



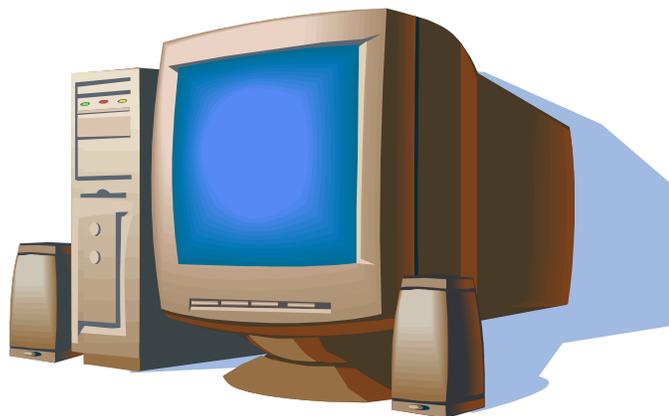
**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ**

TECHNOLOGICAL EDUCATIONAL INSTITUTE OF EPIRUS

ΤΜΗΜΑ ΤΗΛΕΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΗΛΕΚΤΡΟΝΙΚΟ ΚΑΤΑΣΤΗΜΑ



**ΜΠΑΡΚΟΥΛΑΣ ΑΡΙΣΤΕΙΔΗΣ ΑΜ:3524
ΣΑΜΑΡΤΖΗΣ ΓΕΩΡΓΙΟΣ ΑΜ:883**

ΕΠΙΒΛΕΠΩΝ: ΓΛΑΒΑΣ ΕΥΡΗΠΙΔΗΣ

2006

ΠΕΡΙΕΧΟΜΕΝΑ

Περίληψη.....	3
Εισαγωγή.....	4
Κεφάλαιο 1^ο – Internet.....	6
1.1 Η ιστορία του Internet.....	6
1.2 Τι είναι το Internet.....	10
1.2.1 Μερικοί ορισμοί.....	10
1.2.2 Δυο βασικά χαρακτηριστικά του Internet.....	13
1.2.3 Τι μας προσφέρει το Internet.....	13
1.3 Οι υπηρεσίες του Internet.....	14
1.3.1 Το μοντέλο πελάτη-εξυπηρετητή (client-server).....	14
1.3.2 Οι κυριότερες υπηρεσίες του Internet.....	16
1.4 World Wide Web (WWW).....	18
1.4.1 Τι είναι το World Wide Web.....	18
1.4.2 Υπερκείμενο και Υπερμέσα.....	19
1.4.3 Λειτουργία του WWW.....	20
1.4.4 Το WWW και οι άλλες υπηρεσίες του Internet.....	22
1.4.5 Η έννοια του Uniform Resource Locator (URL).....	22
1.4.6 Προγράμματα για το WWW (Browsers).....	25
Κεφάλαιο 2^ο – Εισαγωγή στο ηλεκτρονικό εμπόριο.....	26
2.1 Η ανάγκη του ηλεκτρονικού εμπορίου.....	26
2.2 Ορισμός ηλεκτρονικού εμπορίου.....	27
2.3 Η σημερινή πραγματικότητα.....	27
2.4 Κατηγορίες του ηλεκτρονικού εμπορίου.....	29
2.5 Πλεονεκτήματα ηλεκτρονικού εμπορίου.....	30
2.6 Μειονεκτήματα ηλεκτρονικού εμπορίου.....	33
2.7 Στοιχεία ενός ηλεκτρονικού καταστήματος.....	36
2.8 Πρότυπα ηλεκτρονικού εμπορίου.....	38
2.9 Παραδείγματα ιστοσελίδων ηλεκτρονικού εμπορίου.....	40
Κεφάλαιο 3^ο – Βασικές τεχνολογίες για το ηλεκτρονικό εμπόριο.....	43
Κεφάλαιο 4^ο – HTML (Hypertext Markup Language).....	45
4.1 Πως κατασκευάζεται μια απλή Web σελίδα.....	45
4.2 Ορισμός παραγράφων.....	46
4.3 Αλλαγές γραμμών.....	47
4.4 Παραπομπές (Links).....	47
4.5 Τίτλος σελίδας.....	50
4.6 Πως ορίζουμε το χρώμα του φόντου και του κειμένου της σελίδας.....	50
4.7 Λίστες (αριθμημένες και μη).....	51
4.8 Επικεφαλίδες (Headings).....	53
4.9 Στοίχιση <ALIGN>.....	53
4.10 Ορισμός γραμματοσειρών	54
4.11 Εισαγωγή εικόνων και γραφικών.....	55
4.12 Πως δημιουργούμε απλούς πίνακες.....	57
4.13 Πως δημιουργούμε μια φόρμα επικοινωνίας.....	63
4.14 Πως κάνουμε μια σελίδα να καλεί αυτόματα μια άλλη (redirection).....	68
4.15 Πως κάνουμε μια σελίδα να καλεί κατά τακτά χρονικά διαστήματα τον εαυτό της (refresh).....	69

4.16 Πως προσθέτουμε σχόλια στον κώδικα μιας σελίδας.....	69
Κεφάλαιο 5^ο – SQL (Structured Query Language).....	71
5.1 Τι είναι η SQL.....	71
5.2 Λίγα λόγια για τους πίνακες βάσεων δεδομένων (Database Tables).....	72
5.3 Τα ερωτήματα της SQL (SQL Queries).....	72
5.4 Διαχείριση δεδομένων της SQL (Data Manipulation).....	73
5.5 Ορισμός δεδομένων της SQL (Data Definition).....	73
5.6 Εντολές και λέξεις κλειδιά στην SQL.....	74
5.6.1 Η εντολή Select της SQL.....	74
5.6.2 Το Where Clause της SQL.....	75
5.6.3 Η συνθήκη Like.....	77
5.6.4 Οι λογικοί τελεστές And και Or.....	77
5.6.5 Ο τελεστής Between...And.....	78
5.6.6 Η λέξη κλειδί Distinct.....	80
5.6.7 Η λέξη κλειδί Order By.....	82
5.6.8 Η εντολή Insert Into.....	84
5.6.9 Η εντολή Update.....	85
5.6.10 Η εντολή Delete.....	86
5.6.11 Οι συναρτήσεις Count της SQL.....	87
5.6.12 Οι λέξεις κλειδιά Count και Distinct.....	89
5.6.13 Οι συναρτήσεις της SQL.....	89
5.6.14 Η συνάρτηση AVG (column).....	90
5.6.15 Η συνάρτηση MAX (column).....	90
5.6.16 Η συνάρτηση MIN (column).....	90
5.6.17 Η συνάρτηση SUM (column).....	91
5.6.18 Η λέξη κλειδί Group By.....	91
5.7 Ένωση Πινάκων (Join).....	93
5.8 Δημιουργία βάσης δεδομένων και πίνακα.....	95
5.9 Διαγραφή βάσης δεδομένων και πίνακα.....	96
5.10 Η εντολή Alter Table.....	97
Κεφάλαιο 6^ο – Σύνοψη ανάλυση των τεχνολογιών του ηλεκτρονικού καταστήματος.....	98
Κεφάλαιο 7^ο – Υλοποίηση του ηλεκτρονικού καταστήματος.....	102
7.1 Η σελίδα του Πελάτη.....	102
7.2 Η σελίδα του Διαχειριστή.....	108
7.3 Περιγραφή της Βάσης Δεδομένων.....	113
Συμπεράσματα.....	117
Παράρτημα.....	118
Βιβλιογραφία.....	135

ΠΕΡΙΛΗΨΗ

Σκοπός της εργασίας αυτής ήταν η ανάπτυξη ηλεκτρονικού καταστήματος πώλησης hardware ηλεκτρονικών υπολογιστών. Για την υλοποίηση του δικτυακού τόπου χρησιμοποιήθηκε η τεχνολογία PHP, δημιουργώντας έτσι δυναμικές ιστοσελίδες σε συνδυασμό με τις τεχνολογίες, HTML, MySQL και του server SAMBAR. Το αποτέλεσμα ήταν ένα πλήρες και εύχρηστο ηλεκτρονικό κατάστημα με αρκετές δυνατότητες τόσο για τον πελάτη όσο και για τον ιδιοκτήτη-διαχειριστή. Η ύπαρξη της διασυνδεδεμένης βάσης δεδομένων δίνει στον ιδιοκτήτη τη δυνατότητα της εύκολης διαχείρισης των προϊόντων και των παραγγελιών. Ειδικότερα σχεδιάστηκε και αναπτύχθηκε μια βάση δεδομένων, με τη χρήση της MySQL, για την τοποθέτηση των πληροφοριών των προϊόντων, των πελατών και των παραγγελιών. Το κατάστημα παρέχει τη δυνατότητα εγγραφής του χρήστη, εύκολης προσπέλασης των προϊόντων, δημιουργίας καλαθιού αγορών, παραγγελίας και παρακολούθησής της. Η αγορά των προϊόντων γίνεται με πιστωτική κάρτα, αφού συμπληρωθεί η αντίστοιχη φόρμα από το χρήστη ή με αντικαταβολή αφού τα προϊόντα παραδοθούν μέσω ταχυδρομείου.

ΕΙΣΑΓΩΓΗ

Η ανάπτυξη των τηλεπικοινωνιών και της πληροφορικής έδωσαν νέες δυνατότητες στο διαδίκτυο, κάνοντάς το πιο προσιτό σε μεγαλύτερο αριθμό πληθυσμού. Θεωρείται πλέον δεδομένη η ύπαρξή του σε κάθε σπίτι και χώρο εργασίας και επιπλέον κάθε μήνα καλωσορίζει εκατομμύρια νέους χρήστες ανά τον κόσμο. Η μαζικότητα αυτή του διαδικτύου δε θα μπορούσε να μείνει ανεκμετάλλευτη από τις μεγάλες εταιρείες, οι οποίες εμβαθύνοντας στις δυνατότητες που παρέχει, διαπίστωσαν ότι ανοιγόταν ένας μεγάλος δρόμος μπροστά τους που οδηγούσε με σιγουριά σε αυξημένα και σίγουρα κέρδη. Σύμφωνα με μια πρόσφατη μελέτη που έγινε από τη CISCO SYSTEMS, η οικονομία του internet αναμένεται να φτάσει τα 508 δις. δολάρια. Το ηλεκτρονικό εμπόριο παράγει ήδη περισσότερα χρήματα από τις τηλεπικοινωνίες και τις αεροπορικές εταιρείες.

Ένας τομέας του ηλεκτρονικού εμπορίου είναι τα ηλεκτρονικά καταστήματα που αποτελούν ένα επιτυχημένο μέσο προώθησης και πώλησης προϊόντων. Μεγάλο παράδειγμα οι δυο δικτυακοί τόποι eBay και Amazon που από μικρά ηλεκτρονικά καταστήματα εξελίχθηκαν σε παγκόσμιους κολοσσούς.

Πριν λίγα χρόνια η δημιουργία δικτυακών τόπων με δυνατότητες ηλεκτρονικού εμπορίου ήταν μια χρονοβόρα και πολύπλοκη εργασία. Η διεκπεραίωση της απαιτούσε τη χρήση μιας γλώσσας προγραμματισμού χαμηλού επιπέδου όπως η C++ ή τη χρήση της πολύπλοκης γλώσσας PERL. Με την ανάπτυξη της τεχνολογίας PHP η δημιουργία ηλεκτρονικών καταστημάτων είναι πλέον ευκολότερη, ποιοτικότερη και πιο γρήγορη. Γι' αυτό το λόγο επιλέχθηκε η συγκεκριμένη τεχνολογία για την υλοποίηση του ηλεκτρονικού καταστήματος, που σε συνδυασμό με την HTML και τη MySQL, αποτέλεσαν τα βασικά εργαλεία.

Συγκεκριμένος στόχος της παρούσας εργασίας ήταν η υλοποίηση ενός εύχρηστου ηλεκτρονικού καταστήματος πώλησης hardware υπολογιστών. Ειδικότερα σχεδιάστηκε και αναπτύχθηκε μια βάση δεδομένων, με τη χρήση της MySQL, για τοποθέτηση των πληροφοριών των προϊόντων, των πελατών και των παραγγελιών. Το κατάστημα παρέχει τη δυνατότητα εγγραφής του χρήστη, εύκολης προσπέλασης των προϊόντων, δημιουργίας

καλαθιού αγορών, παραγγελίας και παρακολούθησης της κατάστασής της. Ο ιδιοκτήτης-διαχειριστής του καταστήματος έχει τη δυνατότητα της εύκολης διαχείρισης και μετατροπής, αν το επιθυμεί, της βάσης δεδομένων των προϊόντων και των παραγγελιών. Με τη χρήση των HTML και PHP, σχεδιάστηκαν ευπαρουσίαστες και δυναμικές ιστοσελίδες κάνοντας την πλοήγηση φιλική στο χρήστη.

Συνοπτικά το βιβλίο αυτό χωρίζεται σε επτά κεφάλαια, στο παράρτημα και κάποια γενικά συμπεράσματα.

Κεφάλαιο 1: Γενικά για το Internet (διαδίκτυο), WWW, Browsers, Servers, URLs κ.λ.π.

Κεφάλαιο 2: Η έννοια του ηλεκτρονικού εμπορίου και των ηλεκτρονικών καταστημάτων στο Internet.

Κεφάλαιο 3: Βασικές τεχνολογίες που απαιτούνται για το ηλεκτρονικό εμπόριο

Κεφάλαιο 4: Λεπτομερής ανάλυση της γλώσσας HTML (HyperText Markup Language), δυνατότητες, βασικές της έννοιες και εντολές.

Κεφάλαιο 5: Λεπτομερής ανάλυση της SQL (Structured Query Language), πρόσβαση σε βάση δεδομένων και επεξεργασία των δεδομένων της.

Κεφάλαιο 6: Σύντομη ανάλυση των τεχνολογιών που χρησιμοποιήθηκαν για την ανάπτυξη του ηλεκτρονικού καταστήματος.

Κεφάλαιο 7: Περιγραφή της λειτουργίας του ηλεκτρονικού μας καταστήματος.

Συμπεράσματα, γενικά για το ηλεκτρονικό κατάστημα.

Παράρτημα: Παρατίθεται ο κώδικας των αναγκαίων αρχείων για την πρακτική λειτουργία του ηλεκτρονικού καταστήματος.

ΚΕΦΑΛΑΙΟ 1^ο

INTERNET

Το Internet είναι ένα πλέγμα από εκατομμύρια διασυνδεδεμένους υπολογιστές που εκτείνεται σχεδόν σε κάθε γωνιά του πλανήτη και παρέχει τις υπηρεσίες του σε εκατομμύρια χρήστες.

Αποτελεί ένα “Παγκόσμιο Ηλεκτρονικό Χωριό”, οι “κάτοικοι” του οποίου μοιράζονται πληροφορίες και ανταλλάσσουν απόψεις πέρα από γεωγραφικά σύνορα. Σύμφωνα με τις σχετικές εκτιμήσεις, αυτός ο παγκόσμιος ιστός υπολογιστών και χρηστών αριθμεί σήμερα πάνω από εκατό εκατομμύρια υπολογιστές και ένα δισεκατομμύριο χρήστες, ενώ επεκτείνεται διαρκώς με εκθετικούς ρυθμούς.

1.1 Η ιστορία του Internet

Το σημερινό Internet αποτελεί εξέλιξη του **ARPANET**, ενός δικτύου που άρχισε να αναπτύσσεται σαν ένα στρατιωτικό ερευνητικό έργο στα τέλη της δεκαετίας του 60 στις ΗΠΑ.

Δεκαετία ‘60: ένα ενδιαφέρον πείραμα ξεκινά

Στα πανεπιστήμια των ΗΠΑ οι ερευνητές ξεκινούν να πειραματίζονται με τη διασύνδεση απομακρυσμένων υπολογιστών μεταξύ τους. Το δίκτυο **ARPANET** γεννιέται το 1969 με πόρους του προγράμματος ARPA (Advanced Research Project Agency) του Υπουργείου Άμυνας, με σκοπό να συνδέσει το Υπουργείο με στρατιωτικούς ερευνητικούς οργανισμούς και να αποτελέσει ένα πείραμα για τη μελέτη της αξιόπιστης λειτουργίας των δικτύων. Στην αρχική του μορφή, το πρόγραμμα απέβλεπε στον πειραματισμό με μια νέα τεχνολογία γνωστή σαν μεταγωγή πακέτων (packet switching),

σύμφωνα με την οποία τα προς μετάδοση δεδομένα κόβονται σε πακέτα και πολλοί χρήστες μπορούν να μοιραστούν την ίδια επικοινωνιακή γραμμή.

Στόχος ήταν η δημιουργία ενός διαδικτύου που θα εξασφάλιζε την επικοινωνία μεταξύ απομακρυσμένων δικτύων, έστω και αν κάποια από τα ενδιάμεσα συστήματα βρίσκονταν προσωρινά εκτός λειτουργίας. Κάθε πακέτο θα είχε την πληροφορία που χρειάζονταν για να φτάσει στον προορισμό του, όπου και θα γινόταν η επανασύνθεσή του σε δεδομένα τα οποία μπορούσε να χρησιμοποιήσει ο τελικός χρήστης.

Το παραπάνω σύστημα θα επέτρεπε σε υπολογιστές να μοιράζονται δεδομένα και σε ερευνητές να υλοποιήσουν το ηλεκτρονικό ταχυδρομείο.

Δεκαετία '70: οι πρώτες συνδέσεις

Το 1973, ξεκινά ένα νέο ερευνητικό πρόγραμμα που ονομάζεται Interneting Project (Πρόγραμμα Διαδικτύωσης) προκειμένου να ξεπεραστούν οι διαφορετικοί τρόποι που χρησιμοποιεί κάθε δίκτυο για να διακινεί τα δεδομένα του. Στόχος είναι η διασύνδεση πιθανώς ανόμοιων δικτύων και η ομοιόμορφη διακίνηση δεδομένων από το ένα δίκτυο στο άλλο. Από την έρευνα γεννιέται μια νέα τεχνική, το **Internet Protocol (IP)** (Πρωτόκολλο Διαδικτύωσης), από την οποία θα πάρει αργότερα το όνομά του το Internet. Διαφορετικά δίκτυα που χρησιμοποιούν το κοινό πρωτόκολλο IP μπορούν να συνδέονται και να αποτελούν ένα διαδίκτυο. Σε ένα δίκτυο IP όλοι οι υπολογιστές είναι ισοδύναμοι, οπότε τελικά οποιοσδήποτε υπολογιστής του διαδικτύου μπορεί να επικοινωνεί με οποιονδήποτε άλλον.

Επίσης, σχεδιάζεται μια άλλη τεχνική για τον έλεγχο της μετάδοσης των δεδομένων, το **Transmission Control Protocol (TCP)** (Πρωτόκολλο Ελέγχου Μετάδοσης). Ορίζονται προδιαγραφές για τη μεταφορά αρχείων μεταξύ υπολογιστών (FTP) και για το ηλεκτρονικό ταχυδρομείο (E-mail). Σταδιακά συνδέονται με το ARPANET ιδρύματα από άλλες χώρες, με πρώτα το University College of London (Αγγλία) και το Royal Radar Establishment (Νορβηγία).

Δεκαετία '80: ένα παγκόσμιο δίκτυο για την ακαδημαϊκή κοινότητα

Το 1983, το πρωτόκολλο **TCP/IP** (δηλ. ο συνδυασμός των TCP και IP) αναγνωρίζεται ως πρότυπο από το Υπουργείο Άμυνας των ΗΠΑ. Η έκδοση του λειτουργικού συστήματος Berkeley UNIX το οποίο περιλαμβάνει το TCP/IP συντελεί στη γρήγορη εξάπλωση της διαδικτύωσης των υπολογιστών. Εκατοντάδες Πανεπιστήμια συνδέουν τους υπολογιστές τους στο ARPANET, το οποίο επιβαρύνεται πολύ και το 1983, χωρίζεται σε δύο τμήματα: στο MILNET (για στρατιωτικές επικοινωνίες) και στο νέο ARPANET (για χρήση αποκλειστικά από την πανεπιστημιακή κοινότητα και συνέχιση της έρευνας στη δικτύωση).

Το 1985, το National Science Foundation (NSF) δημιουργεί ένα δικό του γρήγορο δίκτυο, το **NSFNET** χρησιμοποιώντας το πρωτόκολλο TCP/IP, προκειμένου να συνδέσει πέντε κέντρα υπερ-υπολογιστών μεταξύ τους και με την υπόλοιπη επιστημονική κοινότητα. Στα τέλη της δεκαετίας του '80, όλο και περισσότερες χώρες συνδέονται στο NSFNET (Καναδάς, Γαλλία, Σουηδία, Αυστραλία, Γερμανία, Ιταλία, κ.α.). Χιλιάδες πανεπιστήμια και οργανισμοί δημιουργούν τα δικά τους δίκτυα και τα συνδέουν πάνω στο παγκόσμιο αυτό δίκτυο το οποίο αρχίζει να γίνεται γνωστό σαν **INTERNET** και να εξαπλώνεται με τρομερούς ρυθμούς σε ολόκληρο τον κόσμο. Το 1990, το ARPANET πλέον καταργείται.

Δεκαετία '90: ένα παγκόσμιο δίκτυο για όλους

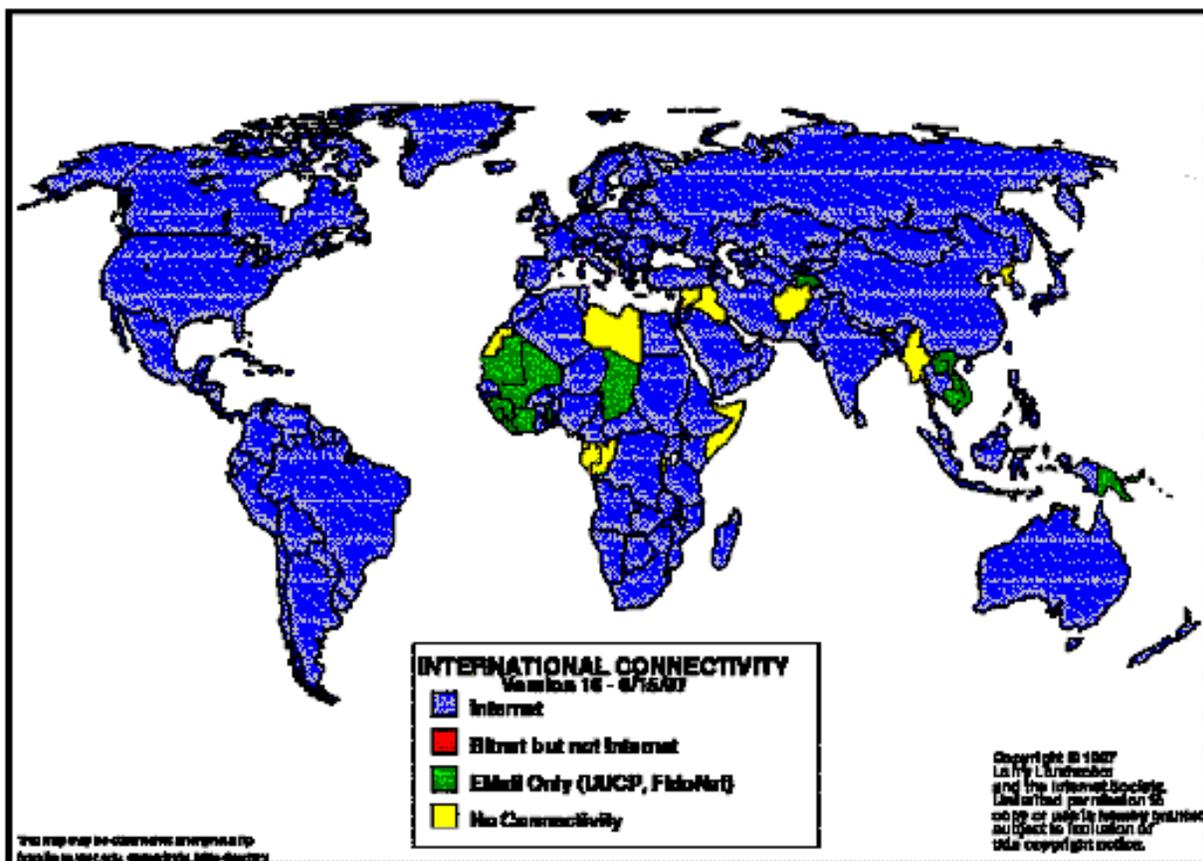
Όλο και περισσότερες χώρες συνδέονται στο NSFNET, μεταξύ των οποίων και η Ελλάδα το 1990.

Το 1993, το εργαστήριο CERN στην Ελβετία παρουσιάζει το **World Wide Web (WWW)** (Παγκόσμιο Ιστό) που αναπτύχθηκε από τον Tim Berners-Lee. Πρόκειται για ένα σύστημα διασύνδεσης πληροφοριών σε μορφή πολυμέσων (multimedia) που βρίσκονται αποθηκευμένες σε χιλιάδες υπολογιστές του Internet σε ολόκληρο τον κόσμο και παρουσίασής τους σε ηλεκτρονικές σελίδες, στις οποίες μπορεί να περιηγηθεί κανείς χρησιμοποιώντας το ποντίκι. Το γραφικό αυτό περιβάλλον έκανε την εξερεύνηση του Internet προσιτή στον απλό χρήστη. Παράλληλα, εμφανίζονται στο Internet διάφορα εμπορικά δίκτυα που ανήκουν σε εταιρίες παροχής υπηρεσιών Internet (Internet Service Providers - ISP) και προσφέρουν πρόσβαση στο Internet για όλους. Οποιοσδήποτε διαθέτει PC και modem μπορεί να συνδεθεί με το Internet σε τιμές που μειώνονται

διαρκώς. Το 1995, το NSFNET καταργείται πλέον επίσημα και το φορτίο του μεταφέρεται σε εμπορικά δίκτυα.

Η ανακάλυψη του WWW σε συνδυασμό με την ευκολία απόκτησης πρόσβασης στο Internet προσέλκυσε έναν μεγάλο αριθμό καινούργιων .

Σήμερα, όπως φαίνεται και από την παραπάνω εικόνα, το μεγαλύτερο μέρος του πληθυσμού της Γης ζει σε χώρες που είναι συνδεδεμένες στο Internet. Παρατηρούμε ότι καθημερινά περιοδικά και εφημερίδες εκδίδονται “on-line” και μας παραπέμπουν στις διευθύνσεις τους, επιχειρήσεις και ιδιώτες φτιάχνουν τις δικές τους σελίδες στο WWW, κλπ. Είναι προφανές ότι το Internet δεν αποτελεί πλέον ένα δίκτυο των φοιτητών και των ερευνητών, αλλά ότι επεκτείνεται και επιδρά στις καθημερινές πρακτικές όλων μας. Ήδη μιλάμε για ηλεκτρονικό εμπόριο, τηλεεργασία, τηλεεκπαίδευση, τηλεϊατρική, κλπ. μέσα από το Internet.



Εικόνα 1.1: Η κατάσταση σύνδεσης ανά χώρα, όπως είχε στις 15/6/1997. Με μπλε εμφανίζονται οι χώρες με πλήρη σύνδεση στο Internet, με πράσινο οι χώρες που διαθέτουν πρόσβαση μόνον στην υπηρεσία E-mail και με κίτρινο οι χώρες που δεν διαθέτουν κανένα είδος σύνδεσης.

1.2 Τι είναι το Internet

1.2.1 Μερικοί ορισμοί

Το Internet είναι ένα **διαδίκτυο**, δηλαδή ένα δίκτυο αποτελούμενο από δίκτυα υπολογιστών. Ας δούμε λοιπόν πρώτα τι είναι **δίκτυο** υπολογιστών:

Δύο ή περισσότεροι υπολογιστές που συνδέονται μεταξύ τους σχηματίζουν ένα δίκτυο. Οι κυριότεροι λόγοι ύπαρξης ενός δικτύου είναι να μπορούν οι χρήστες των υπολογιστών να επικοινωνούν μεταξύ τους και να χρησιμοποιούν από απόσταση τις υπηρεσίες που προσφέρει κάποιος υπολογιστής του δικτύου.

Ένα σύνολο από κανόνες που ονομάζεται **πρωτόκολλο δικτύωσης**, καθορίζει το πώς επικοινωνούν μεταξύ τους οι υπολογιστές του δικτύου. Η φυσική διάταξη των συνδέσεων του δικτύου ονομάζεται **τοπολογία**. Οι τρεις πιο συνηθισμένες τοπολογίες είναι:

Αστέρας (star)

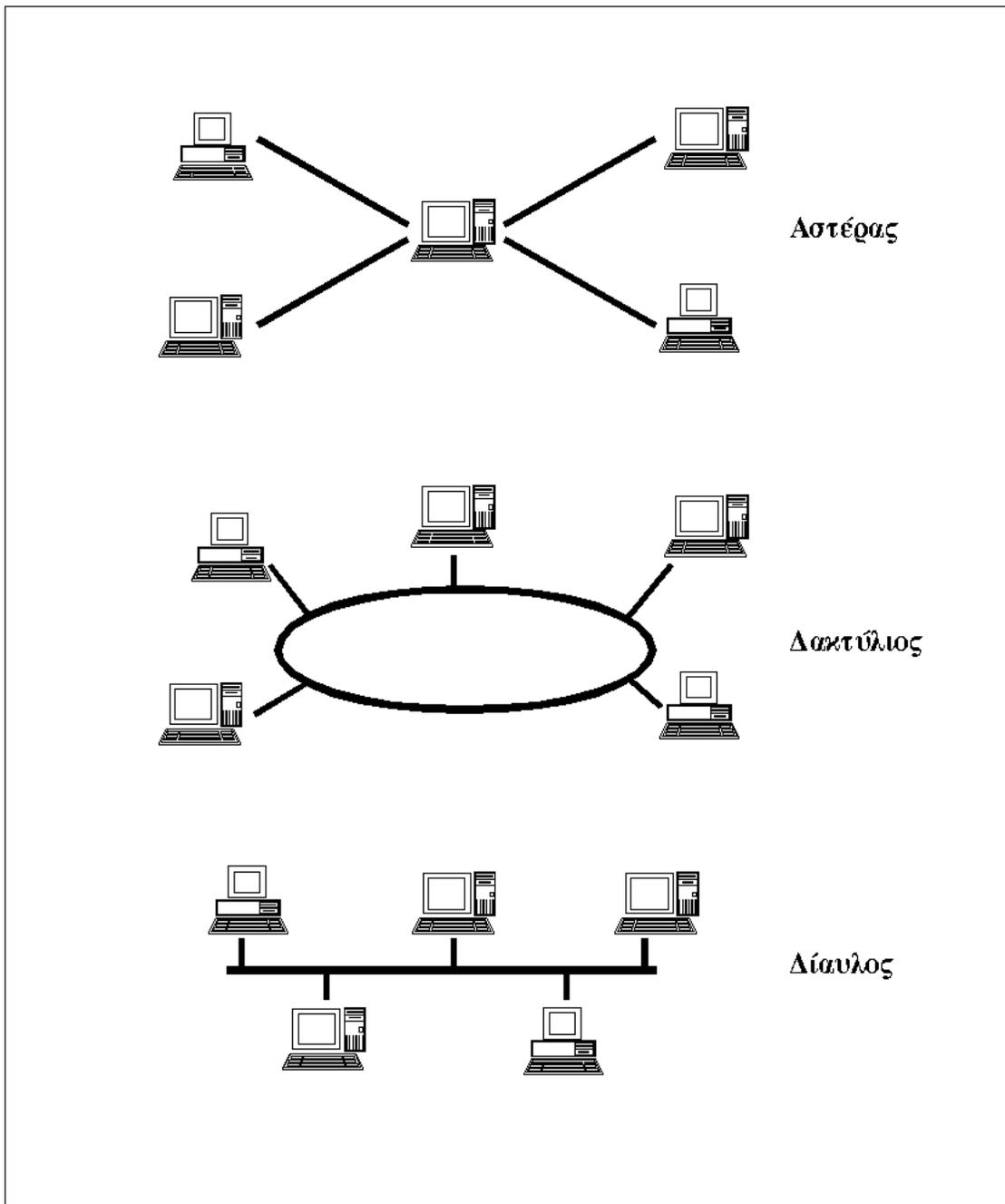
Υπάρχει ένας κεντρικός υπολογιστής στον οποίον συνδέονται οι υπόλοιποι υπολογιστές του δικτύου.

Δακτύλιος (ring)

Όλοι οι υπολογιστές είναι συνδεδεμένοι σε έναν πλήρη κλειστό δακτύλιο.

Δίαυλος (bus)

Όλοι οι υπολογιστές συνδέονται κατά μήκος ενός κεντρικού αγωγού.



Εικόνα 1.2: Οι κυριότερες τοπολογίες δικτύων

Τα δίκτυα, ανάλογα με το εύρος της περιοχής που καλύπτουν, χωρίζονται σε 3 κατηγορίες:

Τοπικά Δίκτυα (Local Area Network - LAN)

Συνδέουν υπολογιστές που απέχουν μεταξύ τους μικρές αποστάσεις, π.χ. υπολογιστές που βρίσκονται στο ίδιο ή σε γειτονικά κτίρια.

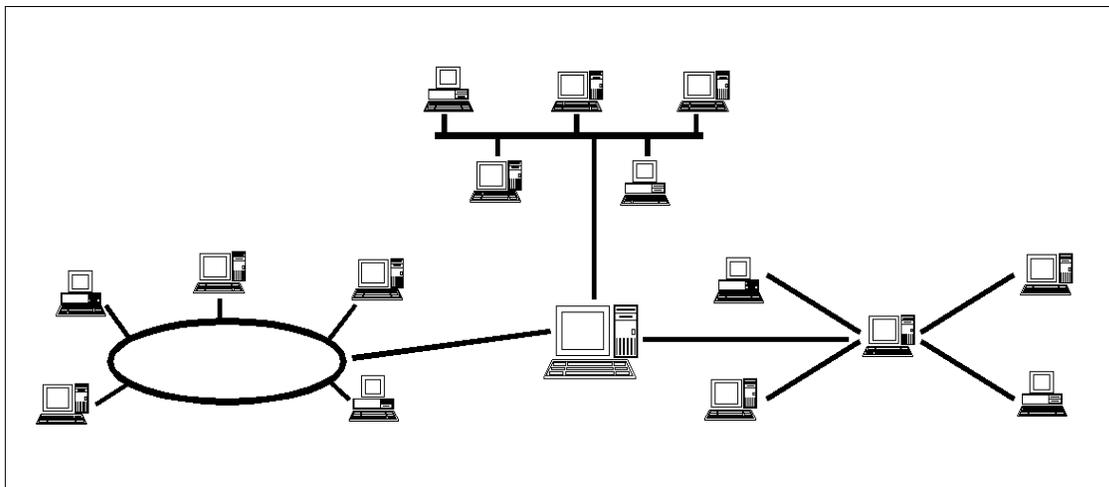
Δίκτυα Μητροπολιτικής Περιοχής (Metropolitan Area Network - MAN)

Συνδέουν υπολογιστές που απέχουν μεταξύ τους μεσαίες αποστάσεις, π.χ. υπολογιστές που βρίσκονται σε διαφορετικά σημεία της ίδιας πόλης.

Δίκτυα Ευρείας Περιοχής (Wide Area Network - WAN)

Συνδέουν υπολογιστές που απέχουν μεταξύ τους μεγάλες αποστάσεις, π.χ. υπολογιστές που βρίσκονται σε διαφορετικές πόλεις.

Διαδίκτυο είναι ένα δίκτυο από δίκτυα. Π.χ. τρία διαφορετικά τοπικά δίκτυα μπορούν να συνδεθούν μεταξύ τους σχηματίζοντας ένα διαδίκτυο, όπως φαίνεται στην εικόνα που ακολουθεί:



Εικόνα 1.3: Ένα δίκτυο δικτύων

Με τον όρο “Internet” δεν εννοούμε οποιοδήποτε διαδίκτυο, αλλά **το Παγκόσμιο Διαδίκτυο**, δηλαδή η συνένωση των χιλιάδων δικτύων διαφόρων μεγεθών που καλύπτει σχεδόν ολόκληρη την υδρόγειο.

1.2.2 Δύο βασικά χαρακτηριστικά του Internet

Ένα βασικό χαρακτηριστικό του Internet είναι ότι μπορεί να **συνδέει υπολογιστές διαφορετικού τύπου**, δηλ. υπολογιστές που μπορεί να διαφέρουν όσον αφορά την αρχιτεκτονική του υλικού (hardware), το λειτουργικό σύστημα που χρησιμοποιούν και το πρωτόκολλο δικτύωσης που εφαρμόζεται στο τοπικό τους δίκτυο. Ακριβώς εξαιτίας αυτής της ευελιξίας του, εξαπλώθηκε σε ολόκληρο τον πλανήτη κατά τη διάρκεια των τελευταίων δεκαετιών.

Ένα άλλο ενδιαφέρον χαρακτηριστικό του Internet είναι ότι είναι **αποκεντρωμένο** και **αυτοδιαχειριζόμενο**. Δεν υπάρχει δηλαδή κάποιος κεντρικός οργανισμός που να το διευθύνει και να παίρνει συνολικά αποφάσεις σχετικά με το είδος των πληροφοριών που διακινούνται, τις υπηρεσίες που παρέχονται από τους διάφορους υπολογιστές του ή τη διαχείρισή του. Καθένα από τα μικρότερα δίκτυα που το αποτελούν διατηρεί την αυτονομία του και είναι το ίδιο υπεύθυνο για το είδος των πληροφοριών που διακινεί, τις υπηρεσίες που προσφέρουν οι υπολογιστές του και τη διαχείρισή του.

1.2.3 Τι μας προσφέρει το Internet

Οι άνθρωποι χρησιμοποιούν το Internet βασικά για δύο πράγματα: α) για να **αντλήσουν πληροφορίες** και β) για να **επικοινωνήσουν** με άλλους ανθρώπους που είναι κι αυτοί χρήστες του.

Μπορούμε να θεωρήσουμε το Internet σαν μια τεράστια αποθήκη πληροφορίας, μια παγκόσμια βιβλιοθήκη. Στους υπολογιστές του, βρίσκονται αποθηκευμένα χιλιάδες Terabytes πληροφορίας, αρκετά από τα οποία διατίθενται ελεύθερα στους χρήστες του. Έτσι λοιπόν έχουμε τη δυνατότητα να χρησιμοποιούμε απομακρυσμένες βάσεις δεδομένων, να ανακτάμε αρχεία με προγράμματα, εικόνες, κείμενα, κλπ., να έχουμε πρόσβαση σε βιβλιοθήκες, να διαβάζουμε ηλεκτρονικές εφημερίδες και περιοδικά, ακόμη και να παρακολουθούμε ραδιοφωνικά προγράμματα.

1.3 Ο υπηρεσίες του Internet

Οι χρήστες Internet σε ολόκληρο τον κόσμο έχουν τη δυνατότητα να χρησιμοποιούν μια ποικιλία υπηρεσιών. Αυτό που είναι σημαντικό να κατανοήσουμε είναι ότι όλοι οι χρήστες, δεν έχουν πρόσβαση στις **ίδιες** υπηρεσίες. Στη συνέχεια, θα εξηγήσουμε το μοντέλο πελάτη-εξυπηρετητή (client-server) και θα αναφέρουμε τις πιο διαδεδομένες υπηρεσίες.

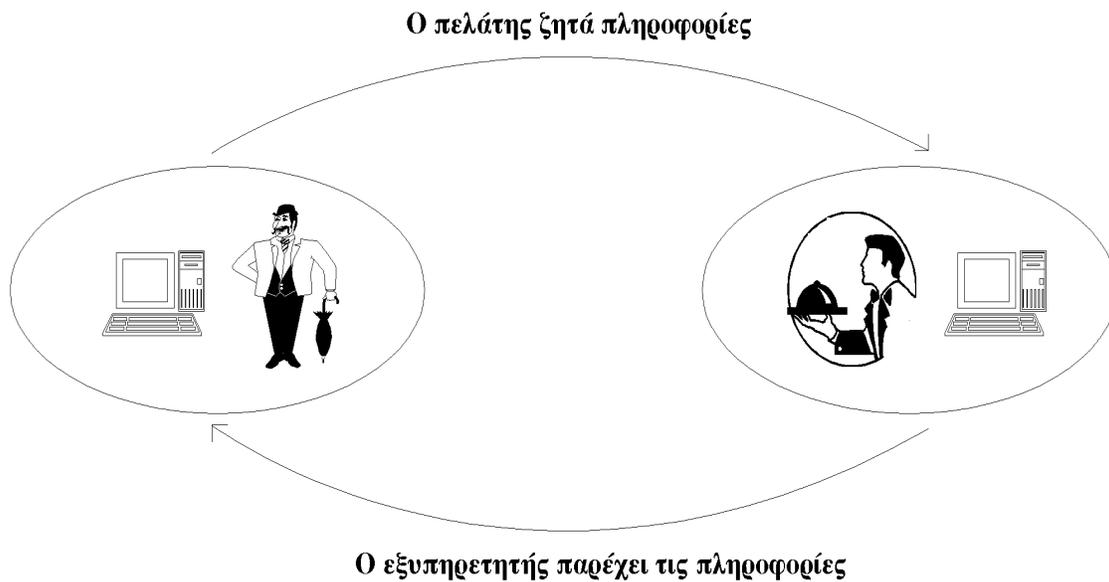
1.3.1 Το μοντέλο πελάτη-εξυπηρετητή (client-server)

Προκειμένου να χρησιμοποιήσουμε μια υπηρεσία του Internet θα πρέπει:

- να έχουμε εγκατεστημένο στον υπολογιστή μας και να εκτελέσουμε το κατάλληλο πρόγραμμα για αυτή την υπηρεσία. Το πρόγραμμα αυτό ονομάζεται **πελάτης (client)**. Μέσω του πελάτη, ζητάμε την παροχή της συγκεκριμένης υπηρεσίας.
- να έχουμε πρόσβαση (μέσω Internet) σε μηχανή που υποστηρίζει την αιτούμενη υπηρεσία. Σε αυτή τη μηχανή πρέπει να εκτελείται ένα πρόγραμμα που παρέχει τη συγκεκριμένη υπηρεσία, ο **εξυπηρετητής (server)**.

Η παροχή των περισσότερων υπηρεσιών στο Internet βασίζεται στο **μοντέλο πελάτη-εξυπηρετητή (client-server)** που λειτουργεί ως εξής:

Ο πελάτης ζητά από τον εξυπηρετητή πληροφορίες και ο τελευταίος εξυπηρετεί το αίτημα παρέχοντάς του τις πληροφορίες αυτές. Αφού τελειώσει η διαδικασία, ο εξυπηρετητής περιμένει έως ότου κάποιος πελάτης υποβάλλει πάλι κάποια αίτηση για εξυπηρέτηση.



Εικόνα 1.6: Το μοντέλο πελάτη-εξυπηρετητή

Κάθε υπηρεσία στο Internet έχει το δικό της ξεχωριστό **πρωτόκολλο**, δηλαδή το δικό της σύνολο από συμβάσεις που καθορίζουν το πώς γίνεται η “συνομιλία” του αντίστοιχου ζεύγους πελάτη-εξυπηρετητή. Έτσι, άλλο πρωτόκολλο χρησιμοποιεί η υπηρεσία WWW, άλλο η υπηρεσία FTP, άλλο η υπηρεσία E-mail, κ.ο.κ.

Ένα σημείο που είναι σημαντικό να κατανοήσουμε είναι ότι μέσω του πελάτη **δεν μπορούμε να ζητάμε την παροχή της συγκεκριμένης υπηρεσίας από οποιονδήποτε υπολογιστή του Internet**. Θα πρέπει ο υπολογιστής στον οποίον απευθύνουμε την αίτησή μας να υποστηρίζει την υπηρεσία αυτή και να εκτελεί τον αντίστοιχο εξυπηρετητή. Έστω π.χ. ότι θέλουμε να δούμε μια σελίδα του World Wide Web (WWW). Τρέχουμε στον υπολογιστή μας έναν πελάτη για WWW για να συνδεθούμε με έναν απομακρυσμένο υπολογιστή όπου είναι αποθηκευμένη η σελίδα και στον οποίον εκτελείται ένας εξυπηρετητής για WWW. Ο εξυπηρετητής στέλνει στον πελάτη μας τη σελίδα και ο πελάτης την εμφανίζει στην οθόνη του υπολογιστή μας. Σημειώνουμε ότι οι πελάτες για την υπηρεσία WWW ονομάζονται και browsers (αναζητητές).

Επίσης, σε έναν υπολογιστή μπορούν να εκτελούνται ταυτόχρονα εξυπηρετητές για περισσότερες από μία υπηρεσίες π.χ. ένας εξυπηρετητής για WWW, ένας εξυπηρετητής για FTP, κι ένας εξυπηρετητής για E-mail. Έτσι, **ο ίδιος υπολογιστής μπορεί να παρέχει περισσότερες από μία υπηρεσίες**.

Υπάρχουν διάφορα προγράμματα - πελάτες για καθεμία από τις υπηρεσίες του Internet για διάφορα λειτουργικά συστήματα. Πολλά από αυτά διατίθενται ελεύθερα μέσω του Internet και μπορούμε να τα μεταφέρουμε στον υπολογιστή μας.

1.3.2 Οι κυριότερες υπηρεσίες του Internet

E-mail (Ηλεκτρονικό Ταχυδρομείο)

Υποστηρίζει την ανταλλαγή μηνυμάτων μεταξύ χρηστών χάρη στην προσωπική ηλεκτρονική διεύθυνση του καθενός. Το περιεχόμενο του μηνύματος μπορεί να είναι κείμενο, ήχος, εικόνα, video ή δεδομένα.

Mailing lists (Λίστες E-mail)

Καθορισμένη ομάδα απομακρυσμένων μεταξύ τους χρηστών που ανταλλάσσουν μηνύματα σχετικά με κάποιο θέμα ορισμένο από κοινού, με κάποιον από αυτούς ως υπεύθυνο για την καλή λειτουργία της λίστας.

Remote Login (Τηλεσύνδεση)

Ένας χρήστης “υπάρχει”, δηλαδή έχει δικαίωμα χρήσης σε έναν ή περισσότερους υπολογιστές του δικτύου. Αν αυτοί είναι απομακρυσμένοι μεταξύ τους τότε, εργαζόμενος σε έναν από αυτούς, μπορεί να συνδεθεί με οποιονδήποτε από τους υπόλοιπους και να (τηλε-) εργαστεί σαν να ήταν παρών, δηλαδή να χρησιμοποιήσει τις δυνατότητες του απομακρυσμένου υπολογιστή σαν να βρίσκονταν στον ίδιο φυσικό χώρο με αυτόν.

Finger

Αναζήτηση της ύπαρξης ενός συγκεκριμένου χρήστη σε κάποιο σημείο του δικτύου.

FTP (File Transfer Protocol)

Μεταφορά αρχείων από απομακρυσμένο υπολογιστή σε τοπικό υπολογιστή και αντίστροφα.

Archie

Αναζήτηση υπολογιστών στο Internet που προσφέρουν την υπηρεσία FTP και περιέχουν πληροφορίες με περιεχόμενο οριζόμενο από το χρήστη.

Usenet

Ανταλλαγή μηνυμάτων οργανωμένη σε “οικογένειες ηλεκτρονικών συζητήσεων” με εξαιρετική ποικιλία θεμάτων προς συζήτηση και παγκόσμια συμμετοχή.

Talk

Ανταλλαγή μηνυμάτων κειμένου σε πραγματικό χρόνο μεταξύ δύο χρηστών που βρίσκονται σε απομακρυσμένα σημεία του Internet.

IRC (Internet Relay Chat)

Παρόμοιο με το Talk αλλά υποστηρίζει μεγαλύτερο αριθμό χρηστών ταυτόχρονα και οργανώνει τις ομαδικές συνομιλίες ανάλογα με το θέμα τους.

Gopher

Αναζήτηση πληροφορίας μέσω επιλογών (menus) σε παγκόσμιο επίπεδο.

Veronica

Αναζήτηση υπολογιστών του δικτύου που προσφέρουν την υπηρεσία Gopher, σχετικά με το θέμα που ορίζει ο χρήστης.

WAIS (Wide Area Information Service)

Έρευνα μέσα σε επιλεγμένες από το χρήστη βάσεις δεδομένων του Internet σχετικά με λέξεις - κλειδιά που ορίζει ο χρήστης.

WWW (World Wide Web)

Διαδικτυωμένες ηλεκτρονικές σελίδες με πληροφορίες σε γραφικό παραθυρικό περιβάλλον, οι οποίες αλληλοσυνδέονται μέσω λέξεων - κλειδιών. Αυτή η υπηρεσία ενοποιεί μέσα στο ίδιο λογισμικό τις FTP, Archie, Gopher, E-mail, Usenet, κλπ.

MUD (Multiple User Dimension)

Παιχνίδια με σενάριο στα οποία ο χρήστης που συνδέεται παίρνει ένα προσωπικό ρόλο και αλληλεπιδρά στο περιβάλλον του παιχνιδιού με τους υπόλοιπους απομακρυσμένους παίκτες.

1.4 World Wide Web (WWW)

1.4.1 Τι είναι το WWW

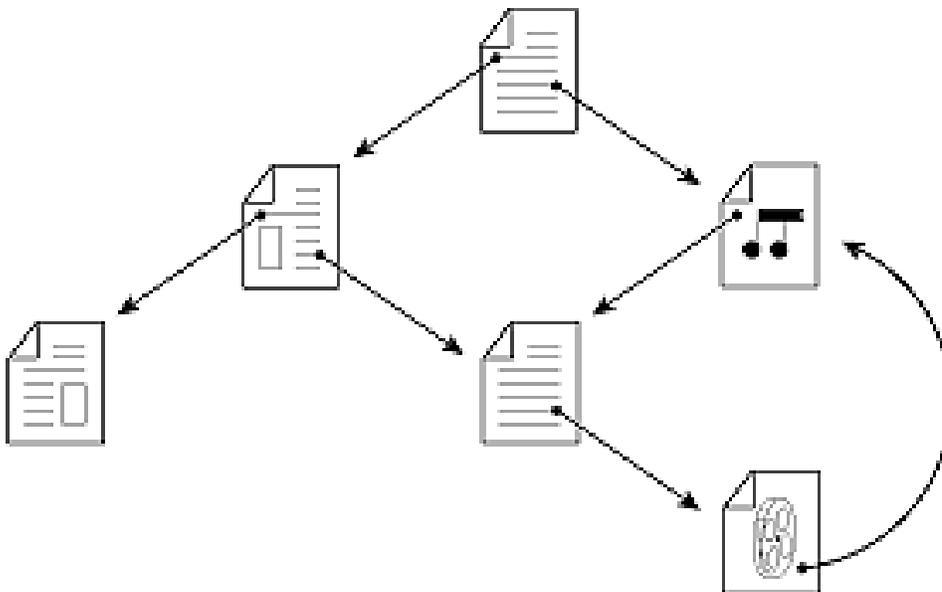
Τα αρχικά **WWW** είναι συντομογραφία της γνωστότερης υπηρεσίας του Internet, του **World Wide Web (Παγκόσμιος Ιστός)** ή απλά **Web**. Τα συναντάμε συχνά σαν πρώτο συνθετικό διευθύνσεων, όπως π.χ. www.upatras.gr (η διεύθυνση του Πανεπιστημίου Πατρών) ή www.microsoft.com (η διεύθυνση της εταιρείας Microsoft), καθώς κάθε πανεπιστήμιο, εταιρεία ή οργανισμός με παρουσία στο Internet προσφέρει συνήθως την υπηρεσία αυτή.

Το WWW γεννήθηκε στο εργαστήριο CERN της Ελβετίας το 1993 και αποτελεί ένα ισχυρό και εύχρηστο μέσο για την προσπέλαση, αναζήτηση και ανεύρεση πληροφοριών στο Internet. Σήμερα, λέγοντας Internet πολλοί εννοούν το WWW, μιας και το WWW είναι πλέον το **επικρατέστερο μέσο** για την πλοήγηση στον ωκεανό πληροφορίας του Internet. Το WWW **διασυνδέει** πληροφορίες που είναι αποθηκευμένες σε χιλιάδες υπολογιστές του Internet, διάσπαρτους σε ολόκληρο τον κόσμο. Οι χρήστες του Διαδικτύου μπορούν να προσπελαίνουν τις διαθέσιμες πληροφορίες χρησιμοποιώντας ένα πρόγραμμα που ονομάζεται **browser** (**πρόγραμμα πλοήγησης**).

Οι πληροφορίες είναι οργανωμένες σε ηλεκτρονικές σελίδες που ονομάζονται **Web σελίδες (Ιστοσελίδες)** και συνδέονται μεταξύ τους με **συνδέσμους**. Μια συλλογή Web σελίδων που βρίσκεται αποθηκευμένη σε ένα συγκεκριμένο σημείο του Internet και διατίθεται δημόσια ονομάζεται **Web site**. Π.χ. το σύνολο των σελίδων που βρίσκονται

αποθηκευμένες στη διεύθυνση www.upatras.gr αποτελεί το Web site του Πανεπιστημίου Πατρών. Η αρχική σελίδα ενός Web site είναι το σημείο εισόδου προς τις υπόλοιπες σελίδες της συλλογής και ονομάζεται **home page** [3].

Μπορούμε να φανταστούμε το WWW σαν μια τεράστια βιβλιοθήκη: τα **Web sites** - κομβικά σημεία του Web - μπορούν να παρομοιαστούν με βιβλία, καθένα από τα οποία αποτελείται από ένα σύνολο σελίδων. Η αρχική σελίδα του Web site μπορεί να παρομοιαστεί με το εξώφυλλο ή τον πίνακα περιεχομένων ενός βιβλίου. Οι σελίδες και οι σύνδεσμοι που τις συνδέουν σχηματίζουν έναν **Ιστό** (Web) πληροφοριών. Μέσω των συνδέσμων, ο χρήστης έχει τη δυνατότητα να μεταπηδά από μια σελίδα σε άλλες:



Εικόνα 1.7: Το WWW διασυνδέει ηλεκτρονικές σελίδες

Βασικό χαρακτηριστικό του WWW είναι η **παγκοσμιότητα** του: οι σελίδες που διασυνδέει **μπορεί να βρίσκονται οπουδήποτε στον κόσμο**. Σαν τελικοί χρήστες όμως, τις προσπελάζουμε όλες με **ομοιόμορφο τρόπο** και έχουμε **ίση πρόσβαση** προς αυτές, χωρίς πρόσθετα έξοδα μεγάλων αποστάσεων ή περιορισμούς.

1.4.2 Υπερκείμενο και Υπερμέσα

Το WWW βασίζεται στην ιδέα του **υπερκειμένου (hypertext)** ή για την ακρίβεια των **υπερμέσων (hypermedia)**.

Το υπερκείμενο είναι μια μορφή ηλεκτρονικού κειμένου, κάποια τμήματα (λέξεις ή φράσεις) του οποίου, που συνήθως εμφανίζονται υπογραμμισμένα, **συνδέονται** με άλλα κείμενα. Αν λοιπόν επιλέξουμε κάνοντας κλικ με το ποντίκι μας τα τμήματα αυτά, τα οποία ονομάζονται **υπερσύνδεσμοι (hyperlinks)** ή πιο απλά **σύνδεσμοι (links)**, στην οθόνη μας εμφανίζεται το συνδεδεμένο κείμενο. Το κείμενο αυτό με τη σειρά του μπορεί να περιέχει άλλους συνδέσμους προς άλλα κείμενα, κ.ο.κ. Έτσι μπορούμε να ταξιδεύουμε από το ένα κείμενο στο άλλο ακολουθώντας τους συνδέσμους που μας ενδιαφέρουν, χωρίς να είμαστε υποχρεωμένοι να διαβάσουμε τα κείμενα με κάποια προδιαγεγραμμένη σειρά.

Αρχικά, οι σελίδες του **Web** περιείχαν υπερκείμενο, δηλαδή κείμενο και συνδέσμους προς άλλες σελίδες που κι αυτές περιείχαν υπερκείμενο. Σιγά - σιγά το υπερκείμενο εμπλουτίστηκε με την ενσωμάτωση **πολυμέσων (multimedia)** απ' όπου προέκυψε ο συνδυασμός των δύο: τα **υπερμέσα (hypermedia)**. Έτσι σήμερα, οι σελίδες του Web είναι πολύ ελκυστικότερες μιας και μπορεί να περιλαμβάνουν: **γραφικά, εικόνες, κινηματογραφικές ταινίες, ήχους, τρισδιάστατους κόσμους** και σχεδόν οποιαδήποτε άλλη μορφή ψηφιακής πληροφορίας μπορούμε να φανταστούμε.

Η **γλώσσα συγγραφής** που χρησιμοποιείται για την κατασκευή των σελίδων είναι η **HTML (Hypertext Markup Language)**.

1.4.3 Λειτουργία του WWW

Το WWW είναι ακόμη ένα παράδειγμα του μοντέλου πελάτη/εξυπηρετητή, στο οποίο δικτυωμένοι υπολογιστές μοιράζονται τη δουλειά που απαιτεί μια διαδικασία. Στο WWW, η επικοινωνία μεταξύ του πελάτη και του εξυπηρετητή γίνεται σύμφωνα με το πρωτόκολλο **HTTP (Hyper Text Transfer Protocol)**.

Εμείς εκτελούμε στον υπολογιστή μας ένα πρόγραμμα **πελάτη**, πιθανότατα το Netscape Navigator ή το Internet Explorer. Όπως έχουμε πει, οι πελάτες για την υπηρεσία WWW ονομάζονται **αναζητητές (browsers)**. Μέσω του browser συνδεόμαστε με έναν απομακρυσμένο υπολογιστή που περιέχει τη σελίδα που θέλουμε να δούμε και στον οποίον εκτελείται ένα άλλο πρόγραμμα που ονομάζεται **εξυπηρετητής Web (Web server)**. Ο Web server είναι υπεύθυνος για τη διαβίβαση της σελίδας και ο browser για την παρουσίασή της στην οθόνη του υπολογιστή μας.

Ο browser υποβάλλει την αίτησή του στον Web server και περιμένει μέχρι να έρθει η απάντηση, οπότε παραλαμβάνει τη σελίδα που ζητήθηκε, που πλέον "φορτώνεται" στη μνήμη του τοπικού μας μηχανήματος, και την εμφανίζει στην οθόνη μας. Κατόπιν η σύνδεση κλείνει. Μόλις ζητήσουμε μια άλλη σελίδα, π.χ. κάνοντας κλικ πάνω σε έναν σύνδεσμο, η ίδια διαδικασία αρχίζει ξανά. Αυτό επαναλαμβάνεται πολλές φορές, σε αντίθεση π.χ. με το FTP που διατηρεί ανοικτή γραμμή καθ' όλη τη διάρκεια της σύνδεσης. Αυτός ακριβώς ο τρόπος επικοινωνίας εξηγεί και τα πολλαπλά μηνύματα που πιθανόν να βλέπουμε στην τελευταία γραμμή της οθόνης του browser όταν προσπαθεί να εμφανίσει μία Web σελίδα ("Contacting Host...", κλπ.)

Η κατανομή της εργασίας μεταξύ του browser και του Web server επιταχύνει τη διαδικασία με πολλούς τρόπους, αλλά σημαίνει επίσης ότι οι δημιουργοί Web σελίδων δεν μπορούν να ελέγξουν την τελική τους εμφάνιση, η οποία εξαρτάται από το πώς είναι διαμορφωμένος ο browser. Για παράδειγμα, ο δικός μας browser μπορεί να χρησιμοποιεί τη γραμματοσειρά Times-Roman για την παρουσίαση του κειμένου, ενώ ο browser ενός άλλου χρήστη μπορεί να χρησιμοποιεί τη γραμματοσειρά Helvetica.

Καθώς "σερφάρουμε" στο Internet χρησιμοποιώντας τον browser μας, προβάλλουμε στην οθόνη του υπολογιστή μας σελίδες που μπορεί να προέρχονται από πολλούς διαφορετικούς Web servers. Από την ίδια Web σελίδα μπορεί να ξεκινούν σύνδεσμοι προς άλλες σελίδες που βρίσκονται διασκορπισμένες σε διάφορους Web servers ανά τον κόσμο..

Πώς όμως, όταν κάνουμε κλικ πάνω σε έναν σύνδεσμο, μπορεί ο browser να γνωρίζει με ποιόν Web server πρέπει να συνδεθεί και ποια από όλες τις σελίδες του να ζητήσει; Κι ακόμη, πώς μπορούμε εμείς να οδηγήσουμε άμεσα τον browser μας να συνδεθεί με έναν συγκεκριμένο Web server και να ζητήσει μια συγκεκριμένη σελίδα;

Η απάντηση στις ερωτήσεις αυτές σχετίζεται με κάτι που ονομάζεται URL και εξηγείται στην παράγραφο 2.4.5.

1.4.4 Το WWW και οι άλλες υπηρεσίες του Internet

Ένα βασικό χαρακτηριστικό του WWW που συντείνει στην καθιέρωση του σαν επικρατέστερου μέσου πλοήγησης στο Διαδίκτυο, είναι ότι παρέχει ενιαία πρόσβαση και σε άλλες υπηρεσίες του Internet.

Οι browsers προσφέρουν στο χρήστη ένα πολύ εύχρηστο και φιλικό περιβάλλον μέσα από το οποίο μπορεί να εκτελεί διάφορες εργασίες στο Internet, χωρίς να χρειάζεται να καταφεύγει σε ξεχωριστά προγράμματα πελάτες για την κάθε υπηρεσία που επιθυμεί να χρησιμοποιήσει. Έτσι, ο browser αποτελεί ένα ισχυρό κέλυφος εργασίας, που ενοποιεί κάτω από την ίδια στέγη σχεδόν όλες τις υπηρεσίες: E-mail, Usenet, anonymous FTP, gopher, telnet.

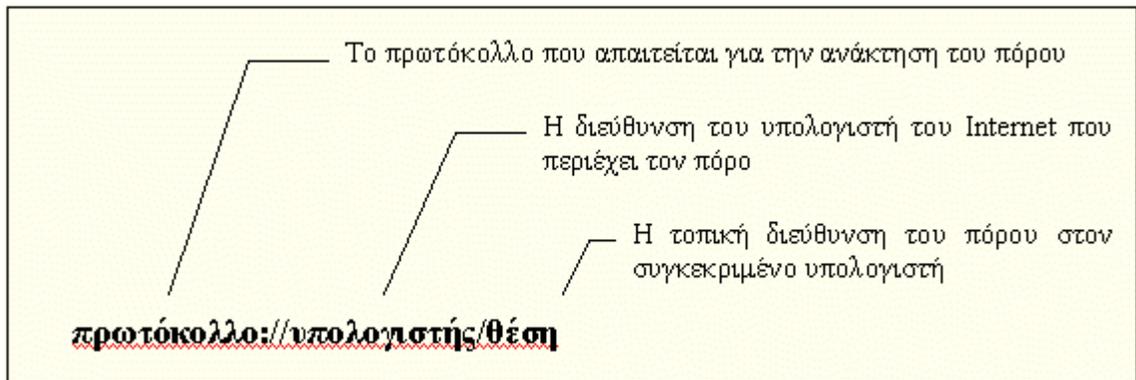
Η ομοιόμορφη αυτή πρόσβαση προς άλλες υπηρεσίες του Internet επιτυγχάνεται μέσω των URL.

1.4.5 Η έννοια του Uniform Resource Locator (URL)

Μέχρι στιγμής έχουμε μιλήσει για διευθύνσεις υπολογιστών στο Internet καθώς και για διευθύνσεις ηλεκτρονικού ταχυδρομείου (E-mail διευθύνσεις) των χρηστών του Internet. Στην παράγραφο αυτή, θα δούμε μια άλλη μορφή διευθύνσεων, τα **URL (Uniform Resource Locator** ή Ενιαίος Εντοπιστής Πόρων). Τα URL κατά κάποιον τρόπο αποτελούν ένα "υπερσύνολο" των διευθύνσεων υπολογιστών και των E-mail διευθύνσεων, μιας και διαθέτουν πρόσθετα χαρακτηριστικά που τα καθιστούν πολύ ισχυρότερα.

Τα URL παρέχουν έναν ενιαίο και σύντομο τρόπο να προσδιορίζεται μοναδικά ένας πόρος στο Internet. Λέγοντας πόρο, εννοούμε π.χ. μια σελίδα του Web, ένα αρχείο που διατίθεται για anonymous FTP, αλλά και μια E-mail διεύθυνση ενός χρήστη, ένα newsgroup του Usenet ή έναν υπολογιστή του Internet. Το URL ενός πόρου προσδιορίζει τη θέση του πόρου στο Διαδίκτυο και το είδος του. Ο browser χρησιμοποιεί το URL για να εντοπίσει το συγκεκριμένο πόρο σ' ολόκληρο το Internet.

Υπάρχουν διάφοροι τύποι URL ανάλογα με το είδος του πόρου που προσδιορίζουν. Στη γενική του μορφή, ένα URL αποτελείται από τρία τμήματα:



Εικόνα 1.8: Διαβάζοντας ένα URL

Διαβάζοντας μια διεύθυνση υπολογιστή από τα δεξιά προς τα αριστερά, μπορούμε να αποκομίσουμε κάποιες πληροφορίες για τον υπολογιστή. Π.χ.:

edu προσδιορίζει εκπαιδευτικό ίδρυμα

com "" εμπορική επιχείρηση

org "" μη-κερδοσκοπική οργάνωση

gov "" κυβερνητική οργάνωση

gr "" υπολογιστή που βρίσκεται στην Ελλάδα

www "" έναν Web server

Ο πιο δημοφιλής τύπος URL είναι σίγουρα ο ακόλουθος:

http://υπολογιστής/θέση

Π.χ.: <http://www.upatras.gr/dir/file.html>

Ας αναλύσουμε το URL αυτό, κομμάτι - κομμάτι:

Η τιμήμα "**http**" δηλώνει ότι πρόκειται για μια σελίδα Web (δηλαδή πόρο που πρέπει να ανακτηθεί μέσω του πρωτοκόλλου **HTTP**).

Το τμήμα "www.upatras.gr" είναι η διεύθυνση του Web server στον οποίο βρίσκεται αποθηκευμένη η σελίδα.

Τέλος, το τμήμα "**dir/file.html**" δηλώνει τη θέση του αρχείου που περιέχει τον κώδικα της σελίδας στο τοπικό σύστημα αρχείων του Web server. Δηλαδή τον κατάλογο στον οποίον βρίσκεται το αρχείο ("**dir**") και το όνομα του αρχείου ("**file.html**"). Σημειώνουμε ότι ένα αρχείο με επέκταση "**.html**" (ή "**.htm**") είναι ένα αρχείο που περιέχει κώδικα HTML, δηλαδή μια σελίδα Web.

Πολλές φορές, συναντάμε URL στα οποία έχει παραληφθεί το τελευταίο τμήμα, δηλαδή υπάρχει μόνον το τμήμα "http" και η διεύθυνση του Web server ακολουθούμενη από μία κάθετο.

Π.χ.: <http://www.upatras.gr/>

Πληκτρολογώντας το URL αυτό στη γραμμή διεύθυνσης του browser, ανακτάμε την **αρχική σελίδα (home page)** του συγκεκριμένου Web server - στο παράδειγμά μας, την αρχική σελίδα του Πανεπιστημίου Πατρών. Παραλείποντας την τελική κάθετο, μερικές φορές μπορεί να προκαλέσει λάθος. Αν βέβαια θέλουμε να ανακτήσουμε κάποια άλλη σελίδα που δεν είναι η αρχική, θα πρέπει να δώσουμε και τη θέση της.

Σημειώνουμε ότι στους σύγχρονους browsers μπορεί κανείς να παραλείψει τελείως το τμήμα "http://". Π.χ. θα μπορούσαμε απλά να δώσουμε:

www.upatras.gr/internet.html

Το URL μιας σελίδας είναι λοιπόν το μόνο στοιχείο που χρειάζεται ο browser για να την εντοπίσει μέσα στο Διαδίκτυο και να την εμφανίσει στην οθόνη μας. Όταν λοιπόν "σερφάρουμε" στο Internet, είτε δίνουμε άμεσα το URL μιας συγκεκριμένης σελίδας που θέλουμε να επισκεφτούμε πληκτρολογώντας το στη γραμμή διεύθυνσης, είτε έμμεσα, κάνοντας κλικ πάνω σε έναν σύνδεσμο, οπότε ο browser χρησιμοποιεί τις πληροφορίες που περιέχονται στο URL του συνδέσμου για να ανακτήσει την αντίστοιχη σελίδα.

Τέλος, υπάρχουν και άλλοι τύποι URL που προσδιορίζουν διαφορετικά είδη πόρων. Χρησιμοποιώντας τα URL αυτά μέσα από το περιβάλλον του browser, αποκτάμε πρόσβαση σε άλλες υπηρεσίες του Internet. Το πρώτο τμήμα τους - που όπως είδαμε

προσδιορίζει το πρωτόκολλο ανάκτησης του πόρου, δηλαδή το είδος της υπηρεσίας - δεν είναι "http", αλλά "**ftp**" (υπηρεσία anonymous FTP), "**telnet**" (υπηρεσία Telnet), "**news**" (υπηρεσία Usenet) και "**mailto**" (υπηρεσία E-mail).

1.4.6 Προγράμματα για το WWW (Browsers)

Η ανάπτυξη του WWW και η εμφάνιση γραφικών προγραμμάτων πλοήγησης έκαναν την εξερεύνηση του Internet προσιτή στους απλούς χρήστες. Ο πρώτος γραφικός browser που κυκλοφόρησε ήταν ο **Mosaic** της NCSA. Σήμερα, πολύ διαδεδομένοι browsers είναι ο **Netscape Navigator** της Netscape Communications και ο **Internet Explorer** της Microsoft. Και τα δύο προγράμματα προσφέρουν τις ίδιες βασικές λειτουργίες και εξελίσσονται διαρκώς, οπότε προκύπτουν νέες εκδόσεις τους.

ΚΕΦΑΛΑΙΟ 2^ο

ΕΙΣΑΓΩΓΗ ΣΤΟ ΗΛΕΚΤΡΟΝΙΚΟ ΕΜΠΟΡΙΟ

2.1 Η ανάγκη του Ηλεκτρονικού Εμπορίου

Μία από τις πιο βασικές φιλοδοξίες κάθε επιχείρησης είναι η δυνατότητα πρόσβασης σε νέες ή και ήδη διαμορφωμένες αγορές. Έτσι κάθε επιχειρηματίας που ζει στην Ελλάδα θα ήθελε να έχει πρόσβαση στο καταναλωτικό κοινό της Σουηδίας της Γερμανίας των Ηνωμένων Πολιτειών. Επίσης θα ήθελε σίγουρα ένας πελάτης που έχει δυσκολία πρόσβασης στο σημείο πώλησης να μπορούσε να του αποστείλει μια παραγγελία με ευκολότερο και γρηγορότερο τρόπο από το να προσπαθήσει να φτάσει στο σημείο πώλησης.

Επίσης από την πλευρά των πελατών, κάποιος καταναλωτής θα κέρδιζε εμφανώς αν είχε την δυνατότητα να ελέγξει τις τιμές και να αγοράσει ένα προϊόν και από άλλα σημεία εκτός από αυτά που του είναι χωροταξικά διαθέσιμα. Π.χ. αν ένα βιβλίο κόστιζε σε ένα κατάστημα στην Πάτρα 17€ και σε ένα άλλο κατάστημα στην Αθήνα 20€ ο Αθηναίος πολίτης θα προτιμούσε να έχει την δυνατότητα ενημέρωσης και παραγγελίας από το κατάστημα στην Πάτρα.

Για τους παραπάνω λόγους αλλά και για πολλούς ακόμα που αφορούν την ανάπτυξη, οργάνωση, διαχείριση και προώθηση των προϊόντων, τα τελευταία χρόνια έχουν αναπτυχθεί τεχνολογίες στην πληροφορική κατάλληλες για την εξυπηρέτηση τέτοιων αναγκών.

Η ταχύτατη πρόοδος στις επικοινωνίες, στα δίκτυα ηλεκτρονικών υπηρεσιών, στα πολυμέσα και τις εφαρμογές client - server δημιούργησαν νέες επιχειρηματικές ευκαιρίες και οδήγησαν σε νέους και αποτελεσματικούς τρόπους για τη διάδοση πληροφοριών, την προώθηση προϊόντων και υπηρεσιών και πολύ πρόσφατα την εκτέλεση χρηματικών ηλεκτρονικών συναλλαγών.

Το κλειδί για όλα αυτά ήταν το παγκόσμιο δίκτυο του Internet που εξαπλώνεται μέρα με την μέρα και παρακινεί ένα διαρκώς αυξανόμενο αριθμό επιχειρήσεων και καταναλωτών παγκόσμια να συμμετάσχουν στο **Ηλεκτρονικό Εμπόριο**.

2.2 Ορισμός Ηλεκτρονικού Εμπορίου

Για το Ηλεκτρονικό Εμπόριο δεν υπάρχει μέχρι τώρα ένας σαφής ορισμός. Ένας πιθανός ορισμός θα μπορούσε να είναι: "οποιαδήποτε μορφή εμπορικών συναλλαγών στις οποίες τα μέλη ενεργούν με ηλεκτρονικές παρά με φυσικές συναλλαγές ή άμεση φυσική επαφή".

Όμως, ενώ ένας τέτοιος ορισμός είναι ακριβής, δύσκολα θα μπορούσε να συλλάβει το πνεύμα του ηλεκτρονικού εμπορίου που στην πραγματικότητα γίνεται πιο καλά κατανοητό ως μία από εκείνες τις σπάνιες περιπτώσεις στις οποίες οι μεταβαλλόμενες ανάγκες και οι νέες τεχνολογίες συναντιούνται για να προκαλέσουν μια επανάσταση στον τρόπο με τον οποίο γίνονται οι επιχειρήσεις.

Έτσι ένας πιο τεχνικός ορισμός του ηλεκτρονικού εμπορίου είναι:

Το Ηλεκτρονικό Εμπόριο είναι η ανταλλαγή επιχειρηματικών πληροφοριών με τη χρησιμοποίηση ηλεκτρονικών τυποποιήσεων που περιλαμβάνουν την Ηλεκτρονική Ανταλλαγή Δεδομένων (EDI), το Ηλεκτρονικό Ταχυδρομείο (e-Mail), τους Ηλεκτρονικούς Πίνακες Ανακοινώσεων (EBBs) και την Ηλεκτρονική Μεταφορά Κεφαλαίων (EFT). Οι τεχνολογίες Ηλεκτρονικού Εμπορίου είναι σχεδιασμένες να αντικαταστήσουν τις παραδοσιακές εργασίες ροής που βασίζονται στα έγγραφα με πιο γρήγορες, πιο επαρκείς και πιο αξιόπιστες επικοινωνίες ανάμεσα σε υπολογιστές και συστήματα πληροφορικής.

2.3 Η σημερινή πραγματικότητα

Η σύγχρονη επιχειρηματική δραστηριότητα χαρακτηρίζεται από διαρκώς αυξανόμενες ικανότητες παροχής, από διαρκώς αυξανόμενο παγκόσμιο ανταγωνισμό, και διαρκώς αυξανόμενες απαιτήσεις από τους πελάτες.

Ανταποκρινόμενες σ' αυτό, οι επιχειρήσεις σε όλο τον κόσμο μεταβάλλουν την οργάνωσή και τον τρόπο λειτουργία τους. Έτσι με χρήση των νέων τεχνολογιών στις επικοινωνίες και την ραγδαία ανάπτυξη της πληροφορικής εγκαταλείπουν τις παλιές ιεραρχικές δομές και καταλύουν τα σύνορα ανάμεσα στα διάφορα τμήματα της επιχείρησης.

Το ηλεκτρονικό εμπόριο είναι ένα μέσο που επιτρέπει και υποστηρίζει αλλαγές σε παγκόσμια κλίμακα. Επιτρέπει στις εταιρίες να είναι πιο ευέλικτες και πιο αποτελεσματικές στην εσωτερική τους λειτουργία, να εργάζονται πιο στενά με τους προμηθευτές τους, και να ανταποκρίνονται καλύτερα στις προσδοκίες και τις ανάγκες των πελατών τους. Επιτρέπει στις εταιρίες να επιλέγουν τους πιο καλούς προμηθευτές, ανεξάρτητα από τη γεωγραφική τους θέση και να κάνουν πωλήσεις στην παγκόσμια αγορά.

Μια ειδική περίπτωση του ηλεκτρονικού εμπορίου είναι οι ηλεκτρονικές πωλήσεις, κατά τις οποίες ένας προμηθευτής προσφέρει αγαθά και υπηρεσίες σε έναν πελάτη με πληρωμή. Μια ειδική περίπτωση ηλεκτρονικής πώλησης είναι το ηλεκτρονικό λιανικό εμπόριο, όπου πελάτης συνήθως είναι ένας συνηθισμένος καταναλωτής και όχι μια εταιρία. Όμως, ενώ αυτές οι ειδικές περιπτώσεις θεωρούνται σημαντικές από οικονομική άποψη, δεν είναι παρά απλά παραδείγματα πιο γενικών περιπτώσεων οποιασδήποτε μορφής επιχειρηματικής λειτουργίας ή συναλλαγής που γίνεται με ηλεκτρονικά μέσα. Άλλα εξίσου έγκυρα παραδείγματα περιλαμβάνουν εσωτερικές συναλλαγές μέσα σε μια μόνο εταιρία, ή την προσφορά πληροφοριών σε μια εξωτερική οργάνωση χωρίς καμιά πληρωμή.

Το ηλεκτρονικό εμπόριο είναι μια τεχνολογία αλλαγής. Οι εταιρίες που επιλέγουν να το θεωρούν σαν ένα ακόμη "πρόσθετο" στους υπάρχοντες τρόπους με τους οποίους κάνουν τις συναλλαγές τους θα αποκομίσουν ελάχιστα. Τα κυριότερα οφέλη θα συσσωρευτούν σε εκείνες τις επιχειρήσεις που είναι πρόθυμες να αλλάξουν την οργάνωσή τους και τις επιχειρηματικές τους διαδικασίες ώστε να εκμεταλλευτούν πλήρως τις ευκαιρίες που προσφέρονται από το ηλεκτρονικό εμπόριο.

Παρόλα αυτά όμως οι περισσότερες επιχειρήσεις τουλάχιστον στον ελληνικό χώρο δεν έχουν προχωρήσει σε τέτοια βήματα. Το πρόβλημα δε γίνεται έντονο κυρίως στις μικρομεσαίες επιχειρήσεις οι οποίες αποτελούν και το θεμέλιο της οικονομικής

δραστηριότητας στην Ελλάδα και την Ευρώπη γενικότερα. Αντιθέτως στις Ηνωμένες Πολιτείες οι περισσότερες επιχειρήσεις τουλάχιστον των μεγάλων αστικών περιοχών έχουν αναπτύξει έστω και μια απλή ηλεκτρονική πύλη στο διαδύκτιο (μια απλή σελίδα με δυνατότητα συμπλήρωσης φόρμας παραγγελίας).

Η συνεχή ανάπτυξη όμως ανάπτυξη του διαδικτύου δημιουργεί σαφώς έντονο ανταγωνιστικό προβάδισμα στις εταιρίες που έχουν προχωρήσει και σε τέτοιες εφαρμογές γιατί οι χρήστες του διαδικτύου μπορούν εύκολα και γρήγορα να πληροφορηθούν και να παραγγείλουν από αυτές.

2.4 Κατηγορίες του Ηλεκτρονικού Εμπορίου

Το ηλεκτρονικό εμπόριο μπορεί να υποδιαιρεθεί σε τέσσερις κατηγορίες

1. Από επιχείρηση σε επιχείρηση
2. Από επιχείρηση σε καταναλωτή
3. Από επιχείρηση προς τη διοίκηση
4. Από καταναλωτή προς τη διοίκηση

Ένα παράδειγμα της κατηγορίας *από επιχείρηση σε επιχείρηση* θα μπορούσε να είναι μια εταιρία που χρησιμοποιεί ένα δίκτυο για τις παραγγελίες στους προμηθευτές της, τα τιμολόγια και τις πληρωμές. Αυτή η κατηγορία ηλεκτρονικού εμπορίου έχει εδραιωθεί πολύ καλά για αρκετά χρόνια, χρησιμοποιώντας κυρίως την Ηλεκτρονική Ανταλλαγή Δεδομένων (EDI - Electronic Data Interchange) μέσα από ιδιωτικά ή προστιθέμενης αξίας δίκτυα.

Η κατηγορία *ανάμεσα σε επιχείρηση και καταναλωτή* αναφέρεται κυρίως στο ηλεκτρονικό λιανικό εμπόριο. Αυτή η κατηγορία αναπτύχθηκε υπερβολικά με την έλευση του World Wide Web. Υπάρχουν τώρα "εμπορικά κέντρα" παντού μέσα στο Internet που προσφέρουν όλων των ειδών τα καταναλωτικά αγαθά, από κέικ και κρασιά μέχρι υπολογιστές και αυτοκίνητα. Είναι δε και η πιο διαδεδομένη και έντονα αναπτυσσόμενη γιατί μέσω αυτού του τρόπου οι εταιρείες μπορούν να πληροφορούν και να δέχονται παραγγελίες από τον άμεσα ενδιαφερόμενο και από οποιοδήποτε σημείο του κόσμου. Αυτό σημαίνει ότι οποιαδήποτε επιχείρηση έχει μια τέτοια πύλη εμπορίου ξαφνικά γίνεται διεθνής τουλάχιστον στο επίπεδο πωλήσεων.

Η κατηγορία ανάμεσα στην *επιχείρηση* και στη *διοίκηση* καλύπτει όλες τις συναλλαγές που γίνονται ανάμεσα σε εταιρίες και κυβερνητικούς οργανισμούς. Για παράδειγμα, στις ΗΠΑ, οι λεπτομέρειες για τις μελλοντικές κρατικές προμήθειες δημοσιεύονται στο Internet και οι εταιρίες μπορούν να ανταποκριθούν ηλεκτρονικά. Τώρα αυτή η κατηγορία βρίσκεται σε νηπιακό στάδιο, αλλά θα μπορούσε να αναπτυχθεί πολύ γοργά καθώς οι κυβερνήσεις χρησιμοποιούν τους δικούς τους μηχανισμούς για να προωθήσουν τη γνώση αναφορικά με το ηλεκτρονικό εμπόριο και την ανάπτυξή του. Πέρα από τις δημόσιες προμήθειες, οι διοικήσεις μπορούν επίσης να προσφέρουν την επιλογή για την εκτέλεση ηλεκτρονικών συναλλαγών όπως είναι η επιστροφή του ΦΠΑ και η πληρωμή των εμπορικών φόρων.

Η κατηγορία που αναφέρεται ανάμεσα στον *καταναλωτή* και στη *διοίκηση* δεν έχει ακόμη αναπτυχθεί. Όμως ως επακόλουθο της ανάπτυξης των κατηγοριών ανάμεσα σε *επιχείρηση* και *καταναλωτή* όσο και *επιχείρηση* και *διοίκηση*, οι κυβερνήσεις μπορούν στο μέλλον να επεκτείνουν την ηλεκτρονική αλληλεπίδραση σε τέτοιες περιοχές όπως είναι τα επιδόματα κοινωνικής προνοίας και οι αυτόματες επιστροφές φόρων.

2.5 Πλεονεκτήματα Ηλεκτρονικού Εμπορίου

Παγκόσμια παρουσία / παγκόσμια επιλογή

Τα όρια του ηλεκτρονικού εμπορίου δεν καθορίζονται από τα γεωγραφικά και εθνικά σύνορα, αλλά περισσότερο από την κάλυψη που έχουν τα δίκτυα υπολογιστών. Από τη στιγμή που τα περισσότερα δίκτυα έχουν παγκόσμια κάλυψη, το ηλεκτρονικό εμπόριο επιτρέπει και στους πιο μικρούς ακόμη προμηθευτές να επιτύχουν μια παγκόσμια παρουσία και να ασκήσουν επιχειρηματική δραστηριότητα σε όλο τον κόσμο.

Το αντίστοιχο όφελος για τον πελάτη είναι η παγκόσμια επιλογή - ένας πελάτης μπορεί να επιλέξει από όλους τους δυνητικούς προμηθευτές το προϊόν ή την υπηρεσία που ζητά, ανεξάρτητα από το πού βρίσκεται γεωγραφικά.

Βελτιωμένη ανταγωνιστικότητα / ποιότητα υπηρεσιών

Το ηλεκτρονικό εμπόριο επιτρέπει στους προμηθευτές να βελτιώσουν την ανταγωνιστικότητά τους με το "να έρθουν πιο κοντά στον πελάτη". Ως απλό παράδειγμα

πολλές εταιρίες χρησιμοποιούν την τεχνολογία του ηλεκτρονικού εμπορίου για να προσφέρουν βελτιωμένα επίπεδα υποστήριξης πριν και μετά την πώληση, με αυξανόμενα επίπεδα πληροφόρησης για τα προϊόντα, οδηγίες για τη χρήση των προϊόντων, και γοργή ανταπόκριση στις ερωτήσεις των πελατών. Το αντίστοιχο όφελος για τον πελάτη είναι βελτιωμένη ποιότητα υπηρεσιών.

Μαζική προσαρμογή σε απαιτήσεις πελατών / Εξατομικευμένα προϊόντα και υπηρεσίες

Με την ηλεκτρονική αλληλεπίδραση, οι προμηθευτές είναι σε θέση να συγκεντρώσουν αναλυτικές πληροφορίες για τις ανάγκες κάθε πελάτη ατομικά και να σχεδιάσουν προϊόντα και υπηρεσίες για εκείνες τις ατομικές ανάγκες. Αυτό καταλήγει σε προσαρμοσμένα προϊόντα που συγκρίνονται με εκείνα που προσφέρονται από εξειδικευμένους προμηθευτές, *αλλά σε τιμές μαζικής αγοράς*. Ένα απλό παράδειγμα είναι ένα online περιοδικό που είναι "κομμένο και ραμμένο" στα μέτρα του αναγνώστη με κάθε πρόσβαση να δίνει έμφαση στα άρθρα που τον ενδιαφέρουν και να αποκλείονται εκείνα τα οποία τα έχει ήδη διαβάσει.

Μείωση ή εξάλειψη των δικτύων διανομής / Γοργή ανταπόκριση στις ανάγκες

Το ηλεκτρονικό εμπόριο επιτρέπει συχνά την δραματική μείωση των παραδοσιακών αλυσίδων εφοδιασμού. Υπάρχουν πολλά εδραιωμένα παραδείγματα εταιριών, στις οποίες τα αγαθά αποστέλλονται κατευθείαν από τον κατασκευαστή στον τελικό καταναλωτή, παρακάμπτοντας τα παραδοσιακά δίκτυα χονδρικής και λιανικής πώλησης. (Τυπικά η συνεισφορά του ηλεκτρονικού εμπορίου δεν είναι ότι κάνει εφικτή μια τέτοια άμεση διανομή - από τη στιγμή που θα μπορούσε να γίνει με έντυπους καταλόγους και τηλεφωνικές ή ταχυδρομικές παραγγελίες- αλλά το ότι μια τέτοια άμεση διανομή γίνεται πρακτική όσον αφορά το κόστος και τη χρονική καθυστέρηση.)

Το ακραίο παράδειγμα εμφανίζεται στην περίπτωση των προϊόντων και των υπηρεσιών που μπορούν να παραδοθούν ηλεκτρονικά, όπου το δίκτυο διανομής θα εκλείψει τελείως. Αυτό έχει μαζικές επιπτώσεις στη βιομηχανία της ψυχαγωγίας (φιλμ, βίντεο, μουσική, περιοδικά, εφημερίδες), τη βιομηχανία της πληροφόρησης και της

εκπαίδευσης (στην οποία περιλαμβάνονται όλες οι μορφές εκδόσεων) και τις εταιρίες που σχετίζονται με την ανάπτυξη και τη διανομή λογισμικού για τους υπολογιστές.

Το αντίστοιχο όφελος για τον πελάτη είναι η ικανότητα να αποκτά γρήγορα το ακριβές προϊόν που χρειάζεται, χωρίς να περιορίζεται σε εκείνο που υπάρχει σε στοκ στον τοπικό προμηθευτή.

Ουσιαστική μείωση κόστους / Ουσιαστική μείωση τιμών

Μια από τις κύριες συνεισφορές του ηλεκτρονικού εμπορίου είναι η μείωση του κόστους των συναλλαγών. Ενώ το κόστος μια επιχειρηματικής συναλλαγής που απαιτεί ανθρώπινη αλληλεπίδραση μπορεί να μετρηθεί σε δολάρια, το κόστος για να γίνει μια παρόμοια συναλλαγή ηλεκτρονικά μπορεί να είναι της τάξης λίγων λεπτών του ευρώ ή ακόμη χαμηλότερο. Επομένως, οποιαδήποτε επιχειρηματική διαδικασία που περιλαμβάνει αλληλεπιδράσεις ρουτίνας ανάμεσα σε ανθρώπους προσφέρει τη δυνατότητα για ουσιαστική μείωση κόστους, η οποία με τη σειρά της μπορεί να μεταφραστεί σε ουσιαστική μείωση τιμών για τους πελάτες.

Νέες επιχειρηματικές ευκαιρίες / Νέα προϊόντα και υπηρεσίες

Πέρα από τον επαναπροσδιορισμό των αγορών για τα υπάρχοντα προϊόντα και υπηρεσίες, το ηλεκτρονικό εμπόριο προσφέρει επίσης την ευκαιρία για την ανάπτυξη εντελώς νέων προϊόντων και υπηρεσιών. Τα παραδείγματα περιλαμβάνουν δικτυακή υποστήριξη, και υπηρεσίες στήριξης, υπηρεσίες καταλόγων, υπηρεσίες υπηρεσιών (δηλ. εδραίωση αρχικής επαφής ανάμεσα στους δυνητικούς πελάτες και στους δυνητικούς προμηθευτές) και πολλά είδη online υπηρεσιών πληροφόρησης.

Ενώ όλες αυτές οι ευκαιρίες και τα ωφέλη είναι διακριτά, σε κάποιο βαθμό σχετίζονται μεταξύ τους. Για παράδειγμα, οι βελτιώσεις στον τομέα του ανταγωνισμού και της ποιότητας της υπηρεσίας μπορούν εν μέρει να προέλθουν από τη μαζική προσαρμογή στις απαιτήσεις των πελατών, ενώ η μείωση των δικτύων διανομής μπορεί να συνεισφέρει στη μείωση του κόστους και τη μείωση τιμών.

Στον παρακάτω πίνακα φαίνονται συνοπτικά τα πλεονεκτήματα του ηλεκτρονικού εμπορίου για τους καταναλωτές αλλά και τους προμηθευτές.

Πλεονεκτήματα για τους προμηθευτές	Πλεονεκτήματα για τους καταναλωτές
Παγκόσμια παρουσία	Παγκόσμια επιλογή
Βελτιωμένη ανταγωνιστικότητα	Ποιότητα υπηρεσιών
Μαζική προσαρμογή σε απαιτήσεις πελατών	Εξατομικευμένα προϊόντα και υπηρεσίες
Μειωμένα ή εξαφανισμένα δίκτυα διανομής	Γοργή ανταπόκριση στις ανάγκες
Ουσιαστική μείωση κόστους	Ουσιαστική μείωση τιμών
Νέες επιχειρηματικές ευκαιρίες	Νέα προϊόντα & και υπηρεσίες

2.6 Μειονεκτήματα Ηλεκτρονικού Εμπορίου

Το πρόβλημα της ασφάλειας

Ασφάλεια στη χρήση προσωπικών δεδομένων

Κατά την διαδικασία μιας παραγγελίας ζητούνται κάποια στοιχεία τους όπως η διεύθυνση τους το email τους ο αριθμός της πιστωτικής τους κάρτας, την ημερομηνία γέννησης κ.α. και ότι άλλο θεωρηθεί απαραίτητο για την συναλλαγή. Όμως το πρόβλημα είναι ότι στο διαδίκτυο πολλοί εξειδικευμένοι χρήστες οι λεγόμενοι hackers μπορούν να κλέψουν και να χρησιμοποιήσουν τα στοιχεία αυτά για δικό τους όφελος. Έτσι συνήθως δημιουργούν παράνομες βάσεις δεδομένων όπου και πουλούν τα στοιχεία αυτά σε άλλες εταιρίες που τα χρησιμοποιούν κυρίως για διαφημιστικούς λόγους, όπως την μαζική αποστολή μηνυμάτων κ.α. χωρίς όμως την άδεια του πελάτη. Πολλές φορές τα ίδια στοιχεία χρησιμοποιούνται και για άκρως παράνομες ενέργειες όπως η πλαστογραφία και η οικονομική κλοπή των ατόμων που έκαναν την παραγγελία.

Προβλήματα ασφάλειας στις οικονομικές συναλλαγές

Τα ποσά που δαπανώνται για την πραγματοποίηση αγορών μέσω του Internet εκτιμάται ότι ανέρχονται σε δισεκατομμύρια δολάρια ετησίως. Ήδη μπορείτε να αγοράσετε οτιδήποτε από το Internet - από ένα βιβλίο ως ένα αυτοκίνητο. Κάποια μέρα οι αγορές μέσω του Internet θα είναι τόσο κοινές όσο αυτές μέσω του ταχυδρομείου ή των επισκέψεων στα καταστήματα λιανικών πωλήσεων.

Ο πιο συνήθης τρόπος πληρωμής μέσα από το Internet είναι μέσω πιστωτικής κάρτας. Υπάρχει όμως ένα σημαντικό πρόβλημα που αφορά στις online αγορές: το Internet δεν είναι ένα ασφαλές δίκτυο!

Καθώς τα πακέτα των πληροφοριών ταξιδεύουν στο Internet, οποιοσδήποτε που βρίσκεται στην πορεία τους θα μπορούσε να τα επεξεργαστεί. Έτσι θα μπορούσε ένας εξειδικευμένος χρήστης να κλέψει τον αριθμό της κάρτας και να πραγματοποιήσει αγορά

Για την επίλυση αυτού του προβλήματος έχουν αναπτυχθεί διάφορες μέθοδοι πληρωμών. Οι περισσότερες χρησιμοποιούν διαδικασίες και πρωτόκολλα που έχουν σχεδιαστεί για να κάνουν τις οικονομικές συναλλαγές στο Internet όσο είναι δυνατόν πιο εμπιστευτικές, χρησιμοποιώντας τεχνολογίες απόκρυψης οι οποίες διασφαλίζουν ότι κανείς δεν μπορεί να κλέψει τον αριθμό της πιστωτικής σας κάρτας. Δεκάδες οργανισμοί, συμπεριλαμβανομένων των Microsoft και Netscape, των μεγαλύτερων εταιριών πιστωτικών καρτών καθώς και των οργάνων που είναι επιφορτισμένα με τα πρότυπα του Web αναπτύσσουν σχήματα που προσφέρουν ασφαλείς κωδικοποιημένες οικονομικές συναλλαγές στο Internet. Δύο είναι τα διαθέσιμα σχήματα ασφαλών συναλλαγών.

- Το πρώτο αποκρύπτει τις προσωπικές οικονομικές πληροφορίες, όπως τον αριθμό της πιστωτικής κάρτας, έτσι ώστε να μπορούν να μεταφερθούν στο Internet χωρίς να επιτρέπεται σε άλλους να διαβάσουν τα δεδομένα.
- Η δεύτερη μέθοδος δημιουργεί ένα σύστημα κυβερνο-δολαρίων, ηλεκτρονικών δηλαδή χρημάτων τα οποία μόνο εξουσιοδοτημένοι έμποροι μπορούν να τα εξαργυρώσουν με πραγματικά χρήματα.

Το πρωτόκολλο Secure Electronic Transaction (SET) έχει υιοθετηθεί από τις VISA, MasterCard, American Express, Microsoft, Netscape κ.ά. Περιγράφει έναν τρόπο με τον οποίο οι χρήστες μπορούν να ψωνίζουν online και να χρεώνουν τις πιστωτικές τους κάρτες. Η υποστήριξη αυτής της μεθόδου από τις σημαντικότερες οικονομικές εταιρίες και εταιρίες Πληροφορικής της δίνει μία πολύ καλή πιθανότητα για την μετατροπή της μελλοντικά σε ένα πρότυπο για την πραγματοποίηση online αγορών.

Εκτός από την μέθοδο των ασφαλών συναλλαγών με τη χρήση πιστωτικών καρτών, ένας αριθμός εταιριών στηρίζεται στη χρήση των κυβερνο-δολαρίων τα οποία επιτρέπουν στους καταναλωτές να αγοράζουν τα αγαθά και τις υπηρεσίες ανώνυμα. Με αυτή τη μέθοδο ο χρήστης χρησιμοποιεί το ψηφιακό ισοδύναμο των χρημάτων για την πραγματοποίηση αγορών και έτσι δεν χρειάζεται να στέλνει προσωπικές πληροφορίες στο Δίκτυο. Με τη χρήση αυτής της μεθόδου ηλεκτρονικών πληρωμών οι καταναλωτές μπορούν να αγοράσουν ηλεκτρονικά κέρματα και να τα χρησιμοποιήσουν για την πραγματοποίηση αγορών. Τα συστήματα πιστωτικών καρτών και ηλεκτρονικού "ρευστού" έχουν τα πλεονεκτήματα και τα μειονεκτήματά τους.

Τα συστήματα απόκρυψης των πιστωτικών καρτών είναι βολικά και δεν απαιτούν από τους πελάτες να αλλάξουν τις αγοραστικές συνήθειές τους. Οι συναλλαγές χρεώνονται στην πιστωτική κάρτα και εμφανίζονται στην κατάσταση που αποστέλλεται στον κάτοχο της ακριβώς όπως και με οποιαδήποτε άλλη αγορά. Τα ηλεκτρονικά χρήματα απαιτούν περισσότερες ενέργειες από την πλευρά του χρήστη αφού πρέπει πρώτα να τα αποκτήσει πριν προχωρήσει στη χρήση τους. Παρ' όλα αυτά εξασφαλίζουν καλύτερα το απόρρητο στις ηλεκτρονικές αγορές.

Το πρόβλημα της ταυτότητας του προμηθευτή – καταναλωτή

Βασικό πρόβλημα στις ηλεκτρονικές αγορές είναι η μη γνώση της ταυτότητας του καταναλωτή ή του προμηθευτή. Έτσι είναι αυξημένη η περίπτωση εξαπάτησης από των προμηθευτή ή από τον πελάτη. Χαρακτηριστικό ήταν η δημιουργία εμπορικού καταστήματος στο Internet με επωνυμία γνωστής πολυεθνικής εταιρίας στην οποία οι πελάτες εξαπατούνταν και έκαναν αγορές με την πιστωτική τους κάρτα χωρίς όμως ποτέ να λάβουν τα προϊόντα τα οποία παράγγελναν. Οι πελάτες εξέφρασαν τα παράπονα τους στην εταιρία που νόμιζαν ότι έκαναν την αγορά χωρίς όμως ποτέ να ανακαλυφθεί ποιοι ήταν πίσω από την συγκεκριμένη απάτη.

Αντίστοιχα πολλά είναι τα παραδείγματα πελατών που κάνουν αγορές χωρίς όμως να δίνουν τα πραγματικά στοιχεία τους. Αυτό συμβαίνει συχνά από άτομα μικρών ηλικιών ή από ανταγωνιστικές άλλες εταιρίες που στόχο έχουν την δημιουργία μηχανογραφικών και γραφειοκρατικών προβλημάτων στις αντίπαλες εταιρείες.

Το πρόβλημα της υπερπληροφορίας

Σύμφωνα με μια έρευνες πάνω στη χρήση του ηλεκτρονικού εμπορίου οι χρήστες του δικτύου συνήθως δε βρίσκουν εύκολα αυτό που ψάχνουν ή βρίσκουν υπερβολικά μεγάλο όγκο από παρόμοια με το αντικείμενο θέματα.

Ο λόγος είναι ότι πολλοί οργανισμοί δίνουν σαν λέξεις κλειδιά για το site τους πολλές παρόμοιες με αυτό λέξεις και όχι ακριβής με αυτό με στόχο και μόνο την επισκεψιμότητα . Το αποτέλεσμα είναι η υπερπληφορόφηση και τελικώς η μη λειτουργικότητα στην αναζήτηση.

Το πρόβλημα αυτό αντιμετωπίζεται με την συνεχή ανάπτυξη των μηχανών αναζήτησης και την όλο και πιο αποτελεσματική χρήση κριτηρίων και φίλτρων αναζήτησης. Αυτό όμως δημιουργεί πρόσθετα προβλήματα μια και αυξάνει το βαθμό πολυπλοκότητας του τρόπου αναζήτησης.

2.7 Στοιχεία ενός Ηλεκτρονικού Καταστήματος

Βασικά στοιχεία

Διεύθυνση (domain name):

Το απαραίτητο στοιχείο που πρέπει να διαθέτει ένα ηλεκτρονικό κατάστημα είναι μια διεύθυνση στο Internet. Η διεύθυνση είναι ένας κωδικός αριθμός όπως ο 192.163.189.100 το λεγόμενο IP του μηχανήματος που φιλοξενεί την σελίδα συνοδευόμενο με το μονοπάτι που οδηγεί στο χώρο (κατάλογο) στον οποίο πραγματικά βρίσκεται η σελίδα. Ο αριθμός αυτό καθώς και το μονοπάτι μπορεί να αντικατασταθεί από ένα όνομα με συνήθη μορφή www.onoma.gr ή www.onoma.com .

Το προτεινόμενο όνομα της διεύθυνσης είναι το ίδιο όνομα της επιχείρησης ή έστω κάτι έντονα παραπλήσιο με αυτήν. Έτσι θα είναι εύκολο σε κάποιον να την επισκεφτεί αν γνωρίζει απλά το όνομα της.

Για να κατοχυρωθεί το όνομα μιας διεύθυνσης και να μην χρησιμοποιηθεί και από άλλους πρέπει να δηλωθεί σε αρμόδια υπηρεσία. Στην Ελλάδα η αντίστοιχη υπηρεσία (HostMaster) είναι το Ίδρυμα Τεχνολογίας και Έρευνας στην Κρήτη. .

Αισθητικά Όμορφο

Ο δικτυακός τόπος ενός καταστήματος το λεγόμενο web-site είναι η βιτρίνα του καταστήματος στο διαδίκτυο. Έτσι όπως και σε ένα μη ηλεκτρονικό κατάστημα η βιτρίνα του είναι το σημείο εκείνο που πρέπει να είναι το καλύτερο από πλευράς αισθητικής. Αν είναι όμορφο αισθητικά και άρτια τεχνικά κατασκευασμένο, θα προκαλέσει την καλή διάθεση του επισκέπτη και το αίσθημα εμπιστοσύνης. Αν ο επισκέπτης-πιθανός πελάτης πεισθεί ότι πίσω από αυτό υπάρχει μια εταιρεία με σοβαρότητα και επαγγελματισμό είναι πιθανότερο να προχωρήσει σε επαφή, συναλλαγή και αγορά από αυτή.

Μέτρα ασφάλειας στις συναλλαγές:

Ο μεγαλύτερος φόβος των χρηστών του Internet που τους εμποδίζει να πραγματοποιήσουν συναλλαγές, είναι τα θέματα ασφαλείας των διαφόρων ηλεκτρονικών καταστημάτων . Από τις προτεραιότητες ενός ηλεκτρονικού καταστήματος, είναι η χρήση των πιο σύγχρονων τεχνικών και προδιαγραφών ασφαλείας (κρυπτογράφηση, αυθεντικοποίηση, σαφείς όροι χρήσης site, πολιτικές προστασίας δεδομένων κτλ) ώστε ο επισκέπτης του, να αισθάνεται ασφάλεια κατά την περιήγηση του στις ιστοσελίδες του και την παραγγελία προϊόντων από αυτό.

Ευκολία καταχώρησης της παραγγελίας και του τρόπου πληρωμής:

Έχει αποδειχθεί με έρευνες και στατιστικές ανάμεσα σε δημοφιλή ηλεκτρονικά καταστήματα και χρήστες τους, ότι όσο πιο απλή και εύκολη είναι η παραγγελία ενός προϊόντος μέσω Internet, τόσο πιο πολλές είναι οι πωλήσεις του καταστήματος και οι ικανοποιημένοι πελάτες του. Έτσι όσο πιο απλή και ολιγόλεπτη είναι η φόρμα παραγγελίας ενός πελάτη τόσο πιο εύκολα παραγγέλνει κανείς το προϊόν. Αντιθέτων αν η

διαδικασία παραγγελίας είναι χρονοβόρα και πολύπλοκη πολλοί είναι εκείνοι οι οποίοι δεν προχωρούν σε παραγγελίες ή ακόμα περισσότερο διακόπτουν την διαδικασία στη μέση.

Όσο αφορά τον τρόπο πληρωμής ένα ηλεκτρονικό κατάστημα δεν πρέπει να είναι προσκολλημένο στη πιστωτική κάρτα αλλά να διαθέτει όλους τους δυνατούς τρόπους πληρωμής όπως τραπεζικούς λογαριασμούς, αντικαταβολή κ.α.

Διαφήμιση και γνωστοποίηση της Διεύθυνσης του site:

Ένα άρτιο ηλεκτρονικό κατάστημα, με καλαίσθητες ιστοσελίδες και σύγχρονες τεχνικές ασφαλείας, είναι καταδικασμένο σε αποτυχία αν οι χρήστες του Internet δε γνωρίζουν γι' αυτό. Η αποτελεσματική διαφημιστική προώθησή του σε sites, μηχανές αναζήτησης και banners θα φέρει τους νέους επισκέπτες-πιθανούς πελάτες. Πέρα από μια απλή καταγραφή του στις μηχανές αναζήτησης, καλό θα ήταν να μπορούσε να εμφανίζεται στις υψηλότερες θέσεις αποτελεσμάτων σε αναζητήσεις με τις λέξεις κλειδιά (keywords) που είναι σχετικά με αυτό και τα προϊόντα που διαθέτει. Πολλές μάλιστα επιχειρήσεις προχωρούν και σε άμεση διαφήμιση του ονόματος του web-site μέσω κλασσικών μέσων όπως ραδιόφωνο, τηλεόραση, εφημερίδες περιοδικά όπου προσπαθούν να γνωστοποιήσουν στους πιθανούς επισκέπτες το όνομα του site.

2.8 Πρότυπα Ηλεκτρονικού Εμπορίου

Το Ηλεκτρονικό Εμπόριο χρησιμοποιεί αρκετά πρότυπα, ειδικά όσον αφορά το πλαίσιο των συναλλαγών ανάμεσα σε επιχειρήσεις.

Η Ηλεκτρονική Ανταλλαγή Δεδομένων (EDI-Electronic Data Interchange)

Καθώς δημιουργήθηκε από την κυβέρνηση στις αρχές του 1970 και τώρα χρησιμοποιείται κατά 95% από τις 1000 εταιρίες του Fortune, η EDI είναι μια κοινή δομή αρχείων που σχεδιάστηκε ώστε να επιτρέψει σε μεγάλους οργανισμούς να μεταδώσουν πληροφορίες μέσα από μεγάλα ιδιωτικά δίκτυα. Η EDI βρίσκει εξίσου ένα ρόλο στους επιχειρηματικούς σταθμούς του Δικτύου.

Ανοιχτή Αγορά στο Internet (OBI - Open Buying on the Internet):

Το πρότυπο OBI, που δημιουργήθηκε από τη Στρογγυλή Τράπεζα Αγορών μέσω του Internet, έχει ως στόχο να διασφαλίσει ότι όλα τα διαφορετικά συστήματα Ηλεκτρονικού Εμπορίου μπορούν να συνομιλούν μεταξύ τους. Η OBI που ανακοινώθηκε από την OBI Consortium τον περασμένο Ιούνιο, υποστηρίζεται από τις μεγαλύτερες τεχνολογικές εταιρίες όπως είναι οι Actra, IntelliSys, Microsoft, Open Market και Oracle.

Το Ανοικτό Πρωτόκολλο Αγορών (OTP - Open Trading Protocol)

Στόχος του πρωτοκόλλου OTP είναι να θέσει τα πρότυπα για μια σειρά από δραστηριότητες που σχετίζονται με τις πληρωμές, στις οποίες περιλαμβάνονται συμφωνίες αγορών, αποδείξεις αγορών, και πληρωμές. Δημιουργήθηκε ως ανταγωνιστικό πρότυπο στο OBI και υποστηρίζεται από ένα όμιλο εταιριών στις οποίες περιλαμβάνονται η AT&T, CyberCash, Hitachi, IBM, Oracle, Sun Microsystems και η British Telecom.

Το Πρότυπο Ανοιχτού Προφίλ (OPS - Open Profiling Standard)

Το πρότυπο OPS που υποστηρίζεται από τη Microsoft και την Firefly, επιτρέπει στους χρήστες να δημιουργήσουν ένα προσωπικό προφίλ προτιμήσεων και ενδιαφερόντων που θα ήθελαν να συμμεριστούν με τους εμπόρους. Η ιδέα πίσω απ' αυτό βρίσκεται να βοηθήσει τους καταναλωτές να προστατεύσουν το ιδιωτικό απόρρητο τους χωρίς να εμποδιστούν στην online συλλογή εμπορικών πληροφοριών.

Περίβλημα Ασφαλούς Θήκης (SSL - Secure Sockets Layer)

Αυτό το πρωτόκολλο σχεδιάστηκε για να δημιουργήσει μια ασφαλή σύνδεση με τον εξυπηρετητή. Το SSL χρησιμοποιεί κλειδί δημόσιας κρυπτογράφησης, μια από τις πιο ισχυρές μεθόδους κρυπτογράφησης που υπάρχουν, με σκοπό να προστατεύσει τα δεδομένα καθώς ταξιδεύουν μέσα στο Internet. Το SSL δημιουργήθηκε από τη Netscape, αλλά τώρα χρησιμοποιείται ευρέως.

Ασφαλείς Ηλεκτρονικές Συναλλαγές (SET - Secure Electronic Transactions)

Το SET κωδικοποιεί τους αριθμούς της πιστωτικής κάρτας που αποθηκεύονται στον εξυπηρετητή του έμπορου. Αυτό το πρότυπο που δημιουργήθηκε από τη Visa και τη MasterCard, απολαμβάνει μεγάλης αποδοχής από την τραπεζική κοινότητα. Το πρώτο εμπορικό κύκλωμα που επιτρέπει τις SET δοκιμάζεται ήδη στην Ασία.

Διαχείριση: αυτός ο συνασπισμός των εταιριών ζητά να δημιουργήσει μια δημόσια εμπιστοσύνη για το Ηλεκτρονικό Εμπόριο τοποθετώντας μια σφραγίδα αποδοχής για ένα είδος Καλής Διαχείρισης στους σταθμούς που δεν παραβιάζουν το ιδιωτικό απόρρητο του καταναλωτή.

2.9 Παραδείγματα Ιστοσελίδων Ηλεκτρονικού Εμπορίου

Όνομα ιστοσελίδας:

<http://www.bookshop.co.uk>

Όνομα εταιρίας: Internet Bookshop

Περιγραφή

Το Internet Bookshop παίζει το ρόλο ενός βιβλιοπωλείου μέσα στο Internet. Υπάρχει μόνο ως σταθμός στο Internet και δεν υπάρχει καμιά φυσική πρόσβαση μέσα από παραδοσιακά καταστήματα πώλησης. Το κατάστημα εξειδικεύεται σε τεχνικά βιβλία και αυτή τη στιγμή προσφέρει περισσότερους από 780.000 τίτλους. Οι πελάτες που επισκέπτονται το IBS μπορούν να ψάξουν χρησιμοποιώντας λέξεις κλειδιά και να βρουν αναλυτικές πληροφορίες για συγκεκριμένους τίτλους, για διάφορα κείμενα, βιβλιογραφικές αναφορές, λίστες περιεχομένων, επισκοπήσεις και προτεινόμενη βιβλιογραφία. Μπορούν να παραγγείλουν και να πληρώσουν για βιβλία, τα οποία μετά παραδίδονται μέσα από εδραιωμένα διεθνή κανάλια διανομής των εκδοτών.



Όνομα ιστοσελίδας

<http://www.amazon.com>

Όνομα εταιρίας: Amazon.com

Περιγραφή

Η ιστοσελίδα amazon.com είναι το μεγαλύτερο και ίσως το πρώτο και πιο γνωστό βιβλιοπωλείο στο internet στο κόσμο. Διαθέτει την μεγαλύτερη συλλογή βιβλίων στο κόσμο όπως και την υπηρεσία να παραγγείλει για τον πελάτη οποιοδήποτε βιβλίο θελήσει κάποιος από οποιοδήποτε εκδοτικό οίκο. Οι τιμές για κάθε βιβλίο σχεδόν πάντα κατώτερες αυτών των κλασικών βιβλιοπωλείων. Τα τελευταία χρόνια δραστηριοποιείται και σε μια σειρά νέων εκπαιδευτικών προϊόντων και προϊόντων πληροφορικής δίνοντας έτσι την δυνατότητα σε κάθε ενδιαφερόμενο να βρει αυτό που ζητά χρησιμοποιώντας απλά τον υπολογιστή του.

Amazon.com - Earth's Biggest Selection - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://www.amazon.com/exec/obidos/tg/browse/-/1507846/ref%3Drd%5Fwww%5Famazon%5Fgr/002-6295019-5839230

Amazon Exclusive!! Order a Segway now! It's only at Amazon

amazon.com

WELCOME POLYD...S STORE BOOKS APPAREL & ACCESSORIES ELECTRONICS TOYS & GAMES MUSIC COMPUTER & VIDEO GAMES SEE MORE STORES

VIEW CART WISH LIST YOUR ACCOUNT HELP

INTERNATIONAL TOP SELLERS TARGET TODAY'S DEALS SELL YOUR STUFF

polydoropoulos, there's free stuff in your Digital Library.

SEARCH: All Products [GO]

BROWSE:

- Featured Stores: Apparel & Accessories, Office Products
- Books, Music, DVD: Books, DVD, Magazine Subscriptions, Music, Video
- Electronics & Office: Electronics, Camera & Photo, Office Products, Software, Computer & Video Games, Computers

Hello, polydoropoulos. We have recommendations for you. (If you're not polydoropoulos, [click here.](#))

Featured Top Seller: Josh Groban, Josh Groban. Amazon.com. There are worse things in life than making your acting debut on the much ballyhooed season finale of *Ally McBeal*, though teen operatic baritone Josh...

Some of the best moments in [comedy](#), from screwball gags to unforgettable one-liners, are now available on DVD.

Gear up for this year's Academy Awards® with previous [Oscar® winners on DVD](#) for as low as \$15.

Top Sellers in Health & Beauty: Health & Beauty. What's new, hot, and happening at our trusted Health & Beauty partner? Visit [drugstore.com](#) Top Sellers. You'll find the hottest makeup and skin care, customer favorites for hair care and shaving, and the best products to keep you happy and healthy all year long. [Read more](#)

More Top Sellers:

- Linksys BEFW11S4 Wireless 4-Port Cable/DSL Router by Linksys
- Harry Potter and the Chamber of Secrets VHS ~ Daniel Radcliffe
- In Health & Beauty

Sharp digital camcorder now \$299.99 after rebate.

NEW FOR YOU: polydoropoulos, check out what's New for You! (If you're not polydoropoulos, [click here.](#))

Your Message Center: You have 5 new messages.

Your Shopping Cart: You have 0 items in your Shopping Cart.

Your New Releases: Friends - The Complete Third Season VHS ~ Lisa Kudrow

More Categories

Done

Start 602Pro LAN SUITE Amazon.com - Earth's ... Интернет 3:37 PM

ΚΕΦΑΛΑΙΟ 3^ο

ΒΑΣΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ ΓΙΑ ΤΟ ΗΛΕΚΤΡΟΝΙΚΟ ΕΜΠΟΡΙΟ

Το ηλεκτρονικό εμπόριο βασίζεται κυρίως στη δομή του World Wide Web. Το WWW έχει σαν βασική τεχνολογία τις τεχνολογίες υπερκειμένου με κύρια την τεχνολογία της γλώσσας υπερκειμένου HTML. Μέσω αυτής της γλώσσας μπορεί κάποιος να κάνει το πρώτο βήμα δημιουργώντας μια απλή ιστοσελίδα με εικόνες, ήχο βίντεο κείμενα κ.λ.π κατάλληλη για να διαφημίσει και να επικοινωνήσει με τον πελάτη του. Η γνώση της HTML θεωρείται η πιο βασική για την ανάπτυξη οποιαδήποτε ιστοσελίδας.

Η χρήση όμως μόνο της γλώσσας υπερκειμένου Html δεν είναι αρκετή για την υλοποίηση μιας επαγγελματικής σελίδας ηλεκτρονικού εμπορίου. Αυτό γιατί η σελίδες Html είναι σελίδες στατικές δηλαδή σελίδες απλής παρουσίασης δεδομένων μη εξαρτώμενες από τις αποφάσεις και κινήσεις του χρήστη.

Η δυναμικότητα της σελίδας εξασφαλίζεται με τεχνολογίες όπως τα script. Τα script είναι προγράμματα και υπορουτίνες γραμμένες σε γλώσσες υποστηριζόμενες από τον Internet Explorer και το Netscape Navigator όπως ή γλώσσες Javascript και Vbscript. Τα script μπορούν να δώσουν έντονη δυναμικότητα σε μια σελίδα κάνοντας τη να φαίνεται σαν ολοκληρωμένη εφαρμογή. Μέσω των script μπορεί ο σχεδιαστής να δώσει ζωή στις φόρμες των σελίδων να ενεργοποιήσει κουμπιά επιλογών να χρησιμοποιήσει πλαίσια κειμένου κ.λ.π.

Επίσης τα Script σε συνδυασμό με την τυπική γλώσσα ερωτημάτων SQL μπορεί να αποτελέσουν την βάση για την δημιουργία ιστοσελίδων εξαρτώμενων από βάσεις δεδομένων. Μπορούν έτσι ο σχεδιαστής να καλέσει μέσω των αντικειμένων ADO βάσεις δεδομένων και η σελίδα συνδέεται και να διαχειρίζεται τα δεδομένα μιας βάσης. Μέσω του τρόπου αυτού ο χρήστης μπορεί να εισάγει τα δεδομένα μιας παραγγελίας στη βάση της εταιρίας ή να λάβει στοιχεία και δεδομένα για τα προϊόντα τα οποία ενδιαφέρεται. Οι

τεχνολογίες αυτές δηλαδή η χρήση script αντικειμένων ADO και χρήση της SQL συνθέτουν τις τεχνολογίες της ASP (Active Server Pages) βασικές για την δημιουργία σελίδων ηλεκτρονικού εμπορίου.

Παρόμοιες και έντονα εφαρμόσιμες τεχνολογίες είναι η εφαρμογή της γλώσσας Perl ή της αντίστοιχης γλώσσας script PHP.

Η Γλώσσα PHP είναι μια γλώσσα τύπου SCRIPT που γράφεται και εκτελείται μόνο από την πλευρά του SERVER, δηλαδή του υπολογιστή που φιλοξενεί την ιστοσελίδα μας και δίνει τη δυνατότητα στους δικτυακούς επισκέπτες να προβάλουν τα περιεχόμενα της στον υπολογιστή τους

Στην εργασία μας θα εφαρμόσουμε τις τεχνολογίες της HTML, SQL, PHP με χρήση του συστήματος βάσης δεδομένων MYSQL.

Φυσικά για να μπορέσει κάποιος να ανταποκριθεί στη δημιουργία ιστοσελίδων οποιασδήποτε χρήσης πρέπει να έχει και βασικές γνώσεις χρήσης και διαχείρισης εικόνας όπως τα προγράμματα Photoshop, Illustator, Paint Shop, Gif Animator κ.α.

ΚΕΦΑΛΑΙΟ 4°

HTML (HyperText Markup Language)

4.1 Πώς κατασκευάζεται μια απλή Web σελίδα

Εισαγωγή:

Μία Web Σελίδα είναι ένα text αρχείο. Για τον λόγο αυτό μπορούμε να χρησιμοποιήσουμε **οποιοδήποτε πρόγραμμα μας δίνει την δυνατότητα να παράγουμε text αρχεία**. Αρχικά σας προτείνουμε να χρησιμοποιήσετε το notepad (σημειωματάριο) των Windows. Αν θέλετε να χρησιμοποιήσετε το Word for Windows μην ξεχάσετε πως το αρχείο πρέπει να σωθεί σε μορφή txt και όχι doc.

Αρκετοί αναγνώστες θα έχουν ακούσει για προγράμματα όπως το Frontpage ή το Composer με τα οποία μπορεί κανείς να κατασκευάσει Web σελίδες. Πρόκειται για πολύ χρήσιμα εργαλεία αλλά με αυτά δεν μαθαίνει κανείς HTML.

Ο HTML κώδικας της πρώτης σελίδας θα έχει την παρακάτω μορφή:

Βήμα 1:

```
<HTML>H  
πρώτη μου σελίδα </HTML>
```

Το αποτέλεσμα θα είναι:

η πρώτη μου σελίδα

```
<HTML>...</HTML>
```

Κάθε Web σελίδα αρχίζει με την οδηγία (tag) <HTML> και τελειώνει με την εντολή </HTML>. Η πρώτη πληροφορεί τον Web client πως το αρχείο είναι γραμμένο σε γλώσσα HTML και η τελευταία πως ο HTML κώδικας τελείωσε.

Βήμα 2:

Αποθηκεύουμε το αρχείο με όποιο όνομα θέλουμε αρκεί η επέκτασή του να είναι htm ή html. Π.χ. μπορούμε να το ονομάσουμε mypage.htm ή mypage.html.

4.2 Ορισμός Παραγράφων

Εισαγωγή:

Όπως είδαμε, αν και γράψαμε τις λέξεις την μια κάτω από την άλλη, ο browser τις ένωσε σε μια ενιαία παράγραφο. **Ο browser θα αγνοήσει οποιαδήποτε μορφοποίηση κάνουμε στην ώρα που γράφουμε το κείμενο.** Γι' αυτό, όλες οι μορφοποιήσεις πρέπει να δηλωθούν ξεχωριστά.

Η εντολή για να ξεκινήσει μια καινούρια παράγραφος είναι <P> (όταν λέμε καινούρια παράγραφος εννοούμε πως ο browser θα αφήσει μια γραμμή κενή μεταξύ του κειμένου που βρίσκεται αριστερά της <P> και εκείνου που βρίσκεται δεξιά της).

Ο κώδικας θα είναι:

```
<HTML><P>Το 1980 ο γνωστός μελλοντολόγος Alvin Toffler οραματίστηκε στο βιβλίο του “Το Τρίτο Κύμα”, έναν κόσμο με μισοάδειες πόλεις.<P>Συνήθως δεν πρέπει να δίνουμε μεγάλη σημασία σε τέτοιες προβλέψεις<P>Στην συγκεκριμένη περίπτωση, η αυξανόμενη δημοτικότητα της τηλεεργασίας, φαίνεται πως θα επιβεβαιώσει σε μεγάλο βαθμό αυτές τις προβλέψεις.</HTML>
```

Θα εμφανιστεί ως:

Το 1980 ο γνωστός μελλοντολόγος Alvin Toffler οραματίστηκε στο βιβλίο του “Το Τρίτο Κύμα”, έναν κόσμο με μισοάδειες πόλεις.

Συνήθως δεν πρέπει να δίνουμε μεγάλη σημασία σε τέτοιες προβλέψεις.

Στην συγκεκριμένη περίπτωση, η αυξανόμενη δημοτικότητα της τηλεεργασίας, φαίνεται πως θα επιβεβαιώσει σε μεγάλο βαθμό αυτές τις προβλέψεις.

4.3 Αλλαγές Γραμμών

Κενές γραμμές

Η
 λειτουργεί όπως το πάτημα ενός Enter σε έναν επεξεργαστή κειμένου. Μπορούμε να αφήσουμε την υπόλοιπη γραμμή κενή και να ξεκινήσει η εμφάνιση του επόμενου κειμένου από μια νέα.

Η
 δεν αφήνει μια κενή γραμμή. Απλώς μεταφέρει το υπόλοιπο κομμάτι της τρέχουσας γραμμής στην από κάτω. Για να αφήσετε κενή γραμμή πρέπει ή να αλλάξετε παράγραφο ή να βάλετε 2 οδηγίες
 (

).

<NOBR>...</NOBR>

Αν θέλουμε να υποχρεώσουμε ένα κείμενο να μείνει ολόκληρο σε μια γραμμή (όσο μεγάλη κι αν χρειαστεί να γίνει αυτή), χρησιμοποιούμε την <NOBR> (NO
 tag).

4.4 Παραπομπές (Links)

Εισαγωγή:

Οι παραπομπές είναι οι ψυχή του Web. Με αυτές μπορούμε να συνδέσουμε 2 σελίδες έτσι ώστε κάνοντας κλικ σε κείμενο (ή εικόνα) της μιας να μεταφερόμαστε στην άλλη. Η γενική σύνταξη μια παραπομπής είναι:

```
<A HREF="URL">το κείμενο της παραπομπής</A>
```

όπου URL είναι η θέση κάποιας Web σελίδας (η παράμετρος HREF προέρχεται από τις λέξεις Hypertext REference).

Περίπτωση 1

Παραπομπές από μια σελίδα σε μια άλλη κάπου στο Internet

Ο κώδικας θα είναι:

Το `Ηλεκτρονικό εμπόριο` αναπτύσσεται ραγδαία.

Αυτό που θα βλέπει ο χρήστης θα είναι:

Το Ηλεκτρονικό εμπόριο αναπτύσσεται ραγδαία.

Με κλικ πάνω στις λέξεις Ηλεκτρονικό εμπόριο ο browser θα καλεί την σελίδα `http://www.e-shop.gr`

Περίπτωση 2

Παραπομπές από ένα σημείο μιας σελίδας σε ένα άλλο

Πολλές φορές χρειάζεται να παραπέμψουμε τον αναγνώστη όχι σε άλλη σελίδα αλλά σε άλλο τμήμα της ίδιας Web σελίδας. Για να το επιτύχουμε αυτό, χρησιμοποιούμε την παράμετρο NAME για να δώσουμε όνομα στο σημείο στο οποίο θέλουμε να οδηγή η παραπομπή.

Όνομα (identifier) μπορούμε να δώσουμε είτε σε μια μεμονωμένη λέξη είτε σε ολόκληρο κείμενο. Για παράδειγμα παίρνουμε τις λέξεις: "Η σχέση Internet και Επιχειρήσεων" και τους δίνουμε το όνομα XXX. Αυτό γίνεται με την οδηγία `Η σχέση Internet και Επιχειρήσεων`

Η παραπομπή στο σημείο της σελίδας που έχει αυτό το όνομα θα είναι:

`το κείμενο της παραπομπής`

Περίπτωση 3

Παραπομπές από μια σελίδα σε ένα σημείο μιας άλλης σελίδας

Πολλές φορές χρειάζεται να παραπέμψουμε τον αναγνώστη όχι στην κορυφή μιας άλλης σελίδας αλλά σε ένα συγκεκριμένο σημείο της.

Όπως και παραπάνω το όνομα (identifier) μπορεί να είναι μια μεμονωμένη λέξη ή ένα ολόκληρο κείμενο. Για παράδειγμα παίρνουμε τις λέξεις: "Η σχέση Internet και Επιχειρήσεων" και τους δίνουμε το όνομα XXX. Αυτό γίνεται με την οδηγία Η σχέση Internet και Επιχειρήσεων

Η παραπομπή στο σημείο της σελίδας που έχει αυτό το όνομα θα είναι:

το κείμενο της παραπομπής

Όπου path είναι η διαδρομή που πρέπει να ακολουθήσει ο browser για να βρει αυτή τη σελίδα. Για παράδειγμα:

το κείμενο της παραπομπής

Περίπτωση 4

Παραπομπές για την αποστολή email (mailto: URL)

Για να δημιουργήσουμε μια παραπομπή η οποία θα ανοίγει στο PC του επισκέπτη της σελίδας μας ένα παράθυρο σύνταξης email με παραλήπτη τη διεύθυνση που επιθυμούμε, θα πρέπει να χρησιμοποιήσουμε το mailto: URL στην ακόλουθη σύνταξη:

Γράψτε μας

Αν ο χρήστης επιλέξει [Γράψτε μας](#) τότε το mail πρόγραμμα που χρησιμοποιεί θα ανοίξει ένα κενό παράθυρο αποστολής email με τη διεύθυνση info@e-shop.gr ως παραλήπτη.

Αν θέλουμε το παράθυρο αποστολής να έχει ήδη γραμμένο και το θέμα (subject) του mail αυτού, η σύνταξη που θα χρησιμοποιήσουμε είναι:

Γράψτε μας

4.5 Τίτλος σελίδας

`<TITLE>...</TITLE>`

Κάθε Web σελίδα πρέπει να έχει τον δικό της τίτλο. Ο τίτλος (που καλό είναι να μην ξεπερνά τις 10 λέξεις) εμφανίζεται στην κορυφή του παραθύρου του browser και πρέπει πάντοτε να υπάρχει αν και δεν είναι τεχνικά απαραίτητος. Κύρια εργασία του είναι η σωστή ταξινόμηση της σελίδας (επηρεάζει την θέση της σελίδας στα εργαλεία αναζήτησης, δίνει τον τίτλο του bookmark κα.).

Ο τίτλος βρίσκεται μέσα στην οδηγία `<HEAD>`. Η γενική σύνταξη των `HEAD` και `TITLE` είναι:

```
<HEAD><TITLE>Το πρώτο μου κείμενο σε HTML</TITLE></HEAD>
```

4.6 Πώς ορίζουμε το χρώμα του φόντου και του κειμένου της σελίδας

Η οδηγία `<BODY>` ρυθμίζει την εμφάνιση του κυρίως σώματος της σελίδας. Η `<BODY>` ανοίγει αμέσως μετά τις `<TITLE>` και `<HEAD>` και κλείνει `</BODY>` προτελευταία αμέσως πριν την `</HTML>`. Το **χρώμα του φόντου** της σελίδας ορίζεται παραμετρικά μέσα στην οδηγία `<BODY>`

Η σύνταξη είναι: `<BODY BGCOLOR="#934CE8">` (Για πράσινο χρώμα) ή `<BODY BGCOLOR=BLUE>` (για μπλε χρώμα).

Αν θέλουμε να ορίσουμε το χρώμα λεκτικά `<BODY BGCOLOR=όνομα χρώματος>` τότε θα χρησιμοποιήσουμε ένα από τα **black, white, green, maroon, olive, navy, purple, gray, red, yellow, blue, teal, lime, aqua, fuchsia, silver**. Για μεγαλύτερη ποικιλία σε χρώματα και αποχρώσεις, χρησιμοποιούμε τον δεκαεξαδικό κωδικό `<BODY BGCOLOR="ο κωδικός">` του κάθε χρώματος (π.χ. το #934CE8 είναι μια απόχρωση του πράσινου).

!!!! Ο δεκαεξαδικός κωδικός, πρέπει να μπαίνει μέσα σε εισαγωγικά. Δεν βάζουμε όμως εισαγωγικά αν το χρώμα δίνεται λεκτικά.

Το **χρώμα του κειμένου** της σελίδας ορίζεται παραμετρικά μέσα στην οδηγία `<BODY>`

Η σύνταξη είναι: <BODY TEXT=ΧΡΩΜΑ LINK=ΧΡΩΜΑ VLINK=ΧΡΩΜΑ ALINK=ΧΡΩΜΑ>

Όπου ΧΡΩΜΑ= Το όνομα ή ο δεκαεξαδικός κωδικός (με εισαγωγικά) του χρώματος που θέλουμε να έχει η αντίστοιχη παράμετρος.

Οι παράμετροι που έχουμε διαθέσιμες για το χρώμα του κειμένου είναι:

TEXT: Καθορίζει το χρώμα του κειμένου

LINK: Καθορίζει το χρώμα του κειμένου που κρύβει μια παραπομπή που δεν έχουμε ακολουθήσει ακόμη (η default παράμετρος είναι μπλε)

VLINK: Καθορίζει το χρώμα του κειμένου που κρύβει μια παραπομπή που έχει ακολουθηθεί στο παρελθόν (η default παράμετρος είναι κόκκινο)

ALINK: Καθορίζει το χρώμα του κειμένου μιας επιλεγμένης παραπομπής. Δηλαδή έχουμε τοποθετήσει τον δρομέα επάνω της, έχουμε πατήσει το αντίστοιχο πλήκτρο του ποντικιού, αλλά δεν το έχουμε ακόμη αφήσει (το κρατάμε πατημένο). Η default παράμετρος είναι συνήθως κόκκινο.

Σημείωση: Η TEXT χρησιμοποιείται για τον καθορισμό του χρώματος ολόκληρου του κειμένου. Για τον χρωματισμό μέρους του κειμένου χρησιμοποιούμε την οδηγία (ορισμός γραμματοσειρών).

Π.χ. κείμενο σε κίτρινο χρώμα

4.7 Λίστες (αριθμημένες και μη)

Αριθμημένες λίστες

Αν θέλουμε να δημιουργήσουμε μια αριθμημένη λίστα του τύπου

1. Άνδρος
2. Μύκονος
3. Τήνος
4. Σίφνος

θα πρέπει να χρησιμοποιήσουμε τις οδηγίες ... και

Η οδηγία (Ordered List) τοποθετείται στην αρχή της λίστας ενώ η οδηγία στο τέλος της.

Κάθε νέα εγγραφή στην λίστα πρέπει να σημειώνεται με την οδηγία (οδηγία δεν χρησιμοποιούμε για τον ίδιο λόγο που δεν χρησιμοποιούμε την </P>)

Έτσι η παραπάνω αριθμημένη λίστα θα πρέπει να γραφτεί ως εξής:

```
<OL><LI>Άνδρος<LI>Μύκονος<LI>Τήνος<LI>Σίφνος</OL>
```

Λίστες χωρίς αρίθμηση (Unordered Lists)

Για να δημιουργήσουμε μια λίστα με κουκίδες όπως η παρακάτω θα ακολουθήσουμε την ίδια ακριβώς μεθοδολογία όπως και με τις αριθμημένες με την διαφορά πως αντί για την οδηγία θα χρησιμοποιήσουμε την .

- Άνδρος
- Μύκονος
- Τήνος
- Σίφνος

Έτσι η παραπάνω μη αριθμημένη λίστα θα πρέπει να γραφτεί ως εξής:

```
<UL><LI>Άνδρος<LI>Μύκονος<LI>Τήνος<LI>Σίφνος</UL>
```

Τα ενδεικτικά σημάδια κάθε μέρους μιας μη αριθμημένης λίστας μπορούν να οριστούν με ειδικές παραμέτρους και να πάρουν τις εξής μορφές: Μαύρα δισκάκια (UL TYPE=DISC που είναι το default), Κυκλάκια (UL TYPE=CIRCLE), Τετραγωνάκια (UL TYPE=SQUARE).

4.8 Επικεφαλίδες (Headings)

Η οδηγία επικεφαλίδας καθορίζει το μέγεθος των γραμμάτων της επικεφαλίδας και ισχύει ανεξάρτητα από την γραμματοσειρά που χρησιμοποιεί ο browser για να διαβάσει την σελίδα.

Η μεγαλύτερη σε μέγεθος επικεφαλίδα είναι η <H1>, ακολουθούμενη από τις <H2>...<H6>.

4.9 Στοιίχιση <ALIGN>

Η στοιίχιση κειμένου ή φωτογραφιών γίνεται με την ALIGN. Σημειώνουμε ότι η ALIGN δεν είναι οδηγία αλλά παράμετρος. Δηλαδή λειτουργεί πάντοτε μέσα σε μια οδηγία. Ποτέ μόνη της.

Για να στοιχίσουμε ένα κείμενο αριστερά (ALIGN=LEFT που είναι το default), δεξιά (ALIGN=RIGHT), ή στο κέντρο (ALIGN=CENTER) χρησιμοποιούμε την ALIGN μέσα σε μια παράγραφο.

Για να στοιχίσουμε το κείμενο μιας επικεφαλίδας, χρησιμοποιούμε την ALIGN ως εξής:

```
<H1 ALIGN=X>κείμενο</H1>
```

Όπου η X μπορεί να πάρει τις τιμές **LEFT CENTER RIGHT** ανάλογα με την θέση που θέλουμε να βρίσκεται.

Παρόμοια οδηγία είναι και η <DIV>...</DIV> Λειτουργεί όπως ακριβώς και η <ALIGN> με την διαφορά πως αναφέρεται σε μια σειρά από επικεφαλίδες ή παραγράφους συνολικά. Στο παράδειγμα που ακολουθεί οι Επικεφαλίδες 1 και 2 και η Παράγραφος θα στοιχηθούν δεξιά.

Κώδικας:

```
<DIV ALIGN=RIGHT><H1>Επικεφαλίδα 1</H1><H3>Επικεφαλίδα  
2</H3><P>Παράγραφος με κείμενο</DIV>
```

Στοιχείση στο κέντρο

Η οδηγία `<CENTER>...</CENTER>` στοιχίζει στο κέντρο όλα όσα περικλύει (πίνακες, εικόνες, κείμενο κ.λπ.). Αν και θεωρείται πιο σωστό να χρησιμοποιούμε την `<ALIGN>` γι' αυτή την εργασία, υπάρχουν πολλές περιπτώσεις που η `<CENTER>` αποδεικνύεται προτιμότερη.

4.10 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.

`...`

Αν το επιθυμούμε, υπάρχει και η δυνατότητα να δώσουμε ξεχωριστά χρώματα σε μεμονωμένες λέξεις, φράσεις ή και χαρακτήρες. Αυτό γίνεται με την οδηγία FONT.

Π.χ. `κείμενο σε κίτρινο χρώμα`

`...`

Καθορίζει τη γραμματοσειρά. Καλό είναι να χρησιμοποιείται με φειδώ διότι αν η γραμματοσειρά δεν υπάρχει στον H/Y του αναγνώστη των σελίδων μπορεί να υπάρξουν προβλήματα (π.χ. να μην φαίνονται τα κείμενα με Ελληνικούς χαρακτήρες).

Π.χ. `...`

4.11 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

Αυτό μπορεί να γίνει με την οδηγία ****

Π.χ. `` Με την οδηγία αυτή η εικόνα θα μπει στην σελίδα στο σημείο που 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.

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>`

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:

ALIGN=TOP

Θα βάλει το πάνω μέρος της εικόνας στο ίδιο επίπεδο με την γραμμή του κειμένου.

ALIGN=MIDDLE

θα βάλει το μεσαίο επίπεδο 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).

Πως κάνουμε μια εικόνα παραπομπή (LINK)

Αυτό μπορεί να γίνει πολύ απλά με τον ίδιο τρόπο που γίνεται και για μια ακολουθία κειμένου. Απλώς αντί για κείμενο, χρησιμοποιούμε την εντολή της εικόνας.

Π.χ. αν το αρχείο dancers.gif έχει δύο χορευτές, τότε η οδηγία για να γίνει το αρχείο παραπομπή σε ένα χορευτικό site θα είναι:

```
<A HREF="http://www.ballroomdancers.com"><IMG SRC=dancers.gif"></A>
```

Οι παράμετροι HEIGHT & WIDTH

Τα height & width ορίζουν το ύψος και το πλάτος της εικόνας, σε pixels. Δεν είναι απαραίτητα αλλά συνιστάται να χρησιμοποιούνται πάντοτε για τον ακόλουθο λόγο:

Η σύνταξη των παραμέτρων είναι:

```
<IMG SRC="image2.gif" border=5 height=38 width=756 >
```

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.

4.12 Πώς δημιουργούμε απλούς πίνακες

Η οδηγία <TABLE> χρησιμοποιείται για να δημιουργούμε πίνακες αλλά και για να έχουμε μεγαλύτερη έλεγχο στο στήσιμο μιας σελίδας.

Μερικοί όροι που πρέπει να γνωρίζετε πριν ασχοληθείτε με τους πίνακες είναι οι:

Caption: Το θέμα του πίνακα

Table headings: Επικεφαλίδες στηλών ή γραμμών (τα περιεχόμενά τους παρουσιάζονται με bold γράμματα)

Table data: Τα δεδομένα των κελιών του πίνακα

Border: Το περιθώριο του πίνακα και των κελιών του

Για να κατασκευάσουμε ένα πίνακα, ορίζουμε την κάθε γραμμή ξεχωριστά (μπορούμε να έχουμε όσες γραμμές θέλουμε) και μέσα σε κάθε γραμμή ορίζουμε τα κελιά της. Οι στήλες ορίζονται αυτόματα με βάση το πόσα κελιά υπάρχουν σε κάθε γραμμή.

Ένα απλό παράδειγμα πίνακα χωρίς ορατά περιθώρια που περιλαμβάνει μια σειρά,

και τέσσερα κελιά (το ένα από τα οποία είναι η επικεφαλίδα) είναι το παρακάτω:

ΕΠΙΚΕΦΑΛΙΔΑ	ΔΕΔΟΜΕΝΑ	ΔΕΔΟΜΕΝΑ	ΔΕΔΟΜΕΝΑ
-------------	----------	----------	----------

Ο κώδικας που μας έδωσε το παραπάνω αποτέλεσμα είναι:

```
<TABLE>  
<TR><TH>ΕΠΙΚΕΦΑΛΙΔΑ</TH><TD>ΔΕΔΟΜΕΝΑ</TD>  
<TD>ΔΕΔΟΜΕΝΑ</TD><TD>ΔΕΔΟΜΕΝΑ</TD></TR> </TABLE>
```

Τα βήματα που κάναμε για να δημιουργήσουμε τον πίνακα ήταν:

1. Καθορίσαμε το σχήμα του πίνακα (αποφασίσαμε δηλαδή πως θα έχει μια γραμμή με 4 κελιά).

2. Ορίσαμε τον πίνακα <TABLE>...</TABLE>
3. Ορίσαμε την γραμμή <TR>...</TR>
4. Ορίσαμε (ένα ένα) τα κελιά της γραμμής <TH>...</TH> (για κελί επικεφαλίδας) και <TD>...</TD> (για κελί δεδομένων)

Παράμετρος BORDER

Η **BORDER** καθορίζει αν ο πίνακας θα έχει περιθώρια ή όχι. Ο πίνακας του παραπάνω παραδείγματος με ορατά περιθώρια είναι:

ΕΠΙΚΕΦΑΛΙΔΑ	ΔΕΔΟΜΕΝΑ	ΔΕΔΟΜΕΝΑ	ΔΕΔΟΜΕΝΑ
--------------------	-----------------	-----------------	-----------------

```
<TABLE BORDER>
```

```
<TR><TH>ΕΠΙΚΕΦΑΛΙΔΑ</TH><TD>ΔΕΔΟΜΕΝΑ</TD>
```

```
<TD>ΔΕΔΟΜΕΝΑ</TD><TD>ΔΕΔΟΜΕΝΑ</TD></TR> </TABLE>
```

Ένας πιο περίπλοκος πίνακας είναι ο ακόλουθος:

Παραγγελία	Αγορά
400	300

```
<TABLE BORDER>
```

```
<TR><TH> Παραγγελία </TH><TH> Αγορά </TH></TR>
```

```
<TR><TD>400</TD><TD>300</TD></TR> </TABLE>
```

Στα παραπάνω παραδείγματα είδαμε πως ένα κελί εκτείνεται για να πάρει το μέγεθος του περιεχομένου του. Με τον τρόπο αυτό φυσικά συμπαρασύρει και τα άλλα κελιά που βρίσκονται στην ίδια στήλη ή/και γραμμή με αυτό, αναπροσαρμόζοντας και τις δικές τους διαστάσεις. Αν θέλουμε να αλλάξουμε τον τρόπο που εμφανίζεται το κείμενο μέσα σε ένα κελί θα πρέπει αντί να βάλουμε το κείμενο σκέτο (χωρίς μορφοποίηση), να ορίσουμε στον browser με ποιο τρόπο θα γίνεται η εμφάνισή του. Π.χ.

Παραγγελία	Αγορά	Αλλαγή
-------------------	--------------	---------------

300	200	
400		

<TABLE BORDER>

<TR><TH> Παραγγελία </TH><TH> Αγορά </TH><TH> Αλλαγή </TH></TR>

<TR><TD>300
400</TD><TD>200</TD><TD>
</TD></TR>

</TABLE>

Βλέπουμε πως υποχρεώσαμε το κείμενο ενός κελιού να αλλάξει γραμμή. Επίσης, έχουμε και ένα κενό κελί. Για την ακρίβεια έχουμε ένα ανύπαρκτο κελί. Για να γίνει κενό, θα πρέπει να του προσθέσουμε μια αλλαγή γραμμής (αντί για <TD></TD> να γίνει <TD>
</TD>).

<CAPTION>...</CAPTION> Τοπ θέμα (Τίτλος) του πίνακα

Η οδηγία <CAPTION> καθορίζει τον τίτλο του πίνακα. Θα μπορούσε φυσικά να χρησιμοποιηθεί και κάποια άλλη οδηγία (π.χ. παραγράφου <p> ή κεφαλίδας <h>) αλλά το <CAPTION> είναι προτιμότερο για λόγους ομοιομορφίας. Το μειονέκτημά του είναι πως δεν μπορεί να γίνει σωστή ρύθμιση (με την παράμετρο ALIGN) της θέσης του (η παράμετρος έχει διαφορετική χρήση στο Netscape από ότι στον Internet Explorer).

Παράδειγμα:

Παραγγελία		
Πάτρα	Ιωάννινα	Αργίνιο
κελί 1 με δύο γραμμές	κελί 2	

<TABLE BORDER> <CAPTION> Παραγγελία </CAPTION>

<TR><TH>Πάτρα</TH><TH>Ιωάννινα</TH><TH>Αργίνιο</TH></TR>

<TR><TD>κελί 1
με δύο γραμμές</TD><TD>κελί

2</TD><TD>
</TD></TR>

</TABLE>

Βλέπουμε πως και μέσα στην <CAPTION> μπορούμε να συμπεριλάβουμε άλλες HTML οδηγίες.

Η θέση ενός πίνακα μέσα σε μια σελίδα

Όσον αφορά την θέση που θα πάρει μέσα στην σελίδα, ο πίνακας αντιμετωπίζεται από τον browser σαν μια εικόνα (τοποθετείται στο αριστερό μέρος της σελίδας με κείμενο μόνο από πάνω ή από κάτω του). Έτσι μπορούμε με τις παραμέτρους ALIGN και CLEAR να τον τοποθετήσουμε αριστερά ή δεξιά στην σελίδα και να βάλουμε ροή κειμένου στο πλάι του.

!!! Στην περίπτωση αυτή, δεν υπάρχει παράμετρος ALIGN=CENTER. Αν θέλουμε να βάλουμε έναν πίνακα στο μέσον μιας σελίδας, θα πρέπει να χρησιμοποιήσουμε την οδηγία <CENTER>. Π.χ. <CENTER><TABLE>.....</TABLE></CENTER>

Η θέση των περιεχομένων των κελιών ενός πίνακα

Μπορούμε να έχουμε τα περιεχόμενα των κελιών στοιχημένα (με την γνωστή παράμετρο ALIGN) στις ακόλουθες θέσεις:

ΟΡΙΖΟΝΤΙΑ:

ALIGN=LEFT (αριστερά) **ALIGN=RIGHT** (δεξιά) **ALIGN=CENTER** (κέντρο, το default)

ΚΑΘΕΤΑ:

VALIGN=TOP (πάνω) **VALIGN=MIDDLE** (μέση, το default) **VALIGN=BOTTOM** (κάτω)

Περίπλοκοι πίνακες

ROWSPAN & COLSPAN, κελιά που εκτείνονται σε περισσότερες από μία σειρές ή στήλες.

Μπορούμε να ορίσουμε κελιά με μέγεθος μεγαλύτερο από μια σειρά ή μια στήλη.

Για να έχει το κελί μας μέγεθος μεγαλύτερο από μια στήλη χρησιμοποιούμε την παράμετρο COLSPAN π.χ. <TH COLSPAN=X>...</TH> όπου X= ο αριθμός των στηλών που θέλουμε να καλύψει το συγκεκριμένο κελί.

Αντίστοιχα χρησιμοποιείται και η ROWSPAN π.χ. <TH ROWSPAN=X>...</TH> όπου X= ο αριθμός των γραμμών που θέλουμε να καλύψει το συγκεκριμένο κελί.

Ο αριθμός των στηλών εκτείνεται πάντοτε προς τα δεξιά, και των γραμμών προς τα κάτω.

Παράδειγμα:

ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΛΟΓΩΝ					
		ΨΗΦΟΙ ΚΟΜΜΑΤΩΝ			ΑΚΥΡΑ-ΛΕΥΚΑ
		ΚΟΜΜΑ Α'	ΚΟΜΜΑ Β'	ΚΟΜΜΑ Γ'	
ΦΥΛΟ	ΑΝΔΡΕΣ	125312	325987	96523	1856
	ΓΥΝΑΙΚΕΣ	165874	319856	117823	148

<TABLE BORDER>

<CAPTION>ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΛΟΓΩΝ</CAPTION>

<TR><TD COLSPAN=2 ROWSPAN=2></TD> <TH COLSPAN=3>ΨΗΦΟΙ ΚΟΜΜΑΤΩΝ</TH> <TH ROWSPAN=2>ΑΚΥΡΑ-ΛΕΥΚΑ</TH></TR>

<TR><TH>ΚΟΜΜΑ Α'</TH><TH>ΚΟΜΜΑ Β'</TH><TH>ΚΟΜΜΑ Γ'</TH><TH></TH> </TR>

<TR><TH ROWSPAN=2>ΦΥΛΟ</TH><TH>ΑΝΔΡΕΣ</TH>
<TD>125312</TD><TD>325987</TD><TD>96523 </TD><TD>1856</TD></TR>

<TR><TH>ΓΥΝΑΙΚΕΣ</TH><TD>165874</TD>
<TD>319856</TD><TD>117823</TD><TD>148 </TD></TR>

</TABLE>

Μέγεθος πίνακα

Το μέγεθος του πίνακα μπορεί να οριστεί και με την παράμετρο WIDTH (<TABLE BORDER WIDTH=X>) όπου X = το πλάτος του πίνακα. Το πλάτος μπορεί να οριστεί σαν ποσοστό του παραθύρου (π.χ. 75%) ή σαν απόλυτος αριθμός (αριθμός pixels π.χ. 500). Προτιμότερο είναι να χρησιμοποιούνται σχετικές ρυθμίσεις για τα πλάτη (ποσοστά) διότι δεν μπορούμε να γνωρίζουμε το μέγεθος του παραθύρου του browser την στιγμή που διαβάζονται οι σελίδες μας.

ΠΡΟΣΟΧΗ! Υπάρχουν περιπτώσεις που ο browser δεν μπορεί να εκτελέσει την παράμετρο WIDTH όπως την επιθυμούμε. Αν τύχει κάτι τέτοιο, τότε θα προσπαθήσει να ορίσει τον πίνακα όσο πιο κοντά στα ζητούμενα από εμάς μπορεί.

Μέγεθος εξωτερικού περιθωρίου κελιών (CELLSPACING)

Η παράμετρος αυτή (που τοποθετείται μέσα στην οδηγία <TABLE>), καθορίζει την απόσταση (σε pixels) μεταξύ των κελιών (άρα και το μεταξύ τους περιθώριο). Η default τιμή είναι 2.

AAA	BBB
111	222

Π.χ.

```
<TABLE BORDER=5 CELLSPACING=4>
```

```
<TR><TH>AAA</TH><TH>BBB</TH></TR>
```

```
<TR><TD>111</TD><TD>222</TD></TR></TABLE>
```

Μέγεθος εσωτερικού περιθωρίου κελιών (CELLPADDING)

Η παράμετρος αυτή (που τοποθετείται μέσα στην οδηγία <TABLE>), καθορίζει την απόσταση (σε pixels) μεταξύ του εξωτερικού περιθωρίου των κελιών και του περιεχομένου τους. Η default τιμή είναι 1. Μπορεί να πάρει και την τιμή 0 αλλά καλό είναι να αποφεύγεται για λόγους αισθητικής.

4.13 Πώς δημιουργούμε μια φόρμα επικοινωνίας

Για να δημιουργήσουμε μια φόρμα στην οποία θα γράφουν σχόλια ή παρατηρήσεις οι επισκέπτες των σελίδων μας, χρειαζόμαστε την οδηγία `<FORM>` η λειτουργία της οποίας ρυθμίζεται από **τις παραμέτρους METHOD** (παίρνει τις τιμές GET και POST) και **ACTION**.

Η παράμετρος **ACTION** καλεί ένα CGI script, με άλλα λόγια ένα προγραμματάκι (τεχνικά δεν είναι πρόγραμμα, απλώς μια ακολουθία εντολών) που θα φροντίσει για την διαχείριση των δεδομένων που θα δώσει ο χρήστης.

Η παράμετρος **METHOD**, καθορίζει με ποιο τρόπο θα λάβει τα δεδομένα το CGI script. Όταν έχει την τιμή **GET**, τότε τα περιεχόμενα της φόρμας θα προστεθούν στο τέλος του URL που θα λάβει το CGI script. Με την τιμή **POST** το CGI script θα λάβει ξεχωριστά τα δεδομένα από το URL το οποίο του τα αποστέλλει (το URL χρειάζεται διότι όπως θα δούμε ένα CGI script μπορεί να εξυπηρετεί πολλές φόρμες, άρα πρέπει να γνωρίζει ποια του στέλνει τα δεδομένα που διαχειρίζεται). Συνήθως χρησιμοποιούμε την τιμή POST διότι δεν έχει περιορισμούς στην ποσότητα δεδομένων που θα δώσει ο χρήστης της φόρμας. Η GET έχει κάποιους περιορισμούς και σε μεγάλες ποσότητες δεδομένων, μπορεί κάποια από αυτά να χαθούν. Επίσης, οι διαχειριστές υπολογιστικών συστημάτων την αποφεύγουν διότι έχει αποδειχθεί ιδιαίτερα ευάλωτη σε επιθέσεις hackers.

Με βάση τα παραπάνω η <FORM> συντάσσεται ως εξής:

```
<FORM METHOD=POST ή GET ACTION="http://διεύθυνση κόμβου/directory/όνομα φόρμας">...</FORM>
```

Στα παρακάτω παραδείγματα θα χρησιμοποιήσουμε ένα CGI script με το όνομα post-query που βρίσκεται στην διεύθυνση `http://www.mcp.com/cgi-bin/post-query`

ΣΥΜΒΟΥΛΗ: Ο κόμβος στον οποίο θα τοποθετήσετε τις σελίδες σας, έχει μια σειρά από CGI scripts. Φροντίστε να μάθετε ποια είναι αυτά διότι αν και μπορείτε να χρησιμοποιήσετε scripts και από άλλους κόμβους θα είναι πολύ ταχύτερη η λειτουργία αν χρησιμοποιήσετε κάποιο που βρίσκεται στον ίδιο κόμβο.

Παράδειγμα:

```
<FORM METHOD=POST ACTION="http://www.mcp.com/cgi-bin/post-  
query">...</FORM>
```

Είδαμε το πλαίσιο μέσα στο οποίο δημιουργείται μια φόρμα. Για να δημιουργήσουμε τους χώρους στους οποίους θα τοποθετήσει τα δεδομένα του ο χρήστης χρησιμοποιούμε μια άλλη οδηγία, την <INPUT>.

<INPUT>

Καθορίζει τους χώρους στους οποίους θα τοποθετήσει τα δεδομένα του ο χρήστης. Λειτουργεί με τις **παραμέτρους TYPE ή/και NAME**.

Η **TYPE** καθορίζει το είδος των δεδομένων που θα πάρει η φόρμα από τον χρήστη. Αν δεν υπάρχει, τότε θεωρείται πως έχει την τιμή text (TYPE="text") που είναι η default (δηλαδή τα δεδομένα που εισάγονται είναι σε μορφή κειμένου).

Η **NAME** καθορίζει το όνομα (λεζάντα) που δίδεται στο πεδίο αυτό και το CGI script μας δίνει το όνομα δίπλα σε κάθε τιμή. Π.χ. αν ο χρήστης θα γράψει το επάγγελμά του τότε μπορεί να δώσουμε στην NAME το όνομα profession έτσι ώστε τα δεδομένα που θα λαμβάνουμε να παρουσιάζονται με μορφή του τύπου profession = economist Αν δεν υπήρχε η NAME θα λαμβάναμε μόνο όσα έγραψε ο χρήστης στο INPUT χωρίς καμία πληροφορία για τη σημασία τους.

Για παράδειγμα αν ζητούσαμε το βάρος και την ηλικία όσων συμπληρώνουν τη φόρμα θα λαμβάναμε μόνα τους τα νούμερα 59 και 55 χωρίς να γνωρίζουμε αν ο χρήστης είναι 55 ετών με βάρος 59 κιλά ή 59 ετών με βάρος 55 κιλά. Αν όμως έχουμε χρησιμοποιήσει την NAME θα λάβουμε:

59

Weight:

Age: 55

όπου Weight και Age είναι οι λεζάντες (NAME) των INPUT πεδίων.

Με την υιοθέτηση της NAME, το παραπάνω παράδειγμα συντάσσεται ως εξής:

```
<FORM METHOD=POST ACTION="http://www.mcp.com/cgi-bin/post-query"> <P>  
Give us your name: <INPUT NAME="theName"> </FORM>
```

Αυτό που μένει τώρα είναι να δώσουμε στο χρήστη την δυνατότητα να μας στείλει τα στοιχεία που μόλις συμπλήρωσε. Αυτό θα γίνει πάλι με την <INPUT> ως εξής: <INPUT TYPE="submit"> με την οποία λέμε στον browser να δημιουργήσει το εικονίδιο υποβολής των δεδομένων της φόρμας.

Έτσι το παράδειγμά μας γίνεται:

```
<FORM METHOD=POST ACTION="http://www.mcp.com/cgi-bin/post-query">
<P> Give us your name: <INPUT NAME="theName"><P><INPUT TYPE="submit">
</FORM>
```

RESET

Πολλές φορές τυχαίνει ο χρήστης που συμπλήρωσε μια φόρμα, να θέλει να ακυρώσει όλες τις επιλογές του για να τις αντικαταστήσει με άλλες. Αυτό μπορεί να γίνει με την τιμή RESET της παραμέτρου TYPE που δημιουργεί ένα άλλο πλήκτρο επιλογής. Π.χ.

```
<INPUT TYPE="RESET" VALUE="Ακύρωση Επιλογών">
```

Άλλες παράμετροι σχετικά με τις φόρμες

VALUE

Με την παράμετρο VALUE, μπορούμε να αλλάξουμε το κείμενο του εικονιδίου υποβολής των δεδομένων της φόρμας. Π.χ. αν θέλουμε αντί για Submit Query να γράφει Υποβολή θα αλλάξουμε την οδηγία σε:

```
<INPUT TYPE="submit" VALUE="Υποβολή">
```

TEXT

Μπορούμε να δώσουμε στον χρήστη την δυνατότητα να γράψει κάποιο μικρό κείμενο που θα συμπεριληφθεί στην φόρμα.

Αν θέλουμε το κείμενο να βρίσκεται όλο σε μια γραμμή, τότε θα χρησιμοποιήσουμε την TEXT ως εξής: <INPUT TYPE="text" NAME="X"> όπου X είναι το όνομα που θα δώσουμε στο πεδίο αυτό.

Το μήκος της γραμμής μπορεί να είναι όσο μεγάλο θέλουμε (αν δεν χωράει θα σκρολλάρει κατά το πλάτος της οθόνης), αλλά για πρακτικούς λόγους καλό είναι να το περιορίζουμε στους 50 χαρακτήρες. Η default τιμή είναι 20 χαρακτήρες αλλά μπορεί να αλλάξει με την παράμετρο SIZE π.χ.

```
<INPUT TYPE="text" NAME="X" SIZE="50">
```

PASSWORD

Η τιμή PASSWORD της παραμέτρου TYPE χρησιμοποιείται για να απεικονίζει στην οθόνη αστερίσκους ή σφαιρίδια αντί για τους χαρακτήρες που πληκτρολογεί ο χρήστης. Π.χ.

```
<INPUT TYPE="password" NAME="XYZ">
```

ΠΡΟΣΟΧΗ! Το να αποστέλλονται passwords με αυτό τον τρόπο δεν είναι ασφαλές. Μπορεί στην οθόνη να μην εμφανίζεται τι γράφει ο χρήστης αλλά αυτά που έγραψε αποστέλλονται στο cgi script για επεξεργασία σε text μορφή.

RADIO

Πιθανώς να χρειάζεται να δημιουργήσουμε μια φόρμα που ο χρήστης θα επιλέγει μεταξύ κάποιων προκαθορισμένων τιμών για αυτά που θα συμπεριληφθούν στην φόρμα που συμπληρώνει (multiple choice). Ένας τρόπος να γίνει αυτό είναι με τα λεγόμενα RADIO BUTTONS. Για να το επιτύχουμε αυτό συνήθως χρησιμοποιούμε μια λίστα όπου TYPE="radio" Name="όποιο θέλουμε αλλά το ίδιο για όλα" Value="η τιμή κάθε επιλογής" Π.χ.

```
<OL>
```

```
<LI><INPUT TYPE="radio" NAME="country" Value="USA">USA
```

```
<LI><INPUT TYPE="radio" NAME="country" Value="CANADA">CANADA
```

```
<LI><INPUT TYPE="radio" NAME="country" Value="UK">UK
```

```
<LI><INPUT TYPE="radio" NAME="country" Value="GREECE">GREECE
```

```
</OL>
```

CHECKBOX

Στα RADIO BUTTONS μπορούμε να επιλέξουμε μόνο μια τιμή από την λίστα τιμών που έχουμε στην διάθεσή μας. Αν θέλουμε να επιλέγονται περισσότερες τιμές, τότε χρησιμοποιούμε την τιμή CHECKBOX στην παράμετρο TYPE. Π.χ.

```
<UL>
<LI><INPUT TYPE="CHECKBOX" NAME="USA" Value="chosen">USA
<LI><INPUT TYPE="CHECKBOX" NAME="CANADA" Value="chosen">CANADA
<LI><INPUT TYPE="CHECKBOX" NAME="UK" Value="chosen">UK
<LI><INPUT TYPE="CHECKBOX" NAME="GREECE" Value="chosen"
CHECKED>GREECE
</UL>
```

<SELECT>...</SELECT>

Η οδηγία αυτή δημιουργεί ένα pull down menu επιλογών από τις οποίες ο χρήστης μπορεί να επιλέξει μια. Αυτή θα σταλεί στο cgi script μαζί με την τιμή της NAME. Π.χ.

```
<P>SELECT A COUNTRY:
<SELECT NAME="COUNTRY">
<OPTION>GREECE
<OPTION>FRANCE
<OPTION>GERMANY
<OPTION>ITALY
</SELECT>
```

Οι επιλογές εμφανίζονται στο παράθυρο που δημιουργείται με την σειρά που τις καταγράψαμε παραπάνω.

Διαχείριση των αποτελεσμάτων μιας φόρμας

Υπάρχουν διάφοροι τρόποι να αποκτά κανείς τα αποτελέσματα μιας φόρμας που κατασκευάζει ανάλογα με το cgi script που χρησιμοποιεί. Ένας από τους πιο συνηθισμένους είναι να τα λαμβάνει με email. Ένα τέτοιο παράδειγμα έχουμε στην παρακάτω περίπτωση όπου χρησιμοποιούμε το script FormMail.pl:

```

<P><FORM METHOD=POST ACTION="http://www.eexi.gr/cgi-bin/FormMail.pl">
<P> Γράψτε το όνομά σας: <INPUT NAME="theName"><P> <UL>
<LI><INPUT TYPE="CHECKBOX" NAME="USA" Value="chosen">USA
<LI><INPUT TYPE="CHECKBOX" NAME="CANADA" Value="chosen">CANADA
<LI><INPUT TYPE="CHECKBOX" NAME="UK" Value="chosen">UK
<LI><INPUT TYPE="CHECKBOX" NAME="GREECE" Value="chosen"
CHECKED>GREECE
</UL><p><TEXTAREA NAME="dokimh" ROWS="10" COLS="45">Γράψτε εδώ τις
παρατηρήσεις ή τα σχόλιά σας:</TEXTAREA><P><INPUT TYPE="hidden"
NAME="recipient" VALUE="gepiti@eexi.gr"><INPUT TYPE="submit"
VALUE="Υποβολή"><P><INPUT TYPE="RESET" VALUE="Ακύρωση Επιλογών">
</FORM>

```

4.14 Πώς κάνουμε μια σελίδα να καλεί αυτόματα μια άλλη (redirection)

Πολλές φορές μια δημοφιλής σελίδα πρέπει να αλλάξει URL. Σε αυτή την περίπτωση, συνήθως στο παλιό URL μπαίνει μια μικρή ανακοίνωση που λέει πως η διεύθυνση έχει αλλάξει και προτρέπει τον αναγνώστη να κάνει κλικ στην νέα διεύθυνση.

Για να διευκολύνουμε τον επισκέπτη της σελίδας μπορούμε, με πολύ εύκολο τρόπο, να αυτοματοποιήσουμε αυτή τη διαδικασία προσθέτοντας στην οδηγία body την εντολή: `onLoad="location.href='http://www.xxxxxx.gr' "`. Για παράδειγμα:

```
<BODY onLoad="location.href='http://www.xxxxxx.gr'">
```

Αν κάποιος ζητήσει αυτή την σελίδα, τότε αυτή θα φορτωθεί και θα καλέσει αμέσως την σελίδα που βρίσκεται στο `http://www.xxxxxx.gr`. Η οδηγία λειτουργεί με όλους τους νέους browsers αλλά προτείνεται να προσθέσετε στη σελίδα και ένα κείμενο του τύπου: *"To URL άλλαξε. Αν δεν μεταφερθείτε αυτόματα στη νέα διεύθυνση κάντε κλικ ..."*

4.15 Πώς κάνουμε μια σελίδα να καλεί κατά τακτά χρονικά διαστήματα τον εαυτό της (refresh)

Πολλές φορές μια σελίδα αλλάζει πάρα πολύ συχνά (π.χ. κάθε μερικά λεπτά). Δυστυχώς, ο μόνος τρόπος για να δει ο χρήστης αν η σελίδα έχει αλλάξει από την τελευταία φορά που την κάλεσε είναι να κάνει Reload. Μόνο έτσι εξασφαλίζει ότι η έκδοση που βλέπει είναι η πιο πρόσφατη.

Αν όμως η σελίδα αλλάζει συνέχεια και δεν σας αρέσει η προοπτική να κάθεται ο χρήστης μπροστά στο pc του και να κάνει συνέχεια κλικ στο Reload μπορείτε να κάνετε τη σελίδα να ανανεώνεται (να κάνει refresh) μόνη της.

Για να το επιτύχουμε αυτό χρησιμοποιούμε την παρακάτω οδηγία:

```
<META HTTP-EQUIV="refresh" CONTENT="5";  
URL=http://host.gr/directory/filename.htm">
```

Όπου `CONTENT="5";` ο χρόνος που μεσολαβεί μεταξύ δύο refresh (στο συγκεκριμένο παράδειγμα η σελίδα κάνει αυτόματα refresh κάθε 5 δευτερόλεπτα).

Όπως όλες οι meta tags, η οδηγία αυτή πρέπει να τοποθετηθεί μέσα στο HEAD και αμέσως μετά τον TITLE. Για παράδειγμα η συνολική διατύπωση θα είναι:

```
<HTML><HEAD><TITLE>Ο τίτλος της σελίδας</TITLE><META HTTP-  
EQUIV="refresh" CONTENT="5";  
URL=http://host.gr/directory/filename.htm"></HEAD><BODY>...
```

4.16 Πώς προσθέτουμε σχόλια στον κώδικα μιας σελίδας

Ένα βασικό χαρακτηριστικό των browsers είναι να αγνοούν τις οδηγίες που δεν καταλαβαίνουν.

Πολλές φορές, έχουμε την ανάγκη να βάλουμε κάπου μέσα στη σελίδα μια παρατήρηση ή κάποιο σχόλιο μόνο για μας (π.χ. μια υπενθύμιση του τύπου: "εδώ τη φωτογραφία"). Θεωρητικά αρκεί να βάλω μια οδηγία του τύπου: <εδώ να βάλω τη φωτογραφία του Κώστα>. Ωστόσο αυτό δεν είναι πολύ βολικό γιατί μπορεί να χαθεί μέσα στον κώδικα και να μην βρίσκεται εύκολα. Γι' αυτό συνήθως ακολουθούμε την ακόλουθη πρακτική:

<!-----εδώ τη φωτογραφία>

Όπως βλέπουμε το θαυμαστικό και τα διαστήματα κάνουν την οδηγία να ξεχωρίζει και γι' αυτό μπορούμε να τη βρούμε πιο εύκολα μέσα στον κώδικα. Μπορούμε ακόμη να κάνουμε και αναζητήσεις (find) με το !---- που δεν χρησιμοποιείται πουθενά αλλού. Έτσι θα βρίσκουμε ακόμη πιο εύκολα τα σχολιά μας.

ΚΕΦΑΛΑΙΟ 5^ο

SQL (Structured Query Language)

5.1 Τι Είναι η SQL

Η SQL αποτελεί μια γλώσσα συγγραφής ερωτημάτων σε βάσεις δεδομένων και όχι μόνο.

Βασικά χαρακτηριστικά SQL

- Τα αρχικά SQL σημαίνουν **Structured Query Language**, δηλ. Δομημένη Γλώσσα Ερωτημάτων.
- Η SQL μάς δίνει τη δυνατότητα να έχουμε πρόσβαση σε μια βάση δεδομένων για την διαχείριση των δεδομένων της.
- Η SQL αποτελεί μια στάνταρ γλώσσα του ANSI (ANSI standard language).

Τι μπορούμε να κάνουμε με την SQL:

- Η SQL μπορεί να εκτελέσει ερωτήματα (queries) σχετικά με μια βάση δεδομένων.
- Η SQL μπορεί να ανακτήσει δεδομένα από μια βάση δεδομένων.
- Η SQL μπορεί να εισαγάγει νέες εγγραφές σε μια βάση δεδομένων.
- Η SQL μπορεί να διαγράψει εγγραφές από μια βάση δεδομένων.
- Η SQL μπορεί να ενημερώσει εγγραφές σε μια βάση δεδομένων.

Η SQL αποτελεί ένα στάνταρ του ANSI (American National Standards Institute) για να μπορούμε να έχουμε πρόσβαση σε συστήματα βάσεων δεδομένων. Οι εντολές της SQL χρησιμοποιούνται για να ανακτήσουμε (retrieve) και να ενημερώσουμε (update) δεδομένα σε μια βάση δεδομένων (database).

Η SQL συνεργάζεται με προγράμματα βάσεων δεδομένων όπως είναι τα εξής : Access, Informix, Microsoft SQL Server, Oracle, Sybase και πολλά άλλα.

5.2 Λίγα λόγια για τους Πίνακες Βάσεων Δεδομένων (Database Tables)

Οι βάσεις δεδομένων (databases) περιέχουν αντικείμενα (objects) που ονομάζονται Πίνακες (Tables). Οι Εγγραφές (Records) των δεδομένων αποθηκεύονται σε αυτούς τους πίνακες. Οι Πίνακες αναγνωρίζονται με τα ονόματά τους, όπως «Personal_Data», «Orders», «Suppliers» κ.ά.

Οι Πίνακες είναι οργανωμένοι σε Πεδία η αλλιώς Στήλες (Columns) και Γραμμές (Rows). Οι Γραμμές (Rows) περιέχουν τις εγγραφές (records), δηλαδή ένα δεδομένο. Με τον όρο δεδομένο αναφερόμαστε στην πλήρη περιγραφή ενός ατόμου, αντικειμένου, φυσικού γεγονότος όπως και σε ενέργειες και πράξεις. Οι Στήλες (Columns) περιέχουν τα δεδομένα, ομαδοποιημένα σύμφωνα με τη λογική και το χαρακτηριστικό τους.

Ακολουθεί ένα παράδειγμα ενός Πίνακα που ονομάζεται «Personal_Data» :

| Last_Name | First_Name | Address | City |
|-----------|------------|-------------------|-------|
| ΖΑΓΓΑΝΑΣ | ΠΑΝΑΓΙΩΤΗΣ | ΚΟΡΥΤΣΑΣ 71 | ΠΑΤΡΑ |
| ΑΡΒΑΝΙΤΗΣ | ΠΟΛΥΔΩΡΑΣ | ΠΕΤΜΕΤΖΑ 30 | ΠΑΤΡΑ |
| ΚΟΡΙΤΣΙΔΗ | ΕΛΕΝΑ | ΚΟΝΟΠΙΣΟΠΟΥΛΟΥ 41 | ΑΘΗΝΑ |

Τα Last_Name, First_Name, Address και City είναι τα Πεδία Στήλες (Columns) του πίνακα. Οι Γραμμές (Rows) περιέχουν τις εγγραφές για τρία άτομα.

5.3 Τα Ερωτήματα της SQL (SQL Queries)

Με την SQL, μπορούμε να κάνουμε ένα ερώτημα (Query) σε μια βάση δεδομένων και να έχουμε ένα αποτέλεσμα (Result) σε μορφή πίνακα (tabular form).

Ένα ερώτημα σαν το εξής :

```
SELECT Last_Name FROM Personal_Data;
```

θα δώσει ένα αποτέλεσμα σαν το εξής :

| |
|------------------|
| Last_name |
| ZAGTANAS |
| APBANITHS |
| KORITSIDH |

Πρέπει να έχουμε υπόψη μας ότι τα συστήματα βάσεων δεδομένων απαιτούν το σύμβολο ; (semicolon) στο τέλος μιας εντολής SQL.

5.4 Διαχείριση Δεδομένων της SQL (Data Manipulation)

Όπως υπονοεί και το όνομά της, η SQL είναι μια σύνταξη για την εκτέλεση ερωτημάτων (queries). Αλλά η γλώσσα της SQL περιλαμβάνει επίσης μια σύνταξη για την ενημέρωση εγγραφών, την εισαγωγή νέων εγγραφών και τη διαγραφή υπαρχόντων εγγραφών.

Αυτές οι εντολές ερωτημάτων και ενημέρωσης αποτελούν μαζί, τη Γλώσσα Χειρισμού Δεδομένων (Data Manipulation Language, **DML**) που αποτελεί κομμάτι της SQL :

- **SELECT** - εξάγει δεδομένα από μια βάση δεδομένων.
- **UPDATE** - ενημερώνει δεδομένα σε μια βάση δεδομένων.
- **DELETE** - διαγράφει δεδομένα από μια βάση δεδομένων.
- **INSERT** - εισάγει νέα δεδομένα σε μια βάση δεδομένων.

5.5 Ορισμός Δεδομένων της SQL (Data Definition)

Η Γλώσσα Ορισμού Δεδομένων (Data Definition Language, **DDL**), που αποτελεί μέρος της SQL, επιτρέπει τη δημιουργία και τη διαγραφή πινάκων μιας βάσης δεδομένων. Μπορούμε επίσης να ορίσουμε indexes (keys), να καθορίσουμε συνδέσμους (links) ανάμεσα στους πίνακες και να επιβάλλουμε περιορισμούς ανάμεσα στους πίνακες μιας βάσης δεδομένων.

Οι σημαντικότερες εντολές DDL στην SQL είναι οι εξής :

- **CREATE TABLE** - δημιουργεί έναν νέον πίνακα σε μια βάση δεδομένων.

- **ALTER TABLE** - τροποποιεί έναν πίνακα σε μια βάση δεδομένων.
- **DROP TABLE** - διαγράφει έναν πίνακα από μια βάση δεδομένων.
- **CREATE INDEX** - δημιουργεί έναν index (search key).
- **DROP INDEX** - διαγράφει έναν index.

Η SQL και οι Ενεργές Σελίδες Διακομιστή

Η SQL αποτελεί ένα σημαντικό κομμάτι της ASP, επειδή το Ενεργό Αντικείμενο Δεδομένων (Active Data Object, **ADO**) που χρησιμοποιείται στην **ASP** (Active Server Pages) για να μπορούμε να έχουμε πρόσβαση σε βάσεις δεδομένων, βασίζεται στην SQL για την πρόσβαση στα δεδομένα.

5.6 Εντολές και λέξεις κλειδιά στην SQL

5.6.1 Η Εντολή Select της SQL

Η εντολή SELECT επιλέγει στήλες (columns) δεδομένων από μια βάση δεδομένων. Το αποτέλεσμα αποθηκεύεται σε μορφή πίνακα και αποκαλείται result set. Την χρησιμοποιούμε για να εμφανίζουμε (επιλέγουμε) πληροφορίες από έναν πίνακα ως εξής :

SELECT ονόματα_στηλών

FROM όνομα_πίνακα ;

Παράδειγμα : Επιλογή Στηλών από έναν Πίνακα

Για να επιλέξουμε τις στήλες «Last_name» και «First_name», χρησιμοποιούμε μια εντολή SELECT, ως εξής :

SELECT Last_name, First_name

FROM Personal_Data;

Το αποτέλεσμα :

| Last_name | First_name |
|-----------|------------|
| ΖΑΓΓΑΝΑΣ | ΠΑΝΑΓΙΩΤΗΣ |
| ΑΡΒΑΝΙΤΗΣ | ΠΟΛΥΔΩΡΑΣ |
| ΚΟΡΙΤΣΙΔΗ | ΕΛΕΝΑ |

5.6.2 To Where Clause της SQL

Το WHERE clause χρησιμοποιείται για να καθορίσουμε ένα κριτήριο επιλογής (selection criteria). Για να μπορέσουμε να επιλέξουμε δεδομένα υπό συνθήκη από έναν πίνακα, πρέπει να προσθέσουμε ένα WHERE clause σε μια εντολή SELECT, ως εξής :

SELECT στήλη1, στήλη2,...

FROM πίνακα

WHERE στήλη συνθήκη τιμή;

Με το WHERE clause, μπορούμε να χρησιμοποιήσουμε αυτές τις συνθήκες :

| Τελεστής (Operator) | Συνθήκη (Condition) |
|---------------------|-------------------------|
| = | Ίσο |
| <> | Όχι ίσο |
| > | Μεγαλύτερο από |
| < | Μικρότερο από |
| >= | Μεγαλύτερο από ή ίσο με |
| <= | Μικρότερο από ή ίσο με |

| | |
|------|-----------|
| LIKE | Υπόδειγμα |
|------|-----------|

Σε μερικές εκδόσεις της SQL, ο τελεστής για το όχι ίσο (\neq), μπορεί να γραφεί ως εξής :
 \neq .

Παράδειγμα : Επιλογή Ατόμων από μια Πόλη

Για να επιλέξουμε μόνο τα άτομα που κατοικούν στην πόλη Πάτρα, προσθέτουμε ένα WHERE clause στην εντολή SELECT, ως εξής :

SELECT *

FROM Personal_Data

WHERE City="Πάτρα";

Το αποτέλεσμα :

| Last_name | First_name | Address | City | Year |
|-----------|------------|-------------|-------|------|
| Ζαγγανάς | Παναγιώτης | Κορυτσάς 71 | Πάτρα | 1970 |
| Αρβανίτης | Πολύδωρος | Πετμεζά 30 | Πάτρα | 1971 |

Έχουμε χρησιμοποιήσει διπλά εισαγωγικά (double quotes) στις τιμές των συνθηκών στα παραδείγματα. Η SQL χρησιμοποιεί μονά εισαγωγικά στις αλφαριθμητικές τιμές

(κείμενο). Τα περισσότερα συστήματα βάσεων δεδομένων αποδέχονται και τα διπλά εισαγωγικά (double quotes). Οι αριθμητικές τιμές δεν πρέπει να περικλείονται σε εισαγωγικά.

Για τις τιμές κειμένου (αλφαριθμητικές) :

Αυτό είναι σωστό :SELECT * FROM Personal_Data WHERE
First_name="Παναγιώτης";

Αυτό δεν είναι σωστό :SELECT * FROM Personal_Data WHERE
First_name=Παναγιώτης;

Για τις αριθμητικές τιμές :

Αυτό είναι σωστό :SELECT * FROM Personal_Data WHERE Year>1965;

Αυτό δεν είναι σωστό :SELECT * FROM Personal_Data WHERE Year>"1965";

5.6.3 Η Συνθήκη LIKE

Η συνθήκη LIKE χρησιμοποιείται για να καθορίσουμε μια αναζήτηση για ένα υπόδειγμα (pattern) σε μια στήλη.

Η σύνταξη είναι ως εξής :

SELECT στήλη

FROM πίνακα

WHERE στήλη

LIKE υπόδειγμα;

Μπορούμε να χρησιμοποιήσουμε το σύμβολο «%» για να ορίσουμε χαρακτήρες μπαλαντέρ (wildcards) πριν και μετά από το υπόδειγμα.

Παράδειγμα : Επιλογή Ατόμων με Υπόδειγμα Ονόματος

Η επόμενη εντολή SQL θα επιστρέψει τα άτομα που το όνομά τους αρχίζει από «Ο».

```
SELECT * FROM Personal_Data
```

```
WHERE First_name LIKE "O%";
```

5.6.4 Οι Λογικοί Τελεστές And και Or

Τα AND και OR ενώνουν δύο ή περισσότερες συνθήκες σε ένα WHERE clause. Ο τελεστής AND εμφανίζει μια γραμμή αν ΟΛΕΣ οι συνθήκες είναι αληθείς (true), ενώ ο τελεστής OR εμφανίζει μια γραμμή αν ΟΠΟΙΑΔΗΠΟΤΕ από τις συνθήκες είναι αληθής (true). Ο αρχικός πίνακας είναι ο εξής :

| Last_name | First_name | Address | City |
|------------|------------|---------------|-------|
| Ζαγγανάς | Παναγιώτης | Κορυτσάς 71 | Πάτρα |
| Αρβανίτης | Πολύδωρος | Πετμεζά 30 | Πάτρα |
| Γεωργιάδης | Νικόλαος | Ι. Καραβίτη 2 | Πάτρα |

Παράδειγμα:

Χρησιμοποιούμε το AND για να εμφανίσουμε αυτούς που το όνομά τους είναι «Νικόλαος» και το επώνυμό τους είναι «Γεωργιάδης» :

```
SELECT * FROM Personal_Data
```

```
WHERE First_name=«Νικόλαος» AND Last_name=«Γεωργιάδης»;
```

Αποτέλεσμα :

| Last_name | First_name | Address | City |
|------------|------------|---------------|-------|
| Γεωργιάδης | Νικόλαος | Ι. Καραβίτη 2 | Πάτρα |

5.6.5 Ο Τελεστής Between ... And

Ο τελεστής BETWEEN ... AND επιλέγει μια περιοχή δεδομένων ανάμεσα σε δύο τιμές. Οι τιμές μπορεί να είναι αριθμοί, κείμενο ή ημερομηνίες.

SELECT όνομα_στήλης **FROM** όνομα_πίνακα

WHERE όνομα_στήλης

BETWEEN τιμή1 **AND**
τιμή2;

Αρχικός πίνακας :

| Last_name | First_name | Address | City |
|-------------|------------|-----------------|--------|
| Καράμπελας | Παναγιώτης | Κορυτσάς 71 | Πάτρα |
| Γεωργιάδης | Πολύδωρος | Πετμεζά 30 | Πάτρα |
| Γεωργιάδης | Νικόλαος | Ι. Καραβίτη 2 | Πάτρα |
| Μαρκόπουλος | Νικόλαος | Φον Κάραγιαν 20 | Κοζάνη |

Παράδειγμα:

Για να εμφανίσουμε τα άτομα που βρίσκονται αλφαβητικά ανάμεσα στους «Γεωργιάδης» και «Μαρκόπουλος», αλλά και να τους περιλαμβάνουν :

SELECT * FROM Personal_Data **WHERE** Last_name

BETWEEN «Γεωργιάδης» **AND** «Μαρκόπουλος»;

Αποτέλεσμα :

| Last_name | First_name | Address | City |
|------------------|-------------------|-----------------|-------------|
| Γεωργιάδης | Πολύδωρος | Πετμεζά 30 | Πάτρα |
| Γεωργιάδης | Νικόλαος | Ι. Καραβίτη 2 | Πάτρα |
| Μαρκόπουλος | Νικόλαος | Φον Κάραγιαν 20 | Κοζάνη |

Παράδειγμα:

Για να εμφανίσουμε τα άτομα που βρίσκονται έξω από την περιοχή που χρησιμοποιήσαμε στο προηγούμενο παράδειγμα, χρησιμοποιούμε τον τελεστή NOT :

```
SELECT * FROM Personal_Data WHERE Last_name
```

```
NOT BETWEEN «Γεωργιάδης» AND «Μαρκόπουλος»;
```

Αποτέλεσμα :

| Last_name | First_name | Address | City |
|------------------|-------------------|----------------|-------------|
| Καράμπελας | Παναγιώτης | Κορυτσάς 71 | Πάτρα |

5.6.6 Η Λέξη Κλειδί Distinct

Η λέξη κλειδί DISTINCT χρησιμοποιείται για να επιστρέφει μόνο διακριτές (διαφορετικές) (distinct, different) τιμές. Η εντολή SELECT της SQL επιστρέφει στοιχεία από τις στήλες ενός πίνακα, αλλά τι μπορούμε να κάνουμε αν θέλουμε να επιλέξουμε μόνο διακριτά στοιχεία (distinct elements);

Στην SQL, αυτό που πρέπει να κάνουμε είναι να προσθέσουμε μια λέξη κλειδί **DISTINCT** στην εντολή **SELECT**, ως εξής :

SELECT DISTINCT ονόματα_στηλών **FROM** όνομα_πίνακα;

Παράδειγμα:

Επιλογή εταιρειών από έναν πίνακα παραγγελιών.

Ένας απλός πίνακας παραγγελιών :

| Company | OrderNumber |
|----------------|--------------------|
| Line Computers | 3412 |
| Sony | 2312 |
| Algorithm | 4678 |
| Sony | 6798 |

Η επόμενη εντολή SQL : **SELECT Company FROM Orders;**

θα δώσει αυτό το αποτέλεσμα :

| Company |
|----------------|
| Line Computers |
| Sony |
| Algorithm |
| Sony |

Βλέπουμε ότι η εταιρεία Sony εμφανίζεται δύο φορές στο αποτέλεσμα. Μερικές φορές δεν το θέλουμε αυτό.

Παράδειγμα:

Επιλογή ξεχωριστών εταιρειών από έναν πίνακα παραγγελιών.

Η επόμενη εντολή SQL :

```
SELECT DISTINCT Company FROM Orders;
```

θα δώσει αυτό το αποτέλεσμα :

| Company |
|----------------|
| Line Computers |
| Sony |
| Algorithm |

Τώρα η εταιρεία Sony εμφανίζεται μόνο μία φορά στο αποτέλεσμα.

5.6.7 Η Λέξη Κλειδί Order By

Η λέξη κλειδί ORDER BY χρησιμοποιείται για να ταξινομήσει το αποτέλεσμα. Το ORDER BY clause χρησιμοποιείται για να ταξινομήσει τις γραμμές.

Πίνακας Παραγγελίες :

| Company | OrderNumber |
|----------------|--------------------|
| Digital Shop | 3412 |
| ABC Shop | 5678 |
| Sony | 2312 |

| | |
|------|------|
| Sony | 6798 |
|------|------|

Παράδειγμα:

Για να εμφανίσουμε τις εταιρείες σε αλφαβητική σειρά :

SELECT Company, OrderNumber FROM Orders ORDER BY Company;

Αποτέλεσμα :

| Company | OrderNumber |
|--------------|-------------|
| ABC Shop | 5678 |
| Digital Shop | 3412 |
| Sony | 6798 |
| Sony | 2312 |

Παράδειγμα:

Για να εμφανίσουμε τις εταιρείες σε αλφαβητική σειρά ΚΑΙ (AND) τις παραγγελίες σε αριθμητική σειρά :

SELECT Company, OrderNumber FROM Orders

ORDER BY Company, OrderNumber;

Αποτέλεσμα :

| Company | OrderNumber |
|----------------|--------------------|
| ABC Shop | 5678 |
| Digital Shop | 3412 |
| Sony | 2312 |
| Sony | 6798 |

Παράδειγμα:

Για να εμφανίσουμε τις εταιρείες σε αντίστροφη αλφαβητική σειρά (reverse alphabetical order) :

```
SELECT Company, OrderNumber FROM Orders
```

```
ORDER BY Company DESC;
```

Αποτέλεσμα :

| Company | OrderNumber |
|----------------|--------------------|
| Sony | 2312 |
| Sony | 6798 |
| Digital Shop | 3412 |
| ABC Shop | 5678 |

5.6.8 Η Εντολή INSERT INTO

Η εντολή INSERT INTO εισάγει νέες γραμμές σε έναν πίνακα. Η σύνταξή της είναι ως εξής :

INSERT INTO όνομα_πίνακα

VALUES (τιμή1, τιμή2, ...);

Μπορούμε επίσης να καθορίσουμε τις στήλες για τις οποίες θέλουμε να εισάγουμε δεδομένα :

INSERT INTO όνομα_πίνακα(στήλη1, στήλη2, ...)

VALUES (τιμή1, τιμή2, ...);

Ο επόμενος πίνακας «Personal_Data» :

| Last_name | First_name | Address | City |
|------------------|-------------------|----------------|-------------|
| Σουμπάση | Μαρία | Καλλέργη 15 | Πάτρα |

και αυτή η εντολή SQL :

INSERT INTO Personal_Data

VALUES («Σιάμκουρης», «Ιωάννης», «Π. Μελά 90»,
«Καστοριά»);

δίνουν αυτό το αποτέλεσμα :

| Last_name | First_name | Address | City |
|------------------|-------------------|----------------|-------------|
|------------------|-------------------|----------------|-------------|

| | | | |
|------------|---------|-------------|----------|
| Σουμπάση | Μαρία | Καλλέργη 15 | Πάτρα |
| Σιάμκουρης | Ιωάννης | Π. Μελά 90 | Καστοριά |

5.6.9 Η Εντολή Update

Η εντολή UPDATE ενημερώνει ή αλλάζει γραμμές. Η σύνταξή της είναι ως εξής :

UPDATE όνομα_πίνακα **SET** όνομα_στήλης=νέα_τιμή

WHERE όνομα_στήλης=τιμή ;

Αρχικός πίνακας Person :

| Last_name | First_name | Address | City |
|------------|------------|----------------|----------|
| Σουμπάση | Μαρία | Καλλέργη 15 | Πάτρα |
| Σιάμκουρης | Ιωάννης | Π. Μελά 90 | Καστοριά |
| Νικολάου | | Ταγμ. Ναούμ 30 | |

Ενημέρωση μίας Στήλης σε μια Γραμμή

Θέλουμε να προσθέσουμε ένα όνομα στο άτομο που έχει το επώνυμο «Νικολάου» :

UPDATE Person SET First_name = «Αθηνά»

WHERE Last_name = «Νικολάου»;

5.6.10 Η Εντολή Delete

Η εντολή DELETE χρησιμοποιείται για να διαγράψουμε γραμμές από έναν πίνακα. Η σύνταξή της είναι ως εξής :

DELETE FROM όνομα_πίνακα

WHERE όνομα_στήλης = τιμή;

Αρχικός πίνακας Person :

| Last_name | First_name | Address | City |
|------------|------------|--------------|----------|
| Σουμπάση | Μαρία | Καλλέργη 15 | Πάτρα |
| Σιάμκουρης | Ιωάννης | Π. Μελά 90 | Καστοριά |
| Νικολάου | Αθηνά | Μεγαρόβου 12 | Πάτρα |

Διαγραφή μιας Γραμμής

Θα διαγράψουμε την «Νικολάου Αθηνά» :

```
DELETE FROM Person WHERE Last_name = «Νικολάου»;
```

Αποτέλεσμα :

| Last_name | First_name | Address | City |
|------------|------------|-------------|----------|
| Σουμπάση | Μαρία | Καλλέργη 15 | Πάτρα |
| Σιάμκουρης | Ιωάννης | Π. Μελά 90 | Καστοριά |

5.6.11 Οι Συναρτήσεις Count της SQL

Η SQL έχει ενσωματωμένες συναρτήσεις για τη μέτρηση (counting) των εγγραφών μιας βάσης δεδομένων.

Η σύνταξη για τις ενσωματωμένες συναρτήσεις COUNT είναι η εξής :

SELECT COUNT(στήλη) FROM πίνακας;

Η Συνάρτηση COUNT(*)

Η συνάρτηση COUNT(*) επιστρέφει τον αριθμό των επιλεγμένων γραμμών από μια επιλογή (selection).

Μ» αυτόν τον πίνακα «Personal_Data» :

| Name | Age |
|----------------------|------------|
| Πολυδωρόπουλος Ηλίας | 34 |
| Μάρκου Ιωάννης | 45 |
| Ιωάννου Γεώργιος | 19 |

Το επόμενο παράδειγμα επιστρέφει τον αριθμό των γραμμών του πίνακα :

SELECT COUNT(*) FROM Personal_Data;

Αποτέλεσμα : 3

Η Συνάρτηση COUNT(column)

Η συνάρτηση COUNT(column) επιστρέφει τον αριθμό των γραμμών χωρίς τιμή NULL στη συγκεκριμένη στήλη.

| Name | Age |
|----------------------|------------|
| Πολυδωρόπουλος Ηλίας | 34 |

| | |
|------------------|----|
| Μάρκου Ιωάννης | 45 |
| Ιωάννου Γεώργιος | |

Το επόμενο παράδειγμα βρίσκει τον αριθμό των ατόμων που έχουν τιμή στο πεδίο «Age» του πίνακα «Personal_Data» :

```
SELECT COUNT(Age) FROM Personal_Data;
```

Αποτέλεσμα : 2

Η συνάρτηση COUNT(column) είναι χρήσιμη για να βρίσκουμε τις στήλες που δεν έχουν τιμή. Το αποτέλεσμα είναι κατά ένα λιγότερο από τον αριθμό των γραμμών του αρχικού πίνακα επειδή ένα από τα άτομα δεν έχει τιμή στο πεδίο age.

5.6.12 Οι Λέξεις Κλειδιά COUNT και DISTINCT

Οι λέξεις κλειδιά DISTINCT και COUNT μπορούν να χρησιμοποιηθούν μαζί για να μετρήσουμε τον αριθμό των διακριτών αποτελεσμάτων (distinct results).

Η σύνταξη είναι ως εξής :

```
SELECT DISTINCT COUNT(στήλες) FROM πίνακας;
```

Με τον επόμενο πίνακα «Orders» :

| Company | OrderNumber |
|---------|-------------|
| Hitachi | 3412 |
| Sony | 2312 |
| ABC | 4678 |
| Sony | 6798 |

Η επόμενη εντολή SQL :

```
SELECT COUNT(Company) FROM Orders;
```

θα δώσει αυτό το αποτέλεσμα : 4

Η επόμενη εντολή SQL :

```
SELECT DISTINCT COUNT(Company) FROM Orders;
```

θα δώσει αυτό το αποτέλεσμα : 3

5.6.13 Οι Συναρτήσεις της SQL

Η SQL έχει πολλές ενσωματωμένες συναρτήσεις για να μπορούμε να κάνουμε μετρήσεις (counting) και υπολογισμούς (calculations).

Η γενική σύνταξη για τις ενσωματωμένες συναρτήσεις της SQL είναι η εξής :

```
SELECT function(στήλη) FROM πίνακας;
```

Ο αρχικός πίνακας :

| Name | Age |
|----------------------|-----|
| Πολυδωρόπουλος Ηλίας | 34 |
| Μάρκου Ιωάννης | 45 |
| Ιωάννου Γεώργιος | 19 |

5.6.14 Η Συνάρτηση AVG(column)

Η συνάρτηση AVG επιστρέφει τη μέση τιμή μιας στήλης σε μια επιλογή. Οι τιμές NULL δεν περιλαμβάνονται στους υπολογισμούς.

Το επόμενο παράδειγμα επιστρέφει τον μέσο όρο ηλικίας των ατόμων που υπάρχουν στον πίνακα «Personal_Data» :

```
SELECT AVG(Age) FROM Personal_Data;
```

Αποτέλεσμα : 32.67

Το επόμενο παράδειγμα επιστρέφει τον μέσο όρο ηλικίας των ατόμων που είναι πάνω από 20 χρονών :

```
SELECT AVG(Age) FROM Personal_Data WHERE Age>20;  Αποτέλεσμα : 39.5
```

5.6.15 Η Συνάρτηση MAX(column)

Η συνάρτηση MAX επιστρέφει την μεγαλύτερη τιμή μιας στήλης. Οι τιμές NULL δεν περιλαμβάνονται στον υπολογισμό.

5.6.16 Η Συνάρτηση MIN(column)

Η συνάρτηση MIN επιστρέφει την μικρότερη τιμή μιας στήλης. Οι τιμές NULL δεν περιλαμβάνονται στον υπολογισμό.

Οι συναρτήσεις MIN και MAX μπορούν επίσης να χρησιμοποιηθούν σε στήλες που περιέχουν κείμενο (text), για να βρούμε την μεγαλύτερη ή μικρότερη τιμή σε αλφαβητική σειρά.

5.6.17 Η Συνάρτηση SUM(column)

Η συνάρτηση SUM επιστρέφει το άθροισμα μιας στήλης για μια συγκεκριμένη επιλογή (selection). Οι τιμές NULL δεν περιλαμβάνονται στον υπολογισμό.

Παράδειγμα:

Το παρακάτω παράδειγμα επιστρέφει το άθροισμα των ηλικιών του πίνακα «person» για τα άτομα που είναι πάνω από 20 χρονών :

```
SELECT SUM(Age) FROM Personal_Data where Age>20;
```

Αποτέλεσμα : 79

5.6.18 Η Λέξη Κλειδί Group By

Η λέξη κλειδί GROUP BY έχει προστεθεί στην SQL επειδή οι αθροιστικές συναρτήσεις (aggregate functions), όπως είναι η SUM, επιστρέφουν το σύνολο όλων των τιμών μιας στήλης κάθε φορά που καλούνται.

Χωρίς την λέξη κλειδί GROUP BY, το να βρούμε το άθροισμα για κάθε ανεξάρτητη ομάδα τιμών μιας στήλης θα ήταν αδύνατο.

Η σύνταξη της GROUP BY είναι η εξής :

SELECT στήλη, **SUM**(στήλη) **FROM** πίνακας **GROUP BY** στήλη;

Παράδειγμα με GROUP BY:

Ο παρακάτω πίνακας «Sales» :

| Company | Amount |
|---------|--------|
| Grundig | 5500 |
| IBM | 4500 |
| Grundig | 7100 |

και αυτή η εντολή SQL :

SELECT Company, **SUM**(Amount) **FROM** Sales;

θα δώσουν αυτό το αποτέλεσμα :

| Company | SUM(Amount) |
|---------|-------------|
| Grundig | 17100 |
| IBM | 17100 |
| Grundig | 17100 |

Ο παραπάνω κώδικας δεν είναι έγκυρος επειδή η στήλη που επιστρέφει δεν αποτελεί μέρος ενός αθροίσματος (aggregate). Ένα GROUP BY clause μπορεί να το διορθώσει αυτό, ως εξής :

```
SELECT Company, SUM(Amount) FROM Sales GROUP BY Company;
```

και θα δώσει αυτό το αποτέλεσμα :

| Company | SUM(Amount) |
|---------|-------------|
| Grundig | 12600 |
| IBM | 4500 |

5.7 Ένωση Πινάκων (Join)

Μερικές φορές πρέπει να επιλέξουμε δεδομένα από δύο πίνακες για να δημιουργήσουμε ένα πιο πολύπλοκο αποτέλεσμα. Θα πρέπει να κάνουμε μια **ένωση** (join).

Οι πίνακες μιας βάσης δεδομένων μπορούν να συσχετιστούν μεταξύ τους με **κλειδιά** (keys). Ένα **πρωτεύον κλειδί** (primary key) είναι μια στήλη με μια μοναδική τιμή στην κάθε γραμμή. Ο σκοπός είναι να ενώσει τα δεδομένα μαζί από διάφορους πίνακες, χωρίς να έχουμε επανάληψη όλων των δεδομένων σε κάθε πίνακα.

Στον πίνακα «Employees» παρακάτω, η στήλη «ID» είναι το πρωτεύον κλειδί (primary key), που σημαίνει ότι δεν μπορούν να υπάρχουν δύο γραμμές που να έχουν το ίδιο ID. Το ID είναι αυτό που ξεχωρίζει δύο άτομα ακόμη κι αν έχουν το ίδιο όνομα.

Στους πίνακες παραδειγμάτων παρακάτω, προσέχουμε τα εξής :

- Η στήλη «ID» είναι το πρωτεύον κλειδί του πίνακα «Employees».

- Η στήλη «ID» του πίνακα «Orders» χρησιμοποιείται για να αναφερόμαστε στα άτομα του πίνακα «Employees», χωρίς να χρησιμοποιούμε τα ονόματά τους.

Πίνακας Employees :

| ID | Name |
|-----------|---------------------|
| 01 | Νικολάου Αθηνά |
| 02 | Γεωργιάδης Ηλίας |
| 03 | Καράμπελας Στέφανος |
| 04 | Σουμπάσης Ιωάννης |

Πίνακας Orders :

| ID | Product |
|-----------|----------------|
| 01 | Printer |
| 03 | Computer |
| 03 | Scanner |

Παράδειγμα

Ποιοι έχουν παραγγείλει προϊόντα και τι έχουν παραγγείλει;

```
SELECT Employees.Name, Orders.Product FROM Employees, Orders
```

```
WHERE Employees.ID = Orders.ID;
```

Αποτέλεσμα :

| Name | Product |
|---------------------|----------|
| Νικολάου Αθηνά | Printer |
| Καράμπελας Στέφανος | Computer |
| Καράμπελας Στέφανος | Scanner |

Παράδειγμα:

Ποιοι έχουν παραγγείλει εκτυπωτή (printer);

```
SELECT Employees.Name FROM Employees, Orders
WHERE Employees.ID =
Orders.ID AND
Orders.Product = «Printer»;
```

Αποτέλεσμα :

| Name |
|----------------|
| Νικολάου Αθηνά |

5.8 Δημιουργία Βάσης Δεδομένων και Πίνακα

Για να δημιουργήσουμε μια βάση δεδομένων, μπορούμε να δώσουμε την εξής εντολή :

```
CREATE DATABASE όνομα_βάσης_δεδομένων;
```

Για να δημιουργήσουμε έναν πίνακα σε μια βάση δεδομένων, μπορούμε να δώσουμε την εξής εντολή :

CREATE TABLE όνομα_πίνακα

(όνομα_στήλης_1 τύπος_δεδομένων, όνομα_στήλης_2 τύπος_δεδομένων,...);

Παράδειγμα:

Θα δημιουργήσουμε έναν πίνακα με όνομα «Person», με τέσσερις στήλες με ονόματα «Last_name», «First_name», «Address» και «Age» :

CREATE TABLE Person

(Last_name varchar, First_name varchar, Address varchar, Age number);

Θα ορίσουμε ένα μέγιστο μήκος για μερικές στήλες :

CREATE TABLE Person

(Last_name varchar(30), First_name varchar, Address varchar, Age number(3));

Ο τύπος δεδομένων (data type) καθορίζει τι είδος δεδομένων θα περιέχει η στήλη. Ο παρακάτω πίνακας περιέχει τους πιο συνηθισμένους τύπους δεδομένων της SQL :

| Τύπος Δεδομένων
(Data Type) | Περιγραφή (Description) |
|--|---|
| char(size) | Περιέχει ένα string σταθερού μήκους που μπορεί να περιέχει γράμματα, αριθμούς και ειδικούς χαρακτήρες. Το σταθερό μέγεθος καθορίζεται στις παρενθέσεις. |
| varchar(size) | Περιέχει ένα string μεταβλητού μήκους που μπορεί να περιέχει γράμματα, αριθμούς και ειδικούς χαρακτήρες. Το μέγιστο μέγεθος καθορίζεται στις παρενθέσεις. |
| number(size) | Περιέχει έναν αριθμό, όπου ο μέγιστος αριθμός των ψηφίων καθορίζεται στις παρενθέσεις. |
| number(size, d) | Περιέχει έναν αριθμό, όπου ο μέγιστος αριθμός των ψηφίων |

| | |
|------|--|
| | καθορίζεται στο size και ο μέγιστος αριθμός των ψηφίων στα δεξιά της υποδιαστολής καθορίζεται στο d. |
| date | Περιέχει μια ημερομηνία. |

5.9 Διαγραφή Βάσης Δεδομένων και Πίνακα

Για να διαγράψουμε μια βάση δεδομένων, χρησιμοποιούμε την εξής εντολή :

DROP DATABASE όνομα_βάσης_δεδομένων;

Για να διαγράψουμε έναν πίνακα, χρησιμοποιούμε την εξής εντολή :

DROP TABLE όνομα_πίνακα;

Για να διαγράψουμε τα δεδομένα ενός πίνακα χωρίς να διαγράψουμε τον πίνακα, χρησιμοποιούμε την εξής εντολή :

DELETE TABLE όνομα_πίνακα;

5.10 Η Εντολή Alter Table

Η εντολή ALTER TABLE χρησιμοποιείται για να προσθέσουμε ή να διαγράψουμε στήλες από έναν υπάρχοντα πίνακα.

Η σύνταξη της για τις δύο αυτές περιπτώσεις είναι ως εξής :

ALTER TABLE όνομα_πίνακα **ADD** όνομα_στήλης τύπος_δεδομένων;

ALTER TABLE όνομα_πίνακα **DROP** όνομα_στήλης;

ΚΕΦΑΛΑΙΟ 6^ο

ΣΥΝΤΟΜΗ ΑΝΑΛΥΣΗ ΤΩΝ ΤΕΧΝΟΛΟΓΙΩΝ ΤΟΥ ΗΛΕΚΤΡΟΝΙΚΟΥ ΚΑΤΑΣΤΗΜΑΤΟΣ

Όπως αναφέραμε και παραπάνω η εργασία μας στηρίχθηκε κυρίως στην εφαρμογή των τεχνολογιών PHP, MYSQL. Η PHP, της οποίας τα αρχικά αντιπροσωπεύουν το "PHP: Hypertext Preprocessor" είναι μια ευρέως χρησιμοποιούμενη, ανοιχτού κώδικα, γενικού σκοπού scripting γλώσσα προγραμματισμού, η οποία είναι ειδικά κατάλληλη για ανάπτυξη εφαρμογών για το Web και μπορεί να ενσωματωθεί στην HTML.

Αυτό που είναι διαφορετικό από ένα script γραμμένο σε άλλες γλώσσες προγραμματισμού όπως η Perl ή η C : Αντί να γράφετε ένα πρόγραμμα με πολλές εντολές για να εξάγετε HTML, γράφετε ένα HTML script με κάποιο ενσωματωμένο

κώδικα για να κάνει κάτι (σε αυτή την περίπτωση, να εμφανίζει κάποιο κείμενο). Ο κώδικας PHP είναι εσώκλειστος σε ειδικά tags (ετικέτες) αρχής και τέλους που σας επιτρέπουν να μεταφέρεστε μέσα και έξω από το "PHP mode" (PHP τρόπο λειτουργίας).

Αυτό που διαχωρίζει την PHP από κάτι σαν client-side Javascript είναι ότι ο κώδικας εκτελείται στον server (εξηγηρητή). Αν είχατε ένα script σαν το παραπάνω στον server σας, ο client θα έπαιρνε τα αποτελέσματα της εκτέλεσης αυτού του script, χωρίς να υπάρχει κανένας τρόπος να καταλάβει τι κώδικας υπάρχει από κάτω. Μπορείτε ακόμη να ρυθμίσετε τον web server σας να χειρίζεται όλα τα HTML αρχεία σας με την PHP, και τότε πραγματικά δεν υπάρχει τρόπος ο χρήστης να καταλάβει τι έχετε κάτω από το μανίκι σας.

Τα καλύτερο στην PHP είναι ότι είναι εξαιρετικά απλή αλλά προσφέρει πολλά προηγμένα χαρακτηριστικά για ένα επαγγελματία προγραμματιστή.

Η PHP επικεντρώνεται κυρίως στο server-side scripting, έτσι μπορείτε να κάνετε οτιδήποτε ένα άλλο CGI πρόγραμμα μπορεί να κάνει, όπως να μαζέψει δεδομένα, να παράγει δυναμικό περιεχόμενο σελίδων, ή να στείλει και να πάρει cookies. Η PHP μπορεί να κάνει πολύ περισσότερα.

Υπάρχουν τρεις κύριοι τομείς που χρησιμοποιείται ένα PHP script.

Server-side scripting. Αυτό είναι το πιο παραδοσιακό και το κύριο πεδίο για την PHP. Χρειάζεστε τρία πράγματα για να δουλέψει. Τον PHP μεταγλωττιστή (parser) (CGI ή server module), ένα webserver (εξηγηρητή σελίδων) και ένα web browser ("φυλλομετρητή"). Πρέπει να τρέξετε τον webserver, με μια συνδεδεμένη εγκατάσταση της PHP. Μπορείτε να προσπελάσετε τα αποτελέσματα του PHP προγράμματος με ένα web browser, βλέποντας την σελίδα PHP μέσα από τον server

Command line scripting. Μπορείτε να φτιάξετε ένα PHP script για να το τρέχετε χωρίς server ή browser. Χρειάζεστε μόνο τον PHP μεταγλωττιστή για να την χρησιμοποιήσετε με αυτό τον τρόπο. Αυτός ο τύπος είναι ιδανικός για script που εκτελούνται συχνά με τη χρήση της cron (σε *nix ή Linux) ή με τον Task Scheduler (στα Windows). Αυτά τα script μπορούν επίσης να χρησιμοποιηθούν για απλές εργασίες επεξεργασίας κειμένου.

Εγγραφή client-side GUI εφαρμογών (Γραφικά περιβάλλοντα χρηστών). Η PHP ίσως να μην είναι η πιο καλή γλώσσα για να γράψει κανείς παραθυριακές εφαρμογές, αλλά αν ξέρετε PHP πολύ καλά και θέλετε να χρησιμοποιήσετε κάποια προχωρημένα χαρακτηριστικά της PHP στις client-side εφαρμογές σας, μπορείτε επίσης να χρησιμοποιήσετε το PHP-GTK για αυτού του είδους τα προγράμματα. Έχετε επίσης τη δυνατότητα να γράφετε cross-platform εφαρμογές με αυτό τον τρόπο. Το PHP-GTK είναι μια επέκταση της PHP και δεν συμπεριλαμβάνεται στην κύρια διανομή.

Βασικό πλεονέκτημα της PHP είναι ότι, μπορεί να χρησιμοποιηθεί σε όλα τα κύρια λειτουργικά συστήματα, συμπεριλαμβανομένου του Linux, πολλών εκδοχών του Unix (HP-UX, Solaris και OpenBSD), Microsoft Windows, Mac OS X, RISC OS και πιθανώς σε άλλα. Η PHP υποστηρίζει επίσης τους Apache, Microsoft Internet Information Server, Personal Web Server, Netscape και iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, Sambar και πολλούς άλλους webserver. Για την πλειοψηφία των server η PHP έχει ένα module, για τους υπόλοιπους η PHP μπορεί να λειτουργήσει ως ένας CGI επεξεργαστής.

Έτσι με την PHP έχουμε την ελευθερία επιλογής ενός λειτουργικού συστήματος και ενός web server. Επιπλέον, έχουμε επίσης την ελευθερία να χρησιμοποιήσουμε συναρτησιακό (procedural) ή αντικειμενοστρεφή (object oriented) προγραμματισμό ή μια ανάμειξη τους. Αν και η παρούσα έκδοση δεν υποστηρίζει όλα τα πρότυπα χαρακτηριστικά, μεγάλες βιβλιοθήκες κώδικα και μεγάλες εφαρμογές (συμπεριλαμβανομένης και της βιβλιοθήκης PEAR) είναι γραμμένες μόνο με αντικειμενοστρεφή κώδικα.

Με την PHP δεν είμαστε περιορισμένοι να εξάγουμε HTML. Οι δυνατότητες της PHP συμπεριλαμβάνουν την εξαγωγή εικόνων, αρχείων PDF, ακόμη και ταινίες Flash (χρησιμοποιώντας τα libswf και Ming) παράγονται αμέσως. Μπορούμε επίσης να εξάγουμε εύκολα οποιοδήποτε κείμενο όπως XHTML και οποιοδήποτε άλλο XML αρχείο. Η PHP μπορεί να δημιουργεί αυτόματα αυτά τα αρχεία και να τα αποθηκεύει στο σύστημα αρχείων, αντί να τα εκτυπώνει, αποτελώντας έτσι μια server-side cache για το δυναμικό μας περιεχόμενο.

Ένα από τα πιο δυνατά και σημαντικά χαρακτηριστικά της PHP είναι η υποστήριξη που έχει για ένα μεγάλο σύνολο βάσεων δεδομένων. Η συγγραφή μιας

σελίδας που υποστηρίζει βάσεις δεδομένων είναι εξαιρετικά απλή. Οι εξής βάσεις δεδομένων υποστηρίζονται μέχρι στιγμής:

| | | |
|---------------------|---------------|------------------------|
| Adabas D | Ingres | Oracle (OCI7 and OCI8) |
| DBase | InterBase | Ovrimos |
| Empress | FrontBase | PostgreSQL |
| FilePro (read-only) | mSQL | Solid |
| Hyperwave | Direct MS-SQL | Sybase |
| IBM DB2 | MySQL | Velocis |
| Informix | ODBC | Unix dbm |

Έχουμε επίσης μια αφαιρετική επέκταση DBX βάσεων δεδομένων (DBX database abstraction extension) που μας επιτρέπει διάφανα να χρησιμοποιήσουμε οποιαδήποτε βάση δεδομένων υποστηρίζεται από αυτή την επέκταση. Επιπλέον η PHP υποστηρίζει το ODBC, το Open Database Connection standard (Ανοιχτό πρότυπο Σύνδεσης Βάσεων δεδομένων) έτσι μπορούμε να συνδέσουμε σε οποιαδήποτε βάση δεδομένων που υποστηρίζει αυτό το παγκόσμιο πρότυπο.

Η PHP έχει επίσης υποστήριξη για επικοινωνία με άλλες υπηρεσίες χρησιμοποιώντας πρωτόκολλα όπως LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (στα Windows) και αμέτρητα άλλα. Μπορούμε επίσης να ανοίξουμε raw network sockets και να αλληλεπιδράσουμε με οποιοδήποτε άλλο πρωτόκολλο. Η PHP έχει ακόμη υποστήριξη για την περίπλοκη ανταλλαγή δεδομένων WDDX μεταξύ σχεδόν όλων των Web programming γλωσσών. Μιλώντας για δια-επικοινωνία, η PHP υποστηρίζει instantiation αντικειμένων Java και τα χρησιμοποιεί διάφανα σαν αντικείμενα PHP. Μπορείτε επίσης να χρησιμοποιήσουμε την CORBA επέκταση μας για να προσπελάσετε remote (απομακρυσμένα) αντικείμενα.

Η PHP έχει εξαιρετικά χρήσιμα χαρακτηριστικά επεξεργασίας κειμένων, από την POSIX επέξταση ή τις Perl regular expressions μέχρι XML parsing αρχείων. Για τη μεταγλώττιση και την πρόσβαση αρχείων XML, υποστηρίζουμε τα πρότυπα SAX και

DOM. Μπορούμε να χρησιμοποιήσουμε την XSLT επέκταση μας για να μετατρέψουμε τα XML αρχεία σε άλλες μορφές.

Καθώς χρησιμοποιούμε την PHP στον τομέα του ηλεκτρονικού εμπορίου, θα βρούμε τις Cybercash payment, CyberMUT, VeriSign Payflow Pro και CCVS συναρτήσεις χρήσιμες για τα online προγράμματα πληρωμής σας.

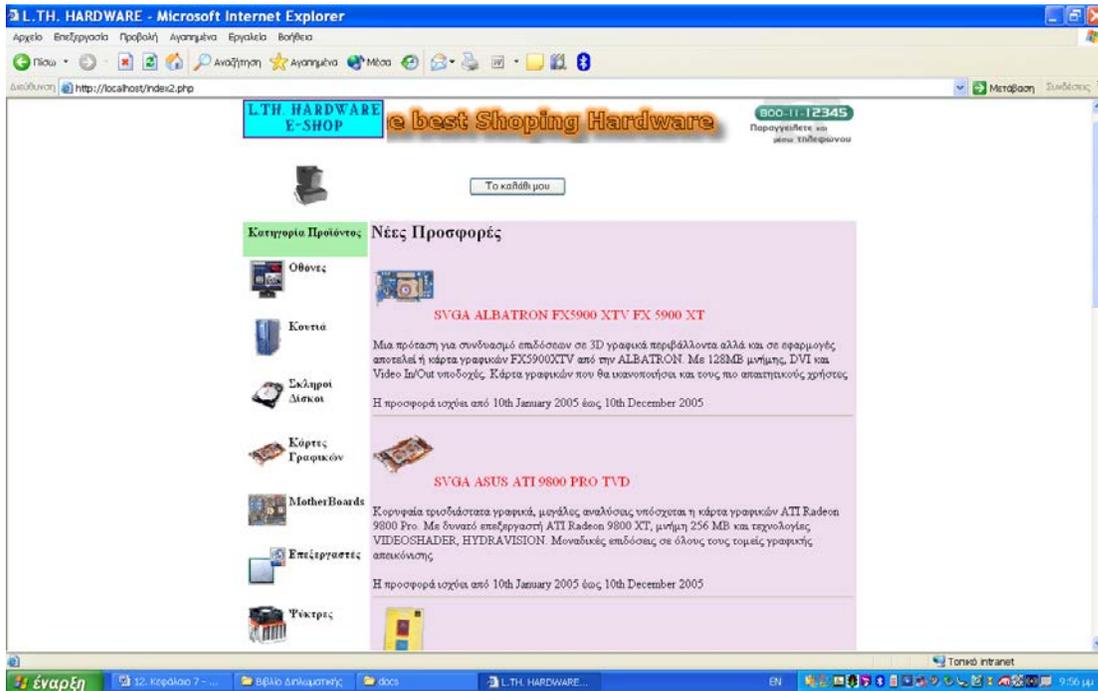
Η επικοινωνία της PHP με μία βάση δεδομένων είναι αρκετά εύκολη μια και το μόνο που χρειάζεται είναι να καλέσουμε εντολές SQL και να εκτελέσουμε στο DATABASE SERVER.

ΚΕΦΑΛΑΙΟ 7^ο

ΥΛΟΠΟΙΗΣΗ ΤΟΥ ΗΛΕΚΤΡΟΝΙΚΟΥ ΚΑΤΑΣΤΗΜΑΤΟΣ

7.1 Η σελίδα του Πελάτη

Για την εισαγωγή του στο ηλεκτρονικό κατάστημα ο πελάτης πληκτρολογεί στη διεύθυνση του φυλλομετρητή το κατάλληλο URL καλώντας την αρχική σελίδα του καταστήματος mainpage.php. Στη σελίδα mainpage.php ο υπολογιστής του CLIENT δέχεται ένα COOKIE (κωδικοποιημένο αρχείο) όπου περιέχει ένα προσωρινό κωδικό που τον χαρακτηρίζει όσο κάνει επιλογές και καταχωρήσεις στο καλάθι του. Αφού δεχτεί το COOKIE μεταβιβάζεται αυτόματα στη σελίδα index2.php όπου φαίνεται παρακάτω:

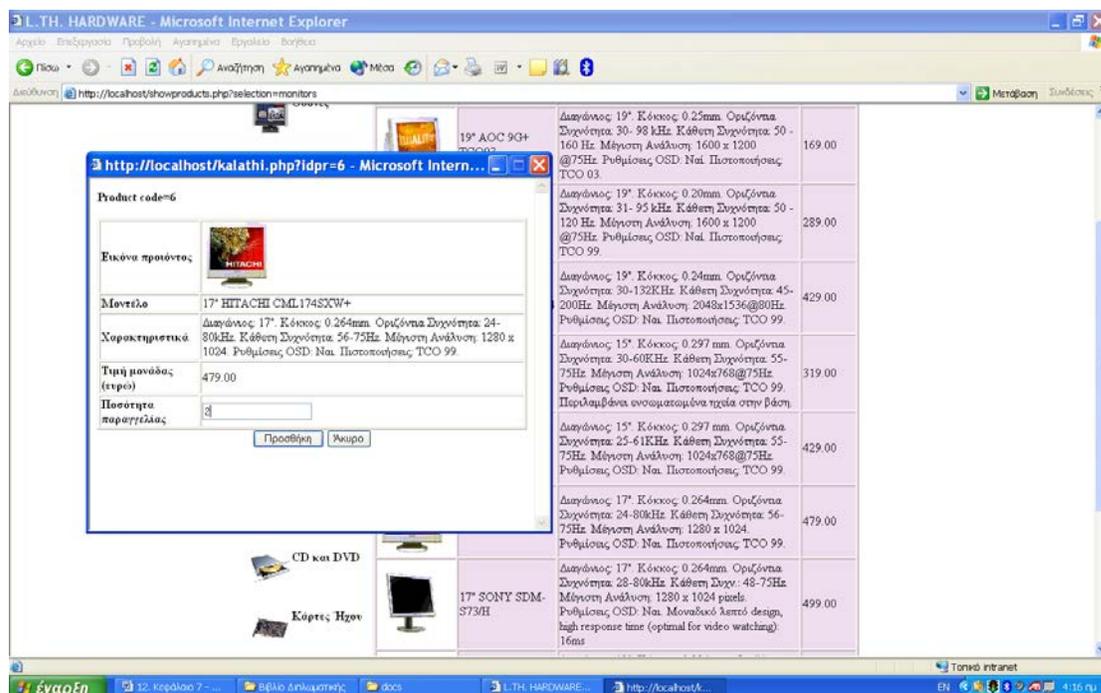


Στη σελίδα αυτή ο χρήστης βλέπει στο κεντρικό μέρος προσφορές που προτείνονται από το κατάστημα ενώ αντίστοιχα έχει την δυνατότητα να περιηγηθεί στις κατηγορίες των προϊόντων που θέλει να πάρει πληροφορίες. Επίσης μπορεί ήδη να επιλέξει να δει το τι περιέχει το καλάθι του αν και αρχικά δεν περιέχει τίποτα. Επίσης όπως συμβαίνει σε όλες τις ιστοσελίδες μπορεί πατώντας πάνω στο λογότυπο της εταιρίας να γυρίζει στην αρχική σελίδα (index2.php) του site.



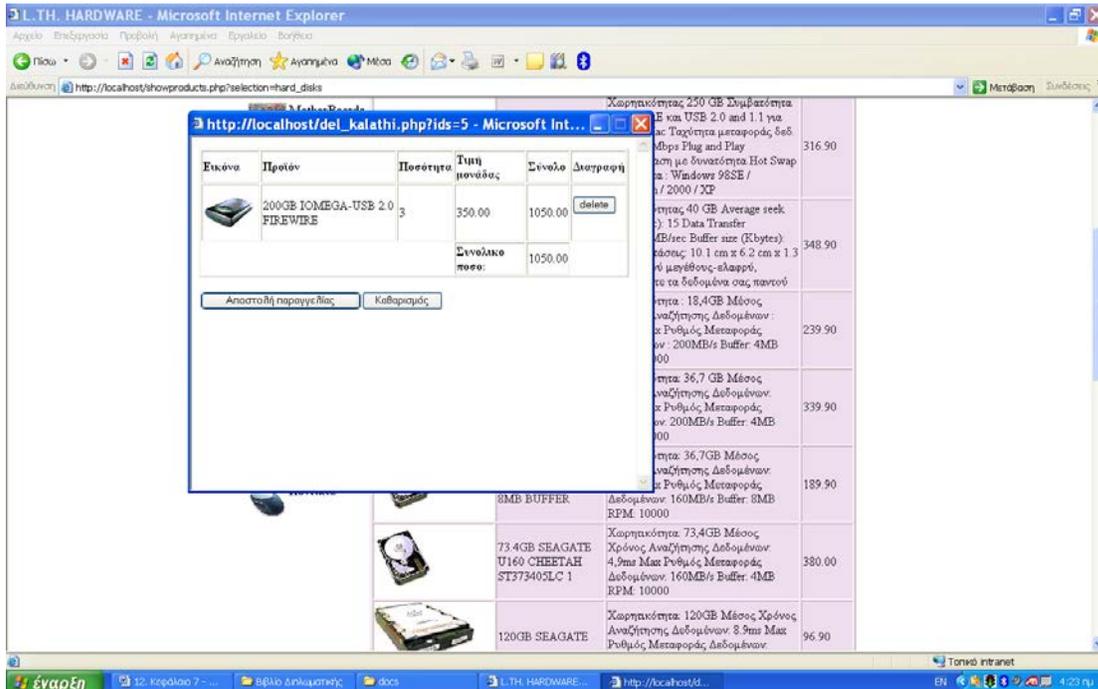
Επιλέγοντας μια κατηγορία προϊόντων ο χρήστης στην ουσία καλεί την ιστοσελίδα showproducts.php με κατάλληλη παράμετρο που να του δίνει σαν αποτέλεσμα μια επιλεγμένη κατηγορία προϊόντων. Έτσι του εμφανίζεται στο κέντρο της σελίδας πίνακας με τα προϊόντα της συγκεκριμένης κατηγορίας που υπάρχουν στη βάση.

Για το κάθε προϊόν παίρνει πλήρεις πληροφορίες με τα χαρακτηριστικά και την τιμή του. Πατώντας πάνω στο εικονίδιο του προϊόντος, καλεί τη σελίδα kalathi.php σε νέο παράθυρο με κατάλληλη παράμετρο που του εμφανίζει το προϊόν με τον αντίστοιχο κωδικό. Στη σελίδα αυτή μπορεί να προσθέσει μια ποσότητα από το προϊόν αυτό στο καλάθι του. Έτσι μέσω της ιστοσελίδας αυτής γίνεται η καταχώρηση προϊόντων στο καλάθι που αν είναι επιτυχής εμφανίζει ανάλογο μήνυμα καλώντας τη σελίδα ins_kalathi.php όπου τελικά εκεί και γίνεται η τελική καταχώρηση (εκτελείται η εντολή insert στο MYSQL).

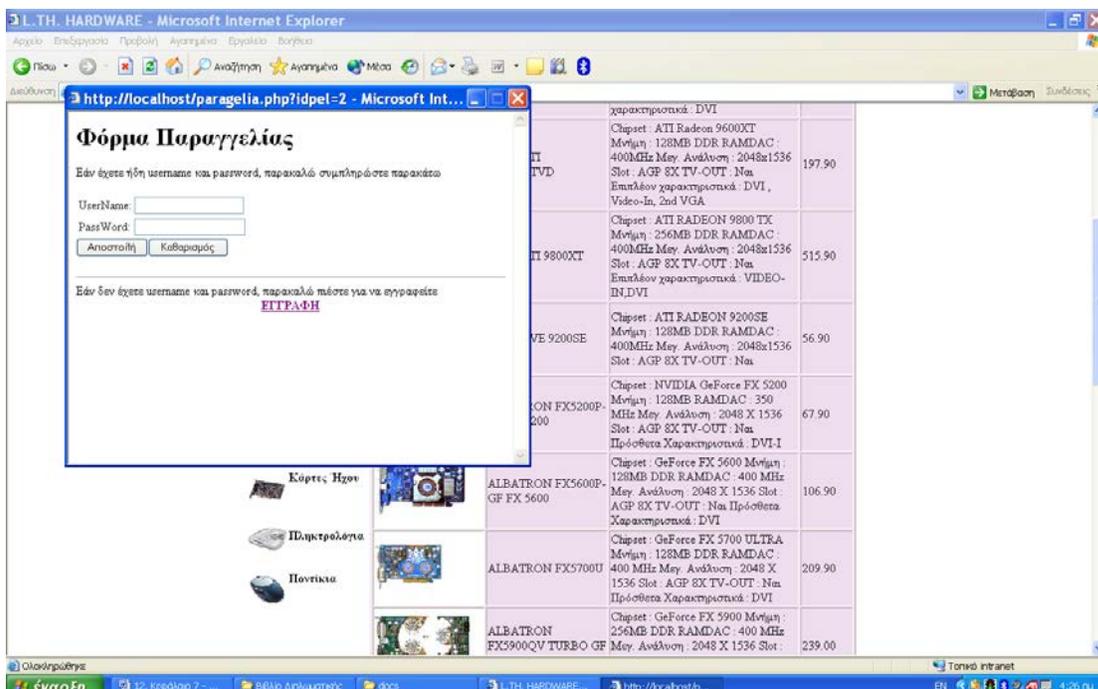


Κατά την διάρκεια της περιήγησης του ο χρήστης μπορεί να επισκέπτεται και να ελέγχει τι έχει καταχωρήσει στο καλάθι του πατώντας απλά πάνω στο κουμπί του καλαθιού στο πάνω κεντρικό μέρος της ιστοσελίδας. Τότε εμφανίζεται η ιστοσελίδα showkalathi.php όπου και δείχνει ποια προϊόντα έχουμε μέχρι εκείνη τη στιγμή επιλέξει καθώς και το συνολικό ποσό που θα μας κοστίσει η τυχόν παραγγελία μας. Επίσης

πατώντας το πλήκτρο «delete» μπορούμε να αφαιρέσουμε αντικείμενα μέσα από το καλάθι καλώντας την βοηθητική ιστοσελίδα del_kalathi.php .

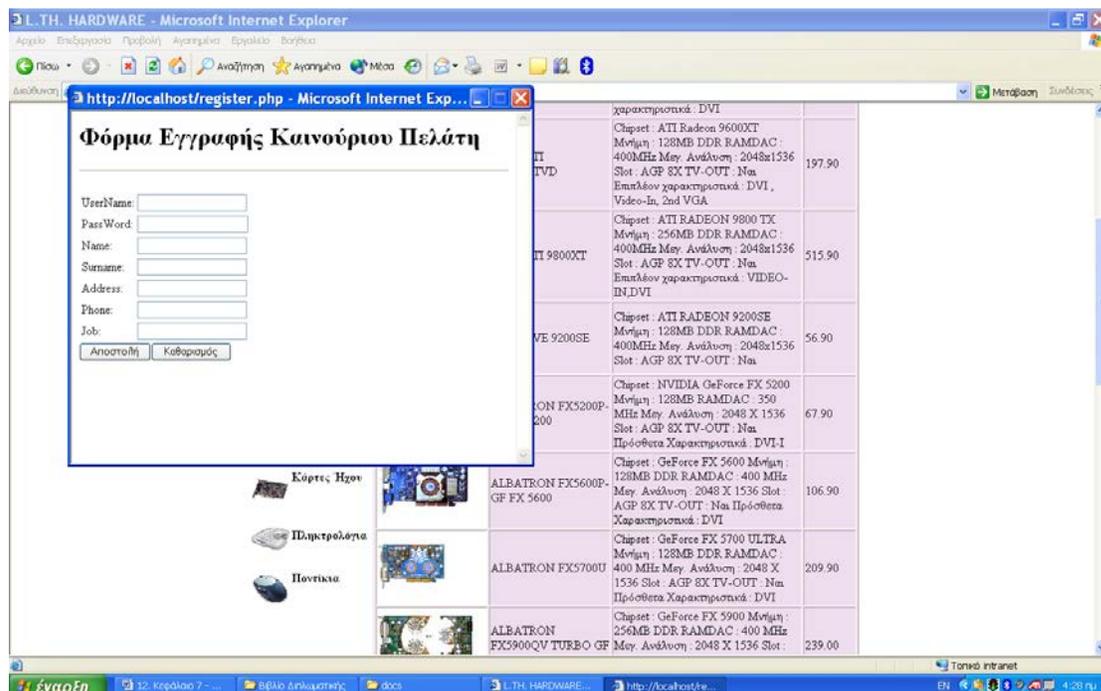


Όταν τελικώς ο χρήστης αποφασίσει να οριστικοποιήσει την παραγγελία του τότε επιλέγει από το καλάθι Αποστολή Παραγγελίας και μεταβαίνει στη φόρμα παραγγελίας δηλαδή στη σελίδα paragegia.php

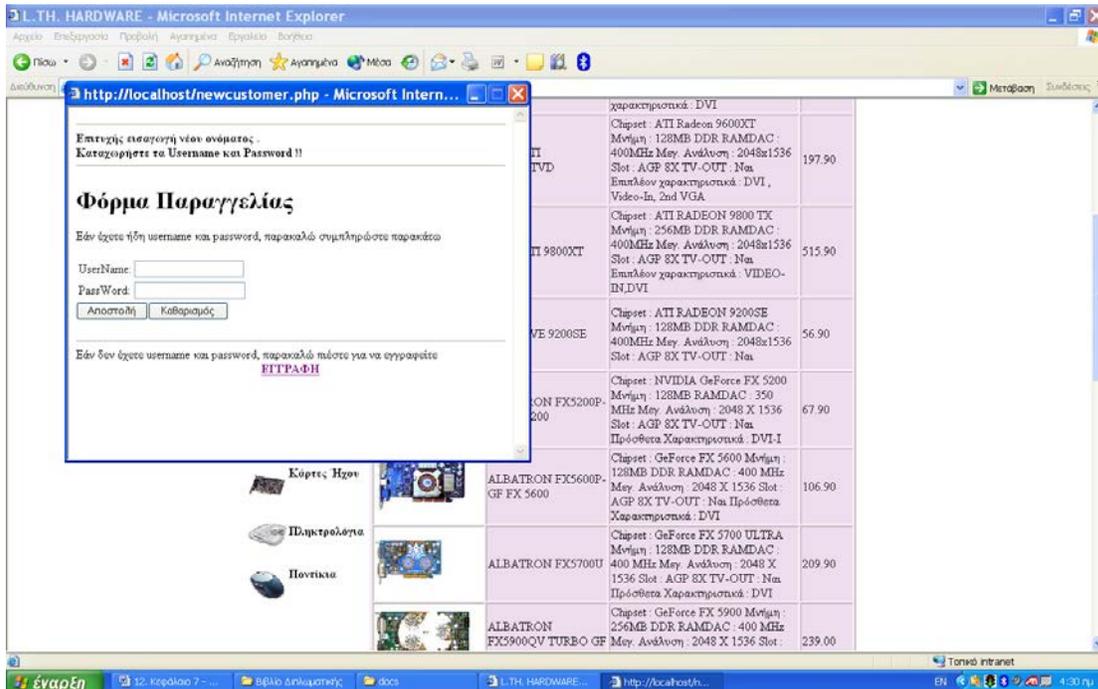


Στη φόρμα παραγγελίας ο χρήστης έχει δύο επιλογές. Αν είναι ήδη χρήστης να δώσει το Username και Password που ήδη έχει αλλιώς να κάνει εγγραφή σαν νέος πελάτης.

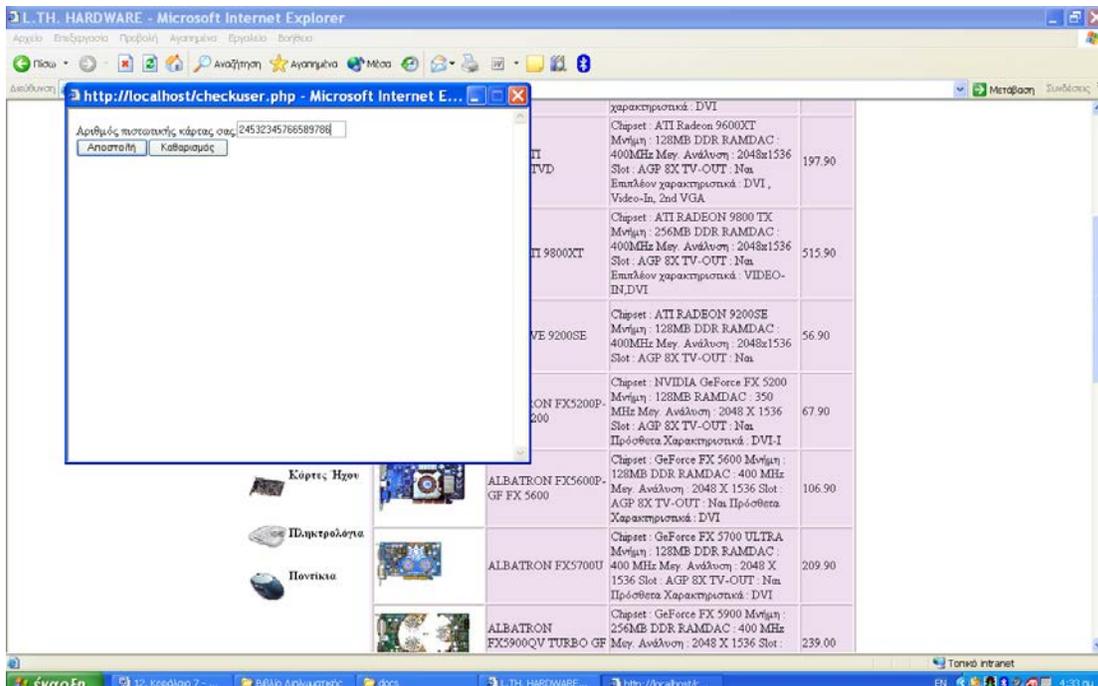
Στην περίπτωση αυτή καλεί την ιστοσελίδα register.php και συμπληρώνει τα στοιχεία που του ζητά η φόρμα. Μέσα σε αυτά είναι και το Username και Password όπου το Username πρέπει να είναι μοναδικό και να μην έχει ξαναδοθεί από κανένα άλλο χρήστη. Σε τέτοια περίπτωση ο χρήστης καλείται και πάλι να δώσει ένα νέο Username.



Καλείται η φόρμα newcustomer.php που μοιάζει με την paragelia.php όπου του εμφανίζεται κατάλληλο μήνυμα ότι η εγγραφή του στη βάση ήταν επιτυχής και τον καλεί να δώσει το νέο του Username και Password προκειμένου να οριστικοποιήσει τη παραγγελία του.

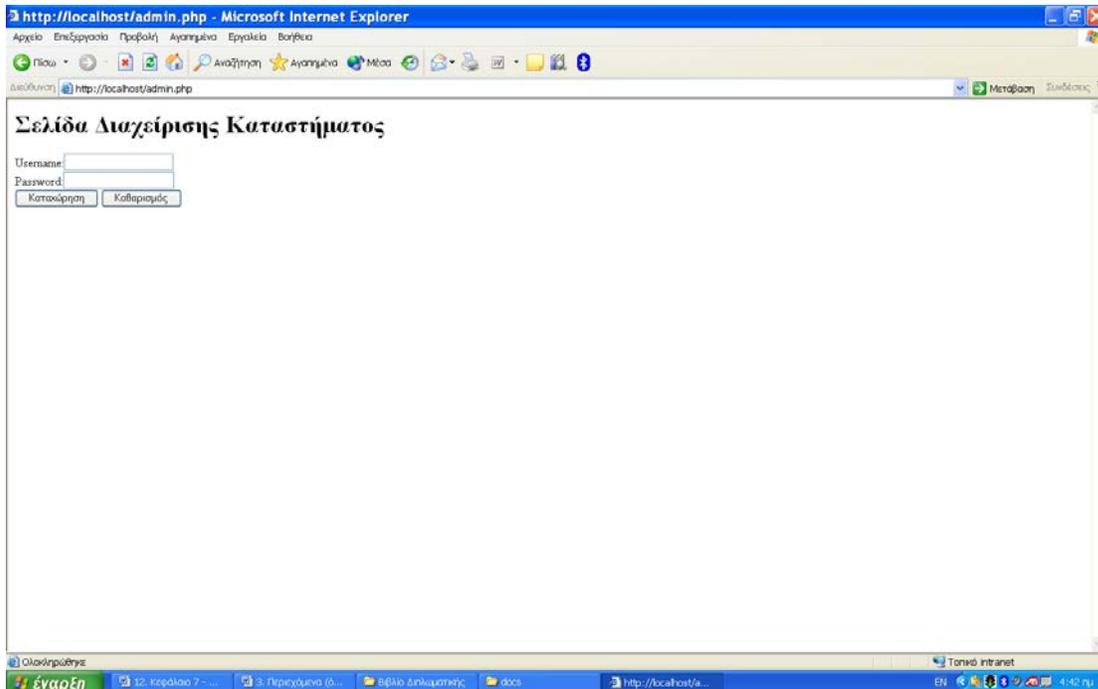


Στη συνέχεια αφού ο χρήστης δώσει τους κωδικούς του εμφανίζεται η σελίδα checkuser.php όπου μας καλεί να δώσουμε τον αριθμό της πιστωτικής κάρτας μας προκειμένου μέσω αυτής να γίνει η πληρωμή της παραγγελίας. Στο τέλος γίνεται επιβεβαίωση της παραγγελίας μας και πλέον ο πελάτης περιμένει την αποστολή των προϊόντων που παράγγειλε .

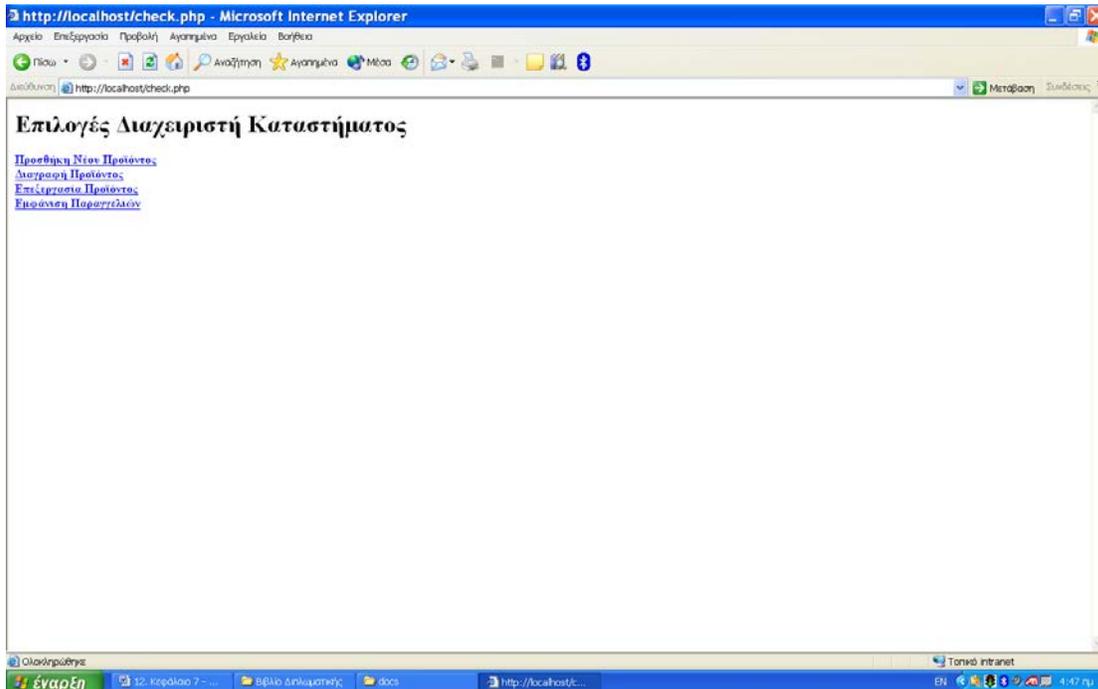


7.2 Η σελίδα του Διαχειριστή

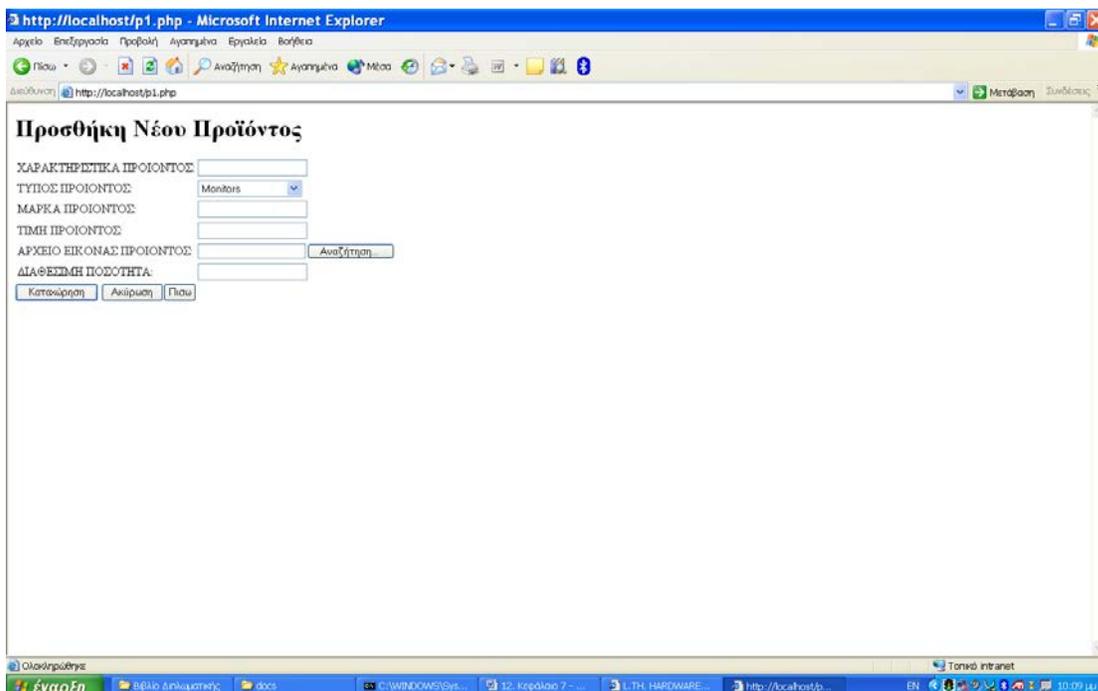
Σαν Διαχειριστής του συστήματος ορίζεται κάποιος εξουσιοδοτημένος χρήστης όπου με κατάλληλο Username και Password μπορεί να διαχειριστεί τα προϊόντα και τις παραγγελίες του καταστήματός μας. Ο διαχειριστής καλεί τη σελίδα admin.php όπου τον καλεί να δώσει το Username και Password προκειμένου να του επιτραπεί η πρόσβαση στις σελίδες διαχείρισης.



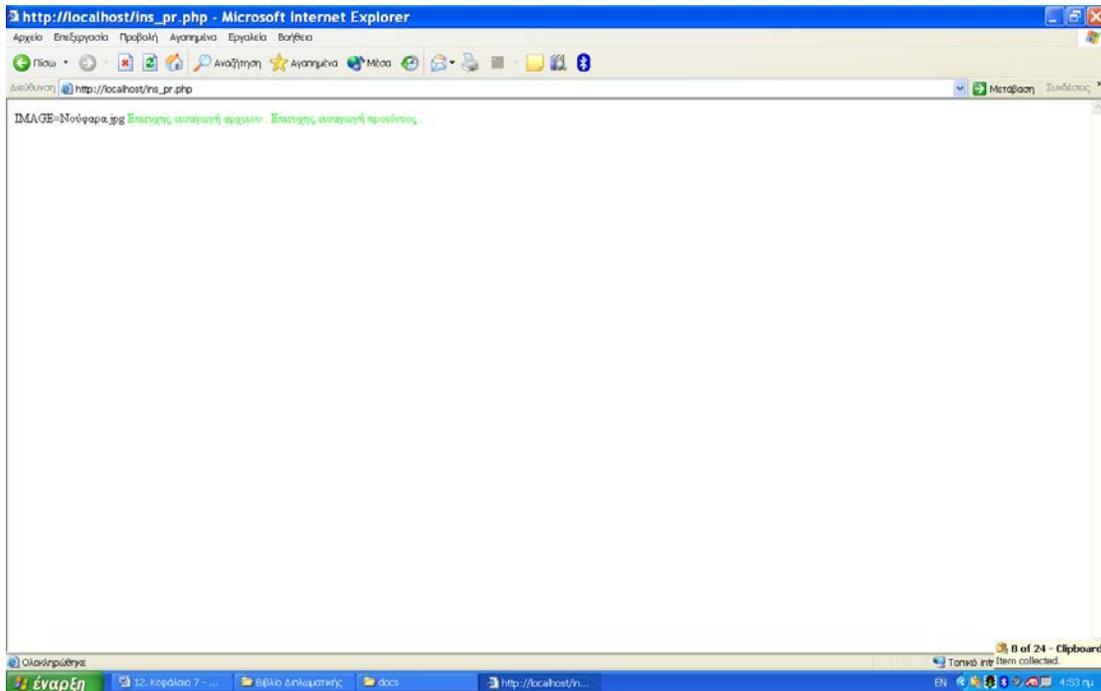
Όταν ο χρήστης δώσει σωστά τους κωδικούς του τότε εμφανίζεται η σελίδα με το βασικό μενού διαχείρισης . Η σελίδα αυτή ονομάζεται check.php . Σε αυτή ο διαχειριστής καλείται να επιλέξει από μια σειρά επιλογών διαχείρισης που είναι η εισαγωγή νέων προϊόντων στη βάση, η διαγραφή προϊόντων από τη βάση, η επεξεργασία των στοιχείων ενός προϊόντος και η εμφάνιση και διαχείριση των παραγγελιών προς διεκπεραίωση.



Κατά την εισαγωγή ενός προϊόντος στη βάση ο διαχειριστής καλείται να δώσει όλα τα απαραίτητα στοιχεία για το προϊόν και πατώντας Καταχώρηση γίνεται η εισαγωγή του προϊόντος στη βάση και εμφανίζεται κατάλληλο μήνυμα.

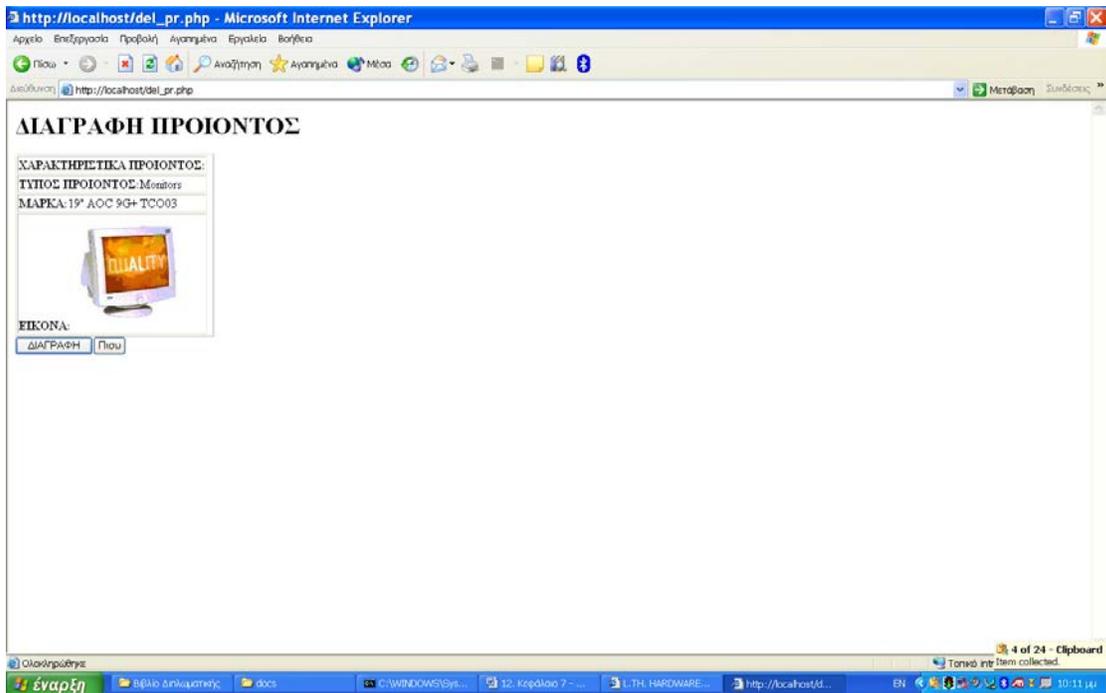
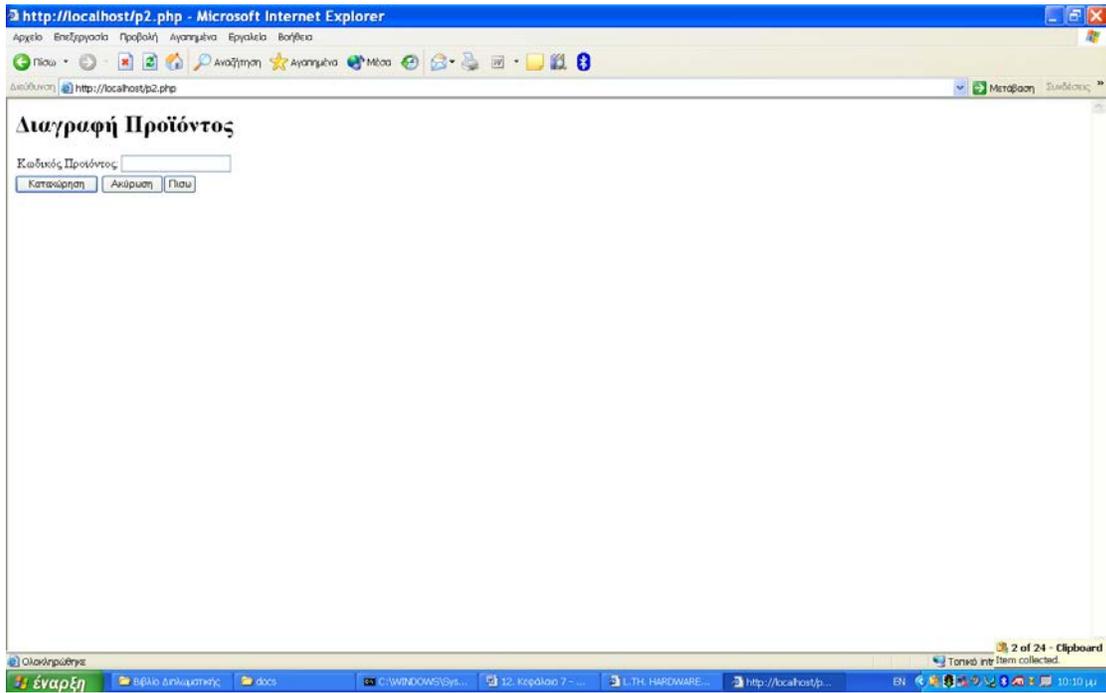


Για την εισαγωγή των προϊόντων καλούνται οι σελίδες p1.php και ins_pr.php αντίστοιχα.

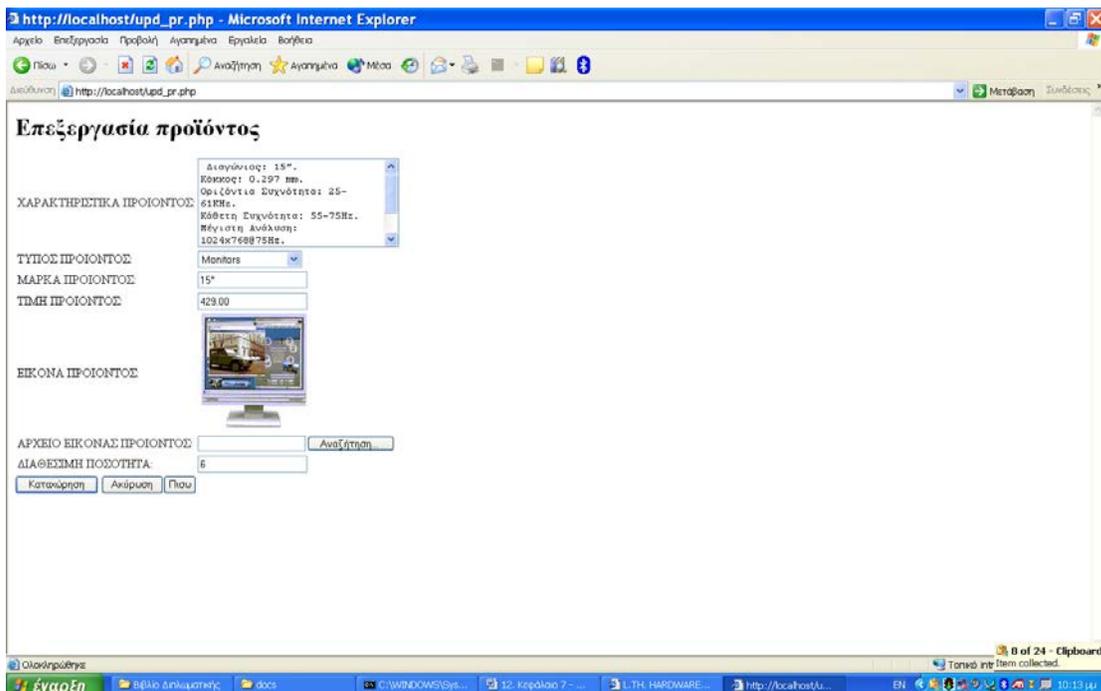
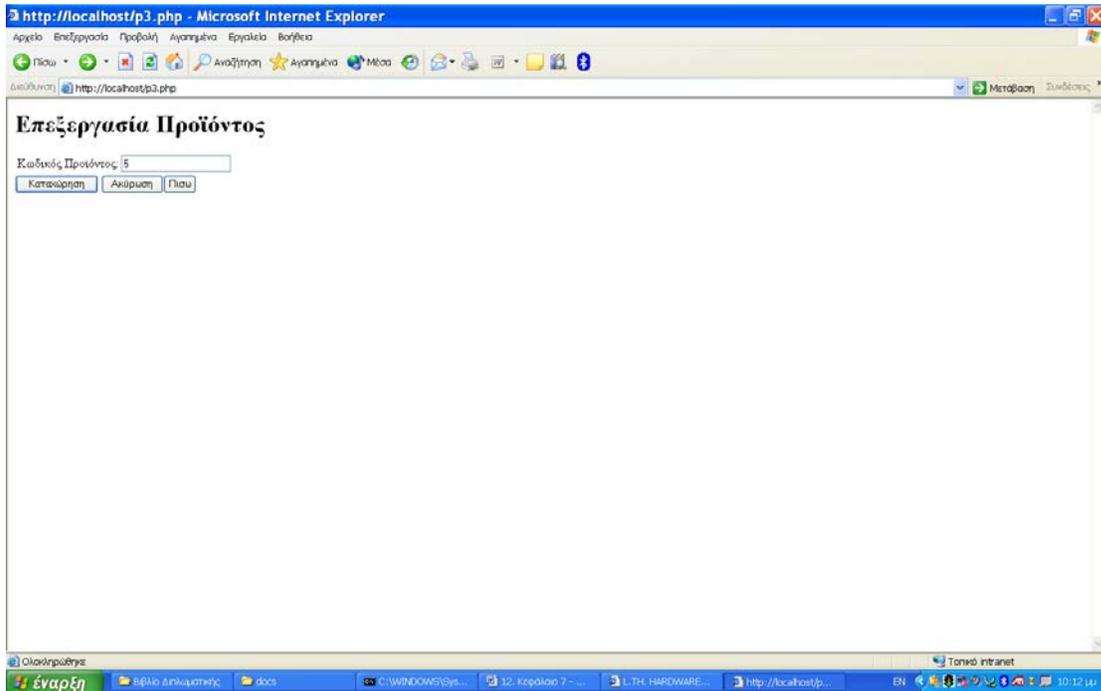


Για τη διαγραφή ενός προϊόντος από την βάση ο διαχειριστής καλεί την ιστοσελίδα p2.php όπου και δίνει σαν είσοδο τον κωδικό του προϊόντος που θέλει να διαγράψει.

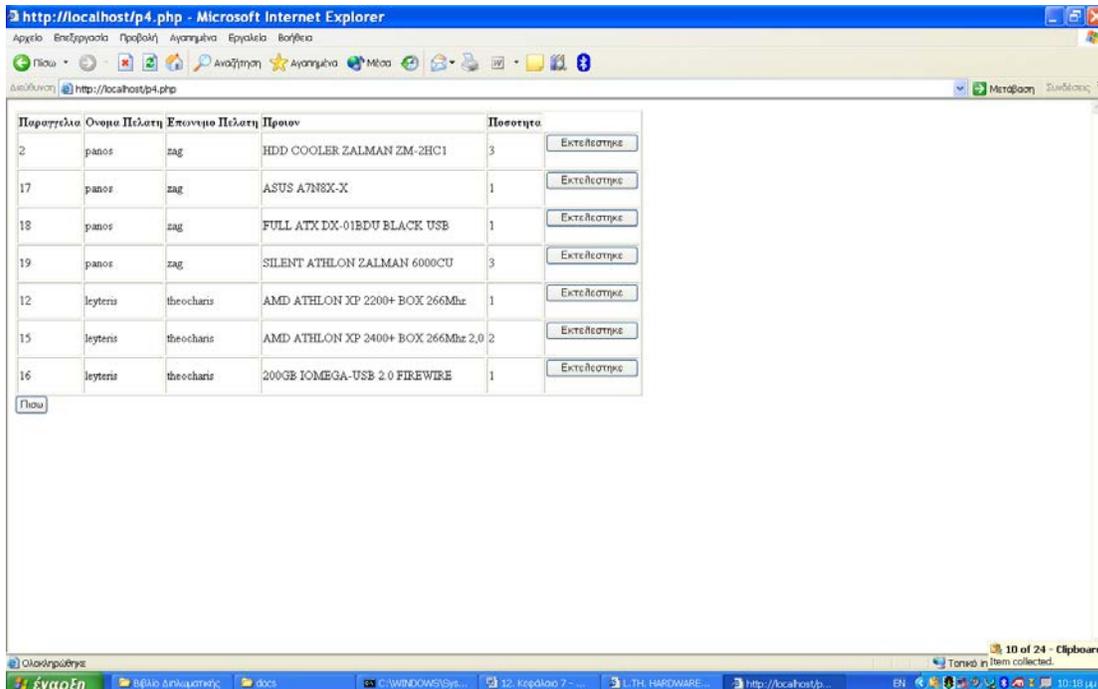
Στη συνέχεια για να μην γίνει λάθος διαγραφή εμφανίζονται τα στοιχεία του προϊόντος με τον αντίστοιχο κωδικό (σελίδα del_pr.php). Ο διαχειριστής αφού επιβεβαιώσει ότι είναι το προϊόν που θέλει να διαγράψει επιλέγει διαγραφή όπου και του εμφανίζεται ανάλογο μήνυμα που επιβεβαιώνει την διαγραφή.



Αντίστοιχα με την διαγραφή γίνεται και στην επεξεργασία όπου καλούνται οι σελίδες p3.php, upd_pr.php, upd_pr2.php.



Για την διαχείριση των παραγγελιών καλείται η σελίδα p4.php όπου όταν μια παραγγελία διεκπεραιωθεί, ο διαχειριστής απλά επιλέγει Εκτελέστηκε και έτσι η συγκεκριμένη παραγγελία φεύγει από την λίστα των εκκρεμών παραγγελιών.



7.3 Περιγραφή της Βάσης Δεδομένων

Η PHP μπορεί να επικοινωνήσει με μία σειρά εντολών άμεσα με την MYSQL κάνοντας το προγραμματισμό σε βάσεις δεδομένων αρκετά απλό και εύκολα υλοποιήσιμο. Οι εντολές που χρησιμοποιήσαμε στο δικό μας ηλεκτρονικό κατάστημα ήταν:

mysql_connect, mysql_select_db, mysql_query, mysql_fetch_array, mysql_num_rows

Το κατάστημα μας στηρίζεται σε μία βάση δεδομένων με όνομα **base**. Η σχεδίαση της βάσης φαίνεται παρακάτω:

Η Βάση μας περιέχει τους παρακάτω πίνακες

| | |
|--------------------|--|
| kalathi | Ο πίνακας που περιγράφει το καλάθι |
| prosorinoi_pelates | Πίνακας που κρατά ένα κωδικό για κάθε πελάτη που επισκέπτεται το κατάστημα |
| pelates | Ο πίνακας που περιγράφει τα στοιχεία κάθε πραγματικού πελάτη (κάποιου που αγόρασε) |
| proionta | Ο πίνακας των προϊόντων του καταστήματος |
| prosfores | Ο πίνακας των προσφορών του καταστήματος |
| sales | Ο πίνακας των πωλήσεων του καταστήματος |
| users_base | Ο πίνακας των εξουσιοδοτημένων χρηστών του καταστήματος |

Ο πίνακας **kalathi** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|--------------|---------------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| id_pel | int(10) | YES | | NULL | |
| id_proiontos | int(10) | YES | | NULL | |
| sale_date | date | YES | | NULL | |
| posotita | int(10) | YES | | NULL | |
| timi_monados | decimal(10,2) | YES | | NULL | |
| synolo | decimal(11,2) | YES | | NULL | |

Ο πίνακας **prosorinoi_pelates** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|--------|---------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| status | int(1) | YES | | NULL | |

Ο πίνακας **users_base** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|----------|-------------|------|-----|---------|-------|
| username | varchar(40) | YES | | NULL | |
| password | varchar(40) | YES | | NULL | |

Ο πίνακας **pelates** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|-------------|-------------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| pel_name | varchar(40) | YES | | NULL | |
| pel_surname | varchar(40) | YES | | NULL | |
| job | varchar(40) | YES | | NULL | |
| address | varchar(80) | YES | | NULL | |
| phone | varchar(40) | YES | | NULL | |
| username | varchar(40) | YES | MUL | NULL | |
| password | varchar(8) | YES | | NULL | |

Ο πίνακας **proionta** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|--------------|---------------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| perigrafi | varchar(255) | YES | | NULL | |
| typos | varchar(40) | YES | | NULL | |
| timi_monadas | decimal(10,2) | YES | | NULL | |
| marka | varchar(40) | YES | | NULL | |
| image | varchar(80) | YES | | NULL | |
| posotita | int(10) | YES | | NULL | |

Ο πίνακας **sales** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|--------------|---------------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| id_pel | int(10) | YES | | NULL | |
| id_proiontos | int(10) | YES | | NULL | |
| sale_date | date | YES | | NULL | |
| posotita | int(10) | YES | | NULL | |
| timi_monados | decimal(10,2) | YES | | NULL | |
| synolo | decimal(11,2) | YES | | NULL | |
| ok_par | int(2) | YES | | NULL | |

Ο πίνακας **prosfores** περιλαμβάνει την παρακάτω σχεδίαση:

| Field | Type | Null | Key | Default | Extra |
|------------|--------------|------|-----|---------|----------------|
| id | int(10) | | PRI | NULL | auto_increment |
| title | varchar(80) | YES | | NULL | |
| start_date | date | YES | | NULL | |
| end_date | date | YES | | NULL | |
| image | varchar(80) | YES | | NULL | |
| Doc | varchar(255) | YES | | NULL | |

Πιο αναλυτικά η βάση μας είναι σχεδιασμένη έτσι ώστε να ακολουθεί την βασική φιλοσοφία του καταστήματος μας.

Όταν ένας χρήστης επισκέπτεται το site μας, λαμβάνει έναν κωδικό από τον πίνακα `prosorinoi_pelates`. Το σύνολο των εγγραφών στον πίνακα `prosorinoi_pelates` περιλαμβάνουν έτσι την πραγματική επισκεψιμότητα του δικτυακού μας τόπου. Ο κωδικός αυτός στέλνεται μέσω ενός cookie στο Client ώστε να μπορεί να γνωρίζει τον κωδικό του και το τοπικό του τερματικό. Με χρήση του κωδικού αυτού μπορεί ο επισκέπτης να γεμίζει το καλάθι του κάνοντας καταχωρίσεις στον πίνακα `kalathi` με χρήση των ανάλογων κωδικών από τα προϊόντα (πίνακας `proionta`). Όταν ο πελάτης οριστικοποιήσει το καλάθι του, μπορεί να δώσει παραγγελία και τότε καλείται να καταχωρήσει τα στοιχεία του στον πίνακα `relates` για να μπορεί να εκτελεστεί η παραγγελία του. Σε αυτόν καταχωρεί τα απαραίτητα προσωπικά δεδομένα (όνομα, επώνυμο, διεύθυνση κ.λ.π.). Αν η καταχώρηση γίνεται για πρώτη φορά, έχουμε και την καταχώρηση του `username` και `password` που στη συνέχεια θα τον χαρακτηρίζουν ώστε να μην κάνει συνεχείς καταχωρήσεις. Σε περίπτωση που έχει δώσει έστω και μια φορά τα στοιχεία, απλά δίνει το `username` και `password` και επιβεβαιώνοντας τα στοιχεία του ολοκληρώνει την παραγγελία του. Ο πίνακας `users_base` χρησιμοποιείται για τους διαχειριστές του καταστήματος όπου επισκέπτονται την σελίδα `admin.php`, ενώ ο πίνακας `prosfores` μόνο για την αρχική σελίδα του καταστήματος όπου εμφανίζει τις προσφορές του καταστήματος που ισχύουν.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Τα ηλεκτρονικά καταστήματα αποτελούν μοναδική ευκαιρία για κάθε επιχείρηση που θέλει να είναι ανταγωνιστική και να επιβιώσει στις νέες συνθήκες έντονου ανταγωνισμού. Η ύπαρξη ενός ηλεκτρονικού καταστήματος για κάθε επιχείρηση θεωρείται αναγκαία. Ο σωστός σχεδιασμός του λογισμικού και της ασφάλειας είναι στοιχεία απαραίτητα για τη σωστή λειτουργία και επιβίωση του καταστήματος. Τα ηλεκτρονικά καταστήματα σε σχέση με τα συμβατικά, έχουν το στρατηγικό πλεονέκτημα της μη αναγκαίας ύπαρξης μεσαζόντων για την έκθεσή τους στον πελάτη, άρα μεγαλύτερο κέρδος για το κατάστημα και χαμηλότερο κόστος για τον πελάτη.

Με την εργασία που πραγματοποιήθηκε, αναπτύχθηκε ένα ικανό ηλεκτρονικό κατάστημα με αρκετές δυνατότητες, αλλά κυρίως τέθηκαν οι βάσεις ενός περισσότερο ολοκληρωμένου καταστήματος, στα πλαίσια του επαγγελματικού.

Μπορεί να αναφερθεί ότι το ηλεκτρονικό κατάστημα επιδέχεται πρόσθετες δυνατότητες και βελτιώσεις, έτσι ώστε να γίνει πιο εύχρηστο, φιλικό και ελκυστικό στον πελάτη-χρήστη, καθώς το απαραίτητο για το σκοπό αυτό λογισμικό εξελίσσεται και δίνει νέες δυνατότητες στο δημιουργό. Άλλες δυνατότητες που θα μπορούσε να έχει το κατάστημα είναι αυτοματοποιημένος έλεγχος αποθεμάτων, εξατομίκευση της αγοραστικής συμπεριφοράς του καταναλωτή, υποστήριξη πολλαπλών γλωσσών για χρήση από ένα πιο ευρύ κοινό, η προσφορά εκπωτικών κουπονιών στους συχνούς αγοραστές καθώς και η ενσωμάτωση στην ιστοσελίδα του καταστήματος πιο σύνθετων γραφικών και άλλων εφφέ ώστε να γίνει πιο ελκυστικό στον πελάτη.

ΠΑΡΑΡΤΗΜΑ

ΚΩΔΙΚΑΣ ΤΟΥ ΗΛΕΚΤΡΟΝΙΚΟΥ ΚΑΤΑΣΤΗΜΑΤΟΣ

Η σελίδα admin.php

```
<html>
<h1>Σελίδα Διαχείρισης Καταστήματος </h1>
<form method="post" action="check.php">
Username:<input type="text" name="usn"><br>
Password:<input type="Password" name="pss"><br>
<input type="submit" value=Καταχώρηση> <input type="reset" value=Καθαρισμός>
</form>
</html>
```

Η σελίδα check.php

```
<?php

$usn=$_POST['usn'];
$pss=$_POST['pss'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select count(*) as cn from users_base where username=".$usn." and
password=".$pss."";

if ($results = mysql_query($sql))
{
$row=mysql_fetch_array($results);
    if ($row[cn]>0)
    {
        echo "<h1> Επιλογές Διαχειριστή Καταστήματος </h1>";
        echo " <a href='p1.php'><b> Προσθήκη Νέου Προϊόντος </b></a> <br>";
        echo " <a href='p2.php'><b> Διαγραφή Προϊόντος </b></a> <br>";
        echo " <a href='p3.php'><b> Επεξεργασία Προϊόντος </b></a> <br>";
        echo " <a href='p4.php'><b> Εμφάνιση Παραγγελιών </b></a> <br>";

    } else
    {
        echo "Ο χρήστης δε βρέθηκε !";
    }
} else
{ echo "σφάλμα σύνδεσης !!!";}

?>
```

Η σελίδα checkuser.php

```
<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select count(*) as cc from pelates where username='".$_POST['usn']."' and
password='".$_POST['pss']."'";

if ($results = mysql_query($sql))
{
$row=mysql_fetch_array($results);

if ($row['cc']>0) {
    $log=mysql_connect ("localhost");
    $db=mysql_select_db("base");
    $sql="select id from pelates where username='".$_POST['usn']."' and
password='".$_POST['pss']."'";
    if ($results = mysql_query($sql))
    {
        $row=mysql_fetch_array($results);
        echo "<form action='teliki.php?ids=" . $row['id'] . "' method='post'>";
        echo "Αριθμός πιστωτικής κάρτας σας:<input type='text
name=pist><br>";
        echo "<input type='submit' value='Αποστολή'>";
        echo "<input type='reset' value='Καθαρισμός'>";
        echo "</form>";
    }
    else
    { echo "error";}
}
else {
include "paragelia.php";
echo "<hr><b>Ο χρήστης δε βρέθηκε !!!<b> <hr>";
}
else
{echo "error";}
?>
```

Η σελίδα cookie.php

```
<?php
if ($_COOKIE['TestCookie'] == "")
{
echo 'test';
}
else
{
echo 'test2'; }
?>
```

Η σελίδα del_kalathi.php

```
<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="delete from kalathi where kalathi.id="._GET['ids'];
$results = mysql_query($sql);
include "showkalathi.php";
?>
```

Η σελίδα del_pr2.php

```
<html>
<?php
$cod=$_GET['ids'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="delete from proionta where id=".$cod;
if ($results = mysql_query($sql))
{
echo "<font color='#00ff00'> Επιτυχής διαγραφή προϊόντος . </font>";
}
else{
echo "<font color='#ff0000'> Ανεπιτυχής διαγραφή προϊόντος !!! </font>";}
?>
```

Η σελίδα del_pr.php

```
<html>
<?php
$cod=$_POST['codproion'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select * from proionta where id=".$cod;
if ($results = mysql_query($sql))
{
$row=mysql_fetch_array($results);
echo "<form method='POST' action='del_pr2.php?ids=".$cod."'>";
?>
<H1>ΔΙΑΓΡΑΦΗ ΠΡΟΙΟΝΤΟΣ</h1>
<Table border=1>
<tr> <td><b>ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΠΡΟΙΟΝΤΟΣ:</b></td> <td>
<?php
echo "<tr> <td><b>ΤΥΠΟΣ ΠΡΟΙΟΝΤΟΣ:</b>".$row[typos]."</tr>";
echo "<tr> <td><b>ΜΑΡΚΑ:</b>".$row[marka]."</tr>";
echo "<tr> <td><b>ΕΙΚΟΝΑ:</b><IMG SRC=".$row[image]."></tr>";
?>
</TABLE>
```

```

<input type="submit" value="ΔΙΑΓΡΑΦΗ">
<button onclick="window.history.back()">Πίσω</button>
</form>
<?php
}
else{
echo "<font color='#ff0000'> Ανεπιτυχής διαγραφή προϊόντος !!! </font>";}
?>
</html>

```

Η σελίδα index2.php

```

<?php
include ('page1.htm');
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select image,title,Doc,
date_format(start_date,'%D %M %Y') as sd,
date_format(end_date,'%D %M %Y') as ed
from prosfores where start_date<=sysdate() and end_date>=sysdate()
order by start_date desc";
if ($results = mysql_query($sql))
{
echo "<h2> Νέες Προσφορές </h2>";
echo "<table >";
while ($row=mysql_fetch_array($results))
{
echo "<tr><td>";
echo "<img height=80 src='".$row[image].">";
echo "<font color='#ff0000' size=4>".$row[title]."</font>";
echo "<p>".$row[Doc]."</p>";
echo "<p> Η προσφορά ισχύει από ".$row[sd]. " έως ".$row[ed];
echo "<hr></td></tr>";
}
echo "</table>";
}else
{
echo "error";
}
include ('page2.htm');
?>

```

Η σελίδα ins kalathi.php

```

<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="insert into kalathi(id_pel,id_proiontos,sale_date,posotita,timi_monados,synolo)

```

```

values ("$_COOKIE['id_pel'].", "$_GET['idp'].", sysdate(), "$_POST['posparag'].",
"$_GET['timi'].", "$_POST['posparag']*$_GET['timi'].");
if ($results = mysql_query($sql))
{
echo "Η παραγγελία καταχωρήθηκε στο καλάθι σας";
}else
{
echo "error insert !!!";
}
?>

```

Η σελίδα ins_pr.php

```

<?php
$per=$_POST['PER'];
$typ=$_POST['TYP'];
$mar=$_POST['MAR'];
$tim=$_POST['TIM'];
$ima=$_FILES['IMA']['name'];
$pos=$_POST['POS'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="insert into proionta(perigrafi,typos,timi_monadas,marka,image,posotita)";
$sql=$sql." values('".$per."','".$typ."','".$tim."','".$mar."','".$ima."','".$pos."')";
echo "IMAGE=".$_FILES['IMA']['name'];
if
(move_uploaded_file($_FILES['IMA']['tmp_name'],'eikones/'.$_FILES['IMA']['name']))
{
echo "<font color='#00ff00'> Επιτυχής εισαγωγή αρχείου . </font>";
} else
{
echo "<font color='#ff0000'> Ανεπιτυχής εισαγωγή αρχείου !!! </font>";}
if ($results = mysql_query($sql))
{
echo "<font color='#00ff00'> Επιτυχής εισαγωγή προϊόντος . </font>";
}
else{
echo "<font color='#ff0000'> Ανεπιτυχής εισαγωγή προϊόντος !!! </font>";}
?>

```

Η σελίδα kalathi.php

```

<?php
echo "<b>Product code=".$_GET['idpr']."</b>";
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select * from proionta where id=".$_GET['idpr']."";
if ($results = mysql_query($sql))

```

```

{
  while ($row=mysql_fetch_array($results))
  {
echo "<form action=ins_kalathi.php?idp=".$row[id]."&timi=".$row[timi_monadas]."
method='post'>";
echo "<table border=1>";
echo "<tr><td> <b> Εικόνα προϊόντος </b> </td>";
echo "<td></td></tr>";
echo "<tr><td> <b> Μοντέλο </b> </td>";
echo "<td>".$row[marka]."</td></tr>";
echo "<tr><td> <b> Χαρακτηριστικά </b> </td>";
echo "<td>".$row[perigrafii]."</td></tr>";
echo "<tr><td> <b> Τιμή μονάδας(ευρώ) </b> </td>";
echo "<td>".$row[timi_monadas]."</td></tr>";
echo "<tr><td> <b> Ποσότητα παραγγελίας</b> </td>";
echo "<td><input type='text' name='posparag'> </td></tr>";
  }
echo "</table><center>";
echo "<input type='submit' value=Προσθήκη </button></a> <button
onclick='window.close()'> Άκυρο </Button></center>";
echo "</form>";
}else
{
echo "error";
}
?>

```

Η σελίδα mainpage.php

```

<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="insert into prosorinoi_pelates(status) values(1)";
$results = mysql_query($sql);
$sql="select max(id) as mm from prosorinoi_pelates";
$results = mysql_query($sql);
$row=mysql_fetch_array($results);
//$f=fopen ("a12.txt","w");
//fwrite ($f,$row['mm']);
setcookie ("id_pel",$row['mm']);
?>

```

```

<script>
window.location.href="index2.php";
</script>

```

Η σελίδα newcustomer.php

```
<?php
$usn=$_POST['usn'];
$pss=$_POST['pss'];
$nm=$_POST['nm'];
$snm=$_POST['snm'];
$ad=$_POST['ad'];
$ph=$_POST['ph'];
$jjo=$_POST['jjo'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="insert into pelates(pel_name,pel_surname,job,address,phone,username,password)";
$sql=$sql."values('".$nm."','".$snm."','".$jjo."','".$ad."','".$ph."','".$usn."','".$pss."')";
if ($results = mysql_query($sql))
{
echo "<hr> <b>Επιτυχής εισαγωγή νέου ονόματος . <br>";
echo "Καταχωρήστε τα Username και Password !!</b><hr>";
include "paragelia.php";
}
else{
echo "<b> Ανεπιτυχής εισαγωγή νέου ονόματος !!! <b><hr>";
include "paragelia.php";
}
?>
```

Η σελίδα order_upd.php

```
<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="update sales set ok_par=1 where sales.id=".$_GET['ids'];
if ($results = mysql_query($sql))
{
echo "Η παραγγελία τέθηκε σε κατάσταση εκτέλεσης";
}
else echo "error !!!";
?>
```

Η σελίδα p1.php

```
<html>
<form enctype="multipart/form-data" method="POST" action="ins_pr.php">
<H1>Προσθήκη Νέου Προϊόντος</h1>
<table>
<tr> <td>ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="text"
name="PER"></td></tr>
<tr> <td>ΤΥΠΟΣ ΠΡΟΙΟΝΤΟΣ:</td> <td><SELECT name="TYP">
```

```

<OPTION> Monitors
<OPTION> box
<OPTION> Motherboards
<OPTION> Display_adaptors
<OPTION> hard_disks
<OPTION> fans
<OPTION> memory
<OPTION> sound_cards
<OPTION> processors
<OPTION> CD_DVD
<OPTION> keyboards
<OPTION> mouse
</SELECT>
</td></tr>
<tr> <td>ΜΑΡΚΑ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="text"
name="MAR"></td></tr>
<tr> <td>ΤΙΜΗ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="text" name="TIM"></td></tr>
<input type="hidden" name="MAX_FILE_SIZE" value="160000">
<tr> <td> ΑΡΧΕΙΟ ΕΙΚΟΝΑΣ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="file"
name="IMA" ></td></tr>
<tr> <td>ΔΙΑΘΕΣΙΜΗ ΠΟΣΟΤΗΤΑ:</td> <td><input type="text"
name="POS"></td></tr>
</TABLE>
<input type="submit" value=Καταχώρηση> <input type="reset" value=Ακύρωση>
<button onclick="window.history.back()">Πίσω</button>
</form>
</html>

```

Η σελίδα p2.php

```

<html>
<form enctype="multipart/form-data" method="POST" action="del_pr.php">
<H1>Διαγραφή Προϊόντος</h1>
<Table>
<tr> <td>Κωδικός Προϊόντος:</td> <td><input type="text"
name="codproion"></td></tr>
</table>
<input type="submit" value=Καταχώρηση> <input type="reset" value=Ακύρωση>
<button onclick="window.history.back()">Πίσω</button>
</form>
</html>

```

Η σελίδα p3.php

```

<html>
<form enctype="multipart/form-data" method="POST" action="upd_pr.php">
<H1>Επεξεργασία Προϊόντος</h1>
<Table>

```

```

<tr> <td>Κωδικός Προϊόντος:</td> <td><input type="text"
name="codproion"></td></tr>
</table>
<input type="submit" value=Καταχώρηση> <input type="reset" value=Ακύρωση>
<button onclick="window.history.back()">Πισω</button>
</form>
</html>

```

Η σελίδα p4.php

```

<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select sales.id,pelates.pel_name,pelates.pel_surname,proionta.marka ";
$sql=$sql.",sales.posotita from proionta,sales,pelates where";
$sql=$sql." pelates.id=sales.id_pel and";
$sql=$sql." sales.id_proiontos=proionta.id and sales.ok_par is null";
echo "<table border=1>";
echo "<tr>";
echo "<td> <b> Παραγγελια </b> </td>";
echo "<td> <b> Ονομα Πελατη </b> </td>";
echo "<td> <b> Επωνυμο Πελατη </b> </td>";
echo "<td> <b> Προιον </b> </td>";
echo "<td> <b> Ποσοτητα </b> </td>";
echo "</tr>";
if ($results = mysql_query($sql))
{
    while ($row=mysql_fetch_array($results))
    {
        echo "<tr>";
        echo "<td>".$row[id]."</td >";
        echo "<td>".$row[pe_name]."</td >";
        echo "<td>".$row[pe_surname]."</td >";
        echo "<td>".$row[marka]."</td >";
        echo "<td>".$row[posotita]."</td >";
        echo "<td> <form action='order_upd.php?ids=".$row[id]."' method='post'>";
        echo "<input type='submit' value='Εκτελεστηκε'></form></td >";
        echo "</tr>";
    }
    echo "</table>";
}else {
echo "error";
}
?>
<button onclick="window.history.back()">Πισω</button>

```

Η σελίδα paragelia.php

```

<body>
<h1> Φόρμα Παραγγελίας </h1>
Εάν έχετε ήδη username και password, παρακαλώ συμπληρώστε παρακάτω
<form action="checkuser.php" method="post">
<table>
<tr><td> UserName: </td><td><input type="text" name=usn></td></tr>
<tr><td> PassWord: </td><td><input type="password" name=pss></td></tr>
</table>
<input type="submit" value="Αποστολή"><input type="reset" value="Καθαρισμός">
</form>
<hr>
Εάν δεν έχετε username και password, παρακαλώ πιέστε για να εγγραφείτε<br>
<center> <a href="register.php"> <b>ΕΓΓΡΑΦΗ</b> </a>
</center>
</body>

```

Η σελίδα register.php

```

<body>
<h1> Φόρμα Εγγραφής Καινούριου Πελάτη </h1>
<hr>
<form action="newcustomer.php" method="post">
<table>
<tr><td> UserName: </td><td><input type="text" name=usn></td></tr>
<tr><td> PassWord: </td><td><input type="password" name=pss></td></tr>
<tr><td> Name: </td><td><input type="text" name=nm></td></tr>
<tr><td> Surname: </td><td><input type="text" name=snm></td></tr>
<tr><td> Address: </td><td><input type="text" name=ad></td></tr>
<tr><td> Phone: </td><td><input type="text" name=ph></td></tr>
<tr><td> Job: </td><td><input type="text" name=jo></td></tr>
</table>
<input type="submit" value="Αποστολή"><input type="reset" value="Καθαρισμός">
</form>
</body>

```

Η σελίδα showkalathi.php

```

<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select kalathi.id,proionta.marka, proionta.image, kalathi.timi_monados,
kalathi.posotita, kalathi.synolo from kalathi,proionta where
proionta.id=kalathi.id_proiontos and kalathi.id_pel=".$_COOKIE['id_pel'];
echo "<table border=1>";
echo "<tr>";
    echo "<td><b> Εικόνα </b></td>";
    echo "<td><b> Προϊόν </b></td>";

```

```

        echo "<td><b> Ποσότητα </b></td>";
        echo "<td><b> Τιμή μονάδας </b></td>";
        echo "<td><b> Σύνολο </b></td>";
        echo "<td><b> Διαγραφή </b></td>";
    echo "</tr>";
    $i=1;
    if ($results = mysql_query($sql))
    {
        while ($row=mysql_fetch_array($results))
        {
    echo "<tr>";
        echo "<td> <img src='\".$row[image].\"' height=70></td>";
        echo "<td>\".$row[marka].\"</td>";
        echo "<td>\".$row[posotita].\"</td>";
        echo "<td>\".$row[timi_monados].\"</td>";
        echo "<td>\".$row[synolo].\"</td>";
        echo "<td><form action='del_kalathi.php?ids=\".$row[id].\"' method='post'><input
type='submit' value='delete'></form></td>";
    echo "</tr>";
    $i=$i+1;
        }
    $sql="select sum(synolo) as ss from kalathi where kalathi.id_pel="._COOKIE['id_pel'];
    if ($results = mysql_query($sql))
    { $row=mysql_fetch_array($results);
        echo "<tr><td></td><td></td><td></td><td><b> Συνολικο
ποσο:</b></td><td>\".$row[ss].\"</td></tr>";
    }
    echo "</table>";
    echo "<form action='paragelia.php?idpel="._COOKIE['id_pel'].\"' method='post'>";
    echo "<input type='submit' value='Αποστολή παραγγελίας'>";
    echo "<input type='reset' value='Καθαρισμός'>";
    echo "</form>";

} else
{
    echo "error";
}
include ('page2.htm');
?>

```

Η σελίδα showparagelies.php

```

<?php
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select sales.id,pelates.id as id_pelat,proionta.id as"; $sql=$sql." pr_id,posotita from
proionta,sales,pelates where ";$sql=$sql." pelates.id=sales.id_pel and";
$sql=$sql." sales.id_proiontos=proionta.id";
echo "<table border=1>";

```

```

echo "<tr>";
    echo "<td> <b> Παραγγελία </b> </td>";
    echo "<td> <b> Πελάτης </b> </td>";
    echo "<td> <b> Προϊόν </b> </td>";
    echo "<td> <b> Ποσότητα </b> </td>";
echo "</tr>";
if ($results = mysql_query($sql))
{
    while ($row=mysql_fetch_array($results))
    {
echo "<tr>";
        echo "<td>".$row[id]."</td >";
        echo "<td>".$row[id_pelat]."</td >";
        echo "<td>".$row[pr_id]."</td >";
        echo "<td>".$row[posotita]."</td >";
echo "</tr>";
    }
echo "</table>";
}else
{
echo "σφάλμα";
}
?>

```

Η σελίδα showproducts.php

```

<?php
include ('page1.htm');
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select * from proionta where typos='".$_GET['selection']."'";
echo "<table border=1>";
echo "<tr>";
echo "<td> <b> Εικόνα προϊόντος </b> </td>";
echo "<td> <b> Μοντέλο </b> </td>";
echo "<td> <b> Χαρακτηριστικά </b> </td>";
echo "<td> <b> Τιμή μονάδας(ευρώ) </b> </td>";
echo "</tr>";
if ($results = mysql_query($sql))
{
    while ($row=mysql_fetch_array($results))
    {
echo "<tr><td bgcolor=ffffff>";
// echo "<a href=kalathi.php?idpr=".$row[id]."></a></td>";
        echo "</td>";
        echo "<td>".$row[marka]."</td>";

```

```

        echo "<td>".$row[perigrafi]."</td>";
        echo "<td>".$row[timi_monadas]."</td>";
    echo "</tr>";
    }
echo "</table>";
}else
{
echo "error";
}
include ('page2.htm');
?>

```

H σελίδα teliki.php

```

<?php
$id_pel=$_GET['ids'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select kalathi.id,proionta.id as idp,proionta.marka, proionta.image,
kalathi.timi_monados, kalathi.posotita, kalathi.synolo from kalathi,proionta where
proionta.id=kalathi.id_proiontos and kalathi.id_pel="._COOKIE['id_pel'];
if ($results = mysql_query($sql))
{
    while ($row=mysql_fetch_array($results))
    {
        $synolo=$row[posotita]*$row[timi_monados];
        $sql_ins="insert into sales(id_pel,id_proiontos,sale_date,posotita,timi_monados,synolo)
values ";
        $sql_ins=$sql_ins."(".$id_pel.", ".$row['idp'].",sysdate(),".$row[posotita].",".$row[timi_m
onados].",".$synolo.)";
        if ($results2 = mysql_query($sql_ins)) { echo "Η παραγγελία καταχωρήθηκε για το
προϊόν ".$row[marka]."<br>";} else {echo "σφάλμα";}
        $sql_del="delete from kalathi where id_pel="._COOKIE['id_pel'];
        $results3 = mysql_query($sql_del);
    }
} ?>

```

H σελίδα upd_pr2.php

```

<?php
$per=$_POST['PER'];
$typ=$_POST['TYP'];
$mar=$_POST['MAR'];
$tim=$_POST['TIM'];
$ima=$_FILES['IMA']['name'];
$pos=$_POST['POS'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");

```

```

$sql="update proionta ";
$sql=$sql."set perigrافي="".$per."";
$sql=$sql."typos="".$styp."";
$sql=$sql."timi_monadas="".$stim."";
$sql=$sql."marka="".$mar."";
$sql=$sql."image='eikones/'".$sima."";
$sql=$sql."posotita="".$pos."";
$sql=$sql." where id="".$_GET['ids'];
if ($sima!=""){
if
(move_uploaded_file($_FILES['IMA']['tmp_name'],'eikones/'$_FILES['IMA']['name']))
{
echo "<font color='#00ff00'> Επιτυχής εισαγωγή αρχείου . </font>";
} else
{
echo "<font color='#ff0000'> Ανεπιτυχής εισαγωγή αρχείου !!! </font>";
}
if ($results = mysql_query($sql))
{
echo "<font color='#00ff00'> Επιτυχής ανανέωση προϊόντος . </font>";
}
else{
echo "<font color='#ff0000'> Ανεπιτυχής ανανέωση προϊόντος !!! </font>";
?>
}
}

```

Η σελίδα upd_pr.php

```

<html>
<?php
$cod=$_POST['codproion'];
$log=mysql_connect ("localhost");
$db=mysql_select_db("base");
$sql="select * from proionta where id="".$cod;
if ($results = mysql_query($sql))
{
$row=mysql_fetch_array($results);
echo "<form method='POST' action='upd_pr2.php?ids="".$cod."">";
?>
<h1>Επεξεργασία προϊόντος</h1>
<table>
<tr> <td>ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΠΡΟΙΟΝΤΟΣ:</td> <td><textarea rows=7 cols=30
name="PER"> <?php echo $row[perigrافي]; ?> </textarea></td></tr>
<tr> <td>ΤΥΠΟΣ ΠΡΟΙΟΝΤΟΣ:</td> <td><SELECT name="TYP" >
<OPTION value=<?php echo $row[typos]; ?>> <?php echo $row[typos]; ?>
<OPTION value='Monitors'> Monitors
<OPTION value='box'> box
<OPTION value='Motherboards'> Motherboards
<OPTION value='Display_adaptors'> Display_adaptors
<OPTION value='hard_disks'> hard_disks

```

```

<OPTION value='fans'> fans
<OPTION value='memory'> memory
<OPTION value='sound_cards'> sound_cards
<OPTION value='processors'> processors
<OPTION value='CD_DVD'> CD_DVD
<OPTION value='keyboards'> keyboards
<OPTION value='mouse'> mouse
</SELECT>
</td></tr>
<tr> <td>ΜΑΡΚΑ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="text" name="MAR"
value=<?php echo $row[marka]; ?>></td></tr>
<tr> <td>ΤΙΜΗ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="text" name="TIM" value=<?php
echo $row[timi_monadas]; ?>></td></tr>
<input type="hidden" name="MAX_FILE_SIZE" value="160000">
<tr> <td> ΕΙΚΟΝΑ ΠΡΟΙΟΝΤΟΣ:</td> <td><img src=<?php echo $row[image];
?>></td></tr>
<tr> <td> ΑΡΧΕΙΟ ΕΙΚΟΝΑΣ ΠΡΟΙΟΝΤΟΣ:</td> <td><input type="file"
name="IMA" ></td></tr>
<tr> <td>ΔΙΑΘΕΣΙΜΗ ΠΟΣΟΤΗΤΑ:</td> <td><input type="text" name="POS"
value=<?php echo $row[posotita]; ?>></td></tr>
</TABLE>
<input type="submit"> <input type="reset">
</form>
</html>
<?php
}
else{
echo "<font color='#ff0000'> Ανεπιτυχής επεξεργασία προϊόντος !!! </font>";}
?>
</html>

```

Η σελίδα page1.htm

```

<HTML>
<TITLE> L.TH. HARDWARE </TITLE>
<BODY>
<table width="700" border="0" cellspacing="0" cellpadding="0" align="center">
<tr><td>
<table border=0>
<tr>
<td width=160><a href="index2.php">  </a></td>
<td width=400> </td>
<td width=140>  </td>
</tr>
</table>
<table border=0 width="550">
<tr>
<td align="center" width=170></td>
<td align="center">

```

```

<button onclick='window.open("showkalathi.php","cart",
"toolbar=no,scrollbars=1,width=600,height=460")>Το καλάθι μου</button></td>
</tr>
</table>
<table><tr>
<td width=17% valign="top">
  <table cellpadding="4" cellspacing="0">
    <tr bgcolor="#aaeeaa"><th><h4>Κατηγορία Προϊόντος</h4></th> </tr>
    <tr><td><a href="showproducts.php?selection=monitors">
</a><h4>Οθόνες</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=box"></a><h4>Κουτιά</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=hard_disks">
</a><h4>Σκληροί Δίσκοι</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=display_adaptors">
</a><h4>Κάρτες Γραφικών</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=motherboards">
</a><h4>MotherBoards</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=processors">
</a><h4>Επεξεργαστές</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=fans">
</a><h4>Ψύκτρες</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=memory">
</a><h4>Μνήμη
RAM</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=CD_DVD">
</a><h4>CD και
DVD</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=sound_cards">
</a><h4>Κάρτες Ήχου</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=keyboards">
</a><h4>Πληκτρολόγια</h4></td> </tr>
    <tr><td><a href="showproducts.php?selection=mouse">
</a><h4>Ποντίκια</h4></td> </tr>
  </table>
</td>
<td width=83% bgcolor="#eeddee" valign="top">

```

```
</td>  
</tr>  
</table>  
</td>  
</tr>  
</table>  
</BODY>  
</HTML>
```

ΒΙΒΛΙΟΓΡΑΦΙΑ

ΒΙΒΛΙΟΓΡΑΦΙΑ

Active Server Pages

- <http://asp.superexpert.com>
- <http://learnasp.com>
- <http://www.asp101.com>

Ηλεκτρονικό Εμπόριο

- <http://www.bcentral.com>
- <http://www.ecommerce.gov>
- <http://www.ecommerceadvisor.com>
- <http://aslipstream.com>

Φιλοξενία ASP

- <http://www.datareturn.com>
- <http://www.bitshop.com>
- <http://www.ewebcity.com>
- <http://www.innerhost.com>