { alert("Η τιμή του i είναι : " + i) } alert("Ο βρόχος ολοκληρώθηκε και το i είναι ίσο με : " + i) }.
```

Η Εντολή While

Η σύνταξη της εντολής while είναι η εξής : while (συνθήκη) { ... εντολές ... }.

Παράδειγμα:

```
function function1() { var i=1 while (i<4) { alert("Η τιμή του i είναι : " + i) i++ } alert("Ο βρόχος ολοκληρώθηκε και το i είναι ίσο με : " + i) }.
```

Η Συνάρτηση Prompt

Η συνάρτηση prompt(εμφανίζει ένα μήνυμα σ' ένα πλαίσιο διαλόγου στην οθόνη και ό,τι γράψουμε το εκχωρεί σε μια μεταβλητή).

Η Εντολή Do While

Η εντολή do while μοιάζει με την εντολή βρόχου while, με τη διαφορά ότι περιέχει τη συνθήκη μετά τις εντολές και αυτό σημαίνει ότι οι εντολές που περιέχει θα εκτελεστούν τουλάχιστον μία φορά, άσχετα από το αν η συνθήκη είναι αληθής ή ψευδής.

Η σύνταξη της εντολής do while είναι η εξής : do { ... εντολές ... } while (συνθήκη).

Η Εντολή Break

Μπορεί να χρησιμοποιηθεί η εντολή break για να βγει κάποιος από έναν βρόχο for ή while και να πάει στην επόμενη εντολή μετά τον βρόχο, χωρίς κατ' ανάγκη να σταματήσει να αληθεύει η συνθήκη του βρόχου.

Η εντολή (συνάρτηση) confirm εμφανίζει ένα πλαίσιο επιβεβαίωσης με το σχετικό κείμενο και με τα πλήκτρα OK και Cancel και αν κάνουμε κλικ στο OK, η μεταβλητή bResult θα γίνει ίση με true, ενώ αν κάνουμε κλικ στο Cancel, η μεταβλητή bResult θα γίνει ίση με false.

Η Εντολή Continue

Μπορεί να χρησιμοποιηθεί η εντολή continue για να πάει κάποιος στην αρχή ενός βρόχου for ή while, χωρίς να εκτελεστούν οι εντολές του βρόχου που ακολουθούν την εντολή continue.

Σύμφωνα με τον παραπάνω κώδικα, αν πατάμε συνέχεια το πλήκτρο Cancel στο πλαίσιο επιβεβαίωσης, θα εκτελείται η εντολή continue και θα πηγαίνουμε στην αρχή του βρόχου χωρίς να εκτελείται η εντολή i++. Άρα, δεν θα αλλάζει η τιμή του μετρητή και δεν θα μπορούμε να βγούμε από τον βρόχο. Για να βγούμε από τον βρόχο, θα πρέπει να πατάμε το πλήκτρο OK στο πλαίσιο επιβεβαίωσης.

Οι Συναρτήσεις (Functions) της JavaScript

Ο ορισμός μιας συνάρτησης (function) είναι μια διαδικασία κατά την οποία δηλώνεται το όνομά της και εξηγείται το τι κάνει. Η σύνταξη μιας συνάρτησης είναι η εξής :

```
function ΌνομαΣυνάρτησης() { ... εντολές ... }
```

Γίνεται να ορισθούν συναρτήσεις οπουδήποτε σε μια ιστοσελίδα, αλλά πρέπει υπάρχει προσοχή να ορισθούν πριν καλεστούν και είναι καλύτερα να οριστούν στο

τμήμα <head> ενός εγγράφου. Οι συναρτήσεις επιστρέφουν τιμές που μπορούν να εκχωρηθούν σε μεταβλητές, αν η κλήση τους γίνει ως εξής μεταβλητή = ΌνομαΣυνάρτησης().

Έχουμε ορίσει μια συνάρτηση με όνομα function1, η οποία περιέχει μια τοπική μεταβλητή στοιχείο-σειράς με όνομα sTypedText και χρησιμοποιούμε την ενσωματωμένη συνάρτηση prompt() για να γράψουμε ένα κείμενο σ' ένα πλαίσιο κειμένου, το οποίο κείμενο καταχωρείται στη μεταβλητή sTypedText και επιστρέφεται εκεί απ' όπου κλήθηκε η συνάρτηση με την εντολή return.

Οι Πίνακες (Arrays)

Για να δημιουργήσουμε έναν πίνακα στην JavaScript, απαιτούνται τα εξής τρία στάδια :

- Δημιουργία μιας νέας τάξης αντικειμένων με μια ειδική συνάρτηση.
- Κλήση της συνάρτησης.
- Εισαγωγή των δεδομένων στον πίνακα.

Αντικείμενα, Ιδιότητες και Μέθοδοι

Ένα αντικείμενο (object) διαθέτει ένα σύνολο από ιδιότητες (properties) και μεθόδους (methods) που συνδέονται μ' αυτό. Η JavaScript περιέχει έναν αριθμό αντικειμένων, όπως είναι το αντικείμενο document, που είναι η ηλεκτρονική σελίδα που δημιουργεί η HTML, το αντικείμενο window και το αντικείμενο form. Οι ιδιότητες ενός αντικειμένου είναι μεταβλητές που ανήκουν στο αντικείμενο και καθορίζονται με τη χρήση μιας τελείας (dot notation), ως εξής :

Όνομα Αντικειμένου-Όνομα Ιδιότητα

Μια μέθοδος είναι απλά μια συνάρτηση που συνδέεται μ' ένα συγκεκριμένο αντικείμενο και μπορεί να επιστρέψει μια τιμή. Ο γενικός τρόπος χρήσης μιας μεθόδου είναι ο εξής : όνομα Αντικειμένου. Όνομα Μεθόδου(παράμετροι).

Παραδείγματα:

Document.write(navigator.appName)

Η ιδιότητα `appName` ανήκει στο αντικείμενο `navigator` και περιέχει το όνομα αναγνώρισης του φυλλομετρητή. Το αντικείμενο `navigator` περιγράφει τον φυλλομετρητή που διαβάζει την ιστοσελίδα που περιέχει το σενάριο. Ακόμη, στο παραπάνω παράδειγμα, η μέθοδος `write()` συνδέεται με το αντικείμενο εγγράφου `document` και εμφανίζει κάποιο κείμενο στην ιστοσελίδα.

document.bgColor="red"

Το παραπάνω παράδειγμα ορίζει την ιδιότητα του χρώματος φόντου της σελίδας σε κόκκινο (`red`).

```
var sText
```

```
var sUpper
```

```
sText = prompt("Γράψτε κάτι με μικρά γράματα", " ")
```

```
sUpper = sText.toUpperCase()
```

```
document.write("Το κείμενο με κεφαλαία γράματα είναι : <p><h3>" + sUpper +  
"</h3>")
```

Εδώ δηλώσαμε τις μεταβλητές `sText` και `sUpper` και με τη μέθοδο `prompt` δώσαμε τιμή στη μεταβλητή `sText`. Η `prompt` είναι μια μέθοδος του αντικειμένου `window`, όπως και οι μέθοδοι `alert` και `confirm`, αλλά δεν είναι ανάγκη να γράψουμε ολόκληρα τα `window.prompt`, `window.alert` και `window.confirm`. Μετά, με τη μέθοδο `toUpperCase()` των αντικειμένων στοιχειοσειράς, μετατρέπουμε σε κεφαλαία γράματα το κείμενο της μεταβλητής `sText` και το καταχωρούμε στη μεταβλητή `sUpper`.

document.Customer.Lastname.value = "Florina"

Η παραπάνω εντολή καταχωρεί το κείμενο Florina στην ιδιότητα value του πλαισίου κειμένου Last_name της φόρμας Customer του τρέχοντος εγγράφου.

Φόρμας Αναφορά σε Φόρμες και σε Στοιχεία

Μπορούμε να αναφερόμαστε σε μια φόρμα χρησιμοποιώντας τον πίνακα forms[], ο οποίος περιέχει μια λίστα απ' όλα τα αντικείμενα φόρμας που υπάρχουν στο έγγραφο.

Παρόμοια, μπορούμε να χρησιμοποιήσουμε την ιδιότητα elements[] μιας φόρμας για να αναφερόμαστε στα στοιχεία μιας φόρμας, η οποία ιδιότητα είναι μια λίστα απ' όλα τα αντικείμενα-στοιχεία που περιέχει η φόρμα. Η αρίθμηση ξεκινά από το 0.

Υπάρχουν δύο τρόποι για να αναφερθούμε σε μια φόρμα. Αν η φόρμα που μας ενδιαφέρει λέγεται secondForm και είναι η δεύτερη φόρμα σε μια ιστοσελίδα και θέλουμε να βρούμε πόσα στοιχεία (elements) περιέχει, μπορούμε να χρησιμοποιήσουμε την ιδιότητα length της φόρμας μ' έναν από τους εξής δύο τρόπους :

document.forms[1].length

document.secondForm.length

Παρόμοια, μπορούμε να αναφερθούμε στα στοιχεία μιας φόρμας με το όνομα ή τη θέση τους μέσα στον πίνακα στοιχείων. Για παράδειγμα, για να πάρουμε την τιμή ενός πλαισίου κειμένου με όνομα Lname που βρίσκεται στη φόρμα secondForm και είναι το τέταρτο κατά σειρά στοιχείο της φόρμας, μπορούμε να χρησιμοποιήσουμε έναν από τους εξής τέσσερις τρόπους :

- document.secondForm.Lname.value
- document.secondForm.elements[3].value
- document.forms[1].Lname.value
- document.forms[1].elements[3].value

Το Αντικείμενο Date

Το αντικείμενο *Date* μάς δίνει τη δυνατότητα να δουλεύουμε με ημερομηνίες και ώρες. Για να γίνει αυτό, πρέπει να δημιουργήσουμε ένα αντίγραφο αντικειμένου με τη λέξη κλειδί *new*, ως εξής :

```
Today = new Date()
```

Όταν δημιουργούμε ένα αντικείμενο *Date*, μπορούμε να πάρουμε την τρέχουσα ημερομηνία και ώρα του συστήματος, όπως κάναμε πριν, ή να ορίσουμε μια δική μας ημερομηνία και ώρα, μ' έναν από τους εξής τρόπους :

- `Date1 = new Date("8, 30, 2001, 10:50:14")`
- `Date2 = new Date(2001, 8, 30)`
- `Date3 = new Date(2001, 8, 30, 10, 50, 14)`

Για το αντικείμενο *Date* υπάρχουν οι εξής μέθοδοι :

- `getDate()`, επιστρέφει την ημερομηνία.
- `getDay()`, επιστρέφει την ημέρα της εβδομάδας.
- `getHours()`, επιστρέφει την ώρα.
- `getMinutes()`, επιστρέφει τα λεπτά.
- `getMonth()`, επιστρέφει τον μήνα.
- `getSeconds()`, επιστρέφει τα δευτερόλεπτα.
- `getTime()`, επιστρέφει την πλήρη ώρα.
- `getTimeZoneoffset()`, επιστρέφει τη διαφορά από την ώρα Greenwich.
- `getFullYear()`, επιστρέφει το έτος,
- `parse()`, επιστρέφει τα χιλιοστά του δευτερολέπτου από 1/1/1970.
- `setDate()`, ορίζει την ημερομηνία.
- `setHours()`, ορίζει την ώρα.

- `setMinutes()`, ορίζει τα λεπτά.
- `setMonth()`, ορίζει τον μήνα.
- `setSeconds()`, ορίζει τα δευτερόλεπτα.
- `setTime()`, ορίζει την πλήρη ώρα.
- `setYear()`, ορίζει το έτος.
- `toGMTString()`, μετατρέπει την ημερομηνία σε στοιχείο-σειρά ώρας Greenwich.
- `toLocaleString()`, μετατρέπει την ημερομηνία σε στοιχείο σειρά.
- `UTC()`, επιστρέφει τα χιλιοστά του δευτερολέπτου από 1/1/1970.

Το Αντικείμενο Math

Το αντικείμενο `Math` περιέχει ιδιότητες και μεθόδους που χρησιμοποιούνται για να εκτελέσουμε μαθηματικούς υπολογισμούς. Οι ιδιότητες του αντικειμένου είναι σταθερές, όπως το π , το e κ.ά., ενώ οι μέθοδοι του αντικειμένου είναι συναρτήσεις εύρεσης μεγίστου, εύρεσης ημιτόνου κ.ά.

Οι ιδιότητες του αντικειμένου Math είναι οι εξής :

- `E`, η σταθερά $e = 2,718$.
- `LN2`, ο φυσικός λογάριθμος του 2 (περίπου 0,693).
- `LN10`, ο φυσικός λογάριθμος του 10 (περίπου 2,302).
- `LOG2E`, ο λογάριθμος του e με βάση το 2 (περίπου 1,442).
- `LOG10E`, ο δεκαδικός λογάριθμος του e (περίπου 0,434).
- `PI`, ο αριθμός $\pi = 3,1415926$.
- `SQRT1_2`, η τετραγωνική ρίζα του $1/2$ (περίπου 0,707).
- `SQRT2`, η τετραγωνική ρίζα του 2 (περίπου 1,414).

Οι μέθοδοι του αντικειμένου *Math* είναι οι εξής :

- `abs()`, επιστρέφει την απόλυτη τιμή ενός αριθμού.
- `acos()`, επιστρέφει το συνημίτονο τόξου ενός αριθμού.
- `asin()`, επιστρέφει το ημίτονο τόξου ενός αριθμού.
- `atan()`, επιστρέφει την εφαπτομένη τόξου ενός αριθμού.
- `ceil()`, επιστρέφει τον αμέσως επόμενο ακέραιο ενός αριθμού.
- `cos()`, επιστρέφει το συνημίτονο ενός αριθμού.
- `exp()`, επιστρέφει τη δύναμη ενός αριθμού με βάση το *e*.
- `floor()`, επιστρέφει τον αμέσως προηγούμενο ακέραιο ενός αριθμού.
- `log()`, επιστρέφει τον φυσικό λογάριθμο ενός αριθμού (με βάση το *e*).
- `max()`, επιστρέφει τον μεγαλύτερο από δύο αριθμούς.
- `min()`, επιστρέφει τον μικρότερο από δύο αριθμούς.
- `pow()`, επιστρέφει τη δύναμη ενός αριθμούς ως προς μια βάση.
- `round()`, στρογγυλοποιεί έναν αριθμό στον πλησιέστερο ακέραιο.
- `sin()`, επιστρέφει το ημίτονο ενός αριθμού.
- `sqrt()`, επιστρέφει την τετραγωνική ρίζα ενός αριθμού.
- `tan()`, επιστρέφει την εφαπτομένη ενός αριθμού.

Παράδειγμα

```
form.result.value = Math.max(form.num1.value, form.num2.value).
```

Η παραπάνω εντολή βρίσκει τον μεγαλύτερο από δύο αριθμούς που είναι καταχωρημένοι σε δύο πλαίσια κειμένου μιας φόρμας και τον καταχωρεί σ' ένα τρίτο πλαίσιο κειμένου.

```
form.result.value = 2*Math.PI*form.num1.value
```

```
form.result2.value = Math.sqrt(form.num1.value)
```

Αντικείμενο String

Ενα αντικείμενο στοιχείο-σειράς (string) έχει μόνο μία ιδιότητα, που είναι το μήκος (length), δηλ. ο αριθμός των χαρακτήρων της στοιχείο-σειράς, αλλά πολλές μεθόδους :

anchor(όνομα), μετατρέπει τη στοιχείο-σειρά σε ετικέτα δεσμού HTML (<a name =), χρησιμοποιώντας το όνομα σαν όνομα δεσμού.

- big(), δίνει μεγαλύτερο μέγεθος στη στοιχείο-σειρά (<big>).
- blink(), κάνει τη στοιχείο-σειρά να αναβοσβήνει (<blink>).
- bold(), μετατρέπει τη στοιχείο-σειρά σε έντονη ().
- charAt(κατάταξη), επιστρέφει τον χαρακτήρα της στοιχειοσειράς από τη θέση κατάταξη, όπου ο πρώτος χαρακτήρας είναι στη θέση 0.
- fixed(), μετατρέπει τη γραματοσειρά της στοιχειοσειράς σε μη αναλογική (<tt>).
- Fontcolor (χρώμα), αλλάζει το χρώμα της στοιχειοσειράς ().
- Fontsize (μέγεθος), αλλάζει το μέγεθος της στοιχειοσειράς ().
- indexOf(χαρακτήρας, από), ψάχνει στη στοιχείο-σειρά για έναν συγκεκριμένο χαρακτήρα από τη θέση από και επιστρέφει τη θέση του χαρακτήρα αυτού, που ο πρώτος χαρακτήρας έχει την θέση 1.
- italics(), μετατρέπει τη στοιχείο-σειρά σε πλάγια γράματα (<i>).
- lastindexOf(χαρακτήρας, από), σαν την indexOf(), αλλά ψάχνει από το τέλος.
- link(href), μετατρέπει τη στοιχείο-σειρά σε ετικέτα δεσμού HTML (<a href =), χρησιμοποιώντας το href σαν όνομα δεσμού.

- `small()`, δίνει μικρότερο μέγεθος στη στοιχείο-σειρά (`<small>`).
- `strike()`, μετατρέπει τη στοιχείο-σειρά σε διαγραμμένη (`<strike>`).
- `sub()`, μετατρέπει τη στοιχείο-σειρά σε μορφή δείκτη (`<sub>`).
- `Substring (κατάταξη1, κατάταξη2)`, επιστρέφει ένα κομμάτι της στοιχείο-σειράς από τη θέση κατάταξη1 έως τη θέση κατάταξη2.
- `sup()`, μετατρέπει τη στοιχείο-σειρά σε μορφή εκθέτη (`<sup>`).
- `toLowerCase()`, μετατρέπει τη στοιχείο-σειρά σε πεζά γράμματα.
- `toUpperCase()`, μετατρέπει τη στοιχείο-σειρά σε κεφαλαία γράμματα.

Παραδείγματα

```
function GetLength(form) { var stextLength = form.string1.value form.result.value = stextLength.length }.
```

```
function Position(form) { var sPos = form.string1.value form.result2.value = sPos.charAt(5) }.
```

Η συνάρτηση `GetLength()` παίρνει το κείμενο του πλαισίου κειμένου `string1` και το τοποθετεί στη μεταβλητή `stextLength`. Μετά βρίσκει πόσοι χαρακτήρες υπάρχουν στη μεταβλητή αυτή, χρησιμοποιώντας την ιδιότητα `length`, και επιστρέφει αυτή την τιμή στο πλαίσιο κειμένου `result` της φόρμας.

Η συνάρτηση `Position()` παίρνει το κείμενο του πλαισίου κειμένου `string1` και το τοποθετεί στη μεταβλητή `sPos`. Μετά χρησιμοποιεί τη μέθοδο `charAt()` για να βρει τον 6ο χαρακτήρα της στοιχειοσειράς και τον καταχωρεί στο πλαίσιο κειμένου `result2` της φόρμας.

Ο Χειριστής Συμβάντων Onblur

Εκτελείται όταν απομακρυνόμαστε από ένα στοιχείο φόρμας. Το `blur` είναι το συμβάν της απομάκρυνσης από την εστίαση (`focus`) σ' ένα στοιχείο φόρμας. Όταν

δηλαδή, φεύγουμε από ένα στοιχείο φόρμας και πάμε σ' ένα άλλο στοιχείο φόρμας ή απλά πατάμε το ποντίκι οπουδήποτε έξω από το στοιχείο φόρμας ή όταν πατάμε το πλήκτρο Tab.

Ο Χειριστής Συμβάντων Onchange

Εκτελείται όταν απομακρυνόμαστε από ένα στοιχείο φόρμας εφόσον έχουμε αλλάξει το περιεχόμενο του στοιχείου. Μοιάζει πολύ με τον χειριστή συμβάντων onblur με τη διαφορά ότι πρέπει να έχουμε κάνει κάποια αλλαγή στο στοιχείο φόρμας πριν απομακρυνθούμε απ' αυτό, για να μπορέσει να ενεργοποιηθεί ο χειριστής.

Ο Χειριστής Συμβάντων Onclick

Εκτελείται όταν κάνουμε κλικ σ' ένα στοιχείο φόρμας ή σύνδεσμο και είναι ο πιο συχνά χρησιμοποιούμενος χειριστής συμβάντων.

Ο Χειριστής Συμβάντων Onfocus

Εκτελείται όταν εστιάζουμε σ' ένα στοιχείο φόρμας. Είναι το αντίθετο του onblur.

Ο Χειριστής Συμβάντων Onload

Εκτελείται όταν φορτώνεται μια ιστοσελίδα ή μια εικόνα. Μπορεί να τοποθετηθεί μόνο στις ετικέτες <body>, και <frameset>.

Ο Χειριστής Συμβάντων Onmouseover

Εκτελείται όταν ο δρομέας βρίσκεται πάνω από έναν σύνδεσμο. Χρησιμοποιείται συνήθως για να εμφανίσουμε μηνύματα στη γραμμή κατάστασης.

Ο Χειριστής Συμβάντων Onselect

Εκτελείται όταν επιλέγουμε κάτι σ' ένα στοιχείο φόρμας.

Ο Χειριστής Συμβάντων Onsubmit

Εκτελείται όταν υποβάλλουμε μια φόρμα. Χρησιμοποιείται συνήθως για να μπορούμε να κάνουμε επικύρωση (validation) μιας φόρμας. Εκτελεί ένα σενάριο όταν κάνουμε κλικ στο πλήκτρο υποβολής Submit μιας φόρμας.

Ο Χειριστής Συμβάντων Onunload

Εκτελείται όταν η τρέχουσα ιστοσελίδα κλείνει ή αντικαθίσταται από μια άλλη. Μπορεί να τοποθετηθεί μόνο στις ετικέτες <body>, και <frameset>.

Ο Χειριστής Συμβάντων Onerror

Εκτελείται όταν δεν μπορεί να φορτωθεί μια εικόνα. Μπορεί να τοποθετηθεί μόνο στην ετικέτα .

Ο Χειριστής Συμβάντων Onabort

Εκτελείται όταν διακόπτουμε το φόρτωμα μιας εικόνας. Μπορεί να τοποθετηθεί μόνο στην ετικέτα .

Το αντικείμενο window

Βρίσκεται στην κορυφή της ιεραρχίας των αντικειμένων και περιέχει ιδιότητες και μεθόδους που σχεδιάστηκαν για να παρέχουν πληροφορίες και να μας βοηθούν να επεξεργαζόμαστε το ίδιο το παράθυρο αλλά και το έγγραφο που περιέχει. Οι ιδιότητες του αντικειμένου window είναι οι εξής :

- defaultStatus, το προκαθορισμένο κείμενο της γραμμής κατάστασης.

- `history`, η λίστα ιστορικού του παραθύρου.
- `length`, ο αριθμός των πλαισίων (`frames`) του παραθύρου.
- `name`, το όνομα του παραθύρου.
- `opener`, αναφέρεται στο παράθυρο στο οποίο ένα σενάριο χρησιμοποίησε την εντολή `window.open` για να ανοίξει το τρέχον παράθυρο.
- `parent`, το γονικό παράθυρο σ' ένα σύνολο πλαισίων.
- `self`, το τρέχον παράθυρο.
- `status`, το κείμενο της γραμμής κατάστασης.
- `top`, το παράθυρο κορυφής.
- `window`, το τρέχον παράθυρο.

Υπάρχουν και οι εξής ιδιότητες, που αποτελούν οι ίδιες αντικείμενα :

- `frame`, περιέχει πληροφορίες για ένα πλαίσιο που περιέχεται στο παράθυρο.
- `frames`, είναι ένα πίνακας (`array`) που περιέχει μια λίστα όλων των αντικειμένων πλαισίου σ' ένα παράθυρο.
- `location`, είναι ένα αντικείμενο που περιέχει πληροφορίες για το URL του εγγράφου που εμφανίζεται στο παράθυρο.
- `history`, είναι ένα αντικείμενο που περιέχει πληροφορίες για τη λίστα ιστορικού του τρέχοντος παραθύρου ή πλαισίου.
- `document`, είναι ένα αντικείμενο που περιέχει πληροφορίες για το έγγραφο που εμφανίζεται στο παράθυρο.

Οι μέθοδοι του αντικειμένου `window` είναι οι εξής:

- `alert()`, εμφανίζει ένα πλαίσιο διαλόγου `Alert`.

- `clearTimeout()`, σταματάει τη λειτουργία της μεθόδου `setTimeout()`.
- `close()`, κλείνει το παράθυρο.
- `confirm()`, εμφανίζει ένα πλαίσιο διαλόγου επιβεβαίωσης `Confirm`.
- `blur()`, απομακρύνει την εστίαση από ένα παράθυρο.
- `focus()`, εστιάζει σ' ένα παράθυρο.
- `open()`, ανοίγει ένα νέο παράθυρο.
- `prompt()`, εμφανίζει ένα πλαίσιο διαλόγου προτροπής.
- `setTimeout()`, περιμένει για έναν προκαθορισμένο αριθμό χιλιοστών του δευτερολέπτου πριν εκτελέσει κάποιες εντολές.

Αντικείμενο Document

Οι ιδιότητες του αντικειμένου document είναι οι εξής :

- `alinkColor`, το χρώμα ενός ενεργού συνδέσμου.
- `anchor`, ένας δεσμός του εγγράφου.
- `anchors`, ένας πίνακας που περιέχει όλους τους δεσμούς του εγγράφου.
- `bgColor`, το χρώμα του παρασκηνίου του εγγράφου.
- `cookie`, ένα κομάτι κειμένου με πληροφορίες για τη σελίδα.
- `fgColor`, το χρώμα του κειμένου του εγγράφου.
- `form`, μια φόρμα του εγγράφου.
- `forms`, ένας πίνακας που περιέχει όλες τις φόρμες του εγγράφου.
- `lastModified`, η ημερομηνία τελευταίας τροποποίησης του εγγράφου.
- `link`, ένας σύνδεσμος του εγγράφου.
- `links`, ένας πίνακας που περιέχει όλους τους συνδέσμους του εγγράφου.
- `location`, το πλήρες URL του εγγράφου.

- `referrer`, το URL του εγγράφου που περιέχει έναν σύνδεσμο στον οποίο κάναμε κλικ για να έρθουμε στο τρέχον έγγραφο.
- `title`, το κείμενο που βρίσκεται ανάμεσα στις ετικέτες `<title>` και `</title>`.
- `vlinkColor`, το χρώμα ενός συνδέσμου που έχουμε επισκεφθεί.

Οι μέθοδοι του αντικειμένου `document` είναι οι εξής :

- `clear()`, καθαρίζει το περιεχόμενο ενός εγγράφου.
- `close()`, κλείνει τη ροή εγγράφου.
- `open()`, ανοίγει τη ροή εγγράφου.
- `write()`, γράφει ένα κείμενο στο έγγραφο.
- `writeln()`, γράφει ένα κείμενο στο έγγραφο και προσθέτει μια αλλαγή.

Παραδείγματα

Εφαρμογή 1: Απλό σενάριο σε JavaScript

Δημιουργήστε ένα αρχείο (`Script.html`), στο Notepad και πληκτρολογήστε τον ακόλουθο κώδικα:

```
<html>

<head>

<title>JavaScript Example</title>

<script type="text/javascript">

document.write("Welcome to the world of JavaScript");

</script>

</head>

<body>

</body>
```

</html>

- Κοιτάξτε τα αποτελέσματα του σεναρίου στο φυλλομετρητή σας.
- Αντιγράψτε το σενάριο, δηλαδή τις γραμμές <script type="text/javascript"> ...</script>, μέσα στο <body> της σελίδας HTML και σβήστε το από το <head>. Αποθηκεύστε τις αλλαγές σας και επιλέξτε "Refresh" στο φυλλομετρητή.
- Σβήστε το χαρακτηριστικό (attribute) type, που προσδιορίζει τη γλώσσα του σεναρίου, δηλαδή τη javascript.
- Αλλάξτε το όρισμα της εντολής document.write (π.χ. δώστε ως όρισμα το "<h1>Welcome
to the world
of JavaScript</h1>") και παρατηρήστε τις αλλαγές στη σελίδα.

Εφαρμογή 2: Σενάριο που υπολογίζει το άθροισμα 2 ακεραίων και το εκτυπώνει στη σελίδα.

Δημιουργήστε ένα αρχείο (Calculations.html) και πληκτρολογήστε τον παρακάτω κώδικα:

```
<html>
```

```
<head><title>Second JavaScript Example </title></head>
```

```
<body>
```

```
<script>
```

```
var firstNumber, secondNumber, Number1, Number2, sum;
```

```
firstNumber = window.prompt("Enter the first number", "0");
```

```
secondNumber = window.prompt("Enter the second Number", "0");
```

```
Number1 = parseInt(firstNumber);
```

```
Number2 = parseInt(secondNumber);

sum = Number1 + Number2;

document.writeln("<h1>The sum is: " + sum + "</h1>");

</script>
```

Click Refresh (or Reload) to run the script again

```
</body>

</html>
```

Αλλάξτε το παραπάνω σενάριο, ώστε να υπολογίζει το γινόμενο και τη διαφορά 2 ακεραίων.

Εφαρμογή 3: Σενάριο που δέχεται στοιχεία από μία φόρμα, τα ελέγχει και τα εμφανίζει σε ένα παράθυρο μηνύματος.

Δημιουργήστε ένα αρχείο (Form.html) και πληκτρολογήστε τον ακόλουθο κώδικα:

```
<html>

<head>

<title>Second JavaScript Example </title>

<script>

function CheckPassword(user, pass1,pass2)

{

if (pass1== pass2) {

alert("Successful Registration! \n Username:" + user + "\n Password:" + pass1) }

else {

alert("Retype your password") } }

</script>

</head>
```

```
<body>

<form>

Username: <input type="text" name= "username" /><br/><br/>

Password: <input type="password" name= "password1"/><br/><br/>

Confirm Password: <input type="password" name= "password2"/><br/><br/>

<input type= "button" value= "Register"

onClick="CheckPassword(form.username.value,
form.password1.value,form.password2.value)" /> <input type="reset" value="Clear"
/> </form> </body> </html>.
```

Δείτε τα αποτελέσματα του σεναρίου στον υπολογιστή σας

Βιβλιοθήκες της Javascript- JQuery

Η jQuery είναι μια καταπληκτική "βιβλιοθήκη" Javascript η οποία σας δίνει τη δυνατότητα να δημιουργείτε διάφορα εφέ στη σελίδα σας, χωρίς όμως να χρειάζεται η ποσότητα κώδικα που θα χρησιμοποιούσατε στην παραδοσιακή Javascript.

Με την jQuery μπορείτε να κάνετε πάρα πολλά πράγματα στις σελίδες σας όπως επιλογή και διαμόρφωση HTML στοιχείων, διαμόρφωση CSS στοιχείων, διεργασίες HTML γεγονότων, εφέ JavaScript και animations, διαμόρφωση του HTML DOM [Document Object Module], χρήση AJAX αλλά και πληθώρα άλλων εφαρμογών.

Αρχικά κατεβάστε τη βιβλιοθήκη της jQuery από την επίσημη ιστοσελίδα [www.jquery.com]. Εκεί έχετε τη δυνατότητα να επιλέξετε ανάμεσα σε 2 εκδόσεις: τη minified, που είναι πιο μικρή σε μέγεθος, και την development, που είναι πλήρης σε σχόλια, για όσους επιθυμούν να επέμβουν στον κώδικα.

Στη συνέχεια, θα πρέπει να κάνετε διαθέσιμη τη βιβλιοθήκη στην ιστοσελίδα σας, ώστε να μπορέσετε να χρησιμοποιήσετε τις εντολές της. Αυτό γίνεται συμπεριλαμβάνοντάς την στα αρχεία της ιστοσελίδας σας και γράφοντας στο

<head> κομμάτι της <script type="text/javascript" src="js/jquery.min.js"></script> όπου το αρχείο της jQuery βρίσκεται μέσα στο φάκελο με το όνομα js.

Έτσι, πλέον έχετε πλήρη πρόσβαση σε όλες τις εντολές της βιβλιοθήκης. Υπάρχει άλλος ένας τρόπος που μπορείτε να συμπεριλάβετε τη βιβλιοθήκη στην ιστοσελίδα σας, μέσω του Code Repository της Google.

Το μόνο που έχετε να κάνετε είναι να γράψετε στο <head> της σελίδας σας τον κώδικας είναι καθαρότερος <script type="text/javascript" src="http://ajax.googleapis.com/ajax/libs/jquery/1.6.1/jquery.min.js"></script> που θα φορτώσει την τελευταία έκδοση της jQuery κατευθείαν από τους servers της Google.

Το μόνο αρνητικό σε αυτήν την περίπτωση είναι ότι χρησιμοποιείτε εξωτερική πηγή για τα αρχεία σας και αυτό μπορεί να επηρεάσει την ταχύτητα της ιστοσελίδας σας. Σύμφωνα με τη λειτουργία της jQuery, και γενικότερα της Javascript, πρώτα επιλέγετε διάφορα στοιχεία της σελίδας και μετά τα αξιοποιείτε μέσω εντολών.

Γενικότερα, με την Javascript μπορείτε να τροποποιήσετε διάφορα στοιχεία που βρίσκονται στο DOM [Document Object Model], δηλαδή στη σελίδα σας.

Αφού φορτώσετε την jQuery στη σελίδα σας, γράψτε μέσα στο <head> της σελίδας σας, και κάτω από τον κώδικα με τον οποίο φορτώσατε την jQuery, το εξής:

```
<script type="text/javascript">
```

```
$(document).ready(function(){});</script>
```

Στη συνέχεια γράψτε την πρώτη σας επιλογή: <script type="text/javascript">

```
$(document).ready(function(){ $('p').css('color','red');}); </script>
```

Το σύμβολο του δολαρίου [\$] είναι ο επιλογές της jQuery.

Έτσι, στον παραπάνω κώδικα, αφού έχει φορτώσει πλήρως η σελίδα, επιλέξετε με επιτυχία όλες τις παραγράφους μέσα στη σελίδα σας και δώστε τους κόκκινο χρώμα, με την εντολή .css(), που προσθέτει στο επιλεγμένο στοιχείο συγκεκριμένες inline ιδιότητες CSS.

Η δομή μιας jQuery επιλογής, όπως βλέπετε από τον παραπάνω κώδικα, είναι \$('στοιχείο που θέλετε να επιλέξετε').εντολή('παράμετρος 1', 'παράμετρος 2'); όπου μέσα στον επιλογές κάνουμε κανονικά επιλογές όπως και στα CSS.

Δηλαδή, αν θέλετε να επιλέξετε κλάσεις, χρησιμοποιείτε τελεία [.] και μετά το όνομα της κλάσης, ενώ για τα ID τη δίεση [#] και αμέσως μετά το όνομα του ID.

Με την τελεία δηλώνετε ότι θέλετε να χρησιμοποιήσετε κάποια εντολή, όπως στο παραπάνω παράδειγμα την εντολή .css(), μαζί με τις παραμέτρους που παίρνει καθεμία από αυτές.

```
<script type="text/javascript"> $(document).ready(function(){$("p").hide() ή  
$("p.test").hide() όπου // Κρύβει όλες τις παραγράφους με class="test" ή  
$("#test").hide() όπου // Κρύβει όλα τα στοιχεία με id="test" });
```

Πλεονεκτήματα της JQuery

Κάποια από τα πλεονεκτήματα της jQuery είναι ότι ακολουθεί την αρχή KISS [Keep It Simple Stupid]. Η βιβλιοθήκη jQuery προσπαθεί να υπεραπλουστεύσει τον προγραμματισμό σε Javascript, με πραγματικά εύκολους μηχανισμούς μέσω του framework της.

Επίσης, παρέχει πλήρη και αναλυτικότερη τεκμηρίωση, που συμπληρώνεται από την εκτεταμένη παρουσία ηλεκτρονικών βοηθημάτων [η παρούσα σειρά άρθρων αποτελεί ένα από αυτά].

Εκτός από την πολύ καλοδουλεμένη τεκμηρίωσή της jQuery, οι ενδιαφερόμενοι μπορούν να ανατρέξουν και στη σελίδα της Visual jQuery [www.visualjquery.com], όπου θα βρουν μια εναλλακτική αλλά πολύ βολική, από άποψη δομής, τεκμηρίωση.

Όπως για τα περισσότερα open source έργα λογισμικού, έτσι και για τη jQuery η ύπαρξη μιας κατά το μέγιστο δυνατή ενεργής κοινότητας αποτελεί τον ακρογωνιαίο λίθο για την ανάπτυξη και ευημερία του.

Εξερευνήστε το κεντρικό site της jQuery και θα ανακαλύψετε ότι υπάρχουν πολλοί τρόποι για να έρθετε σε επαφή με την κοινότητα.

Το γεγονός ότι το βασικό πακέτο της jQuery είναι μόλις 20KB, αφενός επιβεβαιώνει την πρώτη παρατήρηση, ότι δηλαδή η φιλοσοφία της έγκειται στην απλότητα, και αφετέρου κάνει πολύ εύκολη την κατανόηση της αρχιτεκτονικής της.

Τέλος, η jQuery επιτρέπει στο χρήστη να χρησιμοποιήσει σχεδόν το σύνολο των δυνατοτήτων που προσφέρει η γλώσσα JavaScript: από απλά χαρακτηριστικά που σχετίζονται με βασικές λειτουργίες εμφάνισης/απόκρυψης ως Ajax κλήσεις και σύνθετα εφέ.

Η JQuery UI

Η jQuery από μόνη της παρέχει πληθώρα εργαλείων για να χτίσετε τη σελίδα σας όπως ακριβώς τη φανταζόσασταν. Παρ' όλα αυτά, υπάρχει και μια βιβλιοθήκη-επέκταση με το όνομα jQuery UI, η οποία ουσιαστικά προσθέτει νέες λειτουργίες, animations και εφέ στην ήδη υπάρχουσα βιβλιοθήκη της jQuery.

Δημιουργήσαμε μια φόρμα και τοποθετήσαμε μέσα της ένα πλήκτρο εντολής (button) με όνομα WindowButton και με μια ετικέτα (τίτλο) που προτρέπει τον χρήστη να κάνει κλικ.

Οι επιλογείς (selectors)

Η jquery (όπως και η javascript) για να επεξεργαστεί ένα ή περισσότερα στοιχεία (elements) ή ετικέτες (tags) ή αντικείμενα (objects), πρέπει πρώτα να τα επιλέξει και αν δεν υπάρχουν να τα δημιουργήσει.

Για την επιλογή μιας ή περισσότερων ετικετών η jquery χρησιμοποιεί τη συνάρτηση `$()` η οποία είναι μια σύντμηση της `jquery()` και τους επιλογείς (selectors) των css:

```
$ ('p') // επιλέγει όλες τις ετικέτες (p)
```

```
$ (' p#main') // επιλέγει την ετικέτα (p) με id= main
```

```
$ (# main') // επιλέγει την ετικέτα με id = main
```

```
$ ( '.main') // επιλέγει τις ετικέτες με classe = main
```

```
$ ('*') // επιλέγει όλες τις ετικέτες
```

`$ ([href]'') // επιλέγει όλες τις ετικέτες που έχουν την ιδιότητα href`

`$ ('a [target= '_blank ']) // επιλέγει τις ετικέτες a με target = '_ blank`

`$ ('p ' , ' div') // επιλέγει όλες τις ετικέτες p και div`

`$ (' div' 'p ') //επιλέγει όλες τις ετικέτες p που βρίσκονται μέσα σε ετικέτα div`

Εκτεταμένοι επιλεγείς

Οι εκτεταμένοι επιλεγείς λειτουργούν όπως και οι κανονικοί επιλεγείς αλλά τους χρησιμοποιούμε περισσότερο όταν έχουμε ήδη επιλέξει μια συλλογή ετικετών και θέλουμε μέσα από αυτή να κάνουμε μια πιο ειδική επιλογή.

Οι εκτεταμένοι επιλεγείς αποτελούν μια επέκταση των επιλογών τύπου `css`.

Κάποιοι από αυτούς είναι:

`$ (" first") // επιλέγει τη πρώτη ετικέτα από το έγγραφο ή μέσα από μια συλλογή`

`$ (" last ") // επιλέγει τη πρώτη ετικέτα από το έγγραφο ή μέσα από συλλογή`

`$ (" odd ") // επιλέγει τις μονές ετικέτες`

`$ (" : even") // επιλέγει τις ζυγές ετικέτες`

`$ (" : lt (index) *) // επιλέγει όλες τις ετικέτες που βρίσκονται σε θέση μικρότερη του index (π.χ $ (" : lt 5)`

`$ (" :gt (index)*) // επιλέγει όλες τις ετικέτες που βρίσκονται σε θέση μεγαλύτερη του index (π.χ $ (" :gt 5)`

`$ (" :eq (index)*) // επιλέγει την ετικέτα που βρίσκεται στη θέση index (π.χ $ (" : eq(5)")]`

`$ (" : contains (' text' *) // επιλέγει όλες τις ετικέτες που περιέχουν το κείμενο text.`

Το αντικείμενο `$` :

- Global μεταβλητή με όνομα `$`
- Θα μπορούσε να είναι οτιδήποτε, δε σημαίνει κάτι το `$`

- Μέσω αυτής έχουμε πρόσβαση στην jQuery

Ορίζεται μόλις φορτώσει και τρέξει το jquery.js

Φίλτρα (filters)

Όταν κάνετε μια επιλογή όπως για παράδειγμα: `$('#p')` η επιλογή αυτή επιστέφει μια σειρά (συλλογή) από ετικέτες p.

Αν θέλετε να φιλτράρετε τη συλλογή ή να "ξεσκαρτάρετε" κάποιες από αυτές τότε χρησιμοποιείτε τα φίλτρα.

Για να φιλτράρετε μια συλλογή μπορείτε να χρησιμοποιήσετε τους: εκτεταμένους επιλογείς τις συναρτήσεις `filter()` και `not()`.

Φίλτρα με εκτεταμένους επιλογείς

```
$('#p:even');
```

Δηλαδή, μετά τον επιλογέα p "κολλάτε" έναν εκτεταμένο επιλογέα όπως `:even`

Φίλτρα με τη συνάρτηση filter()

Για να φιλτράρετε μία συλλογή που έχετε επιλέξει έστω με τον επιλογέα `$('#p')` γράφετε: Δηλαδή, μετά τον επιλογέα `$('#p')` "κολλάτε" τη συνάρτηση `filter(":even")` η οποία έχει ως παράμετρο έναν εκτεταμένο επιλογέα (μπορεί όμως η παράμετρος να είναι οποιοσδήποτε επιλογέας).

Φίλτρα με τη συνάρτηση not()

Για να φιλτράρετε μία συλλογή που έχετε επιλέξει έστω με τον επιλογέα `$('#p')` γράφετε

Δηλαδή, μετά τον επιλογέα `$('#p')` "κολλάτε" τη συνάρτηση `not(":even")` η οποία έχει ως παράμετρο έναν εκτεταμένο επιλογέα (μπορεί όμως η παράμετρος να είναι οποιοσδήποτε επιλογέας). Η `not()`, σε αντίθεση με την `filter()`, αποκλείει (ξεσκαρτάρει) μια σειρά από ετικέτες.

```

<table width = "0" border = "1" cellpadding = "4">
  <tr><th>A/A</th><th>user</th><th>message</th></tr>
  <tr><td>1</td><td>micky</td><td>good morning</td></tr>
  <tr><td>2</td><td>pedro</td><td>buenos dias</td></tr>
  <tr><td>3</td><td>mini</td><td>bon jorno</td></tr>
</table>
<script>$ ("tr:odd").css ('background-color', "pink");</script>

```

Διαχείριση ιδιοτήτων

Αφού επιλεγεί μία ή περισσότερες ετικέτες εφαρμόζουμε τις εντολές που θέλουμε. Οι εντολές αυτές έχουν σχέση με τη διαχείριση και επεξεργασία που αριθμήσαμε στην αρχή και επεξηγούμε στη συνέχεια.

1 Ανάγνωση (read) τιμής ιδιότητας

```
$("p#good").prop("align"); //επιστρέφει τη τιμή της ιδιότητας align
```

```
$("p#good").attr("align"); // επιστρέφει τη τιμή της ιδιότητας align
```

Για στοιχεία φόρμας τύπου input όπως `<input type = 'text' />`, σε select και textarea μπορείτε να χρησιμοποιήσετε και τη συνάρτηση `val()`.

Το παράδειγμα φαίνεται `$("input#name").val();` //επιστρέφει την τιμή (value) της ετικέτας input με `id="name"`.

Για στοιχεία φόρμας τύπου input όπως `<input type = 'text' />`, σε select και textarea μπορείτε να χρησιμοποιήσετε και τη συνάρτηση `val()`.

```
$("input#name").val("Πηνελόπη");
```

θέτει `value= "Πηνελόπη"`

2 Ανάγνωση (read) τιμής ιδιότητας css

Χρησιμοποιούμε τη συνάρτηση `css()` για να διαβάσουμε την τιμή μιας ιδιότητας css.

Αυτή η ιδιότητα είναι ιδιαίτερα χρήσιμη όταν δημιουργούμε πολύπλοκα σενάρια και τα χρησιμοποιούμε σε πολλές ιστοσελίδες.

3. Γραφή (write) τιμής ιδιότητας css

Χρησιμοποιούμε πάλι τη συνάρτηση `css()` για να γράψουμε (αλλάξουμε) την τιμή μιας ιδιότητας.

```
$("#div#good").css("color","yellow");//θέτει color;yellow
```

Για περισσότερες ιδιότητες μπορείτε να χρησιμοποιήσετε ένα αντικείμενο json στο οποίο θα προσθέσετε όλες τις ιδιότητες με τις αντίστοιχες τιμές.

```
$("#div#good").css({backgroundColor:#ffe',borderLeft:5px solid#ccc'});
```

5. Διαγραφή ιδιότητας css

Εάν γράψετε όπως φαίνεται παραπάνω, αφαιρείται η τιμή της ιδιότητας `color` από την ετικέτα.

```
$("#div#good").css("color","");
```

6. Διαγραφή ιδιότητας html

Για τη διαγραφή ιδιότητας html από ετικέτα χρησιμοποιείτε τις:

- `removeProp()`
- `removeAttr()`

```
var $para=$("#p");
```

```
$para.removeProp("align");
```

Διαχείριση περιεχομένου ετικέτας

Για την ανάγνωση περιεχομένου ετικέτας μπορείτε να χρησιμοποιήσετε τις παρακάτω συναρτήσεις:

- `text()`
- `html()`

Συμβάντα (events)

Η jQuery προσφέρει μια πληθώρα από συναρτήσεις με τις οποίες μπορείτε να προσθέσετε ή να ορίσετε διαχειριστές συμβάντων (event handlers).

Όλες οι συναρτήσεις για τα συμβάντα είναι συντμήσεις της γενικής συνάρτησης `bind()` η οποία συντάσσεται ως εξής όπως φαίνεται στην διαφάνεια. Παίρνει δύο παραμέτρους. Η πρώτη είναι το όνομα του συμβάντος και η δεύτερη μια ανώνυμη συνάρτηση που θα κληθεί και θα εκτελεστεί όταν προκύψει το συμβάν.

```
$('#p').bind(eventname, function() {...})
```

Animation και εφέ

Για animation και εφέ μπορείτε να χρησιμοποιήσετε τη γενική εντολή, η οποία είναι: **`animate(properties [,duration] [,easing] [,complete])`**.

Η πρώτη παράμετρος είναι ένα αντικείμενο json με μία ή περισσότερες ιδιότητες που πρόκειται να αλλάξουν στις αντίστοιχες τιμές. Οι ιδιότητες πρέπει να δέχονται αριθμητικές τιμές όπως `width`, `left` κ.λπ.

Η δεύτερη παράμετρος ορίζει τη διάρκεια του εφέ σε χιλιοστά του δευτερολέπτου (3000).

Η τρίτη παράμετρος ορίζει το είδος του `easing`.

Η τέταρτη παράμετρος είναι μια ανώνυμη συνάρτηση που καλείται όταν το εφέ τελειώσει.

Από τις παραπάνω 4 παραμέτρους μόνο η πρώτη είναι υποχρεωτική.

Για πιο ειδικά εφέ μπορείτε να χρησιμοποιήσετε τις παρακάτω συναρτήσεις:

Για hide/show

hide()

show()

toggle()

Για fadeIn/fadeout

fadeIn()

fadeOut()

fadeTo()

fadeToggle()

Παράδειγμα με fadeToggle φαίνεται παρακάτω:

```
<img src= "/parrot.jpg" id="parrot"/>
```

```
<br/>
```

```
<button id = 'fadeToggleButton'>fadeToggle image</button>
```

```
<script>
```

```
$('#fadeToggleButton').click(function(){
```

```
    $('#parrot').fadeToggle(600);
```

Η τιμή 600 που έχει περαστεί στην fadeToggle(600) είναι προαιρετική και ορίζει τη διάρκεια του εφέ.

Εκπαιδευτική: Ενότητα 6.4 Ανάπτυξη δυναμικών ιστοσελίδων (PHP, MySQL)

Ο συνδυασμός της τεχνολογίας ή γλώσσας προγραμματισμού PHP (Hypertext PreProcessor) με τη βάση δεδομένων MySQL αποτελεί ένα από τα ισχυρότερα εργαλεία ανάπτυξης δυναμικών εφαρμογών στον παγκόσμιο ιστό (Web).

Με τη χρήση τους μπορούμε να δημιουργήσουμε από απλές εφαρμογές που να περιέχουν μια φόρμα παραγγελίας έως και ασφαλείς τοποθεσίες ηλεκτρονικού εμπορίου.

Όπως ήδη γνωρίζουμε, μια ιστοσελίδα που είναι γραμμένη σε απλή HTML είναι κατά βάση στατική, δηλ. τα περιεχόμενά της δεν αλλάζουν αυτόματα και δεν προσαρμόζονται στις απαιτήσεις του χρήστη παρά μόνο αν αποφασίσει να τα ενημερώνει τακτικά ο δημιουργός της ιστοσελίδας, πράγμα όχι πάντα εφικτό.

Ο συνδυασμός, όμως, της γλώσσας προγραμματισμού PHP και της βάσης δεδομένων MySQL, μπορεί να κάνει τις ιστοσελίδες μας να είναι δυναμικές, δηλ. να ενημερώνονται αυτόματα και να προσαρμόζονται στις απαιτήσεις του χρήστη, χωρίς να χρειασθεί να κάνει καμία απολύτως ενημέρωση ο δημιουργός των ιστοσελίδων.

Οι πιο συνηθισμένες εφαρμογές που μπορούμε να επιτύχουμε με τη χρήση των παραπάνω τεχνολογιών είναι οι εξής :

- Έλεγχος ταυτότητας χρήστη.
- Καλάθι αγορών (shopping cart).
- Σύστημα Διαχείρισης Περιεχομένου (CMS, Content Management System).
- Web mail.
- Web forum.
- GuestBook.

Λέξεις κλειδιά : PHP, MySQL, server-side scripting

Τι είναι η PHP;

Όπως αναφέρθηκε παραπάνω, είναι μια γλώσσα δέσμης ενεργειών που δημιουργήθηκε για επικοινωνίες από την πλευρά του διακομιστή. Κατά συνέπεια, είναι σε θέση να χειρίζεται διάφορες λειτουργίες υποστήριξης, όπως συλλογή φορμών δεδομένων, διαχείριση αρχείων διακομιστή, τροποποίηση βάσεων

δεδομένων και πολλά άλλα. Αν και η PHP θεωρείται μια γλώσσα δέσμης ενεργειών γενικής χρήσης, χρησιμοποιείται κυρίως για την ανάπτυξη ιστού . Αυτό οφείλεται σε ένα από τα πιο αξιοσημείωτα χαρακτηριστικά του: την ικανότητα να ενσωματώνεται σε ένα αρχείο HTML. Ας υποθέσουμε ότι δεν θέλετε οι άνθρωποι να βλέπουν τον πηγαίο κώδικα σας. Για να το λύσετε, απλώς αποκρύψτε το με αυτήν τη γλώσσα σεναρίου. Απλώς γράψτε τον κώδικα σε ένα αρχείο PHP, ενσωματώστε τον στο HTML και τότε οι άνθρωποι δεν θα μάθουν ποτέ ποιο είναι το αρχικό περιεχόμενο. Ένα άλλο πλεονέκτημα αυτής της συγκεκριμένης δυνατότητας είναι όταν πρέπει να χρησιμοποιείτε επανειλημμένα τη σήμανση HTML. Αντί να τα ξαναγράφετε ξανά και ξανά, απλώς εφαρμόστε τον κώδικα σε ένα αρχείο PHP. Όταν χρειάζεται να χρησιμοποιήσετε HTML, εισάγετε το αρχείο PHP και είστε έτοιμοι. Με άλλα λόγια, αυτή η σελίδα δημιουργείται με ένα αρχείο php (home.php) που περιέχει ένα μείγμα κώδικα προεπεξεργαστή υπερκειμένου και ετικέτες HTML.

Εάν θέλετε να μάθετε περισσότερα για το πώς γράφεται ένα αρχείο PHP, μπορείτε να δείτε ένα παράδειγμα παρακάτω:

<code>< <u>html</u> ></code>
<code>< κεφάλι ></code>
<code>< τίτλος > Δοκιμή PHP < <u>title</u> ></code>
<code>< /κεφάλι ></code>
<code>< σώμα ></code>
<code>< <u>php</u> echo '<p>Χαιρετισμοί από τον <u>Hostinger</u></p>' ; ? ></code>
<code>< <u>body</u> ></code>
<code>< <u>html</u> ></code>

Εάν τοποθετήσετε αυτό το γραπτό αρχείο PHP στον ριζικό κατάλογο του ιστότοπού σας, θα το βρείτε μεταβαίνοντας στη διεύθυνση yoursite.com/greetings.php.

Η έξοδος αυτού του αρχείου PHP σε HTML θα μοιάζει με αυτό:

< <u>html</u> >
< κεφάλι >
< τίτλος > Δοκιμή PHP < / <u>title</u> >
< /κεφάλι >
< σώμα >
< p > Χαιρετισμούς από τον <u>Hostinger</u> < /p >
< / <u>body</u> >
< / <u>html</u> >

Αυτό είναι το πιο βασικό παράδειγμα, καθώς δεν χρειάζεται καν να εκτελέσετε αυτό το σενάριο ούτως ή άλλως ή σε οποιαδήποτε μορφή. Απλώς χρησιμοποιούμε την εντολή echo για να πούμε στη σελίδα να εμφανίζει Χαιρετισμούς από το Hostinger . Αυτό το παράδειγμα, ωστόσο, δείχνει πώς ο ιστότοπός σας μπορεί να ερμηνεύσει τον κώδικα PHP εντός HTML.

Με λίγα λόγια, η PHP είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται από προγραμματιστές και προγραμματιστές για τη δημιουργία δυναμικών ιστοσελίδων, επεκτάσεων ενοποίησης εφαρμογών και τον εξορθολογισμό της ανάπτυξης ενός συστήματος. Αυτή η γλώσσα είναι παγκοσμίως γνωστή και μια από τις πιο χρησιμοποιούμενες για την ευκολία εκμάθησης και χειρισμού της, εκτός από το ότι είναι συμβατή με σχεδόν όλα τα λειτουργικά συστήματα που υπάρχουν – γεγονός που μειώνει το κόστος της. Σε αυτό το άρθρο θα δείξουμε τη σημασία αυτού του αρκτικόλεξου, εκτός από τη δομή και τα πλεονεκτήματα αυτής της γλώσσας.

PHP: Hypertext Preprocessor , που αρχικά ονομαζόταν Personal Home Page.

Είναι επίσης ένα υποσύνολο γλωσσών δέσμης ενεργειών όπως η JavaScript και η Python . Η διαφορά είναι ότι η PHP χρησιμοποιείται κυρίως για επικοινωνία από την πλευρά του διακομιστή (backend).

Πότε ξεκίνησε και πώς λειτουργεί

Σήμερα, η PHP είναι μια από τις πιο χρησιμοποιούμενες γλώσσες προγραμματισμού στον κόσμο. Ο όρος PHP δημιουργήθηκε με μόνο ένα σωρό κώδικα CGI – ένα στοιχείο που κάνει τη φυσική ή λογική σύνδεση μεταξύ δύο συστημάτων ή διακομιστών, που περιγράφεται σε μια γλώσσα C. Η αρχική ιδέα ήταν η παρακολούθηση της επισκεψιμότητας της προσωπικής ιστοσελίδας του δημιουργού. Τα χρόνια πέρασαν και ο δημιουργός ανέπτυξε σενάρια, τα οποία πρόσθεσαν τις δυνατότητες που είχε ο ιστότοπός του.

Η επιτυχία αυτής της γλώσσας ήταν τόσο μεγάλη που ο δημιουργός, Rasmus Lerdorf, μετέτρεψε το σύμπλεγμα κώδικα CGI σε γλώσσα προγραμματισμού. Ως αποτέλεσμα, η συντριπτική πλειοψηφία των ιστοσελίδων και των εφαρμογών άρχισε να χρησιμοποιεί την PHP ως κύρια γλώσσα

Πλευρά διακομιστή (backend) και πλευρά πελάτη (frontend).

Οι γλώσσες δέσμης ενεργειών μπορούν να εκτελούνται τόσο από την πλευρά του διακομιστή (backend) όσο και από την πλευρά του πελάτη (frontend).

Τα σενάρια Frontend επεξεργάζονται τα προγράμματα περιήγησης. Όταν το πρόγραμμα περιήγησής σας - γνωστό και ως πελάτης - ζητά μια σελίδα που περιέχει σενάρια από την πλευρά του πελάτη, ο διακομιστής αποκρίνεται στέλνοντας τον πηγαίο κώδικα που είναι εκτελέσιμος από το πρόγραμμα περιήγησης.

Από την άλλη πλευρά, η γλώσσα δέσμης ενεργειών υποστήριξης σημαίνει ότι αυτά τα σενάρια εκτελούνται στους διακομιστές πριν σταλούν στο πρόγραμμα περιήγησης.

Έτσι, αντί να στείλουν τον πηγαίο κώδικα, οι διακομιστές ιστού επεξεργάζονται (αναλύουν) τον κώδικα πρώτα μετατρέποντάς τον σε καθαρή μορφή HTML.

Φαίνεται μπερδεμένο;

Αυτός είναι ο λόγος για τον οποίο πρέπει πρώτα να κατανοήσουμε τις γλώσσες δέσμης ενεργειών προτού βουτήξουμε βαθιά στην PHP.

Μια γλώσσα δέσμης ενεργειών χρησιμεύει για την αυτοματοποίηση της

Η PHP (PHP: Hypertext Preprocessor) είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που είτε θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML ή θα επεξεργασθεί τις εισόδους δίχως να προβάλλει την έξοδο στο χρήστη, αλλά θα τις μεταβιβάσει σε κάποιο άλλο PHP script.

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού C ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter.

Το 1997 η PHP/FI έφθασε στην έκδοση 2.0 αριθμώντας περισσότερους από 50.000 ιστότοπους που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP.

Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 5, 6 και 7 της PHP.

Τα περισσότερα από τα διαδεδομένα συστήματα διαχείρισης περιεχομένου χρησιμοποιούν PHP όπως Media wiki, Drupal, Joomla, WordPress και ούτω καθεξής που επιτρέπουν τη δημιουργία ιστοσελίδων χωρίς πολλές δεξιότητες προγραμματισμού.

Το κύριο πλεονέκτημα της PHP είναι ότι είναι διαθέσιμο σε κάθε κοινόχρηστο πάροχο φιλοξενίας. Η PHP θεωρείται ως το πλέον εδραιωμένο περιβάλλον χρόνου εκτέλεσης στον διακομιστή προς το παρόν.

Θα μπορούσε να προσφέρει καλύτερη βαθμολογία μηχανών αναζήτησης και προσβασιμότητα στους παρόχους φιλοξενίας.



Βασικά σημεία της PHP

Η PHP έχει κάποια βασικά σημεία που πρέπει να αναφέρουμε:

- PHP αρχεία, μπορούν να περιέχουν text, HTML tags και scripts.
- Εκτός από την μηχανή εκτέλεσης της PHP θα πρέπει κανείς να έχει εγκατεστημένο έναν web server (π.χ. Apache) για να πειραματιστεί με την γλώσσα.
- Ο κώδικας πρέπει να περιέχεται σε PHP αρχεία (".php", ".php3", ή ".phpml").

Γιατί να χρησιμοποιήσουμε την PHP;

Οι λόγοι για τους οποίους πρέπει να χρησιμοποιήσουμε την PHP είναι οι παρακάτω:

- Υψηλή απόδοση: με ένα φτηνό server μπορούν να εξυπηρετηθούν εκατομύρια επισκέψεων καθημερινά.
- Συνεργάζεται εύκολα με τους περισσότερους database servers (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, κ.α.).
- Είναι σημαντική για το χτίσιμο πληροφοριακών συστημάτων (π.χ εφαρμογές ηλεκτρονικού εμπορίου).
- Έχει ενσωματωμένες βιβλιοθήκες για συνήθεις web διαδικασίες: δυναμική δημιουργία, εικόνων, αποστολή email, χειρισμός cookies.
- Είναι δωρεάν
- Είναι απλή στην κατανόηση και στη χρήση της. Η σύνταξή της βασίζεται σε άλλες γλώσσες (μοιάζει με java, C).
- Μεταφερισιμότητα (portability): ο ίδιος κώδικας δουλεύει χωρίς αλλαγές και σε άλλο λειτουργικό σύστημα (Windows - Linux).
- Διαθεσιμότητα του κώδικα προέλευσης (open source): μπορούν να πραγματοποιηθούν αλλαγές στη γλώσσα.

Παράδειγμα PHP

```
<html>
<head>
<title> Βασική Σελίδα </title>
</head>
<body>
<?
echo "ΓΕΙΑ ΣΑΣ ΑΠΟ ΤΗΝ PHP"; ?>
</body>
```

</html>

Όπως μπορούμε να καταλάβουμε με μία πρώτη ανάγνωση του κώδικα, η PHP περικλείεται μέσα σε κώδικα HTML με την χρήση των TAGS <? ?>. Σκοπός της PHP είναι να δώσει ζωντάνια σε έναν στατικό κώδικα HTML, ώστε να μπορούμε μέσα από απλές σελίδες να εκτελούμε και κώδικα που να προσαρμόζεται στις απαιτήσεις του χρήστη. Οι σελίδες PHP επειδή εκτελούνται από τον server δεν τις ανοίγουμε ποτέ σαν απλά έγγραφα από τον φυλλομετρητή που χρησιμοποιούμε, αλλά πρέπει να συνδεθούμε μέσω του φυλλομετρητή στον απομακρυσμένο διακομιστή σελίδων και να ζητήσουμε την εκτέλεση της σελίδας.

Αν λοιπόν ο χρήστης user έχει στον φάκελο public_html στον κόμβο www.ucert.gr την σελίδα first.php για να την δούμε θα πρέπει να δώσουμε στο πεδίο διεύθυνσεως του φυλλομετρητή μας <http://www.ucert.gr/~user/first.php> Αν από την άλλη δεν έχουμε μόνιμη σύνδεση στο INTERNET (μόνιμο IP) και θέλουμε απλώς να δοκιμάσουμε μία σελίδα στο μηχάνημα που βρισκόμαστε, τότε πολύ απλά δίνουμε στο πεδίο διεύθυνσεως του φυλλομετρητή <http://localhost/~user/first.php>.

Προκαθορισμένες μεταβλητές στην PHP

Με τον όρο μεταβλητή, ορίζουμε την συμβολική ονομασία μιας τιμής, η οποία μπορεί ν' αλλάξει με το πέρασμα του χρόνου.

Η PHP, έχει κάποιες προκαθορισμένες μεταβλητές, που είναι οι παρακάτω:

- `$_SERVER`: Είναι οι μεταβλητές που ορίζονται από τον web server ή διαφορετικά είναι άμεσα συνδεδεμένες με το περιβάλλον εκτέλεσης του τρέχοντος script.
- `$_GET`: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP GET.
- `$_POST`: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP POST.

- `$_COOKIE`: Είναι οι μεταβλητές που παρέχονται στο script μέσω της HTTP cookies.
- `$_SESSION`: Είναι οι μεταβλητές που είναι προς το παρόν εγγεγραμμένες σε ένα session ενός script.
- `$_FILES`: Είναι οι μεταβλητές που παρέχονται στο script μέσω του HTTP post file uploads.
- `$_ENV`: Είναι οι μεταβλητές που παρέχονται στο script μέσω του environment.
- `$_REQUEST`: Είναι οι μεταβλητές που παρέχονται στο script μέσω του μηχανισμού εισαγωγής δεδομένων από το χρήστη.
- `$GLOBALS`: Περιέχουν μια αναφορά σε κάθε μεταβλητή που είναι διαθέσιμη μέσα στην global εμβέλεια του script. Τα κλειδιά αυτού του array είναι τα ονόματα των global μεταβλητών.

Η PHP, εγκαθίσταται σαν module πάνω σε έναν εξυπηρετητή διαδικτύου (web server) και διαχειρίζεται όλα τα αρχεία με επέκταση .php. τα οποία μπορεί:

- ✓ Να είναι κανονικά html αρχεία με ενσωματωμένα κομμάτια κώδικα PHP σε συγκεκριμένα σημεία
- ✓ Ο κώδικας σε PHP παράγει καινούρια κομμάτια HTML στα σημεία που εμφανίζεται,
- ✓ Ο φυλλομετρητής του χρήστη λαμβάνει ένα καθαρό HTML αρχείο χωρίς κώδικα PHP.

Τύποι δεδομένων

Τυπικά η γλώσσα έχει τους ακόλουθους τύπους δεδομένων:

1. Boolean
2. Integer

3. Double

4. String

Ωστόσο, δεν παρέχεται στον προγραμματιστή κάποιος άμεσος τρόπος ώστε να δηλώσει μία μεταβλητή με κάποιον από τους παραπάνω τύπους. Η ανάθεση σε μία μεταβλητή μίας τιμής κατευθύνει τον διερμηνευτή, ώστε να καταλάβει ποιος είναι ο απαιτούμενος τύπος.

Δημιουργεί τέσσερις μεταβλητές, την a με ακέραιο τύπο, την b με boolean τύπο, την c με τύπο πραγματικών αριθμών και τέλος την d με τύπο αλφαριθμητικού. Ωστόσο καμία μεταβλητή δεν δεσμεύεται στατικά με έναν τύπο παρά μόνον στην ανάθεση. Έτσι αν στην ίδια ενότητα κώδικα με την προηγούμενη ή σε κάποια άλλη υπάρχει η ανάθεση $d=-1.7$, τότε δεν θα προκληθεί κάποιο λάθος και απλά ο νέος τύπος για την μεταβλητή \$d είναι double.

Η PHP και η MySQL

Αν και η PHP υποστηρίζει πολλά συστήματα διαχείρισης βάσεων δεδομένων (DMBS, Database Management Systems), η πιο δημοφιλής βάση δεδομένων και αυτή που θα χρησιμοποιήσουμε είναι η MySQL. Αφού έχουμε σιγουρευτεί ότι ο Web server στον οποίο θα στήσουμε την εφαρμογή μας υποστηρίζει την PHP και την MySQL (ΠΣΔ), θα πρέπει να κάνουμε μια σύντομη και απλή δοκιμή πριν συνεχίσουμε.

Δημιουργούμε ένα αρχείο με το Σημειωματάριο (Notepad) των Windows, το οποίο αποθηκεύουμε με το όνομα test.php και με το εξής περιεχόμενο : `<?php
phpinfo();?>`. Το φορτώνουμε (upload) στον Web server και το καλούμε από τον φυλλομετρητή μας. Αν όλα δουλεύουν κανονικά, θα πρέπει να δούμε μια μεγάλη ιστοσελίδα μ' ένα πλήθος από τεχνικά χαρακτηριστικά για την εγκατεστημένη PHP.

Η MySQL είναι ένας SQL Database Server και είναι ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (RDBMS, Relational Database Management System). Ένα RDBMS ορίζει σχέσεις (relations) ανάμεσα σε δομές δεδομένων (data structures).

Τα αρχικά SQL σημαίνουν Structured Query Language, δηλ. Δομημένη Γλώσσα Ερωτημάτων ή Ερωταποκρίσεων. Η γλώσσα SQL μάς δίνει τη δυνατότητα να μιλάμε φιλικά με τον server ώστε να πάρουμε πληροφορίες από τη βάση δεδομένων και να μπορέσουμε να τις διαχειριστούμε.

Η MySQL είναι γρήγορη, πολύ αξιόπιστη και εύκολη στη χρήση. Επίσης, η MySQL αποτελεί την επιλογή της γλώσσας PHP και πολλές εταιρείες Web hosting παρέχουν την MySQL και την PHP ως ένα στάνταρτ πακέτο. Το μεγαλύτερο πλεονέκτημα της MySQL είναι ότι είναι δωρεάν (free) και ανοικτού κώδικα (open source). Επίσης, μπορεί να εκτελεσθεί σε πολλές πλατφόρμες, όπως Windows, Linux, FreeBSD, Solaris κ.ά.

Τι είναι η MySQL;

Η MySQL είναι μία γλώσσα προγραμματισμού για Συστήματα Διαχείρισης Βάσεων Δεδομένων (DBMS). Προκειμένου τα DBMS να μπορούν να έχουν κάποιους κοινούς άξονες αναφοράς έχει αναπτυχθεί η γλώσσα SQL (Structured Query Language) ή οποία επιτρέπει μέσα από ένα σύστημα εντολών, την προσπέλαση των δομημένων δεδομένων που περιέχει ένα DBMS. Κάθε DBMS ουσιαστικά δέχεται και εκτελεί ένα σετ εντολών SQL για να διαχειριστεί τα δεδομένα του. Η MySQL είναι μία διανομή (έκδοση) της γλώσσας SQL και ένα MySQL DBMS είναι ένα σύστημα που δέχεται εντολές της MySQL.

Η MySQL αποτελεί ελεύθερο λογισμικό που μπορεί να χρησιμοποιήσει κανείς ερασιτεχνικά ή επαγγελματικά. Αντίστοιχα με την MySQL υπάρχουν και άλλα συστήματα δωρεάν ή με πληρωμή. Από τα πιο γνωστά DBMS είναι ο SQL Server της Microsoft, ο Oracle SQL server, η DB2 κ.ά.

Καθώς η τεχνολογία των DBMS εξελίχθηκε στο χρόνο προέκυψαν διάφοροι διαχωρισμοί αυτών, ανάλογα με τον τύπο των δεδομένων που αποθηκεύονται σε αυτά. Οι οργανωμένες δομές δεδομένων στις οποίες έχουμε αναφερθεί μέχρι στιγμής, αποθηκεύουν σύνολα δεδομένων (πλειάδες) και τον τρόπο με τον οποίο αυτά συσχετίζονται.

Επειδή η λογική πίσω από τη διαχείριση αυτών των δεδομένων είναι η συσχέτιση (σχέση) των δεδομένων, αυτές οι βάσεις αποκαλούνται Σχεσιακές Βάσεις Δεδομένων και τα συστήματά τους σχεσιακά (RDBMS - Relational Database Managements Systems). Άλλοι τύποι DBMS είναι τα HDBMS (Hierarchical DBMS) όπου τα δεδομένα αποθηκεύονται σαν δέντρο, οπότε η ίδια η δομή της βάσης υποδηλώνει και τη συσχέτιση των δεδομένων της.

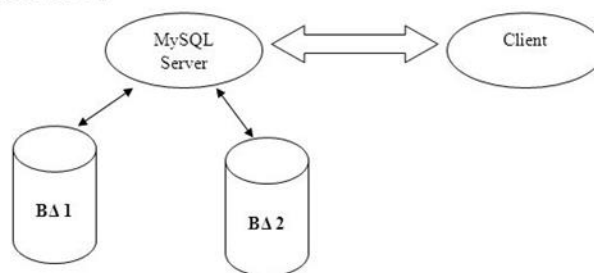
Βασικά χαρακτηριστικά της MySQL

Η MySQL λειτουργεί με το μοντέλο Πελάτη/Εξυπηρετητή (Client/Server). Ο εξυπηρετητής δέχεται αιτήματα από τους πελάτες σχετικά με τη διαχείριση μιας ή περισσότερων βάσεων δεδομένων και προβαίνει στις απαιτούμενες ενέργειες.

MySQL

■ MySQL

- μία από τις πιο δημοφιλείς ανοικτές εφαρμογές διαχείρισης βάσεων δεδομένων
- λειτουργεί με το μοντέλο πελάτη - εξυπηρετητή (client - server), όπου ο εξυπηρετητής διαχειρίζεται τις αιτήσεις των πελατών προς οποιαδήποτε βάση δεδομένων.



Οι κύριες λειτουργίες διαχείρισης βάσεων δεδομένων τις οποίες υλοποιεί ο εξυπηρετητής κατόπιν αιτήσεως από τον πελάτη, δε διαφέρουν σημαντικά από τις αντίστοιχες λειτουργίες των άλλων εφαρμογών βάσεων δεδομένων και περιλαμβάνουν :

- τη δημιουργία-διαγραφή βάσεων δεδομένων,

- την εισαγωγή-τροποποίηση-διαγραφή πινάκων (tables) και πεδίων (fields),
- την εισαγωγή-τροποποίηση-διαγραφή εγγραφών (records) και τέλος
- την ανάκτηση δεδομένων από τη βάση με τη χρήση συγκεκριμένων κριτηρίων.

Για την υλοποίηση των παραπάνω λειτουργιών, η MySQL χρησιμοποιεί την SQL (Δομημένη Γλώσσα Ερωτήσεων – Structured Query Language).

Η SQL είναι η ευρύτερα χρησιμοποιούμενη τυποποιημένη γλώσσα πρόσβασης στις βάσεις δεδομένων και αποτελείται από εντολές οι οποίες επιτρέπουν την ανάκτηση και ενημέρωση δεδομένων σε μια βάση. Σημειώνουμε ότι, εκτός από τη MySQL, η γλώσσα SQL συνεργάζεται με άλλα προγράμματα βάσεων δεδομένων όπως είναι η Access, Microsoft SQL Server, Oracle, Sybase και πολλά άλλα.

Τι είναι μια βάση MySQL?

Μια βάση δεδομένων σε MySQL είναι ένας χώρος που έχουμε διαμορφώσει κατάλληλα προκειμένου να αποθηκεύσουμε τα δεδομένα των ιστοσελίδων μας με σχεσιακό τρόπο.

Για παράδειγμα αν έχουμε στην ιστοσελίδα μας μία φόρμα παραγγελίας προϊόντων, τότε μπορούμε στη βάση μας να έχουμε τον πίνακα (πλειάδα) των χρηστών, όπου θα αποθηκεύουμε τους πελάτες μας και τον πίνακα (πλειάδα) των παραγγελιών, όπου θα αποθηκεύουμε τις παραγγελίες τους. Οι δύο αυτοί πίνακες θα σχετίζονται με σχέση 1 προς πολλά (1:N) επειδή ένα πελάτης μπορεί να κάνει πολλές παραγγελίες.

Πλεονεκτήματα της MySQL

Παρακάτω παρατίθενται μερικά από τα πλεονεκτήματα της MySQL που την καθιστούν ανταγωνιστική έναντι άλλων πακέτων διαχείρισης βάσεων δεδομένων.

Τα βασικά πλεονεκτήματα της είναι:

- Απόδοση: Η MySQL είναι αρκετά γρήγορη. Πολλές δοκιμές που έχουν γίνει δείχνουν ότι υπερέχει σε ταχύτητα έναντι των ανταγωνιστών της.

- **Κόστος:** Η MySQL είναι προϊόν ανοικτού κώδικα και διατίθεται δωρεάν για προσωπική χρήση. Η εμπορική άδεια της διατίθεται σε χαμηλό κόστος. Αυτό σημαίνει ότι αν κάποιος θέλει να τη χρησιμοποιήσει για εφαρμογές προσωπικής χρήσης ή για εφαρμογές που θα είναι ανοικτού κώδικα δεν χρειάζεται να αγοράσει κάποια άδεια. Άδεια απαιτείται αν κάποιος τη χρησιμοποιήσει για εμπορικές εφαρμογές που δεν θα είναι ανοικτού κώδικα.
- **Ευκολία Χρήσης:** Η MySQL είναι αρκετά εύκολη στην εκμάθηση της, ακόμα και όταν κάποιος που δεν έχει ξαναχρησιμοποιήσει παρόμοια προϊόντα κατασκευής βάσεων δεδομένων.
- **Μεταφερσιμότητα:** Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά σύγχρονα λειτουργικά συστήματα. Είναι συμβατή με πολλές εκδόσεις των Microsoft Windows και με λειτουργικά Unix, όπως οι διάφορες εκδόσεις του δημοφιλούς λειτουργικού ανοικτού κώδικα Linux.
- **Πηγαίος Κώδικας:** Η MySQL ανήκει στην οικογένεια του λογισμικού ανοικτού κώδικα. Συνεπώς ο καθένας μπορεί να αποκτήσει και να τροποποιήσει τον πηγαίο κώδικά της, προσαρμόζοντας τον στις ανάγκες του ή διορθώνοντας τυχόν bugs. Επιπλέον το γεγονός ότι είναι διαθέσιμος ο πηγαίος κώδικας βοηθάει στη συνεχή ανανέωση και διόρθωση της MySQL, αφού εκατομύρια άνθρωποι σε όλο τον κόσμο εργάζονται πάνω σε αυτή. Έτσι δεν χρειάζεται κάποιος να περιμένει μια νέα επίσημη έκδοσή κάποιας εταιρείας για τη διόρθωση ενός bug, αφού αυτό γίνεται πολύ γρήγορα από τους χρήστες της. Επιπλέον δε χρειάζεται να ανησυχεί κάποιος για τη μελλοντική υποστήριξη της και τη συνέχιση λειτουργίας της σε μελλοντικές συνθήκες.

Σύνοψη

Στην ενότητα αυτή ο εκπαιδευόμενος γνωρίζει και κατανοεί για την ανάπτυξη των ιστοτόπων, μαθαίνει για τον σχεδιασμό και την ανάπτυξη των ιστοσελίδων. Μπορεί να δημιουργήσει έναν όμορφο ισότοπο . Μαθαίνει για HTML, CSS και JavaScript τα απαραίτητα εργαλεία για τη δημιουργία δυναμικών, διαδραστικών ιστοσελίδων. Καταλαβαίνει την τριάδα των γλωσσών κωδικοποίησης, έτσι μπορείτε να

δημιουργήσετε και να ενημερώσετε ισότοπους για να είναι ακόμα πιο αποτελεσματικοί και μοναδικοί.

Ερωτήσεις

1. Τι είναι Παγκόσμιος Ιστός;

A. Ο Παγκόσμιος Ιστός είναι η διασύνδεση ενός μεγάλου συνόλου δεδομένων και εγγράφων που είναι προσβάσιμα μέσω του Διαδικτύου οπουδήποτε στον κόσμο.

B. Ο Παγκόσμιος Ιστός είναι η διασύνδεση από υπολογιστές αυτονόμους, ή μη αυτόνομους που μπορούν να επικοινωνήσουν μεταξύ τους

Γ. Ο Παγκόσμιος Ιστός είναι η διασύνδεση από υπολογιστές διασυνδεδεμένους που μπορούν να επικοινωνήσουν μεταξύ τους.

2. Πως συμβολίζεται ο Παγκόσμιος Ιστός τι σημαίνει ολόγραφος ;

A. Web Browser και συμβολίζεται wb.

B. World Wide Web και συμβολίζεται www.

3. Τι σημαίνει το HTTP;

A. Το HTTP σημαίνει Πρωτόκολλο μεταφοράς υπερκειμένου.

B. Το HTTP μεταφοράς υπερκειμένου

4. Τι είναι το URL;

A. Το URL είναι Uniform Resource Locator είναι το ίδιο με το HTTP, αλλά προσθέτει ένα επίπεδο ασφάλειας στο πρωτόκολλο.

B. Το URL είναι το ακρωνύμιο του Uniform Resource Locator Είναι μια διεύθυνση που προσδιορίζει πού βρίσκονται σελίδες, έγγραφα και αρχεία στον παγκόσμιο ιστό

5. Τι είναι το HTML;

A. Το HTML είναι το ακρωνύμιο του όρου HyperText Markup Language , το οποίο σημαίνει γλώσσα σήμανσης υπερκειμένου. Είναι μια από τις κύριες γλώσσες του ιστού.

B. Το HTML είναι το ακρωνύμιο του όρου HyperText Markup Language

Είναι μια διεύθυνση που προσδιορίζει πού βρίσκονται σελίδες, έγγραφα και αρχεία στον παγκόσμιο ιστό.

6. Τι είναι Διακομιστής.

A. Ένας διακομιστής είναι ένας υπολογιστής που περιέχει έναν ή περισσότερους ισχυρούς επεξεργαστές και παρέχει υπηρεσίες σε ένα δίκτυο υπολογιστών.

B. Ένας διακομιστής αποθηκεύει τα δεδομένα και τα έγγραφα που συνθέτουν έναν ιστότοπο .

7. Ποια προγράμματα χρησιμοποιεί ένας web designer;

A. Χρησιμοποιήστε εργαλεία λογισμικού όπως το Adobe Photoshop

B. Χρησιμοποιήστε εργαλεία λογισμικού όπως το Adobe Illustrator.

Γ. Όλα τα παραπάνω

8. Σε ποιες κατηγορίες χωρίζονται οι προγραμματιστές ιστού

A. Οι προγραμματιστές front-end και προγραμματιστές back-end.

B. Οι προγραμματιστές Ιστού δε χωρίζονται σε κατηγορίες.

9. Οι προγραμματιστές διεπαφής πόσες και ποιες γλώσσες πρέπει να γνωρίζουν;

A. Την JavaScript (JS).

B. Τουλάχιστον 3 γλώσσες HTML, CSS και JavaScript (JS).

10. Ποιο είναι το πρώτο στοιχείο που πρέπει να συλλέξει ένας σχεδιαστής ιστοσελίδας;

A. Ο Πελάτης είναι το βασικό στοιχείο που πρέπει να γνωρίζει πριν ξεκινήσει τον σχεδιασμό.

B. Οι τάσεις της μόδας είναι το βασικό και από εκεί θα ξεκινήσουν.

Άσκηση

Να δημιουργήσετε ιστοσελίδα, που να δέχεται τους τρεις βαθμούς μαθήματος (Α τετράμηνο/Β τετράμηνο/Γραπτή εξέταση) και να υπολογίζει το μέσο όρο τους.

Βιβλιογραφία

Chris Minnick **HTML, CSS και JavaScript για Dummies** (Exclusive B&N)

Evans Virginia, Dooley Jenny, Rogers Henrietta P. (2018) Career Paths- Art & Design.

Αρσένης, Σπύρος Δ. (2011) Σχεδιασμός πετυχημένων ιστοσελίδων Μάρκετινγκ και πωλήσεις προϊόντων και υπηρεσιών μέσω διαδικτύου: Με 50 μελέτες περιπτώσεων και λυμένες ασκήσεις

Hand Di & Middleditch Steve (2012) Design for Media

A Handbook for Students and Professionals in Journalism, PR, and Advertising

Nixon Robin(2014) Learning PHP, MySQL & JavaScript With jQuery, CSS & HTML5

Εκπαιδευτική Ενότητα 7. Σχεδιασμός και ανάπτυξη διαδικτυακών εφαρμογών

Σκοπός

Σκοπός αυτής της εκπαιδευτικής ενότητας είναι να μάθει ο εκπαιδευόμενος πως και ποιο, είναι το περιβάλλον και η αρχιτεκτονική μιας εφαρμογής, τι είναι ο αντικειμενοστραφής προγραμματισμός σε γλώσσα Java, ποιες είναι οι φάσεις ανάπτυξης μιας εφαρμογής και πως αναπτύσσουμε εφαρμογές σε android λογισμικό.

Προσδοκώμενα αποτελέσματα

Μετά το τέλος της μελέτης του υλικού θα είστε σε θέση να χρησιμοποιείτε τη γλώσσα προγραμματισμού Java, αφού αποκτήσετε γνώσεις σχετικά με το περιβάλλον μιας εφαρμογής και θα μπορείτε να αναπτύσσετε εφαρμογές με android λογισμικό.

Έννοιες- κλειδιά

Διαδικτυακές εφαρμογές είναι οι εφαρμογές που είναι διαθέσιμες στους χρήστες της μέσω του Internet ή του ενδοδικτύου μιας εταιρίας και ο ενδιαφερόμενος χρειάζεται μόνο τον περιηγητή του για να τις χρησιμοποιήσει.

Αντικειμενοστραφής προγραμματισμός, είναι ένα σύστημα ανάπτυξης προγραμμάτων, που υποστηρίζεται από γλώσσες προγραμματισμού, και ο χειρισμός δεδομένων και των διαδικασιών που επηρεάζουν αυτά γίνεται από κοινού, μέσω μίας δομής δεδομένων που τα περιβάλλει ως αυτόνομη οντότητα.

Java, είναι μια ευρέως χρησιμοποιούμενη γλώσσα προγραμματισμού υπολογιστών με πολλαπλές πλατφόρμες, αντικειμενοστρεφείς, γενικές δυνατότητες προγραμματισμού και χρησιμοποιείται ευρέως στην ανάπτυξη εφαρμογών web σε επίπεδο επιχείρησης και στην ανάπτυξη εφαρμογών για κινητά. τύποι δεδομένων,

android, είναι ανοιχτός κώδικας που βασίζεται σε πυρήνα Linux και άλλο λογισμικό ανοιχτού κώδικα. Κινητό λειτουργικό σύστημα, καθοδηγείται και αναπτύσσεται συνεχώς από την Κοινοπραξία Open Handset που ιδρύθηκε από την Google.

Εφαρμογές android (εφαρμογές για συντομία) είναι λογισμικό που επεκτείνει τη λειτουργικότητα της συσκευής και συντάσσεται χρησιμοποιώντας το κιτ ανάπτυξης λογισμικού Android (SDK) [108], συνήθως τη γλώσσα προγραμματισμού Java [109]

Εκπαιδευτική Υποενότητα 7.1 Περιβάλλον ανάπτυξης και αρχιτεκτονική μιας εφαρμογής

Εισαγωγή στις Progressive Web Apps (PWAs)

Οι Progressive Web Applications (PWAs), συγκροτούν ένα σύνολο από τεχνικές και Application Programming Interfaces (APIs), και δίνουν την δυνατότητα στους προγραμματιστές να πραγματοποιήσουν διαδικτυακές εφαρμογές με χαρακτηριστικά χρήσης - λειτουργίας, που συναντώνται σε εγγενείς εφαρμογές φορητών συστημάτων (native mobile applications). Η έννοια των PWAs δεν αφορά έναν συγκεκριμένο τρόπο υλοποίησης διαδικτυακών εφαρμογών με λειτουργικές δυνατότητες, αλλά οτιδήποτε θα μπορούσε να ενσωματωθεί σε μια διαδικτυακή εφαρμογή. Δίνοντάς της τόσο προς την εμφάνιση, όσο και ως προς την λειτουργία, χαρακτηριστικά που παρουσιάζονται σε εφαρμογές που αναπτύσσονται εγγενώς για φορητά συστήματα.



Υπάρχουν τρεις καθοδηγητικές έννοιες που περιγράφουν καλύτερα τις επιδιώξεις που πρέπει να έχει ένας προγραμματιστής διαδικτυακών εφαρμογών για την ανάπτυξη τους ως PWAs (“Progressive Web Apps: A new way to deliver amazing user experiences on the web”, n.d.):

- Αξιοπιστία (Reliability): Μια διαδικτυακή εφαρμογή θα πρέπει να φορτώνεται στιγμιαία, ακόμη και σε αβέβαιες συνθήκες όσον αφορά την δυνατότητα σύνδεσης της συσκευής στο διαδίκτυο.

- Ταχύτητα (Fast): Η διαδικτυακή εφαρμογή θα πρέπει να εκτελείται και να ανταποκρίνεται γρήγορα σε ενέργειες του χρήστη, να παρέχει ομαλά τα κινούμενα μέρη της διεπαφής της και να αποκτά πρόσβαση σε συστατικά του εξοπλισμού της φορητής συσκευής όπως η φωτογραφική μηχανή, χωρίς καθυστερήσεις.
- Ενθάρρυνση χρήσης (Engagement): Αν η διαδικτυακή εφαρμογή προσφέρει την εμπειρία χρήσης μιας εγγενούς εφαρμογής φορητού συστήματος, είναι πιθανότερο να κεντρίσει το ενδιαφέρον των χρηστών της και να ενισχύσει την επισκεψιμότητά της. Για παράδειγμα, μια δυνατότητα των PWAs είναι η προώθηση ειδοποιήσεων (push notifications), δηλαδή η αποστολή μηνυμάτων σχετικών με την εφαρμογή, ακόμη και όταν είναι κλειστή και δεν εκτελείται στην συσκευή. Η αποστολή τέτοιων μηνυμάτων στα πλαίσια της σκοπιμότητας της εφαρμογής, είναι πολύ πιθανό να επαναφέρει το ενδιαφέρον του χρήστη για την εφαρμογή.

Σε σχέση με τις παραπάνω έννοιες, θα μπορούσαν να αναφερθούν ως πιο αντιπροσωπευτικές δυνατότητες μιας διαδικτυακής εφαρμογής, που λειτουργεί ως PWA, τις ακόλουθες:

- Άνοιγμα της διαδικτυακής εφαρμογής από εικονίδιο στην αρχική οθόνη (home screen) της συσκευής.
- Λειτουργία της διαδικτυακής εφαρμογής χωρίς σύνδεση στο διαδίκτυο (offline).
- Πρόσβαση της εφαρμογής σε μέρη του εξοπλισμού της φορητής συσκευής, όπως είναι η φωτογραφική μηχανή, ο δέκτης GNSS, το κλινόμετρο κ.α.
- Αποστολή ενημερώσεων στον χρήστη για την εφαρμογή, ακόμη και όταν έχει κλείσει.
- Συγχρονισμός δεδομένων στο παρασκήνιο, όταν η συσκευή αποκαταστήσει την πρόσβασή της στο διαδίκτυο.

Τι είναι οι διαδικτυακές εφαρμογές (web application)

Διαδικτυακή εφαρμογή (web application ή web app) ονομάζεται κάθε εφαρμογή η οποία είναι διαθέσιμη στους χρήστες της μέσω του Διαδικτύου (Internet) ή του

ενδοδικτύου (Intranet) μιας εταιρίας και ο χρήστης χρειάζεται μόνο τον περιηγητή του για να την χρησιμοποιήσει.

Οι εφαρμογές αυτές συνήθως εκτελούνται σε ισχυρές υπολογιστικές μηχανές, οι οποίες έχουν τον ρόλο του σταθμού εξυπηρέτησης και παρέχουν τις υπηρεσίες τους σε περισσότερους του ενός χρήστη.

Πλεονεκτήματα και μειονεκτήματα διαδικτυακών εφαρμογών

Οι τοπικές εφαρμογές υπολογιστών παραμένουν μέχρι και σήμερα οι πιο δημοφιλείς εφαρμογές και αυτό οφείλεται εν μέρη στο ότι οι τοπικές εφαρμογές ήταν και οι πρώτες εφαρμογές που δημιουργήθηκαν. Με την πάροδο του χρόνου όμως όλο και περισσότερες διαδικτυακές εφαρμογές κάνουν την εμφάνιση τους. Συγκρίνοντας τις δυο κατηγορίες εφαρμογών οι χρήστες θα εντοπίσουν τόσο θετικά όσο και αρνητικά στοιχεία και στις δυο και σε συνάρτηση με τις εργασίες που θέλουν η εφαρμογή τους να πραγματοποιεί, καλούνται να αποφασίσουν ποια από τις δύο κατηγορίες είναι αυτή που τους ταιριάζει.

Πλεονεκτήματα:

- Άμεση πρόσβαση από οποιαδήποτε συσκευή: Οι χρήστες των διαδικτυακών εφαρμογών έχουν άμεση προσβασιμότητα στις εφαρμογές που θέλουν να χρησιμοποιήσουν από οποιονδήποτε υπολογιστή ή άλλη συσκευή έχει internet χωρίς την εγκατάσταση κάποιου επιπρόσθετου λογισμικού. Η μόνη απαραίτητη εφαρμογή είναι ο περιηγητής διαδικτύου ο οποίος είναι προεγκατεστημένος σε όλα τα λειτουργικά συστήματα ακόμα και στις φορητές συσκευές αλλά και στα κινητά τηλέφωνα. Η ιδιότητα αυτή των διαδικτυακών εφαρμογών είναι ιδιαίτερα σημαντική για μεγάλες επιχειρήσεις με πολλούς χρήστες που στην περίπτωση της τοπικής εφαρμογής θα έπρεπε να εγκατασταθεί η εφαρμογή σε κάθε ένα υπολογιστή ξεχωριστά.
- Δυνατότητα χρήσης ανεξαρτήτως τοποθεσίας: Ως συνέχεια του παραπάνω οι χρήστες των διαδικτυακών εφαρμογών μπορούν να τις χρησιμοποιούν ακόμα και αν

δεν βρίσκονται στον χώρο εργασίας τους. Η δυνατότητα αυτή δίνει ευελιξία στους χρήστες ώστε να χρησιμοποιούν τις εφαρμογές οπουδήποτε αυτοί επιθυμούν επιτρέποντας τους ακόμα και να εργάζονται από απομακρυσμένες περιοχές ή και από το σπίτι τους.

- Συμβατές με όλα τα λειτουργικά συστήματα: Ένα ακόμα πλεονέκτημα των διαδικτυακών εφαρμογών είναι ότι είναι συμβατές με όλα τα λειτουργικά συστήματα. Καθώς η εφαρμογή εκτελείται μέσω του περιηγητή του διαδικτύου και όχι στον υπολογιστή του χρήστη, την κάνει ικανή να εκτελείται σε όλα τα λειτουργικά συστήματα. Η ιδιότητα αυτή οφείλεται επίσης και στην προτυποποίηση των γλωσσών προγραμματισμού τις οποίες χρησιμοποιεί η εφαρμογή.
- Δεν καταναλώνουν πόρους: Ως συνέχεια του παραπάνω και εφόσον οι διαδικτυακές εφαρμογές δεν εκτελούνται στον υπολογιστή του χρήστη δεν καταναλώνουν και πόρους από το σύστημα. Για τον λόγο αυτό οι εφαρμογές διαδικτύου είναι ιδιαίτερα ελαφριές για την υπολογιστική μονάδα.
- Δεν καταλαμβάνουν χώρο: Ακολουθώντας την ίδια λογική με νωρίτερα οι εφαρμογές αυτές δεν καταλαμβάνουν καθόλου ή σχεδόν καθόλου χώρο στον δίσκο του χρήστη αφού το σύνολο της εφαρμογής είναι αποθηκευμένο στον εξυπηρετητή και μόνο κατά την χρήση της εφαρμογής μπορεί να υπάρχει μεταφορά δεδομένων προς την υπολογιστική μονάδα του χρήστη και μόνο στην περίπτωση που ο χρήστης το επιθυμεί.
- Γρήγορη αναβάθμιση: Σημαντικό πλεονέκτημα συγκριτικά με τις τοπικές εφαρμογές εμφανίζεται στις περιπτώσεις που η εφαρμογή χρειάζεται κάποια αναβάθμιση. Σε μια κλασική τοπική εφαρμογή η αναβάθμιση του συστήματος θα πρέπει να γίνει σε κάθε ένα υπολογιστή ξεχωριστά πράγμα που απαιτεί χρόνο και χρήμα. Αντίθετα σε μια διαδικτυακή εφαρμογή η αναβάθμιση πραγματοποιείται μόνο στον εξυπηρετητή που φιλοξενεί την εφαρμογή και ταυτόχρονα το αναβαθμισμένο πρόγραμμα είναι διαθέσιμο σε όλους τους χρήστες. Με τον τρόπο αυτό εξοικονομείται χρόνος ο οποίος είναι ιδιαίτερα πολύτιμος κυρίως για τις μεγάλες επιχειρήσεις ενώ ως συνέπεια του παραπάνω σημαντικά μειωμένο είναι

και το κόστος της αναβάθμισης μιας και απαιτείται λιγότερο εργατικό δυναμικό για την διεκπεραίωση της αναβάθμισης.

- **Νέο, βελτιωμένο περιβάλλον:** Ένα ακόμα πλεονέκτημα των διαδικτυακών εφαρμογών είναι ότι πλέον με την εμφάνιση της HTML5 είναι δυνατό ο δημιουργός της εφαρμογής να την εμπλουτίσει έτσι ώστε να είναι πιο φιλική, εύχρηστη και ευχάριστη προς τον χρήστη με εύκολο τρόπο. Παλαιότερα οι εφαρμογές αυτές υστερούσαν στην εμφάνιση ωστόσο πλέον είναι ιδιαίτερα εύκολα να εμπλουτιστούν.
- **Δυνατότητα χρήσης και εκτός διαδικτύου - ενδοδικτύου:** Ένα ακόμα πλεονέκτημα των σύγχρονων διαδικτυακών εφαρμογών (εφαρμογές με χρήση HTML5) είναι η δυνατότητα της εκτός διαδικτύου χρήσης μιας διαδικτυακής εφαρμογής με την προϋπόθεση ότι η εφαρμογή έχει κατασκευαστεί με ανάλογο τρόπο. Για παράδειγμα αν για κάποιο λόγο η σύνδεση στο διαδίκτυο διακοπεί αυτό δεν επηρεάζει τον χρήστη, ο οποίος συνεχίζει να χρησιμοποιεί την εφαρμογή κανονικά. Αυτό επιτυγχάνεται από τον περιηγητή ο οποίος κρατάει ένα αντίγραφο από τα αρχεία, τα οποία είναι απαραίτητα για την εκτός δικτύου χρήση της εφαρμογής, στον υπολογιστή του χρήστη, και τα χρησιμοποιεί όταν αυτό κριθεί απαραίτητο. Η συγκεκριμένη δυνατότητα δεν είναι διαθέσιμη σε όλες τις εφαρμογές που χρησιμοποιούν HTML5 αλλά μόνο σε αυτές που έχει υπάρξει πρόβλεψη για χρήση της εφαρμογής και εκτός διαδικτύου ή ενδοδικτύου.

Μειονεκτήματα:

- **Χρήση της εφαρμογής εκτός διαδικτύου:** Ένα μειονέκτημα που ταυτόχρονα είναι και πλεονέκτημα είναι η χρήση της εφαρμογής εκτός διαδικτύου. Προκειμένου να γίνει αυτό εφικτό, θα πρέπει να έχει γίνει πρόβλεψη κατά την σχεδίαση της εφαρμογής και να έχουν ληφθεί τα κατάλληλα μέτρα. Σε περίπτωση που κάτι τέτοιο δεν έχει γίνει τότε η εφαρμογή δεν είναι δυνατό να χρησιμοποιηθεί χωρίς την σύνδεση του χρήστη με το Διαδίκτυο ή το ενδοδίκτυο της εταιρίας.

- Αδυναμία χρήσης χωρίς σύνδεση στο διαδίκτυο: Σαν συνέχεια του παραπάνω, εφαρμογές που δεν έχουν κατασκευαστεί με χρήση της τελευταίας έκδοσης της HTML, δηλαδή παλαιότερες εφαρμογές δεν είναι δυνατόν να χρησιμοποιηθούν αν δεν υπάρχει σύνδεση με το διαδίκτυο ή το ενδοδίκτυο. Το παραπάνω αποτελεί και το κύριο μειονέκτημα των παλαιότερων διαδικτυακών εφαρμογών καθώς αν για κάποιο λόγο η σύνδεση του χρήστη διακοπεί τότε αυτός δεν μπορεί να χρησιμοποιήσει καθόλου την εφαρμογή. Σε περίπτωση που μια παλαιότερη διαδικτυακή εφαρμογή θελήσει να αυξήσει τις δυνατότητες της προκειμένου να μπορεί να λειτουργεί και εκτός διαδικτύου τότε αυτή θα πρέπει να ξανακατασκευαστεί σε μεγάλο μέρος της από την αρχή.
- Μη πλήρης συμβατότητα των περιηγητών: Ένα ακόμα μειονέκτημα που αφορά την τελευταία έκδοση της HTML είναι η μη πλήρης συμβατότητα των περιηγητών με την έκδοση αυτή. Αν και τα πλεονεκτήματα και οι δυνατότητες της HTML5 είναι πολλά αρκετοί από τους περιηγητές δεν είναι ακόμα πλήρως συμβατοί με αυτά. Έτσι δεν γίνεται πλήρης χρήση των δυνατοτήτων αυτών, πράγμα που περιορίζει τους προγραμματιστές που έχουν αναλάβει ένα έργο. Επίσης, σε περίπτωση που δεν έχει προβλεφθεί η μη λειτουργία κάποιου χαρακτηριστικού της εφαρμογής σε κάποιον περιηγητή, αυτό μπορεί να δημιουργήσει προβλήματα στην εφαρμογή με αποτέλεσμα να μην λειτουργεί σωστά ή να μην λειτουργεί καθόλου. Για το λόγο αυτό ο κατασκευαστής της εφαρμογής με τον πελάτη πρέπει από κοινού να αποφασίζουν ποιος περιηγητής θα είναι ο προτεινόμενος για την εφαρμογή αλλά ταυτόχρονα να προβλέπεται και η περίπτωση χρήσης άλλων περιηγητών. Ένας καλός τρόπος για να ελεγχθεί η συμβατότητα του περιηγητή μας με την HTML5 είναι τα διάφορα διαδικτυακά τεστ που αξιολογούν τις δυνατότητες του.
- Άμεση αναβάθμιση: Ένα ακόμα χαρακτηριστικό παράδειγμα πλεονεκτήματος και μειονεκτήματος ταυτόχρονα αποτελεί και η αναβάθμιση της εφαρμογής. Στην περίπτωση της τοπικής εφαρμογής μια επιχείρηση μπορεί να αναβαθμίσει την εφαρμογή που χρησιμοποιεί όποτε αυτή το κρίνει αναγκαίο κρίνοντας το κόστος αναβάθμισης, την αξιοπιστία της νέας εφαρμογής αλλά και το χρόνο που θα χρειαστούν οι υπάλληλοι της ώστε να προσαρμοστούν στην νέα έκδοση. Αντίθετα στις διαδικτυακές εφαρμογές η αναβάθμιση γίνεται χωρίς πρώτα

να ερωτηθούν όλοι οι χρήστες. Για παράδειγμα στην περίπτωση που η ερχόμενη αναβάθμιση μιας τοπικής εφαρμογής έχει σφάλματα τότε μπορεί κάποιος χρήστης (εταιρία) να μην πραγματοποιήσει την αναβάθμιση έως ότου διορθωθούν αυτά. Στην περίπτωση όμως της διαδικτυακής εφαρμογής ο χρήστης (εταιρία) δεν μπορεί να αποτρέψει την αναβάθμιση αυτή.

- Πιθανή μη συμβατότητα κάποιων στοιχείων της εφαρμογής με μια μελλοντική έκδοση του περιηγητή μας: Τέλος, μειονέκτημα είναι και η πιθανή μη συμβατότητα κάποιων στοιχείων του προγράμματος με μια μελλοντική έκδοση του περιηγητή. Αυτό συμβαίνει διότι συχνά παρατηρείται το φαινόμενο να εγκαταλείπονται κάποια υποστηριζόμενα στοιχεία από τους περιηγητές αν η δημιουργός εταιρία κρίνει ότι αυτά δεν έχουν μέλλον στις εφαρμογές διαδικτύου. Το αποτέλεσμα αυτών των αποφάσεων είναι η δυσλειτουργία κάποιων παλαιότερων εφαρμογών καθιστώντας αναγκαία την αναβάθμιση τους

Αρχιτεκτονική εφαρμογής

Η εφαρμογή χωρίζεται σε τρία αλληλεξαρτώμενα στάδια λειτουργίας:

A. Στο πρώτο στάδιο ο διαχειριστής της εφαρμογής εισέρχεται στο σύστημα μέσω HTML ιστοσελίδας, είτε με τη διαδικασία της εγγραφής, είτε εφόσον είναι ήδη εγγεγραμμένος χρήστης με τη διαδικασία της σύνδεσης. Και στις δύο περιπτώσεις υπάρχει επικοινωνία της φόρμας με τη MySQL βάση δεδομένων, μέσω των PHP σελίδων. Αυτό συμβαίνει είτε γιατί πρέπει να γίνει εγγραφή προσωπικών στοιχείων στη βάση, είτε γιατί πρέπει να γίνει επικύρωση των εισαχθέντων στοιχείων. Ο έλεγχος για την ορθή συμπλήρωση των στοιχείων ασκείται στην πλευρά του χρήστη με HTML και στην πλευρά του διακομιστή με PHP. Στη συνέχεια ο διαχειριστής μπορεί να δημιουργήσει ή να επεξεργαστεί τη μελέτη αξιολόγησης γραφικού σχεδιασμού 8 επιφανειών. Μέσω οδηγού (wizard) καταχωρεί τις απαραίτητες πληροφορίες ανεβάζει τα αρχεία που πρόκειται να αξιολογηθούν και προσκαλεί τους συμμετέχοντες. Η φόρμα δημιουργίας και διαχείρισης είναι HTML και επικοινωνεί με τη βάση δεδομένων μέσω PHP. Επίσης ασκείται έλεγχος για την ορθή συμπλήρωση των στοιχείων και για τυχούσα ύπαρξη όμοιων στοιχείων ή

αρχείων στο διακομιστή. Η πρόσκληση των συμμετεχόντων ελέγχεται για την ορθότητα και την πληρότητα των στοιχείων. Σε όλες τις φόρμες ο έλεγχος ασκείται στην πλευρά του χρήστη με AJAX και στην πλευρά του διακομιστή με PHP.

Β. Στο δεύτερο στάδιο λαμβάνει χώρα η διεξαγωγή της μελέτης είτε με πρόσκληση μέσω ηλεκτρονικού ταχυδρομείου, είτε με απευθείας συμμετοχή μέσω επιλογής κατάλληλου συνδέσμου. Τόσο το μήνυμα όσο και ο σύνδεσμος δημιουργούνται με PHP κώδικα. Η συμμετοχή των χρηστών επικυρώνεται με PHP έλεγχο. Η σελίδα διεξαγωγής του πειράματος είναι HTML και χρησιμοποιεί το drag and drop API της jQuery καθώς και PHP για την αποστολή των αποτελεσμάτων στο διακομιστή.

Γ. Στο τρίτο και τελευταίο στάδιο λαμβάνει χώρα η αποθήκευση των αποτελεσμάτων μέσω PHP κώδικα και η επεξεργασία τους στο διακομιστή. Ο διαχειριστής έχει πρόσβαση στα αποτελέσματα της αξιολόγησης με επικοινωνία με τη βάση δεδομένων μέσω PHP.

Το περιβάλλον ανάπτυξης της εφαρμογής

Τα προγράμματα σε έναν υπολογιστή χωρίζονται σε δύο μεγάλες κατηγορίες: στο Λογισμικό εφαρμογών και στο Λογισμικό συστήματος. Παραδείγματα προγραμμάτων που ανήκουν στο Λογισμικό Εφαρμογών είναι τα προγράμματα επεξεργασίας κειμένου, φωτογραφίας, ήχου, βίντεο, τα προγράμματα παρουσίασης, τα εκπαιδευτικά προγράμματα, τα παιχνίδια. Τα προγράμματα που ανήκουν στο Λογισμικό Εφαρμογών ονομάζονται απλά και εφαρμογές (applications - apps).

Η λέξη «εφαρμογή» χρησιμοποιείται, επειδή κάθε πρόγραμμα έχει μία συγκεκριμένη εφαρμογή για τον χρήστη του και βασίζεται σε μία ανάγκη του. Οι εφαρμογές πρέπει να υποστηρίζουν τις καθημερινές δραστηριότητες των χρηστών των υπολογιστών και των φορητών συσκευών (έξυπνων κινητών, tablets) με αποδοτικό και γρήγορο τρόπο. Για παράδειγμα, ο επεξεργαστής κειμένου μπορεί να χρησιμοποιηθεί από έναν μαθητή για τη συγγραφή και μορφοποίηση μιας εργασίας

σε ένα μάθημα. Το πρόβλημα του μαθητή είναι η συγγραφή της εργασίας του και σε αυτό τον βοηθάει ο υπολογιστής και το πρόγραμμα επεξεργασίας κειμένου (εφαρμογή) που εκτελείται σε αυτόν.

Το περιβάλλον ανάπτυξης μιας εφαρμογής μας παρέχει τη δυνατότητα να σχεδιάζουμε και να υλοποιούμε εφαρμογές. Κάθε περιβάλλον υποστηρίζει συγκεκριμένες γλώσσες προγραμματισμού και έχει ιδιαίτερα χαρακτηριστικά. Αυτό δεν αποκλείει την κατηγοριοποίησή τους με βάση κάποια κοινά χαρακτηριστικά. Έτσι, υπάρχουν τα επαγγελματικά προγραμματιστικά περιβάλλοντα που χρησιμοποιούνται κυρίως από έμπειρους επαγγελματίες προγραμματιστές με στόχο την παραγωγή εφαρμογών για εμπορική χρήση. Ένα τέτοιο περιβάλλον δεν προτείνεται για εκμάθηση προγραμματισμού σε αρχάριους χρήστες, όπως οι μαθητές. Ειδικά γι' αυτό τον σκοπό έχουν σχεδιαστεί απλούστερα περιβάλλοντα, που διευκολύνουν τη σύνταξη των εντολών, για να είναι εύκολη η εκμάθηση της ανάπτυξης ενός προγράμματος, και ονομάζονται εκπαιδευτικά προγραμματιστικά περιβάλλοντα.

Γλώσσες και εργαλεία προγραμματισμού

Γλώσσες προγραμματισμού

Οι εντολές των προγραμμάτων γράφονται από τους προγραμματιστές σε τεχνητές γλώσσες που ονομάζονται γλώσσες προγραμματισμού. Οι γλώσσες προγραμματισμού εξελίσσονται με την πάροδο του χρόνου και οποιαδήποτε εφαρμογή υπάρχει στον υπολογιστή έχει αναπτυχθεί με χρήση μιας από αυτές. Κάθε υπολογιστής μπορεί να κατανοήσει και να εκτελέσει εντολές που είναι διατυπωμένες με έναν καθορισμένο τρόπο, ο οποίος έχει σχέση με τον επεξεργαστή του. Οι γλώσσες προγραμματισμού, στις οποίες γράφονται οι εντολές αυτές, ονομάζονται γλώσσες μηχανής, και αποτελούνται από μια ακολουθία δυαδικών ψηφίων (0 και 1). Τα προγράμματα που είναι γραμμένα σε γλώσσα μηχανής προορίζονται μόνο για τον υπολογιστή για τον οποίο δημιουργήθηκαν, μιας και

κάθε τύπος υπολογιστή (με διαφορετικό επεξεργαστή) έχει τη δική του γλώσσα μηχανής. Μια γλώσσα μηχανής διακρίνεται για την άμεση και γρήγορη εκτέλεση των εντολών από τον επεξεργαστή του υπολογιστή, αλλά ταυτόχρονα είναι δύσκολη η χρήση της, ο εντοπισμός και η διόρθωση τυχόν λαθών.

Για τη διευκόλυνση της εργασίας του προγραμματισμού δημιουργήθηκαν οι συμβολικές γλώσσες ή γλώσσες χαμηλού επιπέδου, όπου οι εντολές είναι συντομογραφίες λέξεων της Αγγλικής γλώσσας. Οι συμβολικές γλώσσες είναι και αυτές στενά συνδεδεμένες με την αρχιτεκτονική των υπολογιστών, και έτσι ένα πρόγραμμα γραμμένο για έναν τύπο υπολογιστή δεν μπορεί να μεταφερθεί και να εκτελεστεί σε άλλο τύπο υπολογιστή. Ένα πρόγραμμα γραμμένο σε συμβολική γλώσσα, για να εκτελεστεί από έναν υπολογιστή, πρέπει να μεταφραστεί στη γλώσσα μηχανής του με ένα ειδικό πρόγραμμα που ονομάζεται συμβολομεταφραστής.

Η ανάγκη για ακόμα ευκολότερη συγγραφή, διόρθωση και συντήρηση προγραμμάτων, ανεξάρτητων από τον τύπο του υπολογιστή στον οποίο θα εκτελεστούν, οδήγησε στη δημιουργία των γλωσσών υψηλού επιπέδου.

Οι γλώσσες υψηλού επιπέδου μοιάζουν με τη φυσική μας γλώσσα και έχουν το δικό τους αλφάβητο, λεξιλόγιο και συντακτικό.

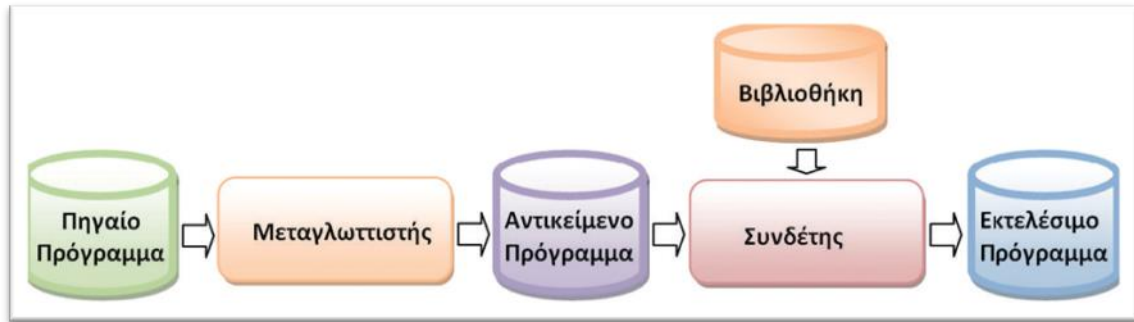
Μερικές από τις πιο δημοφιλείς γλώσσες για ανάπτυξη γενικών ή εξειδικευμένων εφαρμογών είναι η C, η C++, η Java, η PHP, η C#, η Python, η JavaScript, η Perl, η Visual Basic, η Ruby, η Lisp, η Pascal, η Prolog και η MATLAB. Ανάλογα με το είδος της εφαρμογής που πρέπει να αναπτυχθεί (γενικής χρήσης, εμπορική, επιστημονική, τεχνητής νοημοσύνης, παιχνίδι κ.λπ.), επιλέγεται η κατάλληλη γλώσσα προγραμματισμού.

Εργαλεία προγραμματισμού μιας εφαρμογής

Η εργασία σύνταξης ενός προγράμματος ονομάζεται προγραμματισμός ή κωδικοποίηση, και είναι μια εξαιρετικά δημιουργική δραστηριότητα. Τα κύρια εργαλεία που χρησιμοποιεί ένας προγραμματιστής για να αναπτύξει μία εφαρμογή σε μία συγκεκριμένη γλώσσα προγραμματισμού υψηλού επιπέδου είναι:

- ένας συντάκτης κειμένων (editor) με τον οποίο και γράφει το αρχικό πρόγραμμα, που ονομάζεται πηγαίο πρόγραμμα ή κώδικας (source code).
- ένα μεταφραστικό πρόγραμμα (μεταγλωττιστή ή διερμηνευτή), το οποίο μεταφράζει το πηγαίο πρόγραμμα σε αντικείμενο πρόγραμμα ή κώδικα (object code). Το μεταφραστικό πρόγραμμα ελέγχει το πηγαίο πρόγραμμα για συντακτικά λάθη, εμφανίζει κατάλληλα διαγνωστικά μηνύματα, εάν βρεθούν λάθη, και μόνο αν δεν υπάρχουν λάθη παράγεται το αντικείμενο πρόγραμμα. Το αντικείμενο πρόγραμμα είναι σε γλώσσα μηχανής, αλλά δεν είναι ακόμη εκτελέσιμο από τον υπολογιστή και πρέπει να περάσει από κάποιες άλλες διαδικασίες.
- ένα ειδικό πρόγραμμα που ονομάζεται σύνθετης (linker), το οποίο πολλές φορές συνδέει το αντικείμενο πρόγραμμα ή ένα σύνολο από αντικείμενα προγράμματα με έτοιμα υποπρογράμματα της βιβλιοθήκης της γλώσσας προγραμματισμού ή του προγραμματιστή. Το τελικό πρόγραμμα που παράγεται είναι το εκτελέσιμο πρόγραμμα ή κώδικας (executable code), είναι διατυπωμένο σε γλώσσα μηχανής και μπορεί να εκτελεστεί άμεσα από τον επεξεργαστή του υπολογιστή.
- εργαλεία εντοπισμού λαθών (debuggers) με τα οποία ο προγραμματιστής παρακολουθεί τι ακριβώς συμβαίνει στο παρασκήνιο κατά την εκτέλεση ενός προγράμματος.

Ένα περιβάλλον (λογισμικό), που περιλαμβάνει τα παραπάνω εργαλεία και χρησιμοποιείται για την ανάπτυξη εφαρμογών ονομάζεται προγραμματιστικό περιβάλλον ή περιβάλλον ανάπτυξης εφαρμογών.



Επαγγελματικά προγραμματιστικά περιβάλλοντα

Η ανάπτυξη των σύγχρονων εφαρμογών είναι μια απαιτητική και δύσκολη διαδικασία. Οι επαγγελματίες προγραμματιστές χρησιμοποιούν για τη σχεδίαση, την κωδικοποίηση, τον έλεγχο λαθών και τη συντήρηση μιας εφαρμογής ένα ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment -IDE), όπως Dev-C++, BlueJ, JBuilder, NetBeans IDE, Eclipse, Microsoft Visual Studio, Apple Xcode και Komodo IDE & EDIT. Στα ολοκληρωμένα περιβάλλοντα ανάπτυξης συνυπάρχουν σε ένα ενοποιημένο περιβάλλον διάφορα από τα εργαλεία που αναφέρθηκαν παραπάνω, και έτσι διευκολύνεται και επιταχύνεται η ανάπτυξη μιας εφαρμογής είτε από έναν είτε από ομάδα προγραμματιστών.

Μάλιστα κάποια από αυτά διαθέτουν και γραφικά εργαλεία σχεδίασης των εφαρμογών, για παράδειγμα οπτικοποίηση με διαγράμματα των τμημάτων μιας εφαρμογής και της μεταξύ τους αλληλεπίδρασης, αυτόματη συμπλήρωση κώδικα (εντολών), δημιουργία των αντικειμένων της γραφικής διεπαφής χρήστη (μενού επιλογών, κουμπιά, πλαίσια διαλόγου κ.λπ.) με χρήση κατάλληλων εργαλειοθηκών. Επίσης, αρκετά εγκαθίστανται σε διάφορα Λειτουργικά Συστήματα (Windows, Linux, Mac OS X), υποστηρίζουν σχεδόν όλες τις δημοφιλείς γλώσσες προγραμματισμού (Java, C/C++, Python, PHP, JavaScript) και χρησιμοποιούνται για την ανάπτυξη αυτόνομων εφαρμογών για υπολογιστή, εφαρμογών για το Διαδίκτυο και εφαρμογών για φορητές συσκευές (έξυπνα κινητά, tablets).

Εκπαιδευτική Υποενότητα 7.2 Κύκλος ανάπτυξης ολοκληρωμένης εφαρμογής
Η ανάπτυξη εφαρμογών πρέπει να ακολουθεί μια συστηματική διαδικασία με βήματα-φάσεις, ώστε να αποφεύγονται τα σφάλματα, οι δυσλειτουργίες και οι ελλείψεις. Εξάλλου, με το πέρασμα των χρόνων, τα προγράμματα γίνονται εκτενέστερα σε μέγεθος και πιο πολύπλοκα, οπότε και η κατασκευή τους γίνεται πιο απαιτητική.

Μια εφαρμογή ξεκινάει τον κύκλο ζωής της από τη στιγμή που θα καθοριστούν οι απαιτήσεις και οι προδιαγραφές της, και τελειώνει, όταν εξαντληθούν τα περιθώρια συντήρησής της (προσθήκες, αλλαγές και βελτιώσεις). Οι εμπλεκόμενοι στη διαδικασία αυτή είναι ο πελάτης (εταιρεία, οργανισμός ή άτομο) που επενδύει στην ανάπτυξη της εφαρμογής, ο κατασκευαστής (εταιρεία, οργανισμός ή άτομο-προγραμματιστής) που αναπτύσσει την εφαρμογή και οι χρήστες, οι οποίοι θα χρησιμοποιήσουν την εφαρμογή.



Φάσεις ανάπτυξης μιας εφαρμογής

- Φάση ανάλυσης: καταγράφονται αναλυτικά τα δεδομένα και τα ζητούμενα του προβλήματος που καλείται να επιλύσει η υπό ανάπτυξη εφαρμογή. Περιγράφονται οι προδιαγραφές και οι απαιτήσεις των μελλοντικών χρηστών της εφαρμογής: ποιες λειτουργίες θα υποστηρίζει, πώς θα εκτελούνται αυτές οι λειτουργίες, σε ποιο περιβάλλον θα δουλεύει, πόσο αποδοτική, εύχρηστη, ασφαλής και αξιόπιστη θα είναι η εφαρμογή.

- **Φάση σχεδίασης:** καθορίζονται οι ενότητες (μέρη), από τις οποίες θα αποτελείται η εφαρμογή καθώς και οι σχέσεις μεταξύ τους. Σχεδιάζονται οι αλγόριθμοι και επιλέγονται οι δομές δεδομένων που θα χρησιμοποιηθούν σε κάθε ενότητα.
- **Φάση υλοποίησης:** επιλέγεται η γλώσσα προγραμματισμού για την υλοποίηση της εφαρμογής. Οι προγραμματιστές με βάση τους αλγόριθμους και τις δομές δεδομένων της προηγούμενης φάσης γράφουν το πρόγραμμα στην επιλεγμένη γλώσσα προγραμματισμού, αυτό εισάγεται σε ειδικό πρόγραμμα-μεταφραστή, ώστε να μετατραπεί σε «γλώσσα» κατανοητή από τον υπολογιστή, και, αν δεν υπάρχουν συντακτικά λάθη, η εφαρμογή είναι έτοιμη για εκτέλεση και χρήση.
- **Φάση λειτουργίας:** η εφαρμογή δίνεται αρχικά στους χρήστες για δοκιμές και ελέγχους, ώστε να βρεθούν και διορθωθούν πιθανά λάθη και αποκλίσεις από τις αρχικές προδιαγραφές, και έπειτα ξεκινάει η κανονική χρήση της.
- **Φάση συντήρησης:** γίνονται όλες οι απαραίτητες προσαρμογές, αναβαθμίσεις και διορθώσεις της εφαρμογής, προκειμένου αυτή να συνεχίσει να χρησιμοποιείται απρόσκοπτα και αποδοτικά.

Η έννοια του αλγόριθμου στη φάση της σχεδίασης

Είναι μια σειρά από ενεργειών και βημάτων που αποτελούν μια υπολογιστική διαδικασία και εκτελείται σε πεπερασμένο χρόνο. Σε κάθε αλγόριθμο πρέπει να καθορίζονται τα παρακάτω στοιχεία:

- **Είσοδος (input):** Απεικονίζει τις τιμές των δεδομένων του προβλήματος που δίνονται από την αρχή ή παράγονται από ενδιάμεσα τμήματα του και αποτελούν είσοδο για τα επόμενα (και ονομάζονται στιγμιότυπα). Όταν τα δεδομένα εισόδου καλύπτουν κάθε φορά τις προδιαγραφές του συστήματος τότε η είσοδος του αλγόριθμου καλείται νόμιμη.
- **Έξοδος (output):** Η έξοδος του αλγόριθμου είναι η μερική ή τελική λύση του προβλήματος.

- **Πεπερασμένα βήματα:** Κάθε αλγόριθμος πρέπει να αποτελείται από συγκεκριμένα βήματα και να ολοκληρώνεται μετά την εκτέλεση τους.
- **Σαφή απόδοση:** Το γεγονός αυτό σημαίνει ότι ο χρήστης – προγραμματιστής πρέπει να γνωρίζει την ταχύτητα εκτέλεσης του αλγόριθμου του, που είναι ανάλογη του πλήθους των πράξεων που εκτελεί για την επίλυση του προβλήματος, ώστε να μπορεί να τον αξιολογήσει σε σύγκριση με άλλους αντίστοιχους για το ίδιο πρόβλημα.

Δομές δεδομένων στη φάση σχεδίασης

Είναι ένα σύνολο αποθηκευμένων δεδομένων που υφίστανται επεξεργασία από ένα σύνολο λειτουργιών. Κάθε μορφή δομής δεδομένων αποτελείται από ένα σύνολο κόμβων (nodes). Οι βασικές λειτουργίες ή πράξεις των δομών δεδομένων είναι οι παρακάτω:

- **Προσπέλαση (access):** είναι η πρόσβαση σε έναν κόμβο με σκοπό να εξετασθεί ή να τροποποιηθεί το περιεχόμενό του.
- **Εισαγωγή (insertion):** πρόκειται για τη προσθήκη νέων κόμβων σε μια υπάρχουσα δομή.
- **Διαγραφή (deletion):** αποτελεί το αντίστροφο της εισαγωγής, δηλαδή ένας κόμβος αφαιρείται από μία δομή.
- **Αναζήτηση (searching):** είναι η λειτουργία κατά την οποία προσπελούνται οι κόμβοι μιας δομής, προκειμένου να εντοπιστούν ένας ή περισσότεροι που έχουν μια δεδομένη ιδιότητα.
- **Ταξινόμηση (sorting):** οι κόμβοι μιας δομής διατάσσονται κατά αύξουσα ή φθίνουσα σειρά.
- **Αντιγραφή (copying):** όλοι οι κόμβοι ή μερικοί από τους κόμβους μίας δομής αντιγράφονται σε μία άλλη δομή.
- **Συγχώνευση (merging):** δύο ή περισσότερες δομές συνενώνονται σε μία ενιαία δομή.

- Διαχωρισμός (separation): αποτελεί την αντίστροφη πράξη της συγχώνευσης.

Πλαίσια Ανάπτυξης Web Εφαρμογών σε Java (Web Application Frameworks)

Η ανάπτυξη Web εφαρμογών ακολούθησε μια ταχεία εξέλιξη με αλληπάλληλες αλλαγές στις κυρίαρχες τάσεις. Σχετικά με τις τεχνολογίες Java, προέκυψαν οι τάσεις

- ελαχιστοποίησης της χρήσης «καθαρού» κώδικα (scriptlets) στις Web σελίδες, και
- προώθησης τεχνικών που ευνοούσαν την επαναχρησιμοποίηση του κώδικα.

Η τεχνολογία JSP ευνοεί την πρώτη τάση, ενώ άλλες τεχνολογίες, όπως η Java Beans και οι βιβλιοθήκες ετικετών (Custom JSP Taglibs), ευνοούν τη δεύτερη. Για τον όρο «Πλαίσιο Ανάπτυξης Εφαρμογών» υπάρχουν αρκετοί ορισμοί. Μπορεί να περιγραφεί ως «ένα σύνολο δομών και αντικειμένων που υλοποιεί μια αφηρημένη σχεδίαση, η οποία έχει τη δυνατότητα να αποτελέσει λύση σε ένα σύνολο σχετιζόμενων αναγκών» και «μια επαναχρησιμοποιήσιμη, ημιτελής εφαρμογή που μπορεί εύκολα να εξειδικευτεί και να αποτελέσει τη βάση για ένα εύρος διαφορετικών ολοκληρωμένων εφαρμογών». Το πλαίσιο, δηλαδή, διαφέρει από ένα απλό εργαλείο ή βιβλιοθήκη στο ότι δεν χρησιμοποιείται απλά από μια εφαρμογή, αλλά παρέχει το ίδιο το σκελετό και όλη την υποδομή για τη δημιουργία μιας εφαρμογής. Οι τεχνολογίες Servlet και JSP, με όλες τις επεκτάσεις και εξελίξεις τους, όπως αναλύθηκαν και πιο πάνω, παρέχουν στους προγραμματιστές πολλά πλεονεκτήματα για την ανάπτυξη μικρής κλίμακας εφαρμογών. Για πιο σύνθετες εφαρμογές ωστόσο, είναι πολύ πιθανό να προκύψουν προβλήματα στη σχεδίαση και οργάνωση της λειτουργικότητας, καθώς και δυσκολίες συντήρησης και περιορισμοί στη δυνατότητα αναβάθμισης. Για το λόγο αυτό, δόθηκε ιδιαίτερο βάρος στη δημιουργία και χρήση προτύπων σχεδίασης.

Πρότυπα Σχεδίασης: το Πρότυπο MVC

Ένας αρκετά περιγραφικός ορισμός για το πρότυπο σχεδίασης έχει ο εξής: «Ένα πρότυπο σχεδίασης συστηματικά δίνει όνομα, και προωθεί και εξηγεί μια γενική σχεδίαση που αντιμετωπίζει ένα επαναλαμβανόμενο πρόβλημα σε αντικειμενοστραφή συστήματα. Περιγράφει το πρόβλημα, τη λύση, το πώς εφαρμόζεται η λύση, και τις συνέπειές της. Επίσης, δίνει συμβουλές υλοποίησης και παραδείγματα. Η λύση είναι μία γενική δομή αντικειμένων και κλάσεων που επιλύουν το πρόβλημα, και μπορεί να προσαρμοστεί και να εφαρμοστεί για την επίλυσή μέσα σε ένα συγκεκριμένο πλαίσιο». Με άλλα λόγια, το πρότυπο παρέχει σε όλους όσους εμπλέκονται στην ανάπτυξη μιας εφαρμογής, μια κοινή ορολογία για να περιγράψουν τις λεπτομέρειες της υλοποίησης. Επίσης, παρέχει στους προγραμματιστές παραδείγματα υλοποιήσεων που αποδεδειγμένα επιτυγχάνουν την αντιμετώπιση συγκεκριμένων προβλημάτων και μπορούν να επαναχρησιμοποιηθούν. Στον χώρο των Web εφαρμογών της Java, το πιο δημοφιλές και συχνά χρησιμοποιημένο πρότυπο σχεδίασης είναι το Model-View-Controller (MVC).

Περιγραφή του Προτύπου

Το πρότυπο αυτό, που δημιουργήθηκε αρχικά για να καλύψει εφαρμογές σε προγράμματα πελάτη (client-side), προωθεί το διαχωρισμό του κώδικα σε τρεις καλά διακριτές ενότητες:

1. Model (Μοντέλο): η απεικόνιση των πληροφοριών τις οποίες διαχειρίζεται η εφαρμογή – διαφορετικού τύπου κάθε φορά, ανάλογα με το πεδίο δραστηριοποίησης (domain). Το μοντέλο είναι στην ουσία το κομμάτι εκείνο που ασχολείται με τη λογική της εφαρμογής (application logic). Η λογική της εφαρμογής είναι αυτή που δίνει νόημα σε ανεπεξέργαστα δεδομένα (π.χ. προσδιορίζει ότι ένα σύνολο αριθμών αποτελεί στην πραγματικότητα τα ποσά εσόδων και εξόδων μιας εταιρίας). Πολλές εφαρμογές χρησιμοποιούν έναν μόνιμο μηχανισμό αποθήκευσης, όπως συστήματα βάσεων δεδομένων ή αρχεία, για την αποθήκευση δεδομένων. Το MVC δεν προσδιορίζει ρητά τέτοιους διακριτούς μηχανισμούς διαχείρισης πόρων,

επειδή θεωρείται ότι αποτελούν επί μέρους τμήματα που περικλείονται στο Μοντέλο.

2. View (Όψη): το κομμάτι που ασχολείται με το τι είναι ορατό στον τελικό χρήστη και αναλαμβάνει να δημιουργήσει τα στοιχεία διεπαφής (UI) που θα δώσουν στο χρήστη τη δυνατότητα να αλληλεπιδράσει με το Μοντέλο. Τυπικά, στις Web εφαρμογές, αυτά τα στοιχεία περιλαμβάνουν HTML σελίδες που περιέχουν φόρμες ή εκθέτουν πληροφορία.

3. Controller (Ελεγκτής): ο κώδικας που αναλαμβάνει να διασυνδέσει τα δύο παραπάνω επίπεδα, καθορίζοντας το ποιες λειτουργίες γίνονται στο επίπεδο του Μοντέλου ως απόκριση σε ενέργειες που προκαλεί ο χρήστης μέσω της διεπαφής. Τυπικά, όταν ένας χρήστης υποβάλλει τα στοιχεία μιας φόρμας στο επίπεδο View, ο Controller μεταφέρει τον έλεγχο εκτέλεσης σε κάποια κατάλληλη μέθοδο κάποιας κλάσης του επιπέδου Model που θα οδηγήσει στη μόνιμη αποθήκευσή τους.

Είναι προφανές ότι ο κύριος διαχωρισμός, που επιτυγχάνει το MVC, είναι ανάμεσα στο ορατό στον τελικό χρήστη τμήμα της εφαρμογής και σε αυτό που περιέχει τη λογική της εφαρμογής.

Εκτός από το MVC, στην περίπτωση της Java αναπτύχθηκαν και άλλα πρότυπα σχεδίασης, τα οποία σε πολλές περιπτώσεις απλά προτυποποιούν περαιτέρω κάποια από τα τρία διακριτά τμήματα που ορίζει το MVC. Πολλά από αυτά τα πρότυπα αναπτύχθηκαν για να υποστηρίξουν την τεχνολογία J2EE. Ενδεικτικά:

- Front Controller Servlet Pattern: περιέχει προδιαγραφές για το επίπεδο του Controller, ορίζοντας ότι κάθε ενέργεια του τελικού χρήστη που γίνεται στο επίπεδο View κατευθύνεται σε ένα κεντρικό σημείο ελέγχου και «εισόδου» της εφαρμογής. Έτσι, η πρόσβαση στο Model γίνεται πλήρως κεντρικά και ελεγχόμενα –σε αντιδιαστολή προς αυτό, το MVC αφήνει ανοιχτό το ενδεχόμενο για αποκεντρωμένη «επικοινωνία» ανάμεσα στα επίπεδα.
- Session Façade Pattern: προδιαγράφει το διαχωρισμό του επιπέδου Model σε περαιτέρω τμήματα, όπου κάθε ένα εξ αυτών αφορά σε ένα σύνολο λειτουργιών

που συνθέτουν μια συγκεκριμένη και σχετικά αυτόνομη άποψη (Facade) της λογικής της εφαρμογής (Session Beans).

- Data Access Objects (DAO): προδιαγράφει το διαχωρισμό του μηχανισμού αποθήκευσης –τυπικά μιας βάσης δεδομένων– από τις μεθόδους πρόσβασης στα αποθηκευμένα δεδομένα που χρησιμοποιεί το κομμάτι κώδικα που υλοποιεί τη λογική της εφαρμογής, καθώς και την απόκρυψη των λεπτομερειών του μηχανισμού αποθήκευσης.

Το Πρότυπο JSP-2 (MVC-2)

Το πρότυπο JSP-2 αποτελεί την προσπάθεια της Sun να ενσωματώσει το MVC στις Web τεχνολογίες της. Αποτελείται από μια σειρά απλών προδιαγραφών ή οδηγιών για το πώς θα πρέπει να σχεδιαστεί μια εφαρμογή. Ένα τυπικό σενάριο, σύμφωνα με το μοντέλο, έχει ως εξής:

- α. Ένα αίτημα (request) του πελάτη παραλαμβάνεται πάντα από ένα Servlet ελεγκτή (controller).
- β. Το Servlet, ανάλογα με τη φύση του αιτήματος, προωθεί την εκτέλεση σε μεθόδους πρόσβασης σε δεδομένα ή σε βοηθητικές μεθόδους.
- γ. Το ίδιο Servlet αποφασίζει να περάσει τον έλεγχο στο κατάλληλο JSP.
- δ. Μέσω του JSP, παρουσιάζεται στον πελάτη το τελικό αποτέλεσμα.

Με βάση το παραπάνω μοντέλο, δημιουργήθηκε ένα πλήθος από Frameworks, που έγιναν λιγότερο ή περισσότερο δημοφιλή, και που καλύπτουν το σύνολο ή τμήματα του MVC. Μια συνοπτική κατηγοριοποίησή τους μπορεί να γίνει, με βάση το τελευταίο κριτήριο, σε:

- Frameworks που καλύπτουν μόνο το επίπεδο View, αντικαθιστώντας τυπικά τα JSP με άλλη scripting γλώσσα (Velocity, Freemarker, WebMacro, κ.α.).
- Frameworks που αφορούν μόνο στο επίπεδο Model, παρέχοντας μηχανισμό ObjectRelational Mapping (OR) και καθιστώντας μια εφαρμογή φορητή και συμβατή

με διαφορετικά συστήματα βάσεων δεδομένων. Το πιο δημοφιλές είναι το Hibernate.

- Frameworks που καλύπτουν τα επίπεδα View και Controller. Το πιο δημοφιλές είναι το Apache Struts, ίσως αυτό που έχει χρησιμοποιηθεί πιο ευρέως από οποιοδήποτε άλλο για την ανάπτυξη Java Web εφαρμογών.
- Frameworks που καλύπτουν όλα τα επίπεδα του MVC, όπως τα Turbine και Expresso.
- Frameworks που καλύπτουν όλα τα επίπεδα του MVC και επιπλέον το επεκτείνουν. Τα πιο δημοφιλή είναι τα Struts 2, Spring Framework, και Java Server Faces (JSF) Framework

Apache Struts & Struts 2

Το Apache Struts Project ξεκίνησε το Μάιο του 2000 από τον Craig R. McClanahan με στόχο να παρέχει ένα τυποποιημένο πλαίσιο που να είναι συμβατό με το πρότυπο MVC στην κοινότητα της Java. Τον Ιούλιο του 2001, κυκλοφόρησε η έκδοση 1.0. Το Framework Struts 2 ήταν αρχικά γνωστό ως WebWork 2. Αφού αναπτύχθηκαν ξεχωριστά για αρκετά χρόνια, το WebWork και το Struts ενώθηκαν το 2008 για τη δημιουργία του Struts 2, με πιο πρόσφατη έκδοση την 2.2.1. Το Struts 2 διαφέρει σε σημαντικά σημεία από το Apache Struts.

Το Struts 2 Framework ακολουθεί τις γενικές προδιαγραφές για όλα τα επίπεδα του MVC. Το επίπεδο Model αντιστοιχεί, από πλευράς υλοποίησης, σε ειδικά αντικείμενα του Struts 2 που ονομάζονται Actions. Το επίπεδο Controller υλοποιείται μέσω ενός ειδικού αντικειμένου με όνομα FilterDispatcher. Τέλος, για το επίπεδο View μπορούν να χρησιμοποιηθούν διάφορες τεχνολογίες, όπως οι JSP, Velocity, Freemarker, με πιο συνηθισμένη την JSP.

Όπως στην πλειοψηφία των σύγχρονων Java Web Application Frameworks, έτσι και στο Struts 2 μεγάλο μέρος της ανάπτυξης μιας εφαρμογής περιλαμβάνεται σε ειδικά διαμορφωμένα XML αρχεία.

Τα ειδικά διαμορφωμένα αυτά αρχεία ρυθμίσεων (configuration files) εμπεριέχουν μεγάλο μέρος της λογικής της εφαρμογής σε προτυποποιημένη μορφή.

Το πλεονέκτημα είναι ότι απαλλάσσουν τον προγραμματιστή από την ανάγκη να γράψει κώδικα για πολλές κοινές χρήσιμες λειτουργίες, τις οποίες αναλαμβάνει να εκτελέσει το Framework χρησιμοποιώντας τις ρυθμίσεις που έχουν δηλωθεί στα αντίστοιχα XML αρχεία. Επιπλέον, βοηθάνε στη διατήρηση της καθαρότητας του κώδικα.

Κάποια βασικά αντικείμενα και οντότητες στο μηχανισμό των Struts 2 είναι τα παρακάτω:

- **FilterDispatcher:** αντικείμενο που περιέχει τον κώδικα που διαχειρίζεται τα αιτήματα (requests) που λαμβάνει η εφαρμογή.
- **Action:** βασικό αντικείμενο που αναλαμβάνει την εξυπηρέτηση ενός συγκεκριμένου αιτήματος. Όλες τα αιτήματα που μπορεί να εξυπηρετήσει η εφαρμογή δηλώνονται σε αρχεία XML σαν Actions.
- **Interceptor:** τύπος επαναχρησιμοποιήσιμου αντικειμένου που εκτελεί συνήθως κάποια ειδικού τύπου λειτουργία πάνω σε ένα αίτημα. Για παράδειγμα, ένα αντικείμενο τύπου Interceptor μπορεί να ελέγξει την εγκυρότητα των στοιχείων ενός αιτήματος υποβολής φόρμας και να το απορρίψει, αν τα στοιχεία δεν υπακούουν σε συγκεκριμένους κανόνες που ορίζονται από τη λογική της εφαρμογής.
- **Result:** τύπος αντικειμένου που συνδέεται με ένα Action. Ένα σύνολο από αντικείμενα τύπου Result εμπεριέχει όλα τα πιθανά αποτελέσματα που μπορεί να προκύψουν κατά την εκτέλεση ενός Action, δηλαδή τις πιθανές σελίδες που μπορεί να παρουσιαστούν σε ένα χρήστη μετά από κάποιο αίτημά του. Το σύνολο αυτό δηλώνεται σε αρχεία XML για κάθε Action.

Μια ενδεικτική δήλωση ενός Action στο Struts 2 έχει τη μορφή:

```
<action name="Logon" class="mailreader.Logon">
    <result name="input">/pages/Logon.jsp</result>
    <result name="cancel">Welcome</result>
    <result type="redirect-action">MainMenu</result>
</action>
```

Το παραπάνω δείχνει ότι ένα αίτημα του τύπου αυτού μπορεί να οδηγήσει σε τρία πιθανά αποτελέσματα, δηλαδή σε τρεις πιθανές σελίδες.

Ένα τυπικό σενάριο για το πώς διαχειρίζεται το Struts 2 τη λήψη ενός αιτήματος από το χρήστη έχει το εξής:

1. Το αντικείμενο FilterDispatcher λαμβάνει ένα αίτημα για εξυπηρέτηση και αποφασίζει ποια ενέργεια θα εκτελεστεί, δηλαδή ποιο από τα αντικείμενα τύπου Action, εξ αυτών που έχουν δηλωθεί, πρέπει να δημιουργηθεί.
2. Το Framework δημιουργεί ένα αντικείμενο τύπου ActionInvocation για το αντίστοιχο Action και εκτελεί την (προτυποποιημένη) μέθοδο invoke() για να ξεκινήσει η εκτέλεση της ενέργειας.
3. Πριν την εκτέλεση, ελέγχεται αν στο συγκεκριμένο αντικείμενο Action έχουν ανατεθεί, μέσω αρχείων ρυθμίσεων XML, αντικείμενα Interceptors. Αν ναι, καλείται η μέθοδος intercept() του πρώτου από τους Interceptors που έχουν δηλωθεί να εκτελούνται πριν το Action, με τη σειρά με την οποία έχουν δηλωθεί, και μεταφέρεται ο έλεγχος εκτέλεσης εκεί. Η μέθοδος αυτή έχει τη δυνατότητα να τερματίσει την εκτέλεση του Action σε περίπτωση αποτυχίας, ενώ σε περίπτωση επιτυχίας καλεί πάλι τη μέθοδο invoke() του αντικειμένου ActionInvocation, η οποία διαδοχικά επαναλαμβάνει την διαδικασία για όλους τους Interceptors που έχουν δηλωθεί.
4. Μετά την επιτυχή ολοκλήρωση των παραπάνω, η μέθοδος invoke() του αντικειμένου ActionInvocation εκτελείται πάλι και παράγει το αποτέλεσμα του

αιτήματος. Για κάθε Action, όπως είδαμε, τα αποτελέσματα που είναι δυνατό να παραχθούν είναι ήδη προδιαγεγραμμένα.

5. Μια παρόμοια διαδικασία με αυτή του βήματος 3 εκτελείται για τους Interceptors που έχει καθοριστεί να εκτελεστούν μετά την εκτέλεση του Action.
6. Το αποτέλεσμα επιστρέφεται ανάλογα με την τεχνολογία που χρησιμοποιείται για το επίπεδο View και τη φύση του αιτήματος, σαν σελίδα JSP, σελίδα HTML, έγγραφο PDF, κ.α.

Κάποια άλλα ενδιαφέροντα σημεία του Framework:

- Χρησιμοποιείται μια καθολική δομή αποθήκευσης πληροφοριών που σχετίζεται με την εξυπηρέτηση ενός αιτήματος και ονομάζεται ActionContext. Η δομή αυτή περιέχει, εκτός από τα αντικείμενα που σχετίζονται με το Servlet API (request, session κ.α.), το επιπλέον αντικείμενο ValueStack, χρήσιμο για την αποθήκευση πληροφοριών εκτέλεσης και κατάστασης του αιτήματος.
- Υποστηρίζεται η χρήση της γλώσσας Object Graph Navigation Language (OGNL) σε συνδυασμό με την παραπάνω δομή, για προσπέλαση των αντικειμένων του ActionContext. Η γλώσσα OGNL μπορεί να χρησιμοποιηθεί αντί για την γλώσσα EL της JSP.
- Παρέχεται μια εκτενής συλλογή βιβλιοθηκών ετικετών (Struts Tags) που, όπως περιγράφηκε στην αντίστοιχη ενότητα, αποτελούν επαναχρησιμοποιήσιμα κομμάτια κώδικα.
- Εκτός από το βασικό μηχανισμό, το Framework περιλαμβάνει πολλά ακόμα εργαλεία – plugins– που το καθιστούν ακόμα πιο χρήσιμο ως προς την ανάπτυξη εφαρμογών.

Πολλά από αυτά αφορούν στη συνεργασία του Framework με άλλα γνωστά Frameworks και τεχνολογίες, ενώ κάποια άλλα αφορούν στην υποστήριξη άλλων τεχνολογιών, όπως π.χ. το Portlet Plugin υποστηρίζει ενσωμάτωση της εφαρμογής σε Java Portals. Μια ενδεικτική λίστα είναι η παρακάτω:

<ul style="list-style-type: none"> • Codebehind Plugin • Config Browser Plugin • JasperReports Plugin • JFreeChart Plugin • JSF Plugin • Plexus Plugin • Portlet Plugin • DWR Plugin 	<ul style="list-style-type: none"> • REST Plugin(2.1.1+) • SiteGraph Plugin • Sitemesh Plugin • Spring Plugin • Struts 1 Plugin • Tiles Plugin • JUnit Plugin • TestNG Plugin
--	---

Spring Framework

Το Spring Framework θεωρείται από τα πλέον δημοφιλή Πλαίσια Ανάπτυξης Εφαρμογών σε Java. Πολλές φορές αναφέρεται και ως «πλατφόρμα», λόγω του ότι ο πυρήνας του (Core Spring) αποτελεί ένα «ελαφρύ» περιβάλλον που υποστηρίζει τη χρήση και ενσωμάτωση μιας σειράς εργαλείων που παρέχουν ένα πλήθος χρήσιμων λειτουργικοτήτων. Επίσης, σε αυτή την πλατφόρμα υποστηρίζεται η προαιρετική ενσωμάτωση άλλων Frameworks που έχουν αναπτυχθεί στα πλαίσια του προϊόντος, όπως το Spring Web MVC Framework, το Remote Access Framework, και το Batch Framework.

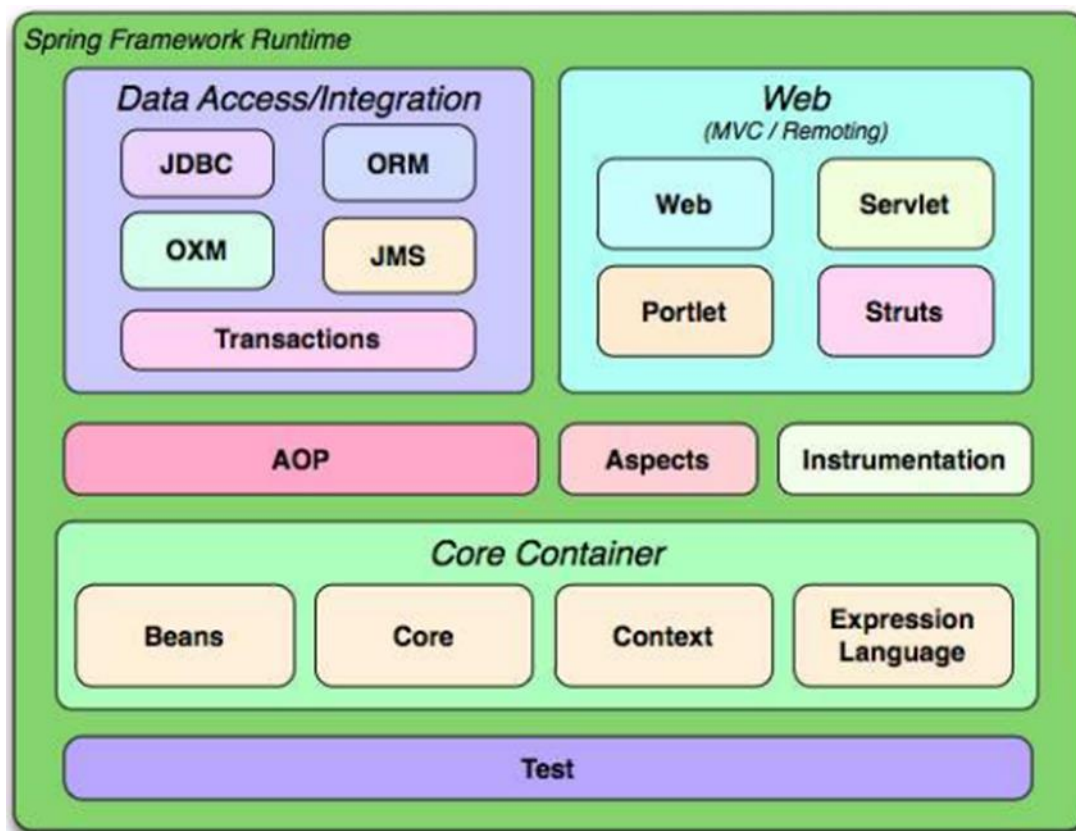
Η πρώτη έκδοση γράφτηκε από τον Rod Johnson τον Οκτώβριο του 2002. Το Framework έγινε διαθέσιμο για πρώτη φορά, ως προϊόν ανοιχτού κώδικα με άδεια Apache 2.0, τον Ιούνιο του 2003.

Η πρώτη έκδοση ορόσημο, η 1.0, κυκλοφόρησε το Μάρτιο του 2004, και με περαιτέρω βελτιώσεις το Σεπτέμβριο του 2004 και το Μάρτιο του 2005. Το Δεκέμβριο του 2009, κυκλοφόρησε η έκδοση 3.0 GA (Γενική Διαθεσιμότητα), ενώ η τρέχουσα έκδοση είναι η 3.0.4.

Περιγραφή

Το Spring Framework περιλαμβάνει μια σειρά από εργαλεία που είναι οργανωμένα σε περίπου 20 ενότητες (modules). Αυτές οι ενότητες είναι περαιτέρω οργανωμένες

σε ομάδες: Πλαίσιο Πυρήνα (Core Container), Προσπέλαση/ Ενσωμάτωση Δεδομένων (Data Access/Integration), Web, AOP (Aspect Oriented Programming), Instrumentation, και Test – όπως φαίνεται στο παρακάτω σχήμα:



Η αρχιτεκτονική του Spring Framework

Πλαίσιο Πυρήνα

Το πλαίσιο πυρήνα περιέχει τα modules Core, Beans, Context, και Expression Language. Τα modules «Core» και «Beans» αποτελούν τη βασική υποδομή του Framework, συμπεριλαμβανομένων των μηχανισμών που υλοποιούν ένα πολύ σημαντικό χαρακτηριστικό της λειτουργίας του Spring: την Αντιστροφή του Ελέγχου (Inversion of Control – IoC).

Η ιδέα του IoC είναι κεντρικό στοιχείο του Framework και παρέχει ένα ισχυρό εργαλείο διαχείρισης Java αντικειμένων. Ο μηχανισμός που υλοποιεί το IoC είναι υπεύθυνος για τη διαχείριση του κύκλου ζωής των αντικειμένων: δημιουργία, αρχικοποίηση, αλληλεξαρτήσεις. Τα αντικείμενα που είναι διαχειρίσιμα με αυτό τον τρόπο ονομάζονται Managed Beans και η δήλωση τους, καθώς και όλες οι

πληροφορίες που χρειάζεται το πλαίσιο για την διαχείριση τους, γράφονται σε ειδικά διαμορφωμένα αρχεία ρυθμίσεων XML.

Τα αντικείμενα Managed Beans είναι προσβάσιμα μέσω μιας διαδικασίας που ονομάζεται Dependency Lookup ή Dependency Injection.

Η πρώτη αφορά στη λειτουργία ανάκτησης ενός αντικειμένου τύπου Managed Bean βάσει ονόματος.

Η δεύτερη αφορά στη συνολική διαδικασία κατά την οποία το IoC πλαίσιο αναλαμβάνει να παρέχει αντικείμενα Managed Beans σε αντικείμενα και μεθόδους που τα χρειάζονται.

Η χρήση του μηχανισμού IoC δεν είναι υποχρεωτική στο Spring Framework, αλλά είναι πολύ συχνή επειδή παρέχει έναν συνεπή και αποτελεσματικό τρόπο ανάπτυξης εφαρμογών εύκολα συντηρήσιμων, με μεγάλες δυνατότητες ενσωμάτωσης νέας λειτουργικότητας και εργαλείων.

Πρακτικά αυτό σημαίνει ότι μια ολόκληρη ενότητα της εφαρμογής που αντιστοιχεί σε ένα Managed Bean μπορεί να τροποποιηθεί ή να αντικατασταθεί χωρίς να αλλάξει τίποτα στον υπόλοιπο κώδικα, επειδή το Dependency Injection εξασφαλίζει ότι οι λεπτομέρειες υλοποίησης ενός Managed Bean αποκρύπτονται από τα υπόλοιπα τμήματα του κώδικα.

Το module «Context» χρησιμοποιεί την υποδομή του πλαισίου για να δημιουργήσει ένα μηχανισμό για αποθήκευση και ανάκτηση αντικειμένων βάσει ονόματος (λειτουργεί παρόμοια με τον μηχανισμό JNDI της Java). Επίσης, υποστηρίζει στοιχεία πολυγλωσσικότητας της εφαρμογής. Ένα ειδικό αντικείμενο τύπου ApplicationContext αποτελεί τον πυρήνα του συγκεκριμένου module.

Το module «Expression Language» αποτελεί επέκταση της γλώσσας EL του JSP 2.1 και παρέχει ένα ισχυρό εργαλείο διαχείρισης και προσπέλασης ιεραρχικά

δομημένων αντικειμένων. Υποστηρίζει, όπως η EL, την ανάκτηση ιδιοτήτων αντικειμένων, τους λογικούς και αριθμητικούς τελεστές, τη διαχείριση πινάκων και συλλογών αντικειμένων, και την ανάκτηση αντικειμένων μέσω του μηχανισμού IoC.

Προσπέλαση/Ενσωμάτωση δεδομένων (Data Access/Integration)

Η ομάδα αυτή περιλαμβάνει τα modules JDBC, ORM, OXM, JMS, και Transaction.

- Το module «JDBC» διευκολύνει τη χρήση της ομώνυμης βιβλιοθήκης, που αφορά στην πρόσβαση σε συστήματα βάσης δεδομένων, παρέχοντας ένα επίπεδο αφαίρεσης σε αυτή και ελαχιστοποιώντας την εξάρτηση του κώδικα από τα ειδικά χαρακτηριστικά του εκάστοτε συστήματος βάσης δεδομένων που χρησιμοποιείται.
- Το module «ORM» παρέχει δυνατότητα ενσωμάτωσης για δημοφιλή εργαλεία object-relational mapping, όπως τα JPA, JDO, Hibernate, και iBatis.
- Το module «OXM» παρέχει δυνατότητα ενσωμάτωσης για δημοφιλείς υλοποιήσεις της τεχνολογίας Object/XML mapping, όπως οι βιβλιοθήκες JAXB, Castor, XMLBeans, JiBX, και Xstream.
- Το module «JMS» (Java Messaging Service) υποστηρίζει την αντίστοιχη τεχνολογία της Java για ανταλλαγή μηνυμάτων.
- Το module «Transaction» υποστηρίζει τη διαχείριση και συνεπή εκτέλεση λειτουργιών που αποτελούνται από υποενέργειες, οι οποίες εκτελούνται στα πλαίσια διαφορετικών modules.

Aspect Oriented Programming, Ενorchήστρωση (Instrumentation), & Test

Η έννοια του Aspect-Oriented Programming (AOP) αποτελεί μια ενισχυμένη εκδοχή του μοντέλου του αντικειμενοστραφή προγραμματισμού και εισάγει την έννοια της Άποψης (Aspect). Ορίζει πρακτικά ότι λειτουργίες που μπορεί να αφορούν και να εφαρμόζονται σε πολλούς διαφορετικούς τύπους αντικειμένων θα πρέπει να διαχωριστούν από τον κώδικα των ίδιων των αντικειμένων και να υλοποιούνται ανεξάρτητα. Στην ουσία αυτό προσδίδει σε ένα αντικείμενο, πέρα από ιδιότητες και

λειτουργίες, επιπλέον στοιχεία συμπεριφοράς. Το module αυτό του Spring παρέχει πλήρη υποστήριξη στην ανάπτυξη Aspect oriented κώδικα. Το module «Ενορχήστρωσης» παρέχει βοηθητικές λειτουργίες που αφορούν στην εγκατάσταση αντικειμένων στο περιβάλλον του διακομιστή και του Servlet Container. Το module «Test» χρησιμοποιείται για τις δοκιμές σωστής λειτουργίας (testing) μιας εφαρμογής στα πλαίσια του Spring Framework, παρέχοντας ενσωμάτωση του γνωστού εργαλείου JUnit.

Web

Η ομάδα Web περιλαμβάνει όλη την απαραίτητη υποδομή για υποστήριξη όλων των βημάτων ανάπτυξης μιας Web Εφαρμογής σε περιβάλλον Java & Java Servlet, μέσω των modules «Web-Servlet» και «Web Portlet». Διευκολύνει τη δημιουργία εφαρμογών συμβατών με το πρότυπο MVC, καθώς και τη δυνατότητα ανάπτυξης εφαρμογών σύμφωνα με το Portlet Specification της Java (JSR-268), προορισμένων δηλαδή να ενσωματωθούν σε περιβάλλον Java Portal.

Web MVC Framework

Στα πλαίσια της συνεχούς ανάπτυξης και εξέλιξης του Spring Framework δημιουργήθηκε ένα νέο, ξεχωριστό Framework ανάπτυξης Web Εφαρμογών που ονομάστηκε Web MVC Framework. Η δημιουργία του δεν είχε σχεδιαστεί από την αρχή, αλλά προέκυψε λόγω της αίσθησης των δημιουργών του Spring ότι τα υπάρχοντα Frameworks, και ειδικά το δημοφιλές Apache Struts, ήταν ανεπαρκή για να υποστηρίξουν την ανάπτυξη μεγάλης κλίμακας ολοκληρωμένων εφαρμογών. Είχαν αδυναμίες όσον αφορά στο σαφή διαχωρισμό των επιπέδων παρουσίασης του Μοντέλου από το επίπεδο Controller του MVC. Το Web MVC δεν είναι κομμάτι του Spring Framework, με την έννοια ότι ο προγραμματιστής μπορεί να αναπτύξει μια Web εφαρμογή στην πλατφόρμα Spring χρησιμοποιώντας κάποιο άλλο Java Framework. Παρόλα αυτά, τα πλεονεκτήματα που προσφέρει το καθιστούν ιδανικό εργαλείο για ανάπτυξη.

Το Web MVC, όπως και το Struts, έχει χτιστεί με άξονα τη λειτουργία εξυπηρέτησης ενός αιτήματος από τον πελάτη (request-based system). Κάποια βασικά αντικείμενα που χρησιμοποιεί είναι τα εξής:

- **DispatcherServlet:** το Servlet που αποτελεί σημείο εισόδου για όλα τα εισερχόμενα αιτήματα. Αναλαμβάνει να μεταφέρει τον έλεγχο εκτέλεσης εκεί όπου χρειάζεται ανάλογα με το είδος του αιτήματος.
- **HandlerMapping:** η κλάση που είναι υπεύθυνη για την αντιστοίχιση ενός αιτήματος στο αντικείμενο που θα τη διαχειριστεί.
- **HandlerAdapter:** τα αντικείμενα που χειρίζονται τα διαφορετικά είδη αιτημάτων.
- **Command Controller:** τα αντικείμενα αυτού του τύπου λειτουργούν ως πύλη ανάμεσα στα επίπεδα Λογικής και Παρουσίασης της εφαρμογής, εξασφαλίζοντας έτσι ότι η συγκεκριμένη ενέργεια του πελάτη θα έχει το επιθυμητό αποτέλεσμα και στα δύο επίπεδα.
- **ModelAndView:** το αντικείμενο που θα επιστραφεί από ένα Command Controller και περιέχει πληροφορία που αφορά τόσο το επίπεδο παρουσίασης (View) όσο και το επίπεδο του μοντέλου (Model), π.χ. μια φόρμα και τα δεδομένα που περιέχει αυτή.
- **View:** το αντικείμενο που θα επιστραφεί ως απάντηση στο αίτημα του πελάτη, τυπικά σε μορφή JSP.
- **ViewResolver:** αναλαμβάνει την επιλογή του αντικειμένου View που θα επιστραφεί ως απάντηση και τον τρόπο που αυτό θα παρουσιαστεί, π.χ. ως JSP σελίδα, ως αρχείο Velocity, κ.α.
- **HandlerInterceptor:** τα αντικείμενα που μεσολαβούν κατά τη λήψη ενός εισερχόμενου αιτήματος και εκτελούνται πριν τον κώδικα του αντίστοιχου HandlerAdapter. Η λειτουργία τους είναι αντίστοιχη με αυτή των φίλτρων στο Servlet API.

Ένα τυπικό σενάριο εξυπηρέτησης αιτήματος στο Web MVC:

1. Ο πελάτης αποστέλλει ένα αίτημα προς εξυπηρέτηση.
2. Το DispatcherServlet βρίσκει τα κατάλληλα αντικείμενα HandlerMapping για το συγκεκριμένο αίτημα.
3. Το αντικείμενο HandlerMapping υποδεικνύει το αντίστοιχο αντικείμενο HandlerAdapter που πρέπει να διαχειριστεί το αίτημα.
4. Εάν έχει καθοριστεί η χρήση HandlerInterceptors για αυτό το αίτημα, αυτά εκτελούνται.
5. Το αντικείμενο HandlerAdapter που επιλέχθηκε στο βήμα 3 μεταφέρει τον έλεγχο της εκτέλεσης στο κατάλληλο αντικείμενο Command Controller.
6. Ο Controller επεξεργάζεται τα στοιχεία του αιτήματος, επιστρέφοντας ένα ειδικό αντικείμενο τύπου ModelAndView στο DispatcherServlet.
7. Το DispatcherServlet χρησιμοποιεί τα διαθέσιμα αντικείμενα τύπου View Resolver για να καταλήξει στο αντικείμενο View που θα επιστραφεί ως απάντηση.
8. Η απάντηση επιστρέφεται στον πελάτη.

Η υλοποίηση των παραπάνω στα πλαίσια του Web MVC αφενός διευκολύνει την ανάπτυξη ολοκληρωμένων εφαρμογών με συμπαγή σχεδίαση, αφετέρου δίνει μεγάλα περιθώρια ευελιξίας στο σχεδιαστή της εφαρμογής, σε πολλά επίπεδα.

Κάποια από τα πιο σημαντικά πλεονεκτήματα του Framework είναι τα εξής:

- Ξεκάθαρος διαχωρισμός του ρόλου, όσο αφορά στις φάσεις εξυπηρέτησης ενός αιτήματος. Τα αντικείμενα τύπου DispatcherServlet, ViewResolver, και CommandController έχουν εξειδικευμένες, μη επικαλυπτόμενες λειτουργίες να επιτελέσουν.
- Ισχυρός μηχανισμός δήλωσης και δημιουργίας των αντικειμένων σε XML αρχεία, με αξιοποίηση της υποδομής του Spring Framework.

- Ευελιξία στο επίπεδο του αντικειμένου Controller, το οποίο μπορεί να διαμορφωθεί για να υποστηρίξει πολλές διαφορετικές δομές και λειτουργικότητες στα πλαίσια ενός αιτήματος (φόρμες, απλές εντολές, αλυσίδα εντολών, ή μια τελείως νέα δομή).
- Ευελιξία στο επίπεδο View: υποστηρίζονται οι τεχνολογίες JSP, Velocity, Freemarker, JSTL, ενώ είναι δυνατόν να ενσωματωθούν νέες τεχνολογίες και να οριστούν νέοι τύποι αντικειμένων View εύκολα από τον προγραμματιστή.
- Υποστήριξη της βιβλιοθήκης ετικετών Spring Tags που παρέχει χρήσιμες λειτουργίες μέσα από επαναχρησιμοποίηση κώδικα, καθώς και μια βιβλιοθήκη ετικετών ειδικά για την εύκολη διαχείριση Web φορμών.

Τέλος, είναι προφανές ότι το πιο σημαντικό πλεονέκτημα του Framework είναι η ενσωμάτωσή του στο Spring Framework και η πρόσβασή του σε όλες τις δυνατότητες των διαθέσιμων modules.

Μια σημαντική επέκταση του Web MVC Framework επιτελέστηκε με την προσθήκη τριών ακόμα εργαλείων:

- Spring Faces: εργαλείο που δίνει τη δυνατότητα να χρησιμοποιηθεί, σε επίπεδο παρουσίασης του περιεχομένου, η τεχνολογία Java Server Faces (JSF) αντί της JSP ή των άλλων εναλλακτικών.
- Spring Javascript: ενσωματωμένη βιβλιοθήκη JavaScript, βασισμένη στη γνωστή βιβλιοθήκη Dojo, που παρέχει δυνατότητα υποστήριξης της τεχνολογίας JavaScript με ασφαλή και αποδοτικό τρόπο, και ενισχύει έτσι την διεπαφή χρήστη.
- Spring Web Flow: εργαλείο που δίνει τη δυνατότητα, με τη χρήση ειδικής γλώσσας, να μοντελοποιηθούν περιπτώσεις αλληλεπίδρασης με το χρήστη που περιλαμβάνουν πολλαπλά διαδοχικά αιτήματα. Αυτό γίνεται μέσω της δημιουργίας σύνθετων λειτουργιών ροής εργασιών (workflow).

Εκπαιδευτική Υποενότητα 7.3 Εισαγωγή στον αντικειμενοστραφή προγραμματισμό (JAVA)

Java

Η Java είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου που εισήχθη από τη Sun Microsystems τον Μάιο του 1995. Η Java μπορεί να εκτελεστεί σε πολλές πλατφόρμες, όπως Windows, Mac OS και διάφορες άλλες εκδόσεις UNIX του συστήματος. Το μεγαλύτερο μέρος του κώδικα για το λειτουργικό σύστημα Android για κινητά είναι προγραμματισμένο στη γλώσσα προγραμματισμού.

Χαρακτηριστικά της Java

Τα πιο βασικά χαρακτηριστικά της Java είναι παρακάτω:

- Είναι σχετικά απλή: Το συγκεκριμένο χαρακτηριστικό αναφέρεται σε πολλές πηγές σχετικές με τη γλώσσα Java, αλλά είναι καθαρά υποκειμενικό. Η Java είναι όσο απλή μπορεί να είναι μία γλώσσα προγραμματισμού. Συγκρινόμενη με τη C++, είναι απλούστερη μιας και έχει εξαλειφθεί η χρήση δεικτών (pointers), ενώ η διαχείριση της μνήμης γίνεται από την ίδια τη γλώσσα.
- Είναι μεταγλωττιζόμενη (compiled) και ερμηνευμένη (interpreted): Σε αντίθεση με τις C/C++ που είναι compiled γλώσσες, η διαδικασία παραγωγής εκτελέσιμου κώδικα στη Java κάνει χρήση και των δύο αυτών τεχνικών.
- Είναι αμιγώς αντικειμενοστραφής (pure OOD): Η Java υποστηρίζει αποκλειστικά το μοντέλο αντικειμενοστραφούς προγραμματισμού (object-oriented paradigm) και δεν υπάρχει δυνατότητα χρήσης της σύμφωνα με κάποιο άλλο μοντέλο (π.χ. διαδικαστικό).
- Είναι φορητή σε επίπεδο μεταγλωττισμένου κώδικα: Ένα από τα πιο ισχυρά χαρακτηριστικά της Java. Αυτό πρακτικά σημαίνει πως ένας προγραμματιστής μπορεί να γράψει και να μεταγλωττίσει ένα πρόγραμμα π.χ. σε Windows και στη συνέχεια να πάρει το αρχείο που παράχθηκε από τη μεταγλώττιση και να το τρέξει σε ένα μηχάνημα Unix χωρίς καμία αλλαγή. Απαραίτητη προϋπόθεση είναι στο μηχάνημα αυτό (Unix) να είναι εγκατεστημένο το αντίστοιχο JRE.

- Κάνει αυστηρό έλεγχο τύπων (strongly typed): Η Java απαιτεί από τον προγραμματιστή να κάνει σωστή χρήση των τύπων (περισσότερα για τους τύπους στην ενότητα 2) και δεν επιτρέπει αυθαίρετες μετατροπές όπως οι C/C++.
- Είναι γλώσσα υψηλού επιπέδου: Η εκμάθηση της Java και η σύνταξη κώδικα είναι σχετικά απλή μιας και η γλώσσα κάνει χρήση λέξεων που βρίσκονται πιο κοντά στη φυσική γλώσσα (Αγγλικά) παρά στη γλώσσα μηχανής.
- Παρέχει υψηλό επίπεδο ασφάλειας: Η εκτέλεση προγραμμάτων ελέγχεται από μηχανισμούς ασφαλείας που αποτρέπουν την κακόβουλου κώδικα.
- Υποστηρίζει πολυμέσα: Η Java είναι από τις ελάχιστες γλώσσες της κατηγορίας που παρέχουν έμφυτη υποστήριξη για την ανάπτυξη πολυμεσικών (multimedia) εφαρμογών.
- Είναι κατάλληλη για προγραμματισμό δικτυακών εφαρμογών: Η Java διευκολύνει την υλοποίηση τόσο δικτυακών (network) όσο και διαδικτυακών (web) εφαρμογών.
- Υποστηρίζει πολυνηματική επεξεργασία (multi-threaded processing): Και στην περίπτωση αυτή, η Java είναι μία από τις ελάχιστες γλώσσες της κατηγορίας της που παρέχει έμφυτη υποστήριξη για την ανάπτυξη multi-threaded εφαρμογών.
- Κάνει αυτόματη διαχείριση μνήμης: Στη Java, η διαχείριση της μνήμης ελέγχεται αποκλειστικά από αυτήν μέσω ενός υποπρογράμματος που ονομάζεται garbage collector (αποκομιστής απορριμάτων) και ο προγραμματιστής δεν εμπλέκεται ποτέ στη διαδικασία αυτή.
- Είναι δυναμική: Προσαρμόζεται εύκολα σε διαφορετικά περιβάλλοντα και απαιτήσεις και είναι ιδανική για τη διασύνδεση και επικοινωνία ετερογενών συστημάτων. Η Java ενημερώνεται συνεχώς ενσωματώνοντας και υποστηρίζοντας τις τελευταίες τεχνολογικές εξελίξεις.
- Κάνει αποκλειστική χρήση της δυναμικής διασύνδεσης (dynamic binding): Στη Java η διασύνδεση των δεδομένων και των μεθόδων που αυτά υποστηρίζουν γίνεται κατά την εκτέλεση του προγράμματος (run-time).

Η φιλοσοφία της γλώσσας, όσον αφορά το προγραμματιστικό μοντέλο που υιοθετεί είναι πως ο προγραμματιστής θα πρέπει να προστατεύεται από τη γλώσσα και τα προγράμματα να είναι ασφαλή και αξιόπιστα.



Αρχιτεκτονική της Java

Για να εκτελέσουμε ένα πρόγραμμα γραμμένο σε Java σε κάποιον υπολογιστή, είναι απαραίτητο να είναι εγκατεστημένο σε αυτόν το αντίστοιχο JRE (Java Runtime Environment) για τον επεξεργαστή που διαθέτει και το λειτουργικό σύστημα που χρησιμοποιεί. Τα πιο σημαντικά κομμάτια της αρχιτεκτονικής της Java είναι τα παρακάτω:

- ✓ Java Virtual Machine (JVM): Η εικονική μηχανή (virtual machine) είναι το πιο σημαντικό συστατικό της αρχιτεκτονικής, αφού είναι υπεύθυνη για την εκτέλεση του ενδιάμεσου κώδικα (bytecode) και τη μετατροπή του στη

γλώσσα που “καταλαβαίνει” το λειτουργικό σύστημα και ο επεξεργαστής του μηχανήματος. Ορισμένες από τις πολλές λειτουργίες του JVM, είναι το φόρτωμα των κλάσεων στη μνήμη από τον class loader, ο έλεγχος εγκυρότητας του bytecode και η αποτροπή εκτέλεσης κακόβουλου κώδικα από τον bytecode verifier, η διαχείριση της μνήμης, η λειτουργία του μεταγλωττιστή τελευταίας στιγμής (JIT compiler), ο αυτόματος χειρισμός εξαιρέσεων.

- ✓ Java SE API (Application Programming Interface): Ένα σετ κλάσεις και interfaces που παρέχουν όλη τη λειτουργικότητα της γλώσσας.
- ✓ Deployment Technologies: Τεχνολογίες που διευκολύνουν το deployment (εγκατάσταση και εκτέλεση) εφαρμογών java είτε αυτόνομα είτε μέσα από κάποιον browser (π.χ εκτέλεση κάποιου applet).

Tools: Ένα σετ από εργαλεία για τον προγραμματιστή που περιέχονται μόνο στο JDK.

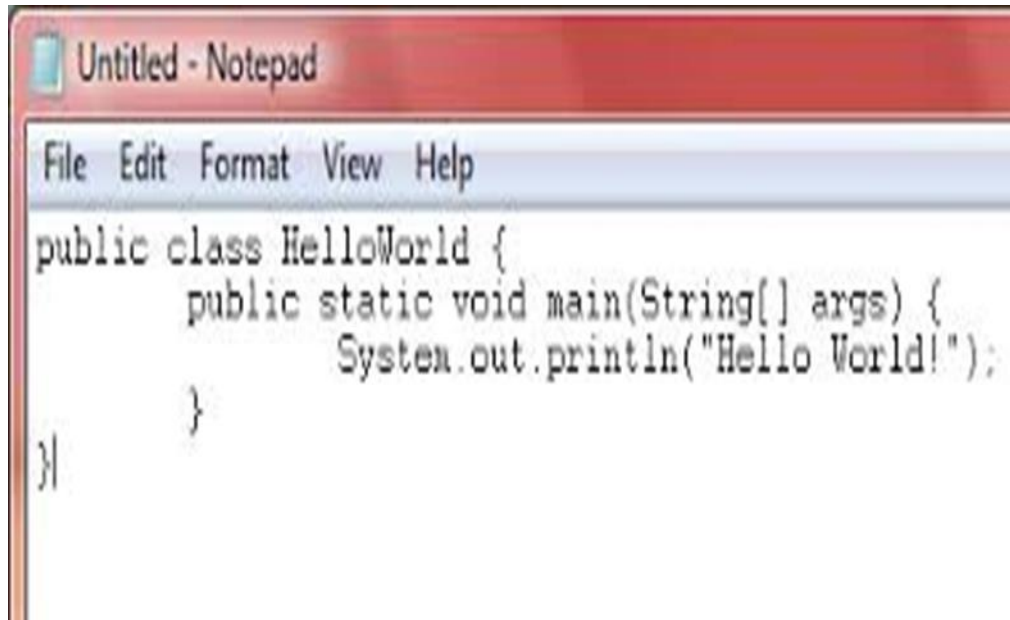
Τα πιο σημαντικά από αυτά είναι:

- o javac: Ο compiler της Java
- o java: Ο διερμηνέας της Java
- o javadoc: Το εργαλείο που δημιουργεί HTML τεκμηρίωση σε στυλ API
- o jar: Το εργαλείο που δημιουργεί και διαχειρίζεται αρχεία τύπου JAR
- o jdb: Ο Java debugger
- o javap: Disassembler αρχείων τύπου .class

Κύκλος υλοποίησης

Ο κύκλος υλοποίησης στη Java αρχίζει με τη σύνταξη του κώδικα από τον προγραμματιστή και την αποθήκευσή του σε ένα αρχείο πηγαίου κώδικα (source code file). Ένα από τα χαρακτηριστικά της Java, όπως και των περισσότερων γλωσσών προγραμματισμού είναι η μη απαίτηση συγκεκριμένου λογισμικού για τη δημιουργία προγραμμάτων.

Τα μόνα που χρειάζονται για να γραφτεί ένα απλό πρόγραμμα σε κάποιον υπολογιστή, είναι ένας οποιοσδήποτε κειμενογράφος (π.χ. notepad) και ένα πρόσφατο JDK.



```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello World!");
    }
}
```

Στο σχήμα γίνεται χρήση του notepad των Windows για τη σύνταξη του κώδικα μιας απλής εφαρμογής, του διάσημου Hello world. Προσπαθήστε να ακολουθήσετε ένα ένα τα βήματα που περιγράφονται ώστε να γράψετε και να τρέξετε το πρώτο σας πρόγραμμα στη Java. Αποθηκεύστε το αρχείο με το όνομα HelloWorld.java. Έχοντας γράψει τον κώδικα, το επόμενο βήμα είναι να τον μεταγλωττίσουμε. Ανοίξτε τη γραμμή εντολών (command) και θέστε ως τοποθεσία τον φάκελο που περιέχει το αρχείο που μόλις δημιουργήσατε. Στη συνέχεια πληκτρολογήστε: javac HelloWorld.java και πατήστε Enter. Αυτό δίνει εντολή καλεί τον compiler της Java να ελέγξει το αρχείο μας για σφάλματα. Αν υπάρχουν σφάλματα (συντακτικά λάθη κλπ) τότε ο compiler θα μας τα υποδείξει, διαφορετικά θα πράξει ένα αρχείο με όνομα HelloWorld.class.

Αν η απόκριση του συστήματος στην εντολή που δώσατε είναι η javac: file not found τότε δεν έχετε θέσει σωστά το command να βλέπει τον φάκελο που περιέχει το αρχείο πηγαίου κώδικα και ο compiler δεν το βρίσκει. Το αρχείο HelloWorld.class είναι αυτό που περιέχει τον ενδιάμεσο κώδικα (bytecode). Για να εκτελέσετε το πρόγραμμα σας πληκτρολογήστε στο command java HelloWorld και πατήστε Enter.

Ανακεφαλαιώνοντας λοιπόν, η εντολή για να μεταγλωττίσουμε ένα αρχείο πηγαίου κώδικα από το command είναι η: `javac ονομα_αρχείου.java` .



Δομή αρχείων στη Java

Σε αντίθεση με τις C/C++, όπου έχουμε αρχεία πηγαίου κώδικα αλλά και βιβλιοθήκες, στη Java υπάρχουν μόνο αρχεία πηγαίου κώδικα. Τα αρχεία αυτά έχουν πάντοτε κατάληξη .java, π.χ. MyClass.java.

Για την ονομασία αρχείων πηγαίου κώδικα στη Java ισχύουν οι εξής κανόνες σωστής πρακτικής:

- Τα ονόματα μπορούν να περιέχουν μόνο αλφαριθμητικούς χαρακτήρες. Περίεργα σύμβολα όπως τα * - & (^ ~ ! κλπ απαγορεύονται. Το μόνο σύμβολο που επιτρέπεται είναι το underscore (_)
- Τα ονόματα θα πρέπει πάντα να ξεκινούν με γράμμα και όχι με αριθμό
- Τα ονόματα δε θα πρέπει να περιέχουν κενά (Spaces). Αν κάποιο όνομα αποτελείται από δύο ή περισσότερες λέξεις, μπορείτε να τις διαχωρίσετε είτε με τη χρήση underscores είτε χρησιμοποιώντας κεφαλαίο χαρακτήρα στο πρώτο γράμμα κάθε λέξης. Για παράδειγμα, αν θέλετε να δημιουργήσετε ένα αρχείο που θα

φιλοξενήσει μία κλάση που φορτώνει φωτογραφίες στη μνήμη (Image Loader) τα ονόματα `ImageLoader` και `Image_Loader` είναι τα πλέον κατάλληλα και πλήρως εναρμονισμένα με το συγκεκριμένο κανόνα.

- Τα ονόματα θα πρέπει να χρησιμοποιούν αποκλειστικά το Αγγλικό αλφάβητο.

Ένα *Java project* μπορεί να περιέχει πολλά αρχεία πηγαίου κώδικα. Για να είναι εκτελέσιμο, θα πρέπει σε τουλάχιστον ένα από τα αρχεία του project να περιέχεται μία κεντρική μέθοδος.

Ένα τυπικό αρχείο πηγαίου κώδικα Java λοιπόν, μπορεί να περιέχει τα εξής:

- Δήλωση πακέτου
- Εντολές `import`
- Ορισμό κλάσης ή `interface`
- Σχόλια
- Μία κεντρική μέθοδο

Σχόλια (Comments)

Τα σχόλια είναι κομμάτια κειμένου που δεν έχουν καμία πρακτική αξία όσον αφορά στη λειτουργικότητα του προγράμματος, αλλά παρέχουν σημαντική βοήθεια σε οποιονδήποτε διαβάσει τον κώδικα διευκολύνοντάς τον να τον κατανοήσει.

Ο λόγος ύπαρξης των σχολίων είναι η τεκμηρίωση του κώδικα.

Για να καταλάβετε καλύτερα τη σημασία των σχολίων, υποθέστε πως ασχολείστε με τη δημιουργία και υλοποίηση ενός πολύπλοκου αλγορίθμου που θα χρησιμοποιηθεί σε κάποιο πρόγραμμα.

Το πρόγραμμα ολοκληρώνεται και παραδίδεται κανονικά. Μετά από έναν χρόνο, ο project manager σας λέει πως θα πρέπει να κάνετε μικροδιορθώσεις στον αλγόριθμο, ώστε να είναι περισσότερο αποτελεσματικός. Χωρίς την ύπαρξη εύστοχων σχολίων σε καίρια σημεία του κώδικα, θα χρειαζόσασταν αρκετό χρόνο διαβάζοντας τον κώδικα ώστε να θυμηθείτε πως ακριβώς είχατε σκεφτεί, όταν τον υλοποιούσατε. Ένα ακόμη πιο τρομακτικό σενάριο (για τον προγραμματιστή που δε γνωρίζει τι τον περιμένει) είναι αυτό, όπου έχετε αποχωρήσει από την εταιρεία και ο κώδικάς σας δίνεται για βελτιστοποίηση σε κάποιον προγραμματιστή που τον βλέπει για πρώτη φορά. Έχοντας κατανοήσει τη σημασία των σχολίων, ας δούμε τη σύνταξή τους. Στη Java υπάρχουν δύο τύποι σχολίων. Τα σχόλια που μπορούν να εκτείνονται σε περισσότερες της μιας γραμμές και αυτά της μίας γραμμής.

Τα σχόλια της πρώτης κατηγορίας ξεκινούν με την ακολουθία χαρακτήρων /* (κάθετος-αστεράκι) και τερματίζουν με την ακολουθία χαρακτήρων */ (αστεράκι-κάθετος). Για παράδειγμα, τα σχόλια που ακολουθούν είναι και τα δύο συντακτικά ορθά.

```
/* auto einai ena syntaktika ortho sxolio */
```

```
/*
```

```
* opws
```

```
* kai
```

```
* ayto
```

```
*/
```

Οτιδήποτε βρίσκεται μεταξύ των χαρακτήρων έναρξης και τερματισμού των σχολίων αγνοείται από τον compiler, πράγμα που σημαίνει πως μπορείτε να γράψετε οτιδήποτε θελήσετε, σχόλια στα Αγγλικά, σχόλια σε Greeklish, σχόλια στα Ελληνικά κλπ.

Τα σχόλια μίας γραμμής ξεκινούν με την ακολουθία χαρακτήρων // (2 κάθετοι) και τελειώνουν στο τέλος της τρέχουσας γραμμής χωρίς να χρειάζεται να τερματιστούν. Το ακόλουθο για παράδειγμα είναι ένα σωστό σχόλιο μίας γραμμής:

```
// this is a single line comment
```

Τα σχόλια συνήθως προβάλλονται στους editors των IDEs με πράσινο χρώμα. Η σωστή πρακτική για τη χρήση σχολίων προτείνει τη χρήση τους με μέτρο, θα πρέπει δηλαδή να χρησιμοποιούνται μόνο εκεί που πραγματικά χρειάζονται.

Κώδικας που δεν περιέχει καθόλου σχόλια είναι αρκετά δυσνόητος αλλά στον αντίποδα, κώδικας που περιέχει υπερβολικό αριθμό σχολίων γίνεται εξαιρετικά δυσανάγνωστος και κουραστικός.

Τέλος, με τη χρήση ειδικών σχολίων (`/** */`) και ειδικών tags που ονομάζονται annotations και σε συνδυασμό με το εργαλείο javadoc του JDK είναι δυνατόν να παράξουμε ένα HTML reference manual για τον κώδικά μας, στο συλ του Java API reference manual.

Στο παράδειγμα που ακολουθεί γίνεται χρήση τέτοιων annotations εντός των σχολίων:

```
/**  
  
 * This class represents a jpg image  
  
 *  
  
 * @author Nikos Panagos  
  
 * @version 1.0, March 23, 2008  
  
 * @see server  
  
 */
```

Κεντρική Μέθοδος (main)

Για να είναι ένα πρόγραμμα εκτελέσιμο, θα πρέπει οπωσδήποτε σε κάποιο από τα αρχεία πηγαίου κώδικα, από τα οποία αποτελείται να υπάρχει τουλάχιστον μία κεντρική μέθοδος (main method). Αυτή είναι μία ακόμη διαφορά της Java από τις C/C++ μιας και στη Java μπορούμε να έχουμε περισσότερες της μιας κεντρικές μεθόδους (σε διαφορετικά βέβαια αρχεία) και ο χρήστης να επιλέξει ποια θα είναι αυτή που θα εκτελεστεί, ενώ στις C/C++ θα πρέπει να υπάρχει αποκλειστικά μία κεντρική συνάρτηση ανά project.

Η κεντρική μέθοδος αποτελεί το σημείο εισόδου στο πρόγραμμα μας και όπως όλες οι μέθοδοι περικλείεται σε ένα ζεύγος από άγκιστρα ({ }) και περιέχει μία σειρά από εντολές.

Όταν δώσουμε εντολή στο διερμηνέα της Java να εκτελέσει το πρόγραμμα μας, π.χ. εκτελώντας.

java MyClass,

Ο διερμηνέας θα ξεκινήσει να εκτελεί τις εντολές που περιέχονται στη main αρχίζοντας από αυτήν που βρίσκεται αμέσως μετά το άγκιστρο έναρξης και καταλήγοντας, αφού εκτελέσει, σε αυτήν που βρίσκεται αμέσως πριν το άγκιστρο τερματισμού.

Οι εντολές εκτελούνται η μία μετά την άλλη με τη σειρά που είναι γραμμένες (σειριακά – sequentially).

Στη Java η κεντρική μέθοδος έχει πάντοτε τη μορφή:

```
public static void main(String[] args)
```

Ως εντολή ορίζουμε στη Java μία 'έκφραση' που ξεκινάει από την αρχή μιας γραμμής και τερματίζει με το χαρακτήρα ; (ερωτηματικό – semi-colon) όντας παράλληλα συντακτικά ορθή.

Σύμφωνα με την ισχύουσα πρακτική, θα πρέπει στα προγράμματά μας να έχουμε μία εντολή ανά γραμμή. Μία τέτοια εντολή λοιπόν μπορεί να είναι η δήλωση μιας μεταβλητής, η ορθή χρήση μιας δεσμευμένης λέξης της γλώσσας, η κλήση μιας μεθόδου, μία αριθμητική πράξη κλπ.

Μία ομάδα εντολών που περικλείονται σε ένα ζεύγος αγκίστρων ({}) ονομάζεται μπλοκ εντολών.

Ένα ακόμη χαρακτηριστικό της κεντρικής μεθόδου στη Java είναι η δυνατότητα που παρέχει να περνάμε παραμέτρους (command line arguments) κατά την εντολή εκτέλεσης του προγράμματος, οι οποίες μπορεί να χρησιμοποιηθούν από τον κώδικα για την επίτευξη συγκεκριμένης λειτουργικότητας.

Τύποι δεδομένων και σταθερές

Το σύστημα τύπων μιας γλώσσας προγραμματισμού αποσκοπεί στην πρόληψη σφαλμάτων που μπορούν να παρουσιαστούν κατά την εκτέλεση των προγραμμάτων. Η βασική λειτουργία του συστήματος τύπων είναι η συσχέτιση τύπων με τις μεταβλητές ενός προγράμματος και με τις εκφράσεις που συναντώνται σε αυτό. Το σύστημα τύπων της Java συνδυάζει χαρακτηριστικά που συναντιούνται σε γλώσσες με στατικά αλλά και με δυναμικά συστήματα τύπων.

Είναι στατικό με την έννοια, ότι κάθε μεταβλητή συσχετίζεται με ένα συγκεκριμένο και μη τετριμμένο τύπο, ο οποίος είναι γνωστός κατά τη μεταγλώττιση.

Είναι δυναμικό με την έννοια, ότι κατά το χρόνο εκτέλεσης διατηρούνται πληροφορίες σχετικές με τον τύπο των αντικειμένων, ώστε να υποστηρίζεται ο πολυμορφισμός με ασφαλή τρόπο. Οι τύποι στη Java χωρίζονται σε δυο μεγάλες κατηγορίες: τους βασικούς τύπους και τους τύπους αναφοράς.

Βασικοί τύποι

Οι βασικοί τύποι της Java χρησιμοποιούνται για την αναπαράσταση θεμελιωδών τύπων δεδομένων, όπως είναι οι αριθμοί, οι χαρακτήρες και οι λογικές τιμές. Συνοπτικά δίνονται στον παρακάτω πίνακα.

Στον ίδιο πίνακα περιγράφονται επίσης οι δυνατές τιμές που μπορούν να πάρουν τα αντίστοιχα δεδομένα.

Πρωτογενείς (Primitive) Τύποι Δεδομένων

Λέξη Κλειδί	Περιγραφή	Μέγεθος/Μορφή
(Ακέραιοι)		
byte	ακέραιος μεγέθους ενός byte	8-bit two's complement
short	μικρός (short) ακέραιος	16-bit two's complement
int	ακέραιος	32-bit two's complement
long	μεγάλος (long) ακέραιος	64-bit two's complement
(Πραγματικοί)		
float	Απλής ακρίβειας αριθ. κιν. υποδιαστ.	32-bit IEEE 754
double	Διπλής ακρίβειας αριθ. κιν. υποδιαστ.	64-bit IEEE 754
(Άλλοι Τύποι)		
char	Ένας χαρακτήρας	16-bit Unicode character
boolean	Μια boolean τιμή (true ή false)	true ή false

Java

Βασικά Στοιχεία της γλώσσας Java

3

Μεταβλητές μπορούν να ορισθούν στο εσωτερικό μεθόδων ή κλάσεων με τρόπο παρόμοιο, όπως στη γλώσσα C. Για παράδειγμα, ο κώδικας: `int x; double d1, d2;` δηλώνει τρεις μεταβλητές με ονόματα `x`, `d1` και `d2`. Η πρώτη έχει τύπο `int`, ενώ οι δυο επόμενες έχουν τύπο `double`. Επίσης, οι μεταβλητές μπορούν να αρχικοποιηθούν κατά τη δήλωσή τους, όπως για παράδειγμα στον παρακάτω κώδικα: `int x = 1; double d1 = 3.14, d2 = -8.0;`

Τύποι αναφοράς

Όλοι οι τύποι που δεν είναι βασικοί ανήκουν στην κατηγορία των τύπων αναφοράς. Στην κατηγορία αυτή ανήκουν όλοι οι τύποι αντικειμένων και, κατά συνέπεια, όλοι οι τύποι δεδομένων που μπορεί να ορίσει ο προγραμματιστής. *Οι τιμές που μπορεί να πάρει ένα δεδομένο που ανήκει σε ένα τύπο αναφοράς είναι στην ουσία αναφορές σε αντικείμενα: όχι τα ίδια τα αντικείμενα αλλά δείκτες σε αυτά.* Μπορούν να παραλληλισθούν με τους δείκτες στη C, με τη διαφορά ότι δεν απαιτείται ειδική μεταχείριση προκειμένου να προσπελάσει κανείς τα αντικείμενα στα οποία δείχνουν.

Η Java δεν επιτρέπει την κατασκευή αναφορών παρά μόνο με την εκχώρηση ή το πέρασα αντικειμένων. Το όνομα “τύποι αναφοράς” έχει προέλθει από το διαφορετικό τρόπο με τον οποίο τα δεδομένα που ανήκουν σε αυτούς τους τύπους διακινούνται κατά την εκτέλεση των προγραμμάτων. Όταν ένα δεδομένο που ανήκει σε κάποιο βασικό τύπο πρόκειται να εκχωρηθεί σε μια μεταβλητή ή να περαστεί ως όρισμα σε μια μέθοδο, η τιμή του απλώς αντιγράφεται. Αντίθετα, όταν ένα δεδομένο που ανήκει σε κάποιο τύπο αναφοράς πρόκειται να εκχωρηθεί σε μια μεταβλητή ή να περαστεί ως όρισμα σε μια μέθοδο, αυτό που αντιγράφεται είναι η αναφορά στο αντικείμενο, όχι το ίδιο το αντικείμενο. Με αυτό τον τρόπο, δημιουργούνται περισσότερες αναφορές στο ίδιο αντικείμενο, το οποίο μπορεί να προσπελασθεί μέσω οποιασδήποτε από αυτές.

Η ειδική τιμή *null* υποδηλώνει ότι ένα δεδομένο, που ανήκει σε κάποιο τύπο αναφοράς, δεν αναφέρεται σε κανένα αντικείμενο. Αυτή είναι και η τιμή που χρησιμοποιείται για την αυτόματη αρχικοποίηση δεδομένων, που ανήκουν σε τύπους αναφοράς. Πριν χρησιμοποιηθούν αυτά τα δεδομένα για οτιδήποτε μη τετριμμένο, είναι απαραίτητο να ληφθεί μέριμνα, ώστε να αναφέρονται σε υπαρκτά αντικείμενα.

Ειδικοί τύποι

Πρόκειται για τύπους, που έχουν ιδιαίτερη σημασία για τη γλώσσα Java και για τους οποίους χρησιμοποιείται ιδιαίτερη σύνταξη. Ο τύπος συμβολοσειράς ονομάζεται String. Οι συμβολοσειρές στη Java είναι ακολουθίες χαρακτήρων Unicode. Σταθερές τύπου συμβολοσειράς μπορούν να γραφούν ανάμεσα σε διπλά εισαγωγικά. Ο μεταγλωττιστής της Java τις μετατρέπει αυτόματα σε κατάλληλα αντικείμενα τύπου String. Εκτός από κοινούς χαρακτήρες, οι σταθερές u963 συμβολοσειρές μπορούν να περιέχουν ακολουθίες διαφυγής.

Παραδείγματα:

- "Java"
- "Hello world!\n"
- "So long \"Java\" world...\n"

Οι τύποι πινάκων ανήκουν επίσης στην κατηγορία των ειδικών τύπων. Πρόκειται για τύπους αντικειμένων με ειδική σύνταξη που διευκολύνει τη δήλωση και τη χρήση των πινάκων. Ένας τύπος πίνακα προκύπτει από κάποιο έγκυρο τύπο, με την προσθήκη δυο κενών αγκυλών []. Οι αγκύλες μπορούν να τοποθετηθούν εναλλακτικά είτε μετά το όνομα του τύπου, ή μετά το όνομα της μεταβλητής που ορίζεται. *Παραδείγματα δήλωσης πινάκων είναι τα ακόλουθα:*

```
int myArray [];
```

```
int [] yourArray;
```

```
String hisArray [];
```

Πολυδιάστατοι πίνακες μπορούν να δηλωθούν με παρόμοιο τρόπο:

```
double [][] myMatrix2D;
```

```
int yourMatrix3D [][][];
```

Εκφράσεις στη Java

Οι εκφράσεις στη Java είναι τμήματα του κώδικα που, όταν εκτελούνται, παράγουν κάποιο αποτέλεσμα. Το αποτέλεσμα που προκύπτει από την εκτέλεση μιας έκφρασης ονομάζεται τιμή της έκφρασης, και για το λόγο αυτό η εκτέλεση μιας έκφρασης ονομάζεται συχνά και αποτίμηση της έκφρασης. Η τιμή μιας έκφρασης στη Java μπορεί να ανήκει σε κάποιον από τους βασικούς τύπους, π.χ. να είναι αριθμητική, να ανήκει σε κάποιο τύπο αναφοράς, ή να ανήκει στον ειδικό τύπο `void`. Ο τύπος κάθε έκφρασης είναι πάντα γνωστός κατά τη μεταγλώττιση του κώδικα και η τιμή που θα προκύψει κατά την αποτίμηση στο χρόνο εκτέλεσης θα ανήκει υποχρεωτικά στον ίδιο τύπο. Ο τύπος `void` είναι κενός, δεν περιέχει δηλαδή καμιά τιμή. Η αποτίμηση μιας έκφρασης με τύπο `void` γίνεται μόνο για τις παρενέργειες που μπορεί να έχει. Το αποτέλεσμα δεν μπορεί να χρησιμοποιηθεί γιατί δεν υπάρχει.

Η σύνταξη των εκφράσεων της Java μοιάζει πολύ με αυτή της γλώσσας C. Ιδιαίτερη έμφαση έχει δοθεί στην απλότητα και τη συνέπεια των εκφράσεων.

Παράδειγμα:

Η έκφραση $c = a + b * 2$ ερμηνεύεται ως $c = (a + (b * 2))$

γιατί ο τελεστής `+` έχει μεγαλύτερη προτεραιότητα από τον τελεστή `=` (11 έναντι 1)

και ο τελεστής `*` έχει μεγαλύτερη προτεραιότητα από τον τελεστή `+` (12 έναντι 11).

Ακολουθίες Διαφυγής (Escape Sequences)

Σε όλες τις γλώσσες προγραμματισμού υπάρχουν κάποια σύμβολα που έχουν ειδική σημασία για τη γλώσσα. Χαρακτηριστικό παράδειγμα στη Java είναι τα διπλά εισαγωγικά (`"`) που σηματοδοτούν την αρχή και το τέλος ενός αλφαριθμητικού (string). Το γεγονός αυτό, ότι δηλαδή τα σύμβολα αυτά έχουν ειδική σημασία για τη γλώσσα δημιουργεί προβλήματα στις περιπτώσεις, που θέλουμε να τα

χρησιμοποιήσουμε με διαφορετικό τρόπο από αυτόν που αποτελεί την πρωταρχική τους λειτουργία.

Για παράδειγμα, ας υποθέσουμε πως θέλετε σε κάποιο πρόγραμμα σας να εμφανίσετε το γνωστό Σαιξπηρικό ερώτημα *"To be or not to be?"* (μαζί με τα εισαγωγικά). Αν χρησιμοποιήσετε την εντολή `System.out.println("To be or not to be?");` θα προβληθεί μεν το μήνυμα αλλά χωρίς τα εισαγωγικά γιατί αυτά όπως είπαμε στη συγκεκριμένη εντολή ορίζουν την αρχή και το τέλος του αλφαριθμητικού. Αν τώρα δοκιμάσετε την εντολή : `System.out.println("To be or not to be?");` θα διαπιστώσετε πως η συγκεκριμένη γραμμή προκαλεί σφάλμα κατά τη μεταγλώττιση. Πράγματι, ο compiler ερμηνεύει το πρώτο ζεύγος εισαγωγικών ως ένα κενό αλφαριθμητικό, ακολουθεί μία σειρά λέξεων, τις οποίες ο compiler δεν αναγνωρίζει (η φράση 'To be or not to be' δεν έχει κανένα νόημα για αυτόν) και τέλος υπάρχει ένα ακόμη κενό αλφαριθμητικό.

Για την αντιμετώπιση τέτοιων καταστάσεων υπάρχουν σε κάθε γλώσσα οι λεγόμενες ακολουθίες διαφυγής (escape sequences). Πρόκειται για ακολουθίες χαρακτήρων που όταν ο compiler τις 'συναντήσει' γνωρίζει πως να τις χειριστεί, π.χ. καταλαβαίνει πως ο προγραμματιστής επιθυμεί να συμπεριλάβει διπλά εισαγωγικά σε κάποιο αλφαριθμητικό κλπ.

`\n` newline

`\t` tab

`\b` backspace

`\f` form feed

`\r` return

`\"` " (διπλά εισαγωγικά)

`'` ' (μονό εισαγωγικό)

`\\` \ (back slash)

\uDDDD Ο χαρακτήρας Unicode με κωδικό DDDD (DDDD είναι 4ψήφιος δεκαεξαδικός αριθμός)

Για τα διπλά εισαγωγικά, η ακολουθία διαφυγής είναι η \". Επιστρέφοντας λοιπόν στο παράδειγμά μας, για να επιτύχουμε την προβολή του επιθυμητού μηνύματος θα γράψαμε: `System.out.println("\"To be or not to be?\");`

Escape Sequences

- Some Java escape sequences:

<u>Escape Sequence</u>	<u>Meaning</u>
<code>\b</code>	backspace
<code>\t</code>	tab
<code>\n</code>	newline
<code>\r</code>	carriage return
<code>\"</code>	double quote
<code>\'</code>	single quote
<code>\\</code>	backslash

- See `Roses.java`

Πακέτα (Packages)

Τα πακέτα είναι το δεύτερο συστατικό της γλώσσας. Πρόκειται για ένα ιδιαίτερα σημαντικό κομμάτι της γλώσσας μιας και παρέχει στους προγραμματιστές έναν μηχανισμό ιεραρχικής οργάνωσης των κλάσεων και διευκολύνει την κατάτμηση του κώδικα (modularity). Φανταστείτε τα πακέτα σαν κουτιά που μέσα τους περιέχουν κλάσεις, interfaces καθώς και άλλα πακέτα. Όλες οι κλάσεις της Java είναι οργανωμένες σε τέτοια πακέτα ενώ παράλληλα δίνεται η δυνατότητα στον προγραμματιστή να δημιουργήσει δικά του. Κάθε πακέτο έχει συγκεκριμένο όνομα γραμμένο κατά σύμβαση με πεζούς πάντοτε χαρακτήρες, ώστε να ξεχωρίζει από τα ονόματα των κλάσεων και μπορεί να περιέχει κλάσεις, interfaces και άλλα πακέτα.

Όταν αναφερόμαστε σε κάποιο πακέτο ξεκινάμε από το κορυφαίο στην ιεραρχία και προχωράμε στο πακέτο που επιθυμούμε διαχωρίζοντάς τα με τελείες.

Για παράδειγμα με την έκφραση `java.util` αναφερόμαστε στο πακέτο `util` που περιέχεται στο πακέτο `java`. Το πακέτο `java` βρίσκεται στην κορυφή της ιεραρχίας. Αν θέλουμε να χρησιμοποιήσουμε στον κώδικά μας μία κλάση που περιέχεται σε κάποιο πακέτο, θα πρέπει να πληκτρολογήσουμε το πλήρες όνομά της (fully qualified name) ώστε ο `compiler` να μπορέσει να την εντοπίσει. Έτσι, η εντολή `java.util.Stack s = new java.util.Stack();` υποδεικνύει στον `compiler` να δημιουργήσει ένα αντικείμενο της κλάσης `Stack` που περιέχεται στο πακέτο `util`, το οποίο με τη σειρά του περιέχεται στο πακέτο `java` που βρίσκεται στην κορυφή της ιεραρχίας. Η αναφορά χρησιμοποιώντας το πλήρες όνομα δεν είναι απαραίτητη, μόνο, στην περίπτωση που οι δύο κλάσεις βρίσκονται στο ίδιο πακέτο και 'βλέπουν' η μία την άλλη.

Είναι προφανές πως η χρήση του πλήρους ονόματος απαιτεί αρκετή πληκτρολόγηση από πλευράς προγραμματιστή και για την καταπολέμηση του συγκεκριμένου προβλήματος η ομάδα της Java δημιούργησε τη λύση του `import`. Κάνοντας χρήση της δεσμευμένης λέξης `import` σε συνδυασμό με το πλήρες όνομα της κλάσης, που θέλουμε να χρησιμοποιήσουμε στον κώδικά μας, δε χρειάζεται πλέον να αναφερόμαστε σε αυτήν με το πλήρες όνομα και αρκεί να χρησιμοποιήσουμε το απλό όνομά της. Το προηγούμενο παράδειγμα δηλαδή, κάνοντας χρήση του `import` θα μετατρεπόταν ως εξής :

```
import java.util.Stack;
```

```
...
```

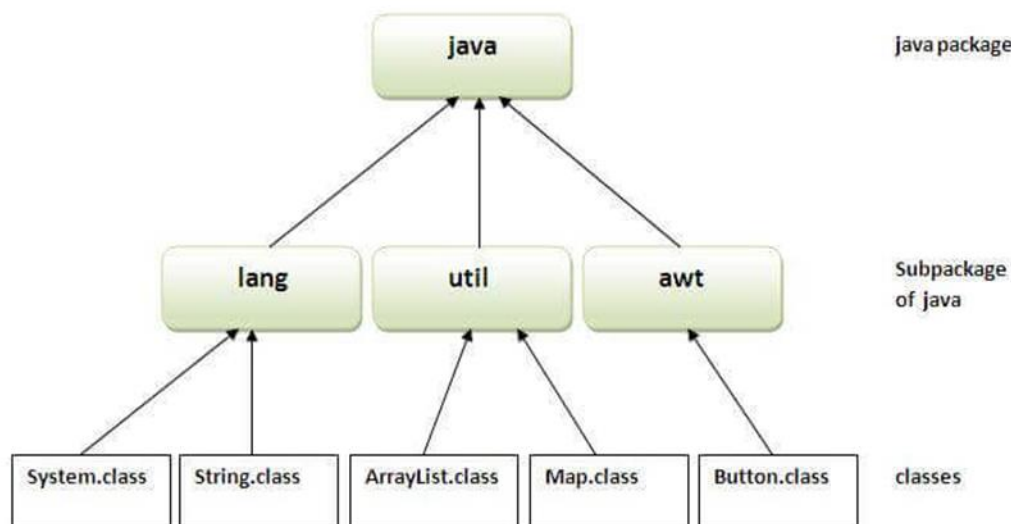
```
Stack s = new Stack();
```

Η χρήση `import` δεν έχει καμία αρνητική συνέπεια όσον αφορά το μέγεθος του προγράματός μας (δεν εισάγεται ο κώδικας της κλάσης) και μπορείτε να το χρησιμοποιείτε κατά βούληση. Ένας άλλος τρόπος χρήσης του `import` είναι σε

συνδυασμό με το χαρακτήρα μπαλαντέρ (wildcard), όπως φαίνεται στην έκφραση που ακολουθεί:

```
import java.util.*;
```

Η συγκεκριμένη εντολή υποδεικνύει στον compiler να συμπεριλάβει τα ονόματα όλων των κλάσεων και interfaces που περιέχονται στο πακέτο java.util (όχι όμως τα ονόματα των πακέτων που μπορεί να περιέχονται σε αυτό). Αν και κάτι τέτοιο δείχνει βολικό, είναι προτιμότερο να εισάστε συγκεκριμένοι όταν υποδεικνύετε στον compiler ποιες κλάσεις θέλετε να εισάγετε και να χρησιμοποιείτε το χαρακτήρα μπαλαντέρ μόνο όταν θέλετε να εισάγετε πολλές κλάσεις από το ίδιο πακέτο.



Κοινοί Διάλογοι (Common Dialogs)

Όλες οι γλώσσες προγραμματισμού που παρέχουν τη δυνατότητα δημιουργίας παραθυρικών εφαρμογών διαθέτουν μία σειρά διαλόγων που ονομάζονται κοινοί διάλογοι (common dialogs) και διευκολύνουν την εισαγωγή δεδομένων από το χρήστη, την προβολή μηνυμάτων κλπ. Η Java δε θα μπορούσε να αποτελεί εξαίρεση και περιέχει τέτοιους διαλόγους στο πακέτο javax.swing, στο οποίο βρίσκονται όλες οι κλάσεις για τη δημιουργία GUI (Graphical User Interface).

Δύο κοινοί διάλογοι που χρησιμοποιεί η java είναι:

- Ο διάλογος προβολής μηνυμάτων στο χρήστη
- Ο διάλογος εισαγωγής δεδομένων

Ο πρώτος εμφανίζεται καλώντας τη μέθοδο `showMessageDialog()` του αντικειμένου `JOptionPane`. Ανοίξτε το IDE της επιλογής σας και ορίστε ένα νέο project. Στο project αυτό δημιουργήστε μία νέα κλάση και ονομάστε την `Example01`. Στη συνέχεια πληκτρολογήστε τον ακόλουθο κώδικα, αποθηκεύστε το αρχείο και εκτελέστε το.

Θα έχετε:

```
import javax.swing.JOptionPane; // program uses JOptionPane

public class Example01 {

    public static void main(String args[]){

        JOptionPane.showMessageDialog(null,

            "Welcome\nto\nJava\nProgramming!");

        System.exit(0); // terminate application with window

    }

}
```

Τι ακριβώς κάνει αυτός ο κώδικας;

Η πρώτη γραμμή κάνει `import` την κλάση `JOptionPane` μιας και θα την χρησιμοποιήσουμε στη συνέχεια. Στην κεντρική μέθοδο υπάρχει η κλήση της μεθόδου `showMessageDialog()` της `JOptionPane`, στην οποία περνάμε ως παραμέτρους `null` και το αλφαριθμητικό `"Welcome\nto\nJava\nProgramming!"`. Η συγκεκριμένη κλήση είναι υπεύθυνη για την εμφάνιση του διαλόγου. Τέλος, η τελευταία γραμμή τερματίζει το πρόγραμμα μας (είναι απαραίτητη μόνο για εφαρμογές που περιλαμβάνουν παράθυρα).

Η `showMessageDialog` έχει διαφορετικές εκδόσεις, επιτρέποντας στον προγραμματιστή να θέσει τον τίτλο του παραθύρου εκτός από το μήνυμα, καθώς και το εικονίδιο που προβάλλεται στο διάλογο.

Ο δεύτερος διάλογος είναι αυτός της εισαγωγής δεδομένων από το χρήστη. Για παράδειγμα έχουμε:

```
import javax.swing.JOptionPane;

public class Example02 {

    public static void main(String args[]){

        JOptionPane.showMessageDialog(null,

            "Hello " + JOptionPane.showInputDialog(

                "Please type your name:"));

        System.exit(0);

    }

}
```

Η μέθοδος `showInputDialog()` της `JOptionPane` είναι αυτή που προβάλλει το διάλογο εισαγωγής δεδομένων. Στον διάλογο αυτόν ο χρήστης πληκτρολογεί ένα σύντομο κείμενο και πατώντας το 'OK' ό,τι έχει δώσει ο χρήστης επιστρέφεται με τη μορφή αλφαριθμητικού και συνήθως το αποθηκεύουμε σε μία μεταβλητή τύπου `String` για περαιτέρω επεξεργασία. Δεδομένου του ότι ακόμη δεν έχουμε μάθει να δηλώνουμε μεταβλητές, στο συγκεκριμένο παράδειγμα το αλφαριθμητικό που επιστρέφει η `showInputDialog()` προστίθεται στο "Hello " και χρησιμοποιείται κατ' ευθείαν από τη `showMessageDialog()`. Αν ο χρήστης πατήσει 'Cancel' τότε η `showInputDialog()` θα επιστρέψει `null`.

Τύποι Δεδομένων (Data Types)

Στον προγραμματισμό, ένας τύπος δεδομένων είναι ένα σύνολο τιμών και οι λειτουργίες (πράξεις) που μπορούν να εφαρμοστούν στις τιμές αυτές. Σχεδόν όλες οι γλώσσες προγραμματισμού μας παρέχουν κάποιον τρόπο για να χειριστούμε τους βασικούς τύπους δεδομένων όπως είναι για παράδειγμα οι αριθμοί. Έτσι λοιπόν και η Java μας παρέχει έμφυτη υποστήριξη τόσο για τους βασικούς τύπους δεδομένων όσο και για κάποιους σύνθετους. Στους βασικούς τύπους που ονομάζονται primitives ή primitives data types ανήκουν οι ακέραιοι αριθμοί, οι πραγματικοί (κινητής υποδιαστολής), οι χαρακτήρες και ο τύπος boolean.

Εδώ θα πρέπει να τονίσουμε πως τα σύνολα των ακεραίων και των πραγματικών, όπως θα δούμε και στη συνέχεια είναι φυσικά πεπερασμένα. Στους σύνθετους τύπους δεδομένων ανήκουν τα αλφαριθμητικά ή strings, οι πίνακες (arrays), οι απαριθμητοί τύποι (enumerated types) και φυσικά οι αφηρημένοι τύποι δεδομένων (abstract data types). Κάθε ένας από τους τύπους αυτούς καταλαμβάνει συγκεκριμένο χώρο στη μνήμη του υπολογιστή και είναι ικανός να αποθηκεύσει τιμές που εκτείνονται σε επίσης συγκεκριμένο εύρος.

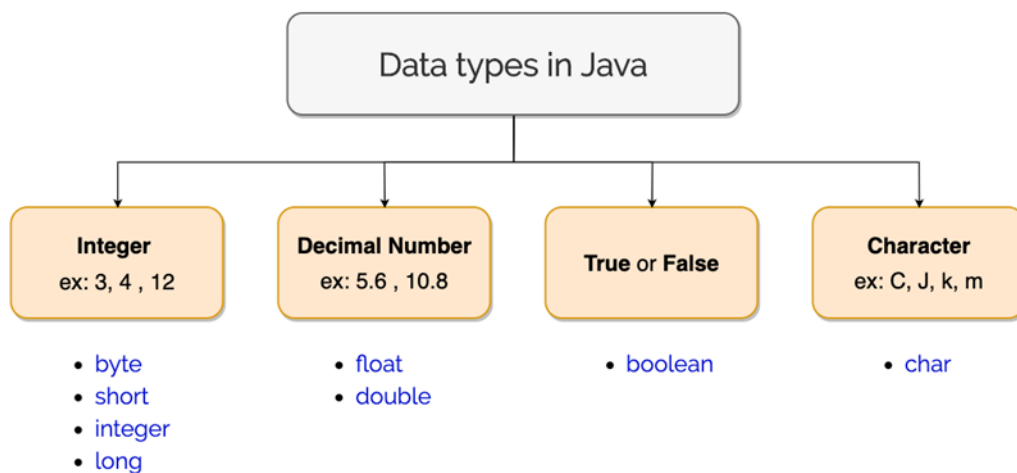
Κάποιοι από τους βασικούς τύπους δεδομένων στη Java:

- ✓ Ο τύπος char (χαρακτήρας – character) χρησιμοποιείται για την αποθήκευση ενός μεμονωμένου χαρακτήρα κωδικοποίησης Unicode. Η Java έχει έμφυτη υποστήριξη Unicode, πράγμα που σημαίνει πως μπορούμε πολύ εύκολα να γράψουμε προγράμματα που υποστηρίζουν την Ελληνική γλώσσα. Έχει μέγεθος 2 bytes (16 bits). Οι ακέραιοι στη Java μπορούν να αναπαρασταθούν μέσω τεσσάρων διαφορετικών τύπων ως προς το μέγεθος και την ακρίβεια.
- ✓ Ο τύπος byte, χρησιμοποιείται κυρίως στις δικτυακές εφαρμογές και καταλαμβάνει μέγεθος ενός byte (8 bits).
- ✓ Ο τύπος short επίσης αποθηκεύει ακέραιους αριθμούς στο εύρος μεταξύ - 215 έως 215-1 και καταλαμβάνει μέγεθος στη μνήμη ίσο με 2 bytes (16 bits).

- ✓ Ο τύπος `int` είναι ο πιο κοινός τύπος για τον χειρισμό και την προσωρινή αποθήκευση ακεραίων αριθμών με μέγεθος 4 bytes (32 bits). Έχει την ικανότητα να αποθηκεύσει αριθμούς που βρίσκονται στο εύρος από -2147483648 έως 2147483647.

Για τους πραγματικούς αριθμούς (κινητής υποδιαστολής), η Java διαθέτει δύο διαφορετικούς τύπους που επίσης διαφέρουν ως προς το μέγεθος που καταλαμβάνουν στη μνήμη και την ακρίβεια.

- ✓ Ο τύπος `float`, (κινητής υποδιαστολής απλής ακρίβειας) έχει μέγεθος 4 bytes (32 bits) και μπορεί να αποθηκεύσει δεκαδικούς που βρίσκονται μεταξύ του $1.4 * e^{-45}$ και του $3.4 * e^{38}$.
- ✓ τύπος `double`, (κινητής υποδιαστολής διπλής ακρίβειας), έχει μέγεθος 8 bytes (64 bits) και μπορεί να αποθηκεύσει μεγαλύτερους αριθμούς και με μεγαλύτερη ακρίβεια. Ο τύπος `boolean`, έχει μέγεθος 1 byte (8 bits) και μπορεί να αποθηκεύσει τιμές `true` ή `false`.



Κυριολεκτικές Τιμές (Literals)

Με τον όρο κυριολεκτική τιμή (literal) αναφερόμαστε στην αναπαράσταση μιας τιμής κάποιου από τους βασικούς τύπους ή αλφαριθμητικού. Για παράδειγμα:

```
x = 5;
```


το 5 είναι μία ακέραια κυριολεκτική τιμή. Στις επόμενες γραμμές υπάρχουν κυριολεκτικές τιμές κάποιων από τους βασικούς τύπους και ενός αλφαριθμητικού.

```
178 // int
```

```
"I like Java" // string
```

```
-872.363 // double
```

```
'c' // char
```

```
true // boolean
```

```
8.1415F // float
```

```
9998883212L // long
```

1. Ακέραιες κυριολεκτικές τιμές

Στη Java μπορούμε να εκφράσουμε κυριολεκτικές τιμές ακεραίων αριθμών σε τρία διαφορετικά συστήματα, στο δεκαδικό, το οκταδικό και το δεκαεξαδικό.

Δεκαδικό:

Οι κυριολεκτικές τιμές ακεραίων στο δεκαδικό σύστημα έχουν την πιο απλή μορφή μιας και είναι η ίδια με αυτήν που γνωρίζουμε από τα μαθηματικά. Αρκεί να πληκτρολογήσουμε τον αριθμό που επιθυμούμε με το πρόσημό του (αν είναι θετικός δε χρειάζεται πρόσημο όπως και στα μαθηματικά). Παραδείγματα κυριολεκτικών τιμών στο δεκαδικό σύστημα: 7, -39, 1023

Οκταδικό:

Το οκταδικό σύστημα χρησιμοποιεί τα ψηφία από το 0 έως το 7 για την αναπαράσταση αριθμών. Για να γράψουμε μία ακέραια κυριολεκτική τιμή στο οκταδικό σύστημα, αρκεί να τοποθετήσουμε το ψηφίο 0 μπροστά από τον αριθμό, π.χ.

```
011 // ο αριθμός 9 στο οκταδικό
```

-072 // ο αριθμός -58 στο οκταδικό

Μπορούμε να αναπαραστήσουμε αριθμούς που περιλαμβάνουν μέχρι 21 ψηφία μη συμπεριλαμβανομένου του αρχικού 0.

Δεκαεξαδικό:

Το δεκαεξαδικό σύστημα χρησιμοποιεί για την αναπαράσταση αριθμών τα ψηφία 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e και f. Για να γράψουμε μία ακέραια κυριολεκτική τιμή στο δεκαεξαδικό σύστημα τοποθετούμε την ακολουθία χαρακτήρων 0x.

0x7fefde02 // ο αριθμός 2146426370 στο δεκαεξαδικό

-0XCAFE // ο αριθμός -51966 στο δεκαεξαδικό

Μπορούμε να αναπαραστήσουμε αριθμούς που περιλαμβάνουν μέχρι 16 ψηφία μη συμπεριλαμβανομένων των αρχικών 0x.

2. Δεκαδικές κυριολεκτικές τιμές

Οι δεκαδικές κυριολεκτικές τιμές εκφράζονται εύκολα στη Java πληκτρολογώντας τον αριθμό που επιθυμούμε προσέχοντας όμως πως έχουμε χρησιμοποιήσει ως υποδιαστολή τον χαρακτήρα τελεία και όχι το κόμμα, π.χ

1130.988 // σωστό

-39,21356 // λάθος, το κόμμα δε λογίζεται ως υποδιαστολή

3. Boolean κυριολεκτικές τιμές

Οι μοναδικές κυριολεκτικές τιμές που μπορεί να λάβει ο τύπος boolean είναι οι true και false.

4. Κυριολεκτικές τιμές χαρακτήρων

Μία κυριολεκτική τιμή τύπου χαρακτήρα αποτελείται από έναν μεμονωμένο χαρακτήρα, ο οποίος περικλείεται μέσα σε ένα ζεύγος από μονά εισαγωγικά.

'a' // ο χαρακτήρας a

'@' // ο χαρακτήρας @

'\n' // ο χαρακτήρας newline

5. Κυριολεκτικές τιμές αλφαριθμητικών

Τα αλφαριθμητικά είναι ο μοναδικός σύνθετος τύπος δεδομένων ο οποίος μπορεί να λάβει κυριολεκτικές τιμές, όπως οι βασικοί τύποι. Μια κυριολεκτική τιμή αλφαριθμητικού αναπαρίσταται από μία ακολουθία χαρακτήρων οι οποίοι περικλείονται μέσα σε ένα ζεύγος από διπλά εισαγωγικά, π.χ.

```
"This is a string literal"
```

Μεταβλητές (Variables)

Κάθε πρόγραμμα επεξεργάζεται κάποιου είδους δεδομένα. Για να συμβεί αυτό, θα πρέπει τα δεδομένα αυτά να αποθηκευτούν προσωρινά στη μνήμη του υπολογιστή. Τον ρόλο αυτόν αναλαμβάνουν να τον φέρουν σε πέρας οι μεταβλητές. Μία μεταβλητή λοιπόν, είναι το μέσο που μας επιτρέπει να αποθηκεύουμε προσωρινά στη μνήμη του υπολογιστή δεδομένα για περαιτέρω επεξεργασία κατά βούληση. Κάθε μεταβλητή έχει τέσσερα βασικά χαρακτηριστικά: τύπο (type), όνομα (name ή identifier), τιμή (value) και διάρκεια ζωής (scope). Και τα τέσσερα αυτά χαρακτηριστικά παίζουν σημαντικό ρόλο στη λειτουργία της κάθε μεταβλητής και καθορίζονται από τον προγραμματιστή (εκτός από την τιμή σε πολλές περιπτώσεις).

Η σύνταξη μιας απλής μεταβλητής στη Java είναι:

```
Τύπος μεταβλητής όνομα μεταβλητής;
```

Π.χ

```
int x;    // δήλωση μεταβλητής τύπου int  
  
double y; // δήλωση μεταβλητής τύπου double  
  
byte b;   // δήλωση μεταβλητής τύπου byte
```

Για να δηλώσουμε λοιπόν μία μεταβλητή θα πρέπει οπωσδήποτε να ενημερώσουμε τον compiler τι τύπου θα είναι για να γνωρίζει ακριβώς το μέγεθος μνήμης που θα πρέπει να δεσμεύσει. Άρα, για μία μεταβλητή τύπου int ο compiler θα δεσμεύσει τέσσερα bytes ενώ για μία μεταβλητή τύπου char θα δεσμεύσει δύο. Το όνομα μιας μεταβλητής είναι αυτό που χρησιμοποιούμε ως προγραμματιστές για να αναφερθούμε σε αυτήν μέσα από τον κώδικά μας, είτε για να διαβάσουμε την τιμή της είτε για να την θέσουμε. Το όνομα κάθε μεταβλητής αντιστοιχεί σε κάποιον δεκαεξαδικό αριθμό.

Ονομασία Μεταβλητών

Η ονομασία μεταβλητών θα πρέπει να ακολουθεί συγκεκριμένους κανόνες. Οι κανόνες αυτοί είναι:

- Τα ονόματα μεταβλητών μπορούν να περιέχουν μόνο γράμματα, δεκαδικά ψηφία και underscores (_)
- Δεν μπορούν να αρχίζουν με δεκαδικό ψηφίο
- Δεν επιτρέπεται η χρήση συμβόλων όπως τα &, #, @ κλπ
- Επιτρέπεται η χρήση συμβόλων νομισμάτων (\$, €, £) αλλά σύμφωνα με τους κανόνες σωστής πρακτικής δε θα πρέπει να χρησιμοποιούνται
- Το όνομα μιας μεταβλητής δε μπορεί να περιέχει κενά
- Δεν μπορούν να χρησιμοποιηθούν ως ονόματα μεταβλητών δεσμευμένες λέξεις (π.χ. long return;)

Αναφορές (References)

Η δημιουργία αντικείμενων γίνεται δυναμικά με τη χρήση του τελεστή `new` και η πρόσβαση στα αντικείμενα αυτά παρέχεται μέσω μιας ειδικής κατηγορίας μεταβλητών που ονομάζονται αναφορές. Για να δημιουργήσουμε ένα αντικείμενο π.χ. της κλάσης `MyClass` στη Java, θα γράφαμε:

```
MyClass mc = new MyClass();
```

Το `mc` είναι μία αναφορά. Μέσω της αναφοράς αυτής έχουμε πρόσβαση στο αντικείμενο τύπου `MyClass` που δημιουργήθηκε και στο οποίο δείχνει.

Χρησιμοποιούμε λοιπόν την αναφορά για να καλέσουμε π.χ. κάποια μέθοδο του αντικειμένου. Για να είναι ικανή η αναφορά να 'δείξει' σε κάποιο αντικείμενο της συγκεκριμένης κλάσης, θα πρέπει να έχει τον ίδιο τύπο με αυτόν που ορίζεται από την κλάση. Κατά τα άλλα, η σύνταξη δήλωσης μιας αναφοράς είναι ακριβώς η ίδια με αυτή των απλών μεταβλητών βασικών τύπων, δηλαδή:

Τύπος αναφοράς όνομα αναφοράς; Η τιμή των αναφορών στη Java εξαρτάται από την υλοποίηση της κάθε JVM και μπορεί να διαφέρει.

Αρχικές Τιμές Μεταβλητών

Κατά τη δήλωση μεταβλητών και αναφορών και ανάλογα με το σημείο του κώδικα που γίνεται η δήλωση αυτές μπορεί να αρχικοποιηθούν με default τιμές ή όχι.

Μεταβλητές μέλη κλάσης (instance variables)

α. Μεταβλητές βασικού τύπου

Οι μεταβλητές μέλη μιας κλάσης που είναι βασικοί τύποι, αρχικοποιούνται πάντοτε αυτόματα από τον compiler με default τιμές,.

β. Αναφορές

Οι αναφορές που είναι δηλωμένες ως μεταβλητές μέλη μιας κλάσης, αρχικοποιούνται πάντοτε κατά τη δημιουργία ενός αντικειμένου με την τιμή null.

Τοπικές μεταβλητές (local – automatic)

α. Μεταβλητές βασικού τύπου

Δεν αρχικοποιούνται αυτόματα με καμία τιμή. Μάλιστα, θα πρέπει για να μας αποτρέψει ο compiler να μεταγλωττίσουμε το πρόγραμμα σωστά, να τις έχουμε αρχικοποιήσει με κάποια τιμή πριν τις χρησιμοποιήσουμε (π.χ. σε μία σύγκριση). Σε αντίθετη περίπτωση θα παραχθεί compiler error.

β. Αναφορές

Δεν αρχικοποιούνται αυτόματα με καμία τιμή, ούτε καν με την τιμή null. Απλά θεωρείται πως δεν περιέχουν τίποτα. Και αυτές θα πρέπει να αρχικοποιηθούν με κάποια τιμή (πραγματικό αντικείμενο ή null) πριν τη χρήση τους, αλλιώς θα παραχθεί compiler error.

Γενικά, μέλη μιας κλάσης (είτε βασικού τύπου είτε αναφορές) αρχικοποιούνται πάντοτε με κάποια default τιμή, ενώ οι τοπικές μεταβλητές θα πρέπει να αρχικοποιηθούν από τον προγραμματιστή.

Σταθερές (constant)

Οι μεταβλητές ονομάζονται έτσι, μιας και η τιμή τους μπορεί να αλλάξει πολλές φορές κατά τη διάρκεια εκτέλεσης ενός προγράμματος. Υπάρχουν όμως και περιπτώσεις που στο πρόγραμά μας θέλουμε να αναπαραστήσουμε μία τιμή που παραμένει σταθερή καθ' όλη τη διάρκεια του. Για να ορίσουμε μία σταθερά στη

Java, απλά προσθέτουμε τη δεσμευμένη λέξη `final` στη σύνταξη που γνωρίζουμε για τη δήλωση μιας απλής μεταβλητής, π.χ

```
final double PI = 3.14159;
```

Η δήλωση μιας σταθεράς γίνεται πάντοτε σε μία γραμμή, στην οποία θα πρέπει να θέσουμε και την τιμή της. Στο παραπάνω παράδειγμα ορίζουμε μία σταθερά τύπου `double` με όνομα `PI` και της θέτουμε την τιμή `3.14159`. Από αυτήν τη γραμμή και μετά, οποτεδήποτε αναφερθούμε στην `PI` κατά τη διάρκεια εκτέλεσης του προγράμματος, αυτή θα μας επιστρέψει την τιμή `3.14159`. Οποιαδήποτε απόπειρα μεταβολής της τιμής της `PI` θα έχει ως αποτέλεσμα να παραχθεί σφάλμα στη μεταγλώττιση.

Τελεστές (Operators)

Κάθε γλώσσα προγραμματισμού μας παρέχει έναν αριθμό από 'ειδικά' σύμβολα τα οποία μπορούμε να χρησιμοποιήσουμε για να εκτελέσουμε πράξεις και να τροποποιήσουμε τα δεδομένα μας. Τα σύμβολα αυτά ονομάζονται τελεστές (operators) και μας επιτρέπουν να φέρουμε σε πέρας μεγάλο πλήθος από διεργασίες, όπως για παράδειγμα αριθμητικές πράξεις. Η τυπική σύνταξη ενός statement που κάνει χρήση ενός τελεστή είναι της μορφής: `a τ b` ή `τ a`, όπου 'τ' είναι ο τελεστής (operator) και `a`, `b` οι τελεσταίοι (operands). Ανάλογα με τον αριθμό των τελεστών, στους οποίους εφαρμόζονται, οι τελεστές χωρίζονται στις εξής κατηγορίες:

- Μοναδιαίοι (unary): εφαρμόζονται σε έναν τελεσταίο (π.χ. ο τελεστής `!` ή το πρόσημο `+`).
- Δυαδικοί (binary): εφαρμόζονται σε δύο τελεσταίους και είναι οι πιο κοινοί. Χαρακτηριστικό παράδειγμα είναι οι τελεστές των γνωστών μας αριθμητικών πράξεων π.χ. αφαίρεσης (`-`), διαίρεσης (`/`) κλπ.
- Τριαδικοί (ternary): εφαρμόζονται σε τρεις τελεσταίους. Υπάρχει μόνο ένας τέτοιος τελεστής.

Επίσης, ανάλογα με τη σημασία τους, χωρίζονται σε αριθμητικούς (arithmetic), σχεσιακούς (relational), λογικούς (logical), επιπέδου bit (bitwise), ανάθεσης (assignment), αύξησης (increment), μείωσης (decrement) και αντικατάστασης (compound assignment). Σε ένα statement που περιλαμβάνει χρήση τελεστών, οι τελευταίοι μπορεί να είναι είτε σταθερές, είτε μεταβλητές, μία κλήση μεθόδου που επιστρέφει τιμή, ή μια πολύπλοκη μαθηματική έκφραση με παρενθέσεις που τελικά θα αποτιμηθεί σε μία και μόνο τιμή.

JAVA Τελεστές(operators)

- Μοναδιαίοι τελεστές π.χ. count++
- Δυαδικοί τελεστές π.χ. num1+num2
- Η Java έχει μόνο ένα τριαδικό τελεστή τον ?

Π.χ. int a,b;

....

```
b= a>100?60:4
```

Ισοδύναμο με

```
if(a>100)
```

```
    b=60
```

```
else
```

```
    b=4;
```

Εκπαιδευτική Υποενότητα 7.4 Εισαγωγή στην ανάπτυξη εφαρμογών για Android
Ορισμός του Android

Πώς ορίζετε το Android;

Android στην πραγματικότητα είναι κάτι περισσότερο από ένα απλό λειτουργικό σύστημα, είναι μια φορητή πλατφόρμα που παρέχεται από την Google, η οποία αποτελείται από:

- Λειτουργικό σύστημα (Linux Kernel).

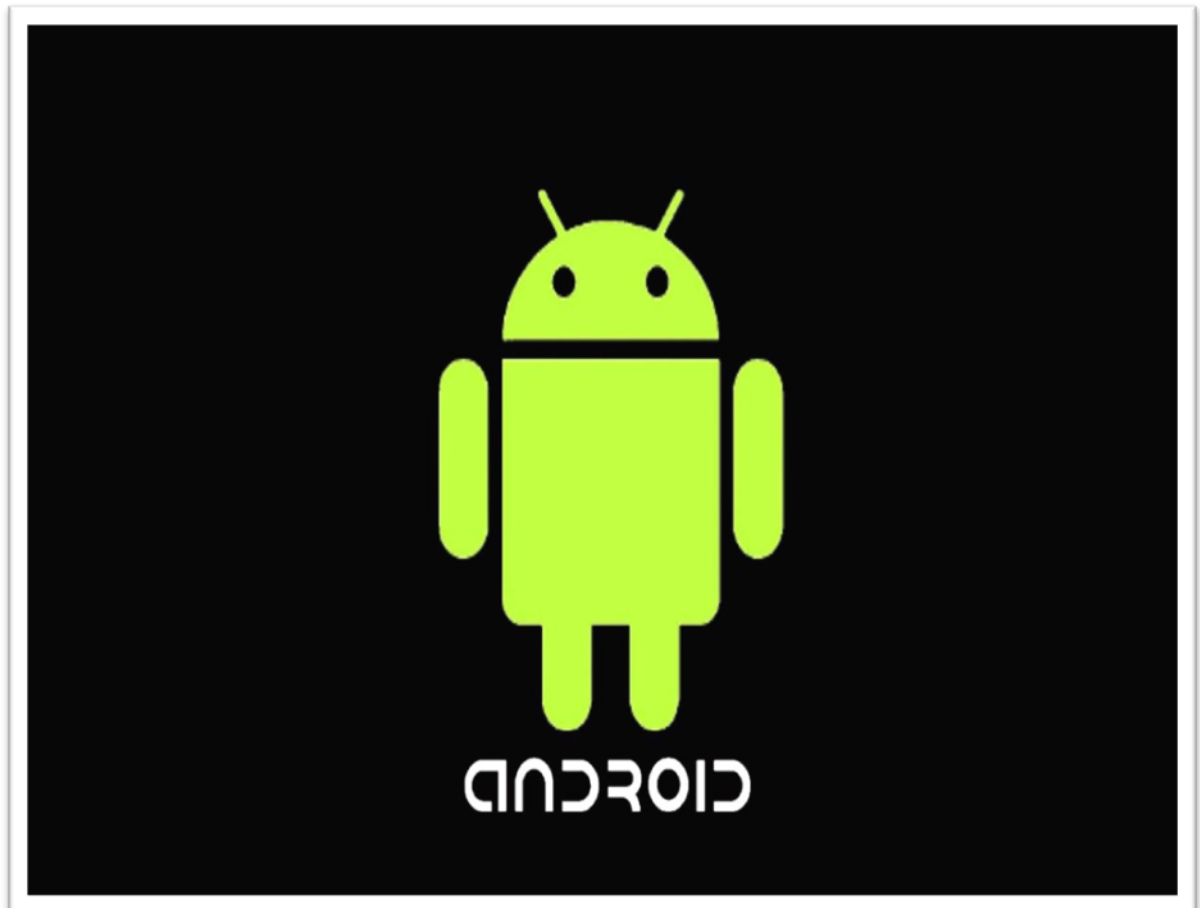
- Middleware (ένα σύνολο βιβλιοθηκών που παρέχονται από την Google).
- Βασικές εφαρμογές (Instagram, Facebook, Gmail, Angry birds ... κ.λπ.).

Το Android δηλαδή είναι ένα λειτουργικό σύστημα για κινητές συσκευές, όπως τα smartphones και τα tablets και είναι βασισμένο στα Linux. Όλες οι λειτουργίες του κινητού τηλεφώνου ελέγχονται από τις εφαρμογές (apps), τις οποίες ο χρήστης μπορεί να κατεβάσει και να εγκαταστήσει ελεύθερα. Αξίζει να σημειωθεί ότι το Android είναι ένα ελεύθερο λογισμικό ανοικτού κώδικα (free and open source software) πράγμα που επιτρέπει στους χρήστες την ανάπτυξη δωρεάν παιχνιδιών και κάθε είδους εφαρμογής. Αυτή ακριβώς η δυνατότητα συντέλεσε στην ταχεία διάδοση του. Επιπλέον, τα εργαλεία για να αναπτύξει εφαρμογές οποιοσδήποτε προγραμματιστής μπορεί να τα βρει δωρεάν στο Internet.

Το android απαρτίζεται από ένα πλαίσιο βασικού λογισμικού και ένα γραφικό περιβάλλον, δίνοντας τη δυνατότητα σε χρήστες με συμβατές συσκευές να εγκαταστήσουν πολλές διαφορετικές εφαρμογές. Το Android αναπτύχθηκε από την εταιρεία Android Inc. που εξαγοράστηκε πλήρως από την Google τον Αύγουστο του 2005. Επιτρέπει στους κατασκευαστές λογισμικού να συνθέτουν κώδικα με την χρήση της γλώσσας προγραμματισμού Java, ελέγχοντας την συσκευή μέσω βιβλιοθηκών λογισμικού ανεπτυγμένων από την Google. Τον Νοέμβριο του 2007, έκανε την εμφάνισή της η Open Handset Alliance, μια κοινοπραξία 48 τηλεπικοινωνιακών εταιρειών, εταιρειών λογισμικού και κατασκευής hardware, ανάμεσα τους ονόματα όπως LG, Intel, HTC, Motorola και NVidia. Στόχος της ήταν η δημιουργία ανοικτών προτύπων για κινητές συσκευές. Παράλληλα, δημοσίευσαν το πρώτο τους προϊόν, το Android, μια πλατφόρμα βασισμένη στον πυρήνα του Linux (Linux kernel). Η Google έδωσε στην δημοσιότητα και το μεγαλύτερο μέρος του πηγαίου κώδικα του Android υπό τους όρους της Apache License.

Το λειτουργικό αυτό σύστημα “τρέχει” στον πυρήνα του λειτουργικού Linux και χρησιμοποιώντας τη δική του εργαλειοθήκη ανάπτυξης συστήματος λογισμικού (Software Development Kit), επιτρέπει στους κατασκευαστές να δημιουργήσουν πρωτοποριακές εφαρμογές. Το Android, επιτρέπει στους κατασκευαστές λογισμικού

να συνθέτουν κώδικα με τη χρήση της γλώσσας Java, ελέγχοντας τη συσκευή μέσω βιβλιοθηκών λογισμικού, που έχουν αναπτυχθεί από τη Google. Δεδομένου ότι, βασίζεται στον ισχυρό κινητήρα Java, η Linux- based πλατφόρμα Android, επιτρέπει την πλήρη πρόσβαση σε όλη τη λειτουργικότητα των κινητών συσκευών.



Η πλατφόρμα Android περιλαμβάνει ένα λειτουργικό σύστημα, το ενδιάμεσο λογισμικό και αρκετές βασικές εφαρμογές και επιτρέπει στους προγραμματιστές τη δημιουργία νέων και καινοτόμων εφαρμογών, που αξιοποιούν πλήρως όλες τις δυνατότητες μιας συσκευής, η οποία συνδέεται στο ίντερνετ ή απευθείας με τα πληροφοριακά συστήματα και τις βάσεις δεδομένων.

Σχεδιασμός

Το Android αποτελείται από έναν kernel βασισμένο σε αυτόν του Linux με το middleware, τις βιβλιοθήκες και τα APIs να είναι γραμμένα σε C και το software των εφαρμογών που τρέχει πάνω σε ένα πλαίσιο (applications framework) που περιλαμβάνει βιβλιοθήκες συμβατές με την Java. Το Android χρησιμοποιεί την

εικονική μηχανή Dalvik (Dalvik virtual machine). Η κύρια πλατφόρμα του hardware είναι η ARM αρχιτεκτονική που χρησιμοποιείται σεευρέως σε 32-bit συστήματα.



Συστατικά στοιχεία εφαρμογής

Με τον όρο αυτό εννοούμε τα απαραίτητα δομικά στοιχεία μιας εφαρμογής Android. Το κάθε στοιχείο στην ουσία είναι ένας τρόπος πρόσβασης του λειτουργικού συστήματος στην εφαρμογή μας. Μπορούμε να τα διακρίνουμε σε τέσσερα βασικά στοιχεία : τις δραστηριότητες (activities), τις υπηρεσίες (services), τους παρόχους περιεχομένου (content providers), και τους καθολικούς παραλήπτες μηνυμάτων (broadcast receivers).

Χαρακτηριστικά του λειτουργικού συστήματος Android

Το λειτουργικό σύστημα Android έχει τα παρακάτω χαρακτηριστικά :

- Υλικό: Παρέχει συμβατότητα με κάμερες, οθόνες αφής, GPS, αισθητήρες επιτάχυνσης, επιταχυντές γραφικών (δισδιάστατους ή τρισδιάστατους).
- Λειτουργία Οθόνης: Προσαρμόζει τις διεπαφές των εφαρμογών σε διαφορετικές αναλύσεις και διαστάσεις. Επίσης, ρυθμίζει και τον προσανατολισμό της οθόνης κατά τη χρήση.

- Υποστήριξη Πολυμέσων: Το Android υποστηρίζει όλους τους τύπους πολυμέσων (ήχο, εικόνα, βίντεο).
- Διαθέσιμες Εφαρμογές: Ένα σύνολο εφαρμογών σε μεγάλη ποικιλία είναι διαθέσιμο για τις συσκευές με λειτουργικό σύστημα Android.
- Αποθήκευση Δεδομένων: Το σύστημα διαχείρισης βάσεων δεδομένων SQLite χρησιμοποιείται από τις εφαρμογές για την αποθήκευση των δεδομένων.
- Συνδεσιμότητα: Το Android υποστηρίζει όλες τις σύγχρονες τεχνολογίες για ενσύρματη και ασύρματη δικτύωση.
- Διαδίκτυο: Η πλοήγηση στο διαδίκτυο γίνεται με ενσωματωμένο φυλλομετρητή (browser).
- Υποστήριξη Java: Η μεταγλώττιση και η εκτέλεση των εφαρμογών που είναι γραμμένες σε Java πραγματοποιούνται στην εικονική μηχανή Dalvik.

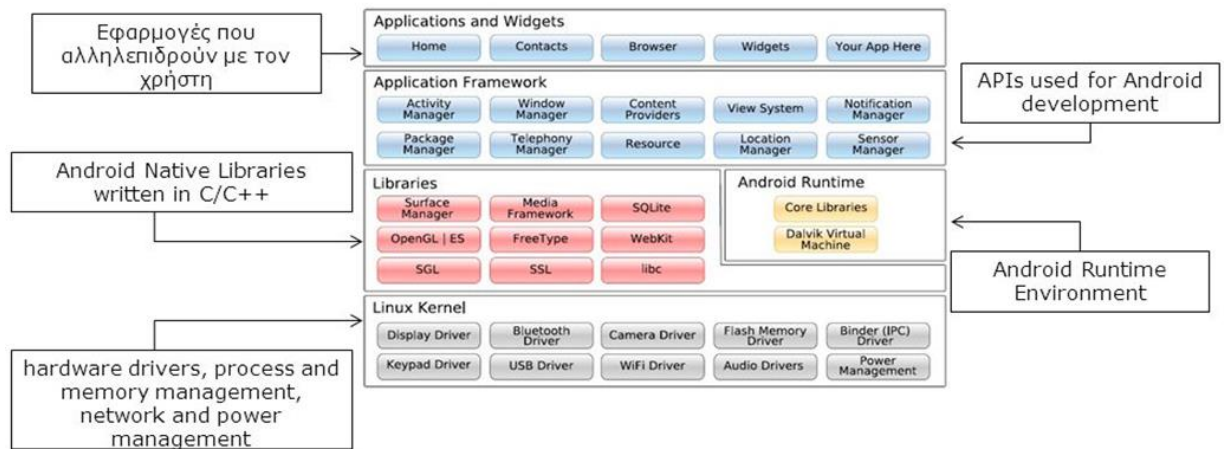
Η αρχιτεκτονική του Android

Η βασική αρχή της αρχιτεκτονικής Android είναι η επαναχρησιμοποίηση κώδικα, με την οποία δίνεται η δυνατότητα δημοσίευσης και μοιράσματος δραστηριοτήτων, υπηρεσιών και δεδομένων με άλλες εφαρμογές, στα πλαίσια της ασφάλειας.

Το android δεν είναι απλά ένα λειτουργικό σύστημα, είναι μια στοίβα λογισμικού, που αποτελείται από:

- Το λειτουργικό σύστημα
- Τις υπηρεσίες διασύνδεσης με τις εφαρμογές (middleware) και
- Τις κύριες εφαρμογές (core), όπως ο email-client, η εφαρμογή διαχείρισης SMS, το ημερολόγιο, ο browser, η εφαρμογής διαχείρισης επαφών και άλλες προεγκατεστημένες.

Αρχιτεκτονική Android



Η αρχιτεκτονική του Android παρέχει:

- Κανένα περιορισμό στις εφαρμογές
- Παράλληλη λειτουργία εφαρμογών
- Ισότητα για κάθε εφαρμογή

Οι στοίβα του Android αποτελείται από 5 επίπεδα και είναι τα εξής:

A. Επίπεδο εφαρμογών (Applications)

Μερικές βασικές εφαρμογές που περιλαμβάνονται στο σύστημα Android, όπως οι email client, πρόγραμμα για SMS μηνύματα, ημερολόγιο, χάρτες (google maps), περιηγητής ιστού και πρόγραμμα για δομημένη αποθήκευση επαφών, χρησιμοποιούν τη Java ως γλώσσα προγραμματισμού.



B. Επίπεδο πλαισίου εφαρμογών (Application Framework)

Αυτό το επίπεδο παρέχει υψηλού επιπέδου δομικές μονάδες που δίνουν τη δυνατότητα στους προγραμματιστές να κατασκευάσουν νέες καινοτόμες εφαρμογές. Με τη δυνατότητα των προγραμματιστών να εκμεταλλεύονται το hardware των συσκευών, έχουν πρόσβαση σε υπηρεσίες εντοπισμού θέσης, δυνατότητα να θέτουν χρονοδιακόπτες για εμφάνιση ειδοποιήσεων, πλήρη πρόσβαση στο ίδιο πλαίσιο από APIs, που έχουν βασικές εφαρμογές του Android.

Η διαμόρφωση της αρχιτεκτονικής αυτού του επιπέδου γίνεται έτσι, ώστε κάθε εφαρμογή να μπορεί να χρησιμοποιεί τις δυνατότητες μιας άλλης εφαρμογής και ταυτόχρονα ο χρήστης να επεμβαίνει στα συστατικά της καθημίας.

APPLICATION FRAMEWORK

- The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.



12/3/2013

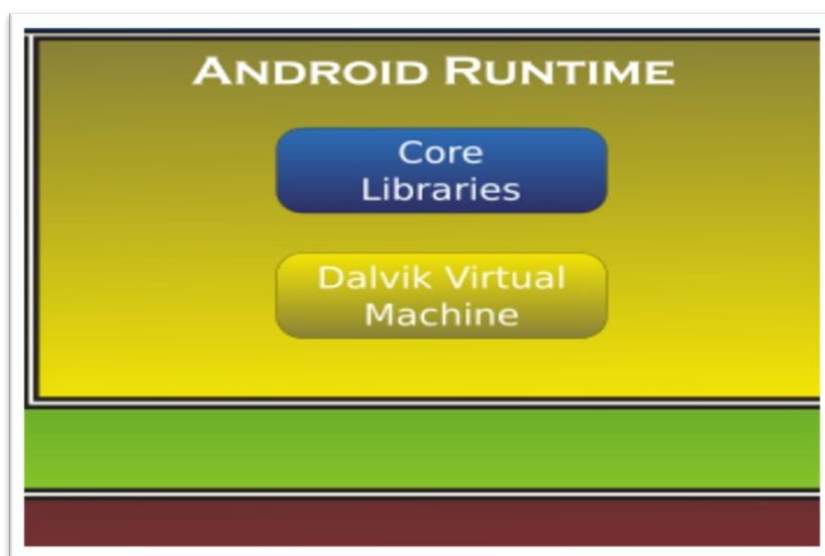
Umbani Bhavik AND Surani Saunak(ImperialInfoSys) 10

Γ. Επίπεδο χρόνου εκτέλεσης εφαρμογής (Android Runtime)

Το επίπεδο αυτό αποτελείται από ένα σύνολο βασικών βιβλιοθηκών και τη Dalvik Virtual Machine. Το σύστημα Android εκτελεί κάθε εφαρμογή ως μια ενιαία διαδικασία μέσα σε μια αποκλειστική παρουσία εικονικής μηχανής που ονομάζεται Dalvik Virtual Machine (DVM). Είναι μια εικονική μηχανή Java για φορητές συσκευές, από τη Google. Σε αυτή εκτελείται ο κώδικας bytecode των εφαρμογών. Για να εκτελεστεί ο κώδικας μέσα στο DVM, ο κώδικας της εφαρμογής πρέπει να μεταγλωττιστεί από τυπικό κώδικα Java (μορφή αρχείου .java) σε κώδικα byte

Dalvik (μορφή αρχείου .dex). Μετά από αυτό, ο κώδικας θα μεταγλωττιστεί για δεύτερη φορά στη μηχανή του δυαδικού κώδικα.

Κάθε εφαρμογή δεν έχει επαφή με άλλη, παρόλο που εκτελούνται ταυτόχρονα. Με τη χρήση της εικονικής μηχανής, εκτελούνται αρχεία “.dex”, τα οποία βρίσκονται σε συμπιεσμένη μορφή, εξοικονομώντας χώρο στη μνήμη και δίνοντας τη δυνατότητα να ‘τρέχουν’ πολλές εικονικές μηχανές ταυτόχρονα στο σύστημα. Ένα από τα χαρακτηριστικά του Android Runtime, είναι η ύπαρξη του JIT (Just in time), μεταφραστής, που χρησιμοποιεί τη μετάφραση από bytcodes σε κώδικα μηχανής, για να αυξηθεί η ταχύτητα εκτέλεσης των αντίστοιχων τμημάτων της εφαρμογής. Ο μηχανισμός JIT υιοθετήθηκε λόγω του τύπου των κινητών συσκευών που ήταν διαθέσιμες εκείνη την εποχή (πριν από το 2013 δεν είχαμε τηλέφωνο 6 Gig RAM).



Δ. Επίπεδο βιβλιοθηκών (Libraries)

Οι βιβλιοθήκες είναι γραμμένες σε γλώσσα προγραμματισμού C και C++ και μεταγλωττίστηκαν για το συγκεκριμένο υλικό που χρησιμοποιείται από διάφορα συστήματα Android. Οι βιβλιοθήκες Android βασίζονται σε JAVA που χρησιμοποιείται ειδικά για την ανάπτυξη Android όπως: προβολές, πρόσβαση σε βάσεις δεδομένων και έλεγχος πολυμέσων... κ.λπ.

Οι κύριες βιβλιοθήκες που συναντώνται στο Android είναι:

- System C Library:

Πρόκειται για μια ενσωμάτωση της standard βιβλιοθήκης συστήματος της C, η οποία έχει τροποποιηθεί ώστε να είναι κατάλληλη για κινητές συσκευές που βασίζονται στο Linux. Βασικός στόχος είναι η μείωση της απαιτούμενης μνήμης.

- Surface Manager:

Αναφέρεται στο υποσύστημα προβολής συνθέτοντας πολυδιάστατα επίπεδα γραφικών που προέρχονται από άλλες εφαρμογές. Είναι ο διαχειριστής σχεδιαστικών επιφανειών. Μια σχεδιαστική επιφάνεια είναι η οθόνη της συσκευής ή η περιοχή της μνήμης που έχει δεσμευθεί για να κρατήσει τα δεδομένα προς απεικόνιση.

- SQLite:

Πρόκειται για μια πολύ ισχυρή βάση δεδομένων. Αφορά μια μηχανή ανοιχτού κώδικα διαχείρισης σχεσιακών βάσεων δεδομένων που παρέχει ένα μηχανισμό για ανάγνωση, εγγραφή, τροποποίηση και άλλες διαδικασίες στα δεδομένα

- Βιβλιοθήκη 3D:

Οι βιβλιοθήκες αυτές χρησιμοποιούν τρισδιάστατη επιτάχυνση υλικού και λογισμικού που βασίζεται στα APIs του OpenGL ES 1.

- SGL:

Πρόκειται για μια μηχανή που προσφέρει τη δυνατότητα σχεδίασης τρισδιάστατων γραφικών.

- Free Type:

Προσφέρει ευκρίνεια στα γραφικά που χρησιμοποιούνται στα bitmaps και στις γραμματοσειρές των εφαρμογών. Συγκεκριμένα, η ποιότητα απεικόνισης μιας διανυσματικής γραμματοσειράς δε μειώνεται καθώς αυξάνεται το μέγεθός της.

- LibWebCore:

Υποστηρίζει την πλοήγηση στο διαδίκτυο και χρησιμοποιείται από το browser του android και τις web views που είναι ενσωματωμένες στις εφαρμογές.

- **WebKit:**

Είναι η μηχανή διάταξης γραφικών που δίνει τη δυνατότητα στο πλοηγό του διαδικτύου να απεικονίσει τις σελίδες, που ο χρήστης επισκέπτεται με γραφικό τρόπο με το κώδικα HTML, που του παρέχει τους ιστότοπους.

- **Βιβλιοθήκες Πολυμέσων:**

Υποστηρίζει την αναπαραγωγή και την εγγραφή πολλών μέσων εικόνας και ήχου. Αποτελεί το λεγόμενο Media Framework του Android που είναι υπεύθυνο για τη σωστή αναπαραγωγή ήχου και βίντεο σε μια συσκευή.



Ε. Επίπεδο πυρήνα Linux

Το λογισμικό του Android βασίζεται στον πυρήνα Linux έκδοση 2.6 και του επιτρέπει να μπορεί να χρησιμοποιηθεί σε ένα μεγάλο εύρος πλατφόρμων στο μέλλον. Ο πυρήνας του linux είναι υπεύθυνος για τις κύριες δραστηριότητες οποιουδήποτε λειτουργικού συστήματος που μπορούν να συνοψιστούν σε 4 σημεία:

- **Διαχείριση μνήμης:**

Είναι η διαδικασία ελέγχου και συντονισμού της μνήμης της συσκευής με την ανάθεση κατάλληλων μπλοκ μνήμης σε διάφορα εκτελούμενα προγράμματα.

- Διαχείριση πόρων:

Αναφέρεται στους διαθέσιμους πόρους της συσκευής και στον τρόπο με τον οποίο μπορούν να κατανεμηθούν τέλεια στις διεργασίες που εκτελούνται για τη βελτιστοποίηση της συνολικής απόδοσης του συστήματος.

- Διαχείριση προγραμμάτων οδήγησης:

Ο κύριος σκοπός των προγραμμάτων οδήγησης συσκευών είναι να παρέχουν αφαίρεση λειτουργώντας ως μεταφραστής μεταξύ μιας συσκευής υλικού και του λειτουργικού συστήματος, βοηθώντας στην εκτέλεση προσαρτημένου υλικού (οθόνη, WI-FI και ήχος).

- Διαχείριση ενέργειας

Στο επίπεδο αυτό ανήκουν οι οδηγίες για την οθόνη, για τις κάρτες μνήμης, για τον ήχο, για το υποσύστημα ασύρματης σύνδεσης στο διαδίκτυο (Wi-Fi), για τη κάμερα και όποιο άλλο περιφερειακό συναντάμε στα κινητά τηλέφωνα.

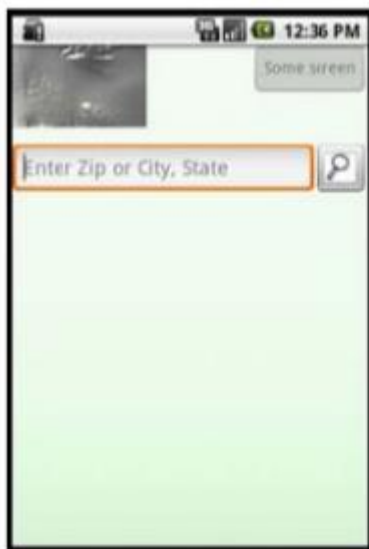


Πιο ειδικά χρησιμοποιείται για τη διαχείριση μνήμης, τη διαχείριση διεργασιών, τις λειτουργίες δικτύου, ασφάλεια λειτουργικού συστήματος και οδηγίες υλικού (hardware drives). Ο πυρήνας του Linux, δρα στη περίπτωση του Android, ως το επωνομαζόμενο επίπεδο αφαίρεσης υλικού (hardware abstraction layer).

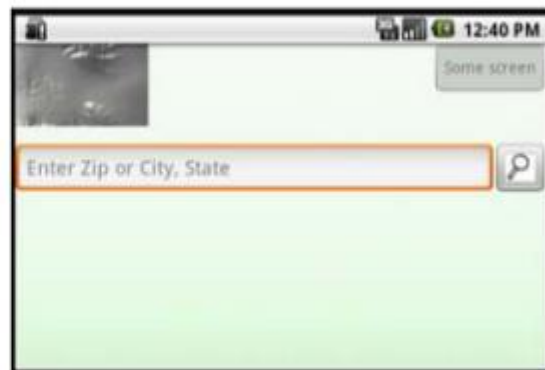
Προσανατολισμός (orientation)

Οι συσκευές με λειτουργικό σύστημα Android μπορούν να τοποθετηθούν είτε σε προσανατολισμό πορτρέτου (portrait mode) είτε σε προσανατολισμό τοπίου (landscape mode). Η πρώτη περίπτωση, αφορά και την πιο συνηθισμένη που

χρησιμοποιεί κανείς το κινητό του τηλέφωνο δηλαδή την όρθια θέση όπου η μεγαλύτερη πλευρά της οθόνης είναι η κατακόρυφη(Εικόνα Α). Η δεύτερη περίπτωση είναι η ακριβώς αντίστροφη όταν δηλαδή η μεγαλύτερη πλευρά της οθόνης είναι η οριζόντια. Είναι εύκολα κατανοητό λοιπόν ότι τα γραφικά στοιχεία που έχουμε σε κάθε οθόνη θα πρέπει να εμφανίζονται με διαφορετικό τρόπο κάθε φορά. Αυτό συμβαίνει γιατί η οθόνη στο landscape mode αντιστρέφεται (Εικόνα Β). Έχει περισσότερο πλάτος παρά μήκος. Αυτό σημαίνει ότι αν η αντιστροφή γινόταν με την προεπιλεγμένη συμπεριφορά, ο χρήστης μπορεί να δει πολλά κενά στη δεξιά πλευρά. Ο προεπιλεγμένος τρόπος είναι να παρουσιάζεται το user interface με την ίδια διάταξη και με τους δύο προσανατολισμούς.



Εικόνα Α



Εικόνα Β

Η διαφορά είναι εμφανής. Φαίνεται ότι στην «Εικόνα» το κείμενο «some text»είναι πολύ μακριά από την εικόνα της εφαρμογής καθώς και το στοιχείο EditText είναι πάρα πολύ μακρύ. Η διάταξη των γραφικών μας στοιχείων δηλώνεται αρχικά σε αρχεία xml. Συγκεκριμένα στο project μας υπάρχει ο φάκελος res/layout στο οποίο τοποθετούμε όλα τα αρχεία xml που αφορούν την εφαρμογή μας. Το λειτουργικό σύστημα Android έχει την δυνατότητα να διαχειριστεί αυτό το φαινόμενο με το να καταστρέφει την εφαρμογή και να την δημιουργεί ξανά ψάχνοντας κάθε φορά για το κατάλληλο user interface που θα χρησιμοποιήσει με την εντολή setContentView(). Έτσι ορίζοντας διαφορετικά αρχεία xml για την κάθε περίπτωση προσανατολισμού μπορούμε να έχουμε μια «όμορφη» εφαρμογή.

Εισαγωγή στις εφαρμογές Android

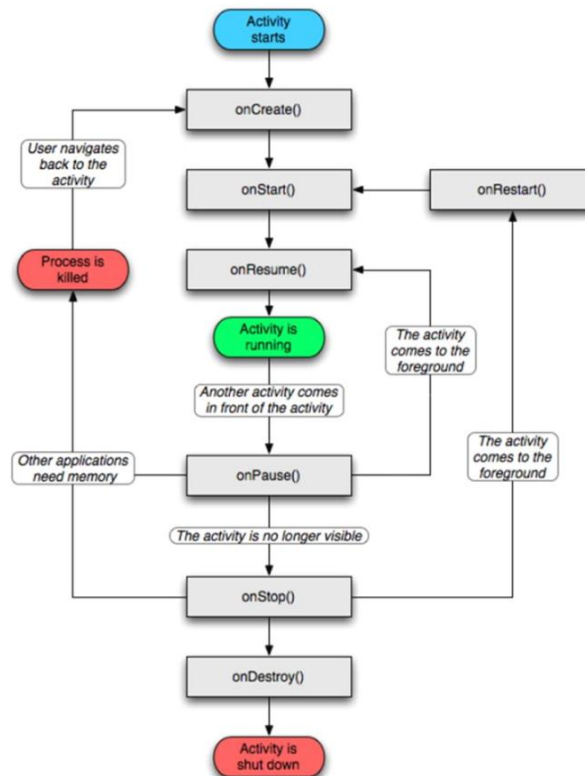
Οι εφαρμογές στο Android γράφονται στη γλώσσα προγραμματισμού Java. Τα εργαλεία του SDK μεταγλωττίζουν τον κώδικα μαζί με δεδομένα και αρχεία πόρων σε ένα Android package, που είναι ένα συμπιεσμένο αρχείο με την κατάληξη .apk. Ο κώδικας σε ένα .apk θεωρείται ως μια εφαρμογή και είναι το αρχείο που χρησιμοποιεί μια συσκευή για να εγκαταστήσει την εφαρμογή.

Μόλις εγκατασταθεί σε κάποια συσκευή, μια εφαρμογή τρέχει στο δικό της 'sandbox', ένα περιορισμένο για λόγους ασφάλειας περιβάλλον εκτέλεσης:

- Το λειτουργικό σύστημα δημιουργεί έναν χρήστη συστήματος για κάθε μια εφαρμογή.
- Προεπιλεγμένα το σύστημα θέτει σε κάθε εφαρμογή ένα μοναδικό ID χρήστη, το οποίο είναι άγνωστο στην εφαρμογή. Το σύστημα θέτει τα δικαιώματα για όλα τα αρχεία σε μια εφαρμογή, έτσι ώστε να μπορούν να προσπελαστούν μόνο από το ID που τίθεται στην εφαρμογή.
- Κάθε διεργασία τρέχει στη δική της VM έτσι ώστε ο κώδικας της να τρέχει απομονωμένος από τις άλλες εφαρμογές.
- Προεπιλεγμένα κάθε εφαρμογή τρέχει στη δική της διεργασία. Το Android εκκινεί αυτή τη διεργασία όταν οποιοδήποτε συστατικό της εφαρμογής πρέπει να εκτελεστεί και την παύει όταν δεν χρειάζεται ή όταν το σύστημα πρέπει να απελευθερώσει πόρους για άλλες εφαρμογές.

Κάθε εφαρμογή του Android δομείται από κάποια στοιχεία, που συνδέονται χρησιμοποιώντας ένα XML αρχείο, το οποίο περιγράφει κάθε στοιχείο και το τρόπο που αλληλεπιδρά με τα άλλα. Ποια είναι αυτά τα στοιχεία και τι σημαίνει το καθένα από αυτά:

- Δραστηριότητα (Activity): Αποτελεί το πιο κοινό από τα δομικά στοιχεία μιας εφαρμογής και είναι το επίπεδο παρουσίασης εφαρμογών (presentation layer). Πρόκειται για μια απλή οθόνη της εφαρμογής. Κάθε δραστηριότητα υλοποιείται σαν μια κλάση, που επεκτείνει τη Βασική Κλάση Δραστηριότητας (Activity Base Class). Στη συγκεκριμένη κλάση προβάλλεται μία διεπαφή χρήστη (user interface) που αποτελείται από όψεις (views) και ανταποκρίνεται σε συμβάντα (events). Ένα σύνολο από δραστηριότητες αποτελούν την εφαρμογή μας, αλλά κάθε δραστηριότητα είναι αυτοτελής και ανεξάρτητη η μία από την άλλη. Μια εφαρμογή, μπορεί να κάνει χρήση μιας ήδη υπάρχουσας Δραστηριότητας σε περίπτωση που το χρειαστεί, όπως η χρήση της φωτογραφικής μηχανής που υπάρχει ήδη σαν



εφαρμογή στο κινητό μας.

- Υπηρεσία (Service): Η υπηρεσία είναι ένας κώδικας που ‘τρέχει’ για μεγάλο χρονικό διάστημα και χωρίς διεπαφή χρήστη. Σκοπός της είναι να ενημερώνει τις πηγές δεδομένων και τις ορατές δραστηριότητες με την ενεργοποίηση ειδοποιήσεων. Π.χ η εφαρμογή media player μπορεί να συνεχίζει να παίζει μουσική ακόμη και αν το κύριο παράθυρο της εφαρμογής δε βρίσκεται στο προσκήνιο. Είναι συνηθισμένο, μια Δραστηριότητα να αναθέτει σε μια Υπηρεσία, μια εργασία

χρονοβόρα, προκειμένου να μην υπάρξει κάποια απώλεια στην ανταποκρισιμότητα της διεπαφής χρήστη.

- **Πάροχος περιεχομένου (Content Provider):** Ο πάροχος περιεχομένου, χρησιμοποιείται ώστε μια εφαρμογή να μοιράζεται με άλλες εφαρμογές, όταν χρειάζεται. Είναι μια κλάση, στην οποία υλοποιούνται μέθοδοι που επιτρέπουν στις εφαρμογές την αποθήκευση και την επαναφορά δεδομένων συγκεκριμένου τύπου, που χειρίζεται ο πάροχος περιεχομένου. Μερικά από τα συνηθέστερα δεδομένα που μοιράζονται μεταξύ των Content Providers, είναι οι επαφές του χρήστη και οι βάσεις δεδομένων SQLite μιας εφαρμογής. Τα δεδομένα μπορεί να είναι αποθηκευμένα στη κάρτα μνήμης του κινητού, σε ένα απομακρυσμένο εξυπηρετητή ή σε μία βάση δεδομένων.
- **Πρόθεση και φίλτρο προθέσεων (Intents):** Η πρόθεση είναι ένα αντικείμενο, το οποίο περιγράφει τι θέλει να κάνει μια εφαρμογή. Βασικά στοιχεία της είναι ποια ενέργεια θέλει η εφαρμογή για να εκτελεστεί και ποια δεδομένα θα χρησιμοποιήσει για να την εκτελέσει. Αντίθετα, το φίλτρο προθέσεως, είναι ένα αντικείμενο που περιγράφει τι είδους προθέσεις είναι δυνατόν να εξυπηρετηθούν.
- **Δείκτες μετάδοσης (Broadcast Receivers):** Πρόκειται για μια υπηρεσία που αντιλαμβάνεται ορισμένα γεγονότα του συστήματος και αναλαμβάνει να ενημερώσει το σύστημα και τις υπόλοιπες εφαρμογές. Δεν έχει γραφικό περιβάλλον και ενημερώνει το χρήστη μέσω της μπάρας ειδοποιήσεων. Τα διάφορα μηνύματα και οι ανακοινώσεις, στέλνονται από το ίδιο το λειτουργικό σύστημα ανά τακτά χρονικά διαστήματα.



Android Manifest

Όταν ο προγραμματιστής ξεκινήσει ένα νέο project δημιουργείται αυτόματα ένα αρχείο που ονομάζεται `AndroidManifest.xml`. Το `Android Manifest` περιέχει βασικές πληροφορίες σχετικά με την εφαρμογή, τις πληροφορίες που το λειτουργικό σύστημα πρέπει να κατέχει προτού τρέξει οποιοδήποτε άλλο κομμάτι κώδικα. Οι σημαντικότερες από αυτές τις πληροφορίες περιγράφονται παρακάτω :

- Η ονομασία του πακέτου της Java της εφαρμογής (Java package).
- Η έκδοση της εφαρμογής (πχ 1.0, 2.7, 3.2)
- Η ελάχιστη έκδοση του λειτουργικού συστήματος Android που απαιτεί η εφαρμογή (min SDK version). Για παράδειγμα, αν έχει δηλωθεί ως min SDK version το 7 που ισοδυναμεί με την έκδοση Android 2.1, τότε αυτή μπορεί να εκτελεστεί σε συσκευές με έκδοση Android 2.1 ή μεγαλύτερη καθώς υπάρχει συμβατότητα προς τα επάνω.
- Το όνομα της εφαρμογής καθώς και το εικονίδιο της.

- Οι άδειες (licenses) που απαιτούνται για να εκτελεστούν ορισμένες λειτουργίες της εφαρμογής. Για παράδειγμα αν η εφαρμογή μας χρησιμοποιεί το διαδίκτυο θα πρέπει να δηλώνεται και η αντίστοιχη άδεια. Το ίδιο ισχύει αν θέλουμε να έχει πρόσβαση στην SD card, αν θέλουμε να στέλνουμε μηνύματα κ.α. Αν δεν δηλώσουμε τις κατάλληλες άδειες τότε η εφαρμογή μας δεν θα δουλεύει όπως προβλέπεται και θα δημιουργεί σφάλμα. Όταν ο χρήστης πάει να εγκαταστήσει την εφαρμογή μας αυτές οι άδειες αναφέρονται και αν δεν συμφωνήσει τότε αυτή δεν θα εγκατασταθεί. Αυτός ο τρόπος είναι και ένα είδος ασφάλειας απέναντι σε κακόβουλες εφαρμογές που μπορεί να υπάρχουν.
- Όλα τα συστατικά στοιχεία (activities, services, content providers, broadcast receivers) της εφαρμογής. Για παράδειγμα, αν δημιουργήσουμε μια activity χωρίς να την δηλώσουμε στο Android Manifest.xml αυτή δε θα μπορέσει να λειτουργήσει.
- Οι εξωτερικές βιβλιοθήκες που χρησιμοποιεί η εφαρμογή μας. Για παράδειγμα, αν η εφαρμογή μας χρησιμοποιεί το google maps θα πρέπει να δηλωθεί η βιβλιοθήκη com.google.android.maps.

Βέλτιστος Σχεδιασμός

Μια εφαρμογή μπορεί λειτουργικά να έχει πετύχει τον σκοπό για τον οποίο σχεδιάστηκε, όμως είναι πάντα βέβαιο ότι υπάρχουν περιθώρια βελτίωσης, ώστε να γίνει γρηγορότερη, να απαιτεί λιγότερη μνήμη και να απευθύνεται σε ευρύτερο κοινό χωρίς να χάνει κάποιο στοιχείο από την λειτουργικότητά της. Για τον λόγο αυτό ο προγραμματιστής πρέπει να λαμβάνει υπόψη διάφορους παράγοντες, ώστε ο κώδικάς του να είναι βελτιστοποιημένος.

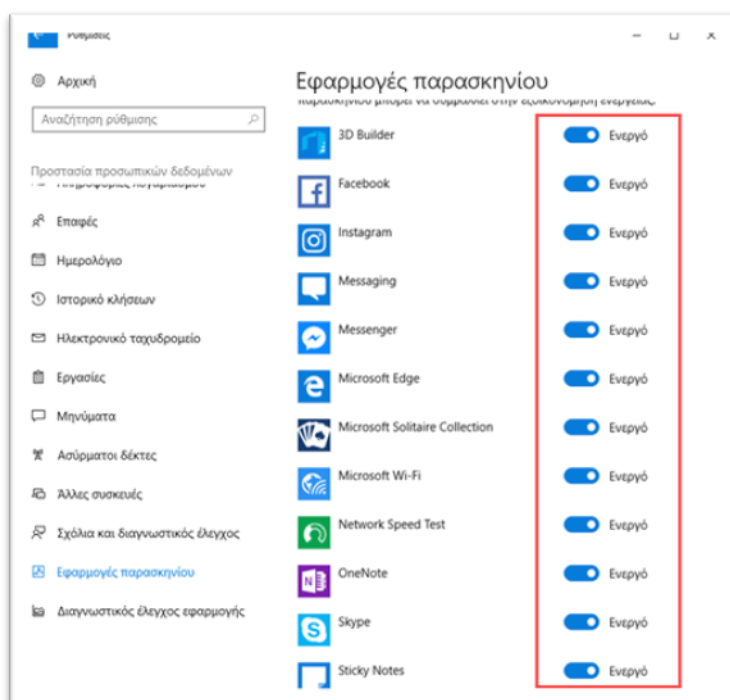
Κατηγορίες εφαρμογών

Εφαρμογές προσκηνίου

Μια εφαρμογή που χρησιμοποιείται μόνο όταν είναι στο προσκήνιο και αναστέλλεται όταν δεν είναι ορατή ανήκει σε αυτήν την κατηγορία. Όταν

αναπτύσσουμε εφαρμογές προσκηνίου (Foreground applications) πρέπει να έχουμε κατά νου τον κύκλο ζωής ενός Activity έτσι ώστε ένα Activity να μπορεί να εναλλάσσεται μεταξύ του παρασκηνίου και του προσκηνίου. Οι εφαρμογές που ανήκουν σε αυτήν την κατηγορία έχουν ελάχιστη ή και καθόλου αλληλεπίδραση με το χρήστη και πέραν της ρύθμισής τους τον περισσότερο χρόνο εκτέλεσής τους παραμένουν κρυφές. Τις περισσότερες φορές παρακολουθούν για μηνύματα ή ενέργειες που προκαλούνται από το υλικό, το σύστημα ή άλλες εφαρμογές αντί να βασίζονται στην αλληλεπίδραση με το χρήστη.

Μπορούμε να δημιουργήσουμε εντελώς αόρατες υπηρεσίες, αλλά στην πράξη είναι καλό να παρέχουμε ένα βασικό επίπεδο ελέγχου στο χρήστη. Κατ' ελάχιστον λοιπόν είναι καλό να επιτρέπουμε στους χρήστες να επιβεβαιώσουν ότι η υπηρεσία τρέχει και να τους αφήνουμε να ρυθμίζουν, να παύουν ή να την τερματίζουν όποτε επιθυμούν.



Διακοπτόμενες εφαρμογές Υπάρχουν περιπτώσεις που χρειάζεται μια εφαρμογή να δέχεται εισόδους από το χρήστη αλλά και να αντιδρά σε γεγονότα όταν δεν είναι στο προσκήνιο. Οι εφαρμογές της κατηγορίας αυτής λοιπόν είναι μια ένωση ορατών Activities και αόρατων υπηρεσιών παρασκηνίου ή broadcast receivers. Τέτοιες

εφαρμογές πρέπει να γνωρίζουν ανά πάσα στιγμή την κατάστασή τους όταν αλληλεπιδρούν με το χρήστη. Για παράδειγμα θα πρέπει να ενημερώνουν το UI όταν είναι ορατό και να στέλνουν notifications για να ενημερώνουν το χρήστη όταν είναι στο παρασκήνιο. Παραδείγματα τέτοιων εφαρμογών είναι αυτές για email ή για τη λήψη ειδήσεων.

Widgets και Live Wallpapers

Μερικές εφαρμογές αναπαρίστανται μόνο σαν ένα widget για το home-screen της συσκευής ή σαν live wallpapers. Δημιουργώντας μια τέτοια εφαρμογή, παρέχουμε διαδραστικά εικονικά στοιχεία με τα οποία μπορούμε να προσθέσουμε λειτουργικότητα στα home screens. Οι εφαρμογές μόνο με widgets είναι χρήσιμες στην παροχή δυναμικής πληροφορίας όπως για παράδειγμα τη στάθμη της μπαταρίας ή την πρόβλεψη του καιρού.

Ανάπτυξη εφαρμογών για Android

Η ανάπτυξη εφαρμογών στο Android είναι μια αρκετά χρονοβόρα διαδικασία. Τα βασικότερα στάδιά της περιγράφονται παρακάτω.

Δυνατότητα Πρόσβασης

Καθώς το Android θεωρείται ένα από τα «έξυπνα» λειτουργικά είναι πολλοί οι χρήστες που έχουν κάποιο είδος αναπηρίας, το οποίο τους αναγκάζει να χρησιμοποιούν την συσκευή με διαφορετικό τρόπο. Για παράδειγμα, είναι πιθανόν να μην μπορούν να χρησιμοποιήσουν αποτελεσματικά μια οθόνη αφής. Το Android περιλαμβάνει λειτουργίες που βοηθάνε τέτοιους χρήστες να πλοηγούνται πιο εύκολα στις συσκευές, όπως υπηρεσίες κείμενο-σε-ομιλία (text-to-speech) ή μια ιχνόσφαιρα (trackball) για πλοήγηση. Αρκούν ορισμένες ενέργειες από τον χρήστη ώστε η εφαρμογή να γίνει προσβάσιμη σε όλους τους δυνατούς χρήστες.

Πολλές συσκευές Android έχουν επιπλέον, clickable trackball (με δυνατότητα κλικ) ή clickable d-pad (σταυρό κατεύθυνσης) πέρα από τα βασικά κουμπιά. Έτσι ο χρήστης μπορεί να προηγηθεί πλήρως χωρίς να χρησιμοποιεί την οθόνη αφής. Δεν απαιτούνται από τον προγραμματιστή κάποια ενέργεια για την χρήση των παραπάνω αλλά οφείλει να το επιβεβαιώσει δοκιμάζοντας την εφαρμογή του σε τέτοιες συσκευές.

Άτομα με προβλήματα όρασης ίσως δεν έχουν την δυνατότητα να δουν όλα τα γραφικά στοιχεία της οθόνης κάθε εφαρμογής ώστε να καταλάβουν την χρήση του καθενός. Η παράμετρος android:contentDescription που παρέχεται από το λειτουργικό σύστημα και μπορούμε να την χρησιμοποιήσουμε σε όλα τα αρχεία xml λύνει το παραπάνω πρόβλημα. Αν ο χρήστης έχει ενεργοποιήσει τα εργαλεία βοηθητικής πρόσβασης στην συσκευή του, όταν εστιάσει σε κάποιο στοιχείο θα ακούει την φράση την οποία έχουμε δηλώσει. Για παράδειγμα το

```
<EditText  
    android:id="@+id/name"  
    android:contentDescription="@string/insert_name"/>
```

όταν γίνει focused από τον χρήστη θα ακουστεί η φράση (string) που έχει οριστεί στο αρχείο strings.xml.

Το Talk Back είναι μια εφαρμογή η οποία είναι προεγκατεστημένη σε πολλές συσκευές Android ή μπορεί απλά να την κατεβάσει κανείς από το Android Market. Με αυτήν την εφαρμογή ενεργοποιημένη, ο χρήστης μπορεί να ακούει την περιγραφή της οθόνης στην οποία βρίσκεται κάθε φορά που αλλάζει οθόνη. Συγκεκριμένα μέσα στις εφαρμογές ακούει όλα τα είδη κειμένου που έχει η οθόνη ώστε να ξέρει που βρίσκεται και ποιες είναι οι επιλογές του. Με αυτόν τον τρόπο, μαζί με αυτούς που περιεγράφηκαν πιο πάνω, θα γνωρίζουμε κάθε φορά την προσβασιμότητα της εφαρμογής μας και τις ενέργειες που πρέπει να κάνουμε ώστε να βελτιώσουμε την εμπειρία του χρήστη. Τέλος, επισημαίνεται ότι υπάρχουν και

εφαρμογές οι οποίες όταν ενεργοποιηθούν δίνουν την δυνατότητα στον χρήστη να εισάγει κείμενο σε πλαίσια μόνο με την ομιλία του (speech-to-text) χωρίς την χρήση του εικονικού πληκτρολογίου.

Εγκατάσταση λογισμικού

Το αρχικό στάδιο στην ανάπτυξη εφαρμογών είναι το στήσιμο του περιβάλλοντος εργασίας, στο οποίο πραγματοποιούνται ο σχεδιασμός, η ανάπτυξη, ο έλεγχος και η λειτουργία των εφαρμογών. Παρέχεται η δυνατότητα επιλογής περιβάλλοντος ανάπτυξης (IDE) και πολλαπλών εργαλείων του Android SDK. Στη συνέχεια είναι απαραίτητη η δημιουργία εικονικών συσκευών στη διαχείριση εικονικών συσκευών AVD και η δοκιμή της λειτουργίας της εφαρμογής σε εύρος πραγματικών συνθηκών λειτουργίας. Αρχικά, για να δημιουργήσουμε μια εφαρμογή για το λειτουργικό σύστημα Android πρέπει να χρησιμοποιήσουμε τη γλώσσα προγραμματισμού Java. Για να γίνει αυτό χρειαζόμαστε ένα απαραίτητο εργαλείο, το Java Development Kit (JDK), που πρέπει να το εγκαταστήσουμε στον υπολογιστή μας, μια διαδικασία εύκολη και σύντομη. Αυτό διατίθεται στο internet δωρεάν.

Εγκατάσταση του Java Development Kit (JDK)

Το JDK περιλαμβάνει ένα σύνολο εργαλείων ανάπτυξης προγραμμάτων για την γλώσσα προγραμματισμού Java όπως είναι ο java (φορτωτής Java εφαρμογών), ο javac (ο μεταφραστής του Java κώδικα σε Java bytecodes), ο javah (δημιουργός C stubs για συγγραφή native μεθόδων), ο jar (ο γνωστός Java archiver) και άλλα. Το JDK είναι το υποσύνολο του Java SDK που είναι απαραίτητο για τον προγραμματισμό και την εκτέλεση εφαρμογών σε Java. Το Java SDK περιλαμβάνει επιπλέον στοιχεία όπως application servers, επιπλέον debuggers και documentation.

Για την εγκατάστασή, επιλέγουμε από τον ιστότοπο το αρχείο εγκατάστασης που αντιστοιχεί στο λειτουργικό μας σύστημα, το αποθηκεύουμε και το εκτελούμε ακολουθώντας τα παρακάτω βήματα:

1. επιλέγουμε τα κομμάτια του JDK που θέλουμε να εγκαταστήσουμε καθώς και την διαδρομή στον δίσκο στην οποία θα πραγματοποιηθεί η εγκατάσταση.
2. Αναμένουμε να ολοκληρωθεί η αντιγραφή των νέων αρχείων και να γίνει η διαγραφή των backup αρχείων.
3. Μετά την ολοκλήρωση του βήματος 2 θα εμφανιστεί μήνυμα που θα ενημερώνει για την επιτυχή εγκατάσταση του JDK και θα ζητά να εγγραφούμε στην Oracle.
4. Στο επόμενο βήμα θα ξεκινήσει αυτόματα η εγκατάσταση του Java FX SDK.
5. Στο επόμενο πλαίσιο διαλόγου λαμβάνουμε κάποιες πληροφορίες για τον χώρο που θα δεσμεύσει η τρέχουσα εγκατάσταση και μας δίνεται η δυνατότητα να επιλέξουμε την διαδρομή στον δίσκο που αυτή θα πραγματοποιηθεί. Επιλέγουμε Next.
6. Αναμένουμε την ολοκλήρωση της εγκατάστασης.
7. Μετά την ενημέρωσή μας για την επιτυχή εγκατάσταση του Java FX SDK επιλέγουμε Close και η εγκατάσταση του JDK έχει ολοκληρωθεί επιτυχώς.

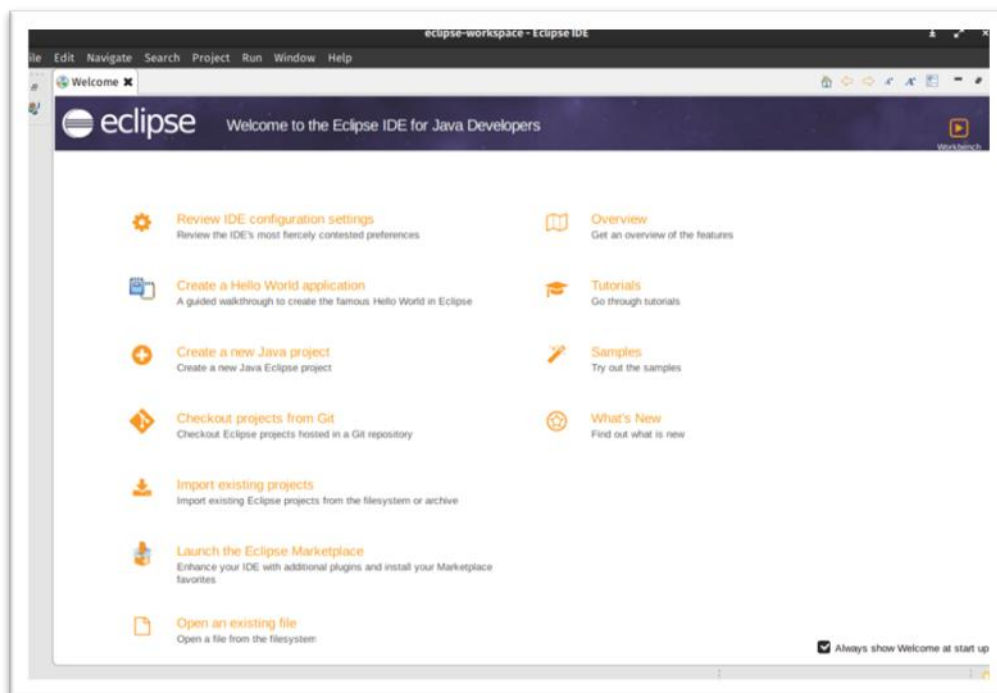


Εγκατάσταση του Eclipse IDE για Java

Το Eclipse IDE (Integrated Development Environment) είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης εφαρμογών. Δεν είναι απαραίτητο να έχει κάποιος ένα περιβάλλον ανάπτυξης προκειμένου να αναπτύξει εφαρμογές, ένα απλό σημειωματάριο αρκεί. Οι τεράστιες, όμως, διευκολύνσεις που παρέχει ένα τέτοιο

περιβάλλον το καθιστούν απαραίτητο για μεσαίου και μεγάλου μεγέθους εφαρμογές. Το Eclipse παρέχει ένα περιβάλλον για την οργάνωση του κώδικα ενός ολόκληρου project. Λειτουργίες όπως, διατήρηση εκδόσεων (versioning), διαμοιρασμός αλλαγών στον κώδικα μεταξύ των διαφόρων μελών της ομάδας ανάπτυξης, εικονικοί εξυπηρετητές για έλεγχο ορθότητας λειτουργίας web εφαρμογών και άλλες παρέχονται εντός του ίδιου του περιβάλλοντος. Επιπλέον, με την υποστήριξη των plug-in το Eclipse μπορεί να επεκτείνει τις λειτουργίες που προσφέρει ακόμα παραπέρα.

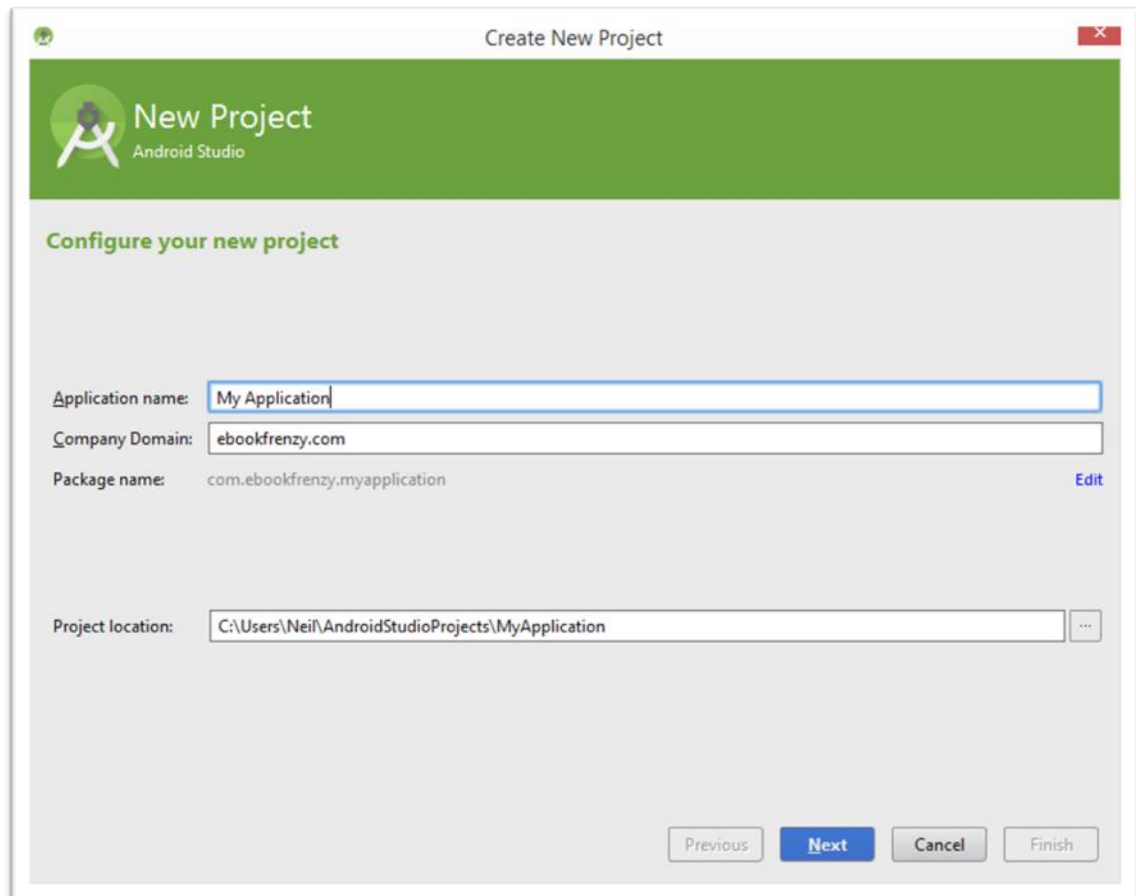
Η εγκατάσταση του Eclipse είναι πολύ απλή υπόθεση. Το μόνο που χρειάζεται να γίνει είναι να υποθηκευθεί ο φάκελος του Eclipse που λαμβάνουμε από τον ιστότοπο www.eclipse.org σε μια επιθυμητή διαδρομή στον δίσκο μας. Από εκεί και πέρα το πρόγραμμα μας είναι πλήρως λειτουργικό.



Ανάπτυξη πηγαίου κώδικα εφαρμογής

Στο στάδιο αυτό ο προγραμματιστής καλείται να αποφασίσει για τις δυνατότητες και το περιεχόμενο της εφαρμογής, να σχεδιάσει ένα λειτουργικό και όχι υπερβολικό layout (σχέδιο) και να καταλήξει σε ένα αρμονικό συνδυασμό του

κώδικα με το layout. Το πρώτο βήμα είναι η δημιουργία ενός project που θα περιλαμβάνει τον πηγαίο κώδικα, τα κείμενα και τις εικόνες. Βασικές προϋποθέσεις είναι να υπάρχει μία τάξη στο υλικό (εικόνες-κείμενα) για να είναι ο κώδικας εύκολα αναγνώσιμος.



Αποσφαλμάτωση (debugging)

Η αποσφαλμάτωση είναι μια πολύ σημαντική αλλά ταυτόχρονα χρονοβόρα διαδικασία. Πολύ συχνά συναντάται η επανάληψή της προκειμένου να διορθωθούν όλα τα σφάλματα που θα παρουσιαστούν στην εφαρμογή. Τα στάδιά της παρουσιάζονται παρακάτω:

- Πρώτο στάδιο: Η δημιουργία και η μεταγλώττιση της εφαρμογής σε debug mode, είναι το πρώτο στάδιο της αποσφαλμάτωσης. Κατά τη μεταγλώττιση

(compile) είναι σημαντική η ειδοποίηση του χρήστη σε περίπτωση εμφάνισης συντακτικού λάθους στον κώδικα. Στη συνέχεια, ακολουθεί η δοκιμή σε εικονική συσκευή (AVD MANAGER) ή σε φυσική συσκευή (ADB PUSH ΕΝΤΟΛΗ).

- Δεύτερο στάδιο: Σε αυτό το στάδιο, εντοπίζονται τα λειτουργικά προβλήματα της εφαρμογής, όπως εμφανίζονται στη συσκευή (εικονική ή φυσική), καθώς και η διόρθωση του κώδικα στις περιοχές που εμφανίζονται σφάλματα. Χρήσιμο εργαλείο της είναι το Log Cat.
- Τρίτο στάδιο: Σε αυτό το στάδιο, πραγματοποιείται η επανάληψη του πρώτου σταδίου από τον προγραμματιστή (μεταγλώττιση- δοκιμή εφαρμογής), με σκοπό να επιβεβαιωθούν τα αποτελέσματα της αποσφαλμάτωσης.
- Τέταρτο και τελευταίο στάδιο: Είναι το στάδιο της δημόσιας δοκιμαστικής φάσης που χρησιμοποιείται προαιρετικά στην αποσφαλμάτωση. Πρόκειται για μια εθελοντική δοκιμή της εφαρμογής σε συσκευή και βάση της χρήσης της να προκύψουν παρατηρήσεις, σχόλια ακόμα και προτάσεις για τη λειτουργία της.

Τελειοποίηση και διάθεση της εφαρμογής

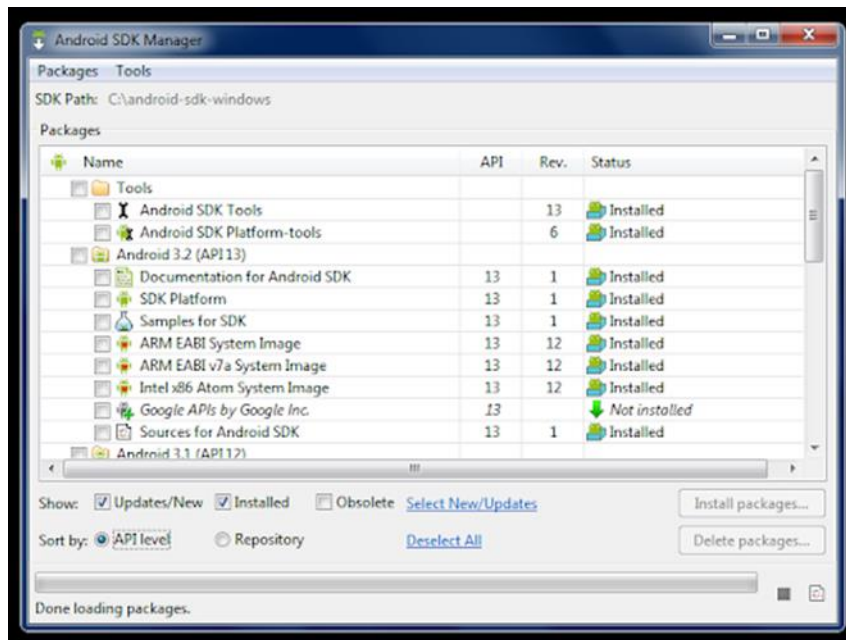
Είναι το τελευταίο στάδιο της ανάπτυξης εφαρμογών. Ο προγραμματιστής ακολουθεί τα παρακάτω βήματα:

- Διόρθωση των σφαλμάτων που παρουσιάστηκαν από την αποσφαλμάτωση
- Πραγματοποίηση των τελικών ρυθμίσεων και μετατροπών της εφαρμογής
- Οριστική μεταγλώττιση της εφαρμογής σε κανονική λειτουργία
- Διάθεση της εφαρμογής στο κοινό. Αυτό μπορεί να γίνει μέσω google play, είτε κάποιων εναλλακτικών καταστημάτων εφαρμογών (π.χ app store).

Παράμετροι για την ανάπτυξη μιας εφαρμογής Android

Λογισμικό ανάπτυξης (Android SDK)

Το Android SDK (Software Developers Kit) είναι το σύνολο των εργαλείων και βιβλιοθηκών που περιλαμβάνουν εξομοιωτές συσκευών (Android Virtual Machine), δείγματα κώδικα και χρησιμοποιούνται στο Android για την ανάπτυξη των εφαρμογών. Ορισμένα από τα πιο γνωστά λειτουργικά συστήματα που υποστηρίζει η SDK, είναι το Linux, το MAC OS X 10.5, τα Windows 7 και οι μεταγενέστερες εκδόσεις τους. Το Android SDK προσφέρεται με τη μορφή πακέτων. Κάποια από αυτά είναι τα SDK tools, SDK Platform-tools, Documentation, SDK Platform και System Images.



Εργαλεία Ανάπτυξης, Δοκιμής και Αποσφαλμάτωσης

Υπάρχουν διάφορα εργαλεία, τα οποία είναι απαραίτητα, για την ανάπτυξη, τη δοκιμή και την αποσφαλμάτωση των Android εφαρμογών. Αυτά είναι:

Android Debug Bridge (ADB)

Το Android Debug Bridge (ADB) είναι ένα πρόγραμμα client-server, που αναλαμβάνει τη σύνδεση του υπολογιστή με τις διάφορες συσκευές. Αναφέρεται συχνά σαν εργαλείο γραμμής εντολών και αποτελείται από:

- Τον client, ο οποίος 'τρέχει' στη συσκευή που είναι εγκατεστημένο το Android SDK
- Τον server, ο οποίος 'τρέχει' στη συσκευή που είναι εγκατεστημένο το Android SDK, όπως και ο client. Επίσης, εξασφαλίζει την επικοινωνία μεταξύ του client και του daemon.
- Ο daemon, ο οποίος 'τρέχει' σαν διεργασία παρασκηνίου στη συσκευή.

Κατά την εκκίνηση του Debug Bridge, ελέγχεται από τον server η ύπαρξη διεργασίας, διαφορετικά προχωρά στη δημιουργία μιας νέας. Μετά δημιουργεί μια τοπική TCP σύνδεση. Ακολουθεί, ο έλεγχος του εύρους των θυρών TCP, στο οποίο επικοινωνούν οι συσκευές εξομοίωσης και ελέγχει ποιες και αν είναι διαθέσιμες. Έτσι, οδηγούμαστε στη δημιουργία Debug Bridge σύνδεσης με τη συσκευή. Οι εντολές που εισάγουμε έχουν την εξής μορφή:

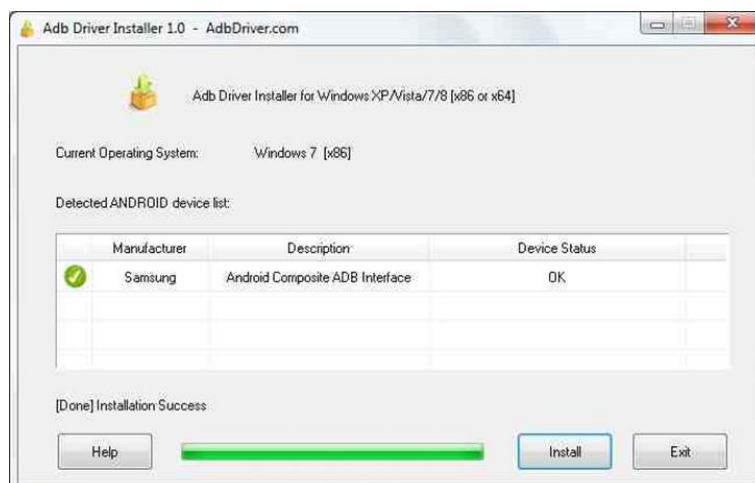
Adb [-d/-e/-s <serialNumber>] <command> όπου,

-d: εάν υπάρχει μία μόνο συνδεδεμένη συσκευή

-e: εάν υπάρχει μόνο μία εικονική συσκευή

-s <serialNumber>: απευθύνεται στη συσκευή που έχει το <SerialNumber>

<command>: η προς εκτέλεση εφαρμογή.

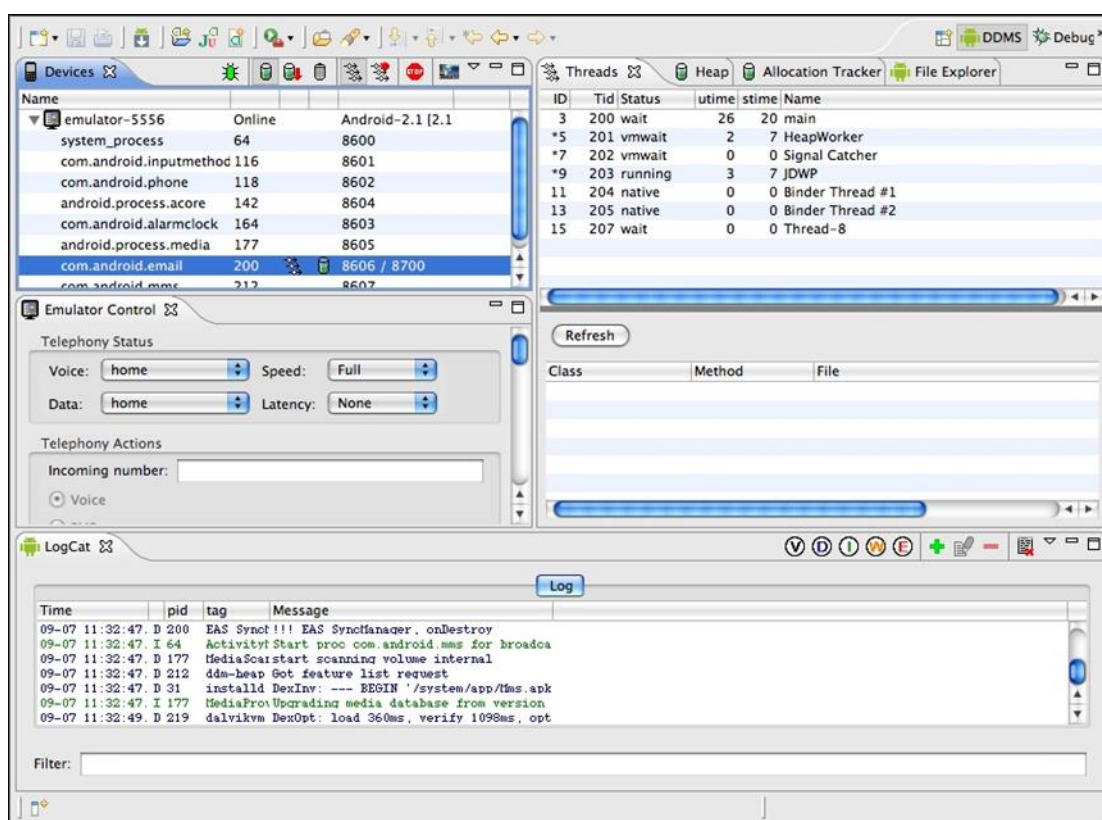


Dalvik Debug Monitor Server (DDMS)

Το συγκεκριμένο εργαλείο αποσφαλμάτωσης παρέχει τα εξής:

- Δημιουργία ψευδών SMS και πληροφοριών τοποθεσίας.
- Πληροφορίες για τις διεργασίες και τα νήματα (threads) της συσκευής.
- Υπηρεσίες προώθησης θυρών.
- Πληροφορίες δικτύου και εισερχομένων κλήσεων.
- Το εργαλείο LogCat.

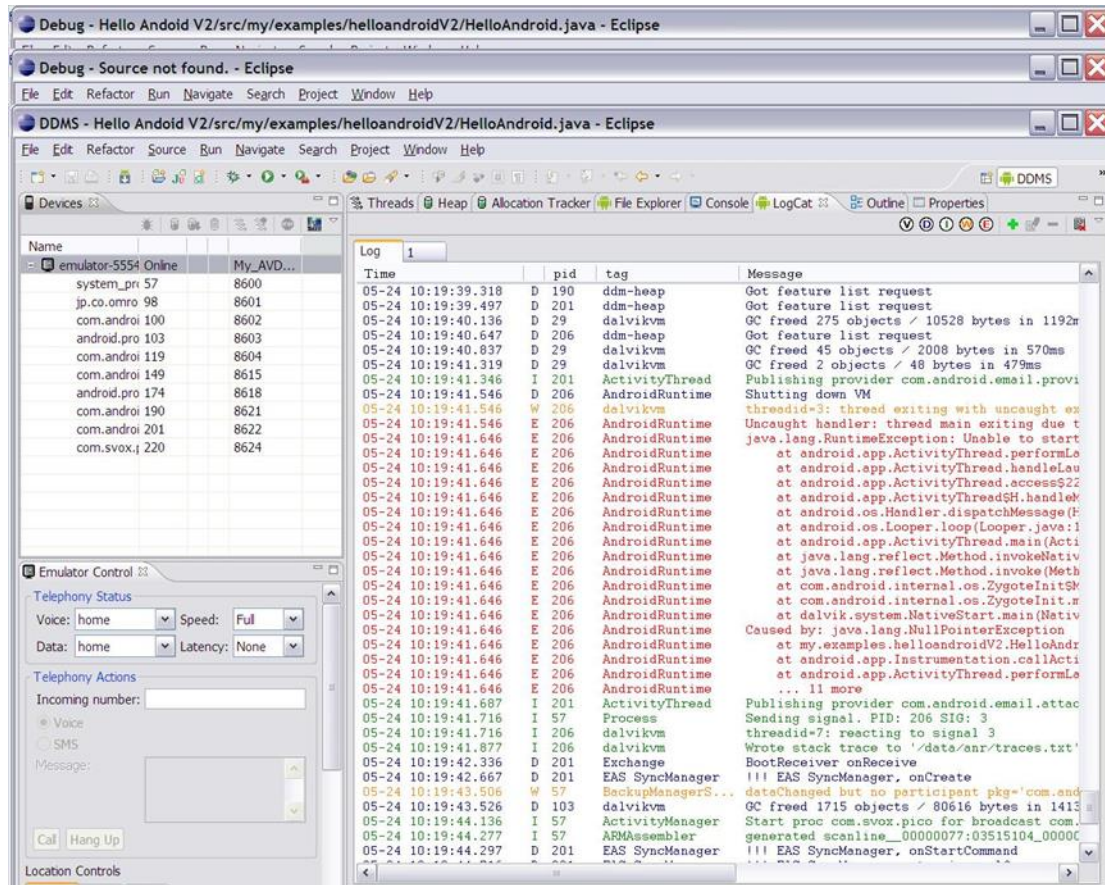
Το συγκεκριμένο εργαλείο το συναντάμε στο Eclipse και το Android SDK.



LogCat- Εργαλείο καταγραφής Συμβάντων

Το LogCat, είναι ένα εργαλείο καταγραφής συμβάντων, που έχει σκοπό τη συλλογή και τη προβολή αρχείων αποσφαλμάτωσης του συστήματος. Είναι μια εντολή, που μπορεί να χρησιμοποιηθεί μέσω του Android Debug Bridge στον έλεγχο των debug logs της συσκευής, ώστε να γίνει ο εντοπισμός της φύσης των σφαλμάτων της εφαρμογής, που αποτελούν exceptions στον κώδικά μας.

Εκceptions (εξαίρεσεις), είναι οι καταστάσεις που προκύπτουν, όταν κάτι δεν πάει καλά, με αποτέλεσμα τη διακοπή λειτουργίας του προγράμματος, εφόσον δεν έχουμε ήδη χειριστεί την εξαίρεση.



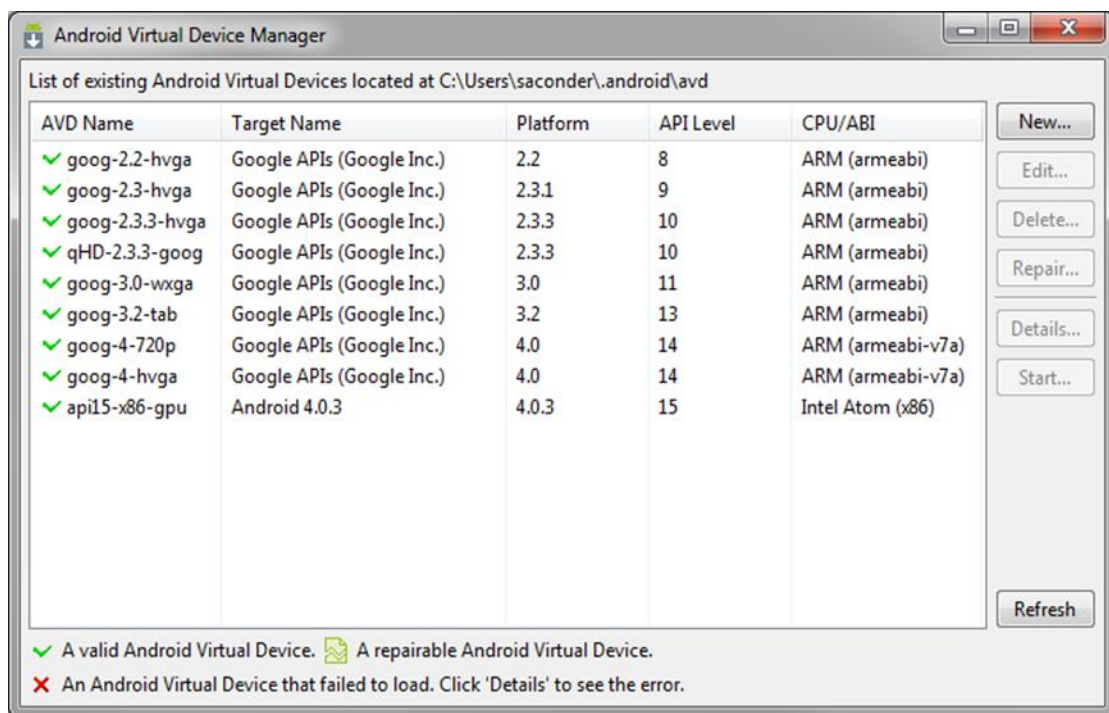
Εικονικές συσκευές Android (Android Virtual Devices-AVD)

Οι εικονικές συσκευές, χρησιμοποιούνται από τους προγραμματιστές, προκειμένου να δοκιμάζουν την εφαρμογή τους πριν την έκδοσή της. Είναι ιδιαίτερα χρήσιμα εργαλεία, αφού τους απαλλάσσει από την υποχρέωση να κατέχουν πολλές φυσικές συσκευές, ώστε να ελέγχουν τη λειτουργία και την εμφάνιση της εφαρμογής τους. Είναι οι συσκευές που εξομοιώνουν την συμπεριφορά των φυσικών συσκευών. Έτσι, οι προγραμματιστές μπορούν να ελέγξουν την εφαρμογή τους σε πραγματικές συνθήκες λειτουργίας και να πάρουν άμεσα feedback σχετικά με τη λειτουργία της.

Τα μέρη που αποτελούν συστατικά μιας εικονικής μηχανής είναι:

- Το προφίλ του υλικού, στο οποίο προσδιορίζονται τα χαρακτηριστικά και οι ιδιότητες της εικονικής μηχανής.
- Η έκδοση του Android, όπου γίνεται επιλογή της έκδοσης του Android, που θέλουμε να εξομοιώσει η εικονική συσκευή, καθώς και η επιλογή μεταξύ ειδικών εκδόσεων της πλατφόρμας (π.χ google tv).
- Ο επιπλέον χώρος αποθήκευσης, με τον ορισμό μιας εικονικής κάρτας μνήμης, που επιτρέπει την επέκταση του αποθηκευτικού χώρου για να αποθηκεύονται όλα τα δεδομένα της εφαρμογής.

Η δημιουργία και διαχείριση των εικονικών συσκευών, πραγματοποιείται μέσω του γραφικού περιβάλλοντος της εφαρμογής AVD MANAGER, που έρχεται μαζί με το SDK, ή ενσωματώνεται στο γραφικό περιβάλλον του ECLIPSE.



Οι πόροι των εφαρμογών

Για την ανάπτυξη μιας εφαρμογής δεν αρκεί μόνο ο κώδικας, αλλά χρειάζονται και πόροι. Με τους πόρους προσφέρεται μια εμπειρία χρήσης υψηλών προδιαγραφών στους χρήστες μιας εφαρμογής. Πόροι, αποτελούν οι εικόνες, τα χρώματα, οι διαστάσεις των στοιχείων των διεπαφών της εφαρμογής. Βασικό πλεονέκτημα των

πόρων είναι η δυνατότητα συντήρησης και ανάπτυξής τους χωριστά από τη συντήρηση και ανάπτυξη του κώδικα της εφαρμογής.

Οι πόροι χωρίζονται σε δύο κατηγορίες:

- Οι προκαθορισμένοι: είναι αυτοί που χρησιμοποιούνται από την εφαρμογή ανεξάρτητα τη συσκευή που 'τρέχει' και
- Οι εναλλακτικοί: είναι αυτοί που θα χρησιμοποιηθούν, όταν η συσκευή βρεθεί στην κατάσταση, στην οποία αντιστοιχούν οι πόροι αυτοί.

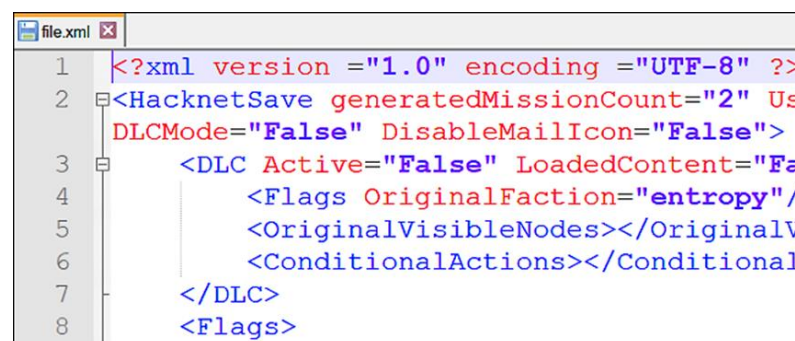
Εκτός από τους δικούς μας πόρους το Android διαθέτει ένα σύνολο δικών του πόρων που μπορούμε να χρησιμοποιήσουμε. Τους βρίσκουμε με τη μορφή:

android.R <όνομα_πόρου>

Οι πόροι παίζουν σημαντικό ρόλο στην ανάπτυξη εφαρμογών. Ένα καλά οργανωμένο σύστημα πόρων, μπορεί να διευκολύνει κατά πολύ την ανάπτυξη μιας εφαρμογής και να την κάνει ευκολότερα συντηρήσιμη και ευκολότερα επεκτάσιμη στο μέλλον.

Χρήση XML για την ανάπτυξη διεπαφών

Η XML, αποτελεί τρόπο ανάπτυξης των διεπαφών χρήστη και προσφέρεται από το ίδιο το android. Σημαντικό πλεονέκτημα αυτού του τρόπου, είναι η χωριστή ανάπτυξη της εφαρμογής από την ανάπτυξη της διεπαφής, μειώνοντας το τελικό χρόνο που απαιτείται για να αναπτυχθεί η εφαρμογή μας.



```
file.xml x
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <HacknetSave generatedMissionCount="2" Use
  DLCMode="False" DisableMailIcon="False">
3   <DLC Active="False" LoadedContent="Fa
4     <Flags OriginalFaction="entropy"/
5     <OriginalVisibleNodes></OriginalV
6     <ConditionalActions></Conditional
7   </DLC>
8   <Flags>
```

Java vs. XML

Όλες οι διεπαφές χρήστη στον κόσμο του Android αποτελούνται από έναν συνδυασμό στιγμιότυπων των κλάσεων View και ViewGroup του Android API. Τα στιγμιότυπα της κλάσης View αποτελούν τα στοιχεία των διεπαφών χρήστη. Είναι, ουσιαστικά, κομμάτια της διεπαφής που έχουν οπτική απεικόνιση στην οθόνη (κουμπιά, πεδία τιμών κ.λπ). Από την άλλη πλευρά, ένα στιγμιότυπο της κλάσης ViewGroup, δεν έχει οπτική αναπαράσταση στην οθόνη του χρήστη, αλλά βασικός σκοπός του είναι να ομαδοποιεί τα στιγμιότυπα της κλάσης View σε συγκεκριμένες δομές με συγκεκριμένες διατάξεις.

Για να δημιουργήσουμε την διεπαφή χρήστη για την εφαρμογή μας έχουμε δυο επιλογές. Η πρώτη, είναι να δημιουργήσουμε όλα τα απαραίτητα στιγμιότυπα των κλάσεων View και ViewGroup στον κώδικα της εφαρμογής μας και στην συνέχεια να τα ενώσουμε σε μια κατάλληλη δενδρική δομή (για μια πραγματική εφαρμογή, η δομή είναι σαφώς πιο πολύπλοκη). Κάτι τέτοιο βλέπουμε αμέσως ότι θα δημιουργούσε έναν τεράστιο όγκο κώδικα ο οποίος δεν θα έχει σχέση με την λογική της εφαρμογής μας αλλά θα πρέπει να τον συντηρούμε πάραυτα μαζί της. Ευτυχώς το Android προσφέρει και τον δεύτερο τρόπο ανάπτυξης της διεπαφής χρήστη. Την περιγραφή της σε XML.

Περιγράφοντας την διεπαφή σε XML επωφελούμαστε από το γεγονός ότι χωρίζεται η ανάπτυξη της λογικής της εφαρμογής από την ανάπτυξη της διεπαφής και πλέον οι δυο αυτές εργασίες είναι σχεδόν ανεξάρτητες και μπορούν να προχωρούν παράλληλα μειώνοντας δραματικά τον χρόνο ανάπτυξης της εφαρμογής μας.

Μέθοδοι που χρησιμοποιούνται για την οργάνωση των διεπαφών

Πέρα από τη σχεδίαση των διεπαφών, βασικό μέλημα των προγραμματιστών, αποτελεί και η δυνατότητά τους να οργανώσουν σωστά όλα τα στοιχεία που απεικονίζονται. Υπάρχουν τέσσερα διαφορετικά είδη μεθόδων, που

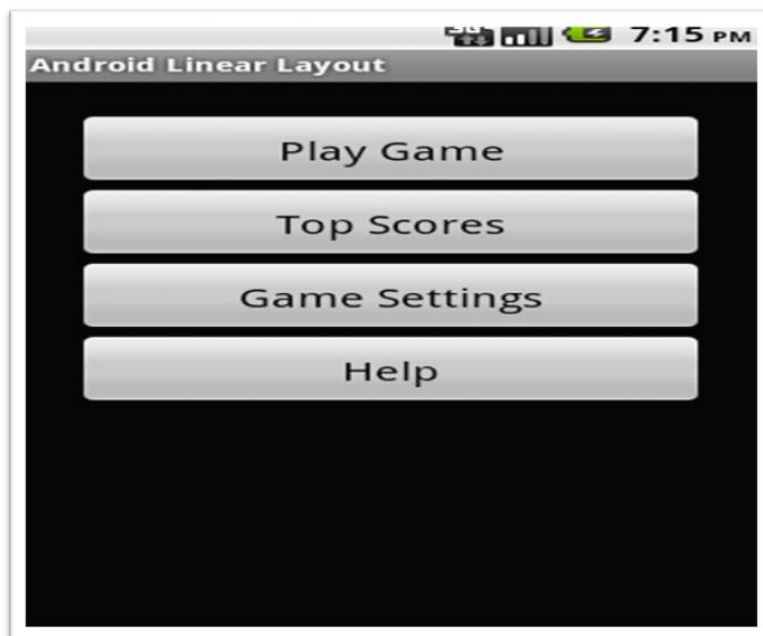
χρησιμοποιούνται προκειμένου η δομή των στοιχείων μιας διεπαφής να οργανώνεται με το καλύτερο δυνατό τρόπο.

Οι μέθοδοι παρουσιάζονται παρακάτω αναλυτικά:

Linear Layout

Με τη χρήση αυτής τη μεθόδου, πραγματοποιείται η οργάνωση των στοιχείων της διεπαφής με γραμμικό τρόπο. Δηλαδή, το ένα στοιχείο βρίσκεται κάτω από το άλλο ή το ένα δίπλα στο άλλο. Η οργάνωση μπορεί να είναι οριζόντια ή κάθετη ανάλογα με τη τιμή που καθορίζουμε:

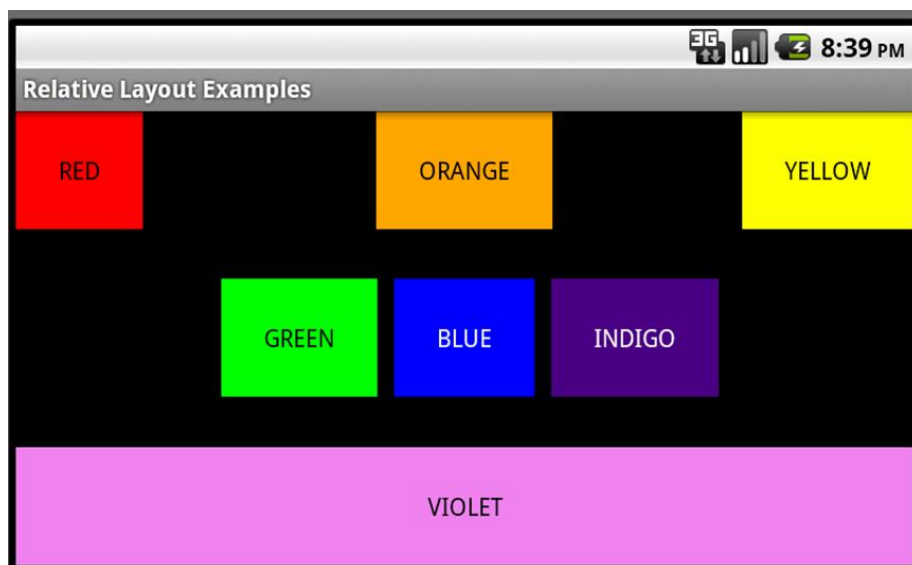
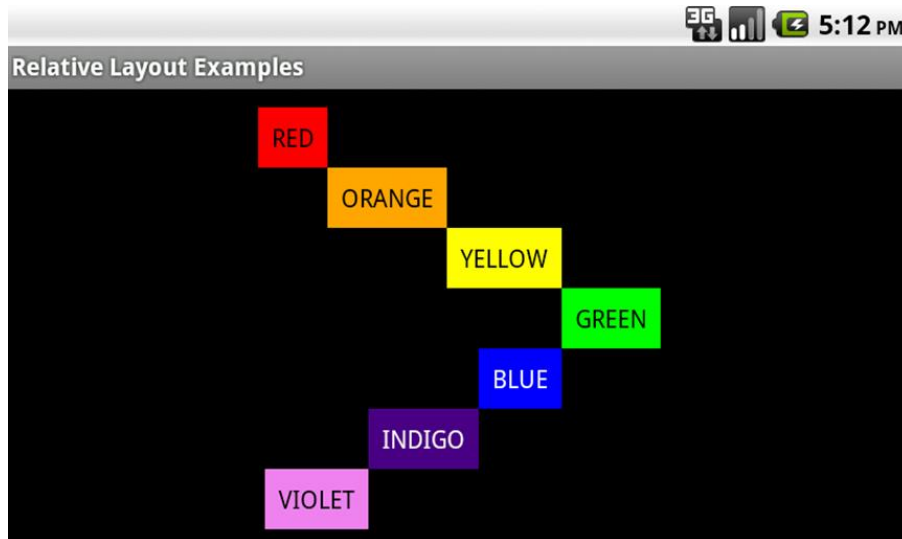
- `Android:orientation = "vertical"` : το ένα στοιχείο θα εμφανίζεται κάτω από το άλλο, από πάνω προς τα κάτω και με τη σειρά που έχουν δηλωθεί στην XML. ή
- `Android:orientation = "horizontal"` : το ένα στοιχείο, θα εμφανίζεται δίπλα στο άλλο, από αριστερά προς τα δεξιά και με τη σειρά που έχουν δηλωθεί με την XML.



Relative Layout

Η συγκεκριμένη μέθοδος παρέχει τη δυνατότητα οργάνωσης των στοιχείων σε συγκεκριμένες θέσεις, ανάλογα με τη θέση που κατέχουν τα υπόλοιπα στοιχεία της

διεπαφής. Μπορούμε να ορίσουμε τη τοποθέτηση ενός στοιχείου, διαγώνια κάτω από ένα άλλο, κεντραρισμένο κάθετα ως προς ένα άλλο, κάτω από ένα άλλο, δεξιά ή αριστερά από κάποιο άλλο.



Πρέπει ν' αναφέρουμε πως κατά τη δημιουργία των διεπαφών είναι χρήσιμο να δίνουμε ένα μοναδικό όνομα σε κάθε στοιχείο, ώστε να μπορούμε να κάνουμε αναφορά σε αυτό στον κώδικά μας, χωρίς να δημιουργείται πρόβλημα.

Table Layout

Το *Table Layout* είναι μια διάταξη δεδομένων που κατανέμεται σε ένα πλέγμα από σειρές και στήλες. Στα έντυπα έγγραφα, συνήθως χρησιμεύουν οι πίνακες μια δευτερεύουσα συνάρτηση, που απεικονίζει κάποιο σημείο που περιγράφεται από

το συνοδευτικό κείμενο. Οι πίνακες εξακολουθούν να εκτελούν αυτήν την ενδεικτική λειτουργία σε έγγραφα HTML.

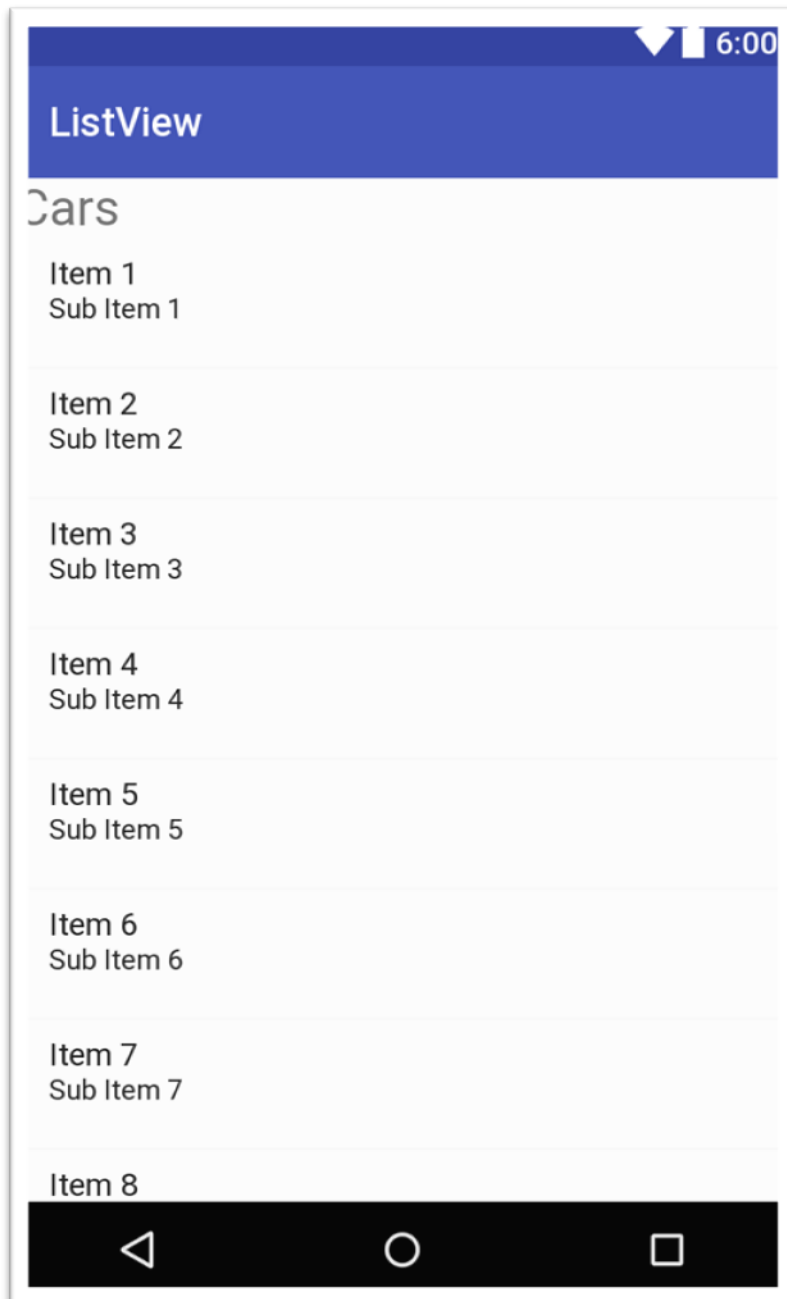
Οι πίνακες μπορούν να είναι ένα πολύ σημαντικό εργαλείο για τη διάταξη σελίδας που βασίζεται σε HTML. Η βάση του γραφικού σχεδιασμού τους είναι η ικανότητα να τακτοποιούνται χωρικά οπτικά στοιχεία μεταξύ τους. Τα Table Layout μπορούν να χρησιμοποιηθούν για τον καθορισμό ενός πλέγματος διάταξης για αυτόν ακριβώς τον σκοπό.



Με τη μέθοδο αυτή οργανώνουμε τα στοιχεία μας σε δομή πίνακα, δηλαδή σε πλέγμα από στήλες και γραμμές. Οι πίνακες μας δίνουν τη δυνατότητα επεξεργασίας. Μπορούμε να μεγεθύνουμε-σμικρύνουμε τις στήλες και τις γραμμές και να καθορίσουμε το τρόπο που στοιχίζεται το κείμενο μέσα στο πλέγμα. Τέλος, μπορούμε να καθορίσουμε τις στήλες που χρειάζεται να καταλάβει ένα στοιχείο. Όλα τα παραπάνω, μπορούμε να τα καθορίσουμε με τη χρήση εντολών στο κώδικα της διεπαφής.

List View

Με την εφαρμογή αυτή, τα αρχεία μας οργανώνονται σε μια κυλιόμενη λίστα. Στη λίστα αυτή, τα στοιχεία τοποθετούνται το ένα κάτω από το άλλο, δίνοντας στο χρήστη τη δυνατότητα να κυλάει πάνω κάτω, ώστε να εμφανίζονται όλα τα στοιχεία, ακόμα και αυτά που δεν χωράνε στην οθόνη. Για να ορίσουμε τη μέθοδο στην XML, χρησιμοποιούμε τον παρακάτω κώδικα:



<List View

```
android: id= "@+id/sbTicketsList"
```

```
android: layout_width= "wrap_content"
```

```
android: layout_height = " wrap_content"
```

```
android: layout weight= "1" / >
```

Κάνοντας σύγκριση της μεθόδου αυτής με τις προηγούμενες διαπιστώνουμε μια μεγάλη διαφορά. Αυτή είναι η δυνατότητα του χρήστη να ορίζει τη δική του XML, στην οποία θα μπορεί να ορίζει τη διάταξη που επιθυμεί να χρησιμοποιήσει.

Διεπαφή χρήστη και σχεδιασμός

Πως ένα τμήμα λογισμικού έρχεται σε επαφή με το χρήστη, αυτό είναι πρωταρχικής σημασίας. Οι οποιοδήποτε είδους ικανότητες που έχει το σύστημα, καθώς και οι χρηστικές δυνατότητες που παρέχει, μπορεί πολύ εύκολα να υποτιμηθούν και να μείνουν αναξιοποίητες αν η διεπαφή χρήστη (user interface) της εφαρμογής δεν είναι προσεκτικά σχεδιασμένη και υλοποιημένη.

Εργονομία, υλικού και λογισμικού, καθορίζουν τις προδιαγραφές που θα πρέπει να ακολουθούνται ώστε να επιτυγχάνεται από την μια μεριά ελαχιστοποίηση της καταπόνησης του χρήστη και από την άλλη μεγιστοποίηση της ωφελιμότητας από την χρήση του συστήματος θα πρέπει να γίνει κατανοητό πως ο τρόπος επικοινωνίας του συστήματος με το χρήστη είναι εξ ίσου σημαντικός με το ίδιο το περιεχόμενο του.

Τι είναι η διεπαφή χρήστη

Ο όρος *διεπαφή χρήστη (user interface)* είναι το σύνολο των συστατικών ενός συστήματος το οποίο επιτρέπει αμφίδρομη επικοινωνία μεταξύ συστήματος και χρήστη. Η διεπαφή χρήστη ενός συστήματος έχει σχέση με το ίδιο το σύστημα, το χρήστη του συστήματος και τον τρόπο που αλληλεπιδρούν μεταξύ τους.

Ο όρος θέλει να δείξει το σημείο επαφής χρήστη και υπολογιστή, την γραμμή επαφής πίσω από την μια μεριά της οποίας βρίσκεται η μηχανή και πίσω από την άλλη μεριά ο άνθρωπος. Έτσι λοιπόν η διεπαφή χρήστη περιέχει στοιχεία που είναι τμήματα τόσο του υλικού του συστήματος, όσο και του λογισμικού που "τρέχει" σε αυτό. Ως στοιχεία του υλικού του συστήματος που περιλαμβάνονται στη διεπαφή χρήστη μπορούν να αναφερθούν μια οθόνη επαφής, μια φωτογραφίδα (lightpen) ή ένα ποντίκι. Μέρη του λογισμικού της διεπαφής χρήστη είναι, για παράδειγμα, τα μηνύματα λάθους, τα ηχητικά μηνύματα, τα εργαλεία πλοήγησης, εικόνες σύμβολα

και αντικείμενα πάνω στην οθόνη, κάθε τι που διαθέτει το λογισμικό σαν στοιχείο αλληλεπίδρασης του συστήματος με το χρήστη.

Ο όρος σε ότι αφορά τη λογιστική του υπόστασης, σημαίνει ένα σύνολο από οπτικές και ακουστικές παραμέτρους, που παρέχει ο υπολογιστής προς το χρήστη, μέσω του εκάστοτε εκτελούμενου προγράμματος, με σκοπό την καλύτερη επικοινωνία και συνεργασία μεταξύ ανθρώπου και μηχανής. Μπορούμε να παρομοιάσουμε τη διεπαφή χρήστη σαν κανάλι επικοινωνίας μεταξύ χρήστη και υπολογιστή.

Κάθε χρονική στιγμή στην καθημερινή μας ζωή, μας περιβάλλουν διάφορα προϊόντα με καλά ή άσχημα σχεδιασμένα διεπαφή χρήστη. Όσο τα προϊόντα γίνονται πιο σύνθετα, ειδικά όταν πρόκειται για ηλεκτρικές και ηλεκτρονικές συσκευές, τόσο πιο συχνά ο χρήστης απογοητεύεται από τη δυσκολία που παρουσιάζει ο τρόπος χρήσης τους και πολύ συχνά δεν έχει τη διάθεση να κάνει καμία προσπάθεια για να τα κατανοήσει και να τα χειριστεί. Η διεπαφή χρήστη (το περιβάλλον εργασίας) είναι το πιο σπουδαίο κομμάτι κατά το σχεδιασμό μιας εφαρμογής. Ο χρήστης στον οποίο απευθύνεται η εφαρμογή μας, έχει τον τελευταίο λόγο σχετικά με την αγορά της εφαρμογής ή όχι. Έτσι, κατά την ανάπτυξη μιας εφαρμογής με σκοπό το κέρδος, είναι να τη προωθήσουμε στην αγορά της. Για να γίνει αυτό, πρέπει να του προσφέρουμε μια καλά σχεδιασμένη διεπαφή, εύχρηστη και ευχάριστη, δίνοντας την αίσθηση στο χρήστη, ότι η εφαρμογή έχει σχεδιαστεί βάση των αναγκών του. Επειδή, αναφερόμαστε στην ανάπτυξη εφαρμογών της πλατφόρμας android, ακολουθούμε συγκεκριμένους κανόνες.

Σχεδίαση διεπαφών στο Android

Η σχεδίαση διεπαφών για το android, δίνει ιδιαίτερο βάρος στο χρήστη και τις ανάγκες του. Πριν ξεκινήσει η σχεδίαση μιας εφαρμογής, πρέπει να ξέρουμε σε ποιο target group (κοινό) στοχεύουμε: ηλικία, κουλτούρα, τι θέλουν να πετύχουν με τη χρήση της εφαρμογής, τι συσκευές θα χρησιμοποιήσουν, που και πότε.

Οι βασικές αρχές στις οποίες ποντάρουμε για να σχεδιάσουμε τις διεπαφές μας είναι:

- Εντυπωσιασμός του χρήστη

- Διευκόλυνση του χρήστη
- Ενίσχυση της ψυχολογίας του

Εντυπωσιασμός του χρήστη

Σε αυτή τη περίπτωση, βασικός σκοπός είναι η *δημιουργία θαυμασμού του χρήστη* προς την εφαρμογή.

Μπορούμε να το πετύχουμε με τη χρήση μιας διεπαφής, που περιλαμβάνει διάφορα εικονικά ή ηχητικά εφέ που προκαλούν εντυπωσιασμό στον χρήστη. Ακόμα, η χρήση εικονιδίων και η δυνατότητα άμεσου χειρισμού τους από το χρήστη, βοηθάει στη δημιουργία αυτού του αισθήματος.

Διευκόλυνση του χρήστη

Σε αυτή τη περίπτωση, βασικός σκοπός είναι η δημιουργία μιας διεπαφής που να είναι εύχρηστη και λιγότερο κουραστική.

Είναι σημαντικό οι διάφορες λειτουργίες να εκτελούνται γρήγορα και τα διάφορα μηνύματα που εμφανίζονται στο χρήστη (ειδοποιήσεις), να είναι σύντομα και με ελάχιστες επαναλήψεις. Επίσης, είναι βασικό να χρησιμοποιούνται όσο το δυνατόν περισσότερες εικόνες και ν' αποφεύγεται το κείμενο, ώστε να μην κουράζεται ο χρήστης και να μπορεί να θυμάται πιο εύκολα.

Ικανότητα της εφαρμογής για απομνημόνευση

Στη τρίτη και τελευταία περίπτωση, ο βασικός σκοπός είναι να δημιουργήσουμε την αίσθηση στο χρήστη, ότι κατέχει την απαιτούμενη γνώση για να χειρίζεται τη συσκευή που διαθέτει. Ένας έξυπνος τρόπος είναι η χρήση σχεδιαστικών μοτίβων, τα οποία είναι ευρέως αποδεκτά από τη κοινότητα του android, όπως οι μπάρες λειτουργιών κ.α. Δημιουργείται μια οικεία αίσθηση στους χρήστες, όταν χρησιμοποιούν τεχνικές που είναι ήδη γνωστές σε αυτούς, κάνοντας την εκμάθηση

της εφαρμογής προσιτή. Τέλος, οι λειτουργίες αυτοματοποιούνται και οι αρχάριοι νιώθουν προχωρημένοι.

Ένα ακόμα σημαντικό σημείο διεπαφής είναι η επιλογή της διάταξης της στην οθόνη. Χρειάζεται ιδιαίτερη προσοχή αναφορικά με την οργάνωση πληροφοριών, λόγω της μικρής οθόνης που διαθέτουν κινητά και tablet. Έχει ιδιαίτερη σημασία να γίνεται σωστή οργάνωση, ώστε νέες ή πιο συχνά χρησιμοποιούμενες λειτουργίες να βρίσκονται σε πρώτο πλάνο, ενώ οι δευτερεύουσες να μπορούν να συγκεντρώνονται σε ένα μενού, ώστε να μην κουράζουν το χρήστη.

Ένα άλλο ιδιαίτερο χαρακτηριστικό είναι η επικοινωνία της εφαρμογής και του χρήστη. Είναι βασικό να δίνεται στο χρήστη η εντύπωση, ότι η οθόνη αντιλαμβάνεται την εντολή του χρήστη, όπως η αλλαγή χρώματος του κουμπιού όταν ο χρήστης το πατήσει.

Ανάπτυξη και δημιουργία εφαρμογών για κινητά για Android

Η δημοτικότητα των εφαρμογών Android αυξάνεται κάθε χρόνο και φέρνει όλο και περισσότερα κέρδη στους κατόχους τους. Για παράδειγμα, σύμφωνα με το Sensor Tower , το 2020 τα έσοδα από το Google Play αυξήθηκαν κατά 30% σε σύγκριση με το προηγούμενο έτος. Το AppCraft στούντιο αναπτύσσει εφαρμογές για κινητές συσκευές οποιασδήποτε πολυπλοκότητας για περισσότερα από 9 χρόνια: κοινωνικά δίκτυα, instant messengers, σύνθετες εταιρικές λύσεις, ηλεκτρονικό εμπόριο και τραπεζικά συστήματα.

Ποιος χρειάζεται ανάπτυξη εφαρμογών Android

Εταιρική επιχείρηση

Η εφαρμογή για κινητά θα βοηθήσει στην αυτοματοποίηση πολύπλοκων ροών εργασίας: δημιουργία ενός βολικού συστήματος για την αλληλεπίδραση με το

προσωπικό και τους πελάτες, την ανάλυση της αποδοτικότητας της εργασίας και τη γρήγορη λήψη σχολίων.

Ηλεκτρονικό Εμπόριο

Η εφαρμογή θα σας επιτρέψει να προσελκύσετε νέους πελάτες και να αυξήσετε τον αριθμό των επαναλαμβανόμενων πωλήσεων, να ενημερώσετε τους χρήστες για νέα προϊόντα και προσφορές και να δημιουργήσετε ένα σύστημα πιστότητας πελατών.

Startups

Θα σας βοηθήσει να μεταφέρετε γρήγορα την ιδέα σας στην αγορά και να έχετε το πρώτο κέρδος, να ενδιαφέρετε τους επενδυτές και να μελετήσετε το κοινό σας χρησιμοποιώντας τα εσωτερικά στατιστικά της εφαρμογής.

Γιατί χρειάζεται να αναπτύξετε μια εφαρμογή;

Η ικανότητα προσέγγισης ενός μεγάλου κοινού-στόχου - το 88% των ανθρώπων χρησιμοποιεί τηλέφωνα Android. Η ποικιλία των gadget στο Android σας επιτρέπει να αντικαταστήσετε πλήρως την επιφάνεια εργασίας, επομένως οι χρήστες χρειάζονται βολικές εφαρμογές για κινητά. Μπορείτε να εφαρμόσετε όλες τις λειτουργίες που θα σας βοηθήσουν να διατηρήσετε επαφή με τους χρήστες και να αυξήσετε τις πωλήσεις. Η δημοτικότητα του Google Play αυξάνεται κάθε χρόνο, γεγονός που καθιστά δυνατή την είσοδο σε νέα τμήματα της αγοράς.

Λειτουργίες ανάπτυξης Android

Υπάρχουν πολλές δυνατότητες στη διαδικασία ανάπτυξης εφαρμογών για android:

- Σε αντίθεση με το iOS, οι εφαρμογές Android είναι μια σχέση ξεχωριστών, λογικά ξεχωριστών στοιχείων, όπως συζητήθηκε παραπάνω. Δηλαδή, δεν μπορείτε απλώς να πάρετε και να εισάγετε μια εφαρμογή σε άλλο λειτουργικό σύστημα ξαναγράφοντας τον κώδικα από τη μια γλώσσα προγραμματισμού στην άλλη. Δηλαδή, όταν δημιουργείτε εφαρμογές για android, πρέπει να τοποθετήσετε μια εντελώς διαφορετική αρχιτεκτονική.
- Κατά την ανάπτυξη εφαρμογών για κινητά για android, είναι σημαντικό να βασίζεστε στο Material Design . Αυτή είναι μια ολόκληρη φιλοσοφία δημιουργίας διεπαφής χρήστη. Η επίσημη τεκμηρίωση αυτής της προσέγγισης περιλαμβάνει εκατοντάδες έγγραφα που περιγράφουν λεπτομερώς τόσο τις αρχές της όσο και συγκεκριμένα παραδείγματα σωστής και εσφαλμένης χρήσης κανόνων για κάθε στοιχείο διεπαφής.
- Η προτεινόμενη γλώσσα προγραμματισμού της Google για εφαρμογές Android είναι αυτή τη στιγμή η Kotlin και όχι η Java. Η διαφορά μεταξύ τους είναι σημαντικά μικρότερη από ό,τι μεταξύ Objective-C και Swift για iOS, αλλά εξακολουθούν να είναι ελαφρώς διαφορετικές προσεγγίσεις στην ανάπτυξη.
- Η δοκιμή σε μεγάλο αριθμό φυσικών συσκευών (όχι εξομοιωτές) είναι εξαιρετικά σημαντική κατά τη δημιουργία εφαρμογών android. Ακόμη και αυτό, λόγω του τεράστιου αριθμού τηλεφώνων στην αγορά, δεν παρέχει απρόσκοπτη λειτουργία σε όλα τα διαθέσιμα μοντέλα, αλλά τουλάχιστον μειώνει την πιθανότητα προβλημάτων στις πιο δημοφιλείς συσκευές.

Σχεδιαστικά λάθη

Κατά τη σχεδιασμό μιας εφαρμογής android πρέπει να αποφεύγονται τα εξής:

- Τη σχεδίαση μιας διεπαφής, που δεν συμβαδίζει με τη γενική φιλοσοφία του android, αλλά αντιγράφει μια υπάρχουσα εφαρμογή κάποιου άλλου λογισμικού (π.χ iOS). Προσπαθώντας να σχεδιάσουμε μια οικεία διεπαφή

στο χρήστη, δεν πρέπει να ξεχνάμε ότι οι εφαρμογές μας πρέπει να έχουν τη δική τους ταυτότητα.

- Την αποφυγή συνεχούς επιβεβαίωσης με τη χρήση πλαισίων διαλόγου. Δεν πρέπει να αναγκάζουμε το χρήστη επανειλημμένως να επιβεβαιώνει για τις διάφορες αλλαγές στη λειτουργία, αλλά η διεπαφή μας να συγκρατεί οποιαδήποτε παρελθοντική αλλαγή του χρήστη και στο μέλλον, δίνοντάς του τη δυνατότητα για αναίρεση όποτε θέλει.
- Δεν πρέπει να χρησιμοποιούμε απόλυτα μεγέθη. Αυτό συμβαίνει επειδή οι εφαρμογές για το λειτουργικό σύστημα android προορίζονται για μια μεγάλη γκάμα συσκευών που διαφέρουν σε μέγεθος και ανάλυση. Αυτό μας οδηγεί στο να μεριμνήσουμε κατά τη σχεδίαση, πως η διάταξη των στοιχείων οφείλει να μη γίνεται με απόλυτες τιμές pixel, αλλά να υπάρχει ρευστότητα. Το android επιτρέπει και προτείνει τη χρήση dp (density independent pixels) αντί του px (pixels). Με τη χρήση dp, δίνεται η δυνατότητα κατά την εκτέλεση μιας εφαρμογής σε οθόνες διαφορετικής πυκνότητας, υπολογισμού των pixel των στοιχείων και επανασχεδιασμού με το μέγεθος.
- Δεν πρέπει να χρησιμοποιούμε μικρές γραματοσειρές. Δεδομένου πως οι οθόνες, που προορίζονται να εκτελεστούν οι εφαρμογές μας έχουν αρκετά μικρό μέγεθος, δεν πρέπει να παρασυρθούμε και να μειώσουμε το μέγεθος της γραματοσειράς, προκειμένου να χωρέσουμε μεγάλο όγκο πληροφοριών.

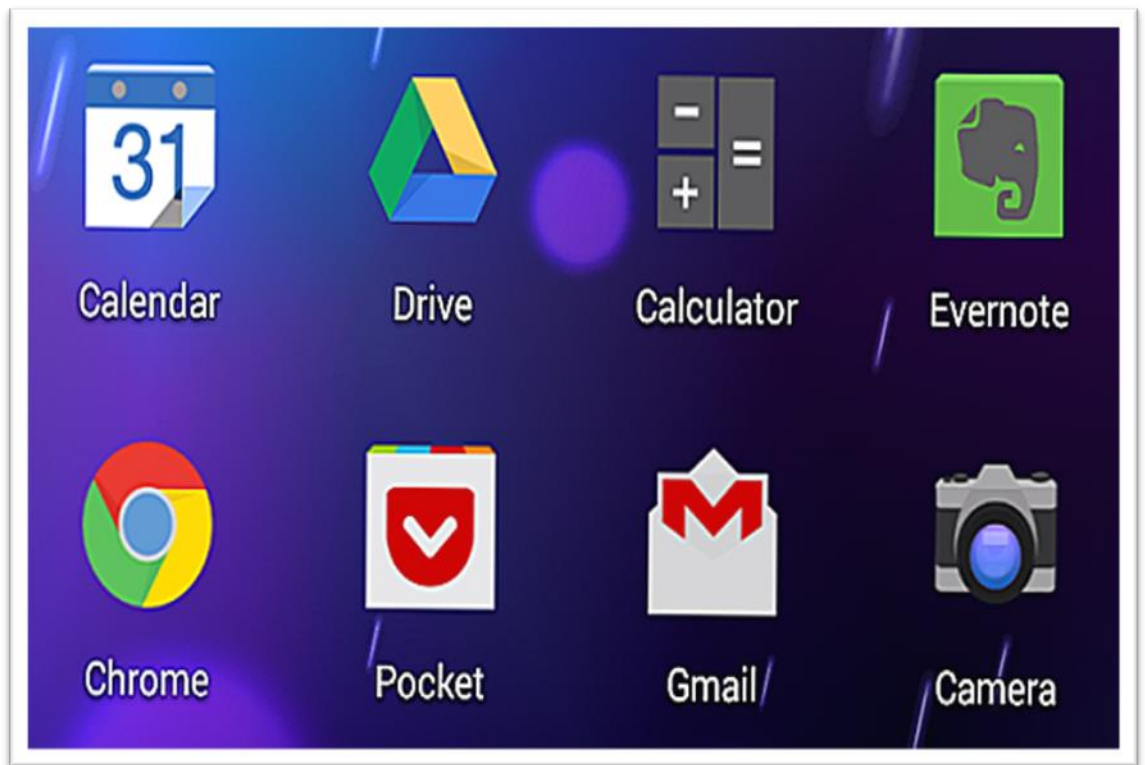
Στρατηγικές σχεδίασης

Πέρα από τα σχεδιαστικά λάθη που πρέπει να αποφεύγουμε, αξίζει να αναφερθούμε σε κάποιες στρατηγικές σχεδίασης διεπαφών, για να είναι το αποτέλεσμα ικανοποιητικό για το χρήστη.

- ❖ **Παροχή εικόνων και γραφικών σε διάφορες αναλύσεις.** Με αυτό το τρόπο θα μπορούν οι διάφορες εικόνες να προσαρμόζονται στις διάφορες οθόνες, με αναλλοίωτη την ανάλυση και τη ποιότητα. Η παραπάνω διαδικασία δεν

είναι υποχρεωτική από τους προγραμματιστές να την ακολουθούν, αλλά καλό είναι να μην την αγνοήσουμε, αφού παρέχεται από το android.

- ❖ **Σχεδιασμός των κουμπιών στην οθόνη της διεπαφής.** Τα κουμπιά που καλείται να επιλέξει ο χρήστης πρέπει να είναι *εύκολα επιλέξιμα*. Σύμφωνα με τις οδηγίες της Google το μέγεθος δεν πρέπει να ξεφύγει των ορίων 7-10 χιλιοστών. Παράλληλα, με τη σχεδίαση των κουμπιών της οθόνης, πρέπει να δοθεί προσοχή στη σχεδίαση και τις αλλαγές που πρέπει να προστεθούν όταν πραγματοποιείται περιστροφή της συσκευής.
- ❖ **Παρουσίαση εφαρμογής.** Είναι από τα πιο σημαντικά κομμάτια στο σχεδιασμό μιας διεπαφής. Όταν μιλάμε για παρουσίαση, αναφερόμαστε στα εικονίδια, στα χρώματα και στις γραμματοσειρές που επιλέγονται. Τα εικονίδια αποτελούν μικρές οθόνες, που παρουσιάζουν με σύντομο τρόπο μια ενέργεια ή κατάσταση, αλλά και μία εφαρμογή. Οι διαστάσεις τους είναι προκαθορισμένες (48x48 dp) και οφείλουν να μείνουν ορατά ανεξαρτήτως του φόντου που βρίσκονται. Πέρα από τα εικονίδια εφαρμογών πρέπει να δίνουμε προσοχή και στα εικονίδια που αναγγέλλουν την έναρξη κάποιας διαδικασίας και βρίσκονται στη μπάρα ενεργειών του android.
- ❖ **Χρήση γραμματοσειρών.** Οι γραμματοσειρές παίζουν σημαντικό ρόλο στη σχεδίαση μιας διεπαφής. Μια σωστή δομημένη γραμματοσειρά, πρέπει να δίνει τη δυνατότητα στο χρήστη να αντιλαμβάνεται άμεσα τη πληροφορία που του παρέχεται, να είναι ξεκούραστη κατά την ανάγνωση και να μπορεί να κάνει γρήγορη αναζήτηση μέσα σε αυτή. Σχεδιάζοντας τη γραμματοσειρά πρέπει να δίνουμε προσοχή στο μέγεθός της, στην απόσταση που έχουν τα γράμματα μεταξύ τους, στην απόσταση που έχουν οι λέξεις μεταξύ τους καθώς και στις παραλλαγές.



Συνήθειες σχεδίασης

Μια πολύ σημαντική παράμετρος που πρέπει να λάβουμε υπόψη μας είναι το πως μια εφαρμογή που δημιουργούμε, αποκρίνεται στις αντιδράσεις του χρήστη.

Ακολουθούμε κάποιες συνήθειες σχεδίασης, που εύκολα μπορεί να κατανοήσει ο χρήστης και να τις χειριστεί. *Αυτές οι συνήθειες πρέπει να είναι η ταυτότητα του λειτουργικού συστήματος **Android**.*

Gestures (χειρονομίες)

Ως gestures, αποκαλούμε το τρόπο, με τον οποίο επιλέγονται τα στοιχεία της οθόνης του κινητού ή του tablet, από το χρήστη. Υπάρχουν διαφόρων ειδών χειρονομίες που αναγνωρίζονται από το android:

- Άγγιγμα (touch): προκαλεί την ενεργοποίηση του στοιχείου. Για παράδειγμα, το πάτημα ενός κουμπιού της οθόνης, θα προκαλέσει την εκτέλεση της ενέργειας που καθορίζει το κουμπί.
- Σύρσιμο (swipe): προκαλεί τη μετακίνηση σε μία άλλη οθόνη που δεν είναι ορατή.

- Τράβηγμα (drag): χειρονομία που προκαλεί την αλλαγή της διάταξης των στοιχείων στη κύρια οθόνη του χρήστη.
- Διπλό άγγιγμα (double touch): είναι η χειρονομία που προκαλεί ζουμάρισμα σε μία περιοχή της διεπαφής.
- Άγγιγμα και άνοιγμα (pinch open): με τη χειρονομία αυτή προκαλείται ζουμ σε μια συγκεκριμένη περιοχή της διεπαφής.
- Άγγιγμα και κλείσιμο (pinch close): με τη χειρονομία αυτή προκαλούμε σμίκρυνση μιας συγκεκριμένης περιοχής της διεπαφής.

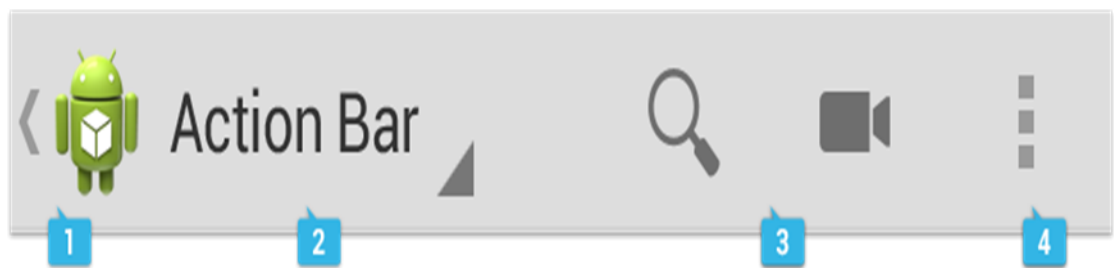
Δομή της οθόνης των εφαρμογών Android

Οι εφαρμογές **android** ανάλογα με τη λειτουργικότητά τους, απαιτούν και ανάλογες οθόνες προκειμένου να ικανοποιήσουν τις απαιτήσεις του χρήστη. Βάση της αρχιτεκτονικής του **android**, υπάρχουν τριών ειδών οθόνες: *οι οθόνες αρχικού επιπέδου, οι οθόνες κατηγοριοποίησης και οι οθόνες λεπτομερειών.*

- *Οθόνες αρχικού επιπέδου*: Δίνει τη δυνατότητα στο χρήστη να μπορεί να δει συνοπτικά τα δεδομένα και να εκτελεί λειτουργίες που είναι διαθέσιμες σε αυτές τις οθόνες.
- *Οθόνες κατηγοριοποίησης*: Οι συγκεκριμένες οθόνες δεν πρέπει να έχουν μεγάλο βάθος, να είναι σύντομες και σαφείς, προκειμένου να μην κουράσουν το χρήστη, αλλά να τον διευκολύνουν να επιλέξει κάτι και να οδηγηθεί στην οθόνη λεπτομερειών που αποτελεί το σκοπό του προγραμματιστή. Για να επιτευχθεί αυτό είναι απαραίτητη η χρήση καρτελών, ακόμα και κυλιόμενων ή καρτέλες με κατηγορίες που να είναι ορατές.
- *Οθόνες λεπτομερειών*: Αυτές αποτελούν τον προορισμό του χρήστη. Παρουσιάζουν το περιεχόμενο που αναζητά ο χρήστης. Σε αυτές τις οθόνες δεν υπάρχουν κανόνες σχετικά με τη σχεδίασή τους. Όλα εξαρτώνται από τη φαντασία του σχεδιαστή.

Η μπάρα ενεργειών

Η μπάρα ενεργειών βρίσκεται στο πάνω μέρος της οθόνης και βοηθάει το χρήστη να έχει πρόσβαση σε διάφορες λειτουργίες μιας εφαρμογής. Στις περισσότερες εφαρμογές android, πάνω στη μπάρα ενεργειών συναντάμε το εικονίδιο της εφαρμογής, το τίτλο της εφαρμογής, διάφορα πλήκτρα που απαιτούνται για να πραγματοποιήσει ο χρήστης ενέργειες στην εφαρμογή και ένα πλήκτρο που με το πάτημά του εμφανίζει κάποιες ενέργειες. Μια μπάρα ενεργειών φιλοξενεί μια πληθώρα από στοιχεία, όπως καρτέλες, απλά κουμπιά ενεργειών, ακόμα και δυνατότητα αλλαγής του περιεχομένου της, ανάλογα με τις επιλογές του χρήστη στην εφαρμογή.



Οι ειδοποιήσεις στις εφαρμογές

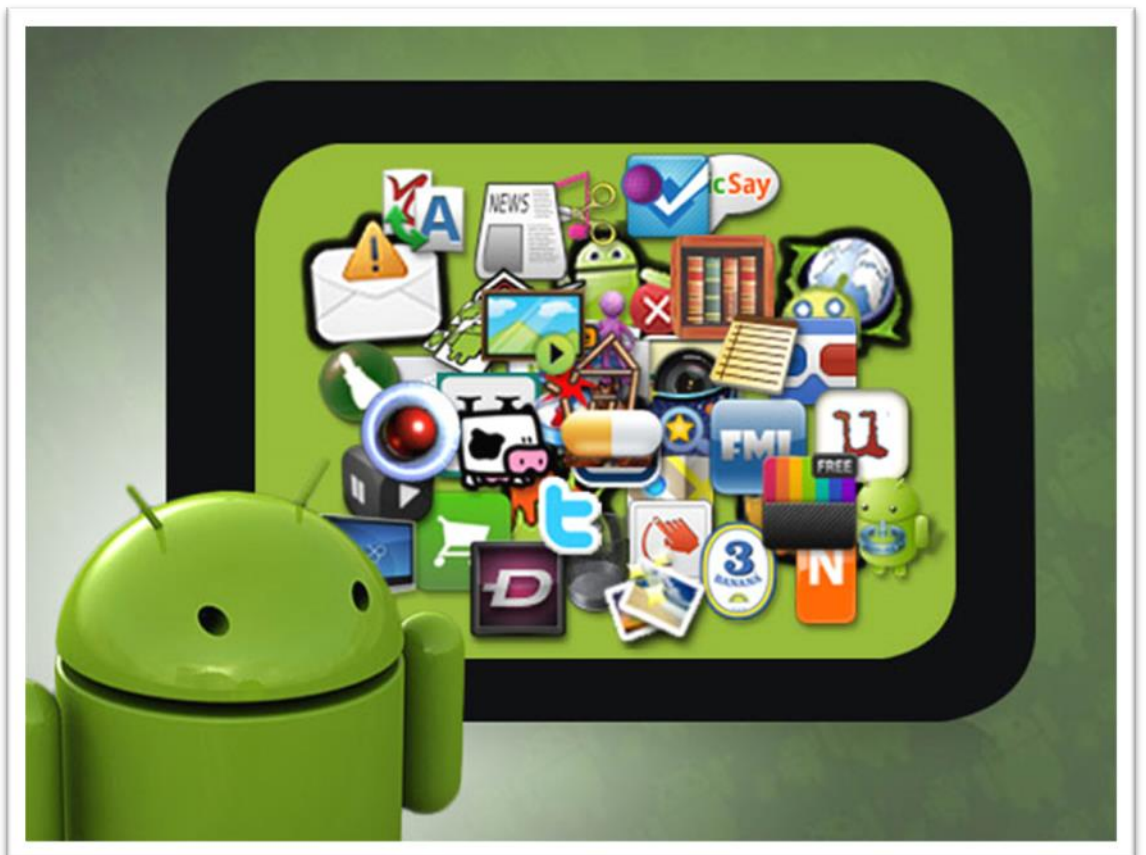
Οι ειδοποιήσεις αποτελούν το μέσο, με το οποίο οι εφαρμογές επικοινωνούν με το χρήστη, ώστε να τον ειδοποιούν για οτιδήποτε γίνεται. Βασικός παράγοντας στη χρήση ειδοποιήσεων για την ενημέρωση του χρήστη, αποτελεί η ορθή τους χρήση. Πρέπει ν' αποφεύγουμε τη συνεχή διακοπή του χρήστη για θέματα που δεν αφορούν ειδικά τον ίδιο, για γεγονότα που δεν είναι χρονικά περιορισμένα, για σφάλματα που μπορούν να επιλυθούν ή να παρακαμφθούν από την ίδια την εφαρμογή και το κυριότερο αποφεύγουμε τη χρήση του χώρου των ειδοποιήσεων για τη διαφήμιση μιας υπηρεσίας.

Ανταπόκριση

Όσο βελτιωμένος και αν είναι ο κώδικας της εφαρμογής μας, δεν θα έχει σημασία αν σε κάποιο χρονικό σημείο σταματήσει να αποκρίνεται. Όταν συμβεί κάτι τέτοιο η εφαρμογή μας θα βγάλει ένα παράθυρο διαλόγου ότι δεν αποκρίνεται και ο χρήστης έχει την δυνατότητα να τερματίσει την εφαρμογή ή να περιμένει. Ο διάλογος αυτός σίγουρα δεν είναι κάτι ευχάριστο για τους χρήστες γι' αυτό τον λόγο πρέπει να προσέχουμε στον

σχεδιασμό ώστε να μην συμβαίνει κάτι τέτοιο. Γενικά, το σύστημα εμφανίζει έναν τέτοιο διάλογο, όταν η εφαρμογή δεν ανταποκρίνεται σε ενέργειες του χρήστη. Μια τέτοια περίπτωση είναι ένα exception που ο προγραμματιστής έχει παραλείψει να πιάσει. Ομοίως, αν η εφαρμογή υλοποιεί πολύπλοκους και χρονοβόρους υπολογισμούς, ή αν έχει εισέλθει σε έναν άπειρο βρόγχο, τότε έχει κολλήσει και τελικά εμφανίζεται αυτός ο διάλογος. Τέτοια προβλήματα μπορούμε να τα ξεπεράσουμε με τους εξής τρόπους:

- Αρχικά θα πρέπει να ελέγξουμε αν ο κώδικάς μας είναι σωστός και όπου υπάρχει περίπτωση να συμβεί ένα exception να το χειριστούμε κατάλληλα.



- Οι Android εφαρμογές τρέχουν σε μια (κύρια) διεργασία (main thread) η οποία είναι υπεύθυνη και για το user interface. Αυτό σημαίνει ότι πρέπει να εκτελεί όσο το δυνατόν λιγότερες ενέργειες. Επομένως, όλες οι χρονοβόρες διεργασίες, για παράδειγμα οι λειτουργίες που έχουν να κάνουν με το διαδίκτυο ή με την προσπέλαση μιας βάσης δεδομένων, θα πρέπει να εκτελούνται στο παρασκήνιο. Αυτό γίνεται είτε με το να γίνουν σε μια ξεχωριστή thread ή σε μια ειδική ασύγχρονη εργασία που περιλαμβάνει το

Android και ονομάζεται AsyncTask. Σε αυτές τις περιπτώσεις για να είναι εμφανές στο χρήστη ότι εκτελείται μια χρονοβόρος εργασία εμφανίζουμε ένα ProgressDialog (π.χ με το μήνυμα Loading...).

- Αν η εφαρμογή μας, πχ ένα παιχνίδι, χρειάζεται αρκετό χρόνο για τις αρχικές λειτουργίες θα ήταν καλό να εμφανίζουμε στον χρήστη μια αρχική οθόνη για όσο διάστημα χρειαστεί.
- Δεν πρέπει να ξεχνάμε ότι το Android είναι λειτουργικό σύστημα για κινητές συσκευές. Αυτό σημαίνει ότι οποιαδήποτε στιγμή μπορεί κάποια activity μιας διαφορετικής εφαρμογής από αυτήν που βρισκόμαστε να ενεργοποιηθεί. Για παράδειγμα, να ενεργοποιηθεί η εφαρμογή που χειρίζεται τις εισερχόμενες τηλεφωνικές κλήσεις και η δικιά μας να περάσει στο παρασκήνιο. Το γεγονός αυτό ενεργοποιεί τις μεθόδους onSaveInstanceState() και onPause(). Αν και τις περισσότερες φορές δεν θα αντιμετωπίσουμε κάποιο πρόβλημα στην εφαρμογή μας θα ήταν προτιμότερο να έχουμε ελέγξει και αυτή την περίπτωση ώστε να προλάβουμε τυχόν σφάλματα.

Ανάπτυξη σχεδίασης UX / UI εφαρμογής για φορητές συσκευές

Ο σχεδιασμός UX και UI είναι η βάση που είναι υπεύθυνη για τα συναισθήματα κατά τη χρήση του προϊόντος. Καθορίζει πόσο βολική θα είναι η υπηρεσία και πόσο γρήγορα ένα άτομο μπορεί να φτάσει στην ενέργεια στόχο: αγορά ενός προϊόντος, κράτηση διαμερίσματος, εγγραφή κ.λπ. Στο άρθρο μιλάμε για τις δυνατότητες και τα στάδια δημιουργίας σχεδίου εφαρμογής για κινητά.

Τι είναι το UX/UI Design

Ο σχεδιασμός UX (User eXperience ή «εμπειρία χρήστη») είναι η διαδικασία σχεδιασμού της διεπαφής μιας εφαρμογής για φορητές συσκευές.

Με τη βοήθεια αναλυτικών στοιχείων και έρευνας, καθορίζεται πώς θα μετακινούνται οι άνθρωποι από τη μια οθόνη στην άλλη, ποιες λειτουργίες θα χρησιμοποιούν και ποια εντύπωση θα δημιουργεί η εφαρμογή μετά από κάθε αλληλεπίδραση.

Σχεδίαση διεπαφής χρήστη (Διασύνδεση χρήστη ή "διεπαφή χρήστη") - πώς θα φαίνεται οπτικά το έργο, που αναπτύχθηκε με βάση την εμπειρία του χρήστη. Εδώ χρησιμοποιούνται όλα τα μέσα: γραφικά και τυπογραφία, κινούμενα σχέδια, βίντεο, ήχοι και φωνή για τη δημιουργία σελίδων, κουμπιών, εικονιδίων και άλλων στοιχείων.

Σε ποιες περιπτώσεις αναπτύσσεται ο σχεδιασμός

Όταν μια εφαρμογή δημοσιεύεται για πρώτη φορά στην αγορά. Ο αριθμός των λήψεων και η πρώτη εντύπωση κατά τη χρήση θα εξαρτηθεί από τον τρόπο με τον οποίο παρουσιάζεται η υπηρεσία στα στιγμιότυπα οθόνης στο κατάστημα. Οι αποτυχίες του σχεδιασμού θα απογοητεύσουν το κοινό σας και θα βλάψουν το τελικό αποτέλεσμα. Εάν ένα άτομο δεν μπορεί να βρει το σωστό κουμπί, τότε δεν θα εκτελέσει την ενέργεια στόχο.

Στην αρχή της εργασίας, είναι σημαντικό να δημιουργήσετε επικοινωνία με τον σχεδιαστή και να δώσετε όσο το δυνατόν περισσότερες πληροφορίες για το προϊόν: για ποιον κατασκευάζεται, ποιους επιχειρηματικούς στόχους πρέπει να εκπληρώσει και πώς πρέπει να λειτουργεί.



Η εφαρμογή εκτελείται ήδη, αλλά υπάρχουν προβλήματα με τη διεπαφή

Εάν οι επισκέπτες συχνά δεν φτάνουν στην τελική ενέργεια, παραπονιούνται στην τεχνική υποστήριξη ακόμη και για βασικά στοιχεία και διαγράφουν την εφαρμογή, υπάρχει λόγος να επανεξετάσουν την ιδέα του σχεδιασμού.

Μπορείτε να χρησιμοποιήσετε έναν έλεγχο από μια εταιρεία ειδικών τρίτων για να αξιολογήσετε αντικειμενικά την κατάσταση και να διορθώσετε τα σφάλματα.

Για παράδειγμα, έχετε ένα ηλεκτρονικό κατάστημα και στο άμεσο μέλλον θέλετε να προσθέσετε τη δυνατότητα εικονικής δοκιμής ρούχων και κάψουλες με έτοιμες εμφανίσεις για κάθε σεζόν. Για να εφαρμόσετε νέες δυνατότητες, θα χρειαστεί να έχετε έναν σχεδιαστή , που θα σας βοηθήσει να εξηγήσετε σωστά τα νέα χαρακτηριστικά και θα τακτοποιήσετε σωστά όλα τα στοιχεία στην οθόνη, έτσι ώστε οι χρήστες να μπορούν να κατανοήσουν γρήγορα τη νέα διεπαφή.

Ποια είναι τα χαρακτηριστικά του σχεδιασμού εφαρμογών για κινητά;

Η δημιουργία μιας εφαρμογής δεν διαφέρει πολύ από την ανάπτυξη ιστοτόπων για οθόνη υπολογιστή, αλλά τα smartphone έχουν τους δικούς τους περιορισμούς και κανόνες. Έτσι, ο σχεδιαστής πρέπει να γνωρίζει τις ιδιαιτερότητες των πλατφορμών για κινητές συσκευές iOS και Android.

Μικρό μέγεθος οθόνης

Η οθόνη του τηλεφώνου περιέχει πολύ λιγότερα στοιχεία, επομένως η πλοήγηση θα πρέπει να είναι πολύ πιο εύκολη και οι πληροφορίες πρέπει να είναι όσο το δυνατόν συνοπτικές, αλλά ενημερωτικές.

Εάν στην οθόνη της επιφάνειας εργασίας ένα άτομο βλέπει ταυτόχρονα το μενού, τα φίλτρα, μέρος του προσωπικού λογαριασμού και διαφημιστικά banner, τότε στην έκδοση για κινητά θα πρέπει να επιλέξετε μόνο στοιχεία προτεραιότητας.

Οθόνη αφής

Οθόνη αφής" , είναι μια επαγωγική συσκευή οθόνης υγρών κρυστάλλων που μπορεί να λαμβάνει σήματα εισόδου, όπως επαφές. Το σύστημα ανάδρασης μπορεί να οδηγεί διάφορες συσκευές σύνδεσης σύμφωνα με το προγραμματισμένο πρόγραμμα, το οποίο μπορεί να χρησιμοποιηθεί για την αντικατάσταση του πίνακα μηχανικών κουμπιών και τη δημιουργία ζωντανών εφέ βίντεο και ήχου. Με χειρονομίες και τα χτυπήματα ελέγχουν κάθε κίνηση, επομένως τα στοιχεία με δυνατότητα **κλικ** γίνονται μεγαλύτερα, ώστε να είναι εύκολο να τα χτυπήσετε και να σύρετε.

Περιορισμένη απόδοση και κίνηση

Τα βαριά αρχεία και τα πολυμέσα επιβραδύνουν τη συνολική απόδοση του διακομιστή και κάνουν την εφαρμογή αργή. Η εφαρμογή θα πρέπει να στέλνει όσο το δυνατόν λιγότερες κλήσεις API ώστε να μην καθυστερεί ο χρήστης.

Τόποι αλληλεπίδρασης με το προϊόν

Εάν ένα άτομο κάθεται σε έναν υπολογιστή, τότε με ένα τηλέφωνο μπορεί να είναι οπουδήποτε και με διαφορετικές ταχύτητες Διαδικτύου, επομένως, στη διεπαφή για φορητές συσκευές, προσθέτουν μόνο βασικές πληροφορίες, πρόσθετες λειτουργίες και παρέχουν μια γρήγορη διαδρομή στις κύριες λειτουργίες.

Για παράδειγμα, στη διεπαφή, μπορείτε συχνά να επιλέξετε ένα "ελαφρύ" ή "σκοτεινό" θέμα για διαφορετικές ώρες της ημέρας ή να φορτώσετε εκ των προτέρων υλικό σε περίπτωση που δεν υπάρχει Διαδίκτυο.

Σύνοψη

Στην ενότητα αυτή ο εκπαιδευόμενος γνωρίζει για το περιβάλλον και την αρχιτεκτονική μιας εφαρμογής, γνωρίζει τι είναι ο αντικειμενοστραφής προγραμματισμός σε γλώσσα Java, ποιες είναι οι φάσεις ανάπτυξης μιας εφαρμογής και πως να αναπτύσσει εφαρμογές σε android λογισμικό.

Ερωτήσεις

A. Τι είναι το Progressive Web Apps (PWAs);

1. Οι Progressive Web Applications (PWAs), συγκροτούν ένα σύνολο από τεχνικές και Application Programming Interfaces (APIs), και δίνουν την δυνατότητα στους προγραμματιστές να πραγματοποιήσουν διαδικτυακές εφαρμογές.
2. Οι Progressive Web Applications (PWAs), είναι μια γλώσσα προγραμματιστών

Β. Τι είναι οι διαδικτυακές εφαρμογές (web application);

1. Διαδικτυακή εφαρμογή (web application ή web app) ονομάζεται κάθε εντολή ευρέσεις δεδομένων στο Διαδίκτυο
2. Διαδικτυακή εφαρμογή (web application ή web app) ονομάζεται κάθε εφαρμογή η οποία είναι διαθέσιμη στους χρήστες της μέσω του Διαδικτύου (Internet) ή του ενδοδικτύου (Intranet)

Γ. Ποια είναι τα πλεονεκτήματα των διαδικτυακών εφαρμογών;

1. Άμεση πρόσβαση από οποιαδήποτε συσκευή- Συμβατές με όλα τα λειτουργικά συστήματα- Δεν καταλαμβάνουν χώρο
2. Δυνατότητα χρήσης ανεξαρτήτως τοποθεσίας- Γρήγορη αναβάθμιση- Δυνατότητα χρήσης και εκτός διαδικτύου - ενδοδικτύου
3. Όλα τα παραπάνω

Δ. Ποια είναι τα μειονεκτήματα των διαδικτυακών εφαρμογών;

1. Χρήση της εφαρμογής εκτός διαδικτύου- Αδυναμία χρήσης χωρίς σύνδεση στο διαδίκτυο- Μη πλήρης συμβατότητα των περιηγητών.
2. Δεν υπάρχουν μειονεκτήματα

Ε. Ποιες είναι δημοφιλείς γλώσσες για ανάπτυξη γενικών ή εξειδικευμένων εφαρμογών;

1. C, η C++, η Java, η PHP, η C#, η Python, η JavaScript, η Perl,
2. Visual Basic, η Ruby, η Lisp, η Pascal, η Prolog και η MATLAB
3. Όλες οι παραπάνω.

Ζ. Η ανάπτυξη εφαρμογών τι πρέπει να ακολουθεί;

1. Η ανάπτυξη εφαρμογών πρέπει να ακολουθεί μια συστηματική διαδικασία με βήματα-φάσεις, ώστε να αποφεύγονται τα σφάλματα, οι δυσλειτουργίες και οι ελλείψεις

2. Η ανάπτυξη εφαρμογών δεν χιάζεται να ακολουθεί μια συστηματική διαδικασία ,ώστε να αποφεύγονται τα σφάλματα, οι δυσλειτουργίες και οι ελλείψεις

Η. Σε κάθε αλγόριθμο ποια στοιχεία πρέπει να καθορίζονται ;

1. Είσοδος (input)- Έξοδος (output)
2. Πεπερασμένα βήματα.
3. Όλα τα παραπάνω

Θ. Ποιες είναι οι βασικές λειτουργίες ή πράξεις των δομών δεδομένων;

1. Προσπέλαση (access)- Εισαγωγή (insertion)- Διαγραφή (deletion)
2. Διαχωρισμός (separation)- Συγχώνευση (merging)
3. Όλα τα παραπάνω

Ι. Ποιες τάσεις προέκυψαν σχετικά με τις τεχνολογίες Java;

1. ελαχιστοποίησης της χρήσης «καθαρού» κώδικα (scriptlets) στις Web σελίδες.
2. προώθησης τεχνικών για την μην επαναχρησιμοποίηση του κώδικα.

Κ. Τι είναι το Πρότυπο JSP-2 (MVC-2);

1. Το πρότυπο αυτό, που δημιουργήθηκε αρχικά για να καλύψει εφαρμογές σε προγράμματα πελάτη (client-side), προωθεί το διαχωρισμό του κώδικα σε τρεις καλά διακριτές ενότητες
2. Το πρότυπο JSP-2 αποτελεί την προσπάθεια της Sun να ενσωματώσει το MVC στις Web τεχνολογίες της.

Άσκηση

Να υλοποιηθούν σε C++ οι κλάσεις andgate, orgate, notgate. Οι κλάσεις αυτές θα προσομοιάζουν τις αντίστοιχες λογικές πύλες. Κάθε κλάση να έχει μεθόδους που να θέτουν τις εισόδους (π.χ. seta, setb) και μέθοδο που να επιστρέφει την τιμή της εξόδου (π.χ. getoutput).

Βιβλιογραφία

Chris Minnick HTML, CSS και JavaScript για Dummies (Exclusive B&N).

Nixon Robin(2014) Learning PHP, MySQL & JavaScript With jQuery, CSS & HTML5.

Δαμιανός Γαβαλάς, Βλάχης Κασαπάκης, Θωμάς Χατζηδημήτρης(2015) Κινητές τεχνολογίες, Κινητός ιστός - Κινητές εφαρμογές στην πλατφόρμα Android - επαυξημένη πραγματικότητα.

Hart-Davis Guy (2015) Teach Yourself VISUALLY Android Phones and Tablets

Εκπαιδευτική ενότητα: 8 Διασύνδεση εφαρμογών με βάσεις δεδομένων

Σκοπός

Σκοπός αυτής της ενότητας είναι να γνωρίσουμε τις τεχνολογίες που χρησιμοποιούμε για να διαχειριστούμε τα δεδομένα μας, όπως επίσης και το τρόπο με τον οποίο συνδέονται οι εφαρμογές μας με μια σχεσιακή βάση δεδομένων, αλλά και πως οι διαδικτυακές εφαρμογές συνδέονται με τη σχεσιακή βάση δεδομένων PHP.

Προσδοκώμενα αποτελέσματα

Μετά το τέλος αυτής της ενότητας ο εκπαιδευόμενος θα είναι σε θέση, να διαχειρίζεστε τα δεδομένα με τεχνολογίες που έχει μάθει και να μπορεί όταν του ζητηθεί να συνδέει εφαρμογές σε σχεσιακή βάση δεδομένων αλλά και διαδικτυακές εφαρμογές στην PHP.

Έννοιες-κλειδιά

Εξόρυξη δεδομένων (data mining) είναι ένας διεπιστημονικός κλάδος της επιστήμης των υπολογιστών. Είναι η υπολογιστική διαδικασία ανακάλυψης προτύπων σε σχετικά μεγάλα σύνολα δεδομένων χρησιμοποιώντας τη διασταύρωση της τεχνητής νοημοσύνης, της μηχανικής μάθησης, των στατιστικών και των βάσεων δεδομένων

Αποθήκες δεδομένων (data warehouses) είναι μια θεωρία αποθήκευσης δεδομένων του πληροφοριακού συστήματος. Αυτή η θεωρία δίνει έμφαση στη χρήση ορισμένων ειδικών μεθόδων αποθήκευσης δεδομένων για να καταστήσει τα δεδομένα που περιέχονται ιδιαίτερα ευνοϊκά για ανάλυση και επεξεργασία, ώστε να δημιουργηθούν πολύτιμες πληροφορίες και να ληφθούν ανάλογα αποφάσεις.

Big Data Τα μεγάλα δεδομένα , γνωστά και ως μαζικά δεδομένα, είναι ένας όρος που αναφέρεται σε μεγάλα ή πολύπλοκα σύνολα δεδομένων που το παραδοσιακό λογισμικό εφαρμογών επεξεργασίας δεδομένων δεν μπορεί να χειριστεί . Τα μεγάλα δεδομένα μπορούν επίσης να οριστούν ως μια μεγάλη ποσότητα μη δομημένων ή δομημένων δεδομένων από διάφορες πηγές . Από ακαδημαϊκή άποψη, η εμφάνιση των μεγάλων δεδομένων έχει οδηγήσει σε νέα έρευνα για ένα ευρύ φάσμα θεμάτων. Αυτό οδήγησε επίσης στην ανάπτυξη διαφόρων στατιστικών μεθόδων για μεγάλα δεδομένα .

Υπολογιστικό νέφος είναι μια υπολογιστική μέθοδος που βασίζεται στο Διαδίκτυο , με αυτόν τον τρόπο, κοινόχρηστοι πόροι λογισμικού και υλικού και πληροφορίες μπορούν να παρέχονται σε διάφορους υπολογιστές κατόπιν ζήτησης Τερματικά και άλλος εξοπλισμός χρησιμοποιούν υποδομή υπολογιστή που παρέχεται από παρόχους υπηρεσιών για υπολογιστές και πόρους.

Διαχείριση αρχείων είναι μία από τις πέντε λειτουργίες του λειτουργικού συστήματος, που περιλαμβάνει κυρίως τη λογική και φυσική οργάνωση των αρχείων, τη δομή και τη διαχείριση των καταλόγων. Η λεγόμενη διαχείριση αρχείων είναι ο γενικός όρος για μια ομάδα λογισμικού, διαχειριζόμενων αρχείων και ορισμένων δομών δεδομένων που απαιτούνται για τη διαχείριση αρχείων στο λειτουργικό σύστημα

ΣΔΒΔ, ένα σύστημα διαχείρισης βάσεων δεδομένων είναι το λογισμικό που επιτρέπει σε ένα υπολογιστή, να αποθηκεύει, να ανακτά, να προσθετεί, να διαγραφεί και να τροποποιεί δεδομένα .

ΒΔ, μέρος όπου αποθηκεύονται ηλεκτρονικά αρχεία και οι χρήστες μπορούν να εκτελούν λειτουργίες όπως η προσθήκη, η ανάκτηση, η ενημέρωση και η διαγραφή των δεδομένων στα αρχεία. "βάση δεδομένων" είναι μια συλλογή δεδομένων που

αποθηκεύονται μαζί με συγκεκριμένο τρόπο , μπορούν να κοινοποιηθούν από πολλούς χρήστες , έχουν όσο το δυνατόν λιγότερο πλεονασμό και είναι ανεξάρτητη από εφαρμογές

Διαχειριστής ΒΔ. Ένας διαχειριστής βάσεων δεδομένων συνεργάζεται με λογισμικό διαχειρίσεις βάσεων δεδομένων και πραγματοποιεί τροποποιήσεις με βάση τον τρόπο με τον οποίο ένα οργανισμός χρήζει να οργανώσει τα δεδομένα του

Εκπαιδευτική υποενότητα : 8.1 Τεχνολογίες διαχείρισης δεδομένων

Ο όρος «εξόρυξη δεδομένων» (data mining)

Η εξόρυξη δεδομένων (αγγλικά: data mining) είναι ένας διεπιστημονικός κλάδος της επιστήμης των υπολογιστών. Είναι η υπολογιστική διαδικασία ανακάλυψης προτύπων σε σχετικά μεγάλα σύνολα δεδομένων χρησιμοποιώντας τη διασταύρωση της τεχνητής νοημοσύνης , της μηχανικής μάθησης , των στατιστικών και των βάσεων δεδομένων.

Ο γενικός στόχος της διαδικασίας εξόρυξης δεδομένων είναι η εξαγωγή πληροφοριών από ένα σύνολο δεδομένων και η μετατροπή τους σε μια κατανοητή δομή για περαιτέρω χρήση. Εκτός από τα αρχικά βήματα ανάλυσης, περιλαμβάνει πτυχές της διαχείρισης βάσεων δεδομένων και δεδομένων , προεπεξεργασία δεδομένων , θεωρήσεις μοντέλων και συμπερασμάτων , μέτρα ενδιαφέροντος, θέματα πολυπλοκότητας , καθώς και ανακάλυψη δομών, οπτικοποίηση και διαδικτυακές ενημερώσεις Περιμένετε για μετα-επεξεργασία.

Η εξόρυξη δεδομένων είναι το βήμα ανάλυσης του «Knowledge-Discovery in Databases» (KDD), που ουσιαστικά ανήκει στην κατηγορία της μηχανικής μάθησης.

Όροι όπως " βυθοκόρηση δεδομένων " , " αλιεία δεδομένων " και " ανίχνευση δεδομένων " αναφέρονται στη χρήση μεθόδων εξόρυξης δεδομένων για τη

δειγματοληψία δεδομένων που είναι (πιθανώς) πολύ μικρά για να συναχθεί αξιόπιστα η εγκυρότητα οποιωνδήποτε προτύπων που βρέθηκαν. Μέρος ενός μεγάλου συνόλου δεδομένων πληθυσμού . Ωστόσο, αυτές οι μέθοδοι μπορούν να δημιουργήσουν νέες υποθέσεις για τη δοκιμή μεγαλύτερων πληθυσμών δεδομένων. Η εξόρυξη δεδομένων έχει αυτούς τους διαφορετικούς ορισμούς:

"Εξαγωγή κρυμμένων παρελθόντων άγνωστων πολύτιμων λανθάνουσας πληροφορίας από δεδομένα"

« Η επιστήμη της εξαγωγής χρήσιμων πληροφοριών από μεγάλο όγκο δεδομένων ή βάσεων δεδομένων »

Αν και συνήθως η εξόρυξη δεδομένων εφαρμόζεται στην ανάλυση δεδομένων, όπως η τεχνητή νοημοσύνη , είναι επίσης ένας όρος με πλούσια σημασία που μπορεί να χρησιμοποιηθεί σε διαφορετικούς τομείς. Η σχέση του με το KDD (ανακάλυψη γνώσης σε βάσεις δεδομένων) είναι: Το KDD είναι η διαδικασία αναγνώρισης έγκυρων, καινοτόμων, δυνητικά χρήσιμων και τελικά κατανοητών προτύπων από δεδομένα, ενώ η εξόρυξη δεδομένων είναι η διαδικασία εύρεσης αποδεκτών προτύπων μέσω συγκεκριμένων αλγορίθμων. δημιουργεί ένα συγκεκριμένο μοτίβο εντός των περιορισμών υπολογιστικής απόδοσης. Στην πραγματικότητα, στη σημερινή βιβλιογραφία, οι δύο όροι χρησιμοποιούνται συχνά αδιάκριτα.

Εξόρυξη δεδομένων

Η εξόρυξη δεδομένων περιλαμβάνει έξι κοινούς τύπους εργασιών:

- Ανίχνευση ανωμαλιών (Ανίχνευση ανωμαλίας/αλλαγής/απόκλισης) – Προσδιορίστε ασυνήθιστα αρχεία δεδομένων, λανθασμένα δεδομένα που απαιτούν περαιτέρω διερεύνηση.
- Εκμάθηση κανόνων συσχέτισης (Μοντελοποίηση εξάρτησης) – Αναζήτηση σχέσεων μεταξύ μεταβλητών. Για παράδειγμα, ένα σούπερ μάρκετ μπορεί να συλλέγει δεδομένα σχετικά με τις αγοραστικές συνήθειες των πελατών. Χρησιμοποιώντας την εκμάθηση κανόνων συσχέτισης, τα σούπερ μάρκετ μπορούν να καθορίσουν ποια προϊόντα αγοράζονται συχνά μαζί και να χρησιμοποιήσουν

αυτές τις πληροφορίες για να βοηθήσουν στο μάρκετινγκ. Αυτό μερικές φορές ονομάζεται ανάλυση καλαθιού αγοράς.

- Ομαδοποίηση – είναι η ανακάλυψη κατηγοριών και δομών δεδομένων κάτω από τη δομή άγνωστων δεδομένων.
- Ταξινόμηση – είναι το καθήκον της γενίκευσης μιας γνωστής δομής σε νέα δεδομένα. Για παράδειγμα, ένα πρόγραμμα email μπορεί να προσπαθήσει να ταξινομήσει ένα email ως "κανονικό" ή "spam".
- Παλινδρόμηση - Προσπάθεια εύρεσης μιας συνάρτησης που μπορεί να μοντελοποιήσει αυτά τα δεδομένα με ελάχιστο σφάλμα.
- Σύνοψη (Αγγλικά : Αυτόματη σύνοψη) - Παρέχει μια πιο συμπαγή αναπαράσταση συνόλων δεδομένων, συμπεριλαμβανομένης της δημιουργίας οπτικοποιήσεων και αναφορών.

Επαλήθευση αποτελεσμάτων

Η αξία της εξόρυξης δεδομένων έχει γενικά έναν συγκεκριμένο σκοπό, και αν αυτός ο σκοπός επιτυγχάνεται μπορεί γενικά να επιτευχθεί μέσω της επαλήθευσης των αποτελεσμάτων. Η επαλήθευση αναφέρεται στην "επιβεβαίωση ότι οι καθορισμένες απαιτήσεις πληρούνται με την παροχή αντικειμενικών αποδεικτικών στοιχείων" και ο σχεδιασμός, η υλοποίηση και η ολοκλήρωση αυτής της δραστηριότητας "διασφάλισης" σχετίζονται στενά με το περιεχόμενο των "καθορισμένων απαιτήσεων".

Ο καθορισμός «καθορισμένων απαιτήσεων» για την επαλήθευση δεδομένων στη διαδικασία εξόρυξης δεδομένων σχετίζεται συχνά με τους βασικούς στόχους, τους στόχους της διαδικασίας και τους τελικούς στόχους που πρέπει να επιτευχθούν με την εξόρυξη δεδομένων. Το αποτέλεσμα της επαλήθευσης μπορεί να είναι ότι οι «καθορισμένες απαιτήσεις» πληρούνται πλήρως ή καθόλου, καθώς και άλλες προϋποθέσεις. Η επικύρωση μπορεί να γίνει από τον ίδιο τον εξόρυξη δεδομένων ή από κάποιον άλλο που συμμετέχει ή εξ ολοκλήρου μέσω του έργου κάποιου άλλου, με τρόπο που δεν έχει καμία σχέση με τον εξόρυξη δεδομένων. Στη γενική διαδικασία επαλήθευσης, είναι αδύνατο για τους εξορύκτες δεδομένων να μην

συμμετέχουν, αλλά η συλλογή αντικειμενικών αποδεικτικών στοιχείων στη διαδικασία επαλήθευσης, η αξιολόγηση της επαλήθευσης και άλλες διαδικασίες είναι συχνά πιο αντικειμενικές εάν εφαρμόζονται από άτομα που δεν έχουν καμία σχέση με ο προτείνων επαλήθευση. Μέσω της επαλήθευσης των αποτελεσμάτων, οι εξορύκτες δεδομένων μπορούν να λάβουν μια αξιολόγηση της αξίας των δεδομένων που εξορύσσουν.

Διαδικασία εξόρυξης δεδομένων

“Διαδικασία” μια λέξη πολύ σημαντική εδώ.

Ο τύπος των δεδομένων μπορεί να είναι δομημένος, ημι-δομημένος ή ακόμα και ετερογενής. Οι μέθοδοι ανακάλυψης της γνώσης μπορεί να είναι μαθηματικές, μη μαθηματικές ή επαγωγικές. Η γνώση που τελικά ανακαλύπτεται μπορεί να χρησιμοποιηθεί για διαχείριση πληροφοριών, βελτιστοποίηση ερωτημάτων, υποστήριξη αποφάσεων και συντήρηση των ίδιων των δεδομένων.

Το αντικείμενο της εξόρυξης δεδομένων μπορεί να είναι οποιοσδήποτε τύπος πηγής δεδομένων. Μπορεί να είναι μια σχεσιακή βάση δεδομένων, μια τέτοια πηγή δεδομένων που περιέχει δομημένα δεδομένα· μπορεί επίσης να είναι μια αποθήκη δεδομένων, κείμενο, δεδομένα πολυμέσων, χωρικά δεδομένα, δεδομένα χρονοσειρών, δεδομένα Ιστού, μια τέτοια πηγή δεδομένων που περιέχει ημιδομημένα δεδομένα ή ακόμη και ετερογενή δεδομένα.

Οι μέθοδοι ανακάλυψης της γνώσης μπορεί να είναι αριθμητικές, μη αριθμητικές ή επαγωγικές. Η γνώση που τελικά ανακαλύπτεται μπορεί να χρησιμοποιηθεί για διαχείριση πληροφοριών, βελτιστοποίηση ερωτημάτων, υποστήριξη αποφάσεων και συντήρηση των ίδιων των δεδομένων.

Πριν από την εφαρμογή της εξόρυξης δεδομένων, είναι απαραίτητο να διατυπώσετε ποια βήματα πρέπει να κάνετε, τι να κάνετε σε κάθε βήμα και ποιους στόχους να επιτύχετε. Μόνο με ένα καλό σχέδιο μπορεί η εξόρυξη δεδομένων να εφαρμοστεί με τάξη και να είναι επιτυχής. Πολλοί προμηθευτές λογισμικού και σύμβουλοι εξόρυξης δεδομένων παρέχουν ορισμένα μοντέλα διαδικασίας εξόρυξης

δεδομένων για να καθοδηγήσουν τους χρήστες τους βήμα προς βήμα στην εξόρυξη δεδομένων. Για παράδειγμα, το SPSS' 5A και το SAS' SEMMA.

Τα βήματα του μοντέλου της διαδικασίας εξόρυξης δεδομένων περιλαμβάνουν κυρίως τον ορισμό του προβλήματος, τη δημιουργία της βιβλιοθήκης εξόρυξης δεδομένων, την ανάλυση των δεδομένων, την προετοιμασία των δεδομένων, τη δημιουργία του μοντέλου, την αξιολόγηση του μοντέλου και την εφαρμογή.

Ας ρίξουμε μια ματιά στο συγκεκριμένο περιεχόμενο κάθε βήματος αναλυτικά:

(1) Ορίστε το πρόβλημα. Η πρώτη και πιο σημαντική προϋπόθεση πριν ξεκινήσετε την ανακάλυψη γνώσης είναι η κατανόηση των δεδομένων και του επιχειρηματικού προβλήματος. Πρέπει να υπάρχει ένας σαφής και ξεκάθαρος ορισμός του στόχου, ο οποίος είναι να αποφασίσετε τι ακριβώς θέλετε να κάνετε. Για παράδειγμα, όταν θέλετε να βελτιώσετε το ποσοστό χρήσης των e-mail, αυτό που θέλετε να κάνετε μπορεί να είναι να "αυξήσετε το ποσοστό χρήσης των χρηστών" ή μπορεί να είναι "να αυξήσετε την αξία της χρήσης ενός χρήστη". Τα μοντέλα που έχουν δημιουργηθεί για την επίλυση αυτών των δύο προβλημάτων είναι σχεδόν εντελώς διαφορετικά. , πρέπει να ληφθεί μια απόφαση.

(2) Δημιουργήστε μια βιβλιοθήκη εξόρυξης δεδομένων. Η δημιουργία μιας βιβλιοθήκης εξόρυξης δεδομένων περιλαμβάνει τα ακόλουθα βήματα: συλλογή δεδομένων, περιγραφή δεδομένων , επιλογή, αξιολόγηση ποιότητας δεδομένων και καθαρισμός δεδομένων, συγχώνευση και ενοποίηση, δημιουργία μεταδεδομένων , φόρτωση βιβλιοθήκης εξόρυξης δεδομένων και συντήρηση βιβλιοθήκης εξόρυξης δεδομένων.

(3) Αναλύστε τα δεδομένα. Ο σκοπός της ανάλυσης είναι να βρει τα πεδία δεδομένων που έχουν τον μεγαλύτερο αντίκτυπο στην προβλεπόμενη παραγωγή και να αποφασίσει εάν θα οριστούν πεδία εξαγωγής. Εάν το σύνολο δεδομένων περιέχει εκατοντάδες ή χιλιάδες πεδία, τότε η περιήγηση και η ανάλυση των δεδομένων θα είναι μια πολύ χρονοβόρα και κουραστική εργασία. Αυτή τη στιγμή, πρέπει να επιλέξετε ένα λογισμικό με καλή διεπαφή και ισχυρές λειτουργίες που θα σας βοηθήσουν να ολοκληρώσετε αυτά τα πράγματα.

(4) Προετοιμάστε δεδομένα. Αυτό είναι το τελευταίο βήμα προετοιμασίας δεδομένων πριν από την κατασκευή του μοντέλου. Αυτό το βήμα μπορεί να χωριστεί σε τέσσερα μέρη: επιλογή μεταβλητών, επιλογή εγγραφών, δημιουργία νέων μεταβλητών και μετατροπή μεταβλητών.

(5) Κατασκευάστε ένα μοντέλο. Η κατασκευή ενός μοντέλου είναι μια επαναληπτική διαδικασία. Διαφορετικά μοντέλα πρέπει να εξεταστούν προσεκτικά για να καθοριστεί ποιο μοντέλο είναι πιο χρήσιμο για το επιχειρηματικό πρόβλημα που αντιμετωπίζει. Χρησιμοποιήστε ένα μέρος των δεδομένων για να δημιουργήσετε το μοντέλο και, στη συνέχεια, χρησιμοποιήστε τα υπόλοιπα δεδομένα για να δοκιμάσετε και να επικυρώσετε το μοντέλο που προκύπτει. Μερικές φορές υπάρχει ένα τρίτο σύνολο δεδομένων, που ονομάζεται σύνολο επικύρωσης, επειδή το σύνολο δοκιμής μπορεί να επηρεαστεί από τα χαρακτηριστικά του μοντέλου και απαιτείται ένα ξεχωριστό σύνολο δεδομένων για την επαλήθευση της ακρίβειας του μοντέλου. Η εκπαίδευση και η δοκιμή μοντέλων εξόρυξης δεδομένων απαιτεί τη διαίρεση των δεδομένων σε τουλάχιστον δύο μέρη, το ένα για την εκπαίδευση μοντέλων και το άλλο για τη δοκιμή μοντέλων.

(6) Μοντέλο αξιολόγησης. Μετά την καθιέρωση του μοντέλου, τα αποτελέσματα που λαμβάνονται πρέπει να αξιολογηθούν και να εξηγηθεί η αξία του μοντέλου. Η ακρίβεια που προκύπτει από το σύνολο δοκιμής έχει νόημα μόνο στα δεδομένα που χρησιμοποιούνται για την κατασκευή του μοντέλου. Σε πρακτικές εφαρμογές, είναι απαραίτητο να κατανοήσουμε περαιτέρω τους τύπους σφαλμάτων και το σχετικό κόστος. Η εμπειρία έχει δείξει ότι ένα αποτελεσματικό μοντέλο δεν είναι απαραίτητα το σωστό μοντέλο. Ο άμεσος λόγος για αυτό είναι οι υποθέσεις που υπονοούνται στο κτίριο του μοντέλου, επομένως είναι σημαντικό να δοκιμάσετε το μοντέλο απευθείας στον πραγματικό κόσμο. Εφαρμόστε το πρώτα σε μια μικρή περιοχή, λάβετε δεδομένα δοκιμών και, στη συνέχεια, προωθήστε το σε μια μεγάλη περιοχή αφού είστε ικανοποιημένοι.

(7) Εφαρμογή. Μόλις κατασκευαστεί και επικυρωθεί το μοντέλο, μπορεί να χρησιμοποιηθεί με δύο βασικούς τρόπους. Η πρώτη είναι η παροχή αναφοράς για τους αναλυτές· η άλλη είναι η εφαρμογή του μοντέλου σε διαφορετικά σύνολα δεδομένων.

Η εξόρυξη δεδομένων χωρίζεται σε κατευθυνόμενη εξόρυξη δεδομένων και μη κατευθυνόμενη εξόρυξη δεδομένων.

Η καθοδηγούμενη εξόρυξη δεδομένων είναι η χρήση διαθέσιμων δεδομένων για την κατασκευή ενός μοντέλου, το οποίο είναι μια περιγραφή μιας συγκεκριμένης ιδιότητας.

Η μη κατευθυνόμενη εξόρυξη δεδομένων αναζητά κάποιο είδος σχέσης μεταξύ όλων των χαρακτηριστικών. Συγκεκριμένα, η ταξινόμηση, η αποτίμηση και η πρόβλεψη ανήκουν στην κατευθυνόμενη εξόρυξη δεδομένων· οι κανόνες συσχέτισης και η ομαδοποίηση ανήκουν στη μη καθοδηγούμενη εξόρυξη δεδομένων.

1. Ταξινόμηση. Αρχικά επιλέγει ένα σύνολο εκπαίδευσης που έχει ταξινομηθεί από τα δεδομένα, χρησιμοποιεί τεχνολογία εξόρυξης δεδομένων στο σύνολο εκπαίδευσης για να δημιουργήσει ένα μοντέλο ταξινόμησης και, στη συνέχεια, χρησιμοποιεί το μοντέλο για να ταξινομήσει μη ταξινομημένα δεδομένα.

2. Εκτίμηση. Η αποτίμηση είναι παρόμοια με την ταξινόμηση, αλλά το τελικό αποτέλεσμα της αποτίμησης είναι μια συνεχής αξία και το ποσό της αποτίμησης δεν είναι προκαθορισμένο.

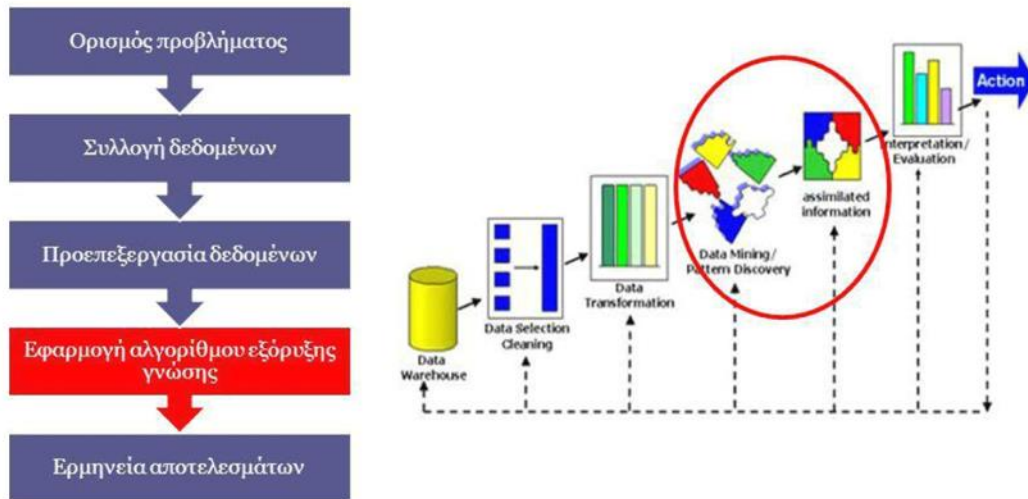
Η αποτίμηση μπορεί να χρησιμοποιηθεί ως προετοιμασία για ταξινόμηση.

3. προλέγω. Διενεργείται με ταξινόμηση ή αξιολόγηση και ένα μοντέλο αποκτάται μέσω της εκπαίδευσης στην ταξινόμηση ή αξιολόγηση. Εάν το μοντέλο έχει υψηλό ποσοστό ακρίβειας για την ομάδα δειγμάτων δοκιμής, το μοντέλο μπορεί να χρησιμοποιηθεί για άγνωστες μεταβλητές νέων δειγμάτων. Κάντε προβλέψεις .

4. Κανόνες ομαδοποίησης συσχέτισης ή συσχέτισης. Ο σκοπός είναι να ανακαλύψουμε ποια πράγματα συμβαίνουν πάντα μαζί.

5. ομαδοποίηση. Είναι μια μέθοδος αυτόματης εύρεσης και θέσπισης κανόνων ομαδοποίησης. Χωρίζει παρόμοια δείγματα σε ένα σύμπλεγμα κρίνοντας την ομοιότητα μεταξύ των δειγμάτων.

Η Διαδικασία Εξόρυξης Γνώσης



Κλασικός αλγόριθμος

Επί του παρόντος, οι αλγόριθμοι της εξόρυξης δεδομένων περιλαμβάνουν κυρίως τη μέθοδο νευρωνικών δικτύων, τη μέθοδο δέντρου αποφάσεων, τον γενετικό αλγόριθμο, τη μέθοδο ακατέργαστων συνόλων, τη μέθοδο ασαφούς συνόλου, τη μέθοδο κανόνων συσχέτισης και ούτω καθεξής.

Μέθοδος νευρωνικού δικτύου

Η μέθοδος του νευρωνικού δικτύου είναι η προσομοίωση της δομής και της λειτουργίας του βιολογικού νευρικού συστήματος. Είναι ένα μη γραμμικό μοντέλο πρόβλεψης που μαθαίνεται μέσω της εκπαίδευσης. Αντιμετωπίζει κάθε σύνδεση ως μονάδα επεξεργασίας και προσπαθεί να προσομοιώσει τη λειτουργία των νευρώνων στον ανθρώπινο εγκέφαλο. Εξόρυξη χαρακτηριστικών και άλλες εργασίες εξόρυξης δεδομένων. Η μέθοδος μάθησης του νευρωνικού δικτύου εκδηλώνεται κυρίως στην τροποποίηση των βαρών. Τα πλεονεκτήματά του είναι ότι έχει λειτουργίες κατά των παρεμβολών, μη γραμμική μάθηση, συνειρμική μνήμη και μπορεί να λάβει ακριβή αποτελέσματα πρόβλεψης για περίπλοκες καταστάσεις· το

μειονέκτημα είναι ότι δεν είναι κατάλληλο για την αντιμετώπιση μεταβλητών υψηλών διαστάσεων και δεν μπορεί να παρατηρήσει την ενδιάμεση διαδικασία μάθησης. Είναι επίσης δύσκολο να εξηγηθεί· δεύτερον, χρειάζεται πολύς χρόνος για να μελετηθεί. Η μέθοδος του νευρωνικού δικτύου χρησιμοποιείται κυρίως στην τεχνολογία ομαδοποίησης της εξόρυξης δεδομένων.

Μέθοδος δέντρου αποφάσεων

Το δέντρο αποφάσεων είναι μια διαδικασία κατασκευής κανόνων ταξινόμησης σύμφωνα με τις διαφορετικές επιδράσεις των μεταβλητών-στόχων. Η διαδικασία ταξινόμησης δεδομένων μέσω μιας σειράς κανόνων είναι παρόμοια με το διάγραμμα ροής της δομής δέντρου. Ο πιο τυπικός αλγόριθμος είναι ο J. R. O. Quinlan πρότεινε τον αλγόριθμο ID3 το 1986 και στη συνέχεια πρότεινε τον εξαιρετικά δημοφιλή αλγόριθμο C4.5 που βασίζεται στον αλγόριθμο ID3. Το πλεονέκτημα της χρήσης της μεθόδου του δέντρου αποφάσεων είναι ότι η διαδικασία λήψης αποφάσεων είναι ορατή, δεν απαιτεί πολύ χρόνο για την κατασκευή της διαδικασίας, η περιγραφή είναι απλή, εύκολη στην κατανόηση και η ταχύτητα ταξινόμησης είναι γρήγορη· το μειονέκτημα είναι ότι είναι δύσκολο να ανακαλύψουμε κανόνες που βασίζονται σε πολλαπλούς συνδυασμούς μεταβλητών. Η μέθοδος δέντρου αποφάσεων είναι καλή στην αντιμετώπιση μη αριθμητικών δεδομένων και είναι ιδιαίτερα κατάλληλη για επεξεργασία δεδομένων μεγάλης κλίμακας. Τα δέντρα αποφάσεων παρέχουν έναν τρόπο εμφάνισης κανόνων όπως ποιες τιμές λαμβάνονται υπό ποιες συνθήκες. Για παράδειγμα, σε μια αίτηση δανείου, είναι απαραίτητο να κριθεί ο κίνδυνος της αίτησης.

Γενετικός αλγόριθμος

Ο γενετικός αλγόριθμος προσομοιώνει τα φαινόμενα αναπαραγωγής, ζευγαρώματος και γονιδιακής μετάλλαξης που συμβαίνουν στη φυσική επιλογή και τη γενετική. Είναι μια μέθοδος μηχανικής μάθησης που βασίζεται στη θεωρία της εξέλιξης που χρησιμοποιεί λειτουργίες όπως γενετικός συνδυασμός, γενετική διασταύρωση και φυσική επιλογή για τη δημιουργία κανόνων εφαρμογής. Η βασική του ιδέα είναι η αρχή της «επιβίωσης του ισχυρότερου», η οποία έχει τις ιδιότητες του άρρητου παραλληλισμού και της εύκολης ενσωμάτωσης με άλλα μοντέλα. Το

κύριο πλεονέκτημα είναι ότι μπορεί να επεξεργαστεί πολλούς τύπους δεδομένων και διάφορα δεδομένα μπορούν να επεξεργαστούν παράλληλα· το μειονέκτημα είναι ότι απαιτεί πάρα πολλές παραμέτρους, η κωδικοποίηση είναι δύσκολη και το γενικό υπολογιστικό φορτίο είναι σχετικά μεγάλο. Οι γενετικοί αλγόριθμοι χρησιμοποιούνται συχνά για τη βελτιστοποίηση των νευρωνικών δικτύων και μπορούν να λύσουν προβλήματα που είναι δύσκολο να λυθούν με άλλες τεχνικές.

Μέθοδος Πρόχειρου Σετ

Η μέθοδος ακατέργαστων συνόλων, γνωστή και ως *ακατέργαστη θεωρία συνόλων*, προτάθηκε από τον Πολωνό μαθηματικό Z Pawlak στις αρχές της δεκαετίας του 1980. Είναι ένα νέο μαθηματικό εργαλείο για την αντιμετώπιση διφορούμενων, ανακριβών και ημιτελών προβλημάτων Ανακάλυψη συσχέτισης, αξιολόγηση της σημασίας δεδομένων κ.λπ. Το πλεονέκτημα είναι ότι ο αλγόριθμος είναι απλός και δεν απαιτείται προηγούμενη γνώση για τα δεδομένα στη διαδικασία επεξεργασίας και οι εγγενείς νόμοι του προβλήματος μπορούν να βρεθούν αυτόματα· το μειονέκτημα είναι ότι είναι δύσκολο να αντιμετωπιστούν άμεσα τα συνεχή χαρακτηριστικά και τα χαρακτηριστικά πρέπει πρώτα να *διακριτοποιηθούν*. Επομένως, η *διακριτοποίηση* των συνεχών χαρακτηριστικών είναι η δυσκολία που περιορίζει την πρακτική εφαρμογή της ακατέργαστης θεωρίας συνόλων. Η ακατέργαστη θεωρία συνόλων χρησιμοποιείται κυρίως στην προσεγγιστική συλλογιστική, την ψηφιακή λογική ανάλυση και απλοποίηση και την καθιέρωση προγνωστικών μοντέλων.

Μέθοδος Ασαφούς Συνόλου

Είναι η χρήση της θεωρίας ασαφών συνόλων για τη διεξαγωγή ασαφούς κρίσης, ασαφούς απόφασης, αναγνώρισης ασαφών προτύπων και ανάλυσης ασαφούς συστάδας σε προβλήματα.

Η θεωρία ασαφών συνόλων χρησιμοποιεί τη συμμετοχή για να περιγράψει τις ιδιότητες των ασαφών πραγμάτων. Όσο μεγαλύτερη είναι η πολυπλοκότητα του συστήματος, τόσο μεγαλύτερη είναι η ασάφεια.

Αποθήκες Δεδομένων (Data Warehouse)

Οι Αποθήκες Δεδομένων (*Data Warehouse*) Στον τομέα των υπολογιστών , μια αποθήκη δεδομένων (γνωστή και ως αποθήκη δεδομένων επιχείρησης) είναι ένα σύστημα που χρησιμοποιείται για αναφορά (Επιχειρηματική αναφορά) και ανάλυση δεδομένων και θεωρείται βασικό συστατικό της επιχειρηματικής ευφυΐας . Μια αποθήκη δεδομένων είναι μια κεντρική αποθήκη ενσωματωμένων δεδομένων από μία ή περισσότερες διαφορετικές πηγές. Μια αποθήκη δεδομένων αποθηκεύει τρέχοντα και ιστορικά δεδομένα μαζί και χρησιμοποιείται για τη δημιουργία αναλυτικών αναφορών για υπαλλήλους σε όλη την επιχείρηση. Τα δεδομένα που αποθηκεύονται στην αποθήκη μεταφορτώνονται από λειτουργικά συστήματα όπως το μάρκετινγκ ή οι πωλήσεις . Αυτά τα δεδομένα ενδέχεται να περάσουν μέσω μιας βάσης δεδομένων ODS και ενδέχεται να απαιτούν πρόσθετο καθαρισμό λειτουργικών δεδομένων για να διασφαλιστεί η ποιότητα των δεδομένων προτού χρησιμοποιηθούν για αναφορά στην αποθήκη δεδομένων .

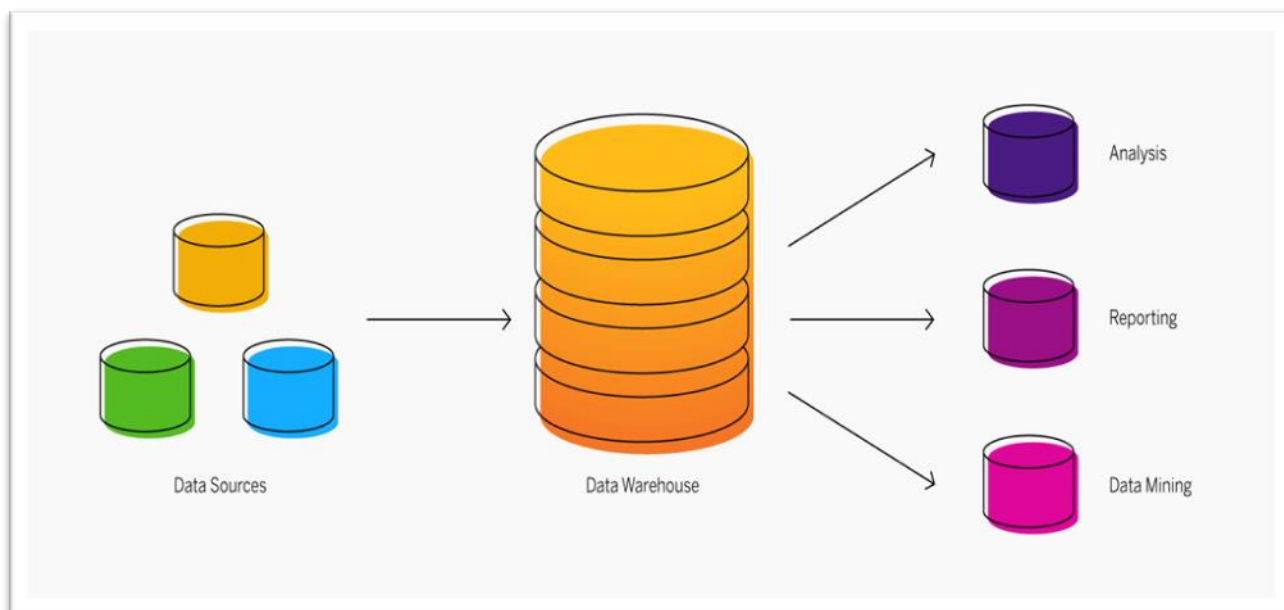
Μια τυπική αποθήκη δεδομένων που βασίζεται σε εξαγωγή, μετασχηματισμό, φόρτωση (ETL) χρησιμοποιεί επίπεδα ιεραρχίας , ολοκλήρωσης δεδομένων και πρόσβασης για να στεγάσει τις βασικές λειτουργίες της.

Το ιεραρχικό επίπεδο ή η ιεραρχική βάση δεδομένων αποθηκεύει τα ακατέργαστα δεδομένα που εξάγονται από καθένα από τα διαφορετικά συστήματα δεδομένων πηγής. Το επίπεδο ενοποίησης συγκεντρώνει διαφορετικά δεδομένα μετασχηματίζοντας τα δεδομένα από το ιεραρχικό επίπεδο, συνήθως αποθηκεύοντας τα μετασχηματισμένα δεδομένα στη βάση δεδομένων ODS. Τα ενσωματωμένα δεδομένα στη συνέχεια μετακινούνται σε μια άλλη βάση δεδομένων (συντά αποκαλούμενη βάση δεδομένων αποθήκης δεδομένων), όπου τα δεδομένα χωρίζονται σε ιεραρχικές ομάδες (συντά αποκαλούμενες διαστάσεις) και σε γεγονότα και συγκεντρωτικά γεγονότα. Ο συνδυασμός γεγονότων και διαστάσεων μερικές φορές ονομάζεται σχήμα αστεριών . Το επίπεδο πρόσβασης βοηθά τους χρήστες να ανακτήσουν δεδομένα.

Οι πρωτογενείς πηγές δεδομένων καθαρίζονται , μετασχηματίζονται, κατηγοριοποιούνται και διατίθενται σε διευθυντές και άλλους επαγγελματίες των

επιχειρήσεων για εξόρυξη δεδομένων , ηλεκτρονική αναλυτική επεξεργασία , έρευνα αγοράς και υποστήριξη αποφάσεων .

Ωστόσο, οι μέθοδοι για την ανάκτηση και ανάλυση δεδομένων, την εξαγωγή, τον μετασχηματισμό και τη φόρτωση δεδομένων και τη διαχείριση λεξικών δεδομένων θεωρούνται επίσης θεμελιώδη στοιχεία ενός συστήματος αποθήκης δεδομένων. Μεγάλο μέρος της βιβλιογραφίας για την αποθήκευση δεδομένων χρησιμοποιεί



αυτό το ευρύτερο πλαίσιο. Έτσι, ο εκτεταμένος ορισμός της αποθήκης δεδομένων περιλαμβάνει εργαλεία επιχειρηματικής ευφυΐας , εργαλεία εξαγωγής , μετατροπής και φόρτωσης δεδομένων σε αποθήκη και εργαλεία για τη διαχείριση και την ανάκτηση μεταδεδομένων

Η αποθήκη δεδομένων είναι μια θεωρία αποθήκευσης δεδομένων του πληροφοριακού συστήματος. Αυτή η θεωρία δίνει έμφαση στη χρήση ορισμένων ειδικών μεθόδων αποθήκευσης δεδομένων για να καταστήσει τα δεδομένα που περιέχονται ιδιαίτερα ευνοϊκά για ανάλυση και επεξεργασία, ώστε να δημιουργηθούν πολύτιμες πληροφορίες και να ληφθούν ανάλογα αποφάσεις.

Τα δεδομένα που αποθηκεύονται στην αποθήκη δεδομένων έχουν το χαρακτηριστικό ότι από τη στιγμή που αποθηκευτούν, δεν θα αλλάξουν με την πάροδο του χρόνου.

Ταυτόχρονα, τα αποθηκευμένα δεδομένα πρέπει να περιέχουν χαρακτηριστικά χρόνου.

Συνήθως, μια αποθήκη δεδομένων θα περιέχει μεγάλο αριθμό ιστορικών δεδομένων, και χρησιμοποιήστε συγκεκριμένα δεδομένα.

Αναλυτικές μέθοδοι για να ανακαλύψετε συγκεκριμένες πληροφορίες από αυτά..

Αλγόριθμοι Ταξινόμησης

Οι αλγόριθμοι ταξινόμησης εφαρμόζονται σε διακριτά δεδομένα, τα οποία έχουν προταξινομηθεί σε συγκεκριμένες κατηγορίες ή κλάσεις με στόχο την εξαγωγή κανόνων, οι οποίοι πιθανόν να χρησιμοποιηθούν αργότερα για την κατηγοριοποίηση καινούργιων δεδομένων στις ίδιες κλάσεις.

Ένα σύνολο εξαγόμενων κανόνων ονομάζεται ταξινομητής (classifier).

Ο όρος ταξινομητής αναφέρεται στη μαθηματική συνάρτηση, που εφαρμόζεται από έναν αλγόριθμο ταξινόμησης, ο οποίος χαρτογραφεί δεδομένα εισόδου σε μια κατηγορία.

Μετά από την εκπαίδευση του αλγορίθμου, ο ταξινομητής που έχει ήδη προκύψει μπορεί να χρησιμοποιηθεί και σε άλλες εγγραφές, οι οποίες δεν έχουν κατηγοριοποιηθεί.

Επιπλέον, με βάση ένα δεύτερο προ - ταξινομημένο σύνολο εγγραφών, το σύνολο ελέγχου (test set), μπορεί να ελεγχθεί η ακρίβεια ταξινόμησης ενός ταξινομητή.

Έπειτα, το εν λόγω σύνολο ελέγχου ταξινομείται εκ νέου, με την χρήση φυσικά του ταξινομητή και τέλος πραγματοποιείται μέτρηση του ποσοστού των λανθασμένων ταξινομήσεων (*error rate*).

Σχετικά με τους αλγόριθμους, υπάρχουν τρεις βασικές λειτουργίες:

1. Ένα σύνολο από δεδομένα εισάγεται στον αλγόριθμο.
2. Ο αλγόριθμος “μαθαίνει” και κατανοεί τον τρόπο και τους κανόνες βάσει των οποίων κατηγοριοποιήθηκαν τα εισαγόμενα στοιχεία.
3. Ακολουθώντας τους κανόνες έχει πλέον την ικανότητα να ταξινομήσει καινούργια δεδομένα.

Ανάλογα με το είδος του ταξινομητή που παράγει, ο κάθε αλγόριθμος διακρίνεται σε δύο βασικούς τύπους:

A) Αλγόριθμοι που παράγουν λίστες αποφάσεων : Αποτελούν μια σχετικά καινούργια μορφή αλγορίθμων (π.χ. Clark). Έχουν την μορφή λογικών κανόνων που βγάζουν ανάλογα συμπεράσματα.

B) Αλγόριθμοι που παράγουν δένδρα αποφάσεων : Αποτελούν ίσως και την πιο παλιά μορφή της τεχνικής του *Data Mining* (π.χ. Quinlan). Έχουν στην ρίζα τους και στους ενδιάμεσους κόμβους τους τις τιμές των διάφορων πεδίων και στα φύλλα τους έχουν τις τιμές του πεδίου κλάσης.

Ο κάθε κόμβος του δένδρου διακλαδώνεται προς τα κάτω έχοντας για κάθε διακριτή τιμή ένα κλαδί του πεδίου ενώ σε περίπτωση συνεχούς αριθμητικού πεδίου, το εύρος του πεδίου χωρίζεται σε διαστήματα και ο κόμβος διακλαδώνεται με βάση αυτά.

Συνήθως, αυτοί οι αλγόριθμοι ακολουθούν αναλυτική προσέγγιση , όπου ουσιαστικά δημιουργούν τα δένδρα από την ρίζα και προχωρούν προς τα κάτω.

Υπάρχουν σημαντικοί αλγόριθμοι ταξινόμησης μερικοί από τους οποίους παρουσιάζονται παρακάτω:

ID3 Αλγόριθμος

Ο ID3 (*Iterative Dichotomiser 3*), αποτελεί έναν από τους βασικότερους αλγορίθμους ταξινόμησης που παράγει δένδρα αποφάσεων.

Ανήκει στην οικογένεια των συστημάτων μάθησης TDIDT (*Top-Down Induction of Decision Trees*), ακολουθώντας την αναλυτική προσέγγιση.

Ο αλγόριθμος αυτός βασίζεται στην επιστημονική αρχή του Occam's Razor (Ξυράφι του Όκαμ), καθώς προτιμά τα μικρότερα δένδρα απόφασης - κατά συνέπεια και την απλούστερη θεωρία - από τα μεγαλύτερα.

Παρόλα αυτά, αυτό δεν σημαίνει ότι παράγει μόνο τα μικρότερα δένδρα.

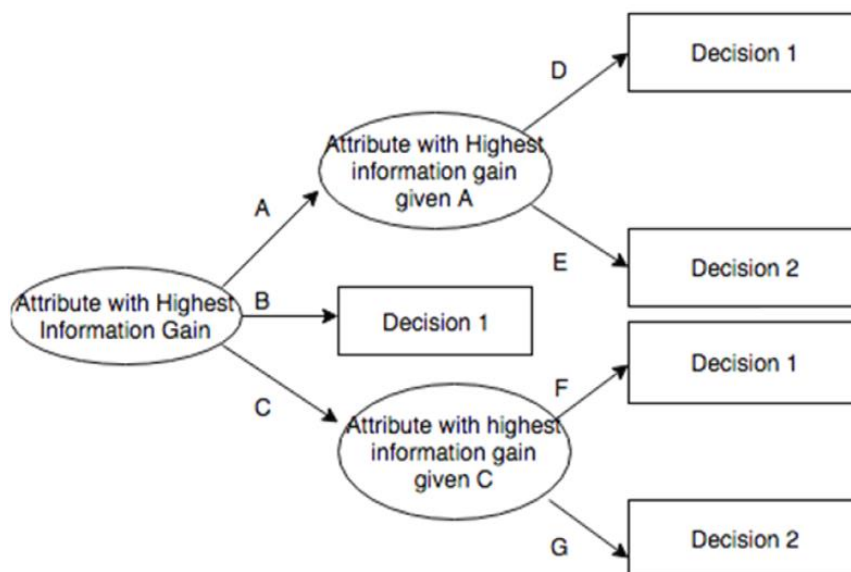
Ο ID3 δέχεται σαν είσοδο ένα σύνολο εκπαίδευσης, όπου οι εγγραφές του έχουν προταξινομηθεί σε κατηγορίες.

Στην αρχική του μορφή θεωρεί δύο διακριτές τιμές κλάσης - η P (positive) και η N (negative) - αν και μπορεί να επεκταθεί και σε περισσότερες από δυο.

Ωστόσο, αντί να εξάγει ολόκληρο ένα σύνολο εκπαίδευσης από το δένδρο, ο αλγόριθμος χρησιμοποιεί ένα υποσύνολο εγγραφών.

Έτσι, με το δένδρο που προκύπτει ταξινομείται ολόκληρο το σύνολο εκπαίδευσης και ελέγχεται και η ακρίβεια της ταξινόμησης.

Αν η ταξινόμηση έχει διεξαχθεί σωστά, τότε το δένδρο γίνεται αποδεκτό και ο



αλγόριθμος τερματίζει, αλλιώς προστίθενται κι άλλες εγγραφές στο υποσύνολο και η διαδικασία επαναλαμβάνεται μέχρι όλες οι εγγραφές να ταξινομηθούν σωστά.

C4.5 Αλγόριθμος

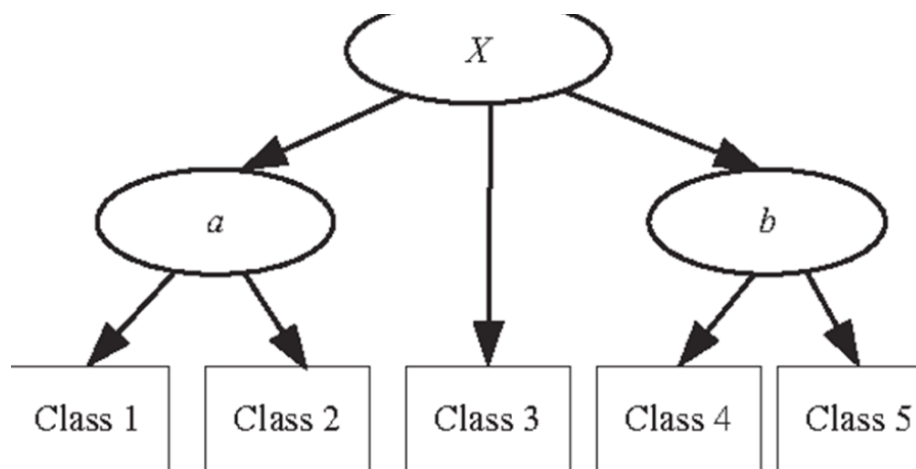
Ο αλγόριθμος αυτός αποτελεί την εξέλιξη του ID3 αλγόριθμου. Τα βασικά πλεονεκτήματα που προκύπτουν είναι:

- Δυνατότητα επεξεργασίας και διαχείρισης ποσοτικών κριτηρίων.
- Δυνατότητα διαχείρισης δεδομένων με ελλιπή στοιχεία.
- Αποφυγή της μεγάλης προσαρμογής στα δεδομένα του δείγματος εκμάθησης (overfitting).

Ο αλγόριθμος ξεκινώντας από ένα σύνολο παραδειγμάτων, τα οποία διαμορφώνουν το δείγμα εκμάθησης, οδηγεί στην ανάπτυξη ενός συνόλου κανόνων αποφάσεων για την ταξινόμηση εναλλακτικών δραστηριοτήτων.

Έπειτα οι κανόνες αυτοί οργανώνονται και δημιουργούν ένα δέντρο αποφάσεων. Ο κάθε κόμβος του δέντρου περιλαμβάνει το κριτήριο αξιολόγησης το οποίο ελέγχεται βάσει των συνθηκών που καθορίζουν τα κλαδιά του δέντρου.

Τα φύλλα με την σειρά τους υποδεικνύουν την κατηγορία στην οποία πρέπει να ενταχθεί μια εναλλακτική δραστηριότητα η οποία επαληθεύει την συνθήκη του κλάδου που καταλήγει στο φύλλο.



Support Vector Machine Αλγόριθμος

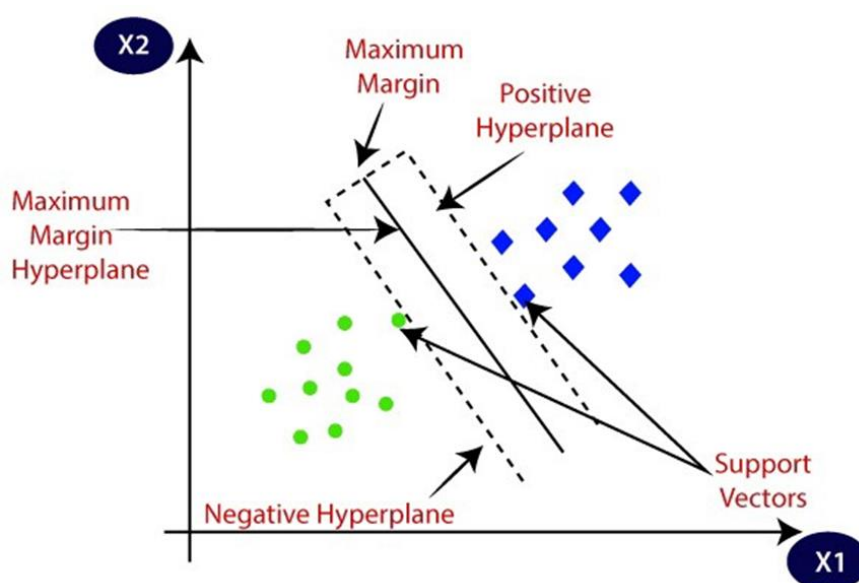
Η SVM (*Support Vector Machine*) αποτελεί μια πολλά υποσχόμενη νέα μέθοδο, η οποία αφορά τη ταξινόμηση γραμμικών και μη γραμμικών δεδομένων.

Ουσιαστικά, είναι ένας αλγόριθμος που λειτουργεί χρησιμοποιώντας μια μη γραμμική χαρτογράφηση για να μετασχηματίσει τα αρχικά δεδομένα εκπαίδευσης σε μια υψηλότερη διάσταση.

Μέσα σε αυτή τη νέα διάσταση, ψάχνει για ένα "όριο απόφασης" που διαχωρίζει τις πλειάδες μιας τάξης από την άλλη (*linear optimal separating hyperplane*).

Με μια κατάλληλη μη γραμμική χαρτογράφηση σε μια αρκετά μεγάλη διάσταση, δεδομένα από δύο κατηγορίες μπορούν πάντα να διαχωριστούν από ένα hyperplane.

Η SVM βρίσκει αυτό το hyperplane χρησιμοποιώντας διανύσματα υποστήριξης, support vectors, ("βασική" εκπαίδευση πλειάδων) και τα περιθώρια, margins, (που ορίζονται από τους φορείς υποστήριξης).



B) Συσταδοποίηση – Clustering

Η ομαδοποίηση ή συσταδοποίηση (clustering) είναι η τεχνική που διαχωρίζει έναν ετερογενή πληθυσμό σε έναν αριθμό των πιο ομοιογενών υποομάδων ή ομάδων (clusters). Αυτό που διαφοροποιεί την ομαδοποίηση από την ταξινόμηση είναι ότι η ομαδοποίηση δεν βασίζεται σε προκαθορισμένες κατηγορίες. Στην ταξινόμηση, κάθε εγγραφή έχει εκχωρηθεί σε μια προκαθορισμένη κλάση με βάση ένα μοντέλο που αναπτύχθηκε μέσω της εκπαίδευσης σε προ-ταξινομημένα παραδείγματα.

Στην ομαδοποίηση, τα αρχεία συγκεντρώνονται με βάση την ομοιότητα. Είναι στο χέρι του χρήστη να προσδιορίσει τι νόημα, αν υπάρχει, θα προσκολληθεί στις ομάδες (clusters) που προέκυψαν.

Η τεχνική της ομαδοποίησης μπορεί να είναι:

- Στατιστική ή Αριθμητική (statistical/numerical clustering) : Σε αυτήν την περίπτωση χρησιμοποιούνται διάφορα αριθμητικά κριτήρια ομοιότητας. Έτσι οι ομάδες που προκύπτουν περιγράφονται από αριθμητικές τιμές.
- Εννοιολογική (conceptual clustering) : Σε αυτήν την περίπτωση ο προσδιορισμός των ομάδων βασίζεται στο νόημα και στις έννοιες που τα διάφορα αριθμητικά στοιχεία αντιπροσωπεύουν. Έτσι οι τιμές που έχουμε είναι κατηγορικές και όχι αριθμητικές. Πολλοί από τους αλγόριθμους ομαδοποίησης απαιτούν το σύνολο εκπαίδευσης που επεξεργάζονται να είναι αριθμητικό (πχ k-means) είτε κατηγορικό (πχ k-modes).

Είναι σημαντικό να γνωρίζουμε τις μεθόδους συσταδοποίησης, οι οποίες αναλύονται παρακάτω:

Μέθοδοι Διαχωρισμού – Partitioning Methods

Δίνοντας μια βάση δεδομένων από n αντικείμενα ή πλειάδες δεδομένων, μια μέθοδος διαχωρισμού κατασκευάζει " k χωρίσματα" των δεδομένων, όπου κάθε χώρισα αντιπροσωπεύει ένα σύμπλεγμα και το k είναι μικρότερο ίσο από το n . Ουσιαστικά, ταξινομεί τα δεδομένα σε ομάδες k , οι οποίες ικανοποιούν τις ακόλουθες απαιτήσεις :

1. Κάθε ομάδα πρέπει να περιέχει τουλάχιστον ένα αντικείμενο.
2. Κάθε αντικείμενο πρέπει να ανήκει σε μία ακριβώς ομάδα.

Ιεραρχικές μέθοδοι – Hierarchical Methods

Αυτή η μέθοδος δημιουργεί μια ιεραρχική αποσύνθεση του δεδομένου συνόλου αντικειμένων δεδομένων (data objects).

Μια ιεραρχική μέθοδος μπορεί να χαρακτηριστεί ως είτε αθροιστική ή διαιρετική, με βάση τον τρόπο με τον οποίο διαμορφώνεται η ιεραρχική αποσύνθεση. Η αθροιστική (*agglomerative or bottom-up*) προσέγγιση ξεκινά με κάθε αντικείμενο να σχηματίζει μια ξεχωριστή ομάδα.

Συγχωνεύει διαδοχικά τα αντικείμενα ή τις ομάδες που βρίσκονται κοντά μεταξύ τους, μέχρις ότου όλες οι ομάδες συγχωνευθούν σε ένα (το ανώτατο επίπεδο της ιεραρχίας), ή μέχρις ότου τεθεί σε ισχύ ένας όρος τερματισμού.

Η διαιρετική προσέγγιση (*divisive or top-down*), ξεκινά με όλα τα αντικείμενα στο ίδιο σύμπλεγμα. Σε κάθε διαδοχική επανάληψη, ένα σύμπλεγμα χωρίζεται σε μικρότερες ομάδες, μέχρι τελικά κάθε αντικείμενο να είναι σε ένα σύμπλεγμα ή μέχρις ότου διατηρηθεί μια κατάσταση τερματισμού.

Μέθοδοι με βάση την πυκνότητα – Density-based Methods

Η γενική ιδέα της μεθόδου αυτής είναι να συνεχίσουν την ανάπτυξη της δεδομένης ομάδας όσο η πυκνότητα (αριθμός αντικειμένων ή δεδομένων σημεία) στην “γειτονιά” υπερβαίνει κάποιο όριο, δηλαδή για κάθε σημείο δεδομένων μέσα σε ένα δεδομένο σύμπλεγμα, η γειτονιά μιας δεδομένης ακτίνας πρέπει να περιέχει τουλάχιστον έναν ελάχιστο αριθμό σημείων. Μια τέτοια μέθοδος μπορεί να χρησιμοποιηθεί για να φιλτράρει τον θόρυβο (απόκλιση) και για την ανίχνευση ομάδων αυθαίρετου σχήματος.

Μέθοδοι με βάση το Πλέγμα – Grid-based Methods

Οι μέθοδοι με βάση το πλέγμα κβαντοποιούν τον χώρο του αντικειμένου σε έναν πεπερασμένο αριθμό των κυττάρων που σχηματίζουν μια δομή πλέγματος.

Όλες οι λειτουργίες ομαδοποίησης εκτελούνται στη δομή πλέγματος. Το κύριο πλεονέκτημά αυτής της προσέγγισης είναι ο γρήγορος χρόνος επεξεργασίας, ο οποίος είναι συνήθως ανεξάρτητος από τον αριθμό των αντικειμένων δεδομένων (data objects) και εξαρτάται μόνο από τον αριθμό των κελιών σε κάθε διάσταση στο κβαντισμένο χώρο. Το STING είναι ένα τυπικό παράδειγμα μιας μεθόδου με βάση το πλέγμα.

Υπάρχουν και στη συσταδοποίηση αλγόριθμοι και παρουσιάζονται παρακάτω:

K-means Αλγόριθμος

Ο k-means διαχωρίζει τα δεδομένα του συνόλου εκπαίδευσης (σύνολο εγγραφών) σε k ομάδες, όπου το k καθορίζεται από τον χρήστη.

Η λειτουργία του βασίζεται σε διαδοχικές επαναλήψεις κατά τις οποίες τα δεδομένα κατατάσσονται σε κάποια ομάδα με βάση την ομοιότητα που παρουσιάζουν με το μέσο αυτής της ομάδας.

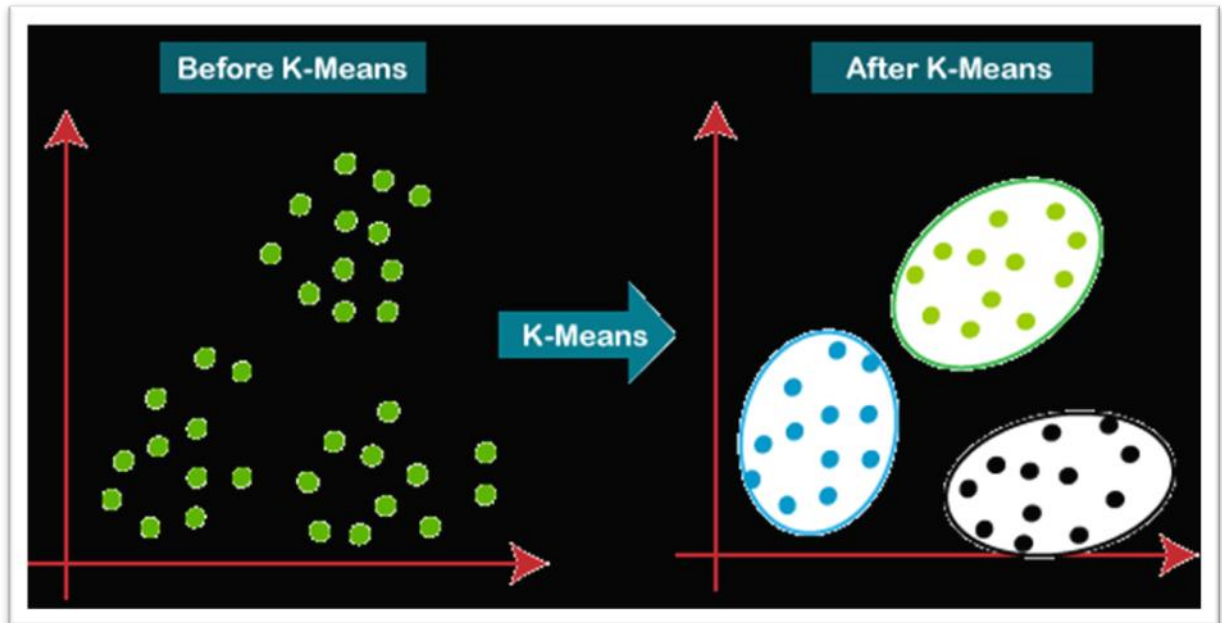
Ουσιαστικά, ο αλγόριθμος στηρίζεται σε κάποια αντιπροσωπευτικά δείγματα (means) κάθε ομάδας.

Κάθε μία από τις k ομάδες που θα δημιουργηθούν θα περιέχει ένα αντιπροσωπευτικό δείγμα το οποίο θα αντιπροσωπεύει την ομάδα, καθώς θα αποτελεί μια μέση περιγραφή αυτής.

Αυτό το αντιπροσωπευτικό δείγμα θεωρείται ότι είναι το κέντρο βάρους της ομάδας. Έπειτα, ο αλγόριθμος προσπαθεί να κατατάξει τις εγγραφές στις διάφορες ομάδες έτσι ώστε μετά τον τερματισμό, κάθε εγγραφή να ανήκει σε εκείνη την ομάδα από της οποίας το αντιπροσωπευτικό δείγμα απέχει λιγότερο σε σχέση με αυτά των άλλων ομάδων.

Τα βήματα του αλγορίθμου είναι τα εξής:

- Προσδιορίζουμε το k.
- Παίρνουμε τα αρχικά k αντιπροσωπευτικά δείγματα (π.χ. Παίρνουμε τις πρώτες k αντιπροσωπευτικές εγγραφές).
- Επαναλαμβάνουμε.
- Βρίσκουμε την απόσταση της κάθε εγγραφής από τα αντιπροσωπευτικά δείγματα και θεωρούμε ότι ανήκει στην ομάδα του πιο κοντινού αντιπροσωπευτικού δείγματος.
- Υπολογίζουμε τα νέα αντιπροσωπευτικά δείγματα (κέντρα βάρους) των ομάδων.
- Εκτελούμε μέχρι να μην γίνονται αλλαγές.



EM Αλγόριθμος

Ο αλγόριθμος EM λειτουργεί εκτιμώντας τα δεδομένα που λείπουν (E-step) και έπειτα εκτιμώντας τις παραμέτρους του μοντέλου με την μεγαλύτερη ομοιότητα (M-step). Η προσέγγιση αυτή απαιτεί η συλλογή αντικειμένων και οι ομάδες τους (clusters) να αναπαρίστανται από ένα στατιστικό μοντέλο. Τα δεδομένα θεωρούνται σαν ένα τυχαίο δείγμα από ένα μίγμα πιθανοτήτων κατανομών (distributions). Αυτές καθορίζουν και τα clusters.

Κανόνες Συσχέτισης – Association Rules

Αυτή η τεχνική χρησιμοποιείται για την ανακάλυψη προτύπων που περιγράφουν σημαντικές αλληλεξαρτήσεις μεταξύ των διαφόρων πεδίων - χαρακτηριστικών ενός συνόλου δεδομένων.

Εφαρμόζεται σε καταστήματα λιανικής πώλησης και βοηθά στο μάρκετινγκ, στη διαφήμιση, στον έλεγχο του καταλόγου απογραφής κλπ.

Οι κανόνες συσχέτισης μελετούν το πρόβλημα της εύρεσης συχνών συνόλων αντικειμένων ή στοιχειοσυνόλων (frequent itemsets) σε βάσεις δεδομένων. Βασίζονται σε ένα κατώφλι που ονομάζεται υποστήριξη (support), το οποίο αναγνωρίζει τα στοιχειοσύνολα.

Οι κανόνες συσχέτισης αντιπροσωπεύουν πρότυπα στα δεδομένα χωρίς συγκεκριμένο στόχο. Ως εκ τούτου, αποτελούν παράδειγμα μη κατευθυνόμενης εξόρυξης δεδομένων.

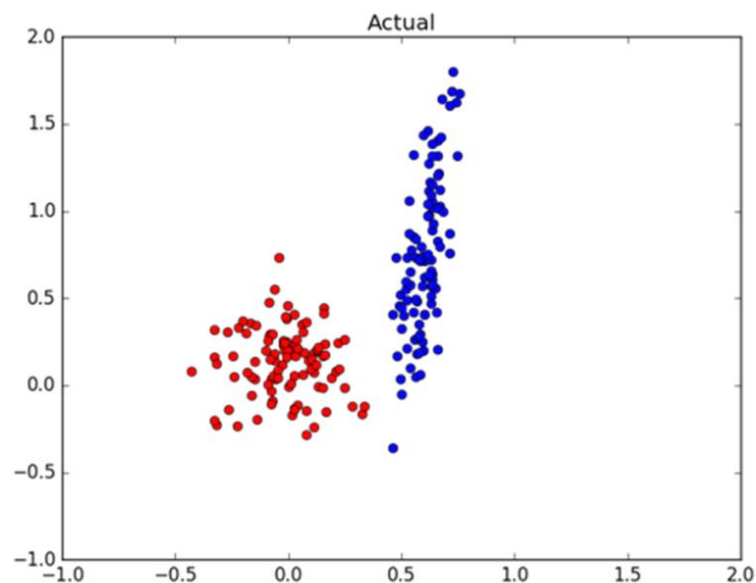
Υπάρχουν τρεις σημαντικές ανησυχίες για τη δημιουργία κανόνων συσχέτισης :

- Η επιλογή του σωστού συνόλου αντικειμένων.
- Η δημιουργία κανόνων αποκρυπτογραφώντας τις μετρήσεις στην συνυπάρχουσα μήτρα (matrix).
- Η υπέρβαση των πρακτικών ορίων που επιβάλλονται από χιλιάδες ή δέκα χιλιάδες αντικείμενα.
-

Big Data (Μεγάλα Δεδομένα)

Ορισμός Big Data

Είναι ο όρος που χρησιμοποιείται για τον τεράστιο όγκο δεδομένων που παράγονται κάθε δευτερόλεπτο, δομημένα ή όχι, από εταιρείες, προϊόντα και



υπηρεσίες, και τα οποία είναι πολύ μεγάλα και/ή πολύ περίπλοκα για να αναλυθούν, χωρίς προηγούμενη θεραπεία. Δηλαδή δεδομένα σε τεράστιο όγκο.

Τα Big Data έχουν πέντε βασικά χαρακτηριστικά, τα "five Vs":

- **Όγκος:** ο όγκος των δεδομένων καθορίζει εάν μπορεί να θεωρηθεί ή όχι Big Data.
- **Ταχύτητα:** η ταχύτητα με την οποία παράγονται τα δεδομένα, συνήθως σε πραγματικό χρόνο.
- **Ποικιλία:** Τα δεδομένα μπορούν να είναι δομημένα (οργανωμένα σε υπολογιστικά φύλλα, βρίσκονται σε βάση δεδομένων) και μη δομημένα (τα οποία δεν μπορούν να οργανωθούν σε σειρές και στήλες όπως βίντεο και σχόλια σε κοινωνικά δίκτυα).
- **Ειλικρίνεια:** η ποιότητα των δεδομένων είναι σημαντική για χρήση.
- **Αξία:** πρέπει να έχουν δυνατότητα δημιουργίας εσόδων, ώστε να δικαιολογείται η προσπάθεια αντιμετώπισής τους.

Λόγω της εκθετικής αύξησης της υπολογιστικής ισχύος που περιγράφεται στο νόμο του Moore, η ποσότητα των δεδομένων δεν μπορεί να είναι ακριβές κριτήριο για το αν είναι μεγάλο.

Για παράδειγμα, σήμερα τα μεγάλα δεδομένα μετρώνται σε terabyte και αύριο θα είναι σε petabyte. Επομένως, το κύριο χαρακτηριστικό των Big Data είναι ο βαθμός δομής και οι επιλογές παρουσίασης τους.

Μια ζωντανή απεικόνιση μεγάλων δεδομένων είναι οι συνεχώς εισερχόμενες πληροφορίες από αισθητήρες ή συσκευές εγγραφής ήχου και βίντεο, ροές μηνυμάτων από κοινωνικά δίκτυα, μετεωρολογικά δεδομένα, συντεταγμένες γεωεντοπισμού συνδρομητών κινητής τηλεφωνίας κ.λπ.

Για παράδειγμα, εδώ είπαμε πώς η PJSC Gazpromneft συνέλεξε και ανέλυσε περισσότερα από 200 εκατομμύρια διαφορετικά αρχεία από ελεγκτές συστημάτων ελέγχου σε πετρελαιοπηγές, αρχεία επανεκκίνησης τάσης από αρχεία καταγραφής έκτακτης ανάγκης, χαρακτηριστικά λειτουργίας της αντλίας και χαρακτηριστικά των συνθηκών των φρεατίων για να σχηματίσει και να δοκιμάσει υποθέσεις σχετικά με

τα αίτια των αστοχιών και ο εντοπισμός προηγουμένως άγνωστων σχέσεων στη λειτουργία του εξοπλισμού άντλησης

Έτσι, πηγές μεγάλων δεδομένων μπορεί να είναι:

- Διαδίκτυο - κοινωνικά δίκτυα, ιστολόγια, μέσα ενημέρωσης, φόρουμ, ιστότοποι, Διαδίκτυο των πραγμάτων (Διαδίκτυο των πραγμάτων , IoT);
- εταιρικές πληροφορίες - συναλλαγές, αρχεία, βάσεις δεδομένων και αποθήκευση αρχείων.
- ενδείξεις οργάνων - αισθητήρες, αισθητήρες, καταγραφείς κ.λπ.

ΣΤΑΔΙΑ ΕΡΓΑΣΙΑΣ ΜΕ BIG DATA

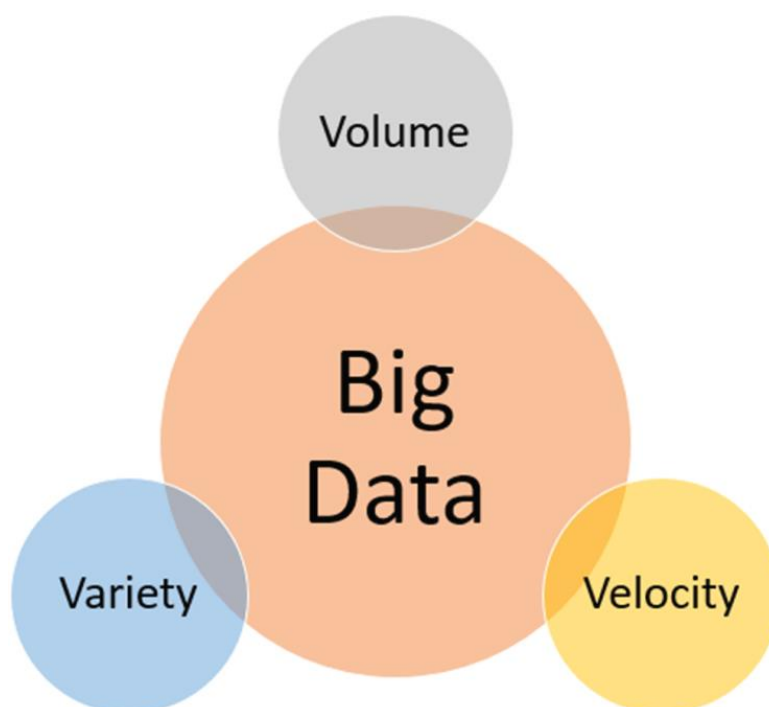
Για να αποκτήσετε μια υπόθεση εργασίας σχετικά με τις αιτίες συγκεκριμένων καταστάσεων, ειδικότερα, πώς οι αστοχίες εξοπλισμού σχετίζονται με τις συνθήκες παροχής τάσης ή για να προβλέψετε το μέλλον, για παράδειγμα, την πιθανότητα έγκαιρης αποπληρωμής ενός δανείου από ιδιώτη δανειολήπτη, η ανάλυση του μεγάλοι όγκοι δομημένων και μη δομημένων πληροφοριών εκτελούνται σε διάφορα στάδια:

- καθαρισμός δεδομένων (καθαρισμός δεδομένων) - αναζήτηση και διόρθωση σφαλμάτων στο κύριο σύνολο πληροφοριών, για παράδειγμα, σφάλματα χειροκίνητης εισαγωγής (τυπογραφικά λάθη), λανθασμένες τιμές από όργανα μέτρησης λόγω βραχυπρόθεσμων βλαβών κ.λπ.
- δημιουργία προγνωστικών παραγόντων (μηχανική χαρακτηριστικών) - μεταβλητές για τη δημιουργία αναλυτικών μοντέλων, για παράδειγμα, εκπαίδευση, εργασιακή εμπειρία, φύλο και ηλικία δυνητικού δανειολήπτη.
- δημιουργία και εκπαίδευση ενός αναλυτικού μοντέλου (επιλογή μοντέλου) για την πρόβλεψη της μεταβλητής στόχου (στόχος). Έτσι ελέγχονται οι υποθέσεις σχετικά με την εξάρτηση της μεταβλητής στόχου από προγνωστικούς παράγοντες. Για παράδειγμα, πόσες ημέρες καθυστερεί ένα δάνειο για δανειολήπτη με δευτεροβάθμια εκπαίδευση και εργασιακή εμπειρία μικρότερη των 3 μηνών.

ΜΕΘΟΔΟΙ ΚΑΙ ΕΡΓΑΛΕΙΑ ΓΙΑ ΕΡΓΑΣΙΑ ΜΕ BIG DATA

Οι κύριες μέθοδοι συλλογής και ανάλυσης μεγάλων δεδομένων περιλαμβάνουν τα ακόλουθα:

- Εξόρυξη δεδομένων - συνειρμικοί κανόνες μάθησης, ταξινόμηση , ανάλυση συστάδων και παλινδρόμησης.
- crowdsourcing - κατηγοριοποίηση και εμπλουτισμός δεδομένων από λαϊκές δυνάμεις, δηλ. με την εθελοντική βοήθεια τρίτων.
- ανάμειξη και ενσωμάτωση ετερογενών δεδομένων όπως η επεξεργασία ψηφιακού σήματος και η επεξεργασία φυσικής γλώσσας.
- μηχανική μάθηση (Machine Learning), συμπεριλαμβανομένων τεχνητών νευρωνικών δικτύων, ανάλυσης δικτύου, μεθόδων βελτιστοποίησης και γενετικών αλγορίθμων.
- αναγνώριση μοτίβου;
- προγνωστική ανάλυση?



- μοντελοποίηση προσομοίωσης.

- χωρική και στατιστική ανάλυση·
- οπτικοποίηση αναλυτικών δεδομένων - σχέδια, γραφήματα, διαγράμματα, πίνακες.

Τα εργαλεία λογισμικού και υλικού για εργασία με Big Data παρέχουν επεκτασιμότητα, παράλληλους υπολογισμούς και διανομή, επειδή Η συνεχής αύξηση του όγκου είναι ένα από τα κύρια χαρακτηριστικά των μεγάλων δεδομένων. Οι κύριες τεχνολογίες περιλαμβάνουν μη σχεσιακές βάσεις δεδομένων (NoSQL), μοντέλο επεξεργασίας πληροφοριών MapReduce , στοιχεία οικοσυστήματος συμπλέγματος Hadoop , γλώσσες προγραμματισμού R και Python, καθώς και εξειδικευμένα προϊόντα Apache (Spark , AirFlow , Kafka , HBase , κ.λπ.

Ο ορισμός της Microsoft για τα Big Data είναι πιο περιεκτικός:

«Big Data, είναι ο όρος που χρησιμοποιείται όλο και περισσότερο για να περιγράψει τη διαδικασία εφαρμογής σημαντικής υπολογιστικής ισχύος - την τελευταία λέξη της μηχανικής μάθησης και της τεχνητής νοημοσύνης - σε μαζικά και εξαιρετικά πολύπλοκα σύνολα πληροφοριών».

Πώς να χρησιμοποιήσετε τα μεγάλα δεδομένα; 3 στάδια εφαρμογής μεγάλων δεδομένων

Κάτω από το ψηφιακό κύμα, όλες οι βιομηχανίες έχουν αρχίσει να σκέφτονται πώς να χρησιμοποιούν τα μεγάλα δεδομένα για να βοηθήσουν την ανάπτυξη των επιχειρήσεων και τη διαμόρφωση πολιτικής. Αλλά πώς να χρησιμοποιήσετε αποτελεσματικά τα μεγάλα δεδομένα ; Μέσω των υπηρεσιών δεδομένων του Vron, μπορείτε να ολοκληρώσετε γρήγορα: συλλογή δεδομένων, ανάλυση δεδομένων και χρήση δεδομένων.

1. Συλλογή δεδομένων

Τα λεγόμενα μεγάλα δεδομένα αναφέρονται στα τεράστια δεδομένα που λαμβάνονται μέσω διαφορετικών πηγών και καναλιών. Σήμερα, εάν οι εταιρείες θέλουν να συλλέξουν δεδομένα, οι μέθοδοι έχουν γίνει πολύ διαφορετικές. Η τεχνολογία παρακολούθησης συμπεριφοράς, η χρήση συσκευών IoT κ.λπ., μπορούν να συλλέξουν πρώτους κατασκευαστές δεδομένα από χρήστες. Επιπλέον, υπάρχουν

ολοένα και περισσότερα δεδομένα μη πρώτου μέρους που μπορούν να συλλάβουν πληροφορίες εκτός του ιστότοπου των χρηστών, συμπεριλαμβανομένων δεδομένων δεύτερου μέρους που λαμβάνονται μέσω ανταλλαγής και κοινής χρήσης, καθώς και οποιωνδήποτε άλλων δεδομένων τρίτων που σχετίζονται με επιχειρηματικές ανάγκες. Ωστόσο, λόγω της διαφοροποίησης των δεδομένων και της μείωσης του ορίου για τη λήψη δεδομένων, η ζήτηση για συλλογή δεδομένων από τις επιχειρήσεις έχει αυξηθεί σημαντικά και έχει προκύψει η ζήτηση για αποθήκευση και εφαρμογή δεδομένων. Με την κλίμακα των μεγάλων δεδομένων, όλο και περισσότερες επιχειρήσεις επιλέγουν να συλλέγουν και να αποθηκεύουν δεδομένα μέσω υπηρεσιών cloud. Μόνο μέσω εξαιρετικά ελαστικών και επεκτάσιμων υπηρεσιών cloud μπορεί να ολοκληρωθεί η κατασκευή ενός περιβάλλοντος μεγάλων δεδομένων.

2. Ανάλυση Δεδομένων

Ένας μεγάλος όγκος δεδομένων που λαμβάνονται μέσω πολλαπλών καναλιών είναι συχνά ακατέργαστα δεδομένα, τα οποία δεν μπορούν να χρησιμοποιηθούν απευθείας. Πρέπει να περάσουν από ένα επίπεδο μετά το άλλο επίπεδο επεξεργασίας, συμπεριλαμβανομένων:

- επιθεώρηση δεδομένων
- καθαρισμός δεδομένων
- μετασχηματισμός δεδομένων

κ.λπ., συνήθως βασίζονται σε μηχανικούς δεδομένων ή επιστήμονες δεδομένων για την ολοκλήρωση. Τα αρχικά δεδομένα περιλαμβάνουν μη δομημένα δεδομένα και δομημένα δεδομένα, τα οποία αποθηκεύονται με τον πιο βολικό τρόπο στη λίμνη δεδομένων (Λίμνη δεδομένων) στο περιβάλλον cloud. Λόγω της πολυπλοκότητας

των δεδομένων σε αυτό το στάδιο, η ανάλυση δεδομένων Το τεχνικό όριο είναι σχετικά υψηλό, και οι αναλυτές δεδομένων πρέπει να χρησιμοποιούν πιο ευέλικτες γλώσσες προγραμματισμού που σχετίζονται με την επιστήμη των δεδομένων (όπως Scala, Python, R, κ.λπ.) για την εκτέλεση ανάλυσης δεδομένων.

Μετά από προκαταρκτική επεξεργασία, τα δεδομένα θα αποθηκευτούν στην αποθήκη δεδομένων (Data Warehouse) σύμφωνα με τον τύπο δεδομένων. Η κοινή αποθήκη δεδομένων μεγάλων δεδομένων είναι η Hive με βάση το Hadoop. Αυτή τη στιγμή, τα δεδομένα είναι δομημένα δεδομένα (δομημένα δεδομένα), μεγάλα Οι αναλυτές δεδομένων μπορούν να ολοκληρώσουν γρήγορα την ανάλυση μεγάλων δεδομένων μέσω SQL (Structured Query Language). Μεταξύ αυτών, η ανάλυση δεδομένων μπορεί απλά να χωριστεί σε περιγραφική ανάλυση και προγνωστική ανάλυση. Σύμφωνα με τα αποτελέσματα που παρουσιάζονται στα δεδομένα, διαμορφώνονται κατάλληλες λύσεις για τη βελτιστοποίηση και τη βελτίωση της τρέχουσας κατάστασης.

3. Εφαρμογή δεδομένων

Οπτικοποιήστε δεδομένα ανάλυσης δεδομένων και προβείτε σε ενέργειες για επαλήθευση.

Οι συνήθεις εφαρμογές δεδομένων στην επιχείρηση περιλαμβάνουν: ομαδοποίηση χρηστών, αναφορά δεδομένων, πίνακες εργαλείων δεδομένων κ.λπ. Μέσω της ανάλυσης μεγάλων δεδομένων και των αλγορίθμων AI, οι χρήστες μπορούν να επισημανθούν σύμφωνα με διαφορετικά χαρακτηριστικά, προτιμήσεις, συμπεριφορές κ.λπ., για να περιγράψουν χρήστες με διαφορετικά προφίλ και να ομαδοποιήσουν τους χρήστες σε διαφορετικές ομάδες για την επίτευξη του σκοπού της προώθησης μάρκετινγκ.

Για παράδειγμα, αφού διακριθούν οι χρήστες με διαφορετικές δυνάμεις δαπανών μέσω αλγορίθμων τεχνητής νοημοσύνης, παρέχονται κατάλληλες πληροφορίες και υπηρεσίες στους χρήστες ανάλογα με τις διαφορετικές δυνάμεις δαπανών τους.

Οι αναφορές δεδομένων είναι μια σημαντική πηγή πληροφοριών για την υποστήριξη της λήψης αποφάσεων στις επιχειρήσεις. Μέσω της ανάλυσης μεγάλων δεδομένων, εξάγονται δεδομένα μεγάλης κλίμακας και πολυδιάστατα για την εξαγωγή σημαντικών και βασικών πληροφοριών και στη συνέχεια περιορίζονται στην προοπτική των επιχειρηματικών προβλημάτων. Μάθετε στρατηγική για την επίλυση του προβλήματος. Η κοινή μέθοδος αναφοράς δεδομένων είναι η διαδικασία PDCA, δηλαδή Σχεδιασμός, Εκτέλεση, Έλεγχος και Δράση, η διαδικασία σχεδιασμού στόχων, εκτέλεσης ανάπτυξης, επαλήθευσης αποτελεσμάτων και προσαρμογής στρατηγικής.

Εκτός από τις αναφορές δεδομένων, οι πίνακες ελέγχου δεδομένων είναι επίσης ένα κοινό σύστημα υποστήριξης αποφάσεων. Μέσω μιας ευέλικτης και διαδραστικής διεπαφής χρήστη, μπορούν να διερευνηθούν διάφορες ρυθμίσεις δεδομένων και αποτελέσματα ανάλυσης και να βρεθεί μια πολιτική επίλυσης προβλημάτων. Τα αναπτυξιακά σχέδια μπορούν επίσης να κάνουν περισσότερα με λιγότερα.

Ανάλυση εφαρμογών μεγάλων δεδομένων σε διάφορους τομείς

Πώς μπορούν λοιπόν να εφαρμοστούν αυτά τα μεγάλα δεδομένα και η ανάλυση μεγάλων δεδομένων; Παρακάτω χρησιμοποιούμε τρία παραδείγματα για να σας εξηγήσουμε και να αναλύσουμε.

Πώς μπορούν να χρησιμοποιηθούν τα Big Data;

Οι λύσεις επεξεργάζονται δεδομένα για την εξαγωγή πληροφοριών από διαφορετικές πηγές: κοινωνικά δίκτυα, υπηρεσίες, χρηματιστήρια, ειδήσεις κ.λπ. Εφαρμόζονται σε βιομηχανίες και εταιρείες.

- Μειώστε το κόστος.
- Αύξηση παραγωγικότητας.
- Μειώστε το χρόνο.
- Βελτιώστε τη λήψη αποφάσεων.
- Αύξηση του περιθωρίου κέρδους.

Κανονικά μόνο οι άνθρωποι θα μπορούσαν να χειριστούν μη δομημένα δεδομένα. Σήμερα, ωστόσο, τα συστήματα μπορούν να αναλύσουν τις εισροές σε σύντομο χρονικό διάστημα και να εξάγουν πολύτιμες πληροφορίες.



Εργαλεία ανάλυσης Big Data

Η ανάγκη για διαχείριση Μεγάλων Δεδομένων, οδήγησε στη δημιουργία νέων προγραμμάτων και εργαλείων, όπως η Hadoop και το οικοσύστημα αυτού NoSQL, αλλά και το Map Reduce που επιτρέπουν την επεξεργασία των δεδομένων σε μεγάλη κλίμακα.

Hadoop

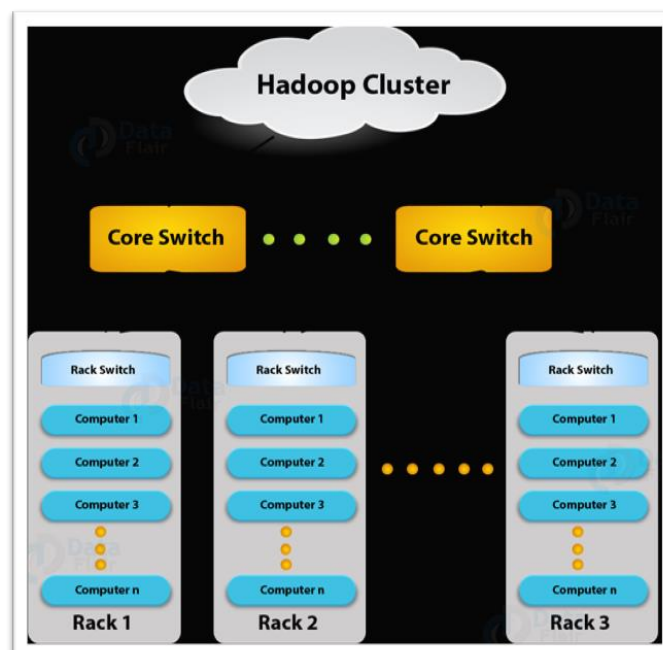
Πρόκειται για ένα λογισμικό ανοιχτού κώδικα που υποστηρίζει κατανεμημένη επεξεργασία μεγάλου όγκου δεδομένων και παρέχει μια υλοποίηση του MapReduce. Η Hadoop βασίστηκε στο Google Map Reduce framework και το Google File System (GFS). Είναι η πιο διαδεδομένη υλοποίηση του MapReduce και χρησιμοποιείται για διδακτικούς σκοπούς σε αρκετά πανεπιστήμια όλου του κόσμου, αλλά και σε μεγάλους οργανισμούς για την επεξεργασία μεγάλων δεδομένων εισόδου. Κάποιοι οργανισμοί που διατηρούν clusters για εκτέλεση Hadoop εργασιών είναι: η Yahoo, η Amazon, η AOL, Facebook, Google κ.α

Το Hadoop, κατανέμει τα δεδομένα και την ανάλυσή τους σε ομάδες υπολογιστών (clusters), ώστε να επεξεργαστούν ταυτόχρονα τα δεδομένα, εξοικονομώντας πόρους και χρόνο.

Ειδικότερα, γίνεται προώθηση των δεδομένων και των προβλημάτων στον υπολογιστή master της ομάδας και αυτός με τη σειρά του κατακερματίζει το πρόβλημα σε μικρότερα. Κάθε νέο πρόβλημα θα το προωθήσει με τη σειρά του σε κάθε ένα από τους υπολογιστές της υπόλοιπης ομάδας.

Κάθε ένας από τους υπολογιστές της ομάδας, θα επιλύσει το δικό του πρόβλημα και θα επιστρέψει τη λύση στο master υπολογιστή, που θα συνδυάσει όλες τις υπάρχουσες λύσεις των υποπροβλημάτων για να βρει τη λύση στο αρχικό πρόβλημα.

Σε όλη αυτή τη διαδικασία στηρίζεται το Hadoop, ενώ ένα επιπλέον προτέρημα είναι ότι επιτυγχάνει να ανακτήσει δεδομένα σε περίπτωση που ένας υπολογιστής πάθει ζημιά.



Πλεονεκτήματα Hadoop

Τα βασικότερα πλεονεκτήματα του Hadoop είναι τα παρακάτω:

Οικονομία πόρων: Κατανομή δεδομένων και επεξεργασίας σε ομάδες υπολογιστών που αποτελούνται από χιλιάδες κοινούς υπολογιστές.

Επεκτασιμότητα: Δυνατότητα αξιόπιστης αποθήκευσης και επεξεργασίας μέχρι και petabytes δεδομένων.

Αποδοτικότητα : Με τη κατανομή των δεδομένων, η επεξεργασία γίνεται ταυτόχρονα σε όλους τους κόμβους, παρέχοντας γρήγορη εκτέλεση των εργασιών.

Αξιοπιστία: Επιτυγχάνεται μέσω της αυτόματης διατήρησης πολλαπλών αντιγράφων των δεδομένων καθώς και αυτόματης ανάθεσης των εργασιών υπολογισμού σε νέους κόμβους σε περίπτωση βλάβης.

Το οικοσύστημα της Hadoop

Το Hadoop χρησιμοποιείται σε συνδυασμό με πολλά εργαλεία της Apache τα οποία ανήκουν ένα πλήρες περιβάλλον Big Data Analytics.

Τα εργαλεία αυτά έχουν:

Pig:

Το pig είναι μια scripting γλώσσα για την διαχείριση δεδομένων με λειτουργίες ETL

Hive:

Το Hive παρέχει μια γλώσσα ερωτημάτων, παρόμοια με την SQL.

Oozie:

Το Oozie είναι ένας χρονοπρογραμματιστής (scheduler) εργασιών που χρησιμοποιείται για τη διαχείριση των εργασιών του Hadoop.

Σqoop:

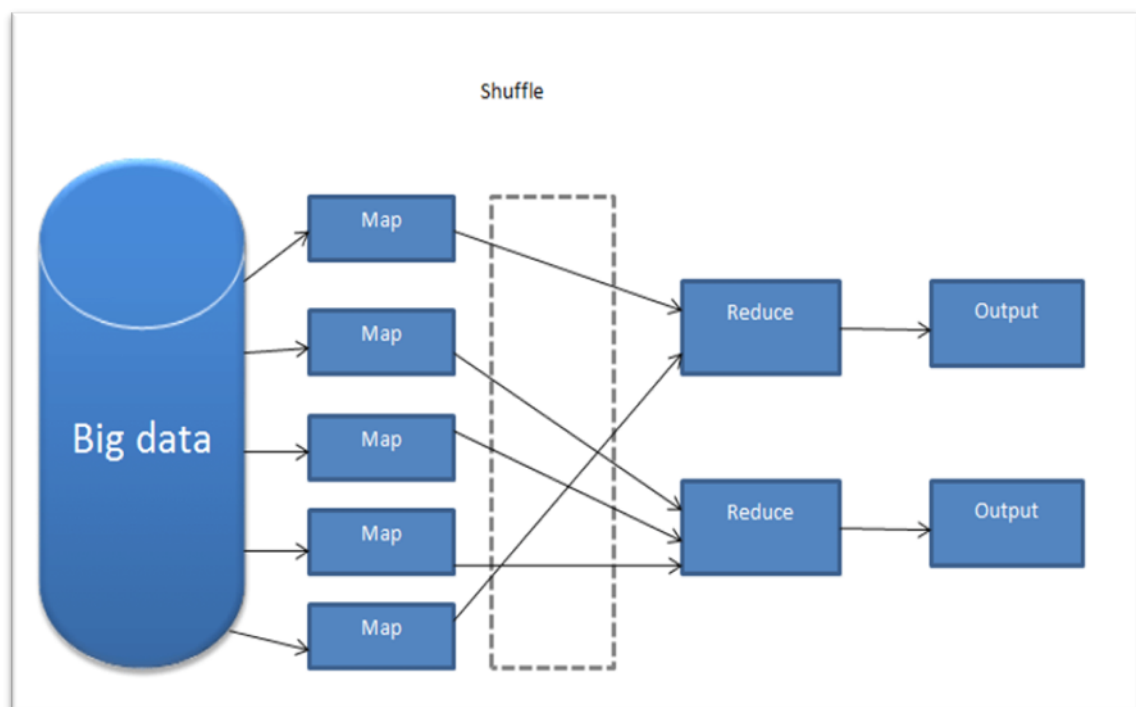
Το Σqoop παρέχει εργαλεία για τη μεταφορά δεδομένων μεταξύ Hadoop και σχεσιακών βάσεων δεδομένων.

Αξίζει να συμποσούμε ότι αρκετοί προμηθευτές που παρέχουν έτοιμες πλατφόρμες με προεγκατεστημένα όλα τα λογισμικά του Hadoop.

Κάποιοι από αυτούς είναι:

- Υπηρεσίες Web της Amazon
- Η πλατφόρμα Google Cloud
- Cloudera
- HortonWorks

MapReduce



Πρόκειται για ένα από τα πιο σημαντικά εργαλεία για την ανάλυση των μεγάλων δεδομένων.

Είναι ένα προγραμματιστικό μοντέλο, σε συνδυασμό με τη σχετική υλοποίηση για δημιουργία και επεξεργασία τεράστιων όγκων δεδομένων.

Παρόλο που το MapReduce είναι απλό, ταυτόχρονα είναι και ένα πολύ δυνατό πλαίσιο το οποίο κάνει χρήση ενός αλγόριθμου, που παραλληλοποιεί και κατανέμει το σύνολο του όγκου που πρόκειται να επεξεργαστεί, μοιράζοντας κομμάτια του σε πολλούς υπολογιστές για επεξεργασία.

Αρχικά, μοιράζει το σύνολο του όγκου των εργασιών σε πολλαπλούς υπολογιστές, οι οποίοι εκτελούν τα κομμάτια που τους αναθέτουν ταυτόχρονα (φάση map).

Στη συνέχεια, όλα τα αποτελέσματα συγκεντρώνονται και αναλύονται συνολικά πριν επιστραφούν (φάση reduce).

Υπολογιστικό νέφος (cloud computing)

cloud computing είναι ένας τύπος κατανεμημένου υπολογισμού, ο οποίος αναφέρεται στην αποσύνθεση τεράστιων προγραμμάτων επεξεργασίας υπολογιστών δεδομένων σε αμέτρητα μικρά προγράμματα μέσω του δικτύου "σύννεφου" και στη συνέχεια επεξεργασία και ανάλυση αυτών των μικρών προγραμμάτων μέσω ενός συστήματος που αποτελείται από πολλούς διακομιστές. Το πρόγραμμα έχει το αποτέλεσμα και το επιστρέφει στον χρήστη. Στις πρώτες μέρες του cloud computing, με απλά λόγια, ήταν απλός κατανεμημένος υπολογισμός, η επίλυση της διανομής εργασιών και η συγχώνευση αποτελεσμάτων υπολογιστών. Επομένως, το cloud computing ονομάζεται επίσης υπολογιστικό δίκτυο. Μέσω αυτής της τεχνολογίας, η επεξεργασία δεκάδων χιλιάδων δεδομένων μπορεί να ολοκληρωθεί σε πολύ σύντομο χρονικό διάστημα (αρκετά δευτερόλεπτα), επιτυγχάνοντας έτσι ισχυρές υπηρεσίες δικτύου.

ΣΦΑΙΡΙΚΗ ΕΙΚΟΝΑ

Το "Cloud" είναι ουσιαστικά ένα δίκτυο. Με μια στενή έννοια, το cloud computing είναι ένα δίκτυο που παρέχει πόρους. Οι χρήστες μπορούν να αποκτήσουν πόρους στο "σύννεφο" ανά πάσα στιγμή, να τους χρησιμοποιήσουν ανάλογα με τις ανάγκες τους και μπορούν να θεωρηθούν ως απεριόριστα επεκτάσιμοι. Μπορείτε να πληρώσετε για τη χρήση. Το "σύννεφο" είναι σαν ένα υδραγωγείο. Μπορούμε να λάβουμε νερό ανά πάσα στιγμή, και δεν υπάρχει όριο σε αυτό. Μπορούμε να

πληρώσουμε τα έργα ύδρευσης ανάλογα με την κατανάλωση νερού του σπιτιού μας.

Με την ευρεία έννοια, το cloud computing είναι μια υπηρεσία που σχετίζεται με την τεχνολογία πληροφοριών, το λογισμικό και το Διαδίκτυο. Αυτή η πεισινα κοινής χρήσης υπολογιστικών πόρων ονομάζεται "σύννεφο". Το cloud computing συγκεντρώνει πολλούς υπολογιστικούς πόρους και αυτοματοποιεί τη διαχείριση μέσω λογισμικού, απαιτώντας μόνο λίγα Το ginseng και το τζίνσενγκ μπορεί να παρασχεθεί γρήγορα. Δηλαδή, η υπολογιστική ισχύς, ως εμπόρευμα, μπορεί να κυκλοφορήσει στο Διαδίκτυο, όπως το νερό, ο ηλεκτρισμός και το φυσικό αέριο, είναι εύκολα προσβάσιμη και η τιμή είναι σχετικά χαμηλή. [2]

Με μια λέξη, το cloud computing δεν είναι μια ολοκαίνουργια τεχνολογία δικτύου, αλλά μια ολοκαίνουργια ιδέα εφαρμογής δικτύου. Η βασική ιδέα του cloud computing είναι να πάρει το Διαδίκτυο ως κέντρο, να παρέχει γρήγορες και ασφαλείς υπηρεσίες υπολογιστικού νέφους και αποθήκευση δεδομένων στον ιστότοπο, έτσι ώστε κάθε χρήστης να μπορεί να χρησιμοποιεί το Διαδίκτυο ως κέντρο Όλοι στο Διαδίκτυο μπορούν να χρησιμοποιούν τους τεράστιους υπολογιστικούς πόρους και τα κέντρα δεδομένων στο δίκτυο. [2]

Το cloud computing είναι μια άλλη νέα καινοτομία στην εποχή της πληροφορίας μετά τους υπολογιστές και το Διαδίκτυο.

Ιστορικό

Το Διαδίκτυο βρίσκεται σε άνοδο από το 1960 και χρησιμοποιείται κυρίως για υπηρεσίες ηλεκτρονικού ταχυδρομείου απλού κειμένου ή ομάδες συζητήσεων μεταξύ του στρατού, των μεγάλων επιχειρήσεων κ.λπ. Μόλις το 1990 άρχισε να εισέρχεται στα συνηθισμένα νοικοκυριά. Με την ανάπτυξη των ιστοσελίδων και του ηλεκτρονικού εμπορίου, το Διαδίκτυο έγινε μια από τις ανάγκες της ζωής που οι άνθρωποι δεν μπορούν να κάνουν χωρίς. Η έννοια του υπολογιστικού νέφους προτάθηκε για πρώτη φορά στο Συνέδριο της Μηχανής Αναζήτησης τον Αύγουστο του 2006 και έγινε η τρίτη επανάσταση του Διαδικτύου.

Το cloud computing γίνεται επίσης το στρατηγικό επίκεντρο της ανάπτυξης της βιομηχανίας τεχνολογίας πληροφοριών και οι εταιρείες τεχνολογίας πληροφοριών σε όλο τον κόσμο μετατρέπονται σε υπολογιστικό νέφος. Για παράδειγμα, κάθε εταιρεία χρειάζεται να κάνει πληροφόρηση δεδομένων, να αποθηκεύει σχετικά επιχειρησιακά δεδομένα, να διεξάγει διαχείριση προϊόντων, διαχείριση προσωπικού, οικονομική διαχείριση κ.λπ., και ο βασικός εξοπλισμός για τη διαχείριση αυτών των δεδομένων είναι ο υπολογιστής.

Για μια επιχείρηση, η υπολογιστική ισχύς ενός υπολογιστή απέχει πολύ από το να καλύψει τις ανάγκες της πληροφορικής δεδομένων, επομένως η εταιρεία πρέπει να αγοράσει έναν υπολογιστή με ισχυρότερη υπολογιστική ισχύ, δηλαδή έναν διακομιστή. Για επιχειρήσεις μεγάλης κλίμακας, η υπολογιστική ισχύς ενός διακομιστή δεν είναι προφανώς αρκετή, τότε η επιχείρηση πρέπει να αγοράσει πολλούς διακομιστές ή ακόμη και να εξελιχθεί σε κέντρο δεδομένων με πολλούς διακομιστές και ο αριθμός των διακομιστών θα επηρεάσει άμεσα αυτό. Η ικανότητα επεξεργασίας της επιχείρησης του κέντρου δεδομένων. Εκτός από το υψηλό αρχικό κόστος κατασκευής, τα χρήματα που δαπανώνται για τους λογαριασμούς ρεύματος στις λειτουργικές δαπάνες των υπολογιστών είναι πολύ υψηλότερα από το κόστος επένδυσης, συν τα έξοδα συντήρησης υπολογιστών και δικτύων, αυτά τα συνολικά έξοδα είναι δύσκολα για τις μικρομεσαίες επιχειρήσεις. Έτσι εμφανίστηκε η έννοια του cloud computing.

Αναπτυξιακή πορεία

Σήμερα, το cloud computing θεωρείται επανάσταση στον τομέα των δικτύων υπολογιστών, καθώς λόγω της εμφάνισής του, ο τρόπος εργασίας και τα επιχειρηματικά μοντέλα της κοινωνίας υφίστανται επίσης τεράστιες αλλαγές.

Επιστρέφοντας στη ρίζα του υπολογιστικού νέφους, η δημιουργία και η ανάπτυξή του συνδέονται στενά με τις τεχνολογίες υπολογιστών όπως ο παράλληλος υπολογισμός και ο κατανεμημένος υπολογισμός που αναφέρθηκαν προηγουμένως, οι οποίες προάγουν την ανάπτυξη του υπολογιστικού νέφους. Ωστόσο, η ιστορία του υπολογιστικού νέφους μπορεί να ανιχνευθεί πίσω στο 1956, όταν ο Christopher Strachey δημοσίευσε μια εργασία για την εικονικοποίηση και πρότεινε επίσημα την

έννοια της εικονικοποίησης. Η εικονικοποίηση είναι ο πυρήνας της σημερινής υποδομής υπολογιστικού νέφους και το θεμέλιο της ανάπτυξης του υπολογιστικού νέφους. Στη συνέχεια, με την ανάπτυξη της τεχνολογίας δικτύου, ο οφθαλμός του cloud computing αναπτύχθηκε σταδιακά.

Στη δεκαετία του 1990, έγινε μια μεγάλη έκρηξη στα δίκτυα υπολογιστών και εμφανίστηκαν μια σειρά από εταιρείες που εκπροσωπούσε η Cisco και στη συνέχεια εμφανίστηκε η εποχή της φούσκας του δικτύου.

Το 2004, πραγματοποιήθηκε το συνέδριο Web2.0 και το Web2.0 έγινε hot spot εκείνη την εποχή, το οποίο σηματοδότησε επίσης το σκάσιμο της φούσκας του Διαδικτύου και η ανάπτυξη των δικτύων υπολογιστών εισήλθε σε ένα νέο στάδιο. Σε αυτό το στάδιο, το να επιτρέπεται σε περισσότερους χρήστες να χρησιμοποιούν τις υπηρεσίες δικτύου εύκολα και γρήγορα έχει καταστεί επείγον πρόβλημα που πρέπει να επιλυθεί στην ανάπτυξη του Διαδικτύου των Πραγμάτων. Ταυτόχρονα, ορισμένες μεγάλες εταιρείες έχουν επίσης αρχίσει να αναπτύσσουν τεχνολογίες με μεγάλες υπολογιστικές δυνατότητες, παρέχοντας στους χρήστες πιο ισχυρές Υπηρεσίες Υπολογιστικής Επεξεργασίας.

Στις 9 Αυγούστου 2006, ο Διευθύνων Σύμβουλος της Google, Eric Schmidt πρότεινε για πρώτη φορά την έννοια του "Cloud Computing" στο συνέδριο της μηχανής αναζήτησης (SESSanJose2006). Αυτή είναι η πρώτη φορά στην ιστορία του υπολογιστικού νέφους που αυτή η ιδέα προτάθηκε επίσημα και έχει μεγάλη ιστορική σημασία.

Από το 2007, το «cloud computing» έχει γίνει ένα από τα πιο ενδιαφέροντα θέματα στον τομέα των υπολογιστών και είναι επίσης μια σημαντική ερευνητική κατεύθυνση για μεγάλες επιχειρήσεις και κατασκευή Διαδικτύου. Λόγω της εισαγωγής του υπολογιστικού νέφους, ένα νέο μοντέλο τεχνολογίας Διαδικτύου και υπηρεσιών πληροφορικής έχει εμφανιστεί, πυροδοτώντας μια επανάσταση.

Το 2008, η Microsoft κυκλοφόρησε τη δημόσια πλατφόρμα cloud computing (Windows Azure Platform), η οποία άνοιξε την αυλαία του cloud computing της Microsoft. Ομοίως, το cloud computing έχει επίσης προκαλέσει θύελλα στην Κίνα

και πολλές μεγάλες εταιρείες δικτύου έχουν ενταχθεί στη σειρά του cloud computing.

Τον Ιανουάριο του 2009, η Alibaba Software ίδρυσε το πρώτο «κέντρο υπολογιστικού νέφους ηλεκτρονικού εμπορίου» στο Nanjing του Jiangsu. Τον Νοέμβριο του ίδιου έτους, ξεκίνησε το σχέδιο «Big Cloud» της πλατφόρμας υπολογιστών cloud της China Mobile. Σε αυτό το στάδιο, το cloud computing έχει εξελιχθεί σε ένα σχετικά ώριμο στάδιο.

Στις 17 Αυγούστου 2019, το Δικαστήριο Διαδικτύου του Πεκίνου δημοσίευσε τη «Λευκή Βίβλο για τη δικαστική εφαρμογή της τεχνολογίας του Διαδικτύου». Στη συνέντευξη Τύπου, αποκαλύφθηκε το Κέντρο Δικαστικών Εφαρμογών Τεχνολογίας Διαδικτύου του Πεκίνου.

Το 2020, η κλίμακα της αγοράς υπολογιστών cloud της χώρας μου θα φτάσει τα 178,1 δισεκατομμύρια γιουάν, με ρυθμό ανάπτυξης 33,6%. Μεταξύ αυτών, η δημόσια αγορά cloud έφτασε τα 99,06 δισεκατομμύρια γιουάν, σημειώνοντας αύξηση 43,7% από έτος σε έτος και η αγορά ιδιωτικού cloud έφτασε τα 79,12 δισεκατομμύρια γιουάν, σημειώνοντας αύξηση 22,6% από έτος σε έτος.

Χαρακτηριστικ cloud computing

Σε σύγκριση με την παραδοσιακή λειτουργία εφαρμογής δικτύου, το cloud computing έχει τα ακόλουθα πλεονεκτήματα και χαρακτηριστικά:

1. Τεχνολογία εικονικοποίησης.

Πρέπει να τονιστεί ότι το virtualization ξεπερνά τα όρια του χρόνου και του χώρου και είναι το πιο σημαντικό χαρακτηριστικό του cloud computing. Οι τεχνολογίες εικονικοποίησης περιλαμβάνουν εικονικοποίηση εφαρμογών και εικονικοποίηση πόρων. Όπως όλοι γνωρίζουμε, δεν υπάρχει χωρική σχέση μεταξύ της φυσικής πλατφόρμας και του περιβάλλοντος στο οποίο αναπτύσσεται η εφαρμογή, αλλά μέσω της εικονικής πλατφόρμας γίνεται η δημιουργία αντιγράφων ασφαλείας, η μετεγκατάσταση και η επέκταση δεδομένων σε αντίστοιχες λειτουργίες τερματικού.

2. Δυναμικά επεκτάσιμο.

Το Cloud Computing έχει αποτελεσματική υπολογιστική ισχύ. Η προσθήκη λειτουργιών υπολογιστικού νέφους στον αρχικό διακομιστή μπορεί να αυξήσει γρήγορα την υπολογιστική ταχύτητα και τελικά να επιτύχει τον σκοπό της επέκτασης της εφαρμογής επεκτείνοντας δυναμικά το επίπεδο εικονικοποίησης.

3. Ανάπτυξη κατ' απαίτηση.

Οι υπολογιστές περιλαμβάνουν πολλές εφαρμογές, λογισμικό προγραμμάτων κ.λπ., διαφορετικές εφαρμογές αντιστοιχούν σε διαφορετικές βιβλιοθήκες πόρων δεδομένων, επομένως οι χρήστες χρειάζονται ισχυρή υπολογιστική ισχύ για να αναπτύξουν πόρους για την εκτέλεση διαφορετικών εφαρμογών και οι πλατφόρμες υπολογιστικού νέφους μπορούν να καταναείμουν γρήγορα την υπολογιστική ισχύ ανάλογα με τις ανάγκες των χρηστών.

4. Υψηλή ευελιξία.

Επί του παρόντος, οι περισσότεροι πόροι πληροφορικής, λογισμικό και υλικό στην αγορά υποστηρίζουν την εικονικοποίηση, όπως το δίκτυο αποθήκευσης, το λειτουργικό σύστημα και το λογισμικό ανάπτυξης και το υλικό. Τα στοιχεία εικονικοποίησης είναι ενοποιημένα στην εικονική δεξαμενή πόρων του συστήματος cloud για διαχείριση. Μπορεί να φανεί ότι η συμβατότητα του υπολογιστικού νέφους είναι πολύ ισχυρή. Δεν μπορεί μόνο να είναι συμβατό με μηχανήματα χαμηλού προφίλ και προϊόντα υλικού από διαφορετικούς κατασκευαστές, αλλά και περιφερειακά για την απόκτηση υπολογιστών υψηλότερης απόδοσης.

5. Υψηλή αξιοπιστία.

Εάν ο διακομιστής αποτύχει, δεν θα επηρεάσει την κανονική λειτουργία των υπολογιστών και των εφαρμογών. Λόγω της αποτυχίας ενός διακομιστή, οι εφαρμογές που διανέμονται σε διαφορετικούς φυσικούς διακομιστές μπορούν να ανακτηθούν μέσω τεχνολογίας εικονικοποίησης ή νέοι διακομιστές μπορούν να αναπτυχθούν για υπολογιστές χρησιμοποιώντας δυνατότητες δυναμικής επέκτασης.

6. Υψηλή απόδοση κόστους.

Η τοποθέτηση πόρων σε μια εικονική ομάδα πόρων για ενοποιημένη διαχείριση βελτιστοποιεί τους φυσικούς πόρους σε κάποιο βαθμό, οι χρήστες δεν χρειάζονται πλέον ακριβούς κεντρικούς υπολογιστές με μεγάλο χώρο αποθήκευσης και μπορούν να επιλέξουν σχετικά φθηνούς υπολογιστές για να σχηματίσουν ένα σύννεφο, το οποίο μειώνει το κόστος αφενός και την απόδοση των υπολογιστών από την άλλη. Δεν είναι κατώτερο από τα mainframes.

7. Επεκτασιμότητα.

Οι χρήστες μπορούν να επωφεληθούν από τις συνθήκες ταχείας ανάπτυξης του λογισμικού εφαρμογής για να επεκτείνουν πιο απλά και γρήγορα τις υπάρχουσες υπηρεσίες και τις νέες υπηρεσίες που χρειάζονται. Για παράδειγμα, εάν ο εξοπλισμός αποτύχει στο σύστημα υπολογιστών νέφους, ο χρήστης δεν θα παρεμποδιστεί ούτε σε επίπεδο υπολογιστή ούτε στη συγκεκριμένη εφαρμογή.

Η λειτουργία δυναμικής επέκτασης του υπολογιστικού νέφους υπολογιστή μπορεί να χρησιμοποιηθεί για να επεκταθεί αποτελεσματικά ο διακομιστής. Αυτό διασφαλίζει ότι οι εργασίες ολοκληρώνονται με τακτικό τρόπο. Στην περίπτωση της δυναμικής επέκτασης εικονικών πόρων, οι εφαρμογές μπορούν να επεκταθούν αποτελεσματικά ταυτόχρονα και το επίπεδο λειτουργίας του υπολογιστικού νέφους μπορεί να βελτιωθεί.

Τύπος υπηρεσίας

Γενικά, οι τύποι υπηρεσιών του χωρίζονται σε τρεις κατηγορίες, δηλαδή την Υποδομή ως Υπηρεσία (IaaS), την Πλατφόρμα ως Υπηρεσία (PaaS) και το Λογισμικό ως Υπηρεσία (SaaS). Αυτές οι 3 υπηρεσίες υπολογιστικού νέφους αναφέρονται μερικές φορές ως στοίβες υπολογιστικού νέφους επειδή δημιουργούν τη στοίβα, κάθονται η μία πάνω στην άλλη, εδώ είναι μια επισκόπηση των τριών υπηρεσιών:

1. Υποδομή ως υπηρεσία (IaaS)

Το Infrastructure-as-a-Service είναι μία από τις κύριες κατηγορίες υπηρεσιών που παρέχει εικονικούς υπολογιστικούς πόρους, όπως εικονικές μηχανές, αποθήκευση,

δικτύωση και λειτουργικά συστήματα σε άτομα ή οργανισμούς παρόχων υπολογιστικού νέφους.

2. Πλατφόρμα ως υπηρεσία (PaaS)

Η πλατφόρμα ως υπηρεσία είναι μια κατηγορία υπηρεσιών που παρέχει στους προγραμματιστές μια πλατφόρμα για τη δημιουργία εφαρμογών και υπηρεσιών μέσω του παγκόσμιου Διαδικτύου. Το PaaS παρέχει ένα περιβάλλον ανάπτυξης κατ' απαίτηση για την ανάπτυξη, τη δοκιμή και τη διαχείριση εφαρμογών λογισμικού.

3. Λογισμικό ως υπηρεσία (SaaS)

Το Software-as-a-Service είναι επίσης μια κατηγορία των υπηρεσιών της, που προσφέρει εφαρμογές λογισμικού pay-as-you-go μέσω Διαδικτύου, οι πάροχοι υπολογιστικού νέφους φιλοξενούν και διαχειρίζονται τις εφαρμογές λογισμικού και επιτρέπουν στους χρήστες τους να συνδέονται και να έχουν πρόσβαση στις εφαρμογές μέσω του παγκόσμιου Διαδικτύου

Software as a service (SaaS)

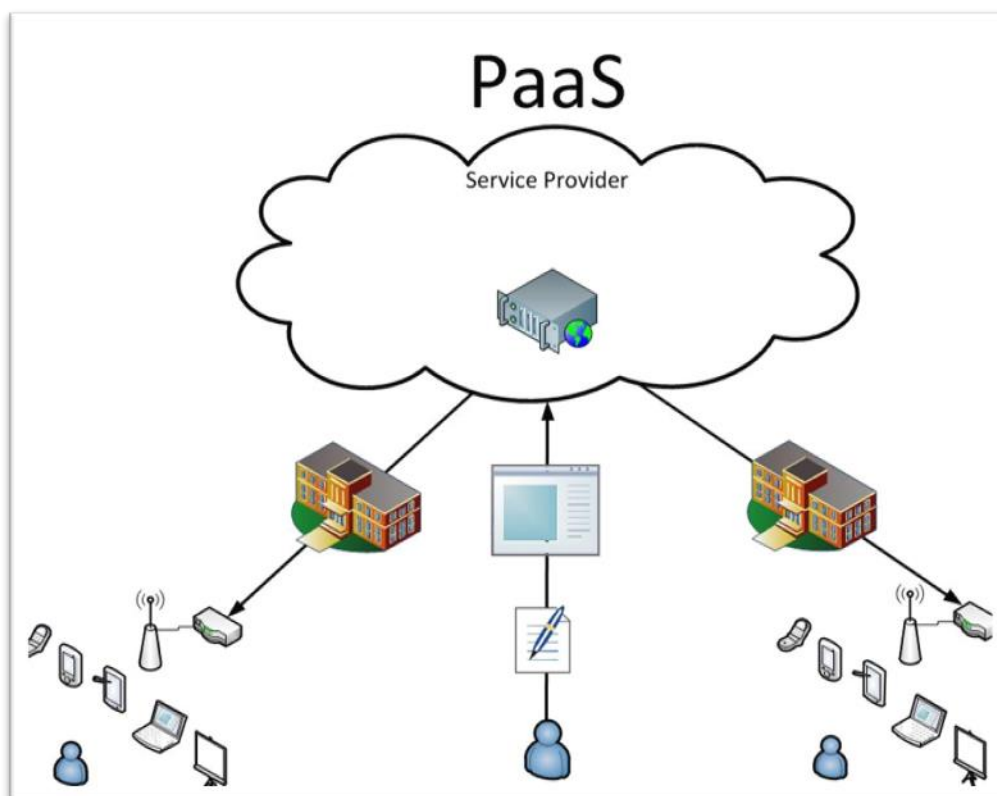
Το λογισμικό ως υπηρεσία (SaaS) είναι ένα μοντέλο αδειοδότησης και παράδοσης λογισμικού στο οποίο το λογισμικό αδειοδοτείται με συνδρομή και φιλοξενείται κεντρικά στο cloud. Οι καταναλωτές δεν χρειάζεται να διαχειρίζονται την αρχιτεκτονική λογισμικού, τη συντήρηση λογισμικού και την υποκείμενη υποδομή. Οι τελικοί χρήστες έχουν πρόσβαση σε εφαρμογές SaaS χρησιμοποιώντας thin clients όπως web clients.

Τεχνικά πλεονεκτήματα SaaS

Το SaaS παρέχει ευέλικτη επεκτασιμότητα και υψηλή διαθεσιμότητα σε διαφορετικές γεωγραφικές τοποθεσίες. Για παράδειγμα, εάν ο αριθμός των χρηστών αυξάνεται γρήγορα, μια εφαρμογή SaaS θα εξυπηρετεί χρήστες στην Ασία και την Ευρώπη χωρίς προβλήματα απόδοσης ή προβλήματα επεκτασιμότητας.

Παραδοσιακά, η εφαρμογή εταιρικού λογισμικού διαρκεί συχνά μήνες από την έναρξη του έργου έως την έναρξη της παραγωγής. Το SaaS μειώνει τον χρόνο παράδοσης από μήνες σε ημέρες ή και ώρες.

Η διαχείριση ενημερώσεων κώδικα των εφαρμογών και των υποκείμενων στοίβών τους είναι πάντα ένα επίπονο σημείο για κάθε επιχείρηση, ειδικά όταν πρόκειται για συντονισμό με πολιτικές ή πρότυπα όπως το PCI DSS. Με το SaaS, οι καταναλωτές δεν χρειάζεται να χρησιμοποιούν δικούς τους πόρους ή να συντονίζονται/επικοινωνούν με διαφορετικούς ενδιαφερόμενους για δραστηριότητες διαχείρισης ενημερώσεων κώδικα.



Platform as a service (PaaS)

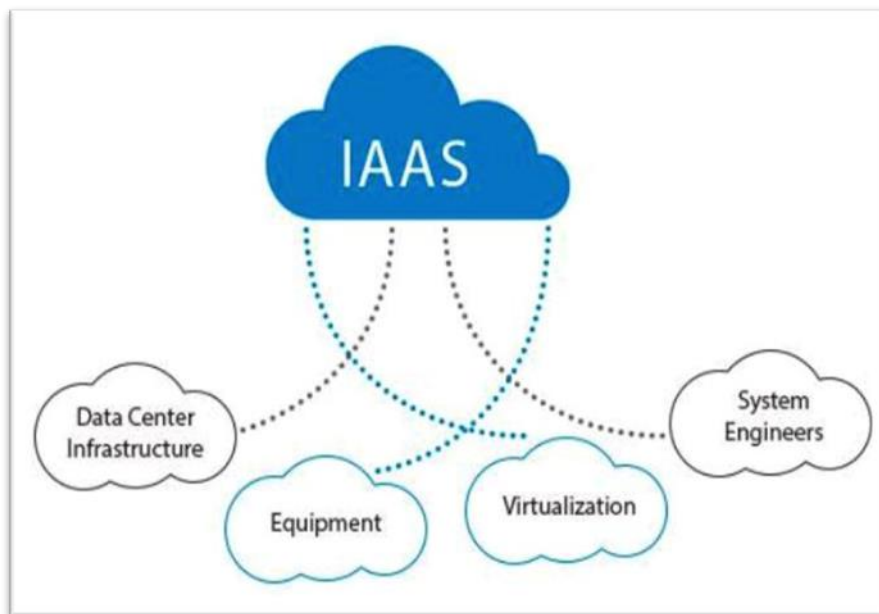
Η πλατφόρμα ως υπηρεσία (αγγλικά: platform as a service , συντομογραφία : PaaS) είναι μια υπηρεσία υπολογιστικού νέφους που παρέχει υπηρεσίες υπολογιστικής πλατφόρμας και λύσεων. Σε ένα τυπικό επίπεδο υπολογιστικού νέφους, το επίπεδο PaaS βρίσκεται μεταξύ λογισμικού ως υπηρεσία και υποδομής ως υπηρεσίας .

Το PaaS παρέχει στους χρήστες να αναπτύξουν και να δημιουργήσουν υποδομή cloud στον πελάτη ή να αποκτήσουν τη χρήση γλωσσών προγραμματισμού , βιβλιοθηκών και υπηρεσιών. Οι χρήστες δεν χρειάζεται να διαχειρίζονται και να ελέγχουν την υποδομή cloud (συμπεριλαμβανομένου του δικτύου, του διακομιστή, του λειτουργικού συστήματος ή της αποθήκευσης), αλλά πρέπει να ελέγχουν την

ανάπτυξη της εφαρμογής ανώτερου επιπέδου και το περιβάλλον φιλοξενίας εφαρμογών.

Το PaaS παίρνει την πλατφόρμα ανάπτυξης λογισμικού ως υπηρεσία και την παραδίδει στους χρήστες σε ένα μοντέλο λογισμικού ως υπηρεσία (SaaS).

Επομένως, το PaaS είναι επίσης μια εφαρμογή του μοντέλου SaaS. Ωστόσο, η εμφάνιση του PaaS μπορεί να επιταχύνει την ανάπτυξη του SaaS, ειδικά την ταχύτητα ανάπτυξης των εφαρμογών SaaS.



Infrastructure as a service (IaaS)

Η υποδομή ως υπηρεσία (Αγγλικά: Infrastructure as a Service , που αναφέρεται ως IaaS) είναι να παρέχει στους καταναλωτές επεξεργασία, αποθήκευση, δίκτυο και διάφορους βασικούς υπολογιστικούς πόρους για την ανάπτυξη και εκτέλεση διαφόρων λογισμικών, όπως λειτουργικά συστήματα ή εφαρμογές.

Το IaaS είναι το κατώτερο επίπεδο των υπηρεσιών cloud και παρέχει κυρίως κάποιους βασικούς πόρους. Η διαφορά μεταξύ αυτού και του PaaS είναι ότι οι χρήστες πρέπει να ελέγχουν το υποκείμενο επίπεδο και να εφαρμόζουν τη λογική χρήσης της υποδομής .

Οι λύσεις IaaS δίνουν την ευελιξία να κλιμακωθούν οι πόροι πληροφορικής πάνω και κάτω με τη ζήτηση. Επίσης, βοηθούν στη παροχή νέων εφαρμογών και στην αύξηση της αξιοπιστίας της υποκείμενης υποδομής.

Desktop as a Service (DaaS)

Το DaaS είναι μια υπηρεσία που βασίζεται σε σύννεφο που επιτρέπει στους χρήστες να έχουν πρόσβαση σε εικονικά περιβάλλοντα επιφάνειας εργασίας μέσω του Διαδικτύου. Το περιβάλλον διαχειρίζεται ένας τρίτος πάροχος υπηρεσιών, παρέχοντας στους χρήστες ένα πλήρες προϊόν.

Το DaaS εξαλείφει την ανάγκη εγκατάστασης ενός συγκεκριμένου λειτουργικού συστήματος στη συσκευή (εκτός από την κάλυψη των απαιτήσεων υλικού) και επίσης επιτρέπει στο περιβάλλον εικονικής επιφάνειας εργασίας να είναι ξεχωριστό από τη συσκευή.

Οι πάροχοι DaaS διαχειρίζονται την υποδομή υλικού και μεταδίδουν εικονικούς επιτραπέζιους υπολογιστές στις συσκευές σας. Σας δίνει πλήρη πρόσβαση στη λειτουργικότητα της επιφάνειας εργασίας σας από εφαρμογές και λειτουργίες έως υποδομές.

Το DaaS λειτουργεί παρόμοια με άλλες υπηρεσίες cloud. Επιτραπέζιοι υπολογιστές, καθώς και λειτουργικά συστήματα που χρησιμοποιούνται, εφαρμογές, λογισμικό και δεδομένα αποθηκεύονται εξ αποστάσεως σε διακομιστές της παρέχουσας εταιρείας.

Οι λύσεις DaaS είναι συνήθως διαθέσιμες με μηνιαίες ή ετήσιες συνδρομές, καθώς και με δυνατότητα αναβάθμισης εύρους ζώνης ρύθμισης και αποθήκευσης. Το κόστος μπορεί να αυξηθεί μακροπρόθεσμα σε σύγκριση με την αγορά εξοπλισμού, αλλά η χρήση του DaaS έχει πολλά οφέλη:

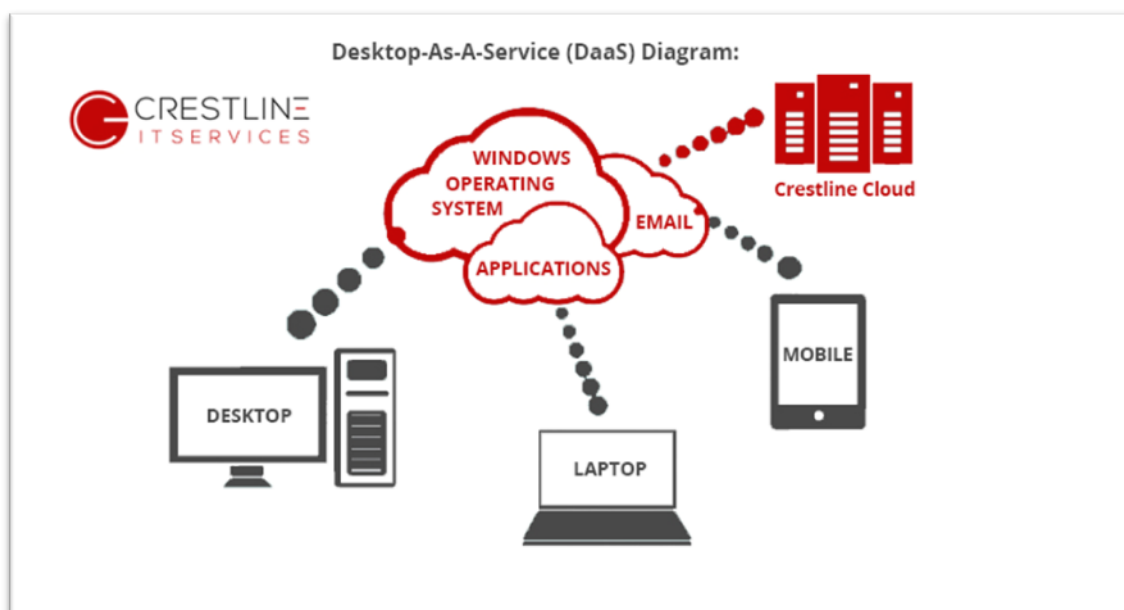
Ασφάλεια : Οι πάροχοι DaaS προστατεύουν το υλικό και τα δεδομένα. Αντί να συνδυάζετε συστήματα ασφαλείας σε επίπεδο χρήστη, μπορείτε να βασιστείτε σε μια ομάδα ειδικών στον τομέα της κυβερνοασφάλειας που είναι υπεύθυνη για τους διακομιστές της εταιρείας σας.

Επεκτασιμότητα : Μπορείτε να αυξάνετε ή να μειώνετε το εύρος ζώνης μηνιαίως ανάλογα με τα έργα στα οποία εργάζεστε και την υπολογιστική ισχύ που χρειάζεστε.

Προσβασιμότητα : Με το DaaS, δεν χρειάζεται να ανησυχείτε για το ότι ταξιδεύετε με έναν ακριβό υπολογιστή ή φορητό υπολογιστή. Αντ' αυτού, μπορείτε να χρησιμοποιήσετε έναν ελαφρύ φορητό υπολογιστή ή οποιαδήποτε άλλη συσκευή είναι διαθέσιμη για να αποκτήσετε πρόσβαση στον διαδικτυακό χώρο εργασίας σας.

Αρχικό κόστος: Η χρήση του DaaS μπορεί να σας κοστίσει περισσότερο μακροπρόθεσμα, αλλά δεν είναι επένδυση. Αυτό το πλεονέκτημα καθιστά το DaaS ιδανικό για φοιτητές με προϋπολογισμό, άτομα που εργάζονται σε βραχυπρόθεσμα, προσωπικά έργα και ελεύθερους επαγγελματίες που θέλουν να ξεκινήσουν αλλά δεν έχουν αρκετά χρήματα για να αγοράσουν έναν ισχυρό υπολογιστή εκ των προτέρων.

Χρόνος διακοπής λειτουργίας: Ανεξάρτητα από τη συντήρηση του εξοπλισμού που χρειάζεστε, είναι βέβαιο ότι θα υπάρξει κάποια απροσδόκητη διακοπή λειτουργίας που θα χρειαστεί χρόνο και χρήμα για να διορθωθεί. Με το DaaS, σπάνια θα υποφέρετε από διακοπές λειτουργίας, εφόσον έχετε αξιόπιστη σύνδεση στο διαδίκτυο και εγγραφείτε σε έναν αξιόπιστο πάροχο DaaS.



Εφαρμογή βασικών τεχνολογιώνεπεξεργασία

Οι βασικές τεχνολογίες υλοποίησης του cloud computing είναι οι εξής:

1. Αρχιτεκτονική

Η υλοποίηση του υπολογιστικού cloud computing χρειάζεται να δημιουργήσει ένα συγκεκριμένο περιβάλλον και συνθήκες, ειδικά η αρχιτεκτονική πρέπει να έχει τα ακόλουθα βασικά χαρακτηριστικά. Πρώτον, το σύστημα πρέπει να είναι έξυπνο, με αυτόνομες δυνατότητες και η αυτοματοποιημένη πλατφόρμα επεξεργασίας θα πρέπει να ανταποκρίνεται έξυπνα στις απαιτήσεις υπό την προϋπόθεση της μείωσης των χειροκίνητων λειτουργιών. Επομένως, το σύστημα cloud θα πρέπει να έχει ενσωματωμένη τεχνολογία αυτοματισμού· δεύτερον, έναντι αλλαγής σημάτων ή σημάτων ζήτησης, το σύστημα cloud πρέπει να υπάρχουν δυνατότητες ευέλικτης απόκρισης, επομένως υπάρχουν ορισμένες ευέλικτες απαιτήσεις για την αρχιτεκτονική του υπολογιστικού νέφους. Ταυτόχρονα, με τις γρήγορες αλλαγές στα επίπεδα υπηρεσιών και τους ρυθμούς ανάπτυξης, το cloud computing αντιμετωπίζει επίσης μεγάλες προκλήσεις και η ενσωματωμένη τεχνολογία ομαδοποίησης και η τεχνολογία εικονικοποίησης μπορούν να αντιμετωπίσουν τέτοιες αλλαγές.

Η αρχιτεκτονική μιας πλατφόρμας υπολογιστικού νέφους αποτελείται από μια διεπαφή χρήστη, κατάλογο υπηρεσιών, σύστημα διαχείρισης, εργαλεία ανάπτυξης, παρακολούθηση και συμπλέγματα διακομιστών:

(1) Διεπαφή χρήστη. Χρησιμοποιείται κυρίως για χρήστες cloud για τη μετάδοση πληροφοριών και αποτελεί διεπαφή για αλληλεπίδραση μεταξύ των δύο μερών.

(2) Κατάλογος υπηρεσιών. Όπως υποδηλώνει το όνομα, παρέχει μια λίστα με τις επιλογές των χρηστών.

(3) Σύστημα διαχείρισης. Αναφέρεται στη διαχείριση πόρων με υψηλή αξία εφαρμογής.

(4) Εργαλεία ανάπτυξης. Οι πόροι μπορούν να αναπτυχθούν αποτελεσματικά και να αντιστοιχιστούν σύμφωνα με τα αιτήματα των χρηστών.

(5) Παρακολούθηση. Διαχειρίζεται και ελέγχει κυρίως τους πόρους στο σύστημα cloud και διαμορφώνει μέτρα.

(6) Σύμπλεγμα διακομιστών. Το σύμπλεγμα διακομιστών περιλαμβάνει εικονικούς διακομιστές και φυσικούς διακομιστές, οι οποίοι ανήκουν στο σύστημα διαχείρισης.

2. Παρακολούθηση πόρων

Τα δεδομένα πόρων στο σύστημα cloud είναι πολύ μεγάλα και οι πληροφορίες πόρων ενημερώνονται γρήγορα. Εάν θέλετε ακριβείς και αξιόπιστες δυναμικές πληροφορίες, χρειάζεστε έναν αποτελεσματικό τρόπο για να διασφαλίσετε την ταχύτητα των πληροφοριών. Το σύστημα cloud μπορεί να αναπτύξει αποτελεσματικά δυναμικές πληροφορίες και έχει επίσης τη λειτουργία παρακολούθησης πόρων, η οποία συμβάλλει στη διαχείριση του φορτίου και της χρήσης των πόρων. Δεύτερον, η παρακολούθηση πόρων, ως το «αίμα» της διαχείρισης πόρων, παίζει βασικό ρόλο στη συνολική απόδοση του συστήματος. Όταν η παρακολούθηση πόρων του συστήματος δεν είναι σε θέση και οι πληροφορίες δεν είναι αξιοπιστίας, άλλα υποσυστήματα θα αναφέρουν λανθασμένες πληροφορίες, οι οποίες θα έχουν αναπόφευκτα αρνητικό αντίκτυπο στην κατανομή των πόρων του συστήματος. Ως εκ τούτου, είναι επιτακτική ανάγκη να εφαρμοστεί η εργασία παρακολούθησης των πόρων. Στη διαδικασία παρακολούθησης πόρων, οι δραστηριότητες διαμόρφωσης και παρακολούθησης μπορούν να εκτελεστούν εφόσον το πρόγραμμα Agent έχει αναπτυχθεί σε κάθε διακομιστή cloud. Για παράδειγμα, κάθε διακομιστής πόρων cloud συνδέεται μέσω ενός διακομιστή παρακολούθησης και, στη συνέχεια, η χρήση των πόρων αποστέλλεται στον βάση δεδομένων σε κυκλική βάση. Ο διακομιστής ενσωματώνει τις αποτελεσματικές πληροφορίες της βάσης δεδομένων για να αναλύσει όλους τους πόρους, να αξιολογήσει τη διαθεσιμότητα των πόρων και να μεγιστοποιήσει την αποτελεσματικότητα των πληροφοριών πόρων.

3. Αυτοματοποιημένη ανάπτυξη

Η ανάπτυξη της επιστημονικής προόδου τείνει να είναι ημιαυτόματη λειτουργία, η οποία πραγματοποιεί εργοστασιακή χρήση ή εύκολη εγκατάσταση και χρήση. Βασικά, αλλάζει και η διαθεσιμότητα υπολογιστικών πόρων, προχωρώντας σταδιακά προς την αυτοματοποιημένη ανάπτυξη. Η αυτόματη ανάπτυξη πόρων cloud αναφέρεται στην αυτόματη διαμόρφωση εξοπλισμού και εργαλείων από

διαφορετικούς κατασκευαστές με βάση την προσαρμογή σεναρίου, προκειμένου να μειωθεί το ποσοστό αλληλεπίδρασης ανθρώπου-υπολογιστή, να βελτιωθεί η αποτελεσματικότητα απόκρισης, να αποφευχθούν οι υπερφορτωμένες χειροκίνητες λειτουργίες και άλλα φαινόμενα και τελικά να προωθηθεί έξυπνη διαδικασία ανάπτυξης. Η αυτόματη ανάπτυξη αναφέρεται κυρίως στην υλοποίηση υπολογιστικών πόρων από την αρχική κατάσταση στη διαθέσιμη κατάσταση μέσω αυτόματης εγκατάστασης και ανάπτυξης. Στην πληροφορική, είναι μια διαδικασία που μπορεί να διαιρέσει, να αναπτύξει και να εγκαταστήσει πόρους σε μια εικονική δεξαμενή πόρων για να παρέχει στους χρήστες διάφορες υπηρεσίες εφαρμογών, συμπεριλαμβανομένης της αποθήκευσης, του δικτύου, του λογισμικού και του υλικού. Υπάρχουν πολλά βήματα ανάπτυξης για τους πόρους του συστήματος. Η αυτόματη ανάπτυξη χρησιμοποιεί κυρίως κλήσεις σεναρίων για αυτόματη διαμόρφωση, ανάπτυξη και διαμόρφωση εργαλείων διαχείρισης εξοπλισμού διαφόρων κατασκευαστών, διασφαλίζοντας ότι η πραγματική κλήση μπορεί να υλοποιηθεί με αθόρυβο τρόπο, αποφεύγοντας την περίπλοκη ανθρώπινη αλληλεπίδραση και επιτρέποντας την ανάπτυξη. Η διαδικασία δεν βασίζεται πλέον σε χειροκίνητες λειτουργίες. Επιπλέον, τα μοντέλα δεδομένων και οι μηχανές ροής εργασίας αποτελούν σημαντικό μέρος των εργαλείων αυτοματοποιημένης διαχείρισης ανάπτυξης και δεν πρέπει να υποτιμώνται. Γενικά, η διαχείριση του μοντέλου δεδομένων συνίσταται στον καθορισμό του συγκεκριμένου λογισμικού και υλικού στο μοντέλο δεδομένων και ο κινητήρας ροής εργασιών αναφέρεται στην ενεργοποίηση και στην κλήση της ροής εργασίας, με στόχο τη βελτίωση της έξυπνης ανάπτυξης και είναι καλός στην ενσωμάτωση διαφορετικών σεναρίων. εφαρμόζεται στη βάση δεδομένων ροής εργασιών με υψηλή συγκέντρωση και υψηλό ρυθμό επανάληψης, γεγονός που είναι ευεργετικό για τη μείωση του φόρτου εργασίας του διακομιστή.

Απειλή για την ασφάλεια

1. Το απόρρητο κλέβεται στην ασφάλεια του cloud computing

Σήμερα, με την εξέλιξη των καιρών, οι άνθρωποι χρησιμοποιούν το Διαδίκτυο για να πραγματοποιούν συναλλαγές ή αγορές. Οι ηλεκτρονικές συναλλαγές πραγματοποιούνται στο εικονικό περιβάλλον του cloud computing και τα δύο μέρη της συναλλαγής θα επικοινωνούν και θα ανταλλάσσουν πληροφορίες στην πλατφόρμα δικτύου. Ωστόσο, υπάρχουν μεγάλοι κίνδυνοι ασφάλειας στις ηλεκτρονικές συναλλαγές. Οι εγκληματίες μπορούν να κλέψουν τις πληροφορίες των χρηστών του δικτύου μέσω υπολογιστικού νέφους. Ταυτόχρονα, μπορούν επίσης να κλέψουν τις πληροφορίες των χρηστών και των εμπόρων όταν οι χρήστες πραγματοποιούν συναλλαγές στο διαδίκτυο με εμπόρους. Μετά την κλοπή πληροφοριών σε στην πλατφόρμα υπολογιστικού νέφους, θα χρησιμοποιηθούν κάποια τεχνικά μέσα για τη διάρρηξη των πληροφοριών και ταυτόχρονα, οι πληροφορίες θα αναλυθούν για να βρεθούν περισσότερες ιδιωτικές πληροφορίες των χρηστών, ακόμη και εγκληματίες που έχουν προθέσεις θα τις κλέψουν μέσω του υπολογιστικού νέφους. πληροφορίες χρήστη και επιχείρησης. [3]

2. Οι πόροι χρησιμοποιούνται με δόλο στο cloud computing

Το περιβάλλον του cloud computing έχει εικονικά χαρακτηριστικά και όταν οι χρήστες πραγματοποιούν συναλλαγές δικτύου μέσω υπολογιστικού νέφους, πρέπει να διασφαλίζουν την ασφάλεια των πληροφοριών δικτύου και των δύο μερών πριν από την εκτέλεση εργασιών δικτύου. Ωστόσο, υπάρχουν πολλές πληροφορίες αποθηκευμένες στο cloud computing. Το περιβάλλον είναι επίσης σχετικά πολύπλοκο και τα δεδομένα στο cloud computing θα γίνει κατάχρηση, η οποία θα επηρεάσει την ασφάλεια των πληροφοριών του χρήστη και ταυτόχρονα θα κάνει ορισμένους εγκληματίες να χρησιμοποιήσουν τις κλεμμένες πληροφορίες για να εξαπατήσουν τους συγγενείς των χρηστών. οι εγκληματίες θα χρησιμοποιήσουν αυτές τις κλεμμένες πληροφορίες στο cloud computing για τη διεξαγωγή παράνομων συναλλαγών, προκαλώντας έτσι οικονομικές απώλειες στους χρήστες στο cloud computing.

3. Οι χάκερ επιθέσεις χάκερ

Η πειρατεία αναφέρεται στη χρήση κάποιων παράνομων μέσων για την είσοδο στο σύστημα ασφαλείας του υπολογιστικού νέφους και την πρόκληση βλαβών στο

δίκτυο ασφαλείας του υπολογιστικού νέφους. Το cloud computing είναι μεγαλύτερο από αυτό που προκαλείται από ιούς. Επιπλέον, η ταχύτητα του hacking είναι πολύ μεγαλύτερη από την ταχύτητα της αξιολόγησης ασφάλειας και της ενημέρωσης του συστήματος ασφαλείας, γεγονός που κάνει τους σημερινούς χάκερ να εισβάλλουν στους υπολογιστές, προκαλώντας τεράστιες απώλειες στο cloud computing και η τεχνολογία δεν μπορεί να αποτρέψει επιθέσεις hacking. Ένα από τα επικίνδυνα προβλήματα.

4. Οι ιοί είναι επιρρεπείς στην εμφάνιση στο cloud computing

Στο cloud computing, ένας μεγάλος αριθμός χρηστών αποθηκεύει δεδομένα σε αυτό μέσω του cloud computing. Αυτή τη στιγμή, όταν ένας μεγάλος αριθμός cloud computing είναι μη φυσιολογικός, θα εμφανιστούν ορισμένοι ιοί. Η εμφάνιση αυτών των ιών θα προκαλέσει τον υπολογιστή που βασίζεται στο cloud computing να αποτυγχάνουν να λειτουργήσουν κανονικά. Ταυτόχρονα, αυτοί οι ιοί μπορούν επίσης να αναπαραχθούν και να εξαπλωθούν μέσω ορισμένων καναλιών, γεγονός που θα προκαλέσει τη συντριβή του υπολογιστή που χρησιμοποιεί το cloud computing ως φορέα. Θα εξαπλωθεί πολύ γρήγορα, γεγονός που θα δημιουργήσει μεγάλη ισχύ επίθεσης.

Εφαρμογή cloud computing

Η σχετικά απλή τεχνολογία υπολογιστικού νέφους έχει γενικά χρησιμοποιηθεί στις σημερινές υπηρεσίες Διαδικτύου, οι πιο συνηθισμένες είναι οι μηχανές αναζήτησης στο Διαδίκτυο και τα γραμματοκιβώτια Διαδικτύου. Οι πιο γνωστές μηχανές αναζήτησης είναι η Google και η Baidu. Ανά πάσα στιγμή, εφόσον χρησιμοποιείτε ένα τερματικό κινητής τηλεφωνίας, μπορείτε να αναζητήσετε οποιουδήποτε πόρους θέλετε στη μηχανή αναζήτησης και να μοιραστείτε πόρους δεδομένων μέσω του cloud. Το ίδιο ισχύει και για τα ηλεκτρονικά γραμματοκιβώτια. Στο παρελθόν, η αποστολή ενός email ήταν μια ενοχλητική και αργή διαδικασία. Με την προώθηση της τεχνολογίας υπολογιστικού νέφους και της τεχνολογίας δικτύου, το email έχει γίνει μέρος της κοινωνικής ζωής. , όσο στο περιβάλλον του δικτύου , μπορείτε να επιτύχετε παράδοση αλληλογραφίας σε πραγματικό χρόνο. Στην

πραγματικότητα, η τεχνολογία cloud computing έχει ενσωματωθεί στη σημερινή κοινωνική ζωή. [1]

1. Σύννεφο αποθήκευσης

Το Storage cloud, γνωστό και ως cloud storage, είναι μια νέα τεχνολογία αποθήκευσης που αναπτύχθηκε στην τεχνολογία cloud computing. Το cloud storage είναι ένα σύστημα υπολογιστικού νέφους με πυρήνα την αποθήκευση και διαχείριση δεδομένων. Οι χρήστες μπορούν να ανεβάσουν τοπικούς πόρους στο cloud και μπορούν να συνδεθούν στο Διαδίκτυο οπουδήποτε για να αποκτήσουν πόρους στο cloud. Γνωστές μεγάλες εταιρείες δικτύου όπως η Google και η Microsoft διαθέτουν υπηρεσίες αποθήκευσης cloud. Στην Κίνα, το Baidu Cloud και το Weiyun είναι τα clouds αποθήκευσης με το μεγαλύτερο μερίδιο αγοράς. Το cloud αποθήκευσης παρέχει στους χρήστες υπηρεσίες κοντέινερ αποθήκευσης, υπηρεσίες δημιουργίας αντιγράφων ασφαλείας, υπηρεσίες αρχειοθέτησης και υπηρεσίες διαχείρισης αρχείων, γεγονός που διευκολύνει σημαντικά τη διαχείριση των πόρων για τους χρήστες.

2. Ιατρικό σύννεφο

Το ιατρικό νέφος αναφέρεται στη χρήση του "υπολογιστικού νέφους" για τη δημιουργία μιας πλατφόρμας cloud για ιατρικές υπηρεσίες και υπηρεσίες υγείας που βασίζεται σε νέες τεχνολογίες όπως το cloud computing, η κινητή τεχνολογία, τα πολυμέσα, η επικοινωνία 4G, τα μεγάλα δεδομένα και το Διαδίκτυο των πραγμάτων, συνδυάζοντας ιατρική τεχνολογία και αξιοποίηση ιατρικών πόρων, κοινή χρήση και επέκταση της ιατρικής κάλυψης. Λόγω του συνδυασμού της τεχνολογίας υπολογιστικού νέφους, το ιατρικό νέφος βελτιώνει την αποτελεσματικότητα των ιατρικών ιδρυμάτων και διευκολύνει τους κατοίκους να αναζητήσουν ιατρική περίθαλψη. Για παράδειγμα, η εγγραφή ραντεβού στο νοσοκομείο, τα ηλεκτρονικά ιατρικά αρχεία, η ιατρική ασφάλιση κ.λπ. είναι όλα προϊόντα του συνδυασμού του cloud computing και του ιατρικού τομέα. Το ιατρικό cloud έχει επίσης τα πλεονεκτήματα της ασφάλειας δεδομένων, της κοινής χρήσης πληροφοριών, της δυναμικής επέκτασης και του εθνικού διάταξη.

3. Οικονομικό σύννεφο

Το χρηματοοικονομικό σύννεφο αναφέρεται στη χρήση του μοντέλου υπολογιστικού νέφους για τη διασπορά πληροφοριών, οικονομικών και υπηρεσιών στο «σύννεφο» του Διαδικτύου που αποτελείται από τεράστια υποκαταστήματα, με στόχο την παροχή υπηρεσιών επεξεργασίας και λειτουργίας Διαδικτύου για χρηματοπιστωτικά ιδρύματα όπως τράπεζες, ασφάλειες και ταμεία. Ταυτόχρονα, οι πόροι του Διαδικτύου μοιράζονται, ώστε να λυθούν τα υπάρχοντα προβλήματα και να επιτευχθεί ο στόχος της υψηλής απόδοσης και του χαμηλού κόστους. Στις 27 Νοεμβρίου 2013, το Alibaba Cloud ενσωμάτωσε τους πόρους της Alibaba και ξεκίνησε την υπηρεσία Alibaba Financial Cloud. Στην πραγματικότητα, αυτή είναι η γρήγορη πληρωμή που είναι βασικά δημοφιλής τώρα. Λόγω του συνδυασμού χρηματοδότησης και υπολογιστικού νέφους, μπορείτε πλέον να ολοκληρώσετε τραπεζικές καταθέσεις, να αγοράσετε ασφάλειες και συναλλαγές χρηματοδότησης με απλές λειτουργίες στο κινητό σας τηλέφωνο. Τώρα, όχι μόνο η Alibaba έχει λανσάρει χρηματοοικονομικές υπηρεσίες cloud, αλλά εταιρείες όπως η Suning Finance και η Tencent έχουν ξεκινήσει τις δικές τους υπηρεσίες χρηματοοικονομικού cloud.

4. Εκπαίδευση Cloud

Το εκπαιδευτικό νέφος, στην ουσία, αναφέρεται στην ανάπτυξη της πληροφορικής της εκπαίδευσης. Συγκεκριμένα, το εκπαιδευτικό νέφος μπορεί να εικονικοποιήσει τυχόν απαιτούμενους πόρους εκπαιδευτικού υλικού και στη συνέχεια να τους μεταδώσει στο Διαδίκτυο για να παρέχει μια βολική και γρήγορη πλατφόρμα για εκπαιδευτικά ιδρύματα και μαθητές και δασκάλους. Τώρα το δημοφιλές MOOC είναι μια εφαρμογή του εκπαιδευτικού cloud. Τα MOOC αναφέρονται σε μαζικά ανοιχτά διαδικτυακά μαθήματα. Σε αυτό το στάδιο, οι τρεις εξαιρετικές πλατφόρμες για τα MOOC είναι οι Coursera, edX και Udacity. Στην Κίνα, τα MOOC των κινεζικών πανεπιστημίων είναι επίσης πολύ καλές πλατφόρμες. Στις 10 Οκτωβρίου 2013, το Πανεπιστήμιο Tsinghua παρουσίασε την πλατφόρμα MOOC, XuetangX, και πολλά πανεπιστήμια έχουν πλέον χρησιμοποιήσει το XuetangX για να προσφέρουν MOOC για ορισμένα μαθήματα.

Αναπτυξιακά θέματα

1. Ζητήματα άδειας πρόσβασης

Οι χρήστες μπορούν να ανεβάσουν τα δικά τους δεδομένα στον πάροχο υπηρεσιών cloud computing. Σε σύγκριση με την παραδοσιακή μέθοδο αποθήκευσης χρησιμοποιώντας τον δικό τους υπολογιστή ή σκληρό δίσκο, πρέπει να δημιουργήσουν λογαριασμό και κωδικό πρόσβασης για να ολοκληρώσουν την αποθήκευση και την απόκτηση εικονικών πληροφοριών. Αν και αυτή η μέθοδος παρέχει ευκολία στους χρήστες να αποκτούν και να αποθηκεύουν πόρους πληροφοριών, οι χρήστες χάνουν τον έλεγχο των πόρων δεδομένων και οι πάροχοι υπηρεσιών ενδέχεται να έχουν μη εξουσιοδοτημένη πρόσβαση σε πόρους, γεγονός που καθιστά δύσκολη την εγγύηση της ασφάλειας του πληροφοριακού υλικού. [7]

2. Θέματα τεχνικού απορρήτου

Η εμπιστευτικότητα των πληροφοριών είναι το πρωταρχικό πρόβλημα της τεχνολογίας υπολογιστικού νέφους, και είναι επίσης το κύριο πρόβλημα της τρέχουσας τεχνολογίας υπολογιστικού νέφους. Για παράδειγμα, ορισμένες επιχειρήσεις μοιράζονται τους πόρους του χρήστη. Η ιδιαιτερότητα του περιβάλλοντος δικτύου επιτρέπει στους ανθρώπους να περιηγούνται ελεύθερα σε σχετικούς πόρους μισθών και η διαρροή πόρων πληροφοριών είναι αναπόφευκτη. Εάν το τεχνικό απόρρητο είναι ανεπαρκές, μπορεί να επηρεάσει σοβαρά τους κατόχους των πόρων πληροφοριών.

3. Ζητήματα ακεραιότητας δεδομένων

Κατά τη χρήση της τεχνολογίας υπολογιστικού νέφους, τα δεδομένα του χρήστη αποθηκεύονται σε διαφορετικές τοποθεσίες στο κέντρο δεδομένων υπολογιστικού νέφους, αντί σε ένα ενιαίο σύστημα, επηρεάζεται η ακεραιότητα των πόρων δεδομένων, καθιστώντας δύσκολη την αποτελεσματική αναπαραγωγή του ρόλου τους. Μια άλλη κατάσταση είναι ότι ο πάροχος υπηρεσιών δεν διαχειρίζεται σωστά και αποτελεσματικά τις πληροφορίες δεδομένων του χρήστη, επηρεάζοντας έτσι την ακεραιότητα της αποθήκευσης δεδομένων και καθιστώντας δύσκολη την αναπαραγωγή του ρόλου εφαρμογής των πληροφοριών.

4. Ελλιπείς νόμοι και κανονισμοί

Η ατέλεια των νόμων και των κανονισμών που σχετίζονται με την τεχνολογία cloud computing είναι επίσης ένα σημαντικό πρόβλημα. Για να παίξει αποτελεσματικά το ρόλο της τεχνολογίας cloud computing, είναι απαραίτητο να βελτιωθούν οι σχετικοί νόμοι και κανονισμοί. Προς το παρόν, οι νόμοι και οι κανονισμοί δεν είναι τέλειοι και ο ρόλος της τεχνολογίας υπολογιστικού νέφους εξακολουθεί να είναι περιορισμένος. Όσον αφορά την τρέχουσα εφαρμογή της τεχνολογίας υπολογιστικού νέφους στα δίκτυα υπολογιστών, δεν διαθέτει τέλεια πρότυπα ασφαλείας, στερούνται τέλειων προτύπων διαχείρισης συμφωνιών σε επίπεδο υπηρεσιών και δεν υπάρχει ξεκάθαρο υπεύθυνο άτομο για να αναλάβει τη νομική ευθύνη για ζητήματα ασφάλειας. Επιπλέον, η έλλειψη ενός ολοκληρωμένου μηχανισμού υπολογισμού ζημιών και μηχανισμού αξιολόγησης ευθυνών για τη διαχείριση της ασφάλειας του υπολογιστικού νέφους και η έλλειψη νομικών κανόνων περιορίζουν επίσης την ανάπτυξη διαφόρων δραστηριοτήτων και δεν είναι εγγυημένη η ασφάλεια του υπολογιστικού νέφους των δικτύων υπολογιστών.

Μέτρα βελτίωσης

1. Ορίστε εύλογα δικαιώματα πρόσβασης για να διασφαλίσετε την ασφάλεια των πληροφοριών χρήστη

Προς το παρόν, οι υπηρεσίες υπολογιστών cloud παρέχονται από προμηθευτές. Για τη διασφάλιση της ασφάλειας των πληροφοριών, οι προμηθευτές θα πρέπει να ορίσουν αντίστοιχα δικαιώματα πρόσβασης σύμφωνα με τις ανάγκες των χρηστών, ώστε να διασφαλίζεται η ασφαλής κοινή χρήση των πόρων πληροφοριών. Σε ένα ανοιχτό περιβάλλον Διαδικτύου, αφενός, οι προμηθευτές πρέπει να κάνουν καλή δουλειά στον καθορισμό δικαιωμάτων πρόσβασης και να ενισχύσουν την εύλογη κοινή χρήση και εφαρμογή των πόρων· από την άλλη, πρέπει να κάνουν καλή δουλειά στην κρυπτογράφηση, και οι προμηθευτές και οι χρήστες θα πρέπει να ενισχύσει την ασφάλεια των πληροφοριών Προστασία, να δώσει προσοχή στην κατασκευή της ασφάλειας του δικτύου και να προστατεύει αποτελεσματικά την ασφάλεια των χρηστών. Ως εκ τούτου, η ανάπτυξη της τεχνολογίας υπολογιστών

cloud θα πρέπει να ενισχύσει την κατασκευή του συστήματος τεχνολογίας ασφάλειας και να βελτιώσει το επίπεδο προστασίας των πληροφοριών στο εύλογο πλαίσιο δικαιωμάτων πρόσβασης. [8]

2. Ενίσχυση της ακεραιότητας των πληροφοριών δεδομένων και προώθηση της ανάπτυξης τεχνολογίας αποθήκευσης

Η τεχνολογία αποθήκευσης είναι ο πυρήνας της τεχνολογίας υπολογιστικού νέφους. Ο τρόπος ενίσχυσης της ακεραιότητας των πληροφοριών δεδομένων είναι μια σημαντική πτυχή της ανάπτυξης της τεχνολογίας υπολογιστικού νέφους. Πρώτον, οι πόροι υπολογιστικού νέφους διανέμονται στο σύστημα νέφους με διακριτό τρόπο. Είναι απαραίτητο να ενισχυθεί η προστασία ασφαλείας των πόρων δεδομένων στο σύστημα cloud και να διασφαλιστεί η ακεραιότητα των δεδομένων, γεγονός που συμβάλλει στη βελτίωση της αξίας εφαρμογής των πόρων πληροφοριών· δεύτερον, για να επιταχυνθεί η αποθήκευση Η τεχνολογική ανάπτυξη, ειδικά στην εποχή των μεγάλων δεδομένων, η ανάπτυξη της τεχνολογίας υπολογιστικού νέφους θα πρέπει να επικεντρωθεί στην καινοτόμο κατασκευή της τεχνολογίας αποθήκευσης· τρίτον, είναι απαραίτητο να βελτιστοποιηθεί το περιβάλλον ανάπτυξης της τεχνολογίας cloud δικτύων υπολογιστών και περαιτέρω προσαρμοστούν στο νέο περιβάλλον ανάπτυξης μέσω της τεχνολογικής καινοτομίας και της εννοιολογικής καινοτομίας, βελτίωση Η αξία εφαρμογής της τεχνολογίας είναι το επίκεντρο ανάπτυξης της τεχνολογίας υπολογιστών cloud δικτύου υπολογιστών στη νέα εποχή. [8]

3. Θέσπιση και βελτίωση νόμων και κανονισμών για τη βελτίωση της ευαισθητοποίησης για την ασφάλεια των χρηστών

Με τη συνεχή ανάπτυξη της τεχνολογίας πληροφοριών δικτύου, τα πεδία των εφαρμογών υπολογιστικού νέφους γίνονται ολοένα και πιο εκτεταμένα. Η θέσπιση τέλειων νόμων και κανονισμών έχει σκοπό να ρυθμίσει καλύτερα την ανάπτυξη της αγοράς, να ενισχύσει τη ρύθμιση και τη διαχείριση της συμπεριφοράς των προμηθευτών και των χρηστών και να προσφέρει καλές συνθήκες για την ανάπτυξη της τεχνολογίας υπολογιστικού νέφους δικτύων υπολογιστών. Επιπλέον, το τερματικό χρήστη θα πρέπει να βελτιώνει την επίγνωση της προστασίας της ασφάλειας και να μπορεί να συμμορφώνεται με νόμους και κανονισμούς και να

τυποποιεί τις λειτουργίες στην απόκτηση πόρων πληροφοριών, έτσι ώστε να αποφεύγονται σοβαρές οικονομικές απώλειες που προκαλούνται από προβλήματα ασφάλειας πληροφοριών. Επομένως, η ανάπτυξη της τεχνολογίας cloud computing δικτύων υπολογιστών στη νέα εποχή πρέπει να προέλθει από την πραγματικότητα και να παρέχει ένα καλό περιβάλλον για την ανάπτυξη της τεχνολογίας υπολογιστικού νέφους μέσω της συνεχούς βελτίωσης των νόμων και των κανονισμών

Αρχιτεκτονική του Υπολογιστικού Νέφους

Το cloud computing είναι μια άλλη επαναστατική αλλαγή στην τεχνολογία της πληροφορίας μετά τη μεγάλη μετάβαση από τους mainframe υπολογιστές στα μοντέλα πελάτη/διακομιστή (C/S) τη δεκαετία του 1980. Στις 9 Αυγούστου 2006, ο Διευθύνων Σύμβουλος της Google, Eric Schmidt, πρότεινε για πρώτη φορά την έννοια του cloud computing στο συνέδριο της μηχανής αναζήτησης (SES San Jose 2006). Το cloud computing είναι το προϊόν της ενοποίησης παραδοσιακών τεχνολογιών υπολογιστών και δικτύων, όπως υπολογισμός πλέγματος, καταναμημένος υπολογισμός, παράλληλος υπολογισμός, τεχνολογία χρησιμότητας, αποθήκευση δικτύου, εικονικοποίηση και εξισορρόπηση φορτίου. Σκοπός του είναι να οργανώνει και να ενσωματώνει κοινόχρηστους πόρους και πληροφορίες λογισμικού/υλισμικού μέσω υπολογιστών που βασίζονται σε δίκτυο και να τις παρέχει σε υπολογιστές και άλλα συστήματα κατά παραγγελία.

Το αρχιτεκτονικό στρώμα του υπολογιστικού νέφουςεπεξεργασία

Επίπεδο οθόνης

Αυτό το επίπεδο των περισσότερων αρχιτεκτονικών υπολογιστικού νέφους κέντρων δεδομένων χρησιμοποιείται κυρίως για την παρουσίαση του περιεχομένου και της εμπειρίας υπηρεσιών που απαιτούνται από τους χρήστες με φιλικό τρόπο και θα χρησιμοποιεί μια ποικιλία υπηρεσιών που παρέχονται από τα ακόλουθα επίπεδα ενδιάμεσου λογισμικού, που περιλαμβάνουν κυρίως πέντε τεχνολογίες:

HTML: Η τυπική τεχνολογία ιστοσελίδων, κυρίως HTML4 , αλλά η επερχόμενη HTML5 θα προωθήσει την ανάπτυξη ιστοσελίδων σε πολλές πτυχές, όπως βίντεο [και τοπική αποθήκευση κ.λπ.

JavaScript: Μια δυναμική Μέσω JavaScript, οι λειτουργίες των ιστοσελίδων μπορούν να εμπλουτιστούν σημαντικά και να δημιουργηθούν πιο διαδραστικές δυναμικές .

CSS: χρησιμοποιείται κυρίως για τον έλεγχο της εμφάνισης των ιστοσελίδων και μπορεί να κάνει το περιεχόμενο της σελίδας κομψά ξεχωριστό από την παρουσίασή της.

Flash Η πιο συχνά χρησιμοποιούμενη τεχνολογία RIA (Rich Internet Applications) στον κλάδο μπορεί να παρέχει πλούσιες εφαρμογές που βασίζονται στο Web που δεν μπορούν να παρέχονται από HTML και άλλες τεχνολογίες σε αυτό το στάδιο, και είναι πολύ καλή όσον αφορά την εμπειρία χρήστη.

Silverlight: από την τεχνολογία RIA του γίγαντα της βιομηχανίας Microsoft , αν και το μερίδιο αγοράς της είναι ελαφρώς κατώτερο από το Flash, είναι πολύ φιλικό προς τους προγραμματιστές επειδή μπορεί να προγραμματιστεί χρησιμοποιώντας C#

Μεσαίο στρώμα

Αυτό το επίπεδο είναι ο σύνδεσμος μεταξύ του προηγούμενου και του επόμενου. Παρέχει μια ποικιλία υπηρεσιών που βασίζονται στους πόρους που παρέχονται από το υποκείμενο επίπεδο υποδομής, όπως υπηρεσίες cache και υπηρεσίες REST, και αυτές οι υπηρεσίες μπορούν να χρησιμοποιηθούν για την υποστήριξη του επιπέδου εμφάνισης ή απευθείας επιτρέπουν στους χρήστες να καλούν και υπάρχουν κυρίως πέντε τεχνικές:

REST: Μέσω της τεχνολογίας REST, είναι πολύ βολικό και κομψό να παρέχετε ορισμένες υπηρεσίες που υποστηρίζονται από το επίπεδο του ενδιάμεσου λογισμικού στον καλούντα.

Multi-tenancy: δηλαδή, μια μεμονωμένη παρουσία εφαρμογής μπορεί να εξυπηρετήσει πολλούς οργανισμούς και να διατηρήσει καλή απομόνωση και

ασφάλεια, και μέσω αυτής της τεχνολογίας, μπορεί να μειώσει αποτελεσματικά το κόστος απόκτησης και συντήρησης της εφαρμογής.

Παράλληλη επεξεργασία : Για την επεξεργασία τεράστιων δεδομένων, πρέπει να χρησιμοποιηθούν τεράστια συμπλέγματα X86 για μαζική παράλληλη επεξεργασία. Το MapReduce της Google είναι ένα αντιπροσωπευτικό έργο από αυτή την άποψη.

Διακομιστής εφαρμογών : Με βάση τον αρχικό διακομιστή εφαρμογών, έχει γίνει ένας ορισμένος βαθμός βελτιστοποίησης για το cloud computing, όπως ο διακομιστής εφαρμογής Jetty για το Google App Engine.

Κατανεμημένη κρυφή μνήμη: Μέσω της τεχνολογίας κατανεμημένης κρυφής μνήμης, όχι μόνο μπορεί να μειώσει αποτελεσματικά την πίεση στον διακομιστή παρασκήνιου, αλλά και να επιταχύνει την αντίστοιχη ταχύτητα απόκρισης. Το πιο διάσημο παράδειγμα κατανεμημένης κρυφής μνήμης είναι το Memcached.

Στρώμα υποδομής

Η λειτουργία αυτού του επιπέδου είναι να προετοιμάζει τους υπολογιστικούς πόρους και τους πόρους αποθήκευσης που απαιτούνται από το επίπεδο του ενδιαμέσου λογισμικού ή τον χρήστη. Υπάρχουν τέσσερις κύριες τεχνολογίες:

Εικονικοποίηση: Μπορεί επίσης να γίνει κατανοητό ως "πολυ-μίσθωση" του επιπέδου υποδομής, επειδή μέσω της τεχνολογίας εικονικοποίησης, μπορούν να δημιουργηθούν πολλαπλές εικονικές μηχανές σε έναν φυσικό διακομιστή και μπορεί να επιτευχθεί πλήρης απομόνωση μεταξύ αυτών των εικονικών μηχανών, έτσι ώστε να μην μπορεί μειώνουν μόνο το κόστος αγοράς του διακομιστή, αλλά και ταυτόχρονα μειώνουν το κόστος λειτουργίας και συντήρησης του διακομιστή. Οι ώριμες τεχνολογίες εικονικοποίησης X86 περιλαμβάνουν το ESX της VMware και το Xen ανοιχτού κώδικα.

Κατανεμημένη αποθήκευση: Για τη μεταφορά τεράστιων ποσοτήτων δεδομένων και ταυτόχρονα τη διασφάλιση της δυνατότητας διαχείρισης αυτών των δεδομένων, απαιτείται ένα σύνολο κατανεμημένων συστημάτων αποθήκευσης .

Σχεσιακή βάση δεδομένων : Είναι βασικά βελτιστοποιημένη για επέκταση και διαχείριση με βάση την αρχική σχεσιακή βάση δεδομένων, καθιστώντας την πιο προσαρμόσιμη στο cloud.

NoSQL: Προκειμένου να επιτευχθούν οι στόχοι που ορισμένες σχεσιακές βάσεις δεδομένων δεν μπορούν να επιτύχουν, όπως η υποστήριξη μαζικών δεδομένων, ορισμένες εταιρείες σχεδιάζουν ειδικά μια παρτίδα βάσεων δεδομένων που δεν βασίζονται στο σχεσιακό μοντέλο.

Διαχείριση

Αυτό το επίπεδο εξυπηρετεί τα οριζόντια τρία επίπεδα και παρέχει μια ποικιλία τεχνολογιών διαχείρισης και συντήρησης για αυτά τα τρία επίπεδα, κυρίως στις ακόλουθες έξι πτυχές.

Διαχείριση λογαριασμού : Μέσω της καλής τεχνολογίας διαχείρισης λογαριασμών, οι χρήστες μπορούν να συνδεθούν άνετα υπό ασφαλείς συνθήκες και είναι βολικό για τους διαχειριστές να διαχειρίζονται λογαριασμούς.

Παρακολούθηση SLA: Παρακολούθηση της απόδοσης εικονικών μηχανών, υπηρεσιών και εφαρμογών που εκτελούνται σε όλα τα επίπεδα, έτσι ώστε να μπορούν όλες να εκτελούνται σύμφωνα με το προκαθορισμένο SLA (Service Level Agreement).

Διαχείριση χρέωσης: δηλαδή η συλλογή στατιστικών στοιχείων σχετικά με τους πόρους που καταναλώνει κάθε χρήστης για την ακριβή χρέωση του χρήστη.

Διαχείριση ασφάλειας: Προστατέψτε πλήρως τους πόρους πληροφορικής, όπως δεδομένα, εφαρμογές και λογαριασμούς από εγκληματίες και κακόβουλα προγράμματα.

Εξισορρόπηση φορτίου : Αντιμετωπίστε απροσδόκητες καταστάσεις κατανέμοντας την κυκλοφορία σε πολλαπλές παρουσίες μιας εφαρμογής ή υπηρεσίας.

Διαχείριση λειτουργίας και συντήρησης : Ο κύριος σκοπός είναι να γίνουν οι λειτουργίες λειτουργίας και συντήρησης όσο το δυνατόν πιο επαγγελματικές και

αυτόματες, μειώνοντας έτσι το κόστος λειτουργίας και συντήρησης του κέντρου υπολογιστών νέφους.

Τρία επίπεδα αρχιτεκτονικής υπολογιστικού νέφους είναι οριζόντια, δηλαδή το επίπεδο οθόνης, το επίπεδο μέσου λογισμικού και το επίπεδο υποδομής. Αυτά τα τρία επίπεδα τεχνολογίας μπορούν να παρέχουν πολύ πλούσιες δυνατότητες υπολογιστικού νέφους και μια φιλική διεπαφή χρήστη. Υπάρχει επίσης ένα κατακόρυφο επίπεδο αρχιτεκτονικής υπολογιστικού νέφους, το επίπεδο διαχείρισης, υπάρχει για την καλύτερη διαχείριση και διατήρηση των οριζόντιων τριών επιπέδων

Ενδοδίκτυο

Πρόκειται για μια μορφή ενδοδικτύου (intranet). Το intranet είναι ένα εσωτερικό, ιδιωτικό δίκτυο που περιγράφει το σύνολο των ιδιωτικών δικτύων των υπολογιστών μέσα σε έναν οργανισμό ή σε μια εταιρεία.

Ένα ενδοδίκτυο χρησιμοποιεί το ίδιο πρωτόκολλο επικοινωνίας με το internet, αλλά περιορίζει την πρόσβαση σε ένα συνδεδεμένο δίκτυο και σε συγκεκριμένες ομάδες ατόμων.

Κάποιες φορές τα εταιρικά ενδοδίκτυα προσφέρουν πρόσβαση στο ενδοδίκτυο αλλά είναι προστατευμένα με firewall έτσι ώστε οι υπολογιστές του ενδοδικτύου να μην είναι προσβάσιμοι απευθείας από το δημόσιο δίκτυο.

Τα ενδοδίκτυα επιτρέπουν σε εταιρείες και οργανισμούς να διατηρούν μια επίσημη εκδοχή όλων των πληροφοριών, που σχετίζονται με την εταιρεία, άμεσα διαθέσιμη στο προσωπικό χωρίς να χρειάζεται να διατηρούνται έντυπα έγγραφα.

Επίσης, δίνει τη δυνατότητα επεξεργασίας ή αλλαγής των δεδομένων, όπως μια λίστα τηλεφωνικών πελατών, σε ένα κεντρικό σημείο, απ' όπου και οι υπόλοιποι υπάλληλοι μπορούν να λάβουν την ενημερωμένη λίστα.

Δημόσιο σύννεφο (Public Cloud)

Ένα σύννεφο ονομάζεται δημόσιο όταν προσφέρονται εφαρμογές και υπηρεσίες στο ευρύ κοινό για χρήση.

Κάποιες από αυτές είναι: Amazon Web Services, Google App Engine, Microsoft Windows Azure κ.α.

Σύννεφο κοινότητας (Community Cloud)

Πρόκειται για το σύννεφο όπου έχουμε διαμοιρασμό υποδομής μεταξύ οργανισμών με κοινή στρατηγική.

Η συγκεκριμένη cloud υποδομή μοιράζεται μεταξύ πολλών οργανισμών και υποστηρίζει μια συγκεκριμένη ενότητα που την απασχολούν κοινά θέματα (π.χ αποστολή, απαιτήσεις ασφαλείας, πολιτική και θέματα συμμόρφωσης).

Η διαδικασία διαχείρισης της συγκεκριμένης υποδομής cloud, μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί αυτό με τη σειρά του να βρίσκεται εντός ή εκτός ορίων του οργανισμού.

Υβριδικό σύννεφο (Hybrid Cloud)

Μια σύνθεση δυο ή και περισσότερων μοντέλων ανάπτυξης που παραμένουν ξεχωριστές οντότητες αλλά δεμένες μεταξύ τους, προσφέροντας τα προνόμια πολλαπλών μοντέλων ανάπτυξης είναι το υβριδικό σύννεφο

Εφαρμογές του Υπολογιστικού Νέφους

Υπάρχουν τρεις κύριες εφαρμογές του cloud computing:

Compute clouds

Το compute clouds επιτρέπει την πρόσβαση σε υπολογιστικούς πόρους που έχουν υψηλή επεκτασιμότητα και μπορούν να τρέξουν οποιοδήποτε κώδικα τους ανατεθεί. Είναι το πιο ευέλικτο όσον αφορά τις υπηρεσίες και προσφέρει μεγάλη ποικιλία λύσεων.

Οι εφαρμογές αυτές ταιριάζουν συνήθως σε κάθε επιχείρηση ανεξαρτήτου μεγέθους, αλλά υπάρχει ένα μειονέκτημα καθώς δε προσφέρουν μία πρότυπη

διαχείριση, παρακολούθηση με αποτέλεσμα να μην είναι τόσο αποδοτικές σε μεγάλους οργανισμούς.

Cloud Storage

Μία από τις πρώτες υπηρεσίες του «σύννεφου» η οποία παραμένει και η πιο δημοφιλής μέχρι σήμερα είναι το cloud storage. Υπάρχουν πάνω από 100 πάροχοι που προσφέρουν παρόμοιες εφαρμογές.

Είναι ιδανική λύση για μία εταιρεία που θέλει να διατηρήσει τα αρχεία της εκτός του χώρου της επιχείρησης. Οι σημαντικότεροι περιορισμοί σε αυτό το πεδίο είναι η ασφάλεια και το κόστος.

Cloud applications

Οι εφαρμογές που παρέχονται εδώ διαφέρουν από αυτές στο compute clouds. Στη συγκεκριμένη περίπτωση εκμηδενίζουν την ανάγκη να εγκαταστήσεις και να τρέξεις μία εφαρμογή σε ένα προσωπικό υπολογιστή της εταιρείας, με αποτέλεσμα να μειώνουν αισθητά την υποχρέωση για διατήρηση, λειτουργία και υποστήριξη του λογισμικού.

Εισαγωγή στο Cloud storage

Το Cloud Storage (Αγγλικά: Cloud storage) είναι ένας τρόπος αποθήκευσης στο διαδίκτυο , δηλαδή αποθήκευση δεδομένων σε πολλούς εικονικούς διακομιστές , που φιλοξενούνται συνήθως από τρίτο μέρος και όχι από αποκλειστικό διακομιστή. Οι εταιρείες φιλοξενίας λειτουργούν κέντρα δεδομένων μεγάλης κλίμακας και όσοι χρειάζονται αποθήκευση και φιλοξενία δεδομένων μπορούν να καλύψουν τις ανάγκες αποθήκευσης δεδομένων αγοράζοντας ή μισθώνοντας χώρο αποθήκευσης από αυτά. Ο χειριστής του κέντρου δεδομένων προετοιμάζει εικονικοποιημένους πόρους αποθήκευσης στο πίσω μέρος σύμφωνα με τις ανάγκες των πελατών και τους παρέχει με τη μορφή μιας δεξαμενής αποθήκευσης και οι πελάτες μπορούν να χρησιμοποιήσουν αυτήν την ομάδα αποθήκευσης για να αποθηκεύσουν αρχεία ή αντικείμενα. Στην πράξη, αυτοί οι πόροι μπορούν να διανεμηθούν σε πολλούς κεντρικούς υπολογιστές. Η πρόσβαση στην υπηρεσία αποθήκευσης cloud γίνεται

μέσω μιας διεπαφής προγραμματισμού εφαρμογών υπηρεσίας Ιστού (API) ή μέσω μιας διεπαφής χρήστη που βασίζεται στον ιστό.



Πιθανές ανησυχίες και μειονεκτήματα

- Όταν τα δεδομένα που πρόκειται να αποθηκευτούν είναι σχετικά εμπιστευτικά, υπάρχουν ανησυχίες σχετικά με την ασφάλεια του παρόχου υπηρεσιών αποθήκευσης cloud.
- Η απόδοση πρόσβασης μπορεί να είναι χαμηλότερη από αυτή των τοπικών συσκευών αποθήκευσης.
- Η αξιοπιστία και η διαθεσιμότητα των δεδομένων θα εξαρτηθεί από το δίκτυο ευρείας περιοχής και τις προφυλάξεις που παρέχονται από τον πάροχο υπηρεσιών.
- Όταν οι χρήστες έχουν ειδικές απαιτήσεις παρακολούθησης εγγραφών χρήσης δεδομένων (όπως όταν ο δημόσιος τομέας χρειάζεται να τηρεί ορισμένα ηλεκτρομαγνητικά αρχεία σύμφωνα με κανονισμούς και κανονισμούς), η χρήση του cloud computing και της αποθήκευσης cloud θα αυξήσει την πολυπλοκότητα της εργασίας.

- Παρόλο που είναι δυνατή η παροχή δεδομένων σε πολλά άτομα ταυτόχρονα ή η μεταφορά δεδομένων σε άτομα σε διαφορετικά μέρη, όταν ένα άτομο μεταφέρει δεδομένα (για παράδειγμα, τα αρχεία μεταφέρονται από κινητό τηλέφωνο σε υπολογιστή ή από υπολογιστή σε κινητό τηλέφωνο) , είναι απαραίτητο να ανεβάσετε ξανά τα δεδομένα." και "Λήψη", θα είναι σαν μια παράκαμψη, δεν είναι τόσο γρήγορη όσο η χρήση μιας γραμμής μεταφοράς.
- Κατά τη μετάδοση μεγάλων δεδομένων, εάν το Διαδίκτυο αποσυνδεθεί ή ο πάροχος υπηρεσιών cloud κάνει κάποιο σφάλμα, το μικρό πρέπει να επαναμεταδοθεί και το μεγάλο μπορεί να οδηγήσει σε σφάλματα ή απώλεια δεδομένων.

Αποθήκευση cloud

Η πύλη αποθήκευσης cloud μπορεί να εγκατασταθεί στον πελάτη όπως απαιτείται, έτσι ώστε η υπηρεσία αποθήκευσης cloud να μπορεί να χρησιμοποιηθεί σαν να ήταν αποθηκευμένη τοπικά. Η πύλη αποθήκευσης cloud είναι η μετατροπή της διεπαφής προγραμματισμού εφαρμογών SOAP ή REST (API) της αποθήκευσης cloud σε πρωτόκολλα αποθήκευσης που βασίζονται σε μπλοκ, όπως το iSCSI και το Fiber Channel (Αντίγραφο ασφαλείας αρχείου σελίδας , αποθηκευμένο στο Αρχείο Διαδικτύου) ή ένα αρχείο που βασίζεται πρωτόκολλο αποθήκευσης δικτύου όπως NFS ή CIFS (SMB).

Σύμφωνα με μια μελέτη που πραγματοποιήθηκε το 2010, οι δημόσιες υπηρεσίες αποθήκευσης cloud γίνονται γρήγορα μια πιο ελκυστική επιλογή για τις επιχειρήσεις. [3] Το 2010, οι πάροχοι υπηρεσιών, συμπεριλαμβανομένων των Nirvanix, Nasuni (αντίγραφο ασφαλείας αρχείου σελίδων , αποθηκευμένο στο Internet Archive), CTERA Networks (αντίγραφο ασφαλείας αρχείου σελίδων , αποθηκευμένο στο Internet Archive) και Petaera άρχισαν να χρησιμοποιούν τη νέα γενιά αντικειμενοστρεφούς αποθήκευσης Η τεχνολογία έχει μειώσει σημαντικά το κόστος αποθήκευσης υπερμεγάλων δεδομένων σε σύγκριση με τα παραδοσιακά εταιρικά συστήματα αποθήκευσης.

Αυτοί οι πάροχοι υπηρεσιών παρέχουν πύλες αποθήκευσης cloud σε εταιρικούς πελάτες για να βοηθήσουν στη μετατροπή των παραδοσιακών πρωτοκόλλων

αποθήκευσης αρχείων σε προσωρινή αποθήκευση αντικειμενοστρεφούς αποθήκευσης.

Ταυτόχρονα, το "Vision Cloud EU Project" (VISION Cloud EU Project (page archive backup , stored in the Internet Archive)) που χορηγείται από την Ευρωπαϊκή Ένωση έχει επίσης δημιουργήσει μια υποκείμενη υποδομή που βασίζεται σε σύννεφο, χρησιμοποιώντας ανοιχτές προδιαγραφές και νέα τεχνολογία που παρέχει μια επεκτάσιμη, εξαιρετικά ευέλικτη και αξιόπιστη αρχιτεκτονική για τη βελτίωση των υπηρεσιών αποθήκευσης για μεγάλες ποσότητες δεδομένων.

Ο αναλυτής αποθήκευσης της IDC , Brad Nisbet, επεσήμανε ότι η παροχή αυτής της «οικονομικής και φιλικής προς τον χρήστη» υπηρεσίας λύσης έχει κάνει όλο και περισσότερες μικρομεσαίες επιχειρήσεις στην αγορά να είναι πρόθυμες να υιοθετήσουν αποθήκευση cloud.

Ασφάλεια δεδομένων

Η ασφάλεια των δεδομένων μας είναι κάτι που απασχολεί πολλούς χρήστες.

Η ανησυχία γίνεται εντονότερη όταν τα δεδομένα μας θεωρούνται «ευαίσθητα».

Ο βαθμός κινδύνου μεγαλώνει όταν δεν έχει πρόσβαση μόνο ένας χρήστης αλλά περισσότεροι. Σε αυτό το σημείο έχουμε επίπεδα ασφαλείας:

- ✓ Το πιο απλό είναι η πιστοποίηση ότι ο χρήστης που έχει δικαίωμα πρόσβασης, έχει έναν και μοναδικό κωδικό. Κατάλληλες διεργασίες πιστοποίησης θα πρέπει να είναι διαθέσιμες για να αναγνωρίσουν ότι ο χρήστης είναι ο κατάλληλος.
- ✓ Στο επόμενο επίπεδο συγκαταλέγεται η προστασία από κακόβουλες επιθέσεις στο διαδίκτυο. Θα πρέπει να υπάρχουν οι κατάλληλες διεργασίες και έλεγχοι για να εγγυηθούν στους χρήστες την αποφυγή ή την άμυνα σε τέτοιου είδους επιθέσεων.

Και στις δύο περιπτώσεις δημιουργείται η κατάλληλη πολιτική γνωστή και ως privacy policy (πολιτική απορρήτου).

Τι είναι η private policy (πολιτική απορρήτου)

Η πολιτική ξεκινάει από τις αρχικές διαπραγματεύσεις μεταξύ παρόχου και συνδρομητή. Η κατανόηση των ανησυχιών του χρήστη και των λόγων που κρύβονται πίσω από αυτές, όπως νομικά ζητήματα, είναι το πρώτο βήμα για τη δημιουργία της πολιτικής.

Ο πάροχος θα μεταφράσει τις απαιτήσεις της επιχείρησης σε τεχνικές απαιτήσεις και θα ολοκληρώσει την πολιτική.

Στη συνέχεια θα μεταφερθεί στους τεχνικούς για να εφαρμοστεί και να διατηρηθεί.

Ένας τέτοιος περιορισμός μπορεί να σχετίζεται με τα άτομα που έχουν πρόσβαση στα δεδομένα.

Η εταιρεία έχει τη δυνατότητα να παραχωρήσει μία λίστα με τα ονόματα των χρηστών που θα τους επιτρέπεται η πρόσβαση.

Επίσης μπορεί να κατηγοριοποιήσει τους χρήστες σε ομάδες.

Για παράδειγμα, ένας υπάλληλος δεν θα μπορεί να έχει πρόσβαση σε σημαντικά αρχεία της εταιρείας, σε αντίθεση ο διευθυντής θα έχει αυτή τη δυνατότητα.

Όσον αφορά την ασφάλεια των δεδομένων κατά την αποστολή και την παραλαβή τους μεταξύ του παρόχου και του συνδρομητή η μέθοδος που χρησιμοποιείται είναι η κρυπτογράφηση.

Βασικές αρχές της Ασφάλειας Δεδομένων

Υπάρχουν κάποιες βασικές αρχές ασφάλειας δεδομένων. Αυτές είναι τρεις: εμπιστευτικότητα, ακεραιότητα και διαθεσιμότητα.

A) Εμπιστευτικότητα

Αναφέρεται στην εκούσια ή ακουσία μη εξουσιοδοτημένη αποκάλυψη περιεχομένου. Ακόμα σχετίζεται και με τις περιοχές των δικαιωμάτων πνευματικής ιδιοκτησίας, ανάλυση κίνησης, διεπαφή, κρυπτογράφηση και καλυμμένα κανάλια.

Η απώλεια της μπορεί να συμβεί με ποικίλους τρόπους όπως για παράδειγμα μέσω κακής εφαρμογής δικαιωμάτων διαδικτύου. Υπάρχουν κάποια στοιχεία των τηλεπικοινωνιών που χρησιμοποιούνται για να διασφαλίσουν την εμπιστευτικότητα.

Αυτά είναι:

- Πρωτόκολλα ασφάλισης δικτύου
- Υπηρεσίες πιστοποίησης δικτύου
- Υπηρεσίες κρυπτογράφησης δεδομένων
- Δικαιώματα πνευματικής ιδιοκτησίας: περιλαμβάνονται εφευρέσεις, έργα τέχνης, μουσικής και φιλολογίας αλλά και σχεδιασμούς, τα οποία καλύπτονται από νόμους πνευματικής ιδιοκτησίας δηλαδή προστατεύουν πνευματικές δημιουργίες, διπλώματα ευρεσιτεχνίας, όπου παρέχονται για νέες εφευρέσεις.
- Καλυμμένα Κανάλια: είναι μία μη εξουσιοδοτημένη και ακούσια οδός επικοινωνίας, όπου επιτρέπει την ανταλλαγή πληροφοριών και όπου μπορούν να επιτευχθούν μέσω χρονισμού των μηνυμάτων ή την ακατάλληλη χρήση των μηχανισμών αποθήκευσης.
- Ανάλυση κίνησης: είναι μια μορφή παραβίασης της εμπιστευτικότητας που μπορεί να πραγματοποιηθεί με την ανάλυση του όγκου, την πηγή αλλά και τον προορισμό της κίνησης του μηνύματος, την ταχύτητα, ακόμη και αν αυτό είναι κωδικοποιημένο. Επίσης η αυξημένη δραστηριότητα μηνυμάτων και υψηλές εξάρσεις κίνησης μπορεί να υποδηλώνουν ότι υπάρχει κάποιο σημαντικό γεγονός που λαμβάνει χώρα.
- Κρυπτογράφηση: περιλαμβάνεται η διαμόρφωση των μηνυμάτων ώστε να είναι δύσκολο να αναγνωριστούν από μη εξουσιοδοτημένη οντότητα ακόμη και εάν έχουν υποκλαπεί. Επίσης το μέγεθος της προσπάθειας που χρειάζεται για την

αποκρυπτογράφηση του μηνύματος σχετίζεται με το πόσο ισχυρό είναι το κλειδί κρυπτογράφησης, την ποιότητα αλλά και δύναμή του αλγόριθμου κρυπτογράφησης.

- Διεπαφή: είναι η δυνατότητα μίας οντότητας να χρησιμοποιεί και να συνδέει
- πληροφορίες που προστατεύονται σε ένα επίπεδο ασφάλειας για να “ξεσκεπάσει” πληροφορίες που προστατεύονται σε ένα πιο υψηλό επίπεδο ασφάλειας. Συνήθως έχει να κάνει με την ασφάλεια της βάσης δεδομένων.

B) Ακεραιότητα

Η Ακεραιότητα αναφέρεται στην αξιοπιστία των δεδομένων. Για να υπάρχει ακεραιότητα, τα δεδομένα πρέπει να προστατεύονται από μη εξουσιοδοτημένη τροποποίηση.

Η απώλεια ακεραιότητας μπορεί να συμβεί μέσω εσκεμμένης επίθεσης για αλλαγή των πληροφοριών (για παράδειγμα μία αλλοίωση ιστοσελίδας), ή περισσότερο συχνά, χωρίς πρόθεση (τα δεδομένα αλλοιώνονται κατά λάθος από έναν χειριστή).

Τα στοιχεία που χρησιμοποιούνται για να εξασφαλίσουν την ακεραιότητα είναι:

- Υπηρεσίες ανίχνευσης εισβολής.
- Υπηρεσίες τοίχου προστασίας.
- Διαχείριση ασφάλειας επικοινωνιών.

Γ) Διαθεσιμότητα

Η διαθεσιμότητα αναφέρεται στη προσβασιμότητα των δεδομένων και διασφαλίζει την έγκαιρη πρόσβαση στα δεδομένα του σύννεφου ή στους πόρους επεξεργασίας του σύννεφου. Ακόμα εγγυάται ότι τα συστήματα λειτουργούν κανονικά όταν χρειάζονται και ότι οι υπηρεσίες ασφαλείας του συστήματος βρίσκονται σε καλή λειτουργική κατάσταση. Τέλος το αντίστροφο της εμπιστευτικότητας, ακεραιότητας και διαθεσιμότητας είναι η αποκάλυψη, η μετατροπή, και η καταστροφή.

Παραδείγματα Παρόχων Cloud Storage

Υπάρχουν πολλά συστήματα παροχής υπηρεσιών cloud storage.

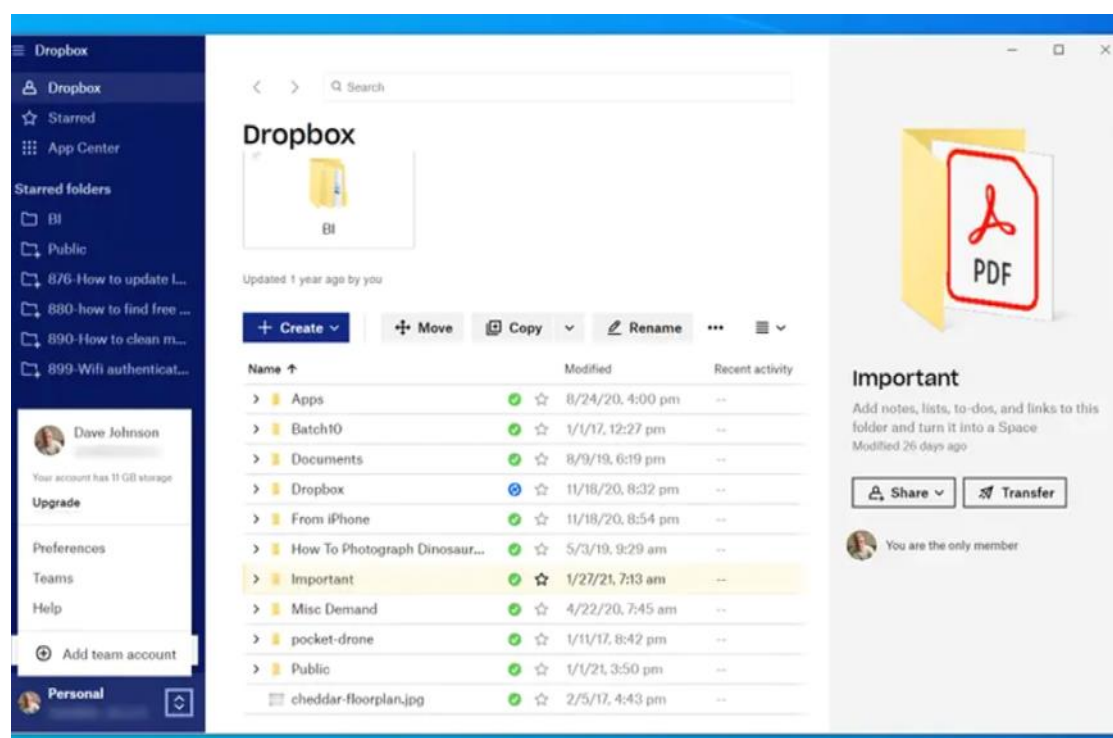
Ενδεικτικά θα αναφέρουμε τους πιο γνωστούς.

Καθένας από αυτούς παρέχει πληθώρα υπηρεσιών.

Dropbox

Το Dropbox είναι μια εφαρμογή αποθηκευτικού νέφους ή με άλλα λόγια μια υπηρεσία που επιτρέπει την αποθήκευση, τον συγχρονισμό και την κοινή χρήση αρχείων μεταξύ διαφορετικών συσκευών που ο χρήστης έχει επιλέξει να συνδέσει στο λογαριασμό του. Η υπηρεσία προσφέρει από 2GB έως 16GB δωρεάν αποθηκευτικού χώρου ενώ είναι διαθέσιμη σε Windows, Mac, Linux, iPhone, iPad, Android και BlackBerry.

Η απλή χρήση που προσφέρει το Dropbox συνδυάζεται με ένα πολύ απλό πρόγραμμα το οποίο το εγκαθίσταται πολύ εύκολα στον τοπικό υπολογιστή. Το πρόγραμμα αυτό δημιουργεί έναν φάκελο στον τοπικό υπολογιστή ο οποίος θα επικοινωνεί με τον εξυπηρετητή μέσω του Διαδικτύου. Έτσι μετακινώντας αρχεία ή φακέλους σε αυτόν τον φάκελο αυτόματα αυτά ανεβαίνουν και στον λογαριασμό



που έχει ο χρήστης στο Διαδίκτυο.

Ο συγχρονισμός των αρχείων, σε μια πολύ απλή έννοια, είναι η αντιγραφή αρχείων από έναν φάκελο σε άλλον.

Με την υπηρεσία του συγχρονισμού επομένως, μπορεί το Dropbox να μοιράσει τα αρχεία σε πολλούς υπολογιστές.



Λάβετε δεδομένα τηλεμετρίας χρησιμοποιώντας το δίαυλο εξυπηρέτησης

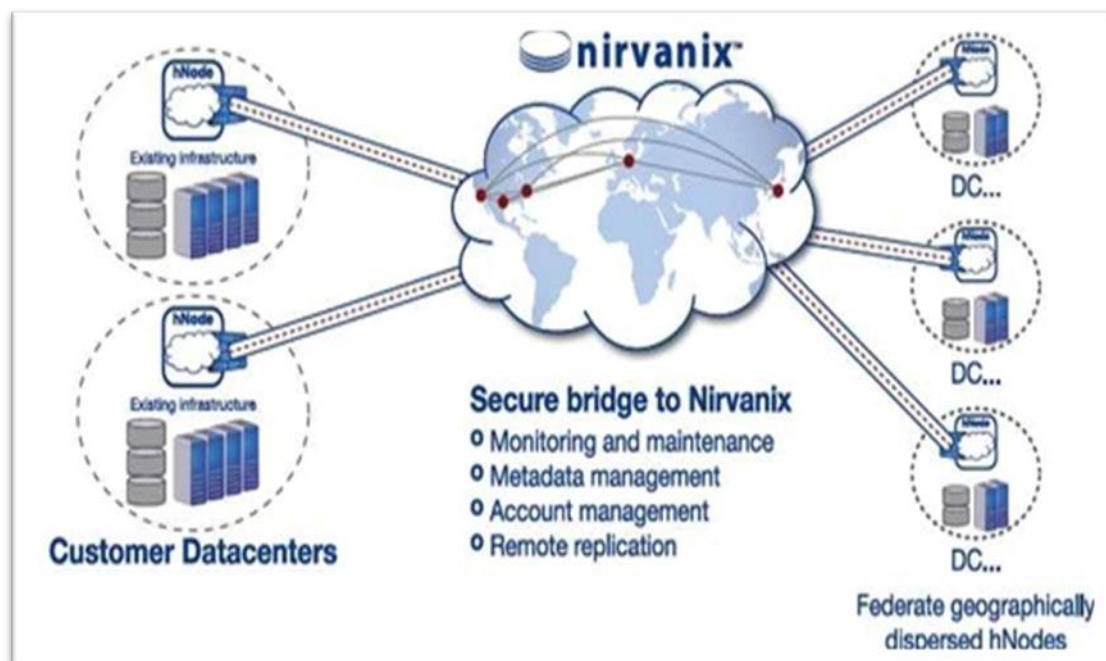
Το *Azure Service Bus* προσφέρει δύο επιλογές συμβολαίου για την αποστολή αλληλογραφίας σε ένα θέμα: HTTP ή AMQP. Σε συνδεδεμένες συσκευές, συχνά με περιορισμένο εύρος ζώνης, το AMQP έχει ορισμένα πλεονεκτήματα. Είναι ένα αποτελεσματικό, δυαδικό, αξιόπιστο και φορητό πρωτόκολλο. Διαθέτει επίσης βιβλιοθήκες για πολλές γλώσσες, περιβάλλοντα χρόνου εκτέλεσης και λειτουργικά συστήματα. Αυτό σας δίνει ευελιξία στέλνοντας μηνύματα τηλεμετρίας όταν οι συσκευές σας είναι συνδεδεμένες απευθείας στο δίαυλο εξυπηρέτησης.

Nirvanix

Η Nirvanix είναι ένας πάροχος επαγγελματικών τηλεφωνικών συστημάτων, υπηρεσιών VoIP, φιλοξενίας PBX, SIP Trunks και υπηρεσιών αποθήκευσης cloud με έδρα την Αμερική, με έδρα το Σαν Ντιέγκο, Καλιφόρνια, Ηνωμένες Πολιτείες. Η εταιρεία προσφέρει μια ποικιλία λύσεων επαγγελματικών τηλεφώνων [buzzword]

για μικρές, μεσαίες καθώς και για επιχειρήσεις. Έχει επίσης ισχυρή παρουσία σε δημόσιες, υβριδικές και ιδιωτικές υπηρεσίες αποθήκευσης cloud με τιμολόγηση βασισμένη στη χρήση. Με βάση τις προηγούμενες εταιρείες παροχής υπηρεσιών που ιδρύθηκαν το 1998, η Nirvanix έκλεισε τον Οκτώβριο του 2013. Τον Ιούλιο του 2021 η Nirvanix επανήλθε στην επιχείρηση με χρηματοδότηση 12 εκατομμυρίων δολαρίων

Το **Nirvanix Storage Delivery Network (SDN)** μετατρέπει έναν απλό server σε ένα χώρο αποθήκευσης χωρίς όρια χωρητικότητας προσβάσιμο από πολλές δημοφιλείς εφαρμογές και ο οποίος σύντομα θα αποτελεί το αρχείο ή το backup μιας επιχείρησης.

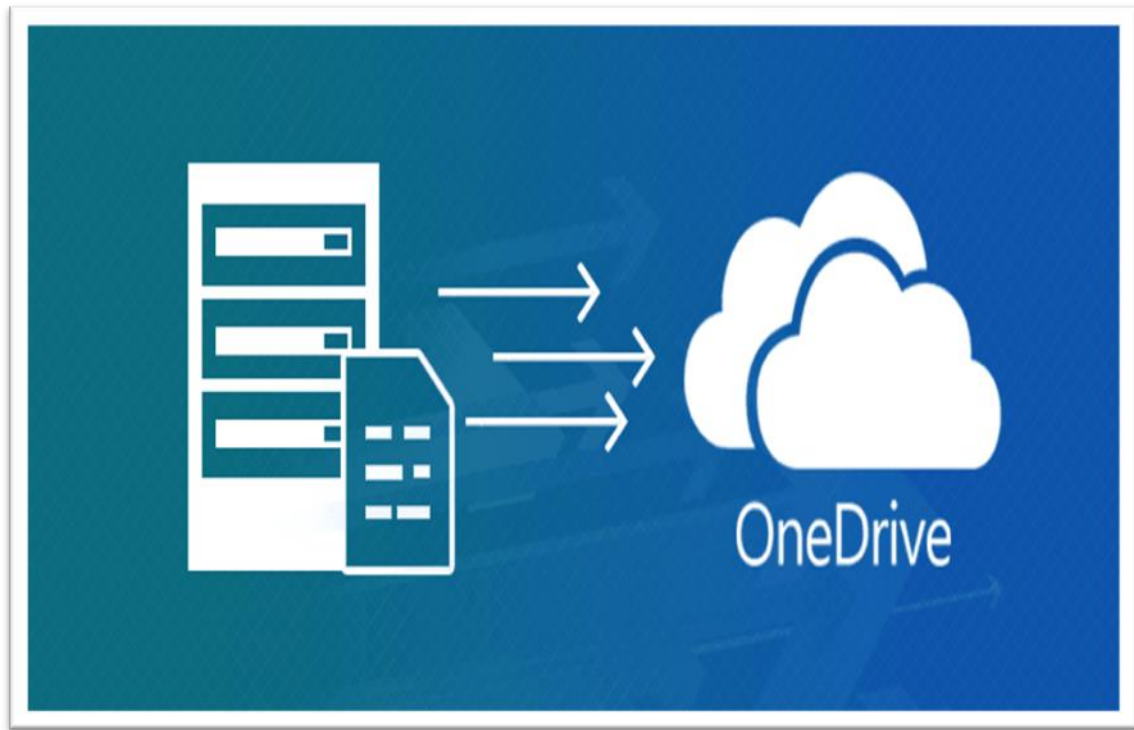


Onedrive

Τα πιο γνωστό και διαδεδομένο εργαλείο της Microsoft για αποθήκευση, και οφείλεται στο γεγονός ότι μπορεί πολύ εύκολα να αποκτήσει πρόσβαση στο

Με λίγα λόγια, όποιος έχει στη κατοχή e-mail της Microsoft, έχει πρόσβαση και στις υπηρεσίες του cloud. Η πρόσβαση στη πλατφόρμα είναι εύκολη, ο χρήστης μπορεί μπει σε αυτή από την τοποθεσία που διαχειρίζεται τα μείλ μέσω ενός εικονιδίου που φέρει το λογότυπο Onedrive.

Το σύστημα ταξινόμησης είναι εύχρηστο γιατί μας βοηθάει να αρχειοθετήσουμε και να ταξινομήσουμε τα δεδομένα μας, για να μπορούμε να βρούμε και να ανασύρουμε παλιά μας αρχεία.



Πλατφόρμα διαχείρισης δεδομένων Oracle

Οι σημερινοί οργανισμοί χρειάζονται μια λύση διαχείρισης δεδομένων που παρέχει έναν αποτελεσματικό τρόπο διαχείρισης δεδομένων σε ένα διαφορετικό αλλά ενοποιημένο επίπεδο δεδομένων. Τα συστήματα διαχείρισης δεδομένων βασίζονται σε πλατφόρμες διαχείρισης δεδομένων και μπορούν να περιλαμβάνουν βάσεις δεδομένων, λίμνες δεδομένων και αποθήκες δεδομένων, μεγάλα συστήματα διαχείρισης δεδομένων, αναλύσεις δεδομένων και άλλα.

Όλα αυτά τα στοιχεία λειτουργούν μαζί ως "βοηθητικό πρόγραμμα δεδομένων" για την παροχή των δυνατοτήτων διαχείρισης δεδομένων που χρειάζεται ένας οργανισμός για τις εφαρμογές του, καθώς και των αναλυτικών στοιχείων και αλγορίθμων που χρησιμοποιούν τα δεδομένα που προέρχονται από αυτές τις εφαρμογές.

Αν και τα τρέχοντα εργαλεία βοηθούν τους διαχειριστές βάσεων δεδομένων (DBAs) να αυτοματοποιήσουν πολλές από τις παραδοσιακές εργασίες διαχείρισης, η

χειροκίνητη παρέμβαση εξακολουθεί να απαιτείται συχνά λόγω του μεγέθους και της πολυπλοκότητας των περισσότερων αναπτύξεων βάσεων δεδομένων.

Όποτε απαιτείται χειροκίνητη παρέμβαση, αυξάνεται η πιθανότητα σφαλμάτων. Η μείωση της ανάγκης για μη αυτόματη διαχείριση δεδομένων είναι βασικός στόχος μιας νέας τεχνολογίας διαχείρισης δεδομένων, της αυτόνομης βάσης δεδομένων .

Μια πλατφόρμα διαχείρισης δεδομένων είναι το θεμελιώδες σύστημα συλλογής και ανάλυσης μεγάλων όγκων δεδομένων σε έναν οργανισμό.

Οι εμπορικές πλατφόρμες δεδομένων συνήθως περιλαμβάνουν εργαλεία λογισμικού για διαχείριση, που αναπτύχθηκαν από τον προμηθευτή βάσεων δεδομένων ή από τρίτους προμηθευτές.

Αυτές οι λύσεις διαχείρισης δεδομένων βοηθούν ομάδες IT και DBA να εκτελούν τυπικές εργασίες όπως:

- Προσδιορισμός, προειδοποίηση, διάγνωση και επίλυση βλαβών στο σύστημα βάσεων δεδομένων ή στην υποκείμενη υποδομή
- Κατανομή μνήμης βάσης δεδομένων και πόρων αποθήκευσης
- Πραγματοποίηση αλλαγών στο σχεδιασμό της βάσης δεδομένων
- Βελτιστοποίηση απαντήσεων σε ερωτήματα βάσης δεδομένων για ταχύτερη απόδοση της εφαρμογής.

Προκλήσεις διαχείρισης δεδομένων

Οι περισσότερες από τις προκλήσεις στη διαχείριση δεδομένων προέρχονται σήμερα από τον ταχύτερο ρυθμό των επιχειρήσεων και τον αυξανόμενο πολλαπλασιασμό των δεδομένων.

Η διαρκώς διευρυνόμενη ποικιλία, ταχύτητα και όγκος δεδομένων που διατίθενται στους οργανισμούς τους ωθεί να αναζητήσουν πιο αποτελεσματικά εργαλεία διαχείρισης για να συνεχίσουν.

Μερικές από τις κορυφαίες προκλήσεις που αντιμετωπίζουν οι οργανισμοί περιλαμβάνουν τα ακόλουθα:

Έλλειψη πληροφοριών:

Συλλέγονται και αποθηκεύονται δεδομένα από έναν αυξανόμενο αριθμό και ποικιλία πηγών, όπως αισθητήρες, έξυπνες συσκευές, κοινωνικά μέσα και βιντεοκάμερες. Ωστόσο, κανένα από αυτά τα δεδομένα δεν είναι χρήσιμο εάν ο οργανισμός δεν γνωρίζει ποια δεδομένα έχει, πού είναι και πώς να τα χρησιμοποιήσει. Οι λύσεις διαχείρισης δεδομένων χρειάζονται κλίμακα και απόδοση για να παρέχουν σημαντικές πληροφορίες εγκαίρως.

Δυσκολία διατήρησης επιπέδων διαχείρισης δεδομένων:

Οι οργανισμοί συλλαμβάνουν, αποθηκεύουν και χρησιμοποιούν περισσότερα δεδομένα όλη την ώρα. Για να διατηρήσουν τους κορυφαίους χρόνους απόκρισης σε αυτό το αναπτυσσόμενο επίπεδο, οι οργανισμοί πρέπει να παρακολουθούν συνεχώς τον τύπο των ερωτήσεων που απαντά η βάση δεδομένων και να αλλάζουν τα ευρετήρια καθώς τα ερωτήματα αλλάζουν - χωρίς να επηρεάζουν την απόδοση.

Προκλήσεις που συμμορφώνονται με τις μεταβαλλόμενες απαιτήσεις δεδομένων:

Οι κανονισμοί συμμόρφωσης είναι περίπλοκοι και πολυσύνθετοι και αλλάζουν συνεχώς. Οι οργανισμοί πρέπει να είναι σε θέση να ελέγχουν εύκολα τα δεδομένα τους και να αναγνωρίζουν οτιδήποτε υπόκειται σε νέες ή τροποποιημένες απαιτήσεις. Συγκεκριμένα, οι προσωπικά αναγνωρίσιμες πληροφορίες (PII) πρέπει να εντοπίζονται, να παρακολουθούνται και να παρακολουθούνται για συμμόρφωση με όλο και πιο αυστηρούς παγκόσμιους κανονισμούς απορρήτου.

Πρέπει να επεξεργαστείτε και να μετατρέψετε εύκολα δεδομένα:

Η συλλογή και η ταυτοποίηση των δεδομένων δεν παρέχει καμία αξία - ο οργανισμός πρέπει να τα επεξεργαστεί. Εάν χρειάζεται πολύς χρόνος και προσπάθεια για τη μετατροπή των δεδομένων σε αυτό που χρειάζονται για ανάλυση, αυτή η ανάλυση δεν θα συμβεί. Ως αποτέλεσμα, η πιθανή τιμή αυτών των δεδομένων χάνεται.

Διαρκής ανάγκη αποθήκευσης δεδομένων αποτελεσματικά:

Στον νέο κόσμο της διαχείρισης δεδομένων, οι οργανισμοί αποθηκεύουν δεδομένα σε πολλαπλά συστήματα, συμπεριλαμβανομένων αποθηκών δεδομένων και μη δομημένων λιμνών δεδομένων που αποθηκεύουν δεδομένα σε οποιαδήποτε μορφή σε ένα μοναδικό αποθετήριο. Οι επιστήμονες δεδομένων ενός οργανισμού χρειάζονται έναν τρόπο γρήγορης και εύκολης μετατροπής των δεδομένων από την αρχική του μορφή σε σχήμα, μορφή ή μοντέλο που χρειάζονται για να είναι σε ένα ευρύ φάσμα αναλύσεων.

Ζητήστε να βελτιστοποιείτε συνεχώς την ευελιξία και το κόστος της πληροφορικής: Με τη διαθεσιμότητα συστημάτων διαχείρισης δεδομένων cloud, οι οργανισμοί μπορούν τώρα να επιλέξουν εάν θα διατηρούν και θα αναλύουν δεδομένα σε περιβάλλοντα εσωτερικού χώρου, στο cloud ή σε ένα υβριδικό μείγμα των δύο. Οι οργανισμοί πληροφορικής πρέπει να αξιολογήσουν το επίπεδο ταυτότητας μεταξύ εσωτερικών χώρων και cloud προκειμένου να διατηρήσουν τη μέγιστη ευελιξία πληροφορικής και το χαμηλότερο κόστος.

Βέλτιστες πρακτικές διαχείρισης δεδομένων

Η αντιμετώπιση των προκλήσεων διαχείρισης δεδομένων απαιτεί ένα ολοκληρωμένο, καλά μελετημένο σύνολο βέλτιστων πρακτικών. Αν και συγκεκριμένες βέλτιστες πρακτικές διαφέρουν ανάλογα με τον τύπο των δεδομένων και τον κλάδο, οι ακόλουθες βέλτιστες πρακτικές αντιμετωπίζουν τις μεγάλες προκλήσεις διαχείρισης δεδομένων που αντιμετωπίζουν σήμερα οι οργανισμοί:

Δημιουργήστε ένα επίπεδο εντοπισμού για να προσδιορίσετε τα δεδομένα σας:

Ένα επίπεδο ανακάλυψης πάνω από το επίπεδο δεδομένων του οργανισμού σας επιτρέπει σε αναλυτές και επιστήμονες δεδομένων να αναζητούν και να αναζητούν σύνολα δεδομένων για να κάνουν τα δεδομένα σας χρησιμοποιήσιμα.

Αναπτύξτε ένα περιβάλλον επιστήμης δεδομένων για την αποτελεσματική επαναχρησιμοποίηση των δεδομένων σας :

Ένα περιβάλλον επιστήμης δεδομένων αυτοματοποιεί όσο το δυνατόν περισσότερες εργασίες μετατροπής δεδομένων, εξορθολογίζοντας τη δημιουργία και την αξιολόγηση μοντέλων δεδομένων. Ένα σύνολο εργαλείων που εξαλείφει την ανάγκη για χειροκίνητο μετασχηματισμό δεδομένων μπορεί να επιταχύνει την υπόθεση και τη δοκιμή νέων μοντέλων.

Χρησιμοποιήστε ένα κοινό επίπεδο ερωτημάτων για τη διαχείριση πολλαπλών και διαφορετικών μορφών αποθήκευσης δεδομένων :

Οι νέες τεχνολογίες καθιστούν δυνατή τη συνεργασία των αποθετηρίων διαχείρισης δεδομένων, κάνοντας τις διαφορές μεταξύ τους να εξαφανίζονται. Ένα κοινό επίπεδο ερωτήσεων που καλύπτει τα πολλά είδη αποθήκευσης δεδομένων επιτρέπει στους επιστήμονες δεδομένων, τους αναλυτές και τις εφαρμογές να έχουν πρόσβαση σε δεδομένα χωρίς να χρειάζεται να γνωρίζουν πού αποθηκεύονται και χωρίς να χρειάζεται να τα μετατρέψουν χειροκίνητα σε μια χρησιμοποιήσιμη μορφή.

Η βάση δεδομένων Oracle

Ένα σύστημα βάσης δεδομένων Oracle διακρίνεται μοναδικά από έναν κωδικό αναγνώρισης συστήματος (SID, Site ID) που αποτελείται από γράμματα και αριθμούς [4] - συμπεριλαμβανομένης τουλάχιστον μιας παρουσίας μιας εφαρμογής και μιας συσκευής αποθήκευσης δεδομένων. Μια παρουσία (στιγμιότυπο) αντιπροσωπεύεται από έναν αριθμό παρουσίας (ή ένα αναγνωριστικό ενεργοποίησης: SYS.V_\$DATABASE.ACTIVATION#)—περιέχει μια συλλογή προγραμμάτων του λειτουργικού συστήματος και εσωτερικών δομών που συνομιλούν με συσκευές αποθήκευσης. Τυπικά προγράμματα είναι το PMON (η παρακολούθηση προγράμματος παρακολούθησης διεργασιών) και το SMON (η παρακολούθηση συστήματος παρακολούθησης συστήματος) Οι χρήστες της βάσης δεδομένων Oracle έχουν σχεδιάσει τη δομή μνήμης SGA (System Global Area) από την πλευρά του διακομιστή. Το SGA διατηρεί πληροφορίες κρυφής μνήμης, όπως buffer δεδομένων, εντολές SQL και πληροφορίες χρήστη. Εκτός του χώρου αποθήκευσης, η βάση δεδομένων αποτελείται από ένα ηλεκτρονικό αρχείο καταγραφής ανάκτησης που διατηρεί το ιστορικό των αλλαγών δεδομένων. Οι

διεργασίες μπορούν να μετατρέψουν τα ηλεκτρονικά αρχεία καταγραφής ανάκτησης σε αρχειοθετημένα αρχεία καταγραφής (αρχεία καταγραφής ανάκτησης εκτός σύνδεσης), παρέχοντας τη βάση για ανάκτηση και αναγέννηση δεδομένων.

Εάν ο διαχειριστής της βάσης δεδομένων Oracle εφαρμόζει το Oracle RAC (Real Application Cluster), τότε πολλές παρουσίες σε διαφορετικούς διακομιστές ανήκουν σε μια κεντρική συστοιχία δίσκων . Αυτό το σχήμα προσφέρει πολλά πλεονεκτήματα, όπως υψηλότερη απόδοση, επεκτασιμότητα και πλεονασμό. Ωστόσο, αυτό καθιστά το σύστημα πιο περίπλοκο, έτσι πολλοί ιστότοποι δεν χρησιμοποιούν RAC. Στην έκδοση 10g, ο υπολογισμός πλέγματος εισήγαγε κοινόχρηστους πόρους, έτσι ώστε ένα στιγμιότυπο να μπορεί να χρησιμοποιήσει (για παράδειγμα) τους πόρους της CPU ενός άλλου κόμβου στο πλέγμα.

Το σύστημα διαχείρισης βάσεων δεδομένων Oracle μπορεί να αποθηκεύσει και να εκτελέσει τις δικές του αποθηκευμένες διαδικασίες και λειτουργίες. Η PL-SQL (η ιδιόκτητη γλώσσα επέκτασης SQL της Oracle) ή η αντικειμενοστραφής γλώσσα Java , μπορεί να καλέσει αυτά τα αντικείμενα κώδικα ή να γράψει τον κώδικα.

Αποθήκευση

Το σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων Oracle αποθηκεύει λογικά δεδομένα σε tablespaces και φυσικά με τη μορφή αρχείων δεδομένων. Ένας πίνακας μπορεί να περιέχει διάφορους τύπους τμημάτων μνήμης , όπως τμήματα δεδομένων, τμήματα ευρετηρίου κ.λπ. Κατά συνέπεια, ένα τμήμα αποτελείται από μία ή περισσότερες εκτάσεις. Μια επέκταση αποτελείται από συνδεδεμένα μπλοκ δεδομένων. Ένα μπλοκ δεδομένων είναι η βασική μονάδα αποθήκευσης δεδομένων.

Το σύστημα διαχείρισης βάσεων δεδομένων Oracle SYSTEM παρακολουθεί την αποθήκευση δεδομένων μέσω πληροφοριών που είναι αποθηκευμένες σε tablespace. SYSTEMO πίνακας περιέχει το λεξικό δεδομένων — καθώς και (από προεπιλογή) ευρετήρια και συμπλέγματα . Το λεξικό δεδομένων περιέχει έναν πίνακα που περιέχει πληροφορίες για όλα τα αντικείμενα χρήστη στη βάση δεδομένων . Ξεκινώντας με την έκδοση 8i, η Oracle έχει υποστηρίξει εγγενώς διαχειριζόμενους επιτραπέζιους χώρους, οι οποίοι αποθηκεύουν πληροφορίες διαχείρισης χώρου στα δικά τους bitmaps κεφαλίδας και όχι σε SYSTEMtablespace.

Παγκοσμιοποίηση

Το λογισμικό βάσης δεδομένων Oracle είναι διαθέσιμο σε 63 γλώσσες (συμπεριλαμβανομένων των τοπικά διαφοροποιημένων γλωσσών, όπως τα Αγγλικά των ΗΠΑ και τα Αγγλικά του Ηνωμένου Βασιλείου). Οι διαφορές μεταξύ των γλωσσικών εκδόσεων περιλαμβάνουν ονόματα ημέρας και μήνα, συντμήσεις, μορφές ώρας και ταξινόμηση.

Η Oracle Corporation έχει μεταφράσει μηνύματα σφάλματος της βάσης δεδομένων Oracle στα Αραβικά, Καταλανικά, Κινέζικα, Τσέχικα, Δανικά, Ολλανδικά, Αγγλικά, Φινλανδικά, Γαλλικά, Γερμανικά, Ελληνικά, Εβραϊκά, Ουγγρικά, Ιταλικά, Ιαπωνικά, Κορεατικά, Νορβηγικά, Πολωνικά, Πορτογαλικά, Ρουμανικά, Ρωσικά , Σλοβακικά, Ισπανικά, Σουηδικά, Ταϊλανδέζικα και Τουρκικά. Η Oracle Corporation παρέχει στους προγραμματιστές βάσεων δεδομένων τα εργαλεία και τους μηχανισμούς για τη δημιουργία διεθνοποιημένων εφαρμογών βάσεων δεδομένων.

Εργαλεία διαχείρισης και ανάπτυξης βάσεων δεδομένων

Το Navicat for Oracle (αντίγραφο ασφαλείας αρχείου σελίδων , αποθηκευμένο στο Internet Archive) είναι ένα ισχυρό εργαλείο διαχείρισης και ανάπτυξης βάσεων δεδομένων ειδικά σχεδιασμένο για την Oracle. Μπορεί να χρησιμοποιηθεί με οποιαδήποτε έκδοση της βάσης δεδομένων Oracle και υποστηρίζει τις περισσότερες από τις λειτουργίες της Oracle, συμπεριλαμβανομένων κανόνων, ευρετηρίων, προβολών κ.λπ.

Το Toad for Oracle (αντίγραφο ασφαλείας αρχείου σελίδων , αποθηκευμένο στο Internet Archive) είναι ένα παλιομοδίτικο εργαλείο διαχείρισης ανάπτυξης Oracle με περισσότερες λειτουργίες από οποιοδήποτε εργαλείο διαχείρισης ανάπτυξης Oracle και έχει πολλαπλές εκδόσεις υποκαταστημάτων για διαφορετικούς ρόλους χρηστών.

Οι εκδόσεις περιλαμβάνουν: Toad DBA Suite for Oracle είναι ένα εργαλείο για τη διαχείριση βάσεων δεδομένων Oracle για Oracle DBA, Toad Development Suite για Oracle είναι ένα εργαλείο ανάπτυξης για την Oracle, Toad DBA Suite για Oracle – To Exadata Edition είναι ένα εργαλείο για Oracle Exadata all-in-one Εργαλεία διαχείρισης βάσεων δεδομένων διακομιστή και Oracle, το Toad DBA Suite για Oracle - RAC Edition είναι ένα εργαλείο διαχείρισης DBA ειδικά κατασκευασμένο για την Oracle για τη δημιουργία συμπλέγματος RAC

Χαρακτηριστικά του Oracle

Μια βάση δεδομένων της Oracle προσφέρει τις ακόλουθες δυνατότητες για την κάλυψη των απαιτήσεων της ισχυρής διαχείρισης βάσης δεδομένων:

Επεκτασιμότητα και απόδοση:

Λειτουργίες όπως το Real Application Clustering και το Portability καθιστούν μια βάση δεδομένων της Oracle επεκτάσιμη ανάλογα με τη χρήση. Σε μια βάση δεδομένων πολλαπλών χρηστών, απαιτείται ο έλεγχος της συνοχής και της ταυτότητας δεδομένων που εξετάζονται από την Oracle.

Διαθεσιμότητα:

Οι εφαρμογές σε πραγματικό χρόνο απαιτούν υψηλή διαθεσιμότητα δεδομένων. Τα υπολογιστικά περιβάλλοντα υψηλής απόδοσης έχουν διαμορφωθεί ώστε να παρέχουν διαθεσιμότητα δεδομένων όλων των εποχών. Τα δεδομένα είναι διαθέσιμα κατά τη διάρκεια των προγραμματισμένων ή μη προγραμματισμένων διακοπών λειτουργίας και αστοχιών.

Δημιουργία αντιγράφων ασφαλείας και ανάκτηση:

Η διάταξή του ολοκληρώνει τις δυνατότητες ανάκτησης για την ανάκτηση δεδομένων από σχεδόν όλα τα είδη βλαβών. Σε περίπτωση αστοχίας, η βάση δεδομένων πρέπει να ανακτηθεί σε σύντομο χρονικό διάστημα για υψηλή διαθεσιμότητα. Μη προσβεβλημένα μέρη δεδομένων είναι διαθέσιμα ενώ τα πληγέντα ανακτούν.

Ασφάλεια

Η ασφάλεια των δεδομένων είναι πάντα η πρώτη προτεραιότητα. Η **Oracle** παρέχει μηχανισμούς για τον έλεγχο της πρόσβασης και της χρήσης δεδομένων. Η εφαρμογή εξουσιοδότησης και η επεξεργασία ενεργειών χρήστη μπορούν να αποτρέψουν τη μη εξουσιοδοτημένη πρόσβαση και να επιτρέψουν ξεχωριστή πρόσβαση στους χρήστες.



Oracle Cloud

Το **Oracle Cloud** είναι το πρώτο δημόσιο **cloud** που σχεδιάστηκε από την αρχή για να είναι το καλύτερο **cloud** για κάθε εφαρμογή.

Με την επανεξέταση του σχεδιασμού στοιχείων και συστημάτων υπολογιστικού νέφους, δημιουργήσαμε καινοτόμες λύσεις που επιταχύνουν τη μετεγκατάσταση, βελτιώνουν την αξιοπιστία και την απόδοση για όλες τις εφαρμογές και παρέχουμε υπηρεσίες από άκρο σε άκρο που χρειάζονται οι πελάτες για να αναπτύξουν καινοτόμες εφαρμογές **cloud**.

Το **OCI Cloud Adoption Framework** βοηθά τις εταιρείες να απλοποιήσουν τη μετάβασή τους στο **cloud**, προσφέροντας ένα πολύτιμο σύνολο πόρων **cloud**, βέλτιστες πρακτικές, σεμινάρια και εργαλεία ανάπτυξης. Η πλατφόρμα βοηθά τους πελάτες να καθορίσουν μια ισχυρή στρατηγική cloud, να σχεδιάσουν την επιτυχή μετεγκατάσταση του φόρτου εργασίας και να εξασφαλίσουν την πλήρη διαχείριση των περιβαλλόντων cloud. Το **Cloud Adoption Framework** παρέχει στους πελάτες μια μεθοδολογία για την αξιοποίηση των ενσωματωμένων δυνατοτήτων του **Oracle Cloud**, όπως οι Υπηρεσίες **Cloud Lift** και βασικά οικονομικά προγράμματα, συμπεριλαμβανομένων των **Oracle Support Rewards**

Το **Oracle Cloud** ορίζεται ως μια υπηρεσία που παρουσιάζεται από την **Oracle Corporation** για **cloud computing** που παρέχει αποθήκευση, διακομιστές, εφαρμογές, υπηρεσίες και δίκτυα μέσω ενός παγκόσμιου δικτύου ολοκληρωμένων κέντρων δεδομένων της **Oracle Corporation**. Η εταιρεία επιτρέπει αυτές τις υπηρεσίες για παροχή κατ' απαίτηση μέσω Διαδικτύου.



Παροχές του Oracle Cloud

Όπως οι διάφορες πλατφόρμες Cloud, μας προσφέρουν τις δικές τους υπηρεσίες, έτσι και το Oracle Cloud παρέχει τα εξής:

- Το Oracle cloud προσφέρει πλατφόρμα ως υπηρεσία (PaaS), υποδομή ως υπηρεσία (IaaS), δεδομένα ως υπηρεσία (DaaS) και λογισμικό ως υπηρεσία (SaaS).

Και πάλι, αυτές οι υπηρεσίες εφαρμόζονται για τη δημιουργία, ενσωμάτωση, επέκταση και ανάπτυξη εφαρμογών στο cloud.

- Αυτή η πλατφόρμα παρέχει αρκετά ανοιχτά πρότυπα πολλές HTML5, SQL, REST κ.λπ., εφαρμογές ανοιχτού κώδικα πολλές Spark, Kafka, Hadoop, MySQL, Terraform, Kubernetes κ.λπ. και πολλές πολλές γλώσσες προγραμματισμού, εργαλεία, βάσεις δεδομένων, και πλαίσια που περιλαμβάνουν Open Source, Oracle Specific καθώς και συστήματα και λογισμικό τρίτων.

Τρόπος λειτουργίας του Oracle Cloud

Η Oracle καθορίζει τις εξελίξεις στην τεχνολογία της πλατφόρμας cloud, δημιουργώντας το Oracle Autonomous Cloud ως πλήρη πλατφόρμα.

Οι δυνατότητες αυτο-οδήγησης που περιέχονται στην πλατφόρμα επηρεάζουν την τεχνητή νοημοσύνη (AI) και επίσης τη μηχανική μάθηση για να ωφελήσουν τους οργανισμούς με μικρότερο κόστος, να μειώσουν τον κίνδυνο, να λάβουν αναλυτικές πληροφορίες και να επιταχύνουν την καινοτομία.

Επίσης, η υπηρεσία της Oracle με την ονομασία PaaS (Platform as a Service) παρέχει εκτεταμένα εργαλεία για τη βελτίωση, την ενσωμάτωση, την ασφάλεια και την παρακολούθηση των εφαρμογών, συμπεριλαμβανομένων των πόρων για την πρακτική της ανάλυσης για την κατασκευή παράνομων, μετασχηματιστικών πληροφοριών.

Πλεονεκτήματα και μειονεκτήματα

Τα πλεονεκτήματα του Oracle cloud αναφέρονται παρακάτω:

- ✓ Προσφέρει εξοικονόμηση κόστους υλικού, ανώτερη αποτελεσματικότητα και ανάπτυξη επιχειρηματικής ευελιξίας μακροπρόθεσμα.
- ✓ Παρέχει αυτοματοποιημένη και απλουστευμένη απόδοση εργασίας χρησιμοποιώντας κοινόχρηστη υποδομή δίνοντας τη δυνατότητα στις εργασίες να εκτελούν γρήγορα κλίμακα προς τα πάνω ή προς τα κάτω για την ικανοποίηση των απαιτήσεων.

- ✓ Παροχή μιας μελλοντικής απόδειξης λύσης με εύκολη απρόσκοπτη εργασία για τη μετακίνηση του παλαιού φόρτου εργασίας σε σύγχρονες πλατφόρμες.
- ✓ Υποστηρίζει βελτιωμένη διαδικασία μετεγκατάστασης εφαρμογών όπως ERP, λογισμικό ανάλυσης και επίσης την παρούσα αρχιτεκτονική στο cloud.
- ✓ Προσφέρει επεκτάσιμη υποδομή, υλικό, δυνατότητες ανάπτυξης, συμπεριλαμβανομένων επιλογών λογισμικού που έχουν προβλέψει για επιχειρήσεις.

Τα μειονεκτήματά του είναι:

- ✓ Τα στοιχεία υποδομής του Oracle Cloud δεν επαρκούν για την εκτέλεση όλων των λύσεων cloud.
- ✓ Απαιτείται βελτίωση για την πλοήγηση UI (User Interface).
- ✓ Η διεπαφή χρήστη αποτελείται από δυσλειτουργίες και απαιτεί τη συμπλήρωση ετικετών για την ανάπτυξη των υπηρεσιών.
- ✓ Ενσωμάτωση με λίγα άλλα εργαλεία ή ότι οι λύσεις cloud δεν είναι εμφανείς ή δεν υπάρχουν.
- ✓ Η ικανότητα εξισορρόπησης φορτίου περιλαμβάνει μερική λειτουργικότητα.
- ✓ Υπάρχουν προβλήματα με τη συνδεσιμότητα της βάσης δεδομένων.
- ✓ Η απαίτηση για επιπλέον πρότυπα.
- ✓ Κίνδυνος εμπιστευτικότητας δεδομένων.
- ✓ Έχει ευπάθεια σε τυχόν κακόβουλες επιθέσεις.

Συστήματα Διαχείρισης Δεδομένων

Γιατί να χρησιμοποιήσετε μια βάση δεδομένων;

Με την ταχεία ανάπτυξη της τεχνολογίας του Διαδικτύου, αναμένεται ότι ο αριθμός των χρηστών του Διαδικτύου στον κόσμο θα φτάσει τα 5 δισεκατομμύρια μέχρι το τέλος του 2020. Η αύξηση του αριθμού των χρηστών του Διαδικτύου έχει οδηγήσει στην ανάπτυξη των διαδικτυακών αγορών, του microblogging, του διαδικτυακού

βίντεο και άλλων βιομηχανιών. Στη συνέχεια, μαζί του έρχεται ένας τεράστιος όγκος δεδομένων δικτύου.

Τεράστιες ποσότητες δεδομένων παράγονται συνεχώς, πώς να τα αποθηκεύσετε, να τα ανακτήσετε και να τα διαχειριστείτε με ασφάλεια και αποτελεσματικότητα; Ως εκ τούτου, τα ζητήματα της αποτελεσματικής αποθήκευσης, της αποτελεσματικής πρόσβασης, της εύκολης κοινής χρήσης και του ελέγχου ασφάλειας των δεδομένων έχουν γίνει ένα πολύ σημαντικό ζήτημα στην εποχή της πληροφορίας.

Η χρήση βάσεων δεδομένων μπορεί να αποθηκεύσει δεδομένα αποτελεσματικά και με καλά οργανωμένο τρόπο, γεγονός που επιτρέπει στους ανθρώπους να διαχειρίζονται τα δεδομένα πιο γρήγορα και εύκολα, κυρίως στις ακόλουθες πτυχές.

1) Η βάση δεδομένων μπορεί να αποθηκεύσει μεγάλο όγκο πληροφοριών δεδομένων με δομημένο τρόπο, που είναι βολικό για τους χρήστες να ανακτούν και να έχουν αποτελεσματική πρόσβαση.

Η βάση δεδομένων μπορεί να ταξινομήσει και να αποθηκεύσει τα δεδομένα και μπορεί να παρέχει γρήγορη ερώτηση. Για παράδειγμα, όταν συνήθως χρησιμοποιούμε το Baidu για την αναζήτηση περιεχομένου, το Baidu βασίζεται επίσης σε βάσεις δεδομένων και τεχνολογία ταξινόμησης δεδομένων για την επίτευξη του σκοπού της γρήγορης αναζήτησης.

2) Η βάση δεδομένων μπορεί να διατηρήσει αποτελεσματικά τη συνέπεια και την ακεραιότητα των πληροφοριών δεδομένων και να μειώσει τον πλεονασμό δεδομένων.

Μπορεί κάλλιστα να διασφαλίσει ότι τα δεδομένα είναι έγκυρα και ότι δεν καταστρέφονται, και η ίδια η βάση δεδομένων έχει τη λειτουργία της αποφυγής διπλών δεδομένων, έτσι ώστε να μειωθεί ο πλεονασμός των δεδομένων.

3) Η βάση δεδομένων μπορεί να καλύψει τις απαιτήσεις κοινής χρήσης και ασφάλειας εφαρμογών, και η τοποθέτηση δεδομένων στη βάση δεδομένων γίνεται επίσης για λόγους ασφαλείας σε πολλές περιπτώσεις.

Για παράδειγμα, εάν όλες οι πληροφορίες των εργαζομένων και τα δεδομένα μισθοδοσίας τοποθετούνταν σε αρχεία δίσκου, η εμπιστευτικότητα της μισθοδοσίας θα ήταν αδύνατη. Εάν τα στοιχεία των εργαζομένων και τα δεδομένα μισθών τοποθετηθούν στη βάση δεδομένων, μόνο τα στοιχεία των εργαζομένων μπορούν να αναζητηθούν και να τροποποιηθούν, ενώ τα στοιχεία μισθού μπορούν να προβληθούν μόνο από καθορισμένα άτομα (όπως το οικονομικό προσωπικό), διασφαλίζοντας έτσι την ασφάλεια των δεδομένων.

4) Η τεχνολογία βάσεων δεδομένων μπορεί να διευκολύνει και να αναλύσει έξυπνα και να δημιουργήσει νέες χρήσιμες πληροφορίες.

Για παράδειγμα, σε ένα σούπερ μάρκετ, οι πληροφορίες πωλήσεων των ειδών αποθηκεύονται στη βάση δεδομένων και η κατάταξη πωλήσεων κάθε μήνα καθορίζει την ποσότητα των αγορών στο δεύτερο εξάμηνο του μήνα. Τα αποτελέσματα των ερωτημάτων της βάσης δεδομένων δημιουργούν στην πραγματικότητα νέες πληροφορίες δεδομένων.

Τεχνολογίες όπως η εξόρυξη δεδομένων και η διαδικτυακή ανάλυση έχουν αναπτυχθεί πολύ γρήγορα τα τελευταία χρόνια και η βασική τους σημασία είναι η ανάλυση χρήσιμων πληροφοριών από ένα σωρό δεδομένων.

Τι βάσεις δεδομένων έχουμε γύρω μας;

Στην καθημερινή μας ζωή, μπορεί να μην γνωρίζουμε τα συστήματα βάσεων δεδομένων που χρησιμοποιούμε συχνά. Πριν μάθουμε επίσημα τη βάση δεδομένων, ας ρίξουμε μια ματιά στις κοινές εφαρμογές συστήματος βάσης δεδομένων γύρω σας.

Ακολουθούν ορισμένες κοινές εφαρμογές βάσεων δεδομένων στη ζωή.

Ψώνια σούπερ μάρκετ

Όταν αγοράζουμε κάτι στο σούπερ μάρκετ, έχουμε πρόσβαση σε μια βάση δεδομένων. Στο ταμείο, ο ταμίας σαρώνει κάθε προϊόν που αγοράζει ένας πελάτης με συσκευή ανάγνωσης γραμμωτού κώδικα. Αυτός ο αναγνώστης γραμμωτού κώδικα συνδέεται με μια εφαρμογή που έχει πρόσβαση στη βάση δεδομένων εμπορευμάτων, το πρόγραμμα βρίσκει την τιμή του εμπορεύματος από τη βάση

δεδομένων εμπορευμάτων σύμφωνα με τον γραμμωτό κώδικα, στη συνέχεια αφαιρεί την ποσότητα αυτού του εμπορεύματος για αυτήν την πώληση από το απόθεμα και εμφανίζει την αντίστοιχη τιμή στο οθόνι. Εάν το απόθεμα είναι χαμηλότερο από το καθορισμένο όριο, το σύστημα βάσης δεδομένων θα σας ζητήσει να αγοράσετε για να συμπληρώσετε το απόθεμα. Εάν ένας πελάτης τηλεφωνήσει στο σούπερ μάρκετ για να παραγγείλει ένα προϊόν, ο πωλητής μπορεί να εκτελέσει την εφαρμογή για να δει αν υπάρχει αρκετό απόθεμα για το προϊόν στη βάση δεδομένων.

Δαπάνες πιστωτικών καρτών

Όταν αγοράζετε αγαθά με πιστωτική κάρτα, ο πωλητής ελέγχει αν ο πελάτης έχει επαρκή πίστωση. Αυτός ο έλεγχος μπορεί να γίνει τηλεφωνικά ή αυτόματα μέσω συσκευής ανάγνωσης πιστωτικών καρτών συνδεδεμένη σε σύστημα υπολογιστή. Είτε έτσι είτε αλλιώς, πρέπει να υπάρχει κάποια βάση δεδομένων με όλες τις πληροφορίες σχετικά με τις αγορές με πιστωτική κάρτα του πελάτη.

Για να ελέγξετε την πιστωτική κατάσταση του πελάτη, απαιτείται μια αίτηση βάσης δεδομένων. Χρησιμοποιώντας τον αριθμό της πιστωτικής κάρτας, μπορείτε να ρωτήσετε τη συνολική τιμή των αγαθών που αγόρασε ο πελάτης αυτόν τον μήνα και την τιμή των αγαθών με τα οποία θέλει να αγοράσει ο πελάτης. Η πιστωτική κάρτα αυτή τη φορά μπορεί να προστεθεί για να διαπιστωθεί εάν εξακολουθεί να είναι εντός του πιστωτικού ορίου. Και, εάν ο χρήστης επιβεβαιώσει την αγορά, τα στοιχεία του αγορασμένου προϊόντος θα καταγραφούν στη βάση δεδομένων.

Η εφαρμογή έχει επίσης πρόσβαση σε μια βάση δεδομένων για να ελέγξει ότι η πιστωτική κάρτα δεν έχει κλαπεί ή χαθεί πριν συμφωνήσετε σε μια αγορά. Επιπλέον, υπάρχουν γενικά άλλες εφαρμογές που είναι υπεύθυνες για την αποστολή αρχείων χρήσης πιστωτικών καρτών στους κατόχους πιστωτικών καρτών σε μηνιαία βάση και την αποστολή πληροφοριών στους κατόχους πιστωτικών καρτών μετά τη λήψη της πληρωμής.

Κράτηση ταξιδιωτικού πρακτορείου για διακοπές

Όταν ένας ταξιδιώτης ρωτά ένα ταξιδιωτικό γραφείο για το δρομολόγιο των διακοπών του, το ταξιδιωτικό γραφείο έχει πρόσβαση σε πολλές βάσεις δεδομένων που περιέχουν λεπτομέρειες διακοπών και πτήσης.

Εάν ένας πελάτης κάνει κράτηση για ένα ταξίδι, το σύστημα βάσης δεδομένων πρέπει να κάνει όλες τις απαραίτητες ρυθμίσεις κράτησης. Στην περίπτωση αυτή, το σύστημα πρέπει να διασφαλίζει ότι μια θέση δεν δεσμεύεται από δύο διαφορετικά ταξιδιωτικά πρακτορεία και ότι οι δεσμευμένες θέσεις για την πτήση δεν υπερβαίνουν τις σταθερές θέσεις για την πτήση.

Για παράδειγμα, ας υποθέσουμε ότι έχει απομείνει μόνο μία τελευταία θέση στην πτήση από το Qinghai προς το Πεκίνο, αλλά δύο ταξιδιωτικά γραφεία ζητούν να κάνουν κράτηση θέσης ταυτόχρονα, το σύστημα πρέπει να μπορεί να διακρίνει και να χειρίζεται αυτήν την κατάσταση, δηλαδή να επιτρέπει μία κράτηση για να συνεχίσετε, ειδοποιήστε το άλλο ταξιδιωτικό γραφείο ότι δεν έχουν απομείνει θέσεις. Ένα ταξιδιωτικό γραφείο μπορεί συχνά να έχει άλλη βάση δεδομένων για χρέωση.

Βιβλιοθήκη

Μπορεί να υπάρχει μια βάση δεδομένων στη βιβλιοθήκη για να αποθηκεύονται τα στοιχεία όλων των βιβλίων στη βιβλιοθήκη και τα στοιχεία των αναγνωστών κ.λπ. Το σύστημα βάσης δεδομένων μπορεί επίσης να παρέχει μια υπηρεσία κρατήσεων, δηλαδή να επιτρέπει στους αναγνώστες να κάνουν κράτηση βιβλίων και όταν τα βιβλία μπορούν να δανειστούν, θα ειδοποιούνται μέσω ταχυδρομείου ή άλλου τρόπου. Το σύστημα μπορεί επίσης να στείλει υπενθυμίσεις σε αναγνώστες που έχουν δανειστεί βιβλία, ενημερώνοντάς τους ότι δεν έχουν επιστρέψει τα δανεισμένα βιβλία εντός της καθορισμένης προθεσμίας. Επιπλέον, όλες οι βιβλιοθήκες είναι πλέον εξοπλισμένες με ηλεκτρονικό σύστημα ευρετηρίου, το οποίο πραγματοποιεί την αυτόματη αναζήτηση των βιβλίων. Χρησιμοποιώντας αυτό το σύστημα, οι αναγνώστες μπορούν να μάθουν τη θέση του βιβλίου που θέλουν να δανειστούν και αν έχουν δανειστεί αναζητώντας πληροφορίες όπως ο τίτλος, ο συγγραφέας ή το έτος έκδοσης.

Αγορά ασφάλισης

Κάθε φορά που ένας ασφαλιστικός μεσίτης θέλει να συνάψει ένα ασφαλιστήριο συμβόλαιο, όπως ασφάλιση ζωής, ασφάλιση ιδιοκτητών σπιτιού ή ασφάλιση αυτοκινήτου, ένας ασφαλιστικός μεσίτης έχει πρόσβαση στις βάσεις δεδομένων πολλών ασφαλιστικών φορέων. Το σύστημα βάσης δεδομένων καθορίζει το ποσό της ασφάλισης με βάση τα προσωπικά στοιχεία του πελάτη που παρέχονται, όπως όνομα, διεύθυνση κατοικίας, ηλικία κ.λπ. Οι μεσίτες ασφαλίσεων μπορούν να αναζητήσουν πολλές βάσεις δεδομένων για να βρουν αυτή που προσφέρει τα περισσότερα οφέλη στους πελάτες τους.

Online αγορές

Οι ιστότοποι αγορών που χρησιμοποιούμε συνήθως καθοδηγούνται από εφαρμογές βάσης δεδομένων, όπως Taobao, Jingdong, Dangdang κ.λπ.

Ενώ υπάρχουν πολλά ηλεκτρονικά βιβλιοπωλεία, επιτρέπει στους πελάτες να περιηγούνται και να αγοράζουν βιβλία σε διάφορες κατηγορίες (π.χ. υπολογιστές ή διαχείριση). Μπορείτε επίσης να περιηγηθείτε σε βιβλία με βάση το όνομα του συγγραφέα. Σε όλες τις περιπτώσεις, υπάρχει μια βάση δεδομένων στον web server του οργανισμού με όλα τα στοιχεία του βιβλίου (τίτλος, συγγραφέας, τιμή, εγγραφές πωλήσεων, εκδότης, σύνοψη και λεπτομερής περιγραφή κ.λπ.), καθώς και διαθεσιμότητα, παράδοση βιβλίων κ.λπ. Κατάσταση, διαθεσιμότητα και πληροφορίες παραγγελίας.

Τα βιβλία μπορούν να διασταυρωθούν στη βάση δεδομένων.

Για παράδειγμα, ένα βιβλίο μπορεί να παρατίθεται σε πολλές κατηγορίες, όπως Υπολογιστές, Γλώσσες Προγραμματισμού, Μπεστ σέλερ και Προτεινόμενα Βιβλία ταυτόχρονα.

Μπορούμε να αγοράσουμε ένα ή περισσότερα βιβλία online.

Το Dangdang.com θα αποθηκεύσει τα αρχεία προηγούμενων συναλλαγών (ιστορικές παραγγελίες) μέσω της βάσης δεδομένων, συμπεριλαμβανομένων στοιχείων όπως ο τίτλος του βιβλίου που αγοράστηκε, η διεύθυνση παράδοσης και η τιμή που καταβλήθηκε, για να παρέχει εξατομικευμένες υπηρεσίες στους χρήστες του ιστότοπου.

Το Πανεπιστήμιο

Εάν σπουδάσετε σε πανεπιστήμιο, το σχολείο θα διαθέτει ένα σύστημα βάσης δεδομένων (σύστημα διαχείρισης μαθητών) που αποθηκεύει όλες τις πληροφορίες σχετικά με τους φοιτητές, συμπεριλαμβανομένων των εγγεγραμμένων μαθημάτων, των διαφόρων υποτροφιών που ελήφθησαν, των μαθημάτων που επιλέχθηκαν τα προηγούμενα έτη και των μαθημάτων που επιλέχθηκαν φέτος και των βαθμολογιών εξετάσεων πληροφορίες κ.λπ. Μπορεί επίσης να υπάρχει μια βάση δεδομένων που χρησιμοποιείται για την αποθήκευση στοιχείων εργαζομένων που εργάζονται στο πανεπιστήμιο. Τα παραπάνω είναι κοινές εφαρμογές βάσεων δεδομένων στη ζωή και σίγουρα θα συναντήσετε άλλα συστήματα βάσεων δεδομένων. Αν και είμαστε εξοικειωμένοι με αυτές τις εφαρμογές σήμερα, τα συστήματα βάσεων δεδομένων είναι στην πραγματικότητα πολύ περίπλοκες τεχνολογίες που έχουν αναπτυχθεί για περισσότερα από εξήντα χρόνια.

Ποιοι είναι οι τύποι βάσεων δεδομένων;

Το DBMS (Σύστημα Διαχείρισης Βάσεων Δεδομένων) ταξινομείται κυρίως με βάση τη μορφή αποθήκευσης δεδομένων και σε αυτό το στάδιο χωρίζεται κυρίως στους παρακάτω τύπους.

Ιεραρχική βάση δεδομένων (HDB)

Η ιεραρχική βάση δεδομένων είναι το παλαιότερο επιτυχώς αναπτυγμένο σύστημα βάσης δεδομένων, το οποίο εκφράζει δεδομένα μέσω ιεραρχικής δομής (δομή δέντρου). Οι ιεραρχικές βάσεις δεδομένων ήταν το κύριο ρεύμα των βάσεων δεδομένων, αλλά με την εμφάνιση και τη δημοτικότητα των σχεσιακών βάσεων δεδομένων, χρησιμοποιούνται πλέον σπάνια.

Μια πιο αντιπροσωπευτική ιεραρχική βάση δεδομένων είναι η βάση δεδομένων IMS (Information Management System), η οποία αναπτύχθηκε με επιτυχία από την IBM.

Σχεσιακή βάση δεδομένων (RDB)

Οι σχεσιακές βάσεις δεδομένων είναι οι πιο ευρέως χρησιμοποιούμενες βάσεις δεδομένων σήμερα. Η σχεσιακή βάση δεδομένων γεννήθηκε το 1969 και έχει μακρά

ιστορία. Όπως τα φύλλα εργασίας του Excel, οι σχεσιακές βάσεις δεδομένων χρησιμοποιούν δισδιάστατους πίνακες που αποτελούνται από γραμμές και στήλες για τη διαχείριση δεδομένων, ώστε να είναι εύκολα κατανοητές. Ταυτόχρονα, χρησιμοποιεί επίσης SQL (Structured Query Language, Structured Query Language) για τη λειτουργία των δεδομένων.

Η παραδοσιακή σχεσιακή βάση δεδομένων υιοθετεί τη μέθοδο αποθήκευσης των πινάκων και τα δεδομένα αποθηκεύονται με τη μορφή γραμμών και στηλών, κάτι που είναι πολύ βολικό για ανάγνωση και αναζήτηση. Ο παρακάτω πίνακας είναι ορισμένα δεδομένα σε σχεσιακή βάση δεδομένων.

Αριθμός προϊόντος	Ονομασία προϊόντος	Είδη αγαθών	τιμή μονάδας	Τιμή μονάδας αγοράς	Ημερομηνία Εγγραφής
0001	Κοντομάνικη μπλούζα	είδη ένδυσης	100	50	2019-10-9
0002	τρύπα	Προμήθειες γραφείου	500	320	5-1-2019
0003	Αθλητικό μπλουζάκι	είδη ένδυσης	400	280	8-3-2019
0004	κουζινομάχαιρο	ΕΙΔΗ ΚΟΥΖΙΝΑΣ	50	30	2019-2-14
0005	χύτρα ταχύτητας	ΕΙΔΗ ΚΟΥΖΙΝΑΣ	200	100	2019-2-13
0006	πιρούνι	ΕΙΔΗ ΚΟΥΖΙΝΑΣ	10	5	2019-2-13
0007	επιφάνεια κοπής	ΕΙΔΗ ΚΟΥΖΙΝΑΣ	80	70	2019-2-13
0008	στυλό στυλό	Προμήθειες γραφείου	10	5	2019-2-13

Οι πιο αντιπροσωπευτικές σχεσιακές βάσεις δεδομένων είναι οι Oracle Database, SQL Server, DB2, PostgreSQL και MySQL .

Βάση δεδομένων προσανατολισμένη στα έγγραφα

Η βάση δεδομένων εγγράφων είναι εμπνευσμένη από το λογισμικό γραφείου Lotus Notes και είναι παρόμοια με την τελευταία βάση δεδομένων αποθήκευσης κλειδιού-τιμής. Η βάση δεδομένων εγγράφων μπορεί να θεωρηθεί ως μια αναβαθμισμένη έκδοση της βάσης δεδομένων κλειδιού-τιμής, επιτρέποντας ένθετα κλειδιά-τιμές. Επιπλέον, η αποτελεσματικότητα ερωτημάτων της βάσης δεδομένων εγγράφων είναι υψηλότερη από αυτή της βάσης δεδομένων κλειδιού-τιμής.

Οι βάσεις δεδομένων προσανατολισμένες στα έγγραφα αποθηκεύουν δεδομένα με τη μορφή εγγράφων. Κάθε έγγραφο είναι μια συλλογή από μια σειρά στοιχείων δεδομένων. Κάθε στοιχείο δεδομένων έχει ένα όνομα και μια αντίστοιχη τιμή, η οποία μπορεί να είναι απλοί τύποι δεδομένων, όπως συμβολοσειρές, αριθμοί και

ημερομηνίες ή σύνθετοι τύποι, όπως ταξινομημένες λίστες και συσχετισμένα αντικείμενα.

Η μικρότερη μονάδα αποθήκευσης δεδομένων είναι ένα έγγραφο. Τα χαρακτηριστικά του εγγράφου που είναι αποθηκευμένα στον ίδιο πίνακα μπορεί να είναι διαφορετικά. Τα δεδομένα μπορούν να αποθηκευτούν σε XML, JSON ή JSONB και άλλες μορφές.

Οι αντιπροσωπευτικές βάσεις δεδομένων που προσανατολίζονται σε έγγραφα περιλαμβάνουν τις MongoDB και CouchDB.

Βάση δεδομένων προσανατολισμένη στη στήλη

Οι βάσεις δεδομένων Columnstore αποθηκεύουν δεδομένα σε οικογένειες στηλών, μια οικογένεια στηλών που χρησιμοποιείται για την αποθήκευση σχετικών δεδομένων για τα οποία συχνά υποβάλλονται ερωτήματα μαζί. Για παράδειγμα, αν είχαμε τάξη Προσώπων, συχνά αναζητούσαμε το όνομα και την ηλικία τους μαζί αντί για τον μισθό. Σε αυτήν την περίπτωση, το όνομα και η ηλικία τοποθετούνται σε μια οικογένεια στήλης και ο μισθός σε μια άλλη.

Οι βάσεις δεδομένων αποθήκευσης στηλών χρησιμοποιούνται συνήθως για την αντιμετώπιση τεράστιων ποσοτήτων δεδομένων σε κατανεμημένη αποθήκευση. Οι αντιπροσωπευτικές βάσεις δεδομένων αποθήκευσης στηλών είναι η Cassandra και η HBase .

Βάση δεδομένων XML (Βάση δεδομένων XML, XMLDB)

Μια βάση δεδομένων XML είναι ένα σύστημα διαχείρισης δεδομένων που υποστηρίζει λειτουργίες όπως η αποθήκευση και η αναζήτηση εγγράφων σε XML (μια εφαρμογή κάτω από την Standard Universal Markup Language). Στο σύστημα, οι προγραμματιστές μπορούν να αναζητήσουν, να εξάγουν και να σειριοποιήσουν τα έγγραφα XML στη βάση δεδομένων στην καθορισμένη μορφή.

Βάση δεδομένων Key-Value Store (Key-Value Store, KVS)

Μια βάση δεδομένων αποθήκευσης κλειδιού-τιμής είναι μια βάση δεδομένων που χρησιμοποιείται για την αποθήκευση του συνδυασμού του πρωτεύοντος κλειδιού

(Κλειδί) και της τιμής (Τιμή) που χρησιμοποιείται από το ερώτημα. Οι αναγνώστες με γνώση γλωσσών προγραμματισμού μπορούν να το θεωρήσουν ως συσχετιστικό πίνακα ή κατακερματισμό.

Τα τελευταία χρόνια, οι βάσεις δεδομένων καταστημάτων με αξία κλειδιού κερδίζουν την προσοχή καθώς εφαρμόζονται σε υπηρεσίες Ιστού όπως η Google που απαιτούν εξαιρετικά υψηλής ταχύτητας αναζήτηση μεγάλων ποσοτήτων δεδομένων.

Σχεδιασμός Βάσεων Δεδομένων

Ο σχεδιασμός βάσης δεδομένων είναι η διαδικασία κατασκευής του καλύτερου μοντέλου αποθήκευσης δεδομένων για αυτό το επιχειρηματικό σύστημα σύμφωνα με τις ειδικές ανάγκες του επιχειρηματικού συστήματος, σε συνδυασμό με τη βάση δεδομένων που επιλέγουμε, για τη δημιουργία της δομής του πίνακα και της σχέσης διαχείρισης μεταξύ πινάκων και πινάκων. Για να μπορεί να αποθηκεύει αποτελεσματικά τα δεδομένα της εφαρμογής και να έχει αποτελεσματική πρόσβαση στα αποθηκευμένα δεδομένα.

Ο σχεδιασμός βάσεων δεδομένων είναι ένα σημαντικό μέρος των συστημάτων βάσεων δεδομένων. Μια καλή βάση δεδομένων μπορεί να φέρει σαφή στατιστικά δεδομένα και λεπτομερή ανάλυση δεδομένων στο σύστημα και να φέρει εύχρηστα και διαισθητικά δεδομένα στο σύστημα. Η κακή σχεδίαση της βάσης δεδομένων θα προκαλέσει αναπόφευκτα πολλά προβλήματα, που κυμαίνονται από την προσθήκη ή την αφαίρεση πεδίων έως την αδυναμία εκτέλεσης του συστήματος.

Στα προηγούμενα κεφάλαια, δημιουργήσαμε ως επί το πλείστον βάσεις δεδομένων, πίνακες και εισάγομε δεδομένα δοκιμών απευθείας σύμφωνα με τις επιχειρηματικές απαιτήσεις και, στη συνέχεια, χειριστήκαμε τα δεδομένα.

Γιατί λοιπόν ξαφνικά να τονίσουμε πρώτα το σχέδιο και μετά να δημιουργήσουμε τη βάση δεδομένων και τον πίνακα;

Ο λόγος είναι απλός, παρομοιάζουμε τη βάση δεδομένων με ένα κτίριο, αν φτιάξεις μια καλύβα ή ένα απλό μπανγκαλόου, δεν υπάρχει αμφιβολία ότι κανείς δεν θα πληρώσει κάποιον για να σχεδιάσει το μοτίβο του σπιτιού. Ωστόσο, εάν ένας

κατασκευαστής ακινήτων θέλει να αναπτύξει ένα νέο κτίριο και να χτίσει μια οικιστική περιοχή με πολλά κτίρια, σίγουρα θα ζητήσει από κάποιον να σχεδιάσει τα σχέδια κατασκευής πριν από την κατασκευή.

Με τον ίδιο τρόπο, στην πραγματική ανάπτυξη του έργου, εάν το σύστημα έχει μεγάλη ποσότητα αποθήκευσης δεδομένων, έχουν σχεδιαστεί πολλοί πίνακες και η σχέση μεταξύ των πινάκων είναι πολύπλοκη, είναι απαραίτητο να σχεδιαστεί η βάση δεδομένων με τυποποιημένο τρόπο και στη συνέχεια να δημιουργηθούν βάσεις δεδομένων, τραπέζια κλπ. Εργασία.

Είτε δημιουργείτε έναν δυναμικό ιστότοπο είτε δημιουργείτε μια εφαρμογή παραθύρου επιφάνειας εργασίας, η σημασία του σχεδιασμού της βάσης δεδομένων δεν μπορεί να υπερτονιστεί. Μόνο ο καλός σχεδιασμός της βάσης δεδομένων μπορεί να βελτιώσει την απόδοση του συστήματος και να παρέχει καλύτερες υπηρεσίες. Ο κακός σχεδιασμός της βάσης δεδομένων θα προκαλέσει πολλά προβλήματα, επηρεάζοντας την αποδοτικότητα της εργασίας μας, την αποδοτικότητα των υπηρεσιών και την αποδοτικότητα των χρηστών.

Ο κακός σχεδιασμός της βάσης δεδομένων εκδηλώνεται με τους ακόλουθους τρόπους:

- Αναποτελεσματική πρόσβαση στα δεδομένα.
- Υπάρχει πολύς πλεονασμός δεδομένων, σπαταλώντας χώρο αποθήκευσης
- Πολλά ζητήματα προκύπτουν κατά την ενημέρωση και την ανάκτηση δεδομένων

Ο καλός σχεδιασμός της βάσης δεδομένων εκδηλώνεται στις ακόλουθες πτυχές:

- Υψηλή απόδοση πρόσβασης
- Μειώστε τον πλεονασμό δεδομένων, εξοικονομήστε χώρο αποθήκευσης και διευκολύνετε την περαιτέρω επέκταση
- Μπορεί να διευκολύνει την ανάπτυξη εφαρμογών

Ο σχεδιασμός του πίνακα στη βάση δεδομένων είναι επίσης ένας σημαντικός παράγοντας για τον προσδιορισμό της αποτελεσματικότητας του συστήματος

βάσης δεδομένων. Ο σχεδιασμός πίνακα είναι η διαδικασία σχεδιασμού και δόμησης των οντοτήτων δεδομένων και των σχέσεων μεταξύ των οντοτήτων δεδομένων στη βάση δεδομένων.

Βασικά βήματα του σχεδιασμού της βάσης δεδομένων.

Πριν κατανοήσουμε τα βήματα σχεδιασμού της βάσης δεδομένων, ας ρίξουμε μια ματιά στον κύκλο ανάπτυξης ενός έργου λογισμικού, ως εξής:

- Ανάλυση ζήτησης
- Σχεδιασμός περιγράμματος
- Λογικός Σχεδιασμός/Λεπτομέρεια Σχεδίαση
- Γραφή κώδικα
- Δοκιμή λογισμικού

Στάδια σχεδιασμού βάσης δεδομένων:



Ο σχεδιασμός βάσης δεδομένων είναι η διαδικασία σχεδιασμού της δομής της βάσης δεδομένων και δημιουργίας βάσης δεδομένων σε ένα συγκεκριμένο σύστημα διαχείρισης, σύμφωνα με τις ανάγκες των χρηστών.

Όταν η βάση δεδομένων είναι πολύπλοκη (όπως μεγάλος όγκος δεδομένων, πολλοί πίνακες και πολύπλοκες επιχειρηματικές σχέσεις), πρέπει πρώτα να σχεδιάσουμε τη βάση δεδομένων, επειδή ένας καλός σχεδιασμός βάσης δεδομένων μπορεί να εξοικονομήσει χώρο αποθήκευσης δεδομένων, να διασφαλίσει την ακεραιότητα των δεδομένων και να διευκολύνει την εφαρμογή της βάσης δεδομένων συστήματα, ανάπτυξη.

Εκπαιδευτική υποενότητα : 8.2 Διασύνδεση εφαρμογών με σχεσιακή βάση δεδομένων
Διαχείριση Βάσης Δεδομένων

Το σύστημα διαχείρισης βάσεων δεδομένων (Database Management System, DBMS) είναι ένα λογισμικό μεγάλης κλίμακας για το χειρισμό και τη διαχείριση βάσεων δεδομένων, και χρησιμοποιείται για τη δημιουργία, τη χρήση και τη συντήρηση βάσεων δεδομένων. Διενεργεί ενοποιημένη διαχείριση και έλεγχο της βάσης δεδομένων για να διασφαλίσει την ασφάλεια και την ακεραιότητα της βάσης δεδομένων. Οι χρήστες έχουν πρόσβαση σε δεδομένα στη βάση δεδομένων μέσω dbms και οι διαχειριστές βάσεων δεδομένων πραγματοποιούν επίσης συντήρηση βάσης δεδομένων μέσω dbms. Παρέχει μια ποικιλία λειτουργιών που επιτρέπουν σε πολλές εφαρμογές και χρήστες να δημιουργούν, να τροποποιούν και να υποβάλλουν ερωτήματα σε βάσεις δεδομένων με διαφορετικούς τρόπους την ίδια στιγμή ή σε διαφορετικές χρονικές στιγμές. Επιτρέπει στους χρήστες να ορίζουν και να χειρίζονται εύκολα δεδομένα, να διατηρούν την ασφάλεια και την ακεραιότητα των δεδομένων και να εκτελούν ταυτόχρονο έλεγχο και ανάκτηση βάσης δεδομένων σε πολλούς χρήστες.

Παράδειγμα σχεσιακής βάσης δεδομένων

Για παράδειγμα, λάβετε υπόψη δύο πίνακες που χρησιμοποιεί μια μικρή επιχείρηση για την επεξεργασία παραγγελιών προϊόντων. Ο πρώτος πίνακας περιέχει πληροφορίες για πελάτες: κάθε καταχώριση σε αυτόν περιλαμβάνει το όνομα και τη διεύθυνση του πελάτη, πληροφορίες πληρωμής και παράδοσης, αριθμό τηλεφώνου κ.λπ. Κάθε στοιχείο πληροφοριών (χαρακτηριστικό) τοποθετείται σε ξεχωριστή στήλη της βάσης δεδομένων, η οποία είναι εκχωρήθηκε ένα μοναδικό αναγνωριστικό (κλειδί) για κάθε σειρά. Στον δεύτερο πίνακα (με πληροφορίες

παραγγελίας), κάθε εγγραφή περιέχει το αναγνωριστικό του πελάτη που έκανε την παραγγελία, το όνομα του προϊόντος που παραγγέλθηκε, την ποσότητα, το μέγεθος ή το χρώμα του κ.λπ. Οι εγγραφές σε αυτόν τον πίνακα δεν περιέχουν δεδομένα όπως π.χ. το όνομα του πελάτη ή τα στοιχεία επικοινωνίας του.

Και οι δύο πίνακες έχουν μόνο ένα κοινό στοιχείο - το αναγνωριστικό στήλης (κλειδί). Έχοντας αυτήν την κοινή στήλη, οι σχεσιακές βάσεις δεδομένων μπορούν να δημιουργήσουν σχέσεις μεταξύ δύο πινάκων. Όταν η εφαρμογή επεξεργασίας παραγγελίας υποβάλλει μια παραγγελία στη βάση δεδομένων, η βάση δεδομένων αποκτά πρόσβαση στον πίνακα πληροφοριών παραγγελίας, ανακτά τις πληροφορίες προϊόντος και χρησιμοποιεί το αναγνωριστικό πελάτη από αυτόν τον πίνακα για να βρει τις πληροφορίες πληρωμής και αποστολής στον πίνακα πληροφοριών παραγγελίας. Στη συνέχεια επιλέγεται το επιθυμητό προϊόν στην αποθήκη, ο πελάτης παραλαμβάνει έγκαιρα την παραγγελία του και πραγματοποιεί την πληρωμή.

Δομή σχεσιακών βάσεων δεδομένων

Το σχεσιακό μοντέλο συνεπάγεται μια λογική δομή δεδομένων: πίνακες, προβολές και ευρετήρια. Η λογική δομή είναι διαφορετική από τη φυσική δομή αποθήκευσης. Αυτός ο διαχωρισμός επιτρέπει στους διαχειριστές να διαχειρίζονται το φυσικό σύστημα αποθήκευσης χωρίς να αλλάζουν τα δεδομένα που περιέχονται στη λογική δομή. Για παράδειγμα, η αλλαγή του ονόματος ενός αρχείου βάσης δεδομένων δεν θα επηρεάσει τους πίνακες που είναι αποθηκευμένοι σε αυτό.

Ο διαχωρισμός μεταξύ του φυσικού και του λογικού επιπέδου επεκτείνεται σε λειτουργίες, οι οποίες είναι καλά καθορισμένες ενέργειες σε δομές δεδομένων και βάσης δεδομένων. Οι λογικές λειτουργίες επιτρέπουν στις εφαρμογές να ορίζουν τις απαιτήσεις περιεχομένου, ενώ οι φυσικές λειτουργίες καθορίζουν τον τρόπο πρόσβασης στα δεδομένα και την εκτέλεση μιας εργασίας.

Για να διασφαλιστεί η ακρίβεια και η διαθεσιμότητα των δεδομένων, οι σχεσιακές βάσεις δεδομένων πρέπει να ακολουθούν ορισμένους κανόνες ακεραιότητας. Για παράδειγμα, οι κανόνες ακεραιότητας μπορούν να αποτρέψουν τη χρήση

διπλότυπων σειρών σε πίνακες για την εξάλειψη της πιθανότητας εισαγωγής εσφαλμένων πληροφοριών στη βάση δεδομένων.

Σχεσιακό μοντέλο

Στις πρώτες βάσεις δεδομένων, τα δεδομένα κάθε εφαρμογής αποθηκεύονταν σε ξεχωριστή, μοναδική δομή. Εάν ένας προγραμματιστής ήθελε να δημιουργήσει μια εφαρμογή για να χρησιμοποιήσει τέτοια δεδομένα, έπρεπε να γνωρίζει καλά τη συγκεκριμένη δομή για να βρει τα απαραίτητα δεδομένα. Αυτή η μέθοδος οργάνωσης ήταν αναποτελεσματική, δύσκολη στη διατήρηση και καθιστούσε δύσκολη τη βελτιστοποίηση της απόδοσης της εφαρμογής. Το σχεσιακό μοντέλο σχεδιάστηκε για να εξαλείψει την ανάγκη για μια ποικιλία δομών δεδομένων.

Παρείχε έναν τυπικό τρόπο παρουσίασης δεδομένων και αποστολής ερωτημάτων που θα μπορούσαν να χρησιμοποιηθούν σε οποιαδήποτε εφαρμογή. Οι προγραμματιστές έχουν μάθει ότι οι πίνακες είναι ένα βασικό πλεονέκτημα των σχεσιακών βάσεων δεδομένων, καθώς παρέχουν έναν διαισθητικό, αποτελεσματικό και ευέλικτο τρόπο αποθήκευσης και πρόσβασης σε δομημένες πληροφορίες.

Με τον καιρό, καθώς οι προγραμματιστές άρχισαν να χρησιμοποιούν τη δομημένη γλώσσα ερωτημάτων (SQL) για να γράφουν δεδομένα στη βάση δεδομένων και να στέλνουν ερωτήματα, ένα άλλο πλεονέκτημα του σχεσιακού μοντέλου έγινε εμφανές. Για πολλά χρόνια, η SQL χρησιμοποιείται ευρέως ως γλώσσα ερωτημάτων σε βάσεις δεδομένων. Βασίζεται σε αλγόριθμους σχεσιακής άλγεβρας και σαφή μαθηματική δομή, η οποία εξασφαλίζει απλότητα και αποτελεσματικότητα στη βελτιστοποίηση οποιωνδήποτε ερωτημάτων βάσης δεδομένων. Για σύγκριση: όταν χρησιμοποιείτε άλλες προσεγγίσεις, πρέπει να δημιουργήσετε ξεχωριστά, μοναδικά ερωτήματα.

Οφέλη ενός συστήματος διαχείρισης σχεσιακών βάσεων δεδομένων

Εταιρείες όλων των τύπων και μεγεθών χρησιμοποιούν ένα απλό αλλά ισχυρό σχεσιακό μοντέλο για την εξυπηρέτηση μιας ποικιλίας αναγκών πληροφοριών. Οι σχεσιακές βάσεις δεδομένων χρησιμοποιούνται για την παρακολούθηση του αποθέματος, την επεξεργασία διαδικτυακών εμπορικών συναλλαγών, τη διαχείριση μεγάλων όγκων κρίσιμων δεδομένων πελατών και πολλά άλλα. Οι σχεσιακές βάσεις δεδομένων μπορούν να προταθούν για την εξυπηρέτηση οποιωνδήποτε αναγκών πληροφοριών όπου τα στοιχεία δεδομένων είναι διασυνδεδεμένα και είναι απαραίτητο να διασφαλιστεί η ασφαλής και αξιόπιστη διαχείρισή τους με βάση τους κανόνες ακεραιότητας.

Οι σχεσιακές βάσεις δεδομένων εμφανίστηκαν τη δεκαετία του 1970. Μέχρι σήμερα, τα πλεονεκτήματα της σχεσιακής προσέγγισης την έχουν καταστήσει το πιο ευρέως χρησιμοποιούμενο μοντέλο βάσης δεδομένων στον κόσμο.

Σχεσιακό μοντέλο και ακεραιότητα δεδομένων

Το σχεσιακό μοντέλο διατηρεί αποτελεσματικότερα την ακεραιότητα των δεδομένων σε εφαρμογές και αντίγραφα βάσεων δεδομένων (στιγμιότυπα). Για παράδειγμα, όταν ένας πελάτης καταθέτει χρήματα σε έναν λογαριασμό χρησιμοποιώντας ένα ATM και στη συνέχεια ελέγχει το υπόλοιπο σε κινητό τηλέφωνο, αναμένει ότι τα χρήματα θα εμφανιστούν αμέσως στον λογαριασμό. Οι σχεσιακές βάσεις δεδομένων είναι εξαιρετικές για τη διατήρηση της ακεραιότητας των δεδομένων σε πολλές περιπτώσεις βάσεων δεδομένων ταυτόχρονα.

Άλλοι τύποι βάσεων δεδομένων δεν μπορούν να διατηρήσουν ταυτόχρονα την ακεραιότητα μεγάλων ποσοτήτων δεδομένων. Ορισμένοι σύγχρονοι τύποι βάσεων δεδομένων, όπως η NoSQL, παρέχουν μόνο τη λεγόμενη «απόλυτη ακεραιότητα». Αυτό σημαίνει ότι όταν τα δεδομένα κλιμακώνονται ή πολλοί χρήστες χρησιμοποιούν τα ίδια δεδομένα ταυτόχρονα, χρειάζεται λίγος χρόνος για να γίνουν αλλαγές. Σε ορισμένες περιπτώσεις, η απόλυτη ακεραιότητα είναι καλή (για παράδειγμα, η ενημέρωση στοιχείων σε έναν κατάλογο προϊόντων), αλλά για κρίσιμες επιχειρηματικές λειτουργίες (για παράδειγμα, συναλλαγές με καλάθι

αγορών), οι σχεσιακές βάσεις δεδομένων εξακολουθούν να είναι το χρυσό πρότυπο.

Διάπραξη αλλαγών και ατομικότητας

Οι σχεσιακές βάσεις δεδομένων έχουν πολύ λεπτομερείς και αυστηρούς επιχειρηματικούς κανόνες και πολιτικές σχετικά με την πραγματοποίηση αλλαγών στη βάση δεδομένων (δηλαδή την μόνιμη αποθήκευση αλλαγών στα δεδομένα). Σκεφτείτε, για παράδειγμα, μια βάση δεδομένων αποθήκης που παρακολουθεί τρία μέρη που χρησιμοποιούνται πάντα σε ένα κιτ. Όταν ένα από αυτά αφαιρεθεί από το απόθεμα, πρέπει να αφαιρεθούν και τα άλλα δύο. Εάν ένα από τα τρία μέρη δεν είναι διαθέσιμο, τα άλλα δύο δεν μπορούν να πωληθούν χωριστά, πράγμα που σημαίνει ότι και τα τρία μέρη πρέπει να είναι διαθέσιμα για να γίνουν αλλαγές στη βάση δεδομένων. Η σχεσιακή βάση δεδομένων δεν θα επιτρέψει την αποθήκευση αλλαγών εάν δεν ισχύουν και για τα τρία ανταλλακτικά. Αυτό το χαρακτηριστικό των σχεσιακών βάσεων δεδομένων ονομάζεται ατομικότητα ή συνέχεια.

Αποθηκευμένες διαδικασίες και σχεσιακές βάσεις δεδομένων

Η πρόσβαση στα δεδομένα περιλαμβάνει πολλές επαναλαμβανόμενες δραστηριότητες. Για παράδειγμα, μερικές φορές ένα απλό ερώτημα για τη λήψη πληροφοριών από έναν πίνακα χρειάζεται να επαναληφθεί εκατό ή χιλιάδες φορές για να ληφθεί το επιθυμητό αποτέλεσμα. Για τέτοια σενάρια πρόσβασης σε βάσεις δεδομένων, απαιτείται κάποιο είδος κώδικα προγραμματισμού. Θα ήταν κουραστικό για τους προγραμματιστές να γράφουν τυπικό κώδικα πρόσβασης δεδομένων για μια νέα εφαρμογή κάθε φορά. Ευτυχώς, οι σχεσιακές βάσεις δεδομένων υποστηρίζουν αποθηκευμένες διαδικασίες, οι οποίες είναι μπλοκ κώδικα που μπορούν να προσπελαστούν με μια απλή κλήση από τον κώδικα εφαρμογής. Για παράδειγμα, η ίδια αποθηκευμένη διαδικασία μπορεί να χρησιμοποιηθεί για τη διαδοχική επισήμανση εγγραφών για διευκόλυνση του χρήστη για διαφορετικές εφαρμογές.

Κλείδωμα βάσης δεδομένων και συγχρονισμός

Όταν πολλοί χρήστες ή εφαρμογές προσπαθούν να αλλάξουν τα ίδια δεδομένα ταυτόχρονα, αυτό μπορεί να οδηγήσει σε διενέξεις βάσης δεδομένων. Οι κλειδαριές και η ταυτόχρονη λειτουργία μειώνουν την πιθανότητα διενέξεων και συμβάλλουν στη διατήρηση της ακεραιότητας των δεδομένων.

Το κλείδωμα εμποδίζει άλλους χρήστες και εφαρμογές να έχουν πρόσβαση στα δεδομένα κατά την ενημέρωσή τους. Σε ορισμένες βάσεις δεδομένων, ένα κλείδωμα μπορεί να εφαρμοστεί σε έναν ολόκληρο πίνακα, το οποίο μπορεί να επηρεάσει αρνητικά την απόδοση της εφαρμογής. Σε άλλους τύπους βάσεων δεδομένων, όπως οι σχεσιακές βάσεις δεδομένων Oracle, το κλείδωμα εκτελείται σε επίπεδο μίας εγγραφής, αφήνοντας διαθέσιμες άλλες εγγραφές στον πίνακα. Αυτή η προσέγγιση βοηθά στη διατήρηση της αποτελεσματικότητας της εφαρμογής.

Το εργαλείο ταυτόχρονης χρήσης χρησιμοποιείται όταν πολλοί χρήστες ή εφαρμογές προσπαθούν να υποβάλουν ερωτήματα στην ίδια βάση δεδομένων ταυτόχρονα. Παρέχει πρόσβαση χρήστη και εφαρμογών στη βάση δεδομένων σύμφωνα με τις πολιτικές ελέγχου.

Χαρακτηριστικά που πρέπει να ληφθούν υπόψη κατά την επιλογή μιας σχεσιακής βάσης δεδομένων

Το λογισμικό που χρησιμοποιείται για την αποθήκευση, διαχείριση, ανάκτηση και αναζήτηση δεδομένων σε μια βάση δεδομένων ονομάζεται σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων (RDBMS). Το RDBMS παρέχει μια διεπαφή μεταξύ των χρηστών και των εφαρμογών και της βάσης δεδομένων, καθώς και διοικητικές λειτουργίες για τη διαχείριση της αποθήκευσης δεδομένων, της αποτελεσματικότητας και της πρόσβασης.

Υπάρχουν διάφοροι παράγοντες που πρέπει να λάβετε υπόψη κατά την επιλογή ενός τύπου βάσης δεδομένων και προϊόντων σχεσιακής βάσης δεδομένων. Η επιλογή του RDBMS εξαρτάται από τις ανάγκες της εταιρείας σας. Κάντε στον εαυτό σας τις παρακάτω ερωτήσεις.

- Ποιες είναι οι απαιτήσεις μας για την ακρίβεια των δεδομένων; Θα χρησιμοποιήσουμε επιχειρηματική λογική για να αποθηκεύσουμε και να διασφαλίσουμε την ακρίβεια των δεδομένων; Τα δεδομένα μας υπόκεινται σε πιο αυστηρές απαιτήσεις ακρίβειας (για παράδειγμα, εάν εργάζεστε με οικονομικά δεδομένα και αναφορές);
- Χρειαζόμαστε επεκτασιμότητα; Πόσα δεδομένα πρέπει να διαχειριστείτε και ποια είναι η προβλεπόμενη ανάπτυξη αυτών των όγκων; Πρέπει το μοντέλο της βάσης δεδομένων να υποστηρίζει αντίγραφα κατοπτρισμού (ως ξεχωριστές παρουσίες) προκειμένου να κλιμακωθεί; Εάν ναι, θα είμαστε σε θέση να διατηρήσουμε την ακεραιότητα των δεδομένων σε αυτές τις περιπτώσεις;
- Πόσο σημαντικός είναι ο συγχρονισμός; Θα χρειαστεί οι χρήστες και οι εφαρμογές να έχουν πρόσβαση σε δεδομένα ταυτόχρονα; Υποστηρίζει το λογισμικό της βάσης δεδομένων ταυτόχρονη πρόσβαση χωρίς να θυσιάζεται η ασφάλεια;
- Ποιες είναι οι ανάγκες μας για αποτελεσματικότητα και αξιοπιστία της βάσης δεδομένων; Χρειαζόμαστε ένα σύστημα υψηλής απόδοσης και αξιοπιστίας; Ποιες είναι οι απαιτήσεις για την ταχύτητα εκτέλεσης ερωτήματος; Ποιες εγγυήσεις παρέχει ο πάροχος υπηρεσιών σύμφωνα με τη συμφωνία επιπέδου υπηρεσιών (SLA) ή σε περίπτωση απρογραμμάτιστου χρόνου διακοπής λειτουργίας;

Σχεσιακή βάση δεδομένων του μέλλοντος: Αυτόνομη βάση δεδομένων

Τα τελευταία χρόνια, οι σχεσιακές βάσεις δεδομένων έχουν βελτιωθεί ως προς την απόδοση, την αξιοπιστία και την ασφάλεια και έχουν γίνει ευκολότερες στη συντήρηση. Ωστόσο, η δομή τους γινόταν όλο και πιο περίπλοκη, και ως αποτέλεσμα, η διαχείριση μιας τέτοιας βάσης δεδομένων άρχισε να απαιτεί σημαντική προσπάθεια. Αντί να χρησιμοποιούν τις δεξιότητές τους για την ανάπτυξη καινοτόμων εφαρμογών που θα αποφέρουν κέρδη στην εταιρεία, οι προγραμματιστές αναγκάζονται να αφιερώσουν τη μερίδα του λέοντος από τον χρόνο τους στη διαχείριση της βάσης δεδομένων για τη βελτιστοποίηση της αποτελεσματικότητάς της.

Οι αυτόνομες τεχνολογίες σήμερα βασίζονται στη δύναμη του σχεσιακού μοντέλου, της τεχνολογίας βάσεων δεδομένων cloud και της μηχανικής μάθησης .ώστε να μπορείτε να δημιουργήσετε έναν νέο τύπο σχεσιακής βάσης δεδομένων. Μια βάση δεδομένων αυτοοδήγησης (αναφέρεται επίσης ως βάση δεδομένων αυτοοδήγησης) διατηρεί όλα τα πλεονεκτήματα και τις δυνατότητες του σχεσιακού μοντέλου και προσθέτει εργαλεία τεχνητής νοημοσύνης, μηχανικής μάθησης και αυτοματισμού για την παρακολούθηση και τη βελτιστοποίηση της ταχύτητας ερωτημάτων και διαχείρισης. Για παράδειγμα, για τη βελτίωση της απόδοσης των ερωτημάτων, η αυτόνομη βάση δεδομένων κάνει προβλέψεις και ελέγχει ευρετήρια και, στη συνέχεια, εφαρμόζει τα καλύτερα αποτελέσματα στην πράξη, όλα αυτά χωρίς τη συμμετοχή του διαχειριστή. Οι αυτοδιαχειριζόμενες βάσεις δεδομένων κάνουν συνεχώς τέτοιες βελτιώσεις στη δική τους λειτουργία χωρίς ανθρώπινη παρέμβαση.

Οι αυτόνομες τεχνολογίες επιτρέπουν στους προγραμματιστές να μην χάνουν πλέον χρόνο σε εργασίες ρουτίνας συντήρησης. Για παράδειγμα, δεν είναι πλέον απαραίτητο να καθοριστούν εκ των προτέρων οι απαιτήσεις υποδομής. Με μια λύση IaaS, μισθώνετε πόρους όπως υπολογιστική ισχύ ή αποθηκευτικό χώρο, έχετε πρόσβαση στους πόρους που χρειάζεστε όπως τους χρειάζεστε και πληρώνετε μόνο για αυτούς που χρησιμοποιεί η εταιρεία σας. Οι προγραμματιστές μπορούν να δημιουργήσουν αυτόνομες σχεσιακές βάσεις δεδομένων σε λίγα μόλις βήματα, επιταχύνοντας τη διαδικασία ανάπτυξης εφαρμογών.

Οι Ρόλοι του Διαχειριστή Βάσης Δεδομένων

Ένα ΣΔΒΔ, όπως και οποιοδήποτε πληροφοριακό σύστημα, απαιτεί καθημερινή διαχείριση προκειμένου να εξασφαλιστεί η σωστή λειτουργία του. Η ανάγκη αυτή είναι αυξημένη στην περίπτωση των ΒΔ καθώς αυτές χρησιμοποιούνται από περισσότερα από ένα άτομα και εφαρμογές, αποτελούν δηλαδή μεριζόμενους πόρους.

Για το σκοπό αυτό, στα τμήματα πληροφορικής προβλέπεται ο ρόλος του Διαχειριστή Βάσης Δεδομένων (Database Administrator ή DBA για συντομία).

Ανάλογα με το πλήθος, το μέγεθος και την πολυπλοκότητα των ΒΔ, ο ρόλος του Διαχειριστή ΒΔ μπορεί να ανατεθεί σε κάποιον παράλληλα με άλλα καθήκοντα (π.χ. διαχειριστής συστήματος, διαχειριστής δικτύου) ή να αποτελεί τα αποκλειστικά του καθήκοντα. Επίσης, δεν είναι σπάνια η περίπτωση ο ρόλος του Διαχειριστή ΒΔ να ανατίθεται από κοινού σε περισσότερα από ένα άτομα.

Οι Διαχειριστές ΒΔ είναι από τις πιο περιζήτητες και καλά αμειβόμενες θέσεις στο χώρο της Πληροφορικής, ειδικά στο εξωτερικό. Αυτό οφείλεται στη σπουδαιότητα του έργου του Διαχειριστή ΒΔ, καθώς και στις εξειδικευμένες γνώσεις και δεξιότητες που πρέπει να κατέχει ένας Διαχειριστής ΒΔ.

Εγκατάσταση και Παραμετροποίηση του ΣΔΒΔ

Πριν χρησιμοποιηθεί μια ΒΔ πρέπει να εγκατασταθεί το αντίστοιχο ΣΔΒΔ. Η εργασία αυτή αποτελεί ένα από τα βασικά καθήκοντα του Διαχειριστή ΒΔ. Η εγκατάσταση συνήθως πραγματοποιείται σε ισχυρούς υπολογιστές που αναλαμβάνουν το ρόλο του εξυπηρετητή ΒΔ (database server).

Το ΣΔΒΔ μπορεί να είναι το μοναδικό λογισμικό (μαζί με το λειτουργικό σύστημα) που θα εγκατασταθεί ή να συνυπάρχει με άλλες εφαρμογές. Η εγκατάσταση του ΣΔΒΔ μπορεί να γίνει και σε λιγότερο ισχυρούς υπολογιστές για τη δημιουργία περιβάλλοντος ανάπτυξης και ελέγχου από τους προγραμματιστές ΒΔ.

Τα αναγκαία αρχεία για την εγκατάσταση του ΣΔΒΔ προσφέρονται από τον κατασκευαστή του ΣΔΒΔ. Στο παρελθόν, τα αρχεία ήταν διαθέσιμα σε οπτικούς δίσκους (CD ή DVD), ενώ σήμερα, χάρη στις υψηλές ταχύτητες σύνδεσης στο Διαδίκτυο, είναι διαθέσιμα για μεταφόρτωση (download) από την ιστοσελίδα του κατασκευαστή.

Για κάθε λειτουργικό σύστημα (Windows, UNIX, Linux, Mac OS) υπάρχει διαφορετική έκδοση των αρχείων εγκατάστασης.

Η διαδικασία εγκατάστασης μπορεί να είναι χειροκίνητη (manual) ή αυτόματη (automatic). Στη χειροκίνητη εγκατάσταση ο Διαχειριστής ΒΔ εργάζεται σε περιβάλλον γραμμής εντολών (command line) και ακολουθώντας τις οδηγίες του

κατασκευαστή του ΣΔΒΔ πραγματοποιεί όλες τις απαραίτητες εργασίες. Η μέθοδος αυτή δεν είναι ιδιαίτερα δημοφιλής εξαιτίας της πολυπλοκότητας της.

Στην αυτόματη εγκατάσταση οι απαραίτητες εργασίες γίνονται με ελάχιστη παρέμβαση του Διαχειριστή ΒΔ. Για την αυτόματη εγκατάσταση σε περιβάλλον MS Windows και Mac OS είναι διαθέσιμα προγράμματα εγκατάστασης (installers) που όταν εκτελεστούν ξεκινάει ένας οδηγός εγκατάστασης (installation wizard) σε περιβάλλον γραμμής εντολών (command line) ή σε γραφικό περιβάλλον (GUI – graphical user interface).

Ο οδηγός εγκατάστασης αρχικά δίνει τη δυνατότητα στον Διαχειριστή ΒΔ να επιλέξει τις βασικές παραμέτρους εγκατάστασης και στη συνέχεια πραγματοποιούνται αυτόματα οι απαραίτητες εργασίες.

Σε περιβάλλον UNIX και Linux η αυτόματη εγκατάσταση πραγματοποιείται μέσω ειδικών δικτυακών τόπων που ονομάζονται αποθετήρια λογισμικού (software repositories). Ο Διαχειριστής ΒΔ συνδέεται στο αποθετήριο λογισμικού και επιλέγει τα πακέτα λογισμικού (software packages) του ΣΔΒΔ για μεταφόρτωση. Με την ολοκλήρωση της μεταφόρτωσης ξεκινάνε αυτόματα οι αναγκαίες εργασίες εγκατάστασης.

Κατά την εγκατάσταση του ΣΔΒΔ, χειροκίνητη ή αυτόματα, πραγματοποιείται η αρχική παραμετροποίηση του ΣΔΒΔ, δηλαδή η ρύθμιση διαφόρων παραμέτρων που σχετίζονται με τη λειτουργία του ΣΔΒΔ. Η παραμετροποίηση συνήθως αφορά τα ακόλουθα:

Φάκελοι εγκατάστασης:

Καθορισμός φακέλων για τα αρχεία εφαρμογών (προγράμματα που αποτελούν το ΣΔΒΔ) και τα αρχεία δεδομένων (περιεχόμενα της ΒΔ). Είναι συνηθισμένη πρακτική ο φάκελος για τα αρχεία δεδομένων να βρίσκεται σε διαφορετικό διαμέρισμα (partition) του δίσκου ή σε διαφορετικό δίσκο από το λειτουργικό σύστημα και το ΣΔΒΔ.

Με αυτό τον τρόπο αν γίνει επανεγκατάσταση του λειτουργικού συστήματος ή του ΣΔΒΔ τα περιεχόμενα της ΒΔ δεν θα επηρεαστούν. Επίσης αν τα αρχεία δεδομένων βρίσκονται σε άλλο δίσκο βελτιώνεται η ταχύτητα της ΒΔ.

Τύπος ΒΔ:

Ανάλογα με τον σκοπό και την προβλεπόμενη χρήση της ΒΔ μπορεί να οριστεί ο ανάλογος τύπος ΒΔ που θα δημιουργείται από την εντολή CREATE DATABASE (αναλύεται σε επόμενη ενότητα). Η επιλογή τύπου ΒΔ καθορίζει συγκεκριμένες παραμέτρους της εγκατάστασης, όπως την κύρια μνήμη που θα δεσμεύει το ΣΔΒΔ, το μέγιστο μέγεθος που μπορεί να έχει η ΒΔ, τον τρόπο αποθήκευσης και οργάνωσης των αρχείων δεδομένων με τα περιεχόμενα της ΒΔ. Σε ορισμένα ΣΔΒΔ η επιλογή τύπου ΒΔ οδηγεί σε επιλογή κατάλληλης μηχανής ΒΔ (database engine). Η μηχανή ΒΔ είναι το τμήμα του ΣΔΒΔ υπεύθυνο για την πρόσβαση στα δεδομένα της ΒΔ, δηλαδή υλοποιεί τις λειτουργίες της δημιουργίας, ανάγνωσης, ενημέρωσης και διαγραφής.

ΣΔΒΔ ως υπηρεσία:

Υπάρχει δυνατότητα να δημιουργηθεί υπηρεσία (service) του λειτουργικού συστήματος ώστε το ΣΔΒΔ να ξεκινάει αυτόματα κατά την εκκίνηση του υπολογιστή που είναι εγκατεστημένο. Εάν το ΣΔΒΔ χρησιμοποιείται σε παραγωγικό περιβάλλον τότε η αυτόματα εκκίνηση είναι επιβεβλημένη. Αντίθετα, εάν το ΣΔΒΔ χρησιμοποιείται για ανάπτυξη και έλεγχο εφαρμογών ΒΔ μπορεί να μην είναι επιθυμητή η αυτόματη εκκίνηση.

Λογαριασμός Διαχειριστή ΒΔ:

Κάθε ΣΔΒΔ πρέπει να έχει τουλάχιστον έναν χρήστη με δικαιώματα Διαχειριστή ΒΔ ο οποίος να έχει απεριόριστο έλεγχο στο ΣΔΒΔ και τις ΒΔ. Ο χρήστης αυτός δημιουργείται αυτόματα κατά την εγκατάσταση του ΣΔΒΔ. Κάθε ΣΔΒΔ δημιουργεί λογαριασμό Διαχειριστή ΒΔ με συγκεκριμένο όνομα χρήστη (user name) και κωδικό τα οποία είναι γνωστά σε όλους καθώς αναφέρονται στις οδηγίες του κατασκευαστή. Γι' αυτό συστήνεται να ορίζεται για τον λογαριασμό του Διαχειριστή

ΒΔ διαφορετικό όνομα και κωδικός από αυτά που προτείνονται κατά την εγκατάσταση.

Σύνολο χαρακτήρων (character set):

Εάν στη ΒΔ θέλουμε να αποθηκευτούν δεδομένα που δεν χρησιμοποιούν το λατινικό αλφάβητο (π.χ. Ελληνικά), πρέπει να δηλωθεί κατά την εγκατάσταση του ΣΔΒΔ το κατάλληλο σύνολο χαρακτήρων. Αυτό είναι απαραίτητο προκειμένου το ΣΔΒΔ να μπορεί να χειριστεί σωστά τα δεδομένα, δηλαδή να μπορεί να τα εμφανίσει, να τα ταξινομήσει και να πραγματοποιεί αναζητήσεις.

Επικοινωνία:

Οι χρήστες και οι εφαρμογές που χρησιμοποιούν μια ΒΔ μπορεί να μην βρίσκονται στον ίδιο υπολογιστή με το ΣΔΒΔ. Σε αυτή την περίπτωση η επικοινωνία με τη ΒΔ είναι απομακρυσμένη και υλοποιείται μέσω του πρωτοκόλλου δικτύου TCP/IP. Κατά την εγκατάσταση του ΣΔΒΔ μπορούν να ρυθμιστούν οι λεπτομέρειες της απομακρυσμένης επικοινωνίας (π.χ. θύρα TCP/IP που θα χρησιμοποιείται) ή να απενεργοποιηθεί η δυνατότητα απομακρυσμένης επικοινωνίας αν δεν προβλέπονται απομακρυσμένοι χρήστες της ΒΔ.

Αρχεία καταγραφής (log files):

Υπάρχει δυνατότητα το ΣΔΒΔ να καταγράφει στοιχεία για τη χρήση και λειτουργία της ΒΔ. Τα περιεχόμενα των αρχείων καταγραφής χρησιμοποιούνται από το Διαχειριστή ΒΔ για την παρακολούθηση της λειτουργίας της ΒΔ, τη βελτίωση της απόδοσής της και την επίλυση προβλημάτων.

Βοηθητικά εργαλεία:

Εκτός από το ΣΔΒΔ μπορούν να εγκατασταθούν πρόσθετα προγράμματα για τη διαχείριση της ΒΔ ή την ανάπτυξη εφαρμογών ΒΔ.

Αντίγραφα ασφαλείας (backup) και αναπαραγωγή (replication):

Κατά την εγκατάσταση του ΣΔΒΔ μπορεί να οριστεί χρονοδιάγραμμα αυτόματης λήψης αντιγράφων ασφαλείας της ΒΔ. Επίσης υπάρχει δυνατότητα να οριστεί

σχέδιο αναπαραγωγής της ΒΔ, για την αυτόματη δημιουργία αντιγράφων της ΒΔ σε άλλους υπολογιστές.

Τα αντίγραφα είναι συγχρονισμένα, δηλαδή οι αλλαγές στα περιεχόμενα της ΒΔ μεταφέρονται αυτόματα στα αντίγραφα (ρεπλίκες) της ΒΔ που βρίσκονται στους άλλους υπολογιστές.

Επιπλέον παραμετροποίηση μπορεί να γίνει και μετά την εγκατάσταση του ΣΔΒΔ χρησιμοποιώντας εργαλεία που προσφέρει το ΣΔΒΔ ή εκτελώντας ξανά τον οδηγό εγκατάστασης και επιλέγοντας τροποποίηση της εγκατάστασης.

Επίσης μπορεί να γίνει απευθείας αλλαγή στα περιεχόμενα των αρχείων ρυθμίσεων (configuration files) του ΣΔΒΔ που δημιουργήθηκαν κατά την εγκατάσταση. Η τελευταία επιλογή απαιτεί ιδιαίτερη προσοχή και αυξημένες γνώσεις καθώς υπάρχει κίνδυνος καταστροφής της ΒΔ από λανθασμένες ενέργειες

Ορισμός του Σχήματος της ΒΔ

Πριν δημιουργηθούν πίνακες σε μια ΒΔ πρέπει να δημιουργηθεί η ΒΔ με την εντολή CREATE DATABASE. Για παράδειγμα η εκτέλεση της εντολής

```
CREATE DATABASE mathitologio;
```

θα δημιουργήσει μια κενή ΒΔ με όνομα mathitologio.

Η εντολή CREATE DATABASE μπορεί να περιλαμβάνει παραμέτρους που θα καθορίζουν τεχνικές λεπτομέρειες της ΒΔ, π.χ. φάκελοι αποθήκευσης των αρχείων με τα περιεχόμενα της ΒΔ, μέγιστο μέγεθος αρχείων της ΒΔ, σύνολο χαρακτήρων που χρησιμοποιείται για αναπαράσταση των δεδομένων της ΒΔ και τρόπος ταξινόμησης των δεδομένων, χρήση αρχείων καταγραφής (log files).

Αν η εντολή δεν περιλαμβάνει παραμέτρους τότε η ΒΔ δημιουργείται με τις προκαθορισμένες τιμές που δηλώθηκαν κατά την εγκατάσταση του ΣΔΒΔ.

Στο ΣΔΒΔ MySQL, κατά τη δημιουργία μιας ΒΔ μπορεί να καθοριστεί το σύνολο χαρακτήρων που χρησιμοποιούν τα δεδομένα (παράμετρος CHARACTER SET) και ο τρόπος ταξινόμησης για το επιλεγμένο σύνολο χαρακτήρων (παράμετρος COLLATE).

Για παράδειγμα, η εντολή: `CREATE DATABASE mathitologio CHARACTER SET greek COLLATE greek_general_ci;` θα δημιουργήσει μια κενή ΒΔ με όνομα `mathitologio`, τα δεδομένα της οποίας μπορούν να περιλαμβάνουν και Ελληνικούς χαρακτήρες (επιπλέον των λατινικών) σύμφωνα με την κωδικοποίηση “ISO 8859-7 Greek” και θα ταξινομούνται σύμφωνα με τη σειρά των γραμμάτων στο Ελληνικό αλφάβητο.

Εναλλακτικά μπορεί να χρησιμοποιηθεί κωδικοποίηση Unicode που υποστηρίζει όλα τα διαθέσιμα αλφάβητα (άρα και το ελληνικό), όπως φαίνεται στο ακόλουθο παράδειγμα.

```
CREATE DATABASE mathitologio CHARACTER SET utf8 COLLATE utf8_general_ci;
```

Ο λογαριασμός χρήστη με δικαιώματα Διαχειριστή ΒΔ που δημιουργείται κατά την εγκατάσταση του ΣΔΒΔ έχει πάντα τη δυνατότητα να δημιουργεί ΒΔ. Για λόγους ασφάλειας συνήθως η δυνατότητα δημιουργίας ΒΔ δεν δίνεται σε άλλους χρήστες. Αφού δημιουργηθεί η ΒΔ, μπορούν με τις κατάλληλες εντολές SQL να δημιουργηθούν πίνακες.

Ο λογαριασμός χρήστη με δικαιώματα Διαχειριστή ΒΔ που δημιουργείται κατά την εγκατάσταση του ΣΔΒΔ έχει πάντα αυτή τη δυνατότητα. Σε περιβάλλον ανάπτυξης και ελέγχου εφαρμογών ΒΔ η δυνατότητα αυτή μπορεί να δοθεί και στα μέλη της ομάδας ανάπτυξης.

Σε παραγωγικό περιβάλλον όμως που η ΒΔ περιέχει πραγματικά δεδομένα και υποστηρίζει κάποια λειτουργία ή παρέχει κάποια υπηρεσία, αυτή τη δυνατότητα έχει συνήθως μόνο ο Διαχειριστής ΒΔ για λόγους ασφάλειας. Σε αυτή την περίπτωση ο Διαχειριστής ΒΔ θα πρέπει να εκτελέσει τις κατάλληλες εντολές SQL, ακολουθώντας τις οδηγίες της ομάδας ανάπτυξης, για να δημιουργήσει πίνακες στη ΒΔ. Σε ορισμένα ΣΔΒΔ, μια ΒΔ μπορεί να περιέχει ένα ή περισσότερα σχήματα. Ένα σχήμα (schema) ομαδοποιεί πίνακες της ΒΔ, π.χ. ανά χρήστη ή εφαρμογή που χρησιμοποιεί τη ΒΔ, με σκοπό την καλύτερη διαχείριση και μεγαλύτερη ασφάλεια των περιεχομένων της ΒΔ. Για τη δημιουργία σχήματος χρησιμοποιείται η εντολή `CREATE SCHEMA` η οποία συνοδεύεται από εντολές SQL για τη δημιουργία πινάκων της ΒΔ. Αυτό είναι ιδιαίτερα χρήσιμο σε ΒΔ που περιλαμβάνουν μεγάλο πλήθος πινάκων.

Δημιουργία Χρηστών και Απόδοση Ρόλων

Κάθε άτομο ή εφαρμογή που θέλει να χρησιμοποιήσει μια ΒΔ πρέπει πρώτα να συνδεθεί στο ΣΔΒΔ χρησιμοποιώντας ένα λογαριασμό χρήστη (user account).

Οι συγκεκριμένοι λογαριασμοί είναι διαφορετικοί από τους λογαριασμούς του λειτουργικού συστήματος. Υπάρχουν ΣΔΒΔ που μπορούν να συνδέσουν τους δύο τύπους λογαριασμών, ώστε ο χρήστης να μην χρειάζεται να ταυτοποιηθεί δύο φορές, μία από το λειτουργικό σύστημα και μία από το ΣΔΒΔ, για να χρησιμοποιήσει τη ΒΔ.

Η δημιουργία λογαριασμών χρήστη στο ΣΔΒΔ είναι από τις σπουδαιότερες αρμοδιότητες του Διαχειριστή ΒΔ καθώς από τη σωστή διαχείριση των λογαριασμών εξαρτάται η ασφάλεια της ΒΔ.

Ο λογαριασμός χρήστη με δικαιώματα Διαχειριστή ΒΔ που δημιουργείται κατά την εγκατάσταση του ΣΔΒΔ είναι αρχικά ο μοναδικός που μπορεί να δημιουργήσει άλλους λογαριασμούς. Η δημιουργία λογαριασμών χρήστη υλοποιείται με διαφορετικό τρόπο από κάθε ΣΔΒΔ, με αποτέλεσμα η σύνταξη των σχετικών εντολών να διαφέρει. Στο ΣΔΒΔ MySQL χρησιμοποιείται η εντολή CREATE USER.

Για παράδειγμα η εντολή :

```
CREATE USER 'g_pappas';
```

δημιουργεί ένα λογαριασμό με όνομα χρήστη g_pappas. Οι λογαριασμοί χρήστη μπορούν να διαθέτουν κωδικό πρόσβασης. Στο ΣΔΒΔ MySQL, αν στην εντολή CREATE USER χρησιμοποιηθεί η παράμετρος IDENTIFIED BY μπορεί να οριστεί κωδικός πρόσβασης. Για παράδειγμα η εντολή

```
CREATE USER 'k_aggelou' IDENTIFIED BY 'p@$$';
```

δημιουργεί ένα λογαριασμό με όνομα χρήστη k_aggelou και κωδικό πρόσβασης p@\$\$.

Τα στοιχεία των λογαριασμών χρήστη αποθηκεύονται σε ειδικό πίνακα που δημιουργείται αυτόματα κατά την εγκατάσταση του ΣΔΒΔ. Στο ΣΔΒΔ MySQL αυτός ο πίνακας ονομάζεται mysql.user. Με την εντολή : SELECT * FROM mysql.USER;

θα εμφανιστούν οι λεπτομέρειες όλων των λογαριασμών που έχουν δημιουργηθεί. Θα παρατηρήσετε ότι οι κωδικοί πρόσβασης είναι ακατάληπτοι. Αυτό συμβαίνει επειδή οι κωδικοί που ορίζονται με την εντολή CREATE USER ... IDENTIFIED BY ...; κωδικοποιούνται πριν αποθηκευτούν στον πίνακα mysql.user. Έτσι ακόμα κι αν ένα μη εξουσιοδοτημένο άτομο αποκτήσει πρόσβαση στον πίνακα mysql.user δεν θα μπορέσει να ανακτήσει τους κωδικούς.

Για να αλλάξουμε τον κωδικό ενός λογαριασμού ή να ορίσουμε κωδικό σε περίπτωση που αυτό δεν είχε γίνει κατά τη δημιουργία του λογαριασμού, στο ΣΔΒΔ MySQL χρησιμοποιούμε την εντολή SET PASSWORD. Για παράδειγμα η εντολή:

```
SET PASSWORD FOR 'a_dimou' = PASSWORD('0p3n');
```

ορίζει κωδικό πρόσβασης 0p3n για τον λογαριασμό με όνομα χρήστη a_dimou. Η συνάρτηση PASSWORD(...) κρυπτογραφεί τον κωδικό πρόσβασης πριν αποθηκευτεί στον πίνακα mysql.user με τα στοιχεία των λογαριασμών.

Υπάρχει δυνατότητα αλλαγής του ονόματος ενός λογαριασμού. Αυτό συνήθως γίνεται για λόγους ασφάλειας. Για παράδειγμα, στο ΣΔΒΔ MySQL το προτεινόμενο όνομα του λογαριασμού με δικαιώματα Διαχειριστή ΒΔ που δημιουργείται κατά την εγκατάσταση του ΣΔΒΔ είναι root.

Αν χρησιμοποιηθεί αυτό το όνομα τότε μειώνεται η ασφάλεια του συστήματος καθώς από τα δύο συστατικά του λογαριασμού (όνομα και κωδικός) το ένα είναι ήδη γνωστό (όνομα) και το μόνο που χρειάζεται κάποιος εισβολέας είναι να μαντέψει το άλλο (κωδικός). Στο ΣΔΒΔ MySQL η μετονομασία ενός λογαριασμού γίνεται με την εντολή RENAME USER. Για παράδειγμα η εντολή

```
RENAME USER 'root' TO 'db_admin';
```

θα μετονομάσει τον λογαριασμό με όνομα root σε db_admin.

Υπάρχουν περιπτώσεις που ένας λογαριασμός πρέπει να καταργηθεί, π.χ. επειδή το άτομο που τον χρησιμοποιούσε μετακινήθηκε σε άλλη θέση ή αποχώρησε.

Για τον σκοπό αυτό χρησιμοποιείται η εντολή DROP USER. Για παράδειγμα η εντολή

```
DROP USER 'm_alexiou';
```

θα διαγράψει από το ΣΔΒΔ το λογαριασμό με όνομα χρήστη m_alexίου.

Οι ενέργειες που μπορεί να εκτελέσει ένας λογαριασμός χρήστη του ΣΔΒΔ καθορίζονται από τα δικαιώματα που διαθέτει ο λογαριασμός. Στην ορολογία των ΒΔ, τα δικαιώματα αυτά ονομάζονται προνόμια (privileges).

Στην ορολογία των ΒΔ, όταν δίνεται ένα προνόμιο λέμε ότι το προνόμιο εκχωρείται (grant) και όταν αφαιρείται λέμε ότι το προνόμιο ανακαλείται (revoke).

Ένα προνόμιο μπορεί να είναι καθολικό, δηλαδή να αφορά όλες τις ΒΔ δεδομένων του ΣΔΒΔ. Εναλλακτικό το προνόμιο μπορεί να είναι για συγκεκριμένη ΒΔ του ΣΔΒΔ, για συγκεκριμένο πίνακα μιας ΒΔ ή για συγκεκριμένη στήλη ενός πίνακα.

Ένας λογαριασμός χρήστη μπορεί να εκχωρήσει σε άλλο λογαριασμό μόνο προνόμια που ήδη κατέχει.

Ο λογαριασμός χρήστη με δικαιώματα Διαχειριστή ΒΔ που δημιουργείται κατά την εγκατάσταση του ΣΔΒΔ κατέχει όλα τα διαθέσιμα προνόμια σε όλα τα επίπεδα και στη συνέχεια μπορεί να τα εκχωρήσει στους λογαριασμούς που θα δημιουργηθούν.

Στο ΣΔΒΔ MySQL η εκχώρηση προνομίου γίνεται με την εντολή GRANT, η οποία έχει την ακόλουθη σύνταξη: GRANT < privilege> ON <level> TO <user> ;

όπου το όνομα του προνομίου, το επίπεδο στο οποίο εφαρμόζεται το προνόμιο και το όνομα του λογαριασμού χρήστη στον οποίο εκχωρείται το προνόμιο.

Το επίπεδο εφαρμογής του προνομίου μπορεί να λάβει τις ακόλουθες τιμές:

. για καθολικό προνόμιο που αφορά όλες τις ΒΔ του ΣΔΒΔ,

db.* για προνόμιο στη ΒΔ με όνομα db,

db.table για προνόμιο στον πίνακα με όνομα table που βρίσκεται μέσα στη ΒΔ με όνομα db,

db.table(column) για προνόμιο στη στήλη με όνομα column του πίνακα με όνομα table που βρίσκεται στη ΒΔ με όνομα db.

Με μία εντολή GRANT μπορούν να εκχωρηθούν περισσότερα από ένα προνόμια σε περισσότερους από έναν λογαριασμούς. Σε αυτή την περίπτωση η εντολή

περιλαμβάνει λίστες από προνόμια και λίστες από χρήστες, με τα στοιχεία κάθε λίστας να διαχωρίζονται με κόμματα.

Για να ανακαλέσει ένας χρήστης προνόμια από κάποιον άλλο χρήστη θα πρέπει να διαθέτει το προνόμιο GRANT OPTION καθώς και τα προνόμια που προσπαθεί να ανακαλέσει.

Καθορισμός Μεθόδων Προσπέλασης

Ο τρόπος που γίνεται η προσπέλαση των περιεχομένων μιας ΒΔ όταν υποβάλλονται ερωτήματα έχει μεγάλη σημασία, καθώς επηρεάζει το χρόνο απόκρισης της ΒΔ, δηλαδή το χρόνο που απαιτείται για να εμφανιστούν τα αποτελέσματα ενός ερωτήματος. Καθώς τα περισσότερα ερωτήματα αφορούν συνήθως ανάκτηση δεδομένων, είναι σημαντικό ο χρόνος εκτέλεσης των εντολών SELECT να μην είναι μεγάλος.

Υπάρχουν περιπτώσεις που για να εκτελεστεί μια εντολή SELECT το ΣΔΒΔ πρέπει να διαβάσει κάθε γραμμή ενός πίνακα προκειμένου να επιλέξει τις γραμμές που θα επιστραφούν ως αποτέλεσμα του ερωτήματος. Αυτό συμβαίνει όταν η εντολή SELECT: περιέχει συνθήκη (WHERE),

περιέχει συναρτήσεις για εύρεση μέγιστης (MAX) και ελάχιστης (MIN) τιμής,

απαιτεί ταξινόμηση των αποτελεσμάτων (ORDER BY),

απαιτεί ομαδοποίηση των αποτελεσμάτων (GROUP BY).

Για παράδειγμα, στη ΒΔ mathitologio υπάρχει ο πίνακας Μάθημα(Όνομα, Ώρες) με τις εβδομαδιαίες ώρες διδασκαλίας κάθε μαθήματος.

Για να εμφανίσουμε τα μαθήματα που διδάσκονται 4 ώρες την εβδομάδα, θα υποβάλουμε το ακόλουθο ερώτημα:

```
SELECT *  
FROM Μάθημα  
WHERE Ώρες=4;
```

Για το συγκεκριμένο ερώτημα, το ΣΔΒΔ θα διαβάσει κάθε γραμμή του πίνακα Μάθημα και για κάθε γραμμή θα ελέγξει αν η τιμή της στήλης Ώρες είναι 4.

Όσες γραμμές ικανοποιούν αυτή τη συνθήκη θα συμπεριληφθούν στα αποτελέσματα τους ερωτήματος.

Συνεχίζοντας το προηγούμενο παράδειγμα, αν θέλουμε να βρούμε το μέγιστο αριθμό ωρών που διδάσκεται ένα μάθημα θα υποβάλουμε το ακόλουθο ερώτημα:

```
SELECT MAX(Ώρες)  
FROM Μάθημα;
```

Για το συγκεκριμένο ερώτημα, το ΣΔΒΔ θα διαβάσει από κάθε γραμμή του πίνακα Μάθημα την τιμή της στήλης Ώρες ώστε να βρει τη μέγιστη τιμή της στήλης.

Γίνεται λοιπόν κατανοητό ότι όσο περισσότερες γραμμές περιέχει ένας πίνακας τόσο αυξάνεται ο χρόνος εκτέλεσης μιας εντολής SELECT. Αν ο χρόνος αυτός δεν είναι αποδεκτός από τους χρήστες της ΒΔ τότε η ΒΔ δεν θα χρησιμοποιηθεί.

Φανταστείτε για παράδειγμα τι θα συμβεί αν η ΒΔ που εξυπηρετεί τα ΑΤΜ μιας τράπεζας χρειάζεται μερικά λεπτά (αντί για δευτερόλεπτα) για να επιστρέψει αποτελέσματα σχετικά με το υπόλοιπο ενός λογαριασμού.

Ο χρόνος εκτέλεσης μιας εντολής SELECT μπορεί να μειωθεί αν χρησιμοποιηθεί ευρετήριο (index). Η έννοια του ευρετηρίου είναι γνωστή καθώς χρησιμοποιείται και σε τομείς εκτός της Πληροφορικής.

Για παράδειγμα, αρκετά βιβλία στο τέλος έχουν ευρετήριο όρων που αποτελείται από ζεύγη λέξεων - αριθμών σελίδας. Έτσι αν κάποιος ψάχνει τις σελίδες που αναφέρονται σε κάποιο θέμα δεν χρειάζεται να διαβάσει ολόκληρο το βιβλίο από την πρώτη μέχρι την τελευταία σελίδα, αλλά μπορεί να συμβουλευτεί το ευρετήριο και να μεταφερθεί στην αντίστοιχη σελίδα.

Με ανάλογο τρόπο στις ΒΔ μπορούν να δημιουργηθούν ευρετήρια για τις τιμές που περιέχει μία ή περισσότερες στήλες ενός πίνακα ώστε να επιταχύνονται ερωτήματα που αφορούν τις συγκεκριμένες στήλες.

Η δημιουργία ευρετηρίου γίνεται με την εντολή CREATE INDEX. Για παράδειγμα, η εντολή

```
CREATE INDEX Ωρες_idx  
ON Μάθημα (Ωρες);
```

θα δημιουργήσει ένα ευρετήριο για τη στήλη Ωρες του πίνακα Μάθημα. Έτσι εντολές SELECT που αφορούν τη στήλη Ωρες θα εκτελούνται πιο γρήγορα καθώς το ΣΔΒΔ θα χρησιμοποιεί το ευρετήριο αντί να ελέγχει τις τιμές της στήλης Ωρες για κάθε γραμμή του πίνακα.

Ένα ευρετήριο μπορεί να αφορά περισσότερες από μία στήλες ενός πίνακα.

Για παράδειγμα, στη ΒΔ mathitologio υπάρχει ο πίνακας Μαθητής(AM, Επώνυμο, Όνομα, Πατρώνυμο, Ημερομηνία_Γέννησης, Φύλο, Πόλη) με τα στοιχεία των μαθητών. Με την εντολή

```
CREATE INDEX ΕπώνυμοΌνομα_idx  
ON Μαθητής (Επώνυμο, Όνομα);
```

θα δημιουργήσουμε ένα ευρετήριο για τις στήλες Επώνυμο και Όνομα. Το ευρετήριο αυτό θα επιταχύνει εντολές SELECT που αφορούν μόνο τη στήλη Επώνυμο ή το συνδυασμό των στηλών Επώνυμο και Όνομα.

Εντολές SELECT όμως που αφορούν μόνο τη στήλη Όνομα δεν θα ωφεληθούν από το συγκεκριμένο ευρετήριο. Για να γίνει αυτό κατανοητό, έστω τα ακόλουθα ερωτήματα:

```
SELECT *  
FROM Μαθητής  
WHERE Επώνυμο='Παπαδόπουλος';  
  
SELECT *  
FROM Μαθητής  
WHERE Επώνυμο='Παπαδόπουλος' AND Όνομα='Ιωάννης';  
  
SELECT *  
FROM Μαθητής  
WHERE Όνομα='Ιωάννης';
```

Τα πρώτα δύο ερωτήματα θα αξιοποιήσουν το ευρετήριο για τις στήλες Επώνυμο και Όνομα, ενώ το τρίτο δεν θα μπορέσει να το χρησιμοποιήσει.

Σε ορισμένα ΣΔΒΔ υπάρχει δυνατότητα στην εντολή CREATE INDEX να οριστούν διάφορες παράμετροι, όπως για παράδειγμα η μέθοδος ταξινόμησης των περιεχομένων του ευρετηρίου (αύξουσα ή φθίνουσα), ο τρόπος οργάνωσης των περιεχομένων του ευρετηρίου (π.χ. B+ δέντρο, πίνακας κατακερματισμού), το μέγεθος και ο τρόπος αποθήκευσης των αρχείων που αποτελούν το ευρετήριο.

Η απόφαση για τη δημιουργία ευρετηρίων λαμβάνεται αρχικά από την ομάδα ανάπτυξης μιας εφαρμογής ΒΔ, με βάση τα ερωτήματα που θα υποβάλλονται στη ΒΔ. Επιπλέον ευρετήρια μπορούν να δημιουργηθούν κατά την παραγωγική λειτουργία μιας ΒΔ. Ειδικότερα, ο Διαχειριστής ΒΔ πρέπει να παρακολουθεί τη λειτουργία της ΒΔ για να εντοπίσει ερωτήματα με μεγάλους χρόνους εκτέλεσης (ο τρόπος που γίνεται αυτό εξηγείται σε επόμενη ενότητα). Σε αυτή την περίπτωση, ο Διαχειριστής ΒΔ ακολουθώντας τις οδηγίες της ομάδας ανάπτυξης θα δημιουργήσει τα κατάλληλα ευρετήρια και στη συνέχεια θα κάνει μετρήσεις για να επιβεβαιώσει ότι ο χρόνος απόκρισης της ΒΔ βελτιώθηκε μετά τη δημιουργία των ευρετηρίων. Είναι προφανές ότι για λόγους ασφάλειας, όταν η ΒΔ βρίσκεται σε παραγωγικό περιβάλλον και περιέχει πραγματικά δεδομένα, η δημιουργία ευρετηρίων πρέπει να γίνεται μόνο από τον Διαχειριστή ΒΔ.

Υπάρχουν περιπτώσεις που η ύπαρξη ενός ευρετηρίου δεν είναι επιθυμητή, π.χ. επειδή διαπιστώθηκε ότι το ευρετήριο δεν βελτίωσε τον χρόνο απόκρισης της ΒΔ ή επειδή δεν υποβάλλονται πλέον ερωτήματα που να αξιοποιούν το συγκεκριμένο ευρετήριο.

Η διαγραφή ενός ευρετηρίου γίνεται με την εντολή DROP INDEX. Για παράδειγμα, με την εντολή

```
DROP INDEX Ωρες_idx  
ON Μάθημα;
```

θα διαγραφεί το ευρετήριο στη στήλη Όρες του πίνακα Μάθημα που δημιουργήθηκε σε προηγούμενο παράδειγμα.

Η δημιουργία ευρετηρίων μειώνει το χρόνο εκτέλεσης των εντολών ανάκτησης γραμμών (SELECT), αλλά αυξάνει το χρόνο εκτέλεσης των εντολών εισαγωγής (INSERT), ενημέρωσης (UPDATE) και διαγραφής (DELETE) γραμμών καθώς εκτός από τον πίνακα πρέπει να ενημερωθούν και τα αντίστοιχα ευρετήρια.

Γι' αυτό ο Διαχειριστής ΒΔ θα πρέπει να παρακολουθεί τους χρόνους εκτέλεσης των ερωτημάτων και να διαχειρίζεται κατάλληλα τα ευρετήρια.

Εργασίες Συντήρησης και Αναβάθμισης

Η σημαντικότερη αρμοδιότητα ενός Διαχειριστή ΒΔ είναι η συνεχής παρακολούθηση και υποστήριξη της ΒΔ ώστε να εξασφαλιστεί η λειτουργία της ΒΔ χωρίς διακοπές και χωρίς προβλήματα.

Για την καλή λειτουργία της ΒΔ, ο Διαχειριστής ΒΔ είναι υπεύθυνος για τις ακόλουθες εργασίες, πολλές από τις οποίες πραγματοποιούνται σε καθημερινή βάση: έναρξη και τερματισμός ΣΔΒΔ, παρακολούθηση λειτουργίας ΣΔΒΔ και των ΒΔ, εντοπισμός και επίλυση προβλημάτων του ΣΔΒΔ και των ΒΔ, δημιουργία και επαναφορά αντιγράφων ασφάλειας, αναβάθμιση ΣΔΒΔ.

Ο τερματισμός του ΣΔΒΔ δεν πραγματοποιείται συχνά, ιδιαίτερα σε περιβάλλον παραγωγής όπου το ΣΔΒΔ φιλοξενεί ΒΔ με πραγματικά δεδομένα. Υπάρχουν περιπτώσεις που ο τερματισμός είναι αναγκαίος, όπως για παράδειγμα για την πραγματοποίηση εργασιών συντήρησης (π.χ. αναβάθμιση στο λογισμικό του ΣΔΒΔ, αναβάθμιση στο λειτουργικό σύστημα του υπολογιστή που είναι εγκατεστημένο του ΣΔΒΔ, μεταφορά του ΣΔΒΔ και των ΒΔ σε άλλο υπολογιστή), για την αντιμετώπιση προβλημάτων (π.χ. αντικατάσταση ελαττωματικών στοιχείων υλικού), για τη λήψη ή επαναφορά αντιγράφου ασφάλειας του ΣΔΒΔ και των ΒΔ που φιλοξενεί. Για λόγους ασφάλειας, σε περιβάλλον παραγωγής η εκκίνηση και ο τερματισμός του ΣΔΒΔ γίνεται μόνο από τον Διαχειριστή ΒΔ.

Ο τερματισμός του ΣΔΒΔ MySQL γίνεται με την εντολή `mysqld stop` και η εκκίνηση με την εντολή `mysql start`. Οι εντολές αυτές πρέπει να εκτελεστούν σε περιβάλλον

γραμμής εντολών του λειτουργικού συστήματος, στον φάκελο που έχει εγκατασταθεί το ΣΔΒΔ MySQL. Σε περιβάλλον Windows ο φάκελος εγκατάστασης του ΣΔΒΔ MySQL είναι ο C:\mysql\bin.

Οπότε η εντολή για τον τερματισμό του ΣΔΒΔ είναι:

```
C:\mysql\bin\mysqld stop
```

Αντίστοιχα για την εκκίνηση του ΣΔΒΔ η εντολή είναι:

```
C:\mysql\bin\mysqld
```

Σε περιβάλλον Linux ο φάκελος εγκατάστασης του ΣΔΒΔ MySQL είναι ο /etc/init.d.

Οπότε για τον τερματισμό του ΣΔΒΔ χρησιμοποιείται η εντολή:

```
/etc/init.d/mysqld stop
```

Αντίστοιχα για την εκκίνηση του ΣΔΒΔ χρησιμοποιείται η εντολή:

```
/etc/init.d/mysqld
```

Για να μπορέσει ένας χρήστης να τερματίσει το ΣΔΒΔ MySQL πρέπει να διαθέτει το καθολικό προνόμιο SHUTDOWN.

Μια από τις καθημερινές εργασίες του Διαχειριστή ΒΔ είναι η παρακολούθηση της λειτουργίας του ΣΔΒΔ προκειμένου να εντοπιστούν δυσλειτουργίες ή προβλήματα. Βασικό εργαλείο για την παρακολούθηση του ΣΔΒΔ είναι τα αρχεία καταγραφής.

Τα αρχεία καταγραφής (log files) του ΣΔΒΔ είναι αρχεία κειμένου που περιέχουν εγγραφές για διάφορα συμβάντα. Κάθε εγγραφή περιλαμβάνει ημερομηνία και ώρα του συμβάντος καθώς και μια σύντομη περιγραφή του συμβάντος.

Στο ΣΔΒΔ MySQL είναι διαθέσιμα τα ακόλουθα αρχεία καταγραφής:

- αρχείο καταγραφής σφαλμάτων (error log)
- αρχείο καταγραφής ερωτημάτων (query log)
- αρχείο καταγραφής αργών ερωτημάτων (slow query log)

Τα αρχεία καταγραφής βρίσκονται στον ίδιο φάκελο με τα αρχεία που περιέχουν τα δεδομένα των ΒΔ.

Το αρχείο καταγραφής σφαλμάτων έχει την κατάληξη .err. Στο αρχείο αυτό καταγράφονται πληροφορίες για την εκκίνηση ή τερματισμό του ΣΔΒΔ καθώς και κρίσιμα σφάλματα που παρουσιάστηκαν κατά τη λειτουργία του ΣΔΒΔ.

Ελέγχοντας τα περιεχόμενα του αρχείου ο Διαχειριστής ΒΔ μπορεί να εντοπίσει σφάλματα στη λειτουργία του ΣΔΒΔ τα οποία πρέπει να επιδιορθώσει.

Το αρχείο καταγραφής ερωτημάτων έχει την κατάληξη .log. Σε αυτό καταγράφονται όλες οι συνδέσεις που γίνονται από τους χρήστες στο ΣΔΒΔ και τα ερωτήματα που υποβάλουν. Το συγκεκριμένο αρχείο προσφέρει πλήρη εικόνα της δραστηριότητας στις ΒΔ του ΣΔΒΔ. Έτσι, αν προκύψει κάποιο πρόβλημα στο ΣΔΒΔ ο Διαχειριστής ΒΔ μπορεί να συμβουλευτεί το συγκεκριμένο αρχείο για να εντοπίσει πιθανές αιτίες του προβλήματος.

Το αρχείο καταγραφής αργών ερωτημάτων έχει όνομα που τελειώνει σε –slow.log. Το αρχείο περιέχει ερωτήματα που ο χρόνος εκτέλεσής τους ξεπερνάει μια προκαθορισμένη τιμή (η τιμή μπορεί να αλλαχθεί από τον Διαχειριστή ΒΔ).

Ο Διαχειριστής ΒΔ πρέπει να ελέγχει αυτό το αρχείο για να εντοπίζει ερωτήματα με μεγάλο χρόνο εκτέλεσης καθώς αυτά τα ερωτήματα μπορεί να προκαλέσουν γενικότερες καθυστερήσεις στη λειτουργία του ΣΔΒΔ.

Ο Διαχειριστής ΒΔ πρέπει να αναλύσει το πλάνο εκτέλεσης (execution plan) των ερωτημάτων με μεγάλο χρόνο εκτέλεσης ώστε να εντοπίσει που οφείλονται οι καθυστερήσεις. Όλα τα ΣΔΒΔ διαθέτουν μια ειδική εντολή που παρουσιάζει το πλάνο εκτέλεσης ενός ερωτήματος. Στο ΣΔΒΔ MySQL αυτό γίνεται αν χρησιμοποιηθεί η εντολή EXPLAIN στην αρχή ενός ερωτήματος.

Για παράδειγμα έχουμε την εντολή

```
EXPLAIN SELECT *  
FROM Μαθητής  
WHERE Επώνυμο='Παπαδόπουλος' AND Όνομα='Ιωάννης';
```

θα εμφανιστούν πληροφορίες για τον τρόπο εκτέλεσης του ερωτήματος. Οι σημαντικότερες πληροφορίες είναι τα κλειδιά που θα χρησιμοποιηθούν για την εκτέλεση του ερωτήματος και μια εκτίμηση του πλήθους των γραμμών που θα ελέγξει το ΣΔΒΔ για να εντοπίσει τις γραμμές που περιλαμβάνει το αποτέλεσμα. Αν από το πλάνο εκτέλεσης προκύψει ότι ένα ερώτημα με μεγάλο χρόνο εκτέλεσης δεν χρησιμοποιεί κάποιο κλειδί, ο Διαχειριστής ΒΔ θα πρέπει να δημιουργήσει κατάλληλα ευρετήρια (όπως εξηγήθηκε σε προηγούμενη ενότητα) για να μειωθεί ο χρόνος εκτέλεσης.

Στις καθημερινές εργασίες του Διαχειριστή ΒΔ περιλαμβάνεται και η δημιουργία αντιγράφου ασφαλείας (backup) της ΒΔ. Τα αντίγραφα ασφαλείας μπορούν να χρησιμοποιηθούν για την επαναφορά (restore) της ΒΔ σε προηγούμενη κατάσταση σε περίπτωση καταστροφικού προβλήματος (π.χ. σφάλμα στο δίσκο του υπολογιστή που είναι εγκατεστημένο το ΣΔΒΔ, καταστροφή των αρχείων με τα περιεχόμενα της ΒΔ, ιός κ.λπ.). Επίσης η δημιουργία αντιγράφου ασφαλείας είναι επιβεβλημένη πριν ξεκινήσουν εργασίες αναβάθμισης λογισμικού (π.χ. λειτουργικό σύστημα, ΣΔΒΔ).

Στο ΣΔΒΔ MySQL η δημιουργία αντιγράφου ασφάλειας γίνεται με την εφαρμογή `mysqldump` που είναι διαθέσιμη σε περιβάλλον γραμμής εντολών του λειτουργικού συστήματος. Η εφαρμογή `mysqldump` δεν δημιουργεί φυσικό αντίγραφο (physical backup) της ΒΔ, δηλαδή δεν αντιγράφει τα αρχεία δεδομένων με τα περιεχόμενα της ΒΔ. Αντίθετα δημιουργεί λογικό αντίγραφο (logical backup), δηλαδή παράγει εντολές SQL για να δημιουργηθούν ξανά η ΒΔ, οι πίνακες και τα ευρετήρια (CREATE DATABASE, CREATE TABLE, CREATE INDEX) και για να εισαχθούν τα δεδομένα στους πίνακες (INSERT). Το πλεονέκτημα του λογικού αντιγράφου είναι ότι το αρχείο που παράγεται είναι πολύ μικρό σε μέγεθος. Η εφαρμογή `mysqldump` πρέπει να εκτελεστεί στον φάκελο εγκατάστασης του ΣΔΒΔ MySQL.

Για παράδειγμα, αν θέλουμε να δημιουργήσουμε αντίγραφο της ΒΔ `mathitologio`, θα εκτελέσουμε την ακόλουθη εντολή:

Σε περιβάλλον Windows:

```
C:\mysql\bin\mysqldump -u root -p mathitologio > mathitologio_bak.sql
```

Σε περιβάλλον Linux:

```
/etc/init.d/mysqldump -u root -p mathitologio > mathitologio_bak.sql
```

Με την επιλογή `-u` καθορίζεται ο χρήστης που θα εκτελέσει την εντολή (στο παράδειγμα ο χρήστης είναι ο `root`) και η επιλογή `-p` καθορίζει ότι θα ζητηθεί κωδικός πριν ξεκινήσει η διαδικασία. Το αποτέλεσμα της εντολής θα αποθηκευτεί στο αρχείο `mathitologio_bak.sql` (το σύμβολο `>` καθορίζει που θα αποθηκευτεί το αποτέλεσμα). Το αποτέλεσμα είναι ένα αρχείο κειμένου με τις απαραίτητες εντολές SQL για να δημιουργηθεί ξανά η ΒΔ.

Ένα ΣΔΒΔ μπορεί να φιλοξενεί περισσότερες από μια ΒΔ. Σε αυτή την περίπτωση ο Διαχειριστής ΒΔ μπορεί να δημιουργήσει αντίγραφο ασφαλείας για όλες τις ΒΔ χρησιμοποιώντας την επιλογή `--all-databases`. Για παράδειγμα, για να δημιουργήσουμε αντίγραφο όλων των ΒΔ θα εκτελέσουμε την επόμενη εντολή:

Σε περιβάλλον Windows:

```
C:\mysql\bin\mysqldump -u root -p --all-databases > full_bak.sql
```

Σε περιβάλλον Linux:

```
/etc/init.d/mysqldump -u root -p --all-databases > full_bak.sql
```

Το αρχείο `full_bak.sql` που θα προκύψει θα περιέχει τις αναγκαίες εντολές SQL για να δημιουργηθούν ξανά όλες οι ΒΔ που φιλοξενεί το ΣΔΒΔ.

Η επαναφορά αντιγράφου ασφαλείας στο ΣΔΒΔ MySQL γίνεται επίσης με την εφαρμογή `mysqldump`. Η σύνταξη είναι ακριβώς ίδια με τη λήψη αντιγράφου με μόνη διαφορά ότι χρησιμοποιείται το σύμβολο `<` για να καθορίσει από που θα γίνει η εισαγωγή των στοιχείων. Για παράδειγμα, για να επαναφέρουμε το αντίγραφο της ΒΔ `mathitologio` θα εκτελεστεί η εντολή:

Σε περιβάλλον Windows:

```
C:\mysql\bin\mysqldump -u root -p new_mathitologio < mathitologio_bak.sql
```

Σε περιβάλλον Linux:

```
/etc/init.d/mysqldump -u root -p new_mathitologio < mathitologio_bak.sql
```

Όπως φαίνεται, χρησιμοποιείται το αρχείο mathitologio_bak.sql που δημιουργήθηκε προηγουμένως, για να γίνει επαναφορά της ΒΔ mathitologio στη ΒΔ new_mathitologio. Ειδικότερα, οι εντολές που περιέχει το αρχείο mathitologio_bak.sql θα εκτελεστούν ώστε να δημιουργηθεί ξανά η ΒΔ. Η ΒΔ στην οποία θα γίνει η επαναφορά πρέπει να έχει δημιουργηθεί με την εντολή CREATE DATABASE πριν ξεκινήσει η διαδικασία.

Η επαναφορά αντιγράφου όλων των ΒΔ του ΣΔΒΔ, θα γίνει με την εντολή:

Σε περιβάλλον Windows:

```
C:\mysql\bin\mysqldump -u root -p < full_bak.sql
```

Σε περιβάλλον Linux

```
/etc/init.d/mysqldump -u root -p < full_bak.sql
```

Όπως φαίνεται, χρησιμοποιείται το αρχείο full_bak.sql που δημιουργήθηκε προηγουμένως, για να γίνει επαναφορά όλων των ΒΔ που φιλοξενεί το ΣΔΒΔ. Σε αυτή την περίπτωση δεν χρειάζεται να δημιουργήσουμε από πριν τις ΒΔ καθώς το αρχείο full_bak.sql αφορά περισσότερες από μία ΒΔ και γι' αυτό περιέχει τις κατάλληλες εντολές CREATE DATABASE.

Όπως συμβαίνει με όλες τις εφαρμογές λογισμικού, έτσι και τα ΣΔΒΔ εξελίσσονται και οι κατασκευαστές τους παρουσιάζουν νεότερες εκδόσεις. Η νεότερη έκδοση μπορεί απλώς να επιλύει προβλήματα (λιγότερο ή περισσότερο σημαντικά) ή να προσφέρει επιπλέον δυνατότητες. Η απόφαση για αναβάθμιση (upgrade) σε νεότερη έκδοση του ΣΔΒΔ είναι πολύ σημαντική, ιδιαίτερα για ΣΔΒΔ που χρησιμοποιούνται σε παραγωγικό περιβάλλον, και θα πρέπει να λαμβάνεται αφού πρώτα μελετηθούν οι απαιτήσεις της νέας έκδοσης σε υλικό και λογισμικό και η συμβατότητα με παλαιότερες εκδόσεις.

Αν αποφασιστεί αναβάθμιση, τότε θα πρέπει να ακολουθηθούν πιστά οι οδηγίες του κατασκευαστή του ΣΔΒΔ, κυρίως για τον τρόπο μεταφοράς των ΒΔ στη νεότερη

έκδοση του ΣΔΒΔ. Η διαδικασία της αναβάθμισης είναι αρμοδιότητα του Διαχειριστή ΒΔ.

Η αναβάθμιση θα πρέπει πρώτα να δοκιμαστεί σε ΣΔΒΔ που δεν φιλοξενεί τις πραγματικές ΒΔ, ώστε να ελεγχθεί η διαδικασία και να εντοπιστούν πιθανά προβλήματα. Εφόσον η δοκιμή πετύχει τότε μπορεί να προχωρήσει η αναβάθμιση και του ΣΔΒΔ που χρησιμοποιείται σε παραγωγικό περιβάλλον.

Πριν την αναβάθμιση θα πρέπει ο Διαχειριστής ΒΔ να δημιουργήσει αντίγραφα ασφάλειας του ΣΔΒΔ και των ΒΔ που φιλοξενεί, ώστε αν παρουσιαστεί πρόβλημα να μπορέσει άμεσα να επαναφέρει το ΣΔΒΔ στην προηγούμενη κατάσταση.

Μετά την ολοκλήρωση της αναβάθμισης και πριν τεθεί το ΣΔΒΔ σε λειτουργία, ο Διαχειριστής ΒΔ χρησιμοποιεί ειδικά εργαλεία για να εξασφαλίσει τη σωστή λειτουργία των ΒΔ στην νεότερη έκδοση του ΣΔΒΔ. Στο ΣΔΒΔ MySQL μετά την αναβάθμιση πρέπει να εκτελεστεί σε περιβάλλον γραμμής εντολών του λειτουργικού συστήματος η εφαρμογή `mysql_upgrade`.

Η εφαρμογή αυτή επιδιορθώνει τη δομή των πινάκων και των ευρετηρίων ώστε να εξασφαλιστεί η συμβατότητα με τη νεότερη έκδοση του ΣΔΒΔ.

Εργαλεία Διαχείρισης ΒΔ

Ένας Διαχειριστής ΒΔ πρέπει να έχει τη δυνατότητα να πραγματοποιεί όλες τις εργασίες διαχείρισης για αυτό, υπάρχουν ειδικές εφαρμογές λογισμικού που κάνουν αυτές τις εργασίες ευκολότερες.

Ένα εργαλείο διαχείρισης ΒΔ διευκολύνει το έργο του Διαχειριστή ΒΔ παρέχοντας ένα γραφικό περιβάλλον εργασίας (GUI – graphical user interface) μέσω του οποίου μπορούν να πραγματοποιηθούν διάφορες εργασίες διαχείρισης της ΒΔ.

Συνήθως οι λειτουργίες που παρέχει ένα εργαλείο διαχείρισης ΒΔ είναι: δημιουργία ΒΔ, ρύθμιση παραμέτρων του ΣΔΒΔ και των ΒΔ, δημιουργία και τροποποίηση πινάκων και ευρετηρίων, υποβολή ερωτημάτων SQL (ανάκτηση, εισαγωγή, ενημέρωση, διαγραφή) με γραφικό τρόπο, σχηματική απεικόνιση των σχέσεων μεταξύ των πινάκων, δημιουργία χρηστών και διαχείριση προνομίων, εισαγωγή και

εξαγωγή δεδομένων, δημιουργία και επαναφορά αντιγράφων ασφάλειας, παρακολούθηση λειτουργίας του ΣΔΒΔ και των ΒΔ, επιδιόρθωση περιεχομένων ΒΔ.

Υπάρχουν προγράμματα που περιλαμβάνονται στην εγκατάσταση εγκατάστασης του ΣΔΒΔ που είναι εργαλεία διαχείρισης ΒΔ και αναπτύσσονται συνήθως από τους κατασκευαστές ΣΔΒΔ. Υπάρχουν όμως και εργαλεία διαχείρισης ΒΔ που έχουν αναπτυχθεί από τρίτους.

Πιο γνωστά εργαλεία διαχείρισης για το ΣΔΒΔ MySQL είναι το MySQL Workbench και το phpMyAdmin. Επίσης γνωστά εργαλεία είναι τα Toad for MySQL, HeidiSQL και Sequel Pro. Φυσικά μπορούν να κατηγοριοποιηθούν σύμφωνα με διάφορα κριτήρια τα εργαλεία διαχείρισης ΒΔ.

Με Κόστος: Κάποια είναι δωρεάν ,ενώ αλλά είναι επί πληρωμή. Για τα εργαλεία επί πληρωμή διατίθενται και δωρεάν δοκιμαστικές εκδόσεις (ο χρόνος χρήσης έχει περιορισμό και περιορισμό στις προσφερόμενες λειτουργίες).

Με την Άδεια χρήσης: Υπάρχουν εργαλεία ελεύθερου/ανοιχτού κώδικα (free/open source) στα οποία ο κώδικας της εφαρμογής είναι διαθέσιμος στο κοινό, σε ορισμένες περιπτώσεις ακόμα και για τροποποίηση. Από την άλλη υπάρχουν εργαλεία κλειστού κώδικα (closed source ή proprietary) που ο κώδικας δεν είναι διαθέσιμος και δεν επιτρέπεται σε κανέναν εκτός από τον κατασκευαστή να τον τροποποιήσει.

Υποστηριζόμενα λειτουργικά συστήματα:Ορισμένα εργαλεία είναι διαθέσιμα μόνο για συγκεκριμένο λειτουργικό σύστημα (π.χ. Windows, UNIX/Linux, MacOS), ενώ άλλα είναι ανεξάρτητα πλατφόρμας (cross-platform), δηλαδή υπάρχουν εκδόσεις για περισσότερα λειτουργικά συστήματα.

Υποστηριζόμενα ΣΔΒΔ: Μερικά εργαλεία συνεργάζονται μόνο με συγκεκριμένο ΣΔΒΔ (π.χ. MySQL, Oracle Database, MS SQL Server, IBM DB2), ενώ άλλα συνεργάζονται με περισσότερα ΣΔΒΔ.

Τύπος εφαρμογής: Υπάρχουν εργαλεία που είναι εφαρμογές πελάτη (client application), δηλαδή η εγκατάσταση τους γίνεται στον υπολογιστή του Διαχειριστή ΒΔ και συνδέονται μέσω δικτύου στον υπολογιστή που είναι εγκατεστημένο το

ΣΔΒΔ. Υπάρχουν όμως και εργαλεία που είναι εφαρμογές εξυπηρετητή (*server application*), δηλαδή η εγκατάστασή τους γίνεται σε κάποιον εξυπηρετητή (συνήθως σε αυτόν που είναι εγκατεστημένο και το ΣΔΒΔ) και ο Διαχειριστής ΒΔ χρησιμοποιεί ένα φυλλομετρητή Διαδικτύου (*web browser*) για να συνδεθεί στο εργαλείο.

Εκπαιδευτική Υποένότητα : 8.3 Διασύνδεση διαδικτυακών εφαρμογών με σχεσιακή βάση δεδομένων (PHP)

Η PHP υποστηρίζει διάφορες συνδέσεις βάσης δεδομένων με αυτήν. Η MySQL είναι μια από τις πιο ευρέως χρησιμοποιούμενες σχεσιακές βάσεις δεδομένων, και επίσης χρησιμοποιείται κυρίως με την PHP. Λαμβάνοντας υπόψη τον όρο σύνδεση βάσης δεδομένων στην PHP, η ίδια η MySQL έχει πολλούς τρόπους για να δημιουργήσει συνδέσεις σε μια εφαρμογή για λειτουργίες βάσης δεδομένων. Δημιουργήστε το αρχείο σύνδεσης της βάσης δεδομένων σε PHP. Δημιουργήστε ένα νέο αρχείο PHP και ονομάστε το `db_connection.php` και αποθηκεύστε το. Γιατί να δημιουργήσετε ένα ξεχωριστό αρχείο σύνδεσης βάσης δεδομένων; Επειδή εάν έχετε δημιουργήσει πολλά αρχεία για την εισαγωγή δεδομένων ή την επιλογή δεδομένων από τη βάση δεδομένων, δεν χρειάζεται να κωδικοποιείτε τη σύνδεση της βάσης δεδομένων κάθε φορά. Σύνδεση βάσης δεδομένων PHP Τελευταία ενημέρωση: 2 Νοεμβρίου 2020 Μια συλλογή σχετικών δεδομένων ονομάζεται βάση δεδομένων. Το XAMPP σημαίνει Cross-Platform, Apache, MySQL, PHP και Perl. Μερικές φορές μπορεί να χρειαστεί να συνδέσετε τον ιστότοπό σας που υποστηρίζεται από PHP σε μια βάση δεδομένων. Στις περισσότερες περιπτώσεις, το σύστημα διαχείρισης περιεχομένου γίνεται μέσω του αρχείου `config.php`. Παρακάτω είναι ένα δείγμα δέσμης ενεργειών PHP που συνδέεται με μια βάση δεδομένων και εμφανίζει όλα τα πεδία για έναν συγκεκριμένο πίνακα που καθορίζετε στον κώδικά σας.

Σύνδεση βάσης δεδομένων PHP Τελευταία ενημέρωση: 2 Νοεμβρίου 2020 Μια συλλογή σχετικών δεδομένων ονομάζεται βάση δεδομένων. Το XAMPP σημαίνει Cross-Platform, Apache, MySQL, PHP και Perl. Μπορεί να υπάρχουν φορές που χρειάζεται να συνδέσετε έναν ιστότοπο που υποστηρίζεται από PHP σε μια βάση

δεδομένων. Στις περισσότερες περιπτώσεις, το σύστημα διαχείρισης περιεχομένου γίνεται μέσω του αρχείου config.php. Παρακάτω είναι ένα δείγμα δέσμης ενεργειών PHP που συνδέεται με μια βάση δεδομένων και εμφανίζει όλα τα πεδία για έναν συγκεκριμένο πίνακα που καθορίζετε στον κώδικά σας. Η PHP υποστηρίζει διάφορες συνδέσεις βάσεων δεδομένων με αυτήν. Η MySQL είναι μια από τις πιο ευρέως χρησιμοποιούμενες σχεσιακές βάσεις δεδομένων, και επίσης χρησιμοποιείται κυρίως με την PHP. Λαμβάνοντας υπόψη τον όρο σύνδεση βάσης δεδομένων στην PHP, η ίδια η MySQL έχει πολλούς τρόπους για να δημιουργήσει συνδέσεις σε μια εφαρμογή για λειτουργίες βάσης δεδομένων. Η PHP 5 και μεταγενέστερη μπορεί να χρησιμοποιήσει τη βάση δεδομένων MySQL χρησιμοποιώντας: επέκταση MySQL

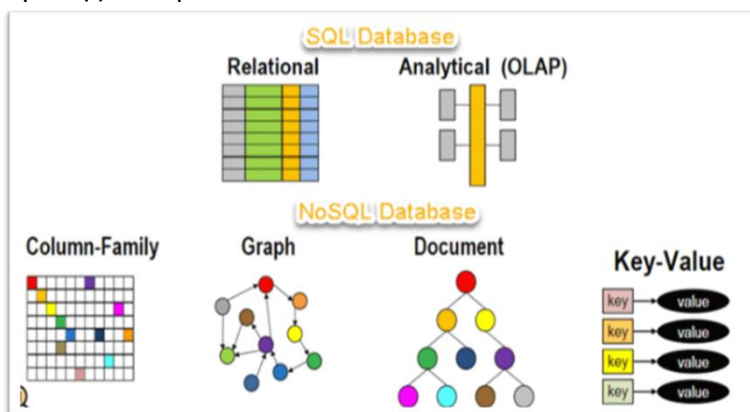
NoSQL Βάσεις Δεδομένων

Τι είναι το NoSQL;

Η βάση δεδομένων NoSQL είναι ένα μη σχεσιακό σύστημα διαχείρισης δεδομένων, το οποίο δεν απαιτεί σταθερό σχήμα. Αποφεύγει τις ενώσεις και είναι εύκολο να κλιμακωθεί. Ο κύριος σκοπός της χρήσης μιας βάσης δεδομένων NoSQL είναι για κατανεμημένες αποθήκες δεδομένων με τεράστιες ανάγκες αποθήκευσης δεδομένων. Το NoSQL χρησιμοποιείται για μεγάλα δεδομένα και εφαρμογές ιστού σε πραγματικό χρόνο. Για παράδειγμα, εταιρείες όπως το Twitter, το Facebook και η Google συλλέγουν terabyte δεδομένων χρηστών κάθε μέρα.

Η βάση δεδομένων NoSQL σημαίνει "Not Only SQL" ή "Not SQL". Αν και ένας καλύτερος όρος θα ήταν "NoREL", το NoSQL έπιασε. Ο Carl Strozzi εισήγαγε την έννοια NoSQL το 1998.

Το παραδοσιακό RDBMS χρησιμοποιεί σύνταξη SQL για την αποθήκευση και την ανάκτηση δεδομένων για περαιτέρω πληροφορίες. Αντίθετα, ένα σύστημα βάσης δεδομένων NoSQL περιλαμβάνει ένα ευρύ φάσμα τεχνολογιών βάσεων δεδομένων που μπορούν να αποθηκεύσουν δομημένα, ημιδομημένα, μη δομημένα και πολυμορφικά δεδομένα. Ας καταλάβουμε για το NoSQL με ένα διάγραμμα σε αυτό το σεμινάριο βάσης δεδομένων NoSQL:



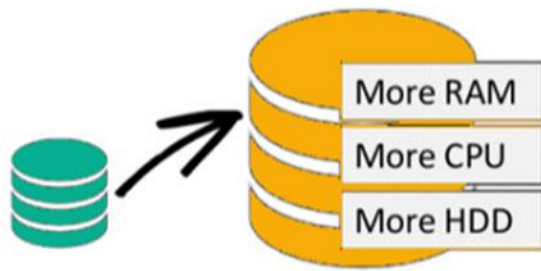
Γιατί να χρησιμοποιήσετε NoSQL;

Η έννοια των βάσεων δεδομένων NoSQL έγινε δημοφιλής από τους γίγαντες του Διαδικτύου όπως η Google, το Facebook, η Amazon κ.λπ. που ασχολούνται με τεράστιους όγκους δεδομένων. Ο χρόνος απόκρισης του συστήματος γίνεται αργός όταν χρησιμοποιείτε RDBMS για τεράστιους όγκους δεδομένων.

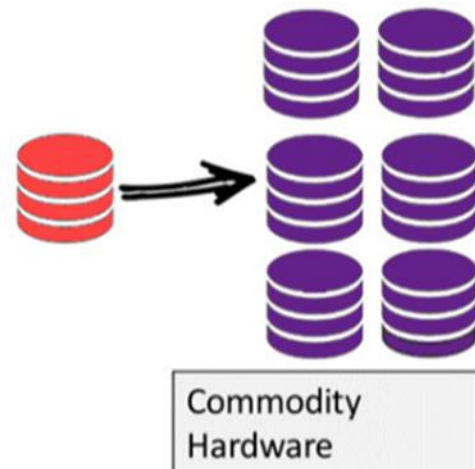
Για να επιλύσουμε αυτό το πρόβλημα, θα μπορούσαμε να «αναβαθμίσουμε» τα συστήματά μας αναβαθμίζοντας το υπάρχον υλικό μας. Αυτή η διαδικασία είναι ακριβή.

Η εναλλακτική λύση για αυτό το ζήτημα είναι η διανομή του φορτίου της βάσης δεδομένων σε πολλούς κεντρικούς υπολογιστές κάθε φορά που το φορτίο αυξάνεται. Αυτή η μέθοδος είναι γνωστή ως "scaling out".

Scale-Up (vertical scaling):



Scale-Out (horizontal scaling):



Η βάση δεδομένων NoSQL είναι μη σχεσιακή, επομένως κλιμακώνεται καλύτερα από τις σχεσιακές βάσεις δεδομένων, καθώς έχουν σχεδιαστεί με γνώμονα τις εφαρμογές Ιστού.

Σύντομο ιστορικό των βάσεων δεδομένων NoSQL

- 1998- Ο Carlo Strozzi χρησιμοποιεί τον όρο NoSQL για την ελαφριά, ανοιχτού κώδικα σχεσιακή βάση δεδομένων του
- 2000- Κυκλοφορεί η βάση δεδομένων γραφημάτων Neo4j
- 2004- Κυκλοφορεί το Google BigTable
- 2005 - Κυκλοφορεί το CouchDB
- 2007- Κυκλοφορεί η ερευνητική εργασία για το Amazon Dynamo
- 2008- Το Facebook ανοίγει πηγές το έργο Cassandra
- 2009- Ο όρος NoSQL επανεισαχθεί

Χαρακτηριστικά του NoSQL

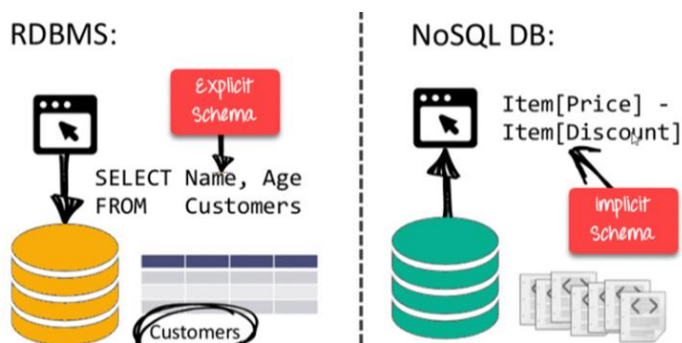
Μη σχέση

- Οι βάσεις δεδομένων NoSQL δεν ακολουθούν ποτέ ένα σχεσιακό μοντέλο
- Ποτέ μην παρέχετε σταθερές εγγραφές σταθερών στηλών για πίνακες

- Χρησιμοποιήστε αυτόνομα αδρανή ή BLOBs
- Δεν απαιτείται αντικειμενική σχεσιακή αντιστοίχιση και κανονικοποίηση δεδομένων
- Δεν υπάρχουν περίπλοκα χαρακτηριστικά όπως η γλώσσα ερωτημάτων, ο σχεδιασμός ερωτημάτων,
- Ενώσεις ακεραιότητας αναφοράς, ACID

Χωρίς σχήμα

- Οι βάσεις δεδομένων NoSQL είναι είτε χωρίς σχήματα είτε έχουν χαλαρά σχήματα
- Δεν απαιτείται κανένας ορισμός του σχήματος των δεδομένων



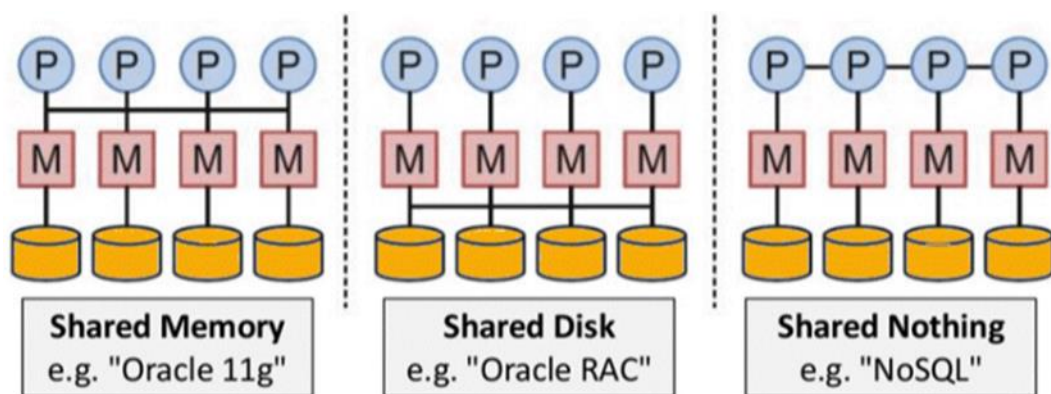
- Προσφέρει ετερογενείς δομές δεδομένων στον ίδιο τομέα.

Απλό API

- Προσφέρει εύχρηστες διεπαφές για αποθήκευση και αναζήτηση δεδομένων
- Τα API επιτρέπουν μεθόδους χειρισμού και επιλογής δεδομένων χαμηλού επιπέδου
- Πρωτόκολλα που βασίζονται σε κείμενο που χρησιμοποιούνται κυρίως με HTTP REST με JSON
- Συνήθως δεν χρησιμοποιείται καμία τυπική γλώσσα ερωτημάτων NoSQL
- Βάσεις δεδομένων με δυνατότητα Web που εκτελούνται ως υπηρεσίες που έχουν πρόσβαση στο Διαδίκτυο.

Διανέμονται

- Πολλές βάσεις δεδομένων NoSQL μπορούν να εκτελεστούν με κατανεμημένο τρόπο
- Προσφέρει δυνατότητες αυτόματης κλιμάκωσης και αποτυχίας
- Συχνά η έννοια του ACID μπορεί να θυσιαστεί για επεκτασιμότητα και απόδοση
- Συνήθως δεν υπάρχει σύγχρονη αναπαραγωγή μεταξύ κατανεμημένων κόμβων.
- Παρέχοντας μόνο την τελική συνέπεια
- Shared Nothing Architecture. Αυτό επιτρέπει λιγότερο συντονισμό και



μεγαλύτερη κατανομή

Τύποι βάσεων δεδομένων NoSQL

Οι βάσεις δεδομένων NoSQL κατηγοριοποιούνται κυρίως σε τέσσερις τύπους: Ζεύγος κλειδιού-τιμής, προσανατολισμένες στη στήλη, βασισμένες σε γραφήματα και προσανατολισμένες σε έγγραφα. Κάθε κατηγορία έχει τα μοναδικά χαρακτηριστικά και τους περιορισμούς της. Καμία από την παραπάνω καθορισμένη βάση δεδομένων δεν είναι καλύτερη για την επίλυση όλων των προβλημάτων. Οι χρήστες θα πρέπει να επιλέξουν τη βάση δεδομένων με βάση τις ανάγκες του προϊόντος τους.

Τύποι βάσεων δεδομένων NoSQL:

- Βάσει ζεύγους κλειδιού-τιμής

- Γράφημα προσανατολισμένο στη στήλη
- Βάσει γραφημάτων
- Προσανατολισμός εγγράφων

Βάσει ζεύγους βασικών τιμών

Τα δεδομένα αποθηκεύονται σε ζεύγη κλειδιών/τιμών. Είναι σχεδιασμένο με τέτοιο τρόπο ώστε να χειρίζεται πολλά δεδομένα και μεγάλο φορτίο.

Οι βάσεις δεδομένων αποθήκευσης ζεύγους κλειδιού-τιμής αποθηκεύουν δεδομένα ως πίνακα κατακερματισμού όπου κάθε κλειδί είναι μοναδικό και η τιμή μπορεί να είναι JSON, BLOB (Διαδικά μεγάλα αντικείμενα), συμβολοσειρά κ.λπ.

Για παράδειγμα, ένα ζεύγος κλειδιού-τιμής μπορεί να περιέχει ένα κλειδί όπως

Key	Value
Name	Joe Bloggs
Age	42
Occupation	Stunt Double
Height	175cm
Weight	77kg

"Ιστότοπος" που σχετίζεται με μια τιμή όπως "Guru99".

Είναι ένα από τα πιο βασικά παραδείγματα βάσης δεδομένων NoSQL. Αυτό το είδος βάσης δεδομένων NoSQL χρησιμοποιείται ως συλλογή, λεξικά, συσχετιστικοί πίνακες, κ.λπ. Οι χώροι αποθήκευσης βασικών τιμών βοηθούν τον προγραμματιστή να αποθηκεύει δεδομένα χωρίς σχήματα. Λειτουργούν καλύτερα για τα περιεχόμενα του καλαθιού αγορών. Τα Redis, Dynamo, Riak είναι μερικά παραδείγματα NoSQL βάσεων δεδομένων αποθήκευσης κλειδιών-τιμών. Όλα βασίζονται στο χαρτί Dynamo της Amazon.

Με βάση τη στήλη

Οι βάσεις δεδομένων προσανατολισμένες στη στήλη λειτουργούν σε στήλες και βασίζονται σε χαρτί BigTable της Google. Κάθε στήλη αντιμετωπίζεται ξεχωριστά. Οι τιμές των βάσεων δεδομένων μιας στήλης αποθηκεύονται συνεχώς.

ColumnFamily			
Row Key	Column Name		
	Key	Key	Key
	Value	Value	Value
	Column Name		
	Key	Key	Key
	Value	Value	Value

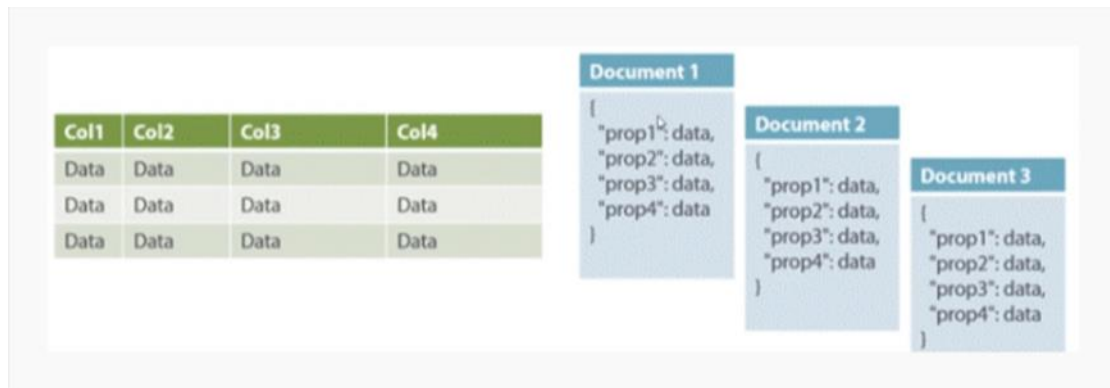
Παρέχουν υψηλή απόδοση σε ερωτήματα συνάθροισης όπως SUM, COUNT, AVG, MIN κ.λπ., καθώς τα δεδομένα είναι άμεσα διαθέσιμα σε μια στήλη.

Οι βάσεις δεδομένων NoSQL που βασίζονται σε στήλες χρησιμοποιούνται ευρέως για τη διαχείριση αποθηκών δεδομένων, επιχειρηματικής ευφυΐας, CRM, καταλόγων καρτών βιβλιοθήκης,

Τα HBase, Cassandra, HBase, Hypertable είναι παραδείγματα ερωτημάτων NoSQL βάσης δεδομένων που βασίζονται σε στήλη.

Προσανατολισμός εγγράφων:

Το NoSQL DB με προσανατολισμό εγγράφων αποθηκεύει και ανακτά δεδομένα ως ζεύγος βασικών τιμών, αλλά το τμήμα τιμής αποθηκεύεται ως έγγραφο. Το έγγραφο αποθηκεύεται σε μορφές JSON ή XML. Η τιμή είναι κατανοητή από το DB και μπορεί να ερωτηθεί.



Σε αυτό το διάγραμμα στα αριστερά σας μπορείτε να δείτε ότι έχουμε γραμμές και στήλες και στα δεξιά, έχουμε μια βάση δεδομένων εγγράφων που έχει παρόμοια δομή με το JSON. Τώρα για τη σχεσιακή βάση δεδομένων, πρέπει να ξέρετε τι στήλες έχετε και ούτω καθεξής. Ωστόσο, για μια βάση δεδομένων εγγράφων, έχετε χώρο αποθήκευσης δεδομένων όπως αντικείμενο JSON. Δεν χρειάζεται να ορίσετε ποιες το καθιστούν ευέλικτο.

Ο τύπος εγγράφου χρησιμοποιείται κυρίως για συστήματα CMS, πλατφόρμες blogging, αναλυτικά στοιχεία σε πραγματικό χρόνο και εφαρμογές ηλεκτρονικού εμπορίου. Δεν θα πρέπει να χρησιμοποιείται για πολύπλοκες συναλλαγές που απαιτούν πολλαπλές λειτουργίες ή ερωτήματα έναντι διαφορετικών συγκεντρωτικών δομών.

Τα Amazon SimpleDB, CouchDB, MongoDB, Riak, Lotus Notes, MongoDB, είναι δημοφιλή συστήματα DBMS που προέρχονται από έγγραφα.

Τι είναι το θεώρημα CAP;

Το θεώρημα CAP ονομάζεται επίσης θεώρημα ζυθοποιίας. Αναφέρει ότι είναι αδύνατο για ένα κατανομημένο κατάστημα δεδομένων να προσφέρει περισσότερες από δύο στις τρεις εγγυήσεις

- Συνοχή
- Διαθεσιμότητα
- Ανοχή κατάτμησης

Συνοχή:

Τα δεδομένα θα πρέπει να παραμένουν συνεπή ακόμη και μετά την εκτέλεση μιας λειτουργίας. Αυτό σημαίνει ότι μόλις γραφτούν τα δεδομένα, κάθε μελλοντικό αίτημα ανάγνωσης θα πρέπει να περιέχει αυτά τα δεδομένα. Για παράδειγμα, μετά την ενημέρωση της κατάστασης παραγγελίας, όλοι οι πελάτες θα πρέπει να μπορούν να δουν τα ίδια δεδομένα.

Διαθεσιμότητα:

Η βάση δεδομένων πρέπει να είναι πάντα διαθέσιμη και να ανταποκρίνεται. Δεν πρέπει να έχει χρόνο διακοπής λειτουργίας.

Ανοχή κατάτμησης:

Το Partition Tolerance σημαίνει ότι το σύστημα θα πρέπει να συνεχίσει να λειτουργεί ακόμα κι αν η επικοινωνία μεταξύ των διακομιστών δεν είναι σταθερή. Για παράδειγμα, οι διακομιστές μπορούν να χωριστούν σε πολλαπλές ομάδες που ενδέχεται να μην επικοινωνούν μεταξύ τους. Εδώ, εάν μέρος της βάσης δεδομένων δεν είναι διαθέσιμο, άλλα μέρη δεν επηρεάζονται πάντα.

Ενδεχόμενη συνέπεια

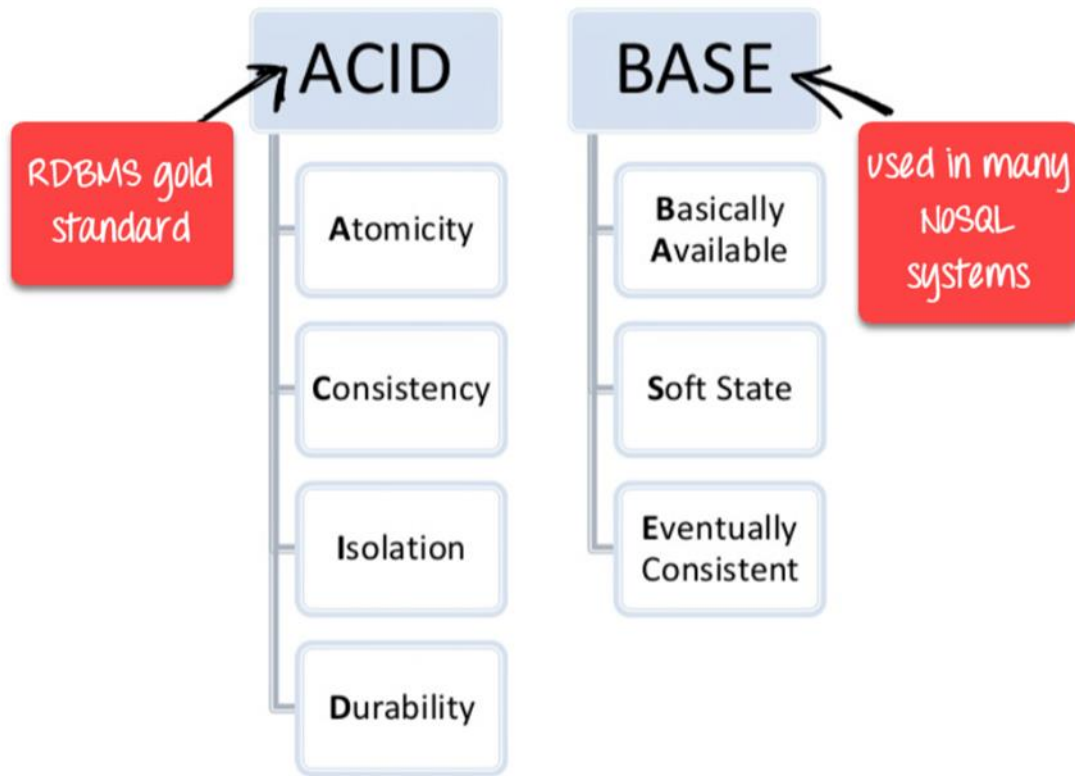
Ο όρος "τελική συνέπεια" σημαίνει ότι έχετε αντίγραφα δεδομένων σε πολλαπλές μηχανές για να έχετε υψηλή διαθεσιμότητα και επεκτασιμότητα. Έτσι, οι αλλαγές που γίνονται σε οποιοδήποτε στοιχείο δεδομένων σε ένα μηχάνημα πρέπει να μεταδοθούν σε άλλα αντίγραφα.

Η αναπαραγωγή δεδομένων ενδέχεται να μην είναι στιγμιαία, καθώς ορισμένα αντίγραφα θα ενημερωθούν αμέσως ενώ άλλα σε εύθετο χρόνο. Αυτά τα αντίγραφα μπορεί να είναι αμοιβαία, αλλά σε εύθετο χρόνο, γίνονται συνεπή. Ως εκ τούτου, το όνομα ενδεχόμενη συνέπεια.

ΒΑΣΗ: Β συνήθως Α διαθέσιμη, Σ συχνά κατάσταση, Ε ενδεχόμενη συνέπεια

- Βασικά, το διαθέσιμο σημαίνει ότι το DB είναι διαθέσιμο όλη την ώρα σύμφωνα με το θεώρημα CAP

- Μαλακή κατάσταση σημαίνει ακόμη και χωρίς είσοδο. η κατάσταση του συστήματος μπορεί να αλλάξει
- Η ενδεχόμενη συνέπεια σημαίνει ότι το σύστημα θα γίνει συνεπές με την πάροδο του χρόνου



To NoSQL

Γενικά αναφέρεται σε μη σχεσιακές βάσεις δεδομένων. Με την άνοδο των **ιστότοπων Διαδικτύου web 2.0**, οι παραδοσιακές σχεσιακές βάσεις δεδομένων έχουν καταστεί ανίκανες να χειριστούν ιστότοπους web 2.0, ειδικά εξαιρετικά μεγάλης κλίμακας και υψηλής ταυτόχρονης τεχνολογίας ιστοσελίδες web 2.0, και έχουν προκύψει πολλά ανυπέρβλητα προβλήματα. η μη σχεσιακή βάση δεδομένων έχει αναπτυχθεί πολύ γρήγορα λόγω των δικών της χαρακτηριστικών. Η βάση δεδομένων NoSQL δημιουργήθηκε για να λύσει τις προκλήσεις που φέρνει η συλλογή δεδομένων μεγάλης κλίμακας και οι πολλαπλοί τύποι δεδομένων, ειδικά η εφαρμογή μεγάλων δεδομένων. Η πιο κοινή ερμηνεία του NoSQL είναι "μη σχεσιακή", "Όχι μόνο SQL" είναι επίσης αποδεκτή από πολλούς ανθρώπους. Το NoSQL είναι απλώς μια έννοια που γενικά αναφέρεται σε μη σχεσιακές βάσεις

δεδομένων, διαφορετικά από τις σχεσιακές βάσεις δεδομένων, δεν εγγυώνται τις ιδιότητες ACID των σχεσιακών δεδομένων. Το NoSQL είναι ένα ολοκαίνουργιο επαναστατικό κίνημα βάσεων δεδομένων. Οι υποστηρικτές του υποστηρίζουν τη χρήση της μη σχεσιακής αποθήκευσης δεδομένων. Σε σύγκριση με τη συντριπτική χρήση των σχεσιακών βάσεων δεδομένων, αυτή η ιδέα είναι αναμφίβολα μια ένεση νέας σκέψης.

Το NoSQL έχει τα ακόλουθα πλεονεκτήματα: εύκολο στην επέκταση, υπάρχουν πολλοί τύποι βάσεων δεδομένων NoSQL, αλλά ένα κοινό χαρακτηριστικό είναι η κατάργηση των σχεσιακών χαρακτηριστικών των σχεσιακών βάσεων δεδομένων. Δεν υπάρχει σχέση μεταξύ των δεδομένων, γεγονός που καθιστά πολύ εύκολη την επέκταση. Το Invisibility φέρνει επίσης κλιμακούμενες δυνατότητες σε αρχιτεκτονικό επίπεδο. Μεγάλος όγκος δεδομένων, υψηλή απόδοση, η βάση δεδομένων NoSQL έχει πολύ υψηλή απόδοση ανάγνωσης και εγγραφής, ειδικά σε μεγάλο όγκο δεδομένων, η απόδοση είναι επίσης εξαιρετική. Χάρη στη μη σχεσιακή της φύση, η δομή της βάσης δεδομένων είναι απλή.

Ταξινόμηση επεξεργασία

Βάση δεδομένων αποθήκευσης κλειδιού-τιμής

Αυτός ο τύπος βάσης δεδομένων χρησιμοποιεί κυρίως έναν πίνακα κατακερματισμού, ο οποίος έχει ένα συγκεκριμένο κλειδί και έναν δείκτη σε συγκεκριμένα δεδομένα. Το πλεονέκτημα του μοντέλου κλειδιού/τιμής για συστήματα πληροφορικής είναι ότι είναι απλό και εύκολο στην ανάπτυξή του. Ωστόσο, εάν ο διαχειριστής της βάσης δεδομένων (DBA) υποβάλει ερωτήματα ή ενημερώσει μόνο μέρος της τιμής, το κλειδί/τιμή καθίσταται αναποτελεσματική. Για παράδειγμα: Tokyo Cabinet/Tyrant, Redis, Voldemort, Oracle BDB.

Βάση δεδομένων αποθήκευσης στήλης

Αυτό το τμήμα της βάσης δεδομένων χρησιμοποιείται συνήθως για την αντιμετώπιση μαζικών δεδομένων σε κατανεμημένη αποθήκευση. Τα κλειδιά εξακολουθούν να υπάρχουν, αλλά χαρακτηρίζονται από το ότι δείχνουν σε πολλές

στήλες. Οι στήλες είναι διατεταγμένες κατά οικογένειες στηλών. Όπως: Κασσάνδρα, HBase, Riak.

Βάση δεδομένων εγγράφων

Η βάση δεδομένων εγγράφων είναι εμπνευσμένη από το λογισμικό γραφείου Lotus Notes και είναι παρόμοια με το πρώτο κατάστημα κλειδιού-τιμής. Το μοντέλο δεδομένων αυτού του τύπου είναι έγγραφα έκδοσης, ημιδομημένα έγγραφα που αποθηκεύονται σε συγκεκριμένη μορφή, όπως το JSON. Οι βάσεις δεδομένων που βασίζονται σε έγγραφα μπορούν να θεωρηθούν ως μια αναβαθμισμένη έκδοση βάσεων δεδομένων κλειδιού-τιμής, επιτρέποντας ένθετα κλειδιά-τιμές. Όταν ασχολούμαστε με πολύπλοκα δεδομένα όπως ιστοσελίδες, οι βάσεις δεδομένων που βασίζονται σε έγγραφα έχουν υψηλότερη απόδοση ερωτημάτων από τις παραδοσιακές βάσεις δεδομένων κλειδιών-τιμών.

Όπως: CouchDB, MongoDB Υπάρχει επίσης μια βάση δεδομένων εγγράφων SequoiaDB στην Κίνα, η οποία είναι ανοιχτού κώδικα.

Βάση δεδομένων γραφημάτων

Σε αντίθεση με άλλες βάσεις δεδομένων SQL γραμμών-στήλων και άκαμπτης δομής, μια βάση δεδομένων με δομή γραφήματος χρησιμοποιεί ένα ευέλικτο μοντέλο γραφήματος και μπορεί να επεκταθεί σε πολλούς διακομιστές. Οι βάσεις δεδομένων NoSQL δεν διαθέτουν τυπική γλώσσα ερωτημάτων (SQL), επομένως η δημιουργία ερωτημάτων βάσης δεδομένων απαιτεί ένα μοντέλο δεδομένων. Πολλές βάσεις δεδομένων NoSQL διαθέτουν διεπαφές δεδομένων RESTful ή API ερωτημάτων. Όπως: Neo4J, InfoGrid, Infinite Graph.

Πλεονεκτήματα του NoSQL

- Μπορεί να χρησιμοποιηθεί ως Πρωτεύουσα ή Αναλυτική Πηγή Δεδομένων
- Δυνατότητα μεγάλων δεδομένων
- Κανένα σημείο αποτυχίας
- Εύκολη αναπαραγωγή
- Δεν χρειάζεται ξεχωριστό επίπεδο προσωρινής αποθήκευσης

- Παρέχει γρήγορη απόδοση και οριζόντια επεκτασιμότητα.
- Μπορεί να χειριστεί δομημένα, ημι-δομημένα και μη δομημένα δεδομένα με ίσο αποτέλεσμα
- Αντικειμενοστραφής προγραμματισμός που είναι εύκολος στη χρήση και ευέλικτος
- Οι βάσεις δεδομένων NoSQL δεν χρειάζονται έναν αποκλειστικό διακομιστή υψηλής απόδοσης
- Υποστήριξη βασικών γλωσσών και πλατφορμών προγραμματιστή
- Απλό στην εφαρμογή παρά με χρήση RDBMS
- Μπορεί να χρησιμεύσει ως η κύρια πηγή δεδομένων για διαδικτυακές εφαρμογές.
- Χειρίζεται μεγάλα δεδομένα που διαχειρίζονται την ταχύτητα, την ποικιλία, τον όγκο και την πολυπλοκότητα των δεδομένων
- Excel σε κατανεμημένες βάσεις δεδομένων και λειτουργίες κέντρου πολλαπλών δεδομένων
- Εξαλείφει την ανάγκη για ένα συγκεκριμένο επίπεδο προσωρινής αποθήκευσης για την αποθήκευση δεδομένων
- Προσφέρει έναν ευέλικτο σχεδιασμό σχήματος που μπορεί εύκολα να τροποποιηθεί χωρίς διακοπές λειτουργίας ή διακοπή της υπηρεσίας

Μειονεκτήματα του NoSQL

- Χωρίς κανόνες τυποποίησης
- Περιορισμένες δυνατότητες αναζήτησης
- Οι βάσεις δεδομένων και τα εργαλεία RDBMS είναι σχετικά ώριμα
- Δεν προσφέρει παραδοσιακές δυνατότητες βάσης δεδομένων, όπως η συνέπεια όταν εκτελούνται πολλές συναλλαγές ταυτόχρονα.

- Όταν ο όγκος των δεδομένων αυξάνεται, είναι δύσκολο να διατηρηθούν μοναδικές τιμές καθώς τα κλειδιά γίνονται δύσκολα
- Δεν λειτουργεί το ίδιο καλά με σχεσιακά δεδομένα
- Η καμπύλη εκμάθησης είναι σκληρή για τους νέους προγραμματιστές
- Επιλογές ανοιχτού κώδικα που δεν είναι τόσο δημοφιλείς για τις επιχειρήσεις.

Το Relational και το NoSQL είναι δύο τύποι συστημάτων βάσεων δεδομένων που συνήθως υλοποιούνται σε εγγενείς εφαρμογές cloud. Δημιουργούνται διαφορετικά, αποθηκεύουν δεδομένα διαφορετικά και έχουν διαφορετική πρόσβαση. Σε αυτή την ενότητα, θα εξετάσουμε και τα δύο. Αργότερα σε αυτό το κεφάλαιο, θα εξετάσουμε μια αναδυόμενη τεχνολογία βάσης δεδομένων που ονομάζεται NewSQL .

Οι σχεσιακές βάσεις δεδομένων είναι μια κυρίαρχη τεχνολογία εδώ και δεκαετίες. Είναι ώριμα, αποδεδειγμένα και ευρέως εφαρμοσμένα. Τα ανταγωνιστικά προϊόντα βάσης δεδομένων, τα εργαλεία και η εμπειρία αφθονούν. Οι σχεσιακές βάσεις δεδομένων παρέχουν μια αποθήκη πινάκων σχετικών δεδομένων. Αυτοί οι πίνακες έχουν σταθερό σχήμα, χρησιμοποιούν SQL (γλώσσα SQL) για τη διαχείριση δεδομένων και υποστηρίζουν εγγυήσεις ACID.

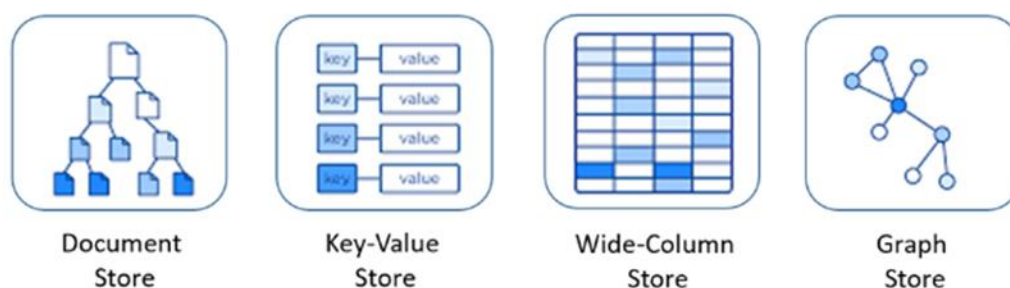
Οι βάσεις δεδομένων που δεν είναι SQL αναφέρονται σε αποθήκες μη σχεσιακών δεδομένων υψηλής απόδοσης. Ξεχωρίζουν για τα χαρακτηριστικά ευκολίας χρήσης, επεκτασιμότητας, ανθεκτικότητας και διαθεσιμότητας.

Αντί να ενώνει πίνακες κανονικοποιημένων δεδομένων, το NoSQL αποθηκεύει μη δομημένα, ή ημιδομημένα δεδομένα, συχνά σε ζεύγη κλειδιών-τιμών ή έγγραφα JSON. Οι βάσεις δεδομένων χωρίς SQL συνήθως δεν παρέχουν εγγυήσεις ACID πέρα από το πεδίο εφαρμογής ενός μεμονωμένου διαμερίσματος βάσης δεδομένων. Υπηρεσίες μεγάλου όγκου που απαιτούν χρόνους απόκρισης μικρότερου του δευτερολέπτου ευνοούν τις αποθήκες δεδομένων NoSQL.

Ο αντίκτυπος των τεχνολογιών NoSQL για τα κατακεκομμένα εγγενή συστήματα cloud δεν μπορεί να υπερεκτιμηθεί.

Ο πολλαπλασιασμός των νέων τεχνολογιών δεδομένων σε αυτόν τον χώρο έχει διαταράξει λύσεις που προηγουμένως βασιζόνταν αποκλειστικά σε σχεσιακές βάσεις δεδομένων.

Οι βάσεις δεδομένων NoSQL περιλαμβάνουν πολλά διαφορετικά μοντέλα πρόσβασης και διαχείρισης δεδομένων, το καθένα κατάλληλο για συγκεκριμένες περιπτώσεις χρήσης. Το σχήμα 5-9 παρουσιάζει τέσσερα κοινά μοντέλα.



Εικόνα 5-9 : Μοντέλα δεδομένων για βάσεις δεδομένων NoSQL

Να μοντελοποιήσω	Χαρακτηριστικά
αποθήκη εγγράφων	Τα δεδομένα και τα μεταδεδομένα αποθηκεύονται ιεραρχικά σε έγγραφα που βασίζονται σε JSON στη βάση δεδομένων.
Κατάστημα βασικών τιμών	Η απλούστερη από τις βάσεις δεδομένων NoSQL, οι οποίες αντιπροσωπεύονται ως μια συλλογή ζευγών κλειδιών-τιμών.
Κατάστημα με ευρεία στήλη	Τα σχετικά δεδομένα αποθηκεύονται ως ένα σύνολο ζευγών τιμών/κλειδιών που είναι ένθετα σε μία στήλη.
αποθήκη γραφημάτων	Τα δεδομένα αποθηκεύονται σε μια δομή γραφήματος ως ιδιότητες κόμβου, ακμής και δεδομένων.

Σκεφτείτε ένα χώρο αποθήκευσης δεδομένων NoSQL όταν:	Σκεφτείτε μια σχεσιακή βάση δεδομένων όταν:
Έχετε φόρτους εργασίας μεγάλου όγκου που απαιτούν προβλέψιμο λανθάνοντα χρόνο σε κλίμακα (π.χ. λανθάνουσα κατάσταση που μετράται σε χιλιοστά του δευτερολέπτου κατά την εκτέλεση εκατομμυρίων συναλλαγών ανά δευτερόλεπτο)	Ο όγκος φόρτου εργασίας συνήθως προσαρμόζεται σε χιλιάδες συναλλαγές ανά δευτερόλεπτο
Τα δεδομένα σας είναι δυναμικά και αλλάζουν συχνά	Τα δεδομένα σας είναι εξαιρετικά δομημένα και απαιτούν ακεραιότητα αναφοράς
Οι σχέσεις μπορούν να είναι αποκανονικοποιημένα μοντέλα δεδομένων	Οι σχέσεις εκφράζονται μέσω ενώσεων πινάκων σε κανονικοποιημένα μοντέλα δεδομένων
Η ανάκτηση δεδομένων είναι απλή και εκφράζεται χωρίς ενώσεις πινάκων	Εργάζεστε με πολύπλοκα ερωτήματα και αναφορές
Τα δεδομένα αναπαράγονται συνήθως σε διάφορες γεωγραφικές περιοχές και απαιτούν καλύτερο έλεγχο της συνέπειας, της διαθεσιμότητας και της απόδοσης	Τα δεδομένα είναι συνήθως συγκεντρωμένα ή μπορούν να αναπαραχθούν ασύγχρονα
Η εφαρμογή σας θα αναπτυχθεί σε υλικό βασικών προϊόντων, όπως με δημόσια σύννεφα	Η εφαρμογή σας θα αναπτυχθεί σε μεγάλο, προηγμένο υλικό

Θεωρήσεις για σχεσιακά συστήματα έναντι NoSQL

Με βάση συγκεκριμένες απαιτήσεις δεδομένων, μια εγγενής μικρουπηρεσία που βασίζεται σε σύννεφο μπορεί να εφαρμόσει ένα σχεσιακό χώρο αποθήκευσης δεδομένων, NoSQL ή και τα δύο.

Στις επόμενες ενότητες, θα εξερευνήσουμε τις διαθέσιμες επιλογές στο cloud Azure για την αποθήκευση και τη διαχείριση των εγγενών δεδομένων σας στο cloud.

Η βάση δεδομένων ως υπηρεσία

Για να ξεκινήσετε, μπορείτε να παρέχετε μια εικονική μηχανή Azure και να εγκαταστήσετε τη βάση δεδομένων της επιλογής σας για κάθε υπηρεσία. Ενώ έχετε τον πλήρη έλεγχο του περιβάλλοντος, θα εγκαταλείπατε πολλές ενσωματωμένες δυνατότητες της πλατφόρμας cloud. Θα είστε επίσης υπεύθυνοι για τη διαχείριση της εικονικής μηχανής και της βάσης δεδομένων για κάθε υπηρεσία. Αυτή η προσέγγιση μπορεί γρήγορα να γίνει χρονοβόρα και δαπανηρή.

Αντίθετα, οι εγγενείς εφαρμογές στο cloud ευνοούν τις υπηρεσίες δεδομένων που εκτίθενται ως DBaaS (Βάση δεδομένων ως υπηρεσία) . Με πλήρη διαχείριση από





έναν προμηθευτή cloud, αυτές οι υπηρεσίες παρέχουν ενσωματωμένη ασφάλεια, επεκτασιμότητα και παρακολούθηση. Αντί να κατέχετε την υπηρεσία, απλώς την καταναλώνετε ως υπηρεσία υποστήριξης. Ο πάροχος χειρίζεται τον πόρο σε κλίμακα και έχει την ευθύνη για την απόδοση και τη συντήρηση.

Μπορούν να διαμορφωθούν σε ζώνες και περιοχές διαθεσιμότητας cloud για υψηλή διαθεσιμότητα. Όλα υποστηρίζουν τη δυνατότητα ακριβώς έγκαιρα και ένα μοντέλο πληρωμής. Το Azure διαθέτει διαφορετικούς τύπους επιλογών υπηρεσίας διαχειριζόμενων δεδομένων, ο καθένας με συγκεκριμένα πλεονεκτήματα.

Αρχικά, θα δούμε τις σχεσιακές υπηρεσίες DBaaS που είναι διαθέσιμες στο Azure. Θα διαπιστώσετε ότι η κορυφαία βάση δεδομένων SQL Server της Microsoft είναι διαθέσιμη μαζί με πολλές επιλογές ανοιχτού κώδικα. Στη συνέχεια, θα μιλήσουμε για τις υπηρεσίες δεδομένων NoSQL στο Azure.

Σχεσιακές βάσεις δεδομένων Azure

Για μικροϋπηρεσίες εγγενούς στο cloud που απαιτούν σχεσιακά δεδομένα, το Azure προσφέρει τέσσερις προσφορές διαχειριζόμενης σχεσιακής βάσης δεδομένων ως υπηρεσία (DBaaS), που φαίνονται στην Εικόνα 5-11.

SQL Database 	MySQL 	MariaDB 	PostgreSQL 
Shared Database Services Platform	<ul style="list-style-type: none"> • Built-in high availability with a 99.99% service level agreement • Automatic upgrades, patching, and backups • On-demand scaling with pay-as-you-go pricing • Intelligence: Advisors, tuning, and monitoring • Active geo-replication with readable secondary databases • Enterprise-grade security to protect data at-rest and in-motion 		
Azure Compute layer			
Azure Storage layer			
Globally available across 54 Azure regions			

Εικόνα 5-11 . Διαχειριζόμενες σχεσιακές βάσεις δεδομένων διαθέσιμες στο Azure

Στο προηγούμενο σχήμα, παρατηρήστε πώς κάθε ένα βρίσκεται σε μια κοινή υποδομή DBaaS που διαθέτει βασικές λειτουργίες χωρίς επιπλέον κόστος.

Αυτά τα χαρακτηριστικά είναι ιδιαίτερα σημαντικά για οργανισμούς που παρέχουν μεγάλο αριθμό βάσεων δεδομένων αλλά έχουν περιορισμένους πόρους για τη διαχείρισή τους. Μπορείτε να παρέχετε μια βάση δεδομένων Azure μέσα σε λίγα λεπτά επιλέγοντας την ποσότητα των πυρήνων επεξεργασίας, της μνήμης και του υποκείμενου χώρου αποθήκευσης. Μπορείτε να κλιμακώσετε τη βάση δεδομένων αμέσως και να προσαρμόσετε δυναμικά τους πόρους με ελάχιστο ή καθόλου χρόνο διακοπής λειτουργίας.

Βάση δεδομένων Azure SQL

Οι ομάδες ανάπτυξης με εμπειρία στον Microsoft SQL Server θα πρέπει να εξετάσουν τη βάση δεδομένων Azure SQL . Είναι μια πλήρως διαχειριζόμενη σχεσιακή βάση δεδομένων ως υπηρεσία (DBaaS) που βασίζεται στη Μηχανή Βάσης Δεδομένων του Microsoft SQL Server. Η υπηρεσία μοιράζεται πολλές δυνατότητες που βρίσκονται στην τοπική έκδοση του SQL Server και εκτελεί την πιο πρόσφατη σταθερή έκδοση του SQL Server Database Engine.

Για χρήση με μια εγγενή microservice, η βάση δεδομένων Azure SQL είναι διαθέσιμη με τρεις επιλογές ανάπτυξης:

- Μια ενιαία βάση δεδομένων αντιπροσωπεύει μια πλήρως διαχειριζόμενη βάση δεδομένων SQL που εκτελείται σε έναν διακομιστή βάσης δεδομένων Azure SQL στο σύννεφο Azure. Η βάση δεδομένων θεωρείται ότι περιέχεται καθώς δεν έχει εξαρτήσεις διαμόρφωσης από τον υποκείμενο διακομιστή βάσης δεδομένων.
- Μια διαχειριζόμενη παρουσία είναι μια πλήρως διαχειριζόμενη παρουσία του Microsoft SQL Server Database Engine που παρέχει σχεδόν 100% συμβατότητα με έναν SQL Server εσωτερικής εγκατάστασης. Αυτή η επιλογή υποστηρίζει μεγαλύτερες βάσεις δεδομένων, έως 35 TB, και τοποθετείται σε ένα εικονικό δίκτυο Azure για καλύτερη απομόνωση.

- Η βάση δεδομένων Azure SQL χωρίς διακομιστή είναι ένα επίπεδο υπολογιστών για μια ενιαία βάση δεδομένων που κλιμακώνεται αυτόματα με βάση τη ζήτηση φόρτου εργασίας. Χρεώνει μόνο για την ποσότητα υπολογισμού που χρησιμοποιείται ανά δευτερόλεπτο. Η υπηρεσία είναι κατάλληλη για φόρτους εργασίας με διαλείποντα και απρόβλεπτα μοτίβα χρήσης που διανθίζονται με περιόδους αδράνειας. Το επίπεδο υπολογισμού χωρίς διακομιστή διακόπτει επίσης αυτόματα τις βάσεις δεδομένων κατά τις περιόδους αδράνειας, ώστε να χρεώνονται μόνο χρεώσεις αποθήκευσης. Συνεχίζεται αυτόματα όταν επιστρέψει η δραστηριότητα.

Εκτός από την παραδοσιακή στοίβα του Microsoft SQL Server, το Azure διαθέτει επίσης διαχειριζόμενες εκδόσεις τριών δημοφιλών βάσεων δεδομένων ανοιχτού κώδικα.

Βάσεις δεδομένων ανοιχτού κώδικα στο Azure

Οι σχεσιακές βάσεις δεδομένων ανοιχτού κώδικα έχουν γίνει μια δημοφιλής επιλογή για εγγενείς εφαρμογές cloud. Πολλές εταιρείες τα προτιμούν έναντι των εμπορικών προϊόντων βάσης δεδομένων, ειδικά για εξοικονόμηση κόστους. Πολλές ομάδες ανάπτυξης απολαμβάνουν την ευελιξία, την ανάπτυξη που υποστηρίζεται από την κοινότητα και το οικοσύστημα εργαλείων και επεκτάσεων. Οι βάσεις δεδομένων ανοιχτού κώδικα μπορούν να αναπτυχθούν σε πολλούς παρόχους cloud, συμβάλλοντας στην ελαχιστοποίηση της ανησυχίας του "κλειδώματος προμηθευτή".

Οι προγραμματιστές μπορούν εύκολα να φιλοξενήσουν οποιαδήποτε βάση δεδομένων ανοιχτού κώδικα σε ένα Azure VM. Με την παροχή πλήρους ελέγχου, αυτή η προσέγγιση σας φέρνει στο προσκήνιο για τη διαχείριση, την παρακολούθηση και τη συντήρηση βάσης δεδομένων και VM.


Ωστόσο, η Microsoft συνεχίζει τη δέσμευσή της να διατηρεί το Azure μια "ανοικτή πλατφόρμα", προσφέροντας πολλές δημοφιλείς βάσεις δεδομένων ανοιχτού κώδικα ως πλήρως διαχειριζόμενες υπηρεσίες DBaaS .

Εφαρμογές Βάσεων Δεδομένων

Καθημερινά χρησιμοποιούμε διάφορες εφαρμογές του Διαδικτύου που χρησιμοποιούν Βάσεις Δεδομένων.

Το κεφάλαιο αυτό αξιοποιεί τρία σύγχρονα παραδείγματα ΒΔ, οργανωμένα ως δραστηριότητες, ώστε μέσα από τη χρήση τους να εξερευνηθούν θέματα σχεδίασης και υλοποίησης Συστημάτων Βάσεων Δεδομένων και ανάπτυξης Διαδικτυακών εφαρμογών.

Διαδικτυακή Βάση Δεδομένων Διαθέσιμων *Ραντεβού Ιατρών ΕΟΠΥΥ* Η Διαδικτυακή εφαρμογή αναζήτησης συμβεβλημένων παροχών υγείας από την επίσημη ιστοσελίδα του Εθνικού Οργανισμού Παροχής Υπηρεσιών Υγείας (ΕΟΠΥΥ), βοηθά στον εντοπισμό



Γιατροί ΕΟΠΥΥ

Γιατροί ΕΟΠΥΥ

Περίοδος * Ιούλιος 2015

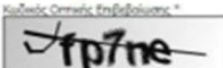
Νομός * [Dropdown]

Ειδικότητα Γιατρού * [Dropdown]

Εμφάνιση Όλα

Περιορισμός Αποτελεσμάτων (Π. Όνομα Γιατρού, Πόλη, ΤΚ κλπ.) [Input]

Κωδικός Οπτικής Επιδόσεως * Εισάγετε εδώ τον Κωδικό Οπτικής Επιδόσεως όπως εμφανίζεται αριστερά




Αναζήτηση [Button]

(Σελίδα 1 από 1)

Γιατρός	Διεύθυνση	Τηλέφωνο	Διαθεσιμότητα/ Πλαφόν	Μεταβολές/ Ακυρώσεις	Χάρτης
(Σελίδα 1 από 1)					

* Ενημερώνεστε καθημερινά για τα διαθέσιμα ραντεβού κάθε γιατρού, καθώς έχουν ήδη αφαιρεθεί τα προγραμματισμένα και τα εκτελεσμένα ραντεβού από τα προβλεπόμενα όρια.



Γιατροί ΕΟΠΥΥ

Γιατροί ΕΟΠΥΥ

Περίοδος * Ιούλιος 2015

Νομός * ΑΤΤΙΚΗΣ

Ειδικότητα Γιατρού * ΓΥΝΑΙΚΟΛΟΓΙΑ

Εμφάνιση Με Διαθεσιμότητα

Περιορισμός Αποτελεσμάτων (Π. Όνομα Γιατρού, Πόλη, ΤΚ κλπ.) [Input]

Αναζήτηση [Button]

(Σελίδα 1 από 9)

Γιατρός	Διεύθυνση	Τηλέφωνο	Διαθεσιμότητα/ Πλαφόν	Μεταβολές/ Ακυρώσεις	Χάρτης
ΑΒΡΑΜΩΣ ΒΑΣΙΛΕΙΟΣ	ΝΕΟΦΥΤΟΥ ΒΑΜΒΑ 4, ΑΘΗΝΑ, 10674	6945437676	190/200	0/0	[Χάρτης]
ΑΓΓΕΛΑΤΟΣ ΓΕΡΑΣΙΜΟΣ	ΠΑΤΗΣΣΙΝ 126 & 10ΑΚΗΣ, ΑΘΗΝΑ, 11257	2108824151	101/200	0/0	[Χάρτης]
ΑΓΚΟΣΤΟΝ ΚΑΤΑΛΙΝ	ΚΥΠΡΩΝ ΗΡΩΩΝ 70, ΗΛΙΟΥΠΟΛΗ, 16341	2109910551	167/200	0/0	[Χάρτης]
ΑΛΑΤΣΙΔΟΥ ΖΗΝΟΒΙΑ	ΣΟΛΩΝΟΣ 34, ΚΟΛΩΝΑΚΙ ΑΘΗΝΑ, 10673	2109244040 6944759665	167/200	0/0	[Χάρτης]
ΑΝΔΡΙΑΝΟΠΟΥΛΟΣ ΕΥΣΤΑΘΙΟΣ - ΔΗΜΟΣΘΕΝ	ΕΘΝΙΚΗΣ ΑΝΤΙΣΤΑΣΕΩΣ 65, ΧΑΛΑΝΔΡΙ, 15231	2114063227 6944265526	176/200	0/0	[Χάρτης]
ΑΡΤΟΠΟΥΛΟΣ ΡΗΝΑΣ	ΚΡΗΤΗΣ 72, ΝΕΑ ΣΙΔΕΑ, 14231	2102717066	193/200	0/0	[Χάρτης]

ιατρών σε όλες τις γεωγραφικές περιοχές της Ελλάδας, με συγκεκριμένα κριτήρια επιλογής.

Χρησιμοποιείται η διεύθυνση: <https://apps.ika.gr/eFindDoctor/> ή επιλογές στην επίσημη ιστοσελίδα του ΕΟΠΥΥ.

Ψηφιοποιημένο Αρχείο Εθνικού Θεάτρου

Το Εθνικό Θέατρο έχει δημιουργήσει ψηφιοποιημένο αρχείο για τις παραστάσεις από το 1932 μέχρι το 2006. Το αρχείο είναι διαθέσιμο στο ευρύ κοινό μέσω του Διαδικτύου.

Για κάθε παράσταση υπάρχουν πληροφορίες για το έργο και τους συντελεστές της παράστασης, φωτογραφίες, προγράμματα παραστάσεων και αποκόματα εφημερίδων. Για παραστάσεις μετά το 1955 υπάρχει ηχητικό υλικό ενώ για παραστάσεις μετά το 1994 υπάρχουν βιντεοσκοπήσεις.

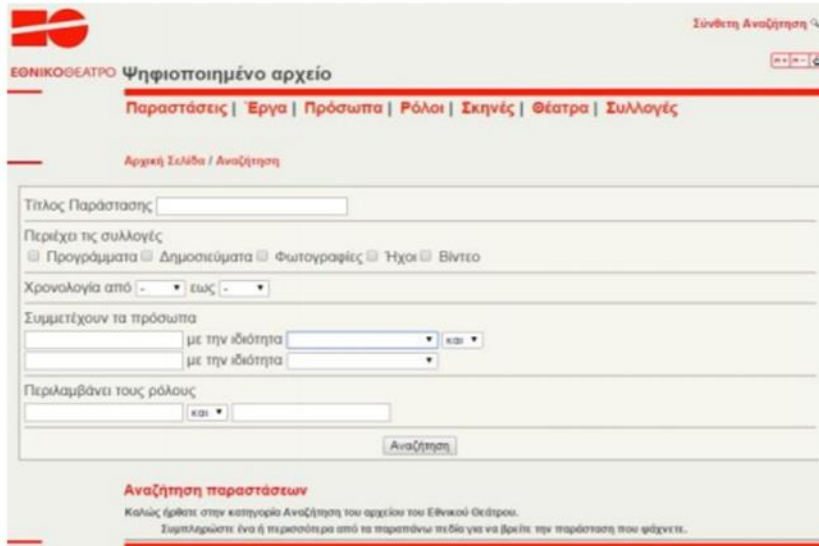
Το ψηφιοποιημένο αρχείο του Εθνικού Θεάτρου αποτελεί χαρακτηριστικό παράδειγμα εφαρμογής ΒΔ στο Διαδίκτυο.

Όλες οι πληροφορίες σε μορφή κειμένου και το πολυμεσικό υλικό (εικόνες, ήχοι, βίντεο) είναι αποθηκευμένα σε ΒΔ που φιλοξενείται σε κάποιο ΣΔΒΔ. Η διάθεση των περιεχομένων της ΒΔ γίνεται μέσω ενός εξυπηρετητή Διαδικτύου (web server) που επικοινωνεί με το ΣΔΒΔ.

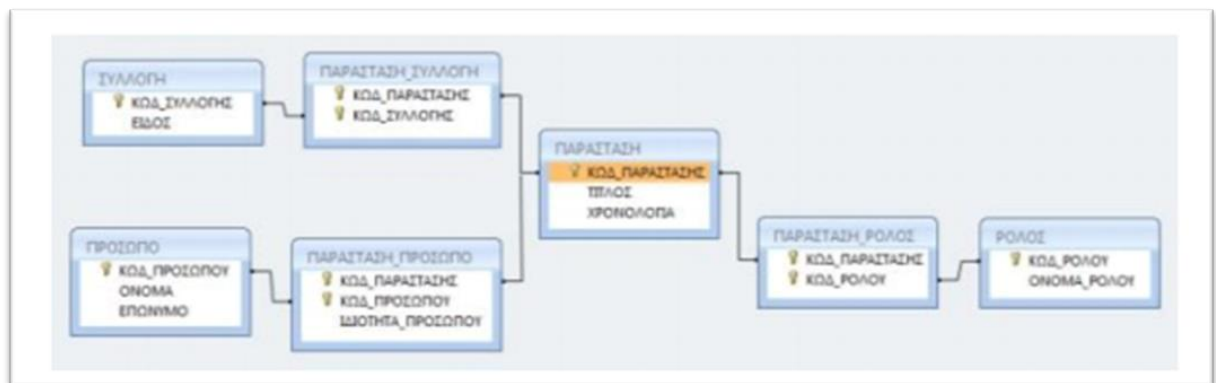
Ο εξυπηρετητής Διαδικτύου φιλοξενεί τη διαδικτυακή εφαρμογή (web application), η οποία έχει τη μορφή μιας φόρμας και επιτρέπει στους χρήστες να πραγματοποιούν αναζητήσεις στα περιεχόμενα του ψηφιοποιημένου αρχείου. Η πρόσβαση στη φόρμα αναζήτησης γίνεται μέσω ενός φυλλομετρητή Διαδικτύου (web browser).

1. Χρησιμοποιήστε ένα φυλλομετρητή Διαδικτύου για να επισκεφθείτε το ψηφιοποιημένο αρχείο του Εθνικού Θεάτρου στη διεύθυνση <http://www.nt-archive.gr/>.
2. Στο επάνω μέρος της σελίδας υπάρχουν σύνδεσμοι για τις φόρμες απλής αναζήτησης. Χρησιμοποιήστε αυτές τις φόρμες για να πραγματοποιήσετε αναζητήσεις για έργα και ρόλους, ώστε να αποκτήσετε μια πρώτη εικόνα για το είδος και το πλήθος των πληροφοριών που είναι διαθέσιμες. Μπορείτε να αναζητήσετε πληροφορίες για κάποιο κλασικό θεατρικό έργο που παρακολουθήσατε πρόσφατα ή για τα έργα που διδαχθήκατε στο Γυμνάσιο στο πλαίσιο του μαθήματος «Δραματική Ποίηση» («Ελένη» του Ευριπίδη και «Όρνιθες» του Αριστοφάνη).

3. Ακολουθήστε το σύνδεσμο στο άνω δεξί τμήμα της οθόνης για να επισκεφθείτε τη φόρμα σύνθετης αναζήτησης (βλ. εικόνα). Κάθε πεδίο της φόρμας πρέπει να αντιστοιχεί σε στήλη ενός πίνακα της ΒΔ.



4. Θεωρήστε ότι το σχήμα της ΒΔ που υλοποιεί το ψηφιοποιημένο αρχείο του Εθνικού Θεάτρου είναι αυτό της εικόνας [πρόκειται για απλοποιημένο σχήμα της ΒΔ που δημιουργήθηκε για εκπαιδευτικούς σκοπούς – η πραγματική ΒΔ σίγουρα περιέχει περισσότερους πίνακες με επιπλέον στήλες].



Ειδικά Θέματα - Παραδείγματα

Διασύνδεση εφαρμογής Java με MySQL

Στο παρόν κεφάλαιο θα παρουσιασθούν βασικά στοιχεία όσον αφορά την σύνδεση ενός προγράμματος Java με την MySQL και την εκτέλεση εντολών σε αυτή.

Εισαγωγικές Έννοιες

Το JAVA JDBC είναι ένα JAVA API με το οποίο μπορούμε να έχουμε πρόσβαση σε οποιαδήποτε μορφής δεδομένα πίνακα και ειδικότερα σε δεδομένα μιας σχεσιακής Βάσης Δεδομένων.

Με το JDBC μπορούμε να γράψουμε Java εφαρμογές, οι οποίες διαχειρίζονται τις παρακάτω λειτουργίες:

1. Σύνδεση στη Βάση Δεδομένων
2. Αποστολή ερωτημάτων και ενημέρωση δεδομένων στη Βάση
3. Ανάκτηση και επεξεργασία των αποτελεσμάτων που λαμβάνονται από τη βάση ως απάντηση στο ερώτημά μας.

```
public void connectToAndQueryDatabase(String username, String password) {
    Connection con = DriverManager.getConnection(
        "jdbc:myDriver:myDatabase",
        username,
        password); // Εντολή σύνδεσης στο ΣΔΒΔ
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT a, b, c FROM Table1");
    while (rs.next()) { // Επανάληψη στις εγγραφές που έχουν επιστραφεί
        int x = rs.getInt("a"); // Επιλογή του χαρακτηριστικού a σε μία
                               // μεταβλητή τύπου int.
        String s = rs.getString("b");
        float f = rs.getFloat("c");
    }
}
```

Στο παραπάνω παράδειγμα χρησιμοποιείται το αντικείμενο DriverManager το οποίο κάνει τη σύνδεση με τη Βάση, το αντικείμενο Statement το οποίο μεταφέρει το SQL ερώτημα στη Βάση και το ResultSet αντικείμενο το οποίο ανακτά τα αποτελέσματα από τα ερωτήματά μας και εκτελεί ένα απλό while loop για την ανάκτηση και εμφάνιση αυτών των αποτελεσμάτων.

Το JDBC API χρησιμοποιεί JDBC drivers για να συνδεθεί με τη βάση. Υπάρχουν αρκετές βιβλιοθήκες που υλοποιούν το API. Συνήθως κάθε σύστημα διαχείρισης βάσεων δεδομένων (MySQL, Oracle, SQL Server) δίνει και μία υλοποίηση αυτού. Συνεπώς με την εγκατάσταση του εξυπηρετητή πρέπει να επιλεγεί και κατάλληλα η JDBC βιβλιοθήκη.

Τι είναι το API (Application Programming Interface)

API (Application Program Interface) είναι ένα σύνολο από πρωτόκολλα και εργαλεία που καθορίζει το πως δύο κομμάτια λογισμικού να επικοινωνούν μεταξύ τους. Ένα δημόσιο API συγκεκριμένα είναι ένα API το οποίο επιτρέπει την πρόσβαση των προγραμματιστών στα δεδομένα μιας διαδικτυακής υπηρεσίας ή εφαρμογής. Είναι ένα έγγραφο τεκμηρίωσης που περιέχει την περιγραφή όλων των χαρακτηριστικών ενός προϊόντος ή λογισμικού ή βιβλιοθήκης. Αναπαραστά κλάσεις και διεπαφές που ακολουθούν τα προγράμματα λογισμικού για να επικοινωνούν μεταξύ τους.

Δημιουργήσουμε ένα API για εφαρμογές, βιβλιοθήκες, λειτουργικά συστήματα, κ.α.

Οι προδιαγραφές αυτών των API καθορίζονται σύμφωνα με πολλούς διαφορετικούς οργανισμούς στον κόσμο (Alljoyn, OSGi, Eclipse, JCP, ESR, κ.λπ.).

Ένας JDBC driver είναι ένα κομμάτι λογισμικού που επιτρέπει σε μια java εφαρμογή να αλληλοεπιδράσει με τη βάση δεδομένων. Υπάρχουν 4 τύποι JDBC drivers:

1. JDBC-ODBC bridge driver
2. Native-API driver
3. Network Protocol driver
4. Thin driver

Σύνδεση με τη Βάση Δεδομένων

Για να συνδεθεί μια java εφαρμογή με τη βάση δεδομένων πρέπει να γίνουν 5 βήματα:

1. Καταγραφή της driver class
2. Δημιουργία σύνδεσης
3. Δημιουργία statement
4. Εκτέλεση ερωτημάτων
5. Κλείσιμο σύνδεσης

Για να γίνει καταγραφή της κλάσης χρησιμοποιείται η μέθοδος `forName()`. Αυτή η μέθοδος φορτώνει τη driver class. Η σύνταξη της `forName()` μεθόδου είναι:

`public static void forName(String className) throws ClassNotFoundException.`

Για να γίνει η σύνδεση με τη βάση δεδομένων χρησιμοποιείται η μέθοδος `getConnection()`.

Η σύνταξη της `getConnection()` μεθόδου είναι:

```
public static Connection getConnection(String url) throws SQLException
```

```
public static Connection getConnection(String url, String name,
```

```
String password) throws SQLException
```

Για τη δημιουργία ενός statement χρησιμοποιούμε τη μέθοδο `createStatement()`.

Η σύνταξη της `createStatement()` μεθόδου είναι:

```
public Statement createStatement() throws SQLException
```

Για να εκτελέσουμε ένα ερώτημα στη βάση δεδομένων χρησιμοποιούμε τη μέθοδο `executeQuery()`.

Η σύνταξη της μεθόδου είναι:

```
public ResultSet executeQuery(String sql) throws SQLException
```

Παράδειγμα που εκτελεί ένα ερώτημα:

```
ResultSet rs=stmt.executeQuery("select * from emp");
```

```
while(rs.next()){ System.out.println(rs.getInt(1)+"
```

```
 "+rs.getString(2)); }.
```

Για να κλείσουμε τη σύνδεση χρησιμοποιούμε τη μέθοδο `close()`. Η σύνταξη της μεθόδου `close()` είναι, `public void close()throws SQLException`

Παράδειγμα `con.close()`

Για να συνδεθεί μια εφαρμογή με τη mysql βάση δεδομένων πρέπει να γνωρίζουμε τα παρακάτω στοιχεία:

1) Driver class: Η driver class για τη mysql είναι `com.mysql.jdbc.Driver`.

2) Connection URL: Το connection url για τη mysql ΒΔ είναι jdbc:mysql://localhost:3306/databasename, όπου jdbc είναι το API, mysql είναι η βάση δεδομένων, localhost είναι ο server στον οποίο τρέχει η mysql(μπορούμε να βάλουμε και IP διεύθυνση), 3306 είναι η θύρα και databasename είναι το όνομα της βάσης δεδομένων.

3) Username: Το username που χρησιμοποιούμε για τη mysql ΒΔ (π.χ. root).

4) Password: Είναι ο κωδικός που έχουμε βάλει κατά την εγκατάσταση της mysql βάσης δεδομένων.

Παράδειγμα:

Δημιουργούμε μια ΒΔ company και τον πίνακα emp.

```
create database company;
```

```
use sonoo;
```

```
create table emp(id int(10),name varchar(40),age int(3));
```

Σύνδεση της java εφαρμογής με τη mysql ΒΔ

Το παραπάνω παράδειγμα θα επιστρέψει και εκτυπώνει όλες τις εγγραφές του πίνακα emp.

Για να γίνει η σύνδεση της java εφαρμογής με τη mysql ΒΔ, πρέπει να φορτωθεί το αρχείο mysqlconnector.jar.

Υπάρχουν δύο τρόποι για να φορτωθεί το αρχείο:

1. Επικόλληση στο φάκελο jre/lib/ext όπου βρίσκονται οι jar βιβλιοθήκες
2. Ορισμός της διαδρομής αρχείου του jar στο classpath.

DriverManager class

Η κλάση DriverManager λειτουργεί ως διεπαφή μεταξύ χρηστών και drivers.

Ελέγχει ποιοι drivers είναι διαθέσιμοι και διαχειρίζεται μια σύνδεση μεταξύ της βάσης και του κατάλληλου driver.

Οι κυριότερες μέθοδοι που χρησιμοποιεί η κλάση είναι:

1. `public static void registerDriver(Driver driver)`
2. `public static void deregisterDriver(Driver driver)`
3. `public static Connection getConnection(String url)`
4. `public static Connection getConnection(String url, String userName,String password).`

Connection Interface

Η σύνδεση γίνεται μεταξύ της java εφαρμογής και της βάσης δεδομένων.

Οι κυριότερες μέθοδοι είναι:

1. `public Statement createStatement()`
2. `public Statement createStatement(int resultSetType,int resultSetConcurrency)`
3. `public void setAutoCommit(boolean status)`
4. `public void commit()`
5. `public void rollback()`
6. `public void close()`

Statement Interface

Το Statement interface παρέχει μεθόδους οι οποίες εκτελούν ερωτήματα στη βάση δεδομένων.

Οι κυριότερες μέθοδοι που χρησιμοποιούνται είναι:

1. `public ResultSet executeQuery(String sql)`
2. `public int executeUpdate(String sql)`
3. `public boolean execute(String sql)`

4. public int[] executeBatch()

Παράδειγμα (εισαγωγή, ενημέρωση και διαγραφή μιας εγγραφής)

```
import java.sql.*;
class FetchRecord{
public static void main(String args[])throws Exception{

    Class.forName("com.mysql.jdbc.Driver ");

    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/company","username","password");
    Statement stmt=con.createStatement();

    //stmt.executeUpdate("insert into emp values(33,'Papadopoulos',15000)");
    //int result=stmt.executeUpdate("update emp set name='Papadopoulos', salary=10000 where
id=33");
    int result=stmt.executeUpdate("delete from emp where id=33");
    System.out.println(result+" records affected");
    con.close();
}}
```

PreparedStatement Interface:

Η συγκεκριμένη διεπαφή εκτελεί παραμετροποιημένα ερωτήματα.

Σχεδίαση και ανάπτυξη διαδικτυακής εφαρμογής με τη χρήση της PHP

Η PHP είναι μια γλώσσα προγραμματισμού **script** που από την πλευρά του διακομιστή, σχεδιασμένη ειδικά για το διαδίκτυο.

Είναι τόσο ισχυρή που χρησιμοποιείται από το μεγαλύτερο σύστημα blog στο διαδίκτυο που είναι το WordPress αλλά και από το Facebook.

Ένα αρχείο PHP μπορεί να περιέχει κείμενο, κώδικα HTML, CSS, JavaScript και PHP, με τον κώδικα να εκτελείται στον server και το αποτέλεσμα να επιστρέφει στον browser ως απλές σελίδες HTML. Ο PHP κώδικας διαφοροποιείται από το υπόλοιπο HTML έγγραφο με τις ετικέτες (tags) αρχής. Μεταφράζεται στην πλευρά του διαδικτυακού διακομιστή και δημιουργεί HTML ή άλλη έξοδο, την οποία θα δει ο

επισκέπτης. Ο επισκέπτης θα δει μόνο την HTML έξοδο που παράγει η PHP από την πλευρά του διακομιστή και δεν χρειάζεται να έχει εγκατεστημένη την PHP στον δικό του υπολογιστή.

Είναι ένα προϊόν ανοιχτού κώδικα, γεγονός που σημαίνει ότι διανέμεται δωρεάν και επίσης έχουμε πρόσβαση στον κώδικα προέλευσής του. Έτσι, μπορούμε να τον τροποποιήσουμε και να τον επαναχρησιμοποιήσουμε, σύμφωνα με τις ανάγκες και τις απαιτήσεις μας.

Ένα site αποκτά διαδραστικότητα π.χ: η εισαγωγή ενός username και ενός κωδικού σε μια σελίδα και η επαλήθευσή τους ώστε ο χρήστης να συνδεθεί σε αυτό το site –

Ο PHP κώδικας διαφοροποιείται από το υπόλοιπο HTML έγγραφο με τις ετικέτες (tags) αρχής .

Η PHP δημιουργήθηκε το 1994 και αρχικά σήμαινε Personal Home Page (προσωπική αρχική σελίδα), όταν ο Rasmus Lerdorf έγραψε αρκετά CGI προγράμματα σε C, τα οποία χρησιμοποιούσε για να συντηρεί το προσωπικό του site. Τα επέκτεινε για να δουλεύουν με φόρμες και να επικοινωνούν με βάσεις δεδομένων, και αυτή την υλοποίηση την ονόμασε Personal Home Page/Forms Interpreter – PHP/FI.

Πλέον PHP είναι το ακρωνύμιο για το Hypertext Preprocessor. Η γλώσσα εξελίσσεται και νέες εκδόσεις της βγαίνουν ανά τακτά χρονικά διαστήματα. Σύμφωνα όμως με τη σύμβαση GNU (General Public License) άλλαξε σημαίνει PHP Hypertext Preprocessor (προεπεξεργαστής υπερκειμένου PHP). Από τη δημιουργία της η PHP δέχτηκε πολλές τροποποιήσεις και βελτιώσεις. Το βασικό πλεονέκτημα της PHP είναι ότι λειτουργεί δυναμικά.

Αυτό σημαίνει ότι τα αποτελέσματα που παράγει, αλλάζουν σύμφωνα με τις ανάγκες του χρήστη.

Ωστόσο, ο δυναμικός τρόπος λειτουργίας δεν παύει να εφαρμόζεται ακόμα και μέσα στο εσωτερικό της PHP.

Για παράδειγμα, έχει τη δυνατότητα να αλλάζει τον τύπο των μεταβλητών δυναμικά, σύμφωνα με τα δεδομένα που κάθε χρονική στιγμή είναι αποθηκευμένα σε αυτές.

Τι άλλο μπορεί να κάνει η PHP

Μπορεί να δημιουργήσει, να ανοίξει, να κλείσει, να διαβάσει, να γράψει και να διαγράψει αρχεία σε έναν server. Ακόμα μπορεί να στείλει και να λάβει cookies, επίσης να προσθέσει, να διαγράψει και να τροποποιήσει δεδομένα στη Βάση Δεδομένων σας. Τέλος μπορεί ακόμη και να κρυπτογραφήσει δεδομένα.

Δεν περιορίζει η PHP στο να παράγει HTML.

Μπορεί να παράγει εικόνες, αρχεία PDF, ακόμα και ταινίες Flash, καθώς και οποιοδήποτε κείμενο όπως XHTML και XML.

Σε σύγκριση με τους βασικούς της ανταγωνιστές (Perl, ASP και JSP), η PHP έχει πολλά πλεονεκτήματα που την καθιστούν επικρατούσα.

- Υψηλή απόδοση. Είναι πολύ αποτελεσματική.
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων. Έχει εγγενείς συνδέσεις για πολλά συστήματα βάσεων δεδομένων. Εκτός από τη MySQL, μπορούμε να συνδεθούμε κατευθείαν με πολλές βάσεις δεδομένων μερικές από τις οποίες είναι οι mSQL, Oracle, Hyperwave, Informix, InterBase, filePro και πολλές άλλες. Χρησιμοποιώντας Open Database Connectivity Standard (ODCB), μπορούμε να συνδεθούμε σε οποιαδήποτε βάση δεδομένων παρέχει ένα πρόγραμμα οδήγησης ODCB.
- Έχει ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες διαδικασίες διαδικτύου. Επειδή σχεδιάστηκε για να χρησιμοποιείται στο διαδίκτυο, έχει πολλές ενσωματωμένες βιβλιοθήκες, που εκτελούν πολλές χρήσιμες λειτουργίες σχετικές με το Web. Μπορούμε να δημιουργούμε εικόνες GIF δυναμικά, να συνδεθούμε με άλλες υπηρεσίες δικτύων, να στείλουμε ηλεκτρονικό ταχυδρομείο, να δουλέψουμε με cookies και συνόδους και να δημιουργήσουμε PDF έγγραφα. Και αυτές είναι μόνο ελάχιστες από τις απεριόριστες δυνατότητες της PHP!
- Χαμηλό κόστος. Η PHP παρέχεται δωρεάν.

- Ευκολία μάθησης και χρήσης. Η σύνταξη της PHP βασίζεται σε άλλες γλώσσες προγραμματισμού, βασικά στη C και στην Perl.
- Φορητότητα. Είναι διαθέσιμη για πολλά λειτουργικά συστήματα. Μπορούμε να γράψουμε κώδικα για πολλές διαφορετικές εκδόσεις των Microsoft Windows και πολλά συστήματα τύπου Unix.
- Διαθεσιμότητα του κώδικα προέλευσης. Έχουμε πρόσβαση στον κώδικα προέλευσης της PHP, αντίθετα με εμπορικά κλειστά συστήματα. Επίσης, μπορούμε να διορθώσουμε ή να μεταβάλλουμε τον κώδικά της, προσαρμόζοντάς τον στις ανάγκες μας.

Όλα αυτά έχουν κάνει την PHP να είναι ένα πανίσχυρο και ευρέως εργαλείο χρησιμοποιούμενο παγκοσμίως.

Γιατί να χρησιμοποιήσει κανείς PHP

Η PHP τρέχει σε όλες τις πλατφόρμες όπως Windows, Linux, Unix, Mac OS X κ.λπ. Είναι συμβατή με σχεδόν όλους τους server που χρησιμοποιούνται σήμερα (Apache, IIS, κ.λπ.) Υποστηρίζει ένα ευρύ φάσμα απο Βάσεις Δεδομένων, είναι δωρεάν, είναι εύκολη να τη μάθει κανείς και τρέχει αποτελεσματικά στους server.

Η χρήση της MySQL Βάσης Δεδομένων

Είναι ένα πολύ γρήγορο και δυνατό σύστημα διαχείρισης βάσεων δεδομένων. Μια βάση δεδομένων μας επιτρέπει να αποθηκεύουμε, να αναζητάμε, να ταξινομούμε και να ανακαλούμε τα δεδομένα αποτελεσματικά. Ουσιαστικά αποτελεί τη “δεξαμενή” από την οποία αντλούμε τα δεδομένα.

Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα μας, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα, για να παρέχει γρήγορη πρόσβαση και να διασφαλίσει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση.

Συνεπώς, η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών.

Χρησιμοποιεί την SQL (δομημένη γλώσσα ερωτημάτων), την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια.

Η SQL είναι η πιο τυπική γλώσσα για πρόσβαση σε συστήματα διαχείρισης βάσεων δεδομένων (DBMS). Χρησιμοποιείται για να αποθηκεύσουμε και να ανακαλέσουμε δεδομένα από και προς μια βάση δεδομένων. Χρησιμοποιείται σε πολλές βάσεις δεδομένων όπως η *Oracle*, *PostgreSQL*, *Sybase*, *Microsoft SQL Server* και φυσικά στη *MySQL*. Υπάρχει μια ANSI (Αμερικάνικο εθνικό πρότυπο) τυποποίηση για την SQL και τα συστήματα βάσεων δεδομένων, όπως η *MySQL*, χρησιμοποιούν αυτή την τυποποίηση.

Έχει συστατικά στοιχεία της, μια γλώσσα ορισμού βάσεων δεδομένων (DDL) και μια γλώσσα χειρισμού ερωτημάτων (DML). Χρησιμοποιούμε DDL όταν αρχικά διαμορφώνουμε μια βάση δεδομένων ενώ DML όταν αποθηκεύουμε και ανακαλούμε πραγματικά δεδομένα από τη βάση μας.

Οι σχεσιακές βάσεις δεδομένων σχεδιάστηκαν για να αντιμετωπίσουν όλα εκείνα τα προβλήματα που προέρχονταν από τη χρήση επίπεδων αρχείων (απλών αρχείων κειμένου) για την αποθήκευση και ανάκληση δεδομένων.

Η χρήση των επίπεδων αρχείων είναι κάτι πολύ εύκολο και απλό, ωστόσο πολύ περιοριστικό. Για τον λόγο αυτό χρησιμοποιούνται σε μερικές μόνο περιπτώσεις όπου ο όγκος των δεδομένων είναι μικρός.

Οι σχεσιακές βάσεις δεδομένων ή καλύτερα τα σχεσιακά συστήματα διαχείρισης βάσεων δεδομένων (RDBMS) από την άλλη, προσφέρουν πολλά περισσότερα πλεονεκτήματα:

- Παρέχουν γρηγορότερη πρόσβαση στα δεδομένα. Συγκεκριμένα η MySQL έχει από τις γρηγορότερες επιδόσεις που υπάρχουν στα RDBMS.
- Μπορούν εύκολα να γίνουν ερωτήματα για να εξαχθούν σύνολα από δεδομένα που ανταποκρίνονται σε κάποια κριτήρια.
- Έχουν ενσωματωμένους μηχανισμούς για αντιμετώπιση ταυτόχρονης πρόσβασης.
- Παρέχουν τυχαία πρόσβαση στα δεδομένα.
- Έχουν ενσωματωμένα συστήματα δικαιωμάτων

Η MySQL είναι λογισμικό ανοιχτού κώδικα, το οποίο προσφέρεται δωρεάν και επιπλέον μαζί με το εκτελέσιμο πρόγραμμα διατίθεται και ο πηγαίος κώδικας όλων των συστατικών του, για να μπορεί κάποιος να τα τροποποιήσει και να τα χρησιμοποιήσει ανάλογα με τις ανάγκες του.

Δυναμικές ιστοσελίδες με PHP & MySQL

Οι πρώτες δυναμικές ιστοσελίδες ξεκίνησαν χρησιμοποιώντας σενάρια εντολών σε γλώσσα προγραμματισμού **C** ή **Perl** τα οποία καλούνταν από την ιστοσελίδα μέσω σεναρίων εντολών **CGI (Common Gateway Interface)**.

Ο σχεδιασμός δυναμικών ιστοσελίδων με τη μορφή αυτή δεν ήταν εύκολη υπόθεση και για αυτό το λόγο αναπτύχθηκαν τεχνολογίες ανάπτυξης ιστοσελίδων όπως για παράδειγμα η εμπορική τεχνολογία **ASP.NET (της Microsoft)** ή τεχνολογίες ανοικτού κώδικα όπως της γλώσσας **PHP & MySQL** που παρουσιάζονται στον οδηγό αυτό.

Η ιδέα είναι ότι ο διαχειριστής της ιστοσελίδας χρησιμοποιεί σενάρια εντολών σε γλώσσα **PHP** τα οποία αποθηκεύουν το περιεχόμενο της ιστοσελίδας σε μια βάση δεδομένων **MySQL**.

Ο χρήστης όταν συνδέεται στον απομακρυσμένο διακομιστή ιστοσελίδων καλεί ένα σενάριο εντολών **php** το οποίο χρησιμοποιώντας δεδομένα από τον σκληρό δίσκο (συνήθως πολυμεσικά αρχεία - εικόνες - ήχους) και την βάση **MySQL** δημιουργεί την σελίδα δυναμικά. Στις δυναμικές ιστοσελίδες αν ο διαχειριστής θέλει να αλλάξει



την γενική γραφική παρουσίαση, απλά τροποποιεί το κώδικα php και όλη η ιστοσελίδα αποκτά διαφορετική εμφάνιση (σε αντίθεση με τις στατικές ιστοσελίδες όπου πρέπει να αλλάξουν όλες οι τοπικά αποθηκευμένες σελίδες html).

Για την εκμάθηση της γλώσσας **PHP** και της βάσης **MySQL** για τον σχεδιασμό ιστοσελίδων προτείνουμε την εγκατάσταση τοπικά στον υπολογιστή ενός εξυπηρετητή ιστοσελίδων Apache της βάσης MySQL και της γλώσσας PHP. Επίσης προτείνεται και η εγκατάσταση του εργαλείου **phpMyAdmin** για την διαχείριση των βάσεων δεδομένων MySQL μέσα από το περιβάλλον του φυλλομετρητή.



Το λογότυπο του XAMPP

Η εγκατάσταση των παραπάνω εργαλείων μπορεί να γίνει χειροκίνητα για το κάθε εργαλείο ή να εγκατασταθεί μέσω λογισμικών που τα εγκαθιστούν και τα ρυθμίζουν με αυτόματο τρόπο.

Ένα τέτοιο λογισμικό είναι το **XAMPP** το οποίο περιέχει: **Apache HTTP** εξυπηρετητή, **MySQL**, **PHP** και **Perl** και είναι ανεξάρτητο πλατφόρμας (τρέχει σε **Windows, Linux** αλλά και **Mac**).

Η γλώσσα σεναρίων **PHP** ή η βάση **MySQL** μπορεί να εγκατασταθεί αυτόνομα. Για παράδειγμα η γλώσσα σεναρίων **PHP** μπορεί αυτόνομα να χρησιμοποιηθεί ως γλώσσα προγραμματισμού σεναρίων τα οποία τρέχουν αυτόνομα σε επίπεδο γραμμής εντολών.

Στην παρακάτω ενότητα περιγράψουμε την εγκατάσταση τοπικά του εξυπηρετητή ιστοσελίδων **Apache** καθώς και της εγκατάστασης της **PHP** σε αυτόν ως ξεχωριστό module μαζί με την βάση MySQL.



Κάνουμε επίσης εγκατάσταση στο phpMyAdmin το οποίο είναι ένα λογισμικό σε γλώσσα php για την διαχείριση βάσεων MySQL κατευθείαν από τον φυλλομετρητή του χρήστη-πελάτη. Η διαχείριση των βάσεων μέσω του phpMyAdmin είναι ευκολότερη από την διαχείριση αυτών μέσω των εργαλείων σε επίπεδο γραμμής εντολών.

Εγκατάσταση PHP & MySQL στο Ubuntu

Εδώ θα δούμε πως χειροκίνητα θα γίνει η εγκατάσταση του PHP, MySQL και phpMyAdmin στο λειτουργικό σύστημα Ubuntu.

Η εγκατάσταση αυτή έχει τσεκαριστεί ότι δουλεύει στο Ubuntu 10.04 (LTS) και στο Ubuntu 12.04 (LTS).

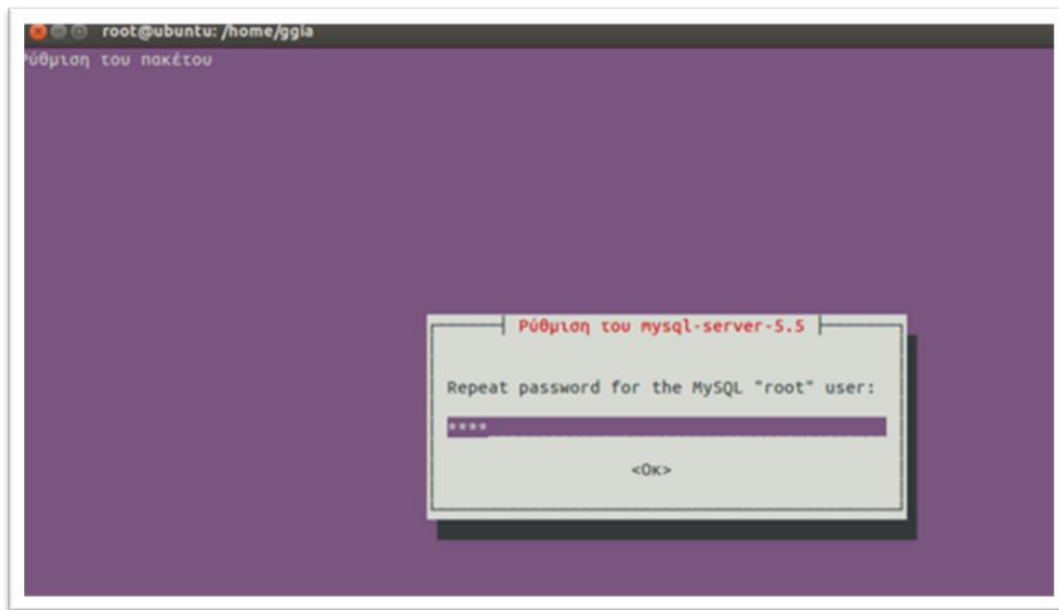
Αρχικά Ανοίγουμε το τερματικό του Ubuntu και τρέχουμε τις παρακάτω εντολές:

- `sudo su`
- `apt-get install aptitude` (στην περίπτωση που δεν είναι εγκατεστημένο το aptitude)
- `aptitude install mysql-server mysql-client` (εγκατάσταση του εξυπηρετητή βάσης MySQL)

Στην συνέχεια βάζουμε κωδικό για τον διαχειριστή root της βάσης MySQL

artitude install apache2 Γίνεται εγκατάσταση του εξυπηρετητή ιστοσελίδων Apache: Περαιτέρω αρχεία ρυθμίσεως βρίσκονται στους υποκαταλόγους /etc/apache2 και στο /etc/apache2/mods-enabled (για τα Apache modules), /etc/apache2/sites-enabled (για virtual hosts) αλλά και το /etc/apache2/conf.d.

Για να ελέγξουμε ότι ο εξυπηρετητής Apache λειτουργεί ανοίγουμε την σελίδα



<http://localhost> ή την σελίδα <http://127.0.0.1>. Εμφανίζεται ένα μήνυμα ότι ο εξυπηρετητής Apache είναι ενεργός (online) και λειτουργεί.

Βάζουμε ξανά τον κωδικό για τον διαχειριστή root της βάσης MySQL.

artitude install php5 libapache2-mod-php5 Εγκατάσταση γλώσσας PHP.

Ελέγξουμε για την σωστή εγκατάσταση της PHP φτιάξουμε το αρχείο info.php με το παρακάτω περιεχόμενο και το βάζουμε στον κατάλογο /var/www.

```
<?php
```

```
phpinfo();
```

```
?>
```

Στην διεύθυνση <http://localhost/info.php> ή το <http://127.0.0.1/info.php> μπορούμε να δούμε πληροφορίες για την εγκατεστημένη PHP. Η συνάρτηση phpinfo() μέσα

στο παραπάνω μινιμαλιστικό σενάριο εντολών (script) εμφανίζει τις παραμέτρους του συστήματος.

`/etc/init.d/apache2 restart` - Δίνουμε εντολή να κάνει επανεκκίνηση ο εξυπηρετητής Apache (για να ενημερωθεί με τις αλλαγές στην εγκατάσταση της PHP).

`aptitude search php5` - Βλέπουμε τα διαθέσιμα modules για την php5 - τα οποία μπορούμε να επιλέξουμε για εγκατάσταση.

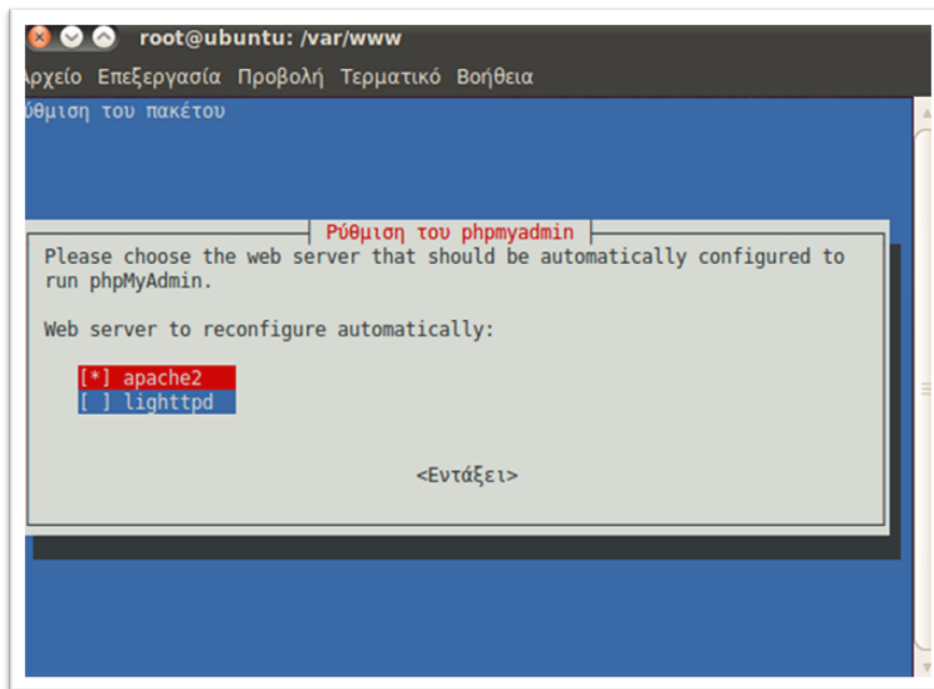
`aptitude install php5-mysql php5-curl php5-gd php5-idn php-pear php5-imagick php5-imap php5-mcrypt php5-memcache php5-mhash php5-ming php5-ps php5-pspell php5-recode php5-snmp php5-sqlite php5-tidy php5-xmlrpc php5-xsl php5-json` (επιλογή για εγκατάσταση των συγκεκριμένων modules).

PHP Version 5.3.2-1ubuntu4.14	
System	Linux ubuntu 2.6.32-41-generic #88-Ubuntu SMP Thu Mar 29 13:08:43 UTC 2012 i686
Build Date	Feb 11 2012 06:38:33
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/curl.ini, /etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/idn.ini, /etc/php5/apache2/conf.d/imagick.ini, /etc/php5/apache2/conf.d/imap.ini, /etc/php5/apache2/conf.d/mcrypt.ini, /etc/php5/apache2/conf.d/memcache.ini, /etc/php5/apache2/conf.d/ming.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini, /etc/php5/apache2/conf.d/pdo_sqlite.ini, /etc/php5/apache2/conf.d/ps.ini, /etc/php5/apache2/conf.d/pspell.ini, /etc/php5/apache2/conf.d/recode.ini, /etc/php5/apache2/conf.d/snmp.ini, /etc/php5/apache2/conf.d/sqlite.ini, /etc/php5/apache2/conf.d/sqlite3.ini, /etc/php5/apache2/conf.d/tidy.ini, /etc/php5/apache2/conf.d/xmlrpc.ini, /etc/php5/apache2/conf.d/xsl.ini
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626

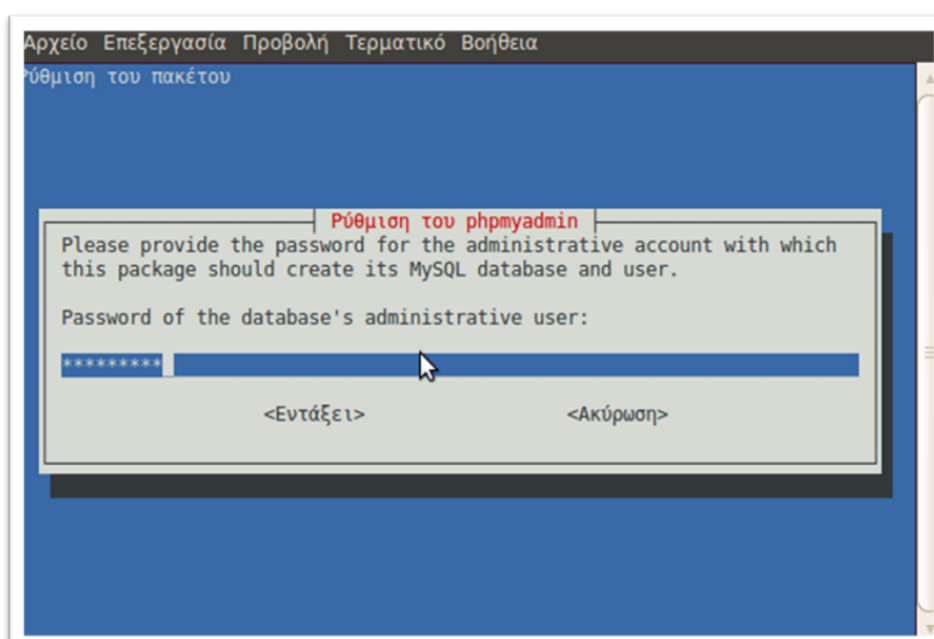
Ανοίγουμε την διεύθυνση `http://localhost/info.php` ή το `http://127.0.0.1/info.php` και εμφανίζονται οι παράμετροι και τα εγκατεστημένα modules της PHP.

`/etc/init.d/apache2 restart` (επανεκκίνηση εξυπηρετητή ιστοσελίδων Apache - για να ενεργοποιηθούν τα νέα modules)

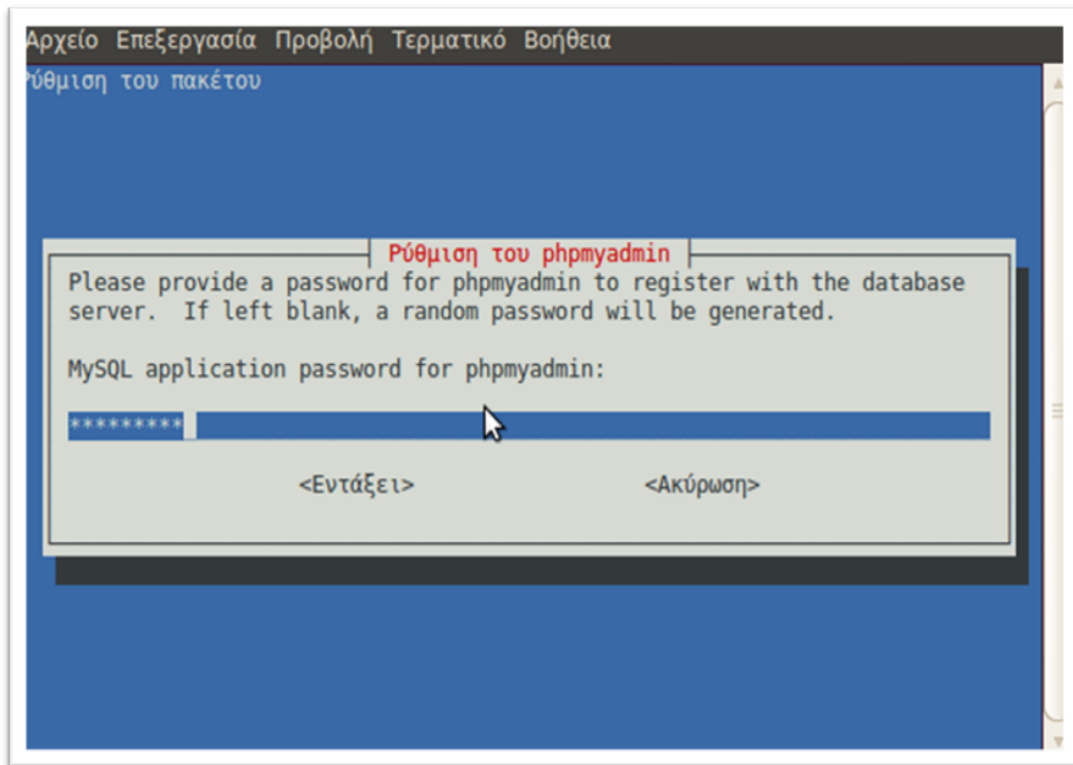
`aptitude install phpmyadmin`



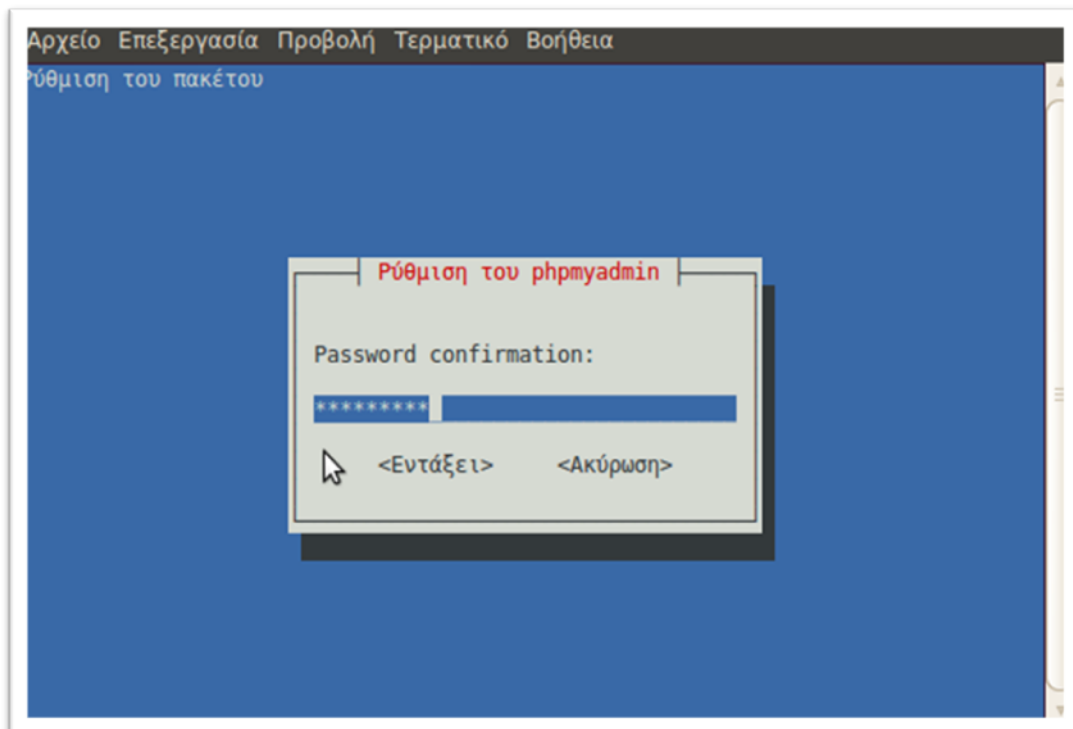
Επιλέγουμε την αυτόματη ρύθμιση του εξυπηρετητή ιστοσελίδων Apache.



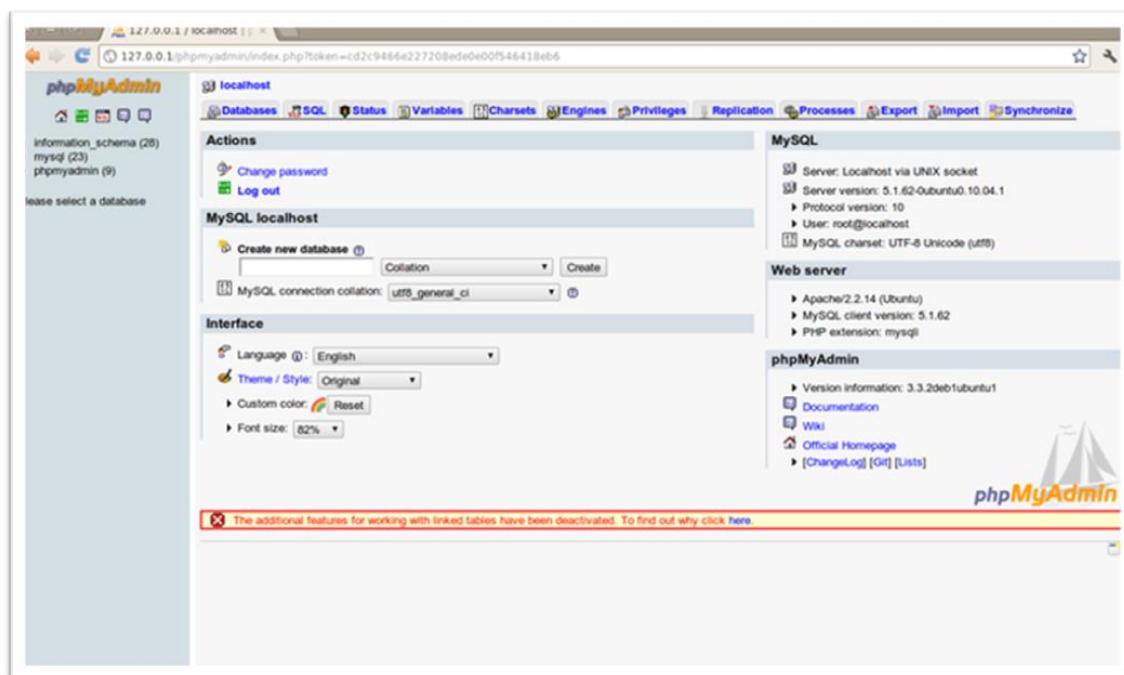
Δίνουμε τον κωδικό **root** του εξυπηρετητή βάσης MySQL (που έχουμε επιλέξει στην εγκατάσταση της MySQL).



Ρυθμίζουμε τον κωδικό πρόσβασης στο περιβάλλον myPφrAd min.



Δίνουμε πάλι τον κωδικό πρόσβασης στο περιβάλλον myPherAdmin που μόλις δώσαμε.



Εγκατεστημένο το phpMyAdmin

Πως συνδέομαι με την βάση μου μέσω PHP?

Η γλώσσα προγραμματισμού **PHP (Personal Home Pages)** ξεκίνησε αρχικά σαν μια *scripting* γλώσσα, με την εξέλιξη όμως του διαδικτύου και το πέρασμα των ετών έχει μετεξελιχθεί σε μία πλήρη προδιαγραφών (Full Stack) γλώσσα με βασικό της άξονα εξέλιξης την αντικειμενοστρέφεια (Object Oriented).

Άμεσες είναι οι συνέπειες του αντικειμενοστραφούς προγραμματισμού σε όλες τις διεπαφές που δημιουργεί η γλώσσα με συνεργαζόμενες υπηρεσίες όπως η **MySQL**. Κατά συνέπεια έχουμε την εμφάνιση της **PDO (PHP Data Objects)** διεπαφής για την **MySQL** και φυσικά της ήδη παλιάς διεπαφής **MySQLi** που είναι επίσης **object oriented**, υποστηρίζει όμως και παλαιού - διαδικασιακού (procedural) τύπου συνδέσεις .

Η υποστήριξη της παλαιότερου τύπου διεπαφής (απλό MySQL connection) έχει πάψει από το 2012, οπότε συνιστούμε τη χρήση των πιο νέων προτύπων MySQLi και PDO.

Σύγκριση PDO με MySQLi

Η βιβλιοθήκη PDO είναι αμιγώς αντικειμενοστραφής και επειδή το μέλλον της PHP είναι κοντά στις object oriented προσεγγίσεις, σίγουρα η χρήση της PDO είναι μία εγγυημένη λύση μακροπρόθεσμα.

Επίσης διαθέτει δυνατότητα σύνδεσης με 12 διαφορετικά συστήματα διαχείρισης βάσεων δεδομένων εν αντιθέσει με την MySQLi προσέγγιση που συνεργάζεται μόνο με την MySQL.

Η βιβλιοθήκη MySQLi (το i σημαίνει improved) όπως προαναφέρθηκε διαθέτει και μια διαδικασιακή διεπαφή (procedural API), οπότε κάνει αφενός πιο εύκολη την μετάβαση από παλαιού τύπου απλές MySQL συνδέσεις σε συνδέσεις MySQLi, αφετέρου δε η procedural διεπαφή απέχει από τη φιλοσοφία του μέλλοντος, οπότε αργά ή γρήγορα ενδεχομένως να χρειαστεί να αλλάξετε εκ νέου τις οποίες procedural προσεγγίσεις υλοποιήσετε.

Τόσο η PDO όσο και η MySQLi βιβλιοθήκες υποστηρίζουν Prepared Statements οι οποίες βοηθούν στην προστασία της εφαρμογής σας από SQL injections και είναι ιδιαίτερα σημαντικές για την ασφάλεια της διαδικτυακής σας εφαρμογής.

Δημιουργία σύνδεσης στην MySQL

Παρακάτω περιγράφονται 3 τρόποι σύνδεσης στην MySQL. Οι δύο αφορούν την MySQLi (Object Oriented και Procedural) και ο τρίτος την PDO.

1) MySQLi αντικειμενοστραφής (Object Oriented)

```
$host = "localhost";
```

```
$username = "username";
```

```
$password = "password"
```

```
// Δημιουργία σύνδεσης

$connection = new mysqli($host, $username, $password);

// Έλεγχος σύνδεσης

if ($connection->connect_error) {

    die("Connection failed: " . $connection->connect_error);

}

echo "Connected successfully";

$connection->close(); // Κλείσιμο σύνδεσης

?>
```

2) MySQLi διαδικασιακή (Procedural)

```
$host = "localhost";

$username = "username";

$password = "password";

// Δημιουργία σύνδεσης

$connection = mysqli_connect($host, $username, $password);
```

```
// Έλεγχος σύνδεσης

if (!$connection) {

    die("Connection failed: " . mysqli_connect_error());

}

echo "Connected successfully";

mysqli_close($connection); // Κλείσιμο σύνδεσης

?>
```

3) PDO

```
$host = "localhost";

$username = "username";

$password = "password";

try {

    $connection = new PDO("mysql:host=$host;dbname=myDB", $username,

$password);

// set the PDO error mode to exception

    $connection->setAttribute(PDO::ATTR_ERRMODE,
```

```

PDO::ERRMODE_EXCEPTION);

    echo "Connected successfully";

}

catch(PDOException $e)

{

    echo "Connection failed: " . $e->getMessage();

}

$conn = null; // Κλείσιμο σύνδεσης

?>

```

Αξίζει να σημειωθεί ότι στο τέλος κάθε *script* κλείνουμε τη σύνδεση στη MySQL για να απελευθερώνονται οι πόροι που δεσμεύει η σύνδεση.

Ο συνδυασμός της τεχνολογίας ή γλώσσας προγραμματισμού **PHP (Hypertext PreProcessor)** με τη βάση δεδομένων **MySQL** αποτελεί ένα από τα ισχυρότερα εργαλεία ανάπτυξης δυναμικών εφαρμογών στον παγκόσμιο ιστό (Web).

Με τη χρήση τους μπορούμε να δημιουργήσουμε από απλές εφαρμογές που να περιέχουν μια φόρμα παραγγελίας έως και ασφαλείς τοποθεσίες ηλεκτρονικού εμπορίου.

Όπως ήδη γνωρίζουμε, μια ιστοσελίδα που είναι γραμμένη σε απλή HTML είναι κατά βάση στατική, δηλ. τα περιεχόμενά της δεν αλλάζουν αυτόματα και δεν προσαρμόζονται στις απαιτήσεις του χρήστη παρά μόνο αν αποφασίσει να τα ενημερώνει τακτικά ο δημιουργός της ιστοσελίδας, πράγμα όχι πάντα εφικτό.

Ο συνδυασμός, όμως, της γλώσσας προγραμματισμού PHP και της βάσης δεδομένων MySQL, μπορεί να κάνει τις ιστοσελίδες μας να είναι δυναμικές, δηλ. να

ενημερώνονται αυτόματα και να προσαρμόζονται στις απαιτήσεις του χρήστη,
χωρίς να χρειασθεί να κάνει καμία απολύτως ενημέρωση ο δημιουργός των
ιστοσελίδων.

*Οι πιο συνηθισμένες εφαρμογές που μπορούμε να επιτύχουμε με τη χρήση των
παραπάνω τεχνολογιών είναι οι εξής :*

- Έλεγχος ταυτότητας χρήστη.
- Καλάθι αγορών (shopping charts).
- Σύστημα Διαχείρισης Περιεχομένου (CMS, Content Management System).
- Web mail.
- Web forum.
- GuestBook

PHP

Η PHP (" PHP: **Hypertext Preprocessor**", ακρωνύμιο του Hypertext Preprocessor)
είναι μια ευρέως χρησιμοποιούμενη ανοιχτού κώδικα γλώσσα δέσμης ενεργειών
πολλαπλών χρήσεων που μπορεί να ενσωματωθεί σε **HTML**, ειδικά για ανάπτυξη
ιστού. Ε αντίθεση με την **JavaScript** από την πλευρά του πελάτη, ο κώδικας PHP
εκτελείται στον διακομιστή. Εάν ο κώδικας όπως το παραπάνω παράδειγμα είναι
χτισμένος στον διακομιστή, μετά την εκτέλεση του σεναρίου, οι πελάτες θα
μπορούν να λάβουν τα αποτελέσματά του, αλλά δεν έχουν τρόπο να γνωρίζουν πώς
λειτουργεί ο κώδικας πίσω από αυτό. Ο διακομιστής **web** μπορεί ακόμη και να
ρυθμιστεί ώστε να επιτρέπει στην **PHP** να χειρίζεται όλα τα αρχεία **HTML**, έτσι ώστε
ο χρήστης να μην έχει τρόπο να γνωρίζει τι κάνει ο διακομιστής.

Το μεγαλύτερο πλεονέκτημα της χρήσης της PHP είναι ότι είναι εξαιρετικά απλή για
αρχάριους, ενώ παρέχει επίσης μια ποικιλία προηγμένων δυνατοτήτων για
επαγγελματίες προγραμματιστές. Μην φοβάστε όταν βλέπετε τη μεγάλη λίστα

δυνατοτήτων της PHP. Μπορείτε να ξεκινήσετε γρήγορα και μπορείτε να γράψετε μόνοι σας μερικά απλά σενάρια μέσα σε λίγες μόνο ώρες.

Αν και η PHP αναπτύχθηκε για δέσμες ενεργειών από την πλευρά του διακομιστή, στην πραγματικότητα οι δυνατότητές της είναι πολύ περισσότερες από αυτό.

PHP και MySQL

Απλό διαδικτυακό σύστημα έρευνας που χρησιμοποιεί **PHP/MySQLi** και πηγαίο κώδικα. Η δέσμη ενεργειών που βασίζεται σε **jQuery** χρησιμοποιεί πίνακες βάσης δεδομένων **MySQL** και σενάρια **PHP** για τη διεξαγωγή διαδικτυακών ερευνών. Το **Poll Survey System** είναι μια διαδικτυακή εφαρμογή που αναπτύχθηκε με **PHP** και **MySQL** και εκτελείται σε διακομιστή **WAMP, XAMP ή Apache2**. Οι κύριες ενότητες του έργου είναι *Δημοσκοπήσεις, Χρήστες, Έρευνα, Αποτελέσματα, Εγγραφή*.

Δημιουργία ερευνών χρηστών PHP: Πίνακες βάσεων δεδομένων και αρχεία κλάσης
Το **php-survey-builder** είναι μια διαδικτυακή εφαρμογή PHP που σας επιτρέπει να δημιουργείτε έρευνες και να συλλέγετε απαντήσεις σε έρευνες. Τα αποτελέσματα μπορούν να προβληθούν σε ένα γράφημα και να εξαχθούν σε **CSV**. Τα δεδομένα έρευνας αποθηκεύονται στη βάση δεδομένων **sqlite3** από προεπιλογή και μπορούν επίσης να αποθηκευτούν στη βάση δεδομένων **mysql ή postgresql**. Επεξεργαστείτε το **config/Database.ini** για να χρησιμοποιήσετε **mysql ή postgresql**. Το **PHP Survey** είναι ένα σενάριο ιστού **php** που είναι εύκολο στην εγκατάσταση (δεν χρησιμοποιεί καμία βάση δεδομένων **MySQL** και δεν απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων) που επιτρέπει τη δημιουργία διαδικτυακών ερευνών όπως η ικανοποίηση πελατών ή η αξιολόγηση προϊόντων ή άλλες έρευνες με ερωτήσεις και απαντήσεις χρειάζομαι.

Δημιουργία Έρευνας Χρηστών PHP: Πίνακες Βάσεων Δεδομένων και Αρχεία Τάξης
Το σύστημα ψηφοφορίας είναι μια διαδικτυακή εφαρμογή που αναπτύχθηκε σε PHP και MySQL και εκτελείται σε διακομιστή **WAMP, XAMP ή Apache2**. Οι κύριες ενότητες του έργου είναι *δημοσκοπήσεις, χρήστες, έρευνες, αποτελέσματα, εγγραφή*. Το **php-survey-builder** είναι μια διαδικτυακή εφαρμογή **PHP** που σας επιτρέπει να δημιουργείτε έρευνες και να συλλέγετε απαντήσεις σε έρευνες. Τα

αποτελέσματα μπορούν να προβληθούν σε ένα γράφημα και να εξαχθούν σε CSV. Τα δεδομένα έρευνας αποθηκεύονται στη βάση δεδομένων *sqlite3* από προεπιλογή και μπορούν επίσης να αποθηκευτούν στη βάση δεδομένων *mysql* ή *postgresql*. Επεξεργαστείτε το *config/Database.ini* για να χρησιμοποιήσετε *mysql* ή *postgresql*. [rhp tutorial](#): δημιουργήστε τη δική σας εφαρμογή έρευνας/ψηφοφορίας. Το *PHP Survey* είναι ένα σενάριο *web rhp* που είναι εύκολο στην εγκατάσταση (δεν χρησιμοποιεί καμία βάση δεδομένων *MySQL* και δεν απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων) επιτρέπει τη δημιουργία διαδικτυακών ερευνών όπως η ικανοποίηση πελατών ή οι κριτικές προϊόντων ή άλλα. Μια έρευνα με ερωτήσεις και απαντήσεις που μπορεί να χρειαστείτε. Δημιουργήστε μια φόρμα δημοσκόπησης *MySQL*. Αρχικά, πρέπει να δημιουργήσουμε έναν πίνακα βάσης δεδομένων για να περιέχει πληροφορίες για κάθε συγκεκριμένη ψήφο. Ο πίνακας θα περιέχει: Αριθμός αναγνωριστικού πρωτεύοντος κλειδιού.

rhp tutorial: δημιουργία της δικής σας εφαρμογής έρευνας/ψηφοφορίας. Το *rhp-survey-builder* είναι μια διαδικτυακή εφαρμογή *PHP* που σας επιτρέπει να δημιουργείτε έρευνες και να συλλέγετε απαντήσεις σε έρευνες. Τα αποτελέσματα μπορούν να προβληθούν σε ένα γράφημα και να εξαχθούν σε CSV. Τα δεδομένα έρευνας αποθηκεύονται στη βάση δεδομένων *sqlite3* από προεπιλογή και μπορούν επίσης να αποθηκευτούν στη βάση δεδομένων *mysql* ή *postgresql*. Επεξεργαστείτε το *config/Database.ini* για να χρησιμοποιήσετε *mysql* ή *postgresql*.

Το **PHP Survey**, είναι ένα σενάριο ιστού *rhp*, που είναι εύκολο στην εγκατάσταση (δεν χρησιμοποιεί καμία βάση δεδομένων *MySQL* και δεν απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων), που επιτρέπει τη δημιουργία διαδικτυακών ερευνών όπως η ικανοποίηση πελατών, ή η αξιολόγηση προϊόντων ή άλλες έρευνες με ερωτήσεις και απαντήσεις Σύστημα ψηφοφορίας και έρευνας σε *PHP*, *MySQL*, *AJAX*, με αποτελέσματα δημοσκόπησης που δημιουργούν πίνακα εκλογών *MySQL*. Αρχικά, πρέπει να δημιουργήσουμε έναν πίνακα βάσης δεδομένων για να περιέχει πληροφορίες για κάθε συγκεκριμένη ψήφο. Ο πίνακας θα περιέχει, Αριθμός αναγνωριστικού πρωτεύοντος κλειδιού

Τι είναι η έρευνα *PHP*;

Το PHP Survey είναι ένα εύκολο στην εγκατάσταση σενάριο web php (δεν χρησιμοποιεί καμία βάση δεδομένων MySQL και δεν απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων).

Σύστημα ψηφοφορίας και έρευνας σε PHP, Mysql, AJAX με αποτελέσματα δημοσκοπήσης Το PHP Survey είναι ένα σενάριο ιστού php που είναι εύκολο στην εγκατάσταση (δεν χρησιμοποιεί καμία βάση δεδομένων MySQL, δεν απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων) επιτρέπει τη δημιουργία διαδικτυακών ερευνών όπως η ικανοποίηση πελατών ή προϊόντα Μια αξιολόγηση ή άλλη έρευνα με ερωτήσεις και απαντήσεις που μπορεί να χρειαστείτε. Δημιουργήστε έναν πίνακα δημοσκοπήσεων MySQL. Αρχικά, πρέπει να δημιουργήσουμε έναν πίνακα βάσης δεδομένων για να περιέχει πληροφορίες για κάθε συγκεκριμένη ψήφο. Ο πίνακας θα περιέχει: Αριθμός αναγνωριστικού πρωτεύοντος κλειδιού.

phpdave11/php-survey-builder: Μια εφαρμογή Ιστού PHP

Τι είναι η έρευνα PHP;

Το PHP Survey είναι ένα εύκολο στην εγκατάσταση script ιστού php (δεν χρησιμοποιεί καμία βάση δεδομένων MySQL, ούτε απαιτεί καμία ρύθμιση παραμέτρων βάσης δεδομένων) για τη δημιουργία του MySQL Polls Table. Αρχικά, πρέπει να δημιουργήσουμε έναν πίνακα βάσης δεδομένων για να περιέχει πληροφορίες για κάθε συγκεκριμένη ψήφο. Ο πίνακας θα περιέχει: αριθμό αναγνωριστικού πρωτεύοντος κλειδιού, συνολικές ψήφους στη δημοσκόπηση, ερώτηση δημοσκόπησης, κατάσταση δημοσκόπησης (ενεργή/ανενεργή), εάν η δημοσκόπηση διαγράφηκε, ημερομηνία/ώρα δημιουργίας της δημοσκόπησης

phpdave11/php-survey-builder: Μια εφαρμογή web PHP που δημιουργεί φόρμες δημοσκόπησης MySQL. Αρχικά, πρέπει να δημιουργήσουμε έναν πίνακα βάσης δεδομένων για να περιέχει πληροφορίες για κάθε συγκεκριμένη ψήφο. Ο πίνακας θα περιέχει: Αριθμός αναγνωριστικού πρωτεύοντος κλειδιού Χρήση δέσμης ενεργειών ηλεκτρονικής έρευνας JQuery PHP MySQL για εμφάνιση του προβλήματος,.

Ο Βασικός Οδηγός για τον Ορισμό του Site

Σελίδα ή σύνολο σελίδων στο Διαδίκτυο με διάφορες πληροφορίες , προσβάσιμες μέσω υπολογιστή ή άλλων μέσων. Ο σχεδιασμού στο Διαδίκτυο είναι πολυσύνθετος, αλλά βασικός κατά την εφαρμογή του. Πριν τη δημιουργία και το «ανέβασμα» και την δημοσίευση μιας ιστοσελίδας, είναι σημαντικό να σχεδιαστεί ακριβώς ότι είναι αναγκαίο για την ιστοσελίδα. Ο σχεδιασμός στο διαδίκτυο από μια άποψη είναι παρόμοιος με την κλασσική τυπογραφία. Κάθε ιστοσελίδα είναι ένα σύνολο αναπαραγόμενων πληροφοριών, όπως ένα βιβλίο. Κάθε σελίδα της ιστοσελίδας είναι αντίστοιχη της σελίδας ενός βιβλίου. Όμως η σχεδίαση ιστοσελίδων χρησιμοποιείται ένα πλαίσιο εργασίας (framework) βασισμένο σε ένα ψηφιακό κώδικα και σε μια τεχνολογία απεικονίσεων για να δημιουργήσει και να διατηρήσει ένα περιβάλλον που θα μοιράζει την πληροφορία σε πολλαπλές μορφές αρχείων (formats) όπως pdf, html, doc, κλπ.

Αν αποφασίσουμε να χρησιμοποιήσουμε το πρόγραμμα Dreamweaver MX, υπάρχουν τέσσερις διαφορετικοί τρόποι για να ορίσουμε το site μας με τον οδηγό ορισμού :

- Δημιουργία και δοκιμή τοπικά.
- Δημιουργία τοπικά και upload για δοκιμή στον απομακρυσμένο server.
- Δημιουργία απευθείας μέσω δικτύου (network).
- Δημιουργία απευθείας μέσω FTP.

Αρχικά για να ορίσουμε ένα Site με τον βασικό οδηγό ορισμού είναι πρώτα η ονομασία του site και προς τον σκοπό αυτό πηγαίνουμε στο μενού Site και επιλέγουμε New Site... Θα ξεκινήσει ο οδηγός ορισμού του site (Site Definition Wizard). Στο πρώτο πλαίσιο διαλόγου θα πρέπει να δώσουμε ένα όνομα στο site μας, όπως για παράδειγμα The Dreamweaver Hotel, το οποίο όνομα είναι καθαρά τυπικό για την αναγνώριση του Site από εμάς και δεν πρόκειται να χρησιμοποιηθεί πουθενά.

Στην συνέχεια αφού έχουμε δώσει ένα όνομα για το Site θα πρέπει να ορίσουμε την τεχνολογία server που θα χρησιμοποιήσει το site στο δεύτερο βήμα του οδηγού. Θα

πρέπει να αποφασίσουμε αν το site μας θα χρησιμοποιεί δυναμικές τεχνολογίες, όπως είναι οι ASP, JSP, CFML ή PHP, ή θα επιστρέφει στατικές HTML σελίδες.

Διαλέγουμε το σωστό πλήκτρο επιλογής (Yes, I want to use a server technology) και από την πτυσσόμενη λίστα που θα εμφανισθεί αμέσως από κάτω (Which server technology?), επιλέγουμε PHP MySQL.

Τώρα μπορούμε να προχωρήσουμε για να δώσουμε στο **Dreamweaver** πληροφορίες σχετικά με τα αρχεία που θα δουλέψουμε. Θα καθορίσουμε πού θα αποθηκευτούν τα πηγαία αρχεία που θα δημιουργήσουμε καθώς επίσης αν ο **Web server** θα είναι ένα τοπικό μηχάνημα, ή ένα απομακρυσμένο μηχάνημα σ' ένα τοπικό δίκτυο, ή ένα απομακρυσμένο μηχάνημα που χρειάζεται μια ενεργή σύνδεση στο **Internet**.

Ακολουθεί το βήματα του οδηγού θα διαφέρουν ανάλογα με την επιλογή που κάνουμε εδώ.

- Η πρώτη επιλογή είναι η *Edit and test locally* (my testing server is on this computer), την οποία θα πρέπει να επιλέξουμε αν όλα δουλεύουν από το ίδιο μηχάνημα.
- Η δεύτερη επιλογή *Edit locally, then upload to remote testing server*. Αυτή η επιλογή είναι η προτιμότερη όταν σχεδιάζουμε τις ιστοσελίδες μας και τις δημοσιεύουμε σ' έναν *server* που βρίσκεται σ' ένα διαφορετικό μηχάνημα, είτε σ' ένα τοπικό δίκτυο ή στο Internet.

Μ' αυτήν την επιλογή, η μόνη πληροφορία που θα χρειασθεί ο οδηγός είναι ο φάκελος όπου θα αποθηκεύσουμε τοπικά όλα τα αρχεία τα σχετικά με το Web site. Στο επόμενο βήμα του οδηγού θα πρέπει να απαντήσουμε στην ερώτηση "How do you connect to your testing server?" επιλέγοντας μία από τέσσερις επιλογές (I'll set this up later, FTP, Local/Network, RDS). Επιλέγοντας την FTP, που είναι το πιθανότερο, εμφανίζονται τέσσερα πλαίσια κειμένου όπου θα πρέπει να καταχωρήσουμε το hostname ή την FTP address του server, το όνομα του φακέλου στον server όπου θα αποθηκευθεί το Web site και τέλος το login (user name) και το password για να μπορέσει να γίνει η FTP σύνδεση. Έχουμε δυνατότητα να

αποθηκεύσουμε τους κωδικούς για την FTP σύνδεση αν επιλέξουμε το πλαίσιο ελέγχου Save , κάνουμε κλικ στο πλήκτρο Test Connection ώστε να διαπιστώσουμε αν η σύνδεση λειτουργεί κανονικά. Παρατηρούμε ότι ο οδηγός ορισμού site είναι πολύ απλός και φιλικός προς τον χρήστη, υπάρχουν πολλές λεπτομέρειες που δεν καλύπτει και έτσι είναι απαραίτητο να δούμε και τον προηγμένο ορισμό site.

Επιλέγοντας την Advanced του πλαισίου διαλόγου Site Definition βλέπουμε επτά κατηγορίες στην αριστερή πλευρά να εμφανίζονται. Η κατηγορία Local Info περιέχει στοιχεία που μας είναι ήδη γνωστά, όπως είναι το όνομα του site (Site Name) και ο φάκελος αποθήκευσης των αρχείων (Local Root Folder). Η κατηγορία Remote Info έχει να κάνει με τις ρυθμίσεις του απομακρυσμένου server, όπως είναι ο τρόπος πρόσβασης (π.χ. FTP), η διεύθυνση του FTP Host, το Host Directory καθώς και το Login και το Password για να μπορέσουμε να συνδεθούμε.

Βάση Δεδομένων – MySQL

Χαρακτηριστικά και πλεονεκτήματα της MySQL

Τι είναι η MySQL:

Με την πάροδο του χρόνου, οι βάσεις δεδομένων ανοιχτού κώδικα έγιναν σταδιακά δημοφιλείς σε εφαρμογές χαμηλού και μεσαίου επιπέδου, καταλαμβάνοντας μεγάλο μερίδιο αγοράς.

Οι βάσεις δεδομένων ανοιχτού κώδικα έχουν τα χαρακτηριστικά της ελεύθερης χρήσης, της απλής διαμόρφωσης, της καλής σταθερότητας και της εξαιρετικής απόδοσης, και η βάση δεδομένων MySQL είναι ένας εξαιρετικός εκπρόσωπος βάσεων δεδομένων ανοιχτού κώδικα. *Η MySQL είναι μια από τις πιο δημοφιλείς βάσεις δεδομένων και είναι ένα δωρεάν και ανοιχτού κώδικα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων, αλλά αυτό δεν σημαίνει ότι η βάση δεδομένων είναι εντελώς δωρεάν.* Η MySQL αναπτύχθηκε από τη σουηδική εταιρεία MySQL AB και επί του παρόντος ανήκει στην Oracle Corporation. Η MySQL είναι κατάλληλη για μικρού και μεσαίου μεγέθους λογισμικό και προτιμάται από μεμονωμένους χρήστες και μικρές και μεσαίες επιχειρήσεις.

Για διαφορετικούς χρήστες, η MySQL χωρίζεται σε δύο εκδόσεις

MySQL Community Server (Community Edition) : Αυτή η έκδοση είναι δωρεάν για λήψη και εντελώς δωρεάν, αλλά δεν παρέχεται επίσημη τεχνική υποστήριξη.

MySQL Enterprise Server (Enterprise Edition) : Αυτή η έκδοση χρεώνεται και δεν μπορεί να ληφθεί, αλλά αυτή η έκδοση διαθέτει πλήρη τεχνική υποστήριξη (επίσημη τηλεφωνική τεχνική υποστήριξη).

Χαρακτηριστικά και πλεονεκτήματα της MySQL

Το σύστημα διαχείρισης βάσεων δεδομένων MySQL έχει πολλά πλεονεκτήματα, μερικά από τα οποία συνοψίζονται παρακάτω.

1) Η MySQL είναι μια βάση δεδομένων ανοιχτού κώδικα

Η MySQL είναι μια βάση δεδομένων ανοιχτού κώδικα της οποίας ο πηγαίος κώδικας είναι διαθέσιμος σε οποιονδήποτε. Αυτό επιτρέπει σε οποιονδήποτε να διορθώσει σφάλματα στη MySQL και σε οποιονδήποτε να χρησιμοποιήσει τη βάση δεδομένων για οποιονδήποτε σκοπό. Η MySQL είναι μια δωρεάν διαθέσιμη βάση δεδομένων.

2) Cross-platform της MySQL

Η MySQL μπορεί να εκτελεστεί όχι μόνο στην οικογένεια λειτουργικών συστημάτων Windows, αλλά και σε λειτουργικά συστήματα όπως το UNIX, το Linux και το Mac OS. Επειδή πολλοί ιστότοποι επιλέγουν το UNIX και το Linux ως διακομιστές ιστοτόπων, η διαπλατφορμική φύση της MySQL διασφαλίζει τα πλεονεκτήματά της στις εφαρμογές Ιστού. Αν και η βάση δεδομένων SQL Server της Microsoft είναι μια εξαιρετική εμπορική βάση δεδομένων, μπορεί να εκτελεστεί μόνο στην οικογένεια λειτουργικών συστημάτων των Windows. Επομένως, η διαπλατφορμική φύση των βάσεων δεδομένων MySQL είναι ένα μεγάλο πλεονέκτημα.

3) Πλεονέκτημα τιμής

Η βάση δεδομένων MySQL είναι ένα δωρεάν λογισμικό, ο καθένας μπορεί να κατεβάσει το λογισμικό από τον επίσημο ιστότοπο της MySQL, αυτές οι κοινοτικές εκδόσεις της MySQL είναι δωρεάν για δοκιμή, ακόμη και οι επιπλέον δυνατότητες που πρέπει να πληρωθούν είναι πολύ φθηνές. Σε σύγκριση με τα Oracle, DB2 και

SQL Server, αυτά τα ακριβά εμπορικά λογισμικά, η MySQL έχει απόλυτο πλεονέκτημα τιμής.

4) Ισχυρό και εύκολο στη χρήση

Η MySQL είναι ένας πραγματικός διακομιστής βάσης δεδομένων SQL πολλαπλών χρηστών και πολλαπλών νημάτων. Μπορεί να επεξεργαστεί μεγάλες ποσότητες δεδομένων γρήγορα, αποτελεσματικά και με ασφάλεια. Σε σύγκριση με βάσεις δεδομένων όπως η Oracle, η χρήση της MySQL είναι πολύ απλή. Οι κύριοι στόχοι της MySQL είναι να είναι γρήγορη, ισχυρή και εύκολη στη χρήση.

*Σε σύγκριση με τις κοινώς χρησιμοποιούμενες βάσεις δεδομένων όπως η **Oracle** και ο **SQL Server**, το κύριο χαρακτηριστικό της **MySQL** είναι ότι είναι δωρεάν και μπορεί να χρησιμοποιηθεί σε οποιαδήποτε πλατφόρμα, καταλαμβάνοντας σχετικά λίγο χώρο. Ωστόσο, η MySQL έχει επίσης κάποιες ελλείψεις: Για παράδειγμα, για μεγάλα έργα, η χωρητικότητα και η ασφάλεια της MySQL είναι ελαφρώς κατώτερες από τη βάση δεδομένων της Oracle.*

Δημιουργία της Σύνδεσης με τη Βάση Δεδομένων

Μια σύνδεση βάσης δεδομένων σας επιτρέπει να εργάζεστε με πίνακες βάσεων δεδομένων απευθείας στο Insights . Μπορούν να δημιουργηθούν συνδέσεις βάσεων δεδομένων για υποστηριζόμενες βάσεις και πρόσθετες σχεσιακές βάσεις δεδομένων που χρησιμοποιούν πρόγραμμα οδήγησης *Java Database Connectivity (JDBC)* .

Πριν δημιουργήσετε μια σύνδεση βάσης δεδομένων, πρέπει να πληρούνται οι ακόλουθες προϋποθέσεις:

Η κατάλληλη σύνδεση πρέπει να προστεθεί πριν από τη δημιουργία μιας σύνδεσης βάσης δεδομένων.

Απαιτούμενα αρχεία προμηθευτή και Διαχείριση τύπων εφαρμογών σύνδεσης .

Πρέπει να έχετε τα δικαιώματα για την κατάλληλη βάση δεδομένων στην οποία θέλετε να συνδεθείτε. Εάν δεν έχετε δικαιώματα βάσης δεδομένων, επικοινωνήστε με τον διαχειριστή της βάσης δεδομένων σας.

Πρέπει να μπορείτε να επαληθεύσετε τη σύνδεση. Για τις περισσότερες βάσεις δεδομένων, ένα όνομα χρήστη και ένας κωδικός πρόσβασης χρησιμοποιούνται για έλεγχο ταυτότητας.

Ο SQL Server και το BigQuery χρησιμοποιούν τις ακόλουθες εναλλακτικές μεθόδους ελέγχου ταυτότητας:

Ο SQL Server μπορεί να χρησιμοποιήσει έλεγχο ταυτότητας SQL Server (όνομα χρήστη και κωδικό πρόσβασης) ή έλεγχο ταυτότητας OS, για περισσότερες πληροφορίες, ανατρέξτε στο θέμα *Ενεργοποίηση ελέγχου ταυτότητας λειτουργικού συστήματος* .

Το BigQuery χρησιμοποιεί έλεγχο ταυτότητας λογαριασμού υπηρεσίας. Για περισσότερες πληροφορίες, ανατρέξτε στο θέμα *Δημιουργία λογαριασμού υπηρεσίας και ιδιωτικού κλειδιού* .

Δημιουργήστε συνδέσεις βάσης δεδομένων

Μπορείτε να προβάλετε στοιχεία σύνδεσης σχεσιακής βάσης δεδομένων από την καρτέλα Συνδέσεις στην αρχική σελίδα ή, εάν χρησιμοποιείτε το Insights στο ArcGIS Enterprise στην πύλη ArcGIS Enterprise . Οι συνδέσεις μπορούν να δημιουργηθούν από την αρχική σελίδα ή πριν προσθέσετε δεδομένα στο βιβλίο εργασίας σας .

Δημιουργία μιας Κράτησης (Booking)

Θα αρχίσουμε τώρα να δημιουργούμε το σύστημα κράτησης θέσεων στο ξενοδοχείο, ώστε να μπορούν οι επισκέπτες να κάνουν κρατήσεις δωματίων καθώς και να τροποποιούν και να ακυρώνουν τις κρατήσεις τους.

Ουσιαστικά θα δούμε το πώς μπορούμε να προσθέτουμε (add), ενημερώνουμε (update) και διαγράφουμε (delete) δεδομένα στους πίνακες (tables) μιας βάσης δεδομένων.

Αν κάποιος χρήστης έχει ήδη κάνει μια κράτηση, τότε θα υπάρχουν τα στοιχεία του στον πίνακα clients της βάσης δεδομένων.

Για να μπορέσει να γίνει η αναγνώριση του χρήστη, θα του ζητηθεί να καταχωρήσει την διεύθυνση του e-mail του καθώς αυτή είναι μοναδική για τον κάθε χρήστη.

Αν ο χρήστης δεν έχει κάνει κάποια κράτηση παλαιότερα, τότε θα πρέπει να ζητήσουμε τα στοιχεία του και να τα καταχωρήσουμε στον πίνακα *clients*.

Μετά θα χρειασθούμε μια άλλη σελίδα για να καταχωρήσουμε τα στοιχεία της κράτησης του χρήστη στον πίνακα *bookings*.

Θα δημιουργήσουμε τις εξής ιστοσελίδες :

• **add_user_record.php**, όπου ο χρήστης να καταχωρεί τα στοιχεία του σε μια φόρμα και θα αποθηκεύονται στον πίνακα *clients*.

Μετά την καταχώρηση της εγγραφής, ο χρήστης θα κατευθύνεται στην ιστοσελίδα *booking_details.php*, όπου θα καταχωρήσει τα στοιχεία για την κράτησή του.

Η παράμετρος που θα συνδέει τις δύο ιστοσελίδες θα είναι η *e-mail* διεύθυνση του χρήστη.

• **book_a_room.php**, που είναι στην ουσία η πρώτη σελίδα που βλέπει ο χρήστης όταν κάνει την επιλογή κράτησης δωματίου από το κύριο μενού.

Αν είναι καινούργιος χρήστης, θα υπάρχει ένας σύνδεσμος (link) με την σελίδα *add_user_record.php* για να μπορεί να καταχωρήσει τα στοιχεία του. Αν είναι παλιός χρήστης, τότε θα μπορούμε να εμφανίσουμε τα στοιχεία του από τη βάση δεδομένων.

Προς τον σκοπό αυτό θα του ζητήσουμε να καταχωρήσει το e-mail του ώστε να μπορούμε να κάνουμε αναζήτηση στον πίνακα *clients*.

• **booking_details.php**, που είναι η σελίδα όπου θα καταχωρήσει ο χρήστης τα στοιχεία για την κράτηση και τα οποία θα προστεθούν στον πίνακα *bookings*.

Τι είναι οι υπηρεσίες πληροφορικής;

Τελειώνοντας αυτό το βιβλίο στον ΕΙΔΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ ΙΣΤΟΣΕΛΙΔΩΝ ΚΑΙ ΕΦΑΡΜΟΓΩΝ θα αναφερθούμε λίγο και στις υπηρεσίες πληροφορικής.

Οι υπηρεσίες πληροφορικής αναφέρονται σε ένα σύνολο δραστηριοτήτων και λύσεων που περιλαμβάνουν βάσεις δεδομένων, υλικό, λογισμικό και δίκτυα.

Όλες αυτές οι ενέργειες εκτελούνται για να βοηθήσουν στην ανάλυση και διαχείριση των πληροφοριών.

Ο τομέας της πληροφορικής παρέχει υπηρεσίες σε όλους τους άλλους τομείς που χρησιμοποιούν τεχνολογία, δηλαδή δρα άμεσα στον τομέα της τεχνολογικής ανάπτυξης για να ανταποκριθεί στις απαιτήσεις ενός συγκεκριμένου χρήστη ή συστήματος.

Η υπηρεσία πληροφορικής χωρίζεται σε τρεις κύριους τομείς δραστηριότητας εντός της αγοράς:

- Υποδομή
- Λογισμικό
- Βάση δεδομένων και δίκτυα.

Στην υποδομή, που ενεργούν είναι οι αναλυτές τεχνικής υποστήριξης και οι διαχειριστές δικτύων. Στον τομέα του λογισμικού υπάρχουν προγραμματιστές και τεχνολογικοί προγραμματιστές. Στη βάση δεδομένων και τα δίκτυα, αυτοί που ενεργούν άμεσα είναι οι διαχειριστές της βάσης δεδομένων, που αναγνωρίζονται ως ADB και ειδικοί. Υπάρχουν πολλές άλλες υποδιαιρέσεις εντός της Πληροφορικής, αλλά όλες είναι αλληλένδετες. Ο στόχος είναι πάντα να συνεργαζόμαστε για να διασφαλίζουμε καλύτερα αποτελέσματα σε δυναμικές ενέργειες. Για να κατανοήσετε καλύτερα τα κύρια χαρακτηριστικά και τους δείκτες ποιότητας του, συνεχίστε την ανάγνωση.

Τα 3 χαρακτηριστικά των υπηρεσιών πληροφορικής

Όπως όλοι οι συγκεκριμένοι τομείς, ο τομέας της πληροφορικής έχει τα δικά του χαρακτηριστικά, τα οποία με τη σειρά τους επηρεάζουν άμεσα την παροχή υπηρεσιών: **άυλο, ετερογένεια και αδιαχώριστο.**

Θα εξηγήσουμε το καθένα από αυτά παρακάτω:

Ανεπίδεκτο

Το άυλο αφορά ό,τι δεν μπορεί να αγγίξει, ό,τι δεν είναι χειροπιαστό, όπως συμβαίνει με τις υπηρεσίες πληροφορικής. Λόγω του άυλου χαρακτήρα, υπάρχει μεγάλη υποκειμενικότητα στην αξιολόγηση της υπηρεσίας από τον πελάτη, εκτός από τη δυσκολία επικοινωνίας στο ευρύ κοινό των ιδιοτήτων της προσφερόμενης υπηρεσίας. Καθώς δεν είναι κάτι απτό (*ένα προϊόν, για παράδειγμα*), ο κίνδυνος που αντιλαμβάνεται ο αγοραστής φαίνεται πολύ μεγάλος. Για να μετριάσουν αυτές οι επιπτώσεις, οι διευθυντές πρέπει να ενδιαφέρονται για τα φυσικά χαρακτηριστικά της εταιρείας ή του κλάδου και τα συγκεντρωτικά προϊόντα, όπως εκθέσεις, αποτελέσματα δειγμάτων, τεχνική ικανότητα της ομάδας κ.λπ. Είναι απαραίτητο να δημιουργηθούν «απτές αξίες». Λόγω όλων αυτών των παραγόντων, το άυλο θεωρείται το πιο εντυπωσιακό χαρακτηριστικό και επίσης το πιο δύσκολο στη διαχείριση πληροφορικής.

Ανομοιογένεια

Η ετερογένεια μας λέει ότι οι Υπηρεσίες IT δεν θα προσφέρονται *ποτέ με τον ίδιο τρόπο* σε δύο διαφορετικούς πελάτες ή σε δύο διαφορετικούς χρόνους. Αυτό το «πρόβλημα» μπορεί να προκληθεί από διάφορους παράγοντες, τα προσόντα εκείνων που παρέχουν την υπηρεσία, την ανάγκη εξατομίκευσης που περιλαμβάνει συγκεκριμένες υπηρεσίες, ακόμη και τον τρόπο με τον οποίο ο πελάτης επικοινωνήσει την ανάγκη του. Στις περιπτώσεις εξατομίκευσης και προσαρμογής, η ετερογένεια μπορεί να γίνει ανταγωνιστικό πλεονέκτημα. *Ωστόσο, είναι σημαντικό να θυμάστε ότι μπορεί να επηρεάσει το κόστος και τον χρόνο ολοκλήρωσης της υπηρεσίας.* Η εντατική εκπαίδευση των εργαζομένων και η προσπάθεια καθιέρωσης μιας τυποποίησης των υπηρεσιών μπορεί να είναι βιώσιμες λύσεις για την ελαχιστοποίηση αυτών των επιπτώσεων.

Όπως συμβαίνει με όλες τις υπηρεσίες πληροφορικής, είναι επίσης σημαντικό να παρακολουθείται η απόδοση των δραστηριοτήτων και η ικανοποίηση των πελατών προκειμένου να προτείνονται βελτιώσεις και να εντοπιστούν οι αδυναμίες.

Αχώριστο

Ασχολείται με την αδυναμία διαχωρισμού του παρόχου από την υπηρεσία που προσφέρει. Στο πλαίσιο αυτό, είναι απαραίτητο να επενδύσουμε τόσο στην τεχνική ικανότητα του υπαλλήλου πληροφορικής όσο και στην ανάπτυξη της υπηρεσίας τους, διευκολύνοντας τη σχέση με τον πελάτη σε όλα τα επίπεδα παροχής υπηρεσιών.

Δείκτες ποιότητας υπηρεσιών πληροφορικής

Είναι απαραίτητο να προσέχουμε τους δείκτες ποιότητας στην ανάθεση υπηρεσιών και την παρακολούθηση, στην περίπτωση της πληροφορικής, δεν διαφέρει.

Σημειώστε λοιπόν:

Χρησιμότητα

Το να έχεις έναν επαγγελματία πληροφορικής επικεντρωμένο ειδικά στη δυναμική της εταιρείας είναι μια εξαιρετική ευκαιρία για βελτίωση της απόδοσης και μείωση του κόστους. Όταν αυτός ο επαγγελματίας είναι εσωτερικός, μπορεί να χρειαστεί χρόνος για να αποκτήσει πρόσβαση στις αδυναμίες της εταιρείας, καθώς είναι επικεντρωμένος σε όλες τις απαιτήσεις, χωρίς ιδιαιτερότητες.

Εγγύηση

Η αναζήτηση συστάσεων, η πραγματοποίηση δοκιμών πριν από την πρόσληψη της υπηρεσίας και η γνώση των ρητρών του συμβολαίου θα σας δώσει μεγαλύτερη εγγύηση όταν προσλαμβάνετε μια εταιρεία που έχει ανατεθεί σε εξωτερικούς συνεργάτες για τη φροντίδα αυτού του τομέα.

Λήξη

Η ωριμότητα είναι η ικανότητα του τομέα της πληροφορικής να προσφέρει αξία στον οργανισμό μέσω της τεχνολογίας. Αυτός ο δείκτης μπορεί να αναλυθεί από το χώρο που έχει αυτός ο τομέας στον οργανισμό.

Είναι επίσης αυτός ο δείκτης που καταδεικνύει την ανάγκη πρόσληψης άλλων υπηρεσιών και την ποιότητα αυτού που έχει αναπτυχθεί, είτε ανταποκρίνεται στη ζήτηση της εταιρείας είτε όχι.

τεχνογνωσία προμηθευτή

Μια ομάδα πληροφορικής πρέπει να έχει καλό ακαδημαϊκό υπόβαθρο και μεγάλη εμπειρία στην αγορά, δηλαδή επαγγελματίες με υψηλά προσόντα στον κλάδο.

Ψάξτε για αναφορές, αναλύστε ποιος έχει συνεργαστεί με αυτήν την εταιρεία, χρόνο στην αγορά, τεχνική ικανότητα και μην καθοδηγηθείτε ποτέ μόνο από την τιμή. Σε αυτές τις περιπτώσεις, η φροντίδα είναι μικρή και το φθηνό μπορεί να είναι ακριβό.

Όπως είδαμε, οι υπηρεσίες πληροφορικής είναι εξειδικευμένες και διαφοροποιημένες, προσφέρονται από εταιρείες με αποκλειστική αφοσίωση στον τομέα της τεχνολογίας και που είναι πάντα ενημερωμένες με τις τελευταίες τεχνικές της αγοράς.

Ο τομέας πληροφορικής προσφέρει πολλές δυνατότητες. Ο διευθυντής που επιλέγει να αναθέσει αυτές τις υπηρεσίες σε εξωτερικούς συνεργάτες πρέπει να αξιολογήσει ποιες είναι οι ανάγκες της εταιρείας του και, επομένως, ποιες λειτουργίες θα δώσει προτεραιότητα η εταιρεία που έχει αναθέσει σε εξωτερικούς συνεργάτες κατά την εφαρμογή των δυνατοτήτων της στην πράξη.

Ως εκ τούτου, οι κύριες εργασίες που εκτελεί η περιοχή είναι:

Η υποδομή πληροφορικής μπορεί να χωριστεί σε τρεις ομάδες:

1. Υλικό

Μέσα στην υποδομή πληροφορικής, το υλικό περιλαμβάνει κάθε μέρος, από διακομιστές, κέντρα δεδομένων, προσωπικούς υπολογιστές, δρομολογητές, διακόπτες και άλλες συσκευές. Οι εγκαταστάσεις που είναι υπεύθυνες για τη στέγαση, την ψύξη και την τροφοδοσία αυτού του εξοπλισμού, που υποστηρίζουν το κέντρο δεδομένων, ανήκουν επίσης σε αυτήν την κατηγορία.

2. Λογισμικό

Στο λογισμικό είναι τα λειτουργικά συστήματα, δηλαδή άυλα. Εδώ συγκεντρώνονται οι εφαρμογές που χρησιμοποιεί η εταιρεία, όπως web servers, συστήματα διαχείρισης περιεχομένου και διαδικτυακές πλατφόρμες.

Το λειτουργικό σύστημα είναι υπεύθυνο για τη διαχείριση του υλικού και όλων των πόρων του συστήματος.

3. Εγκατάσταση δικτύων

Η εγκατάσταση δικτύου είναι η ομάδα που συνδέει το υλικό και το λογισμικό μαζί. Τα διασυνδεδεμένα στοιχεία του δικτύου επιτρέπουν την εσωτερική και εξωτερική επικοινωνία, διαχείριση και λειτουργίες δικτύου. Η εγκατάσταση δικτύων περιλαμβάνει τη σύνδεση με το Διαδίκτυο, την ενεργοποίηση, τα τείχη προστασίας και την ασφάλεια στον κυβερνοχώρο, επιπλέον του υλικού όπως δρομολογητές, διακόπτες και καλώδια.

Τηλεπικοινωνίες

Η υποδομή του τηλεπικοινωνιακού δικτύου αντιστοιχεί στην προγραμματισμένη και σωστή εγκατάσταση καλωδίωσης και εξοπλισμού. Στόχος, λοιπόν, είναι η σύνδεση, διασύνδεση και υποστήριξη ολόκληρου του δικτύου επικοινωνίας με τον καταλληλότερο τρόπο για τις απαιτήσεις του οργανισμού. Ακούγεται απλό, αλλά αυτός ο τομέας είναι απαραίτητος σε κάθε εταιρεία. Αυτό οφείλεται στο γεγονός ότι, για να είναι αποτελεσματικός, ο τομέας πληροφορικής εξαρτάται από μια σύγχρονη και προηγμένη δομή που υποστηρίζει την κυκλοφορία και τη ζήτηση των χρηστών. Εάν η δομή είναι σε τέλεια ευθυγράμμιση με τις απαιτήσεις του οργανισμού, οι δραστηριότητες ρέουν καλύτερα και το κόστος μπορεί να μειωθεί με τη βελτίωση της απόδοσης και της παραγωγικότητας, εκτός από τη δημιουργία κερδών.

Υπάρχουν διάφοροι τύποι διαθέσιμοι στην αγορά, όπως ασύρματα δίκτυα, απομακρυσμένα δίκτυα (WAN — δίκτυο ευρείας περιοχής), μητροπολιτικά δίκτυα (MAN — δίκτυο μητροπολιτικής περιοχής), μεταξύ άλλων. Για να επιλέξετε το

καταλληλότερο, πρέπει να λάβετε υπόψη παράγοντες όπως η διαθεσιμότητα, η ασφάλεια, η σχέση κόστους-αποτελεσματικότητας και η απόδοση των προϊόντων που σχετίζονται με το δίκτυο ανάλογα με τη διανομή, προσαρμοσμένα στις ανάγκες της εταιρείας.

Τεχνική υποστήριξη

Μία από τις πιο δημοφιλείς υπηρεσίες πληροφορικής είναι η υποστήριξη .

Από τη διαμόρφωση των διαδικασιών εξυπηρέτησης έως την επιλογή εργαλείων, μέσω της εκπαίδευσης και της πρόσληψης της ομάδας, ο συνεργάτης που ανατίθεται σε εξωτερικούς συνεργάτες είναι αυτός που είναι αφοσιωμένος στην προσαρμογή και την κλιμάκωση των υπηρεσιών σύμφωνα με τις ανάγκες της εταιρείας σας.

Βάση δεδομένων

Κάθε εταιρεία καταλήγει να δημιουργεί μια μεγάλη βάση δεδομένων, ακόμα κι αν δεν τη γνωρίζει, αφού ασχολείται καθημερινά με δεδομένα από πελάτες, προμηθευτές και υπαλλήλους.

Η ανησυχία για την ανάπτυξη, αποθήκευση και διαχείριση αυτών των δεδομένων έχει γίνει ακόμη πιο σημαντική. Η επεξεργασία δεδομένων , για παράδειγμα, είναι μια εργασία που απαιτεί πολύ συγκεκριμένες γνώσεις στον τομέα της πληροφορικής και μπορεί να είναι δύσκολο για την εταιρεία σας να προσλάβει έναν ικανό επαγγελματία για αυτήν τη λειτουργία. Η επεξεργασία δεδομένων πραγματοποιείται μέσω μιας σειράς ενεργειών που εφαρμόζονται με τάξη , με στόχο τη διευθέτηση πληροφοριών που θα χρησιμεύσουν ως βάση για την επίτευξη των στόχων του οργανισμού.

Προσλαμβάνοντας μια ομάδα εξειδικευμένη στις υπηρεσίες πληροφορικής, η εταιρεία θα βασιστεί στο ποιος είναι πραγματικά ικανός να εφαρμόσει αυτές τις ενέργειες ακολουθώντας προκαθορισμένα και τυποποιημένα κριτήρια, πάντα με υψηλή απόδοση και ποιότητα.

Η ευκολία αυτών των επαγγελματιών στην αντιμετώπιση των δεδομένων είναι ένα πολύ σημαντικό σημείο, ειδικά όταν πρόκειται για την επιτάχυνση των διαδικασιών και την πρόληψη σφαλμάτων με την ίδια οξυδέρκεια.

Ασφάλεια

Η ρουτίνας προστασίας που υιοθετεί μια εταιρεία θα πρέπει να εξαλείψουν τα τρωτά σημεία και να συμβάλουν στη δημιουργία ενός καλά καθορισμένου συστήματος κυβερνοασφάλειας .

Ένας ιός υπολογιστή είναι ένα κακόβουλο πρόγραμμα που αναπτύχθηκε από προγραμματιστές, το οποίο, όπως ένας βιολογικός ιός, μολύνει το σύστημα, δημιουργεί αντίγραφα του εαυτού του και προσπαθεί να διαδοθεί σε άλλους υπολογιστές, χρησιμοποιώντας διάφορα μέσα, ειδικά email που αποστέλλονται από αμφίβολους αποστολείς.

Ένα σκουλήκι , στον τομέα της τεχνολογίας, είναι παρόμοιο με έναν ιό, αλλά με μια διαφορά, είναι ένα αυτοαναπαράγόμενο πρόγραμμα, επομένως δεν χρειάζεται άλλο πρόγραμμα για να διαδοθεί. Τα ανεπιθύμητα μηνύματα έχουν ελκυστικό χαρακτήρα και, στις περισσότερες περιπτώσεις, είναι άβολα. Γενικά, αυτά τα ανεπιθύμητα email στέλνουν μόνο διαφημίσεις, ωστόσο, σε ορισμένες περιπτώσεις, υπάρχει και η παρουσία ιών, οπότε πρέπει να είστε προσεκτικοί.

Η παρακολούθηση ευπάθειας πρέπει να ευθυγραμμίζεται με την καταπολέμηση επιθέσεων και κακόβουλου λογισμικού, που συμβαίνουν μέσω ιών, spam, worms και email, δηλαδή συχνά συμβαίνουν λόγω αποτυχιών κωδικού πρόσβασης, παραβιάσεων στο σύστημα, πρόσβασης και έλλειψης επικοινωνίας μεταξύ των

Αποκατάσταση καταστροφών

Αναφέρεται σε ένα σύνολο ενεργειών που πρέπει να αναπτυχθούν και να εφαρμοστούν, με προληπτικό τρόπο, κατά τη διάρκεια της εκδήλωσης του γεγονότος και μετά από μια τεχνολογική καταστροφή.

Ο σχεδιασμός πρέπει να είναι συνεπής προκειμένου να διασφαλίζεται η συνέχεια των επιχειρησιακών διαδικασιών — και για να είναι κατάλληλα διαθέσιμοι οι προγραμματισμένοι πόροι σε κάθε έκτακτη ανάγκη— όλα πρέπει να διαμορφώνονται προσεκτικά.

Ο κύριος στόχος του σχεδιασμού αποκατάστασης από καταστροφές είναι η παροχή προστασίας από ένα περιστατικό που μπορεί να καταλήξει να αφήσει άχρηστες θέσεις εργασίας και υπηρεσίες, ειδικά εκείνες που σχετίζονται με τον τομέα της πληροφορικής.

Σύνοψη

Τέλος στην ενότητα αυτή ολοκληρώνετε η εκπαίδευση, εδώ ο εκπαιδευόμενος γίνεται γνώστης στην διασύνδεση εφαρμογών με βάσεις δεδομένων, στις τεχνολογίες διαχείρισης δεδομένων, στην διασύνδεση εφαρμογών με σχεσιακή βάση δεδομένων και τέλος στην διασύνδεση διαδικτυακών εφαρμογών με σχεσιακή βάση δεδομένων (PHP).

Ερώτηση 8 ενότητας

A. Για να συνδεθεί μια java εφαρμογή σε μια βάση δεδομένων τι πρέπει να γίνει;

1. πρέπει να γίνουν 5 βήματα. Καταγραφή της driver class, Δημιουργία σύνδεσης, Δημιουργία statement, Εκτέλεση ερωτημάτων, Κλείσιμο σύνδεσης.
2. δεν χιάζετε να γίνει τίποτα για να συνδεθεί μια java εφαρμογή σε μια βάση δεδομένων.

B. Ποιοι είναι οι τύποι JDBC drivers και πόσοι;

1. 2 τύποι JDBC drivers και είναι JDBC-ODBC bridge driver- Native-API driver.
2. 4 τύποι JDBC drivers και είναι JDBC-ODBC bridge driver- Native-API driver- Network Protocol driver- Thin driver

Γ. Ποια είναι τα πλεονεκτήματα Hadoop;

1. Οικονομία πόρων και Επεκτασιμότητα

2. Αποδοτικότητα και Αξιοπιστία

3. Όλα τα παραπάνω

Δ. Τι είναι το "Cloud;

1. Το "Cloud" είναι ουσιαστικά ένα δίκτυο. Με μια στενή έννοια, το cloud computing είναι ένα δίκτυο που παρέχει πόρους

2. Το "Cloud" είναι μια εφαρμογή.

Ε. Τι είναι υποδομή ως υπηρεσία (IaaS);

1. Το Infrastructure-as-a-Service είναι μία από τις κατηγορίες υπηρεσιών

2. Το Infrastructure-as-a-Service είναι μία από τις κύριες κατηγορίες υπηρεσιών που παρέχει εικονικούς υπολογιστικούς πόρους, όπως εικονικές μηχανές, αποθήκευση, δικτύωση και λειτουργικά συστήματα σε άτομα ή οργανισμούς παρόχων υπολογιστικού νέφους.

Ζ. Ποια είναι τεχνικά πλεονεκτήματα SaaS;

1. Το SaaS παρέχει ευέλικτη επεκτασιμότητα και υψηλή διαθεσιμότητα σε διαφορετικές γεωγραφικές τοποθεσίες.

2. Το SaaS μειώνει τον χρόνο παράδοσης από μήνες σε ημέρες ή και ώρες.

3. Όλα τα παραπάνω

Η. Τι είναι IaaS ;

1. Το IaaS είναι το κατώτερο επίπεδο των υπηρεσιών cloud και παρέχει κυρίως κάποιους βασικούς πόρους.

2. Το IaaS παρέχει ευέλικτη επεκτασιμότητα και υψηλή διαθεσιμότητα σε διαφορετικές γεωγραφικές τοποθεσίες.

Θ. Ποια η διαφορά μεταξύ IaaS και του PaaS;

1. Η διαφορά μεταξύ IaaS και του PaaS είναι ότι οι χρήστες πρέπει να ελέγχουν το υποκείμενο επίπεδο και να εφαρμόζουν τη λογική χρήσης της υποδομής .

2. Δεν υπάρχουν διαφορές μεταξύ IaaS και του PaaS

Ι. Τι ονομάζετε δημόσιο σύννεφο (Public Cloud);

1. Ονομάζεται δημόσιο όταν προσφέρονται υπηρεσίες όχι σε ατομικό επίπεδο
2. Ονομάζεται δημόσιο όταν προσφέρονται εφαρμογές και υπηρεσίες στο ευρύ κοινό για χρήση.

Κ. Τι είναι το NoSQL;

1. Το NoSQL είναι το κατώτερο επίπεδο των υπηρεσιών cloud και παρέχει κυρίως κάποιους βασικούς πόρους.
2. NoSQL είναι ένα μη σχεσιακό σύστημα διαχείρισης δεδομένων, το οποίο δεν απαιτεί σταθερό σχήμα.

Άσκηση

Να γράψετε ένα applet το οποίο να εμφανίζει η φράση “JAVA IS FUN !!” δύο φορές με χρώματα κόκκινο και πράσινο και με χαρακτήρες BOLD+ITALIC των 35 points. Στη συνέχεια γράψτε ένα απλό αρχείο HTML το οποίο να καλεί το applet. Χρησιμοποιείστε έναν Web Browser για να εκτελέσετε την εφαρμογή.

Βιβλιογραφία

Γκλαβά, Μαρία (2019) Συστήματα βάσεων δεδομένων Θεωρία και πρακτικές εφαρμογές με: MS Access, MS SQL Server, MySQL.

Ramez Elmasri(2007) Θεμελιώδεις αρχές συστημάτων βάσεων δεδομένων.

Αθανάσιος Σταυρακούδης (2010) Βάσεις δεδομένων και SQL, Μια πρακτική προσέγγιση.

Αριστείδης Σ. Μπούρας,(2017) Αλγοριθμική και προγραμματισμός υπολογιστών σε Python, Οδηγός για νέους και έμπειρους προγραμματιστές.

Απαντήσεις

Ενότητα 1 Απαντήσεις

- Α-3
- Β-1
- Γ-2
- Δ-2
- Ε-1
- Ζ-3
- Η-3
- Θ-2
- Ι-1
- Κ-3

Απαντήσεις Ενότητας 2

- Α.1
- Β.2
- Γ.3
- Δ.2
- Ε.1
- Ζ.1
- Η.3
- Θ.1
- Ι.3
- Κ.1

Απαντήσεις ενότητας 3

- Α.2
- Β.1
- Γ.3
- Δ.3
- Ε.1
- Ζ.2
- Η.3
- Θ.3
- Ι.3
- Κ.1

Απαντήσεις Ενότητας 4

- Α. 1
- Β. 3
- Γ. 3
- Δ.1
- Ε. 2
- Ζ. 1
- Η.1
- Θ.3
- Ι.2
- Κ.2

Απαντήσεις ενότητας 5

- Α. 1
- Β.2
- Γ. 3
- Δ 1
- Ε. 2
- Ζ. 1
- Η. 2

- Θ-1
- Ι.2
- Κ-2

Απαντήσεις ενότητας 6

- Α .1
- Β .2
- Γ. 3
- Δ .2
- Ε .1
- Ζ . 1
- Η.3
- Θ. 1
- Ι. 2
- Κ . 1

Απαντήσεις 7

- Α. 1
- Β.2
- Γ.3
- Δ.1
- Ε. 3
- Ζ. 1
- Η. 3
- Θ.3
- Ι.1
- Κ.2

Απαντήσεις 8

- Α.1
- Β.2
- Γ. 3

- Δ. 1
- E.2
- Z. 3
- H. 1
- Θ. 1
- I. 2
- K.2