

# Τμήμα Τεχνολογίας Πληροφορικής & Τηλεπικοινωνιών



## Δημιουργία Ιστοσελίδας Φοιτητικών Υπηρεσιών

Πτυχιακή Εργασία

Επιβλέποντες Καθηγητές:

Μακροδημήτρης Γεώργιος

Πικραμένος Ιωάννης

Φοιτήτρια:

Φλέσσα Σοφία

Μάιος 2012

Σπάρτη

## Περιεχόμενα

1 Κεφάλαιο: Εισαγωγή .....	4
2 Κεφάλαιο: Το διαδίκτυο, Ο παγκόσμιος Ιστός, ο Σχεδιασμός Ιστοσελίδων & οι Γλώσσες Προγραμματισμού.....	5
2.1 Διαδίκτυο .....	5
2.1.1 Φυσική Υποδομή του Διαδικτύου .....	5
2.2 Παγκόσμιος Ιστός .....	5
2.2.1 Ορισμός.....	6
2.2.2 Ιστορική Αναδρομή .....	6
2.3 Η Ιστοσελίδα .....	7
2.3.1 Τι είναι δικτυακός Τόπος; .....	8
2.3.2 Σχεδιασμος Σελίδων Ιστού (Web Design).....	8
2.3.3 Στατικές Ιστοσελίδες.....	9
2.3.4 Δυναμικές Ιστοσελίδες.....	10
2.3.5 Διαφορές Δυναμικών & Στατικών Ιστοσελίδων.....	11
2.4 Γλώσσες Προγραμματισμού .....	12
2.4.1. Χαρακτηριστικά & Κατηγοριοποίηση Γλώσσών .....	13
2.5 Η Γλώσσα HTML .....	15
2.5.1 Βασική Οργάνωση ενός Εγγράφου HTML .....	15
2.6 Η Γλώσσα PHP .....	25
2.6.1 Δομικά Στοιχεία της PHP .....	26
2.6.2 Δομές Ελέγχου Ροής .....	28
2.6.3 Χρήση & Διαχείριση Φορμών .....	33
2.7 MySQL.....	38
2.7.1 Σπουδαιότητα της Καλής Σχεδίασης μιας Βάσης Δεδομένων.....	38
2.7.2 Τύποι Δεδομένων .....	39
2.7.3 Δημιουργία Πινάκων.....	40
2.7.4 Χρήση εντολών INSERT, SELECT .....	41
2.8 CSS Γλώσσα .....	41
2.9 Ενοποίηση των γλωσσών PHP & MySQL .....	43
2.9.1 Σύνδεση στην MySQL με την PHP .....	44

2.9.2	Δουλεύοντας με δεδομένα της MySQL .....	45
3	Κεφάλαιο: Δημιουργία ιστοσελίδας Φοιτητικών Υπηρεσιών .....	47
3.1	Σχεδιασμός και Δημιουργία των Πινάκων της Βάσης Δεδομένων .....	47
3.2	Δημιουργία Μενού.....	49
3.2.1	Δημιουργία Ιστοσελίδων & Υλοποίηση του Μηχανισμού Αποστολής Φορμών .....	59
3.3	Έλεγχος της πρόσβασης.....	79
3.3.1	Έλεγχος της πρόσβασης με βάση τις τιμές των Cookies.....	80
4	Μελλοντικές Βελτιώσεις.....	88
5	Κεφάλαιο: Βιβλιογραφία .....	89

## 1 Κεφάλαιο: Εισαγωγή

Ζούμε, πλέον, σε μία εποχή όπου ο ρόλος της γνώσης και της πληροφορίας βρίσκεται στο επίκεντρο των εξελίξεων, για αυτό το λόγο είναι σημαντικό οι πληροφορίες και οι γνώσεις αυτές να είναι εύκολα προσβάσιμες από κάθε ενδιφερόμενο. Ο πιο αποτελεσματικός τρόπος να γίνει κάτι τέτοιο είναι μέσω των δικτυακών τόπων. Στην παρούσα εργασία περιγράφεται λεπτομερώς η υλοποίηση ενός ιστότοπου που αναλαμβάνει να παρέχει στους φοιτητές όλες τις υπηρεσίες που τους παρέχει η γραμματεία του τμήματος.

Στόχος της παρούσας πτυχιακής εργασίας είναι να ανεδείξει την κατασκευή ενός δικτυακού τόπου όπου οι φοιτητές θα έχουν την δυνατότητα να κάνουν ηλεκτρονικά όποια αίτηση χρειάζονται, να βλέπουν τις βαθμολογίες τους, να κάνουν τη δήλωση των μαθημάτων τους και να βλέπουν τις ανακοινώσεις του τμήματός τους. Ακόμα ο χρόνος σε εργατοώρες θα μειωθεί μέσω των ηλεκτρονικών υποβολών. Για παράδειγμα ένας φοιτητής χρειάζεται αρχικά να κάνει αίτηση στην γραμματεία. Ύστερα η γραμματεία να ελέγξει την αίτηση και να όλα είναι σωστά και ύστερα να την τυπώσει. Μια τέτοια διαδικασία παίρνει δυο με τρεις μέρες. Αντίθετα με την ηλεκτρονική υποβολή ο φοιτητής θα κάνει την αίτηση από το σπίτι του, η γραμματεία θα την λαμβάνει και εφόσον υπάρχουν στη βάση τα στοιχεία του η αίτησή του θα τυπώνεται μέσα σε λίγες ώρες.

Η ιστοσελίδα είναι υλοποιημένη σε HTML, PHP και MySQL οι οποίες είναι πάρα πολύ δημοφιλείς σε διαδικτυακές εφαρμογές και αν και αποτελούν πολύ δυνατά εργαλεία από μόνες τους, είναι ο συναδυασμός τους που μας δίνει τη δυνατότητα ανάπτυξης πολύ λειτουργικών και δυναμικών εφαρμογών.

Η εργασία χωρίζεται σε δύο βασικά μέρη. Στο πρώτο μέρος αναλύονται γενικότερα ορισμένες θεωρητικές έννοιες του "Παγκόσμιου Ιστού", των "Ιστοσελίδων και περιγράφονται οι παραπάνω γλώσσες προγραμματισμού που απαιτούνται για την κατασκευή της εφαρμογής και παρουσιάζεται κυρίως ο κώδικας της γλώσσας που θα χρησιμοποιήσουμε στην εφαρμογή. Σκοπός του κεφαλαίου είναι να εισάγει τον αναγνώστη στη νέα κατάσταση που επικρατεί αλλά και να ανδείξει τη βάση των νέων τεχνολογιών που είναι οι ιστοσελίδες και οι δυνατότητές του.

Στο δεύτερο κεφάλαιο παρουσιάζεται ο ιστότοπος που δημιουργήσαμε καθώς και οι κώδικες και τα αρχεία που χρησιμοποιήθηκαν για την κατασκευή. Για τη υλοποίηση του κώδικα χρησιμοποιήθηκε το πρόγραμμα eclipse και για την σχεδίαση της βάσης χρησιμοποιήθηκε η MySQL καθώς και ένας web server.

Στο τελευταίο κεφάλαιο αναλύονται τα αποτελέσματα που έχει επιφέρει η κατασκευή του ιστότοπου αυτού καθώς και πως μπορεί να αναπτυχθεί μελλοντικά.

## **2 Κεφάλαιο: Το διαδίκτυο, Ο παγκόσμιος Ιστός, ο Σχεδιασμός Ιστοσελίδων & οι Γλώσσες Προγραμματισμού**

### **2.1 Διαδίκτυο**

Το διαδίκτυο (internet) είναι το πιο επαναστατικό φαινόμενο στην ιστορία μας και από την εποχή της διάδοσης των προσωπικών υπολογιστών κατά την δεκαετία 1970.

Αποτελεί βασική τεχνολογική αλλαγή όχι επειδή είναι απλό και ανεξάρτητο από τη προγενέστερη τεχνολογία αλλά επειδή συνιστά το θεμέλιο πλήθους άλλων τεχνολογικών μετασχηματισμών. Το διαδίκτυο είναι αποτέλεσμα της ψηφιοποίησης και της σύγκλισης των τεχνολογιών, ενσωματώνει όλα τα προϋπάρχοντα δίκτυα και γι' αυτό αρμόζει ο προσδιορισμός: το δίκτυο των δικτύων. Το σπουδαιότερο όλων.

#### **2.1.1 Φυσική Υποδομή του Διαδικτύου**

Δύο ή περισσότεροι υπολογιστές που συνδέονται μεταξύ τους σχηματίζουν ένα δίκτυο υπολογιστών, γεγονός που επιτρέπει στους χρήστες να χρησιμοποιούν από απόσταση τις υπηρεσίες που προσφέρει κάποιος δικτυωμένος υπολογιστής. Η λέξη "internet" αναφέρεται στο διεθνές δίκτυο (INTERNational NETwork) ψηφιακής επικοινωνίας, το οποίο προκύπτει από τη συνένωση χιλιάδων δικτύων – τοπικών (LAN = Local Area Network), πανεπιστημιακών (CWIS = Campus Wide Information System) και άλλων (WAN = Wide Area Network, τοπικό ή εθνικό) τα οποία αλληλεπιδρούν μέσω ορισμένων πρωτοκόλλων κοινών ανά τον κόσμο.

Το διαδίκτυο ως φυσική υποδομή, είναι ένα μεταφορέας πληροφοριών ο οποίος συνίσταται σ' ένα παγκόσμιο πλέγμα δικτυακών υπολογιστών. Αυτό που κάνει ένα δίκτυο μέρος της ίδιας δομής είναι η ικανότητά του για αμφίδρομη επικοινωνία, χάρη σε ορισμένα κοινά πρωτόκολλα τηλεπικοινωνιών, συμπεριλαμβανομένης της απαραίτητης ακολουθίας TCP/IP (Transmission Control Protocol/ Internet Protocol, τα δύο θεμελιώδη πρότυπα που καθορίζουν τα χαρακτηριστικά των πακέτων πληροφοριών, τις διαδικασίες για τη πραγματοποίηση μιας σύνδεσης με σκοπό την ανταλλαγή δεδομένων, τις διαδικασίες ασφαλιμάτων κ.τ.λ.). Το Internet είναι ένα ανοιχτό δίκτυο ανεξάρτητων δικτύων, στο οποίο κάθε υποδίκτυο λειτουργεί και διοικείται αυτόνομα.

### **2.2 Παγκόσμιος Ιστός**



Παγκόσμιος ιστός είναι το δίκτυο των συνδεδεμένων υπολογιστών και δικτύων σε παγκόσμια κλίμακα. Κάθε δίκτυο – δομική μονάδα του διαδικτύου αποτελείται από συνδεδεμένους υπολογιστές σε τοπικό επίπεδο, για παράδειγμα το δίκτυο υπολογιστών των κεντρικών γραφείων μιας εταιρείας. Αυτά τα δίκτυα με τη σειρά τους συνδέονται σε ευρύτερα δίκτυα, όπως εθνικά και υπερεθνικά. Το ευρύτερο δίκτυο στον κόσμο λέγεται παγκόσμιος ιστός το οποίο είναι μοναδικό (δηλαδή δεν υπάρχουν παραπάνω από ένα δίκτυα υπολογιστών παγκόσμιας κλίμακας), και συμπεριλαμβάνεται τόσο τα γήινα δίκτυα, όσο και τα δίκτυα των δορυφόρων της και άλλων διαστημικών συσκευών που είναι συνδεδεμένα σε αυτό.

Η τεχνολογία του ιστού καθιστά δυνατή την δημιουργία “υπερκειμένων”, μια διασύνδεση δηλαδή πάρα πολλών μη ιεραρχημένων στοιχείων που παλαιότερα ήταν απομονωμένα. Τα στοιχεία αυτά μπορούν να πάρουν και άλλες μορφές πέρα της μορφής του γραπτού κειμένου, όπως εικόνας και ήχου.

### 2.2.1 Ορισμός

Ο παγκόσμιος Ιστός είναι μια τεράστια συλλογή από ψηφιακή έγγραφη (τις ιστοσελίδες), που βρίσκονται αποθηκευμένα σε υπολογιστές του Διαδικτύου.

### 2.2.2 Ιστορική Αναδρομή

Η τεχνολογία του ιστού δημιουργήθηκε το 1989 από τον Βρετανό Τιμ Μπέρνερς Λι, που εκείνη την εποχή εργαζόταν στον Ευρωπαϊκό Οργανισμό Πυρηνικών Ερευνών (CERN) στην Γενεύη της Ελβετίας. Επινόησε τον Παγκόσμιο Ιστό, προσπαθώντας να βρει ένα τρόπο να αρχειοθετεί τις επιστημονικές μελέτες των συνεργατών του CERN. Η επιτυχία του ήταν τόσο μεγάλη, ώστε πολύ γρήγορα ενσωματώθηκε στις υπηρεσίες του Διαδικτύου γνωρίζοντας τεράστια απήχηση χάρη στον απλό και ελκυστικό τρόπο περιήγησης και αναζήτησης πληροφοριών.

Το όνομα που έδωσε στην εφεύρεσή του ο ίδιος ο Λι είναι World Wide Web, όρος γνωστός στους περισσότερους από το “www”. Το σχέδιο του ήταν ο καθένας να μπορεί να ανταλλάσσει πληροφορίες και ιδέες άμεσα και να είναι προσβάσιμες από τους υπόλοιπους. Το σημείο στο οποίο έδωσε ιδιαίτερο βάρος ήταν η μη ιεράρχηση των διασυνδεδεμένων στοιχείων. Οραματίστηκε κάθε στοιχείο, κάθε κόμβο του ιστού ίσο ως προς την προσβασιμότητα με τα υπόλοιπα.

Λόγω της μεγάλης απήχησης του Παγκόσμιου Ιστού πολλές φορές ταυτίζουμε τον Παγκόσμιο Ιστό με το Διαδίκτυο. Στην πραγματικότητα αυτή η αντίληψη είναι λανθασμένη. Ο Παγκόσμιος

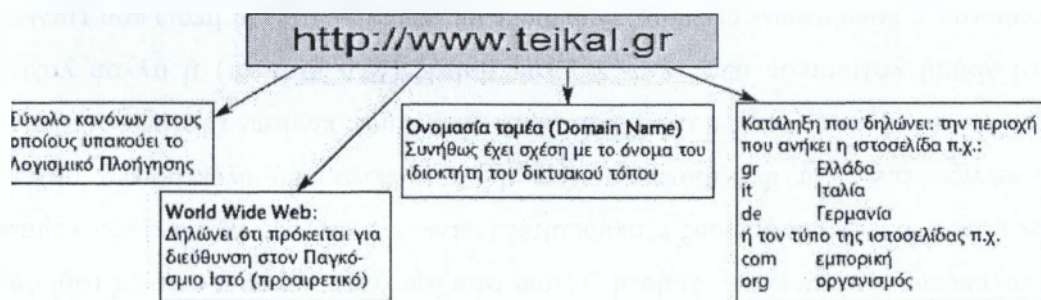
Ιστός είναι μία τεράστια συλλογή από ψηφιακά έγγραφα (τις ιστοσελίδες), που βρίσκονται αποθηκευμένα σε υπολογιστές του Διαδικτύου. Το διαδίκτυο, αντίθετα, είναι ένα παγκόσμιο δίκτυο υπολογιστών συνδεδεμένων μεταξύ τους.

### 2.3 Η Ιστοσελίδα

Η ιστοσελίδα είναι ένα είδος εγγράφου του παγκόσμιου Ιστού που περιλαμβάνει πληροφορίες με την μορφή κειμένου, υπερκειμένου, εικόνας, βίντεο και ήχου. Όπως ένα φορτίο μεταφέρεται με κάποιο φορτηγό σε έναν αυτοκινητόδρομο, έτσι και μία ιστοσελίδα του Παγκόσμιου Ιστού μεταφέρεται στο Διαδίκτυο “ταξιδεύοντας” από κάποιον κεντρικό υπολογιστή, στον οποίο είναι αποθηκευμένη, στον υπολογιστή μας.

Βιβλιοθήκες, σχολεία, δημόσιες υπηρεσίες, πανεπιστήμια, ιδιωτικοί οργανισμοί διατηρούν σε έναν υπολογιστή του Διαδικτύου μια συλλογή από ιστοσελίδες (έναν δικτυακό τόπο – web site), που μπορούμε να επισκεφτούμε, όποια στιγμή θέλουμε.

Κάθε ιστοσελίδα που περιέχεται σε δικτυακό τόπο, έχει τη δική της διεύθυνση στον Παγκόσμιο Ιστό. Αν θέλουμε να επισκεφτούμε μία ιστοσελίδα, πρέπει να ξέρουμε τη διεύθυνσή της. Η διεύθυνση αυτή καλείται URL (Uniform Resource Locator) – Ενιαίος Προσδιοριστής Πόρου) ή απλούστερα διεύθυνση ιστοσελίδας. Μια διεύθυνση ιστοσελίδας είναι μοναδική και έχει συνήθως την εξής μορφή:



Οι ιστοσελίδες στον Παγκόσμιο Ιστό είναι συνδεδεμένες μεταξύ τους, έτσι ώστε με τη βοήθεια του ποντικιού να μπορούμε εύκολα να περιηγούμαστε ανάμεσά τους. Στη γλώσσα της πληροφορικής, όταν επισκεπτόμαστε τη μία ιστοσελίδα μετά την άλλη, λέμε ότι “πλοηγούμαστε” στον Παγκόσμιο Ιστό. Για να έχουμε τη δυνατότητα να βλέπουμε το περιεχόμενο των ιστοσελίδων στην οθόνη του υπολογιστή μας και να πλοηγούμαστε ανάμεσά τους, πρέπει να έχουμε εγκαταστήσει στον υπολογιστή μας ένα Λογισμικό Πλοήγησης ή Φυλλομετρητή (Browser). Οι πιο δημοφιλείς φυλλομετρητές είναι: Microsoft Internet Explorer, Netscape Navigator, Google Chrome, Mozilla, Firefox κ.α.

Αν θέλουμε να επισκεφτούμε μια ιστοσελίδα, αρκεί να πληκτρολογήσουμε τη διεύθυνσή της στο Λογισμικό Πλοήγησης που διαθέτουμε. Για παράδειγμα, αν θέλουμε να επισκεφτούμε την αρχική σελίδα του δικτυακού τόπου του ΤΕΙ Καλαμάτας, πληκτρολογούμε την διεύθυνση [www.teikal.gr](http://www.teikal.gr). Μέσα σε λίγα δευτερόλεπτα θα εμφανιστεί η σελίδα στη οθόνη του υπολογιστή μας και θα αρχίσουμε την πλοήγηση.



### 2.3.1 Τι είναι δικτυακός Τόπος;

Ένας ιστότοπος, ιστοχώρος ή δικτυακός τόπος (web site) είναι μια συλλογή από σελίδες, εικόνες, βίντεο και άλλα ψηφιακά στοιχεία, τα οποία φιλοξενοούνται στην ίδια περιοχή (domain name) του Παγκόσμιου Ιστού. Βασίζεται στην υπηρεσία www, μια από τις υπηρεσίες που παρέχονται στο Διαδίκτυο, με τη χρησιμοποίηση του πρωτοκόλλου http. Η υπηρεσία αυτή δίνει τη δυνατότητα στους χρήστες του διαδικτύου να δημιουργήσουν οποιοδήποτε είδους περιεχόμενο στις ιστοσελίδες τους. Το σύνολο των ιστοτόπων αποτελεί το world wide web.

### 2.3.2 Σχεδιασμός Σελίδων Ιστού (Web Design)

Ο σχεδιασμός στο Διαδίκτυο αφορά την ικανότητα δημιουργίας παρουσιάσεων περιεχομένου (συνήθως κειμένου ή πολυμέσων) οι οποίες φτάνουν στον τελικό – χρήστη μέσω του Παγκοσμίου Ιστού, με τη χρήση λογισμικού όπως ένας φυλλομετρητής (web browser) ή άλλου λογισμικού σχεδιασμένου για το διαδίκτυο όπως η τηλεόραση μέσω διαδικτύου, κινητών τηλεφώνων κλπ.

Η ιστοσελίδα είναι ένα ηλεκτρονικό αρχείο ή ένα σύνολο από ηλεκτρονικά αρχεία που υπάρχουν σε έναν ή και περισσότερους εξυπηρετητές (server/servers) και παρουσιάζει κείμενα και εφαρμογές πολυμέσων στον τελικό – χρήστη. Τέτοια στοιχεία όπως κείμενο, εικόνες (σύμφωνα με τα πρότυπα SVG, BMP, GIF, JPEG, PNG) και φόρμες μπορούν να τοποθετηθούν στη σελίδα με τη χρήση γλωσσών σήμανσης υπερκειμένου όπως HTML/XHTML. Η αναπαραγωγή πιο σύνθετων πολυμέσων (ανυσματικών γραφικών, βίντεο, ήχων, γραφικών με ενσωματωμένο ήχο και εικόνα) μπορεί να γίνει με πρόσθετα (plug-ins).

Οι καινούργιες εκδόσεις των προγραμμάτων περιήγησης που ακολουθούν τα W3C πρότυπα οδήγησαν σε μια ευρεία αποδοχή και χρήση των XHTML/XML σε συνδυασμό με τα CSS (Cascading Style Sheets) για την τοποθέτηση και διαχείριση των στοιχείων και αντικειμένων της ιστοσελίδας. Τα τελευταία πρότυπα στοχεύουν στο να αποκτήσουν τα προγράμματα περιήγησης την



δυνατότητα να προσφέρουν μια ευρεία γκάμα επιλογής πολυμέσων και πρόσβασης στους πελάτες χωρίς τη χρήση των προσθέτων.

Για τη διαδικασία σχεδιασμού μιας ιστοσελίδας, μιας εφαρμογής ή ενός πολυμέσου για το διαδίκτυο μπορεί να συνδυάζονται πολλοί κλάδοι όπως animation, συγγραφή, επικοινωνιακός σχεδιασμός, εταιρική ταυτότητα, σχεδιασμός γραφικών, αλληλεπίδραση ανθρώπου – υπολογιστή, αρχιτεκτονική υπολογιστών, σχεδιασμός αλληλεπίδρασης, marketing, φωτογραφία, βελτιστοποίηση μηχανών αναζήτησης και τυπογραφία.

Τα εργαλεία και οι τεχνολογίες που χρησιμοποιούνται περιλαμβάνουν μεταξύ άλλων:

- Γλώσσες σήμανσης: HTML, XHTML, XML
- Γλώσσες φύλλων στυλ: CSS, XML
- Σενάρια στον πελάτη (Client – side scripting): Javascript
- Σενάρια στον εξυπηρετητή (Server – side scripting): PHP, ASP
- Τεχνολογίες βάσεων δεδομένων: MySQL, Microsoft SQL Server
- Τεχνολογίες πολυμέσων: Flash

Ο σκοπός του σχεδιασμού στο Διαδίκτυο είναι πολυσύνθετος αλλά βασικός κατά την εφαρμογή του. Πριν τη δημιουργία και το ανέβασμα μιας ιστοσελίδας είναι σημαντικό να σχεδιαστεί ακριβώς ότι είναι αναγκαίο για την ιστοσελίδα. Είναι πολύ σημαντικό να αποφασιστούν, το κοινό στο οποίο θα απευθύνεται, ο σκοπός της δημιουργίας της και το περιεχόμενό της.

Ο σχεδιασμός που αφορά το διαδίκτυο είναι κατά μία έννοια παρόμοιος με την κλασική τυπογραφία. Κάθε ιστοσελίδα είναι ένα σύνολο αναπαραγόμενων πληροφοριών, όπως ένα βιβλίο και κάθε σελίδα της ιστοσελίδας είναι το αντίστοιχο της σελίδας ενός βιβλίου. Ωστόσο στο σχεδιασμό χρησιμοποιείται ένα πλαίσιο εργασίας βασισμένο σε ένα ψηφιακό κώδικα και σε μια τεχνολογία απεικονίσεων για να δημιουργήσει και να διατηρήσει ένα περιβάλλον που θα διανέμει την πληροφορία σε πολλαπλές μορφές αρχείων όπως pdf, html, doc, κλπ. Οι δυνατότητες του σχεδιασμού το καθιστούν αδιαμφισβήτητα ως τον πλέον κομψό και σύνθετα αναπτυσσόμενο τρόπο επικοινωνίας στο σημερινό κόσμο.

### 2.3.3 Στατικές Ιστοσελίδες

Οι στατικές ιστοσελίδες είναι ουσιαστικά απλά ηλεκτρονικά “έγγραφα”. Μπορούν να περιέχουν κείμενα, φωτογραφίες, συνδέσμους (links), αρχεία για κατέβασμα κινούμενα σχέδια (animated graphics). Είναι κατάλληλες κυρίως για την δημιουργία μόνιμων/στατικών παρουσιάσεων, όπου δεν υπάρχει συχνά η ανάγκη να τροποποιείται το περιεχόμενό τους. Δηλαδή, θα μπορούσε να

χρησιμοποιηθεί μια στατική ιστοσελίδα για την γενική παρουσίαση μιας επιχείρησης και των προϊόντων ή υπηρεσιών της.

Ο χαρακτηρισμός “στατική”, δεν σημαίνει ότι τίποτα στην σελίδα δεν κουνιέται, αντιθέτως μπορεί να έχει κινούμενα γραφικά και άλλα μπιχλιμπίδια. Σημαίνει ότι το περιεχόμενο της κάθε σελίδας είναι σταθερό και συγκεκριμένο. Πρόκειται δηλαδή για σταθερά ηλεκτρονικά έγγραφα.

Σήμερα κυριαρχεί μια λανθασμένη εντύπωση ότι οι στατικές ιστοσελίδες είναι ξεπερασμένες, κυρίως επειδή για να κατασκευαστούν βασίζονται σε παλαιότερες τεχνολογίες. Αυτή η εντύπωση όμως είναι άκρως λανθασμένη. Μερικές από τις πιο μοντέρνες ιστοσελίδες είναι στατικές, ενώ οι στατικές ιστοσελίδες προσφέρουν πλεονεκτήματα που δεν έχουν οι δυναμικές. Ο όρος “στατικές” αναφέρεται στο μόνιμο περιεχόμενο των στατικών ιστοσελίδων και όχι σε κάποιο είδος κίνησης. Το κύριο μειονέκτημά τους είναι ότι δεν είναι πρακτική/βολική η χρήση τους όταν χρειάζεται το περιεχόμενο της ιστοσελίδας να αλλάζει συχνά, π.χ. τουλάχιστον μία φορά την εβδομάδα να προστίθεται κάποιο άρθρο, ή να αλλάζουν τα προϊόντα κτλ. Αυτό συμβαίνει γιατί για να τροποποιηθεί μια στατική ιστοσελίδα, πρέπει να γίνουν επεμβάσεις απευθείας πάνω στην ιστοσελίδα, με κάποιο σχετικό πρόγραμμα (html editor).

Οι πρώτες ιστοσελίδες πλησιάζουν περισσότερο τις σημερινές στατικές. Ο Βρετανός Τιμ Μπέρνες Λι δημοσίευσε τον Αύγουστο του 1991 αυτό που θεωρούμε σήμερα ως την πρώτη ιστοσελίδα, ή οποία ήταν γραμμένη σε html, δηλαδή στη γλώσσα όπου γράφονται οι στατικές ιστοσελίδες.

#### 2.3.4 Δυναμικές Ιστοσελίδες

Οι δυναμικές ιστοσελίδες, μπορεί στην εμφάνιση, σε πολλές περιπτώσεις, να μην έχουν μεγάλη διαφορά με τις στατικές, όμως οι δυνατότητές είναι πολύ περισσότερες, από πολλές πλευρές, καθώς στην περίπτωση αυτή ουσιαστικά πρόκειται για μία εφαρμογή (πρόγραμμα), και όχι ένα απλό ηλεκτρονικό έγγραφο.

Συνήθως, οι δυναμικές ιστοσελίδες, χρησιμοποιούν κάποια βάση δεδομένων (database), όπου αποθηκεύουν πληροφορίες και απ' όπου αντλούν το περιεχόμενό τους, ανάλογα με το τι ζητάει ο χρήστης/επισκέπτης σε κάθε του “κλικ”. Η χρήση των βάσεων δεδομένων, είναι αυτή που επιτρέπει την εύκολη προσθαφαίρεση περιεχομένου στις δυναμικές ιστοσελίδες, καθώς δεν απαιτείται να επεξεργάζεται κανείς κάθε φορά την ίδια την ιστοσελίδα, αλλά απλά να διαχειρίζεται έμμεσα το περιεχόμενο στην βάση δεδομένων και οι υπόλοιπες διαδικασίες γίνονται αυτοματοποιημένα από τον “μηχανισμό” της ιστοσελίδας.

Για την εύκολη διαχείριση του περιεχομένου σε μια δυναμική ιστοσελίδα υπάρχει πρόσθετα στην ιστοσελίδα ένας εύχρηστος μηχανισμός CMS (Content Management System), μέσω του οποίου η προσθαφαίρεση περιεχομένου μπορεί να γίνει ακόμη και από κάποιον αρχάριο. Φυσικά, η “περιοχή διαχείρισης” της ιστοσελίδας, προστατεύεται με κωδικό πρόσβασης, και δεν μπορούν να εισέλθουν σε αυτή οι επισκέπτες της ιστοσελίδας.

### 2.3.5 Διαφορές Δυναμικών & Στατικών Ιστοσελίδων

Γενικά, αυτό που προσφέρουν οι δυναμικές ιστοσελίδες, είναι μεγαλύτερη αλληλεπίδραση του χρήστη με την σελίδα π.χ. να προσθέτει τα σχόλια του στην σελίδα, αλλά και πολλές ευκολίες στον διαχειριστή του περιεχομένου της ιστοσελίδας π.χ. τον ιδιοκτήτη της σελίδας. Όλες οι ιστοσελίδες είναι ένα ηλεκτρονικό αρχείο ή ένα σύνολο από ηλεκτρονικά αρχεία αποθηκευμένα σε σκληρούς δίσκου που λέγονται servers.

Τα ηλεκτρονικά αρχεία μπορεί να είναι εικόνες τα οποία τοποθετούνται στη σελίδα μας, με τη χρήση των κατάλληλων κωδικών σήμανσης, ώστε να αναγνωρίζονται από τον περιηγητή μας, ο οποίος με τη σειρά του θα μας τα απεικονίσει στην οθόνη μας. Οι κατάλληλοι κωδικοί σήμανσης όπως έχουν προαναφερθεί είναι html/xhtml/xml. Βέβαια σήμερα τα πράγματα δεν είναι τόσο απλά. Έχουν εξελιχθεί σε μεγάλο βαθμό. Η απλή γλώσσα σήμανσης html, έχει συνδυαστεί με εφαρμογές Javascript και με CSS, και έτσι τα πράγματα έχουν γίνει πολύ πιο περίπλοκα.

Η στατική ιστοσελίδα έχει πλεονεκτήματα που δεν έχουν οι δυναμικές και το αντίθετο. Για αυτό δεν μπορούμε να πούμε ότι η μία είναι καλύτερη από την άλλη. Η βασική διαφορά είναι ότι οι στατικές ιστοσελίδες δεν αλληλεπιδρούν με τον επισκέπτη, πράγμα που κάνουν οι δυναμικές.

Στα υπέρ τη στατικής σελίδας είναι το χαμηλό κόστος κατασκευής και συντήρησης, γιατί μια στατική σελίδα, δεν απαιτεί μεγάλο χώρο σε φιλοξενία από webserver. Μια στατική σελίδα αναπτύσσεται/σχεδιάζεται γρηγορότερα από ότι μια δυναμική. Επίσης οι στατικές σελίδες φορτώνουν πιο γρήγορα από ότι οι δυναμικές.

Μεγάλο πλεονέκτημα των στατικών ιστοσελίδων έναντι των δυναμικών είναι επίσης ότι πιο SEO friendly. Το SEO σημαίνει Search engine optimization και στα ελληνικά “Βελτιστοποίηση Σελίδας για τις μηχανές αναζήτησης”. Οι στατικές σελίδες λοιπόν είναι πιο φιλικές προς τις μηχανές αναζήτησης. Αυτό συμβαίνει γιατί στις στατικές ιστοσελίδες υπάρχει δυνατότητα τοποθέτησης στον κώδικα html, των meta tags τα οποία αναγνωρίζουν οι μηχανές αναζήτησης και κατατάσσουν την στατική σελίδα πάνω από μια δυναμική που δεν έχει ακριβώς αυτή την δυνατότητα. Αυτό είναι πολύ ισχυρό πλεονέκτημα των στατικών



ιστοσελιδών, γιατί όπως καταλαβαίνετε έχει άμεση σχέση με την επισκεψιμότητά τους. Και η επισκεψιμότητα σε έναν ιστότοπο μπορεί να μεταφραστεί σε κέρδος. Και όπως είναι γνωστό, οι μηχανές αναζήτησης είναι αυτές που παραπέμπουν τους χρήστες στις ιστοσελίδες. Οπότε όσο καλύτερα κατεταγμένη η σελίδα στις μηχανές αναζήτησης σε διάφορες λέξεις – κλειδιά, τόσο μεγαλύτερο πλεονέκτημα έχει έναντι των ανταγωνιστών της.

Στα κατά μιας στατικής ιστοσελίδας είναι ότι ο χρήστης πρέπει να έχει γνώσεις προγραμματισμού, για να μπορεί να ανανεώσει το περιεχόμενο της σελίδας. Κι αυτό γιατί θα πρέπει να επεμβαίνει στον κώδικα. Σε μια στατική ιστοσελίδα ο χρήστης δεν έχει κανένα είδους αλληλεπίδραση με τη σελίδα. Κάποιος θα επέλεγε μια στατική ιστοσελίδα αν δεν έχει σκοπό να ανανεώνει συχνά τις σελίδες, οι οποίες δεν θα ξεπερνούν τις 15-20.

Αντίθετα σε μια δυναμική σελίδα, ο χρήστης πρέπει απλά να ξέρει πως να διαχειρίζεται το περιεχόμενο στη βάση δεδομένων και όλα τα υπόλοιπα γίνονται αυτοματοποιημένα από το πρόγραμμα. Οι δυναμικές ιστοσελίδες είναι εφαρμογές – προγράμματα. Για την δημιουργία ενός δυναμικού site χρειάζονται πολλές – πολλές γραμμές κώδικα προγραμματισμού. Παρόλο όμως που είναι σχετικά δύσκολο να κατασκευαστεί ένα δυναμικό site, είναι εύκολο να διαχειριστεί και να ανανεωθεί. Εδώ είναι και το μεγάλο πλεονέκτημα, έναντι των στατικών. Για να μπορεί ένας απλός χρήστης – επισκέπτης να διαχειρίζεται (να το αλλάζει, να το τροποποιεί ή να το διαγράφει) το περιεχόμενο μιας δυναμικής σελίδας εύκολα, χωρίς γνώσεις προγραμματισμού.

Κάποιος θα επιλέξει μια δυναμική ιστοσελίδα αν θέλει να έχει μια δυναμική παρουσία στο διαδίκτυο. Αν έχει σκοπό να ανανεώνει συχνά το περιεχόμενό της και να θέλει να έχει απεριόριστο αριθμό διαφορετικών σελίδων. Στα υπέρ είναι το περιεχόμενό της μπορεί να ανανεωθεί και να τροποποιηθεί από ανθρώπους που δεν γνωρίζουν προγραμματισμό, σε αντίθεση με τις στατικές, όπου αυτό καθιστάτε αδύνατον. Επίσης σε μια δυναμική ιστοσελίδα είναι πολύ εύκολη η διαχείριση χρηστών, διαχειριστών και μελών της ιστοσελίδας.

## 2.4 Γλώσσες Προγραμματισμού

Γλώσσα προγραμματισμού λέγεται μια τεχνητή γλώσσα που μπορεί να χρησιμοποιηθεί για τον έλεγχο μιας μηχανής, συνήθως ενός υπολογιστή. Οι γλώσσες προγραμματισμού (όπως άλλωστε και οι ανθρώπινες γλώσσες) ορίζονται από ένα σύνολο συντακτικών και εννοιολογικών κανόνων, που ορίζουν τη δομή και το νόημα, αντίστοιχα, των προτάσεων της γλώσσας.

Οι γλώσσες προγραμματισμού χρησιμοποιούνται για να διευκολύνουν την οργάνωση και διαχείριση πληροφοριών, αλλά και για την ακριβή διατύπωση αλγορίθμων. Ορισμένοι ειδικοί χρησιμοποιούν τον όρο γλώσσα προγραμματισμού μόνο για τυπικές γλώσσες που μπορούν να

εκφράσουν όλου τους πιθανούς αλγορίθμους. Μη – υπολογιστικές γλώσσες όπως η HTML ή τυπικές γραμματικές όπως η BNF δεν λέγονται συνήθως γλώσσες προγραμματισμού.

#### 2.4.1. Χαρακτηριστικά & Κατηγοριοποίηση Γλώσσών

Υπάρχουν χιλιάδες διαφορετικές γλώσσες προγραμματισμού, και κάθε χρόνο δημιουργούνται περισσότερες. Κάθε γλώσσα προγραμματισμού έχει το δικό της σύνολο τυπικών προδιαγραφών (ή κανόνων) που αφορούν το συντακτικό, το λεξιλόγιο και το νόημα της. Για πολλές γλώσσες που χρησιμοποιούνται ευρέως και έχουν χρησιμοποιηθεί για αρκετό καιρό χρονικό διάστημα, υπάρχουν ειδικοί φορείς τυποποίησης, οι οποίοι μέσα από τακτές συναντήσεις δημιουργούν, τροποποιούν ή επεκτείνουν τις τυπικές προδιαγραφές που διέπουν τη χρήση μιας γλώσσας προγραμματισμού. Άλλες γλώσσες δεν περιγράφονται σε κάποιο επίσημο πρότυπο αλλά ορίζονται μόνο με βάση κάποια υλοποίησή τους.

Δεν υπάρχει απλός τρόπος να κατηγοριοποιηθούν οι γλώσσες προγραμματισμού. Αυτό συμβαίνει γιατί συνήθως κάθε γλώσσα προγραμματισμού περιέχει επιρροές από πολλές προηγούμενες γλώσσες, συνδυάζοντας θετικά στοιχεία και προσθέτοντας νέα. Χαρακτηριστικά που εμφανίζονται σε μια γλώσσα και έχουν θετική αποδοχή, συνήθως υιοθετούνται από εμταγενέστερες γλώσσες ακόμα και αν πρόκειται για γλώσσες που ανήκουν σε διαφορετική κατηγορία.

Η κατηγοριοποίηση είναι ακόμα πιο περίπλοκη για το λόγο ότι πολλές γλώσσες συνήθως ανήκουν σε παραπάνω από μία κατηγορίες. Για παράδειγμα, η Java είναι τόσο αντικειμενοστρεφής όσο και παράλληλη γλώσσα, δεδομένου ότι υποστηρίζει την οργάνωση των δεδομένων και υπολογισμών σε αντικείμενα, αλλά επιτρέπει επίσης και την δημιουργία προγραμμάτων με ταυτόχρονα νήματα (threads) που εκτελούνται παράλληλα.

Δεδομένης της δυσκολίας στην κατηγοριοποίηση, μπορούμε να κατηγοριοποιήσουμε τις γλώσσες προγραμματισμού με διάφορους τρόπους. Οι συνηθέστεροι τρόποι είναι:

- με βάση τον τρόπο οργάνωσης του προγράμματος
- με βάση τον στόχο που έχει η γλώσσα
- με βάση τον τρόπο που περιγράφουν το ζητούμενο αποτέλεσμα

Στην πρώτη περίπτωση προκύπτουν κατηγορίες όπως:

- Διαδικαστικές γλώσσες (procedural) όπου το πρόγραμμα είναι οργανωμένα σε διαδικασίες, που αποτελούνται από σειρές εντολών που περιγράφουν αλγορίθμους.

- Αντικειμενοστρεφείς γλώσσες (object – oriented) όπου το πρόγραμμα είναι οργανωμένο σε αντικείμενα. Ένα αντικείμενο είναι μια μονάδα που αποτελείται από την περιγραφή κάποιων

δεδομένων και την περιγραφή των αλγορίθμων που τα επεξεργάζονται. Ένα αντικειμενοστρεφές πρόγραμμα αποτελείται από διάφορα αντικείμενα που αλληλεπιδρούν μεταξύ τους.

- Συναρτησιακές γλώσσες (functional) όπου οι υπολογισμοί εκφράζονται ως εφαρμογές μαθηματικών συναρτήσεων, σε αντίθεση με τα άλλα είδη προγραμματισμού όπου οι υπολογισμοί εκφράζονται ως σειρές εντολών, όπου η κάθε μία αλλάζει με κάποιο τρόπο την κατάσταση του συστήματος. Θεωρητικό τους υπόβαθρο είναι ο λ-λογισμός.

Στην περίπτωση που η κατηγοριοποίηση των γλωσσών προγραμματισμού γίνει με βάση το στόχο που έχει η γλώσσα, υπάρχουν οι παρακάτω κατηγορίες:

- Γλώσσες γενικής χρήσης. Σε αυτήν την κατηγορία ταξινομούνται γλώσσες που δημιουργήθηκαν για τον προγραμματισμό γενικών εφαρμογών, καθώς και πολλές εκπαιδευτικές γλώσσες που αποδείχθηκαν χρήσιμες για την ανάπτυξη γενικών εφαρμογών.

- Γλώσσες προγραμματισμού συστημάτων, που χρησιμοποιούνται συνήθως για τον προγραμματισμό λειτουργικών συστημάτων ή οδηγών (drivers) υλικού, όπου χρειάζεται πολλές φορές ο προγραμματιστής να έχει έλεγχο και γνώση του πώς λειτουργεί το υλικό.

- Γλώσσες Σεναρίων (scripting). Αυτές οι γλώσσες χρησιμοποιούνται συνήθως για τη γρήγορη ανάπτυξη μικρών προγραμμάτων, σε περιπτώσεις που ο χρόνος του προγραμματιστή είναι πιο πολύτιμος από την ταχύτητα εκτέλεσης του προγράμματος, όπως για παράδειγμα συμβαίνει όταν το πρόγραμμα απλά αυτοματοποιεί απλές λειτουργίες.

- Γλώσσες ειδικών εφαρμογών. Σε αυτή την κατηγορία ανήκουν γλώσσες που αναπτύχθηκαν ειδικά για μια συγκεκριμένη εφαρμογή.

- Παράλληλες ή κατανεμημένες γλώσσες. Στη συγκεκριμένη κατηγορία ταξινομούνται γλώσσες που επιτρέπουν την ανάπτυξη παράλληλων προγραμμάτων, όπου πολλές εντολές εκτελούνται ταυτόχρονα σε πολλούς υπολογιστές, έτσι ώστε το τελικό αποτέλεσμα να προκύψει γρηγορότερα. Οι παράλληλες γλώσσες προσφέρουν συνήθως εύκολους τρόπους επικοινωνίας μεταξύ των νημάτων που εκτελούνται παράλληλα, καθώς και τρόπους ώστε να δημιουργούνται καινούργιες παράλληλες εκτελέσεις.

Τέλος, στην περίπτωση που η κατηγοριοποίηση γίνεται με βάση τον τρόπο που περιγράφεται το ζητούμενο, υπάρχουν οι παρακάτω κατηγορίες:

- Προστακτικές γλώσσες προγραμματισμού (imperative) είναι οι γλώσσες που περιγράφουν το ζητούμενο αποτέλεσμα κατασκευαστικά, δίνοντας μια σειρά εντολών που όταν εκτελεστούν παράγουν το ζητούμενο αποτέλεσμα.

- Δηλωτικές γλώσσες προγραμματισμού (declarative) είναι οι γλώσσες που περιγράφουν το ζητούμενο αποτέλεσμα χρησιμοποιώντας τις ιδιότητες που έχει, και όχι τον τρόπο με τον οποίο υπολογίζεται.

## 2.5 Η Γλώσσα HTML



Η HTML (ακρωνύμιο του αγγλικού HyperText Markup Language – Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα βασικά δομικά στοιχεία των ιστοσελίδων. Δημιουργήθηκε στο ερευνητικό κέντρο φυσική CERN που βρίσκεται στην Ελβετία, και επιτρέπει τη δημιουργία κειμένου με τη μορφή που επιθυμούμε, την προσθήκη γραφικών, ήχων, και βίντεο από ξεχωριστά αρχεία, και την αποθήκευση όλων αυτών των στοιχείων σε ένα απλό αρχείο ASCII με προέκταση .html ή .htm. Ένα μεγάλο πλεονέκτημα που διαθέτει ένα αρχείο HTML είναι ότι μπορεί να περιλαμβάνει συνδέσμους υπερμέσων που να οδηγούν σε άλλα έγγραφα HTML ή άλλης μορφής. Το αρχείο αυτό μπορεί να διαβαστεί από οποιονδήποτε υπολογιστή διαθέτει ένα ειδικό πρόγραμμα περιήγησης (ή φυλλομετρητής) του Παγκόσμιου Ιστού (Web browser). Τέτοια προγράμματα είναι ο Internet Explorer της εταιρείας Microsoft, ο Netscape Communicator κ.α. Τα έγγραφα μορφής HTML που βρίσκονται στον Παγκόσμιο Ιστό ονομάζονται ιστοσελίδες.

### 2.5.1 Βασική Οργάνωση ενός Εγγράφου HTML

Για να δημιουργήσουμε ένα έγγραφο HTML αρκεί να χρησιμοποιήσουμε έναν απλό συντάκτη κειμένου όπως είναι π.χ. το Σημειωματάριο των Windows.

Το έγγραφο HTML ξεκινά πάντα με την <HTML> και κλείνει με την </HTML>. Το έγγραφο περιέχει δύο βασικές ενότητες, την ενότητα head και την ενότητα body. Στην ενότητα head τοποθετούνται γενικές πληροφορίες για την ιστοσελίδα, όπως π.χ. η προαιρετική ετικέτα title στην οποία τοποθετούμαι το κείμενο που θα εμφανίζεται στην γραμμή τίτλου του προγράμματος περιήγησης όταν φορτωθεί το έγγραφο HTML. Η ενότητα body αποτελεί το κυρίως σώμα του εγγράφου της ιστοσελίδας, και περιέχει την πλειονότητα των ετικετών του εγγράφου.

Στην οργάνωση της ιστοσελίδας, πολλές φορές χρειάζεται να χρησιμοποιήσουμε επικεφαλίδες, να αλλάξουμε γραμμή, να ξεκινήσουμε νέα παράγραφο, ή να τοποθετήσουμε σχόλια. Αν χρειαστεί να προστεθούν σχόλια στο έγγραφο HTML, αυτά τοποθετούνται ανάμεσα στις ετικέτες <!-- -->.

```
<!DOCTYPE html>
<html>
<body>
```

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>

</body>
</html>
```

Your Result:

**This is heading 1**

**This is heading 2**

**This is heading 3**

**This is heading 4**

**This is heading 5**

**This is heading 6**

2.2.2.α Μορφοποίηση Κειμένου &

Εισαγωγή Εικόνων

Με τη χρήση μιας ομάδας ετικετών της γλώσσας HTML μπορούμε να προσδώσουμε στο κείμενό μας διάφορα χαρακτηριστικά. Οι σημαντικότερες από τις ετικέτες μορφοποίησης είναι:

Κείμενο	Ετικέτα
Κεντραρισμένο	<CENTER></CENTER>
Έντονο	<B></B>
Πλάγιο	<I></I>
Υπογραμμισμένο	<U></U>
Διακριτή γραφή	<STRIKE></STRIKE>
Με έμφαση	<EM></EM>
Εκθέτης	<SUP></SUP>
Δείκτης	<SUB></SUB>
Προμορφοποιημένο	<PRE></PRE>
Μεγάλα γράμματα	<BIG></BIG>
Μικρά γράμματα	<SMALL></SMALL>

```
<!DOCTYPE html>
<html>
```



```
<body>
```

```
<p><b>This text is bold</b></p>
```

```
<p><strong>This text is strong</strong></p>
```

```
<p><big>This text is big</big></p>
```

```
<p><i>This text is italic</i></p>
```

```
<p><em>This text is emphasized</em></p>
```

```
<p><code>This is computer output</code></p>
```

```
<p>This is<sub> subscript</sub> and
```

```
<sup>superscript</sup></p>
```

```
</body>
```

```
</html>
```

Your Result:

**This text is bold**

**This text is strong**

**This text is big**

*This text is italic*

*This text is emphasized*

This is computer output

This is <sub>subscript</sub> and <sup>superscript</sup>

Για να εισάγουμε μια εικόνα σε ένα έγγραφο HTML

χρησιμοποιούμε την ετικέτα <IMG>, η οποία περιλαμβάνει αρκετές ιδιότητες όπως οι src, align, alt, hspace, vspace, width, height. Η πιο βασική ιδιότητα είναι η src η οποία προσδιορίζει το όνομα και τη θέση της εικόνας που θα τοποθετηθεί στο έγγραφο. Έτσι, για να εισάγουμε στο έγγραφό μας μια εικόνα θα γράψουμε:

```
<img src = "tei.gif">
```

Μπορούμε να καθορίσουμε τις διαστάσεις της εικόνας ώστε το πρόγραμμα περιήγησης να δημιουργήσει το περίγραμμά της και να εμφανίσει μέσα σε αυτό κάποιο κείμενο έως ότου η εικόνα φορτωθεί. Οι τιμές των ιδιοτήτων width και height καθορίζουν το μέγεθος του πλαισίου της εικόνας σε πίξελ.

```
<img src = "tei.gif" width="100" height="100">
```

Αν θέλουμε να δημιουργήσουμε εναλλακτικό κείμενο το οποίο θα εμφανίζεται στο πλαίσιο της εικόνας σε περίπτωση που για κάποιο λόγο δεν φορτωθεί ή ίδια η εικόνα, χρησιμοποιούμε την ιδιότητα alt με τιμή το επιθυμητό κείμενο.

```
<img src = "tei.gif" width="100" height="100" alt="Tei Spartis">
```

Αν θέλουμε το κείμενο να ανδισπλώνεται από τη μια πλευρά της εικόνας χρησιμοποιούμε την ιδιότητα align, η οποία παίρνει τις τιμές left και right και τοποθετεί την εικόνα αριστερά και δεξιά αντίστοιχα.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>An image
```

```

with align="bottom".</p>

<p>An image

with align="middle".</p>

<p>An image

with align="top".</p>

<p><b>Tip:</b> align="bottom" is default!</p>

<p>
An image before the text.</p>

<p>An image after the text.
</p>

</body>
</html>
```

Your Result:

An image  with align="bottom".

An image  with align="middle".

An image  with align="top".

**Tip:** align="bottom" is default!

 An image before the text.

An image after the text. 

### 2.2.2.β Σύνδεσμοι & Αγκυρώσεις

Ο παγκόσμιος Ιστός αποτελείται από έγγραφα υπερμέσων. Ένα έγγραφο υπερμέσων εκτός από κείμενο περιέχει και άλλου είδους στοιχεία, όπως εικόνες, κινούμενες ή μη, ήχους και βίντεο. Επιπλέον, όμως, περιέχει και συνδέσμους που οδηγούν σε άλλα μέρη του ίδιου εγγράφου, ή σε άλλα έγγραφα. Με τη βοήθεια των συνδέσμων, το πρόγραμμα περιήγησης του χρήστη μπορεί να φορτώσει και να παρουσιάσει νέες ιστοσελίδες.

Για να δημιουργήσουμε έναν υπερσύνδεσμο χρησιμοποιούμε την ετικέτα `<A></A>`. Οι βασικές της ιδιότητες είναι `href` και `name`. Η πρώτη καθορίζει ότι η αγκύρωση είναι η αρχή ενός υπερσυνδέσμου. Η τιμή της ιδιότητας `href` είναι η ιστοσελίδα προορισμού.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<a
```

```
href="http://www.spartiteikal.gr" target="_blank">Visit Tei Spartis!</a>
```

```
</body>
```

```
</html>
```

Your Result:

[Visit Tei Spartis!](http://www.spartiteikal.gr)

Εάν ένα έγγραφο είναι μεγάλο ή περιέχει πολλές ενότητες, μπορούμε να δημιουργήσουμε εσωτερικές αγκυρώσεις με τη χρήση των ιδιοτήτων αυτών.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>
```

```
<a href="#C4">See also Chapter 4.</a>
```

```
</p>
```

```
<h2>Chapter 1</h2>
```

```
<p>This chapter explains ba bla bla</p>
```

```
<h2>Chapter 2</h2>
```

```
<p>This chapter explains ba bla bla</p>
```

```
<h2>Chapter 3</h2>
<p>This chapter explains ba bla bla</p>
```

```
<h2><a name="C4">Chapter 4</a></h2>
<p>This chapter explains ba bla bla</p>
```

```
</body>
```

```
</html>
```

Your Result:

[See also Chapter 4.](#)

### Chapter 1

This chapter explains ba bla bla

### Chapter 2

This chapter explains ba bla bla

### Chapter 3

This chapter explains ba bla bla

### Chapter 4

This chapter explains ba bla bla

## 2.2.2.γ Λίστες & Πίνακες

Υπάρχουν πολλές περιπτώσεις όπου θέλουμε να παραθέσουμε διατεταγμένα στοιχεία ή να κάνουμε κάποια στοιχεία πιο ευανάγνωστα. Για το σκοπό αυτό χρησιμοποιούμε τις διάφορες ετικέτες λίστας.

Για να παραθέσουμε κάποια στοιχεία ως μη διατεταγμένη λίστα, χρησιμοποιούμε τις ετικέτες `<ul></ul>` σε συνδυασμό με τις ετικέτες `<li></li>`. Εάν θέλουμε η λίστα να περιλαμβάνει διατεταγμένα στοιχεία χρησιμοποιούμε ετικέτες `<ol></ol>`.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h4>An Unordered List:</h4>
```

```
<ul>
```

```
<li>Coffee</li>
```

```
<li>Tea</li>
```

```
<li>Milk</li>
```

```

</ul>

</body>
</html>

<!DOCTYPE html>
<html>
<body>

<h4>An Ordered List:</h4>
<ol>
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ol>

</body>
</html>

```

Ο πίνακας είναι ένα πολύ χρήσιμο εργαλείο για την παρουσίαση των δεδομένων σε οργανωμένη μορφή. Η δημιουργία ενός πίνακα γίνεται με την ετικέτα `<table></table>` η οποία έχει αρκετές ιδιότητες, μερικές από τις οποίες είναι:

- **align.** Καθορίζει την οριζόντια στοίχιση του πίνακα.
- **Border.** Καθορίζει το πάχος σε πίξελ που θα έχει το περίγραμμα του πίνακα
- **width.** Καθορίζει το πλάτος που θα έχει ο πίνακας, είτε σε πίξελ είτε ως ποσοστό του συνολικού μήκους της γραμμής.
  - **Height.** Καθορίζει το ύψος που θα έχει ο πίνακας σε πίξελ.
  - **Nowrap.** Η ιδιότητα αυτή χρησιμοποιείται όταν θέλουμε να διατηρείται όλο το κείμενο του κελιού στην ίδια γραμμή.
  - **Cellspacing.** Η τιμή που παίρνει η ιδιότητα αυτή προσδιορίζει το διάστημα σε πίξελ που θέλουμε να υπάρχει ανάμεσα στα κελιά του πίνακα. Η προεπιλεγμένη τιμή είναι 2 πίξελ.
  - **Cellpadding.** Η τιμή που παίρνει η ιδιότητα αυτή προσδιορίζει το διάστημα σε πίξελ ανάμεσα στα περιεχόμενα και στα τοιχώματα του κελιού. Η προεπιλεγμένη τιμή είναι 1 πίξελ.

Για να δημιουργήσουμε γραμμές μέσα στον πίνακα χρησιμοποιούμε την ετικέτα `<tr></tr>`. Η ετικέτα αυτή διαθέτει τις ιδιότητες `align`, `nowrap`, `valign`, οι οποίες λειτουργούν όπως και στην

ετικέτα <table> όσον αφορά τη στοίχιση και την τοποθέτηση του κειμένου ελαφρά πιο έντονο και πιο μεγάλο κείμενο από το κανονικό.

Για να κατασκευάσουμε επικεφαλίδες σε έναν πίνακα χρησιμοποιούμε τις ετικέτες <th></th> και να κατασκευάσουμε κελιά μέσα στις γραμμές του πίνακα απαραίτητη είναι η χρήση της ετικέτας <td></td>.

Μπορούμε να ενώσουμε κάποιες γραμμές ή κάποιες στήλες ενός πίνακα. Για το σκοπό αυτό χρησιμοποιούμε τις ιδιότητες colspan και rowspan ορίζοντας τον αριθμό των κελιών ή των γραμμών που θέλουμε να ενωθούν.

```
<!DOCTYPE html>
<html>
<body>

<h4>Table headers:</h4>
<table border="1">
<tr>
<th>Name</th>
<th>Telephone</th>
<th>Telephone</th>
</tr>
<tr>
<td>Bill Gates</td>
<td>555 77 854</td>
<td>555 77 855</td>
</tr>
</table>

<h4>Vertical headers:</h4>
<table border="1">
<tr>
<th>First Name:</th>
<td>Bill Gates</td>
</tr>
<tr>
<th>Telephone:</th>
<td>555 77 854</td>
</tr>
<tr>
<th>Telephone:</th>
<td>555 77 855</td>
```

```

</tr>
</table>

</body>
</html>

```

Your Result:

**Table headers:**

Name	Telephone	Telephone
Bill Gates	555 77 854	555 77 855

**Vertical headers:**

<b>First Name:</b>	Bill Gates
<b>Telephone:</b>	555 77 854
<b>Telephone:</b>	555 77 855

#### 2.2.2.δ Φόρμες

Η φόρμα μας επιτρέπει να συγκεντρώνουμε πληροφορίες από το χρήστη για να τις χρησιμοποιήσουμε ή να τις αποθηκεύσουμε. Μια φόρμα δημιουργείται με την ετικέτα `<form></form>` και περιλαμβάνει διάφορα στοιχεία, όπως:

- πλαίσιο κειμένου. Ένα πλαίσιο κειμένου δημιουργείται με την ετικέτα

```
<input type = "text" name ="name" size = "n">
```

- περιοχή κειμένου. Μια περιοχή κειμένου μπορεί να περιέχει πολλές γραμμές κειμένου.

Δημιουργείται με την ετικέτα:

```
<textarea name = "name" rows = "n" cols = "n"></textarea>
```

- ραδιοπλήκτρο. Από μια ομάδα ραδιοπλήκτρων μπορεί να επιλεγεί μόνο το ένα. Η δημιουργία τους γίνεται με την ετικέτα:

```
<input type = "radio" value = "value" checked>
```

- πλαίσια ελέγχου. Ενώ τα ραδιοπλήκτρα δέχονται μόνο μία απάντηση ανά ομάδα, στα πλαίσια ελέγχου μπορούμε να επιλέξουμε όσα από αυτά θέλουμε. Για να δημιουργήσουμε ένα πλαίσιο ελέγχου χρησιμοποιούμε την ετικέτα:

```
<input type = "checkbox" value = "value" checked>
```

• πλαίσια καταλόγου. Όταν θέλουμε να επιλέξουμε μεταξύ πολλών τιμών, χρησιμοποιούμε μια ετικέτα πλαισίου καταλόγου `<select>/select>`. Μέσα στην ετικέτα αυτή τοποθετείται η ετικέτα `<option>`, της οποίας η ιδιότητα `value` περιέχει το αναγνωριστικό κάθε στοιχείου της λίστας.

```
<!DOCTYPE html>
<html>
<body>

<form action="">
<select name="cars">
<option value="volvo">Volvo</option>
<option value="saab">Saab</option>
<option value="fiat">Fiat</option>
<option value="audi">Audi</option>
</select>
</form>

</body>
</html>
```

Your Result:

The image shows a browser window with a dropdown menu. The selected option is 'Volvo'. Below the dropdown, a list of options is visible: Volvo, Saab, Fiat, and Audi.

• Τα κουμπιά υποβολής φόρμας και επαναφοράς. Οι πληροφορίες που συμπληρώνουμε σε μια φόρμα έχουν ξία εφόσον υποβληθούν στο διακομιστή. Για το λόγο αυτόν υπάρχει το κουμπί υποβολής. Εάν έχουμε κάνει λάθος στις πληροφορίες που συμπληρώσαμε, υπάρχει το κουμπία επαναφοράς με το οποίο διαγράφουμε όλα τα στοιχεία που έχουν γραφτεί στα διάφορα στοιχεία της φόρμας. Για να δημιουργήσουμε ένα κουμπί υποβολής γράφουμε:

```
<input type = "submit" value = "message">
```

Για να δημιουργήσουμε ένα κουμπί επαναφοράς γράφουμε:

```
<input type = "reset" value = "message">
```

```
<!DOCTYPE html>
<html>
<body>
```

```
<form action="">
<fieldset>
<legend>Personal information:</legend>
Name: <input type="text" size="30" /><br />
E-mail: <input type="text" size="30" /><br />
Date of birth: <input type="text" size="10" /><br />
<form name="input" action="html_form_action.asp" method="get">
```



```



```

Your Result:

**Personal information:**

Name:

E-mail:

Date of birth:

I have a bike

I have a car

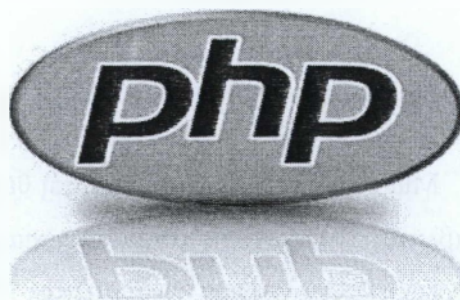
Male

Female

Comment:

## 2.6 Η Γλώσσα PHP

Η php είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα php περνά από επεξεργασία από ένα συμβατό διακομιστή του παγκόσμιου Ιστού, ώστε να παραχθεί σε πραγματικό



χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα html.

Ένα αρχείο με κώδικα php θα πρέπει να έχει την κατάλληλη επέκταση. Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται και να μεταγλωτίζει τον κώδικα php σε html.

### 2.6.1 Δομικά Στοιχεία της PHP

Οι εντολές που υπάρχουν σε μία ενότητα php διαχωρίζονται μεταξύ τους με το ελληνικό ερωτηματικό (;) ακριβώς όπως θα κάναμε στη γλώσσα C.

```
<html>
<head>
<title> Βασική Σελίδα </title>
</head>
<body>
<?
echo "ΓΕΙΑ ΣΑΣ ΑΠΟ ΤΗΝ PHP <br />";
echo "Αυτό είναι το δεύτερο μήνυμα";
?>
</body>
</html>
```

Αν ανοίξουμε την παραπάνω σελίδα θα δούμε στην οθόνη το κείμενο:

ΓΕΙΑ Σας ΑΠΟ ΤΗΝ PHP

Αυτό είναι το δεύτερο μήνυμα.

Αν οι δυνατότητες της php περιορίζονταν στο να γράφει απλώς κείμενο επί της οθόνηςμ τότε δεν θα ήταν περισσότερο χρήσιμη από ότι η απλή html. Ωστόσο πρόκειται για μία γλώσσα προγραμματισμού και σαν τέτοια έχει και μεταβλητές και τελεστές αλλά και δομές ελέγχου.

#### 2.6.1.α Μεταβλητές

Μια μεταβλητή είναι μία ειδική θέση, την οποία ορίζουμε για την αποθήκευση μιας τιμής. Οι μεταβλητές είναι ζωτικής σημασίας για τον προγραμματισμό. Χωρίς τις μεταβλητές, θα ήμασταν υποχρεωμένοι να ενσωματώνουμε στον κώδικα των script όλες τις τιμές που χρειαζόμαστε.

Οι μεταβλητές μας δίνουν τη δυνατότητα να δημιουργούμε “πρότυπα” για την εκτέλεση διαφόρων ενεργειών (π.χ. την πρόσθεση δύο αριθμών) χωρίς να ανησυχούμε για τις τιμές που περιέχουν οι μεταβλητές. Οι τιμές ανατίθενται στις μεταβλητές κατά την εκτέλεση του script μέσω ενός ερωτήματος το οποίο εκτελείται σε μια βάση δεδομένων, ή πιθανώς από το αποτέλεσμα κάποιας προηγούμενης ενέργειας του script.

Μία μεταβλητή αποτελείται από ένα όνομα, το έχει ως πρόθεμα ένα σύμβολο δολαρίου (\$). Τα ονόματα των μεταβλητών μπορούν να περιλαμβάνουν γράμματα, αριθμούς και τον χαρακτήρα της κάτω παύλας (\_). Τα ονόματα των μεταβλητών δεν μπορούν να έχουν κενά διαστήματα. Πρέπει πάντα να ξεκινούν με ένα γράμμα ή με τον χαρακτήρα της κάτω παύλας. Το ακόλουθο απόσπασμα κώδικα ορίζει ορισμένες έγκυρες μεταβλητές:

```
$a;  
$a_longish_variable_name;  
$345;
```

### 2.6.1.β Γενικές και Υπερ-Γενικές Μεταβλητές

Επιπρόσθετα με τους κανόνες ονομασίας των μεταβλητών, υπάρχουν επίσης κανόνες που καθορίζουν την διαθεσιμότητά τους. Γενικά, η τιμή που έχει εκχωρηθεί σε μία μεταβλητή είναι ορατή μόνο μέσα στην συνάρτηση ή το script στο οποίο βρίσκεται. Ωστόσο, υπάρχει η δυνατότητα να οριστεί μια μεταβλητή σαν γενική (global) μέσα σ' ένα script ή μία συνάρτηση.

Η php διαθέτει αρκετές προκαθορισμένες μεταβλητές οι οποίες αποκαλούνται υπερ-γενικές (superglobals). Αυτές οι μεταβλητές είναι πάντα παρούσες και οι τιμές τους είναι διαθέσιμες σε όλα τα script. Κάθε μία από τις υπερ-γενικές μεταβλητές είναι στην πραγματικότητα μια διάταξη (array) άλλων μεταβλητών:

- `$_GET`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script μέσω της μεθόδου GET.
- `$_POST`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script μέσω της μεθόδου POST.
- `$_COOKIE`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script μέσω ενός cookie.
- `$_FILES`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script μέσω μεταφορών αρχείων (file uploads).
- `$_SERVER`: περιέχει πληροφορίες όπως κεφαλίδες (headers), θέσεις αρχείων (διαδρομές καταλόγων) και θέσεις των scripts.
- `$_ENV`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script από το περιβάλλον του server.

- `$_REQUEST`: περιέχει οποιεσδήποτε μεταβλητές παρέχονται σ' ένα script μέσω ενός μηχανισμού, ο οποίος λαμβάνει είσοδο από τον χρήστη.
- `$_SESSION`: περιέχει οποιεσδήποτε μεταβλητές είναι επί του παρόντος δηλωμένες (registered) σε μία σύνοδο (session)

## 2.6.2 Δομές Ελέγχου Ροής

Μία γλώσσα προγραμματισμού θεωρείται νεκρή χωρίς δομές ελέγχου. Οι δομές Ελέγχου πάντα περιορίζονταν σε δύο κατηγορίες: τις δομές επιλογής και τις δομές ανακυκλώσεως. Στις πρώτες αποφασίζουμε για το ποια θα είναι η επόμενη εντολή που θα εκτελεστεί χρησιμοποιώντας μάλιστα λογική συνθήκη. Στις δομές ανακυκλώσεως μια ομάδα εντολών εκτελείται πολλές φορές με βάση μία λογική συνθήκη ή έναν προκαθορισμένο αριθμό επαναλήψεων.

### 2.6.2.α Εντολή IF

Η εντολή `if` αποφασίζει για την επόμενη εντολή που θα εκτελεστεί με την χρήση μια λογικής συνθήκης. Η σύνταξή του είναι:

```
if (condition)
{
group1;
}
```

Που σημαίνει πως αν ισχύει η λογική συνθήκη, τότε εκτελείται η ομάδα εντολών. Διαφορετικά αυτή η ομάδα εντολών δεν εκτελείται.

```
<?php
$mood = "happy";
if ($mood == "happy") {
echo "Hooray, I'm in a good mood!";
}
?>
```

Ο τελεστής σύγκρισης `==` συγκρίνει την τιμή της μεταβλητής με το αλφαριθμητικό "happy". Εάν ταιριάζουν η έκφραση αποτιμάται σε αληθή και εμφανίζει η έξοδος:

```
Hooray, I'm in a good mood!
```

Φυσικά δεν υπάρχει ένας μόνο δρόμος στις αποφάσεις. Για αυτόν τον λόγο έχει προβλεφτεί και ένα `else` για να καλύψει αυτήν την περίπτωση, όπως δείχνει το επόμενο παράδειγμα:

```
<?php
$mood = "sad";
if ($mood == "happy") {
```

```

echo "Hooray, I'm in a good mood!";
} else {
echo "I'm in a $mood mood.";
}
?>

```

Εφόσον το αλφαριθμητικό “sad” δεν είναι ίδιο με το “happy” η έκφραση ελέγχου της εντολής στην γραμμή 3 αποτιμάται σε λάθος. Αυτό σημαίνει ότι το πρώτο τμήμα κώδικα δεν θα εκτελεστεί. Αντίθετα θα εκτελεστεί το τμήμα κώδικα με τά από την else και θα εμφανιστεί το μήνυμα:

```
I'm in a sad mood.
```

### 2.6.2.β Εντολή Switch

Εκτός του απλού επιλογέα υπάρχει και ο πολλαπλός επιλογέας, στον οποίο δεν εξετάζουμε μόνο μία περίπτωση για να πάρουμε μια απόφαση για την επόμενη εντολή, αλλά περισσότερες. Η δομή του πολλαπλού επιλογέα έχει ως εξής:

```

switch (variable)
{
case value1:
group1;
break;
case value2:
group2;
break;
.....
default:
group-default;
break;
}

```

Αν η τιμή της μεταβλητής είναι ίση με value1, τότε εκτελούνται οι εντολές group1. Διαφορετικά ελέγχουμε αν είναι ίση με value2. Αν είναι έτσι εκτελούνται οι δεύτερες εντολές. Που ακολουθούν πρόταση default. Εδώ θα πρέπει να επισημάνουμε τα ακόλουθα σε σχέση με την switch.

1. Οι εντολές break δεν είναι απαραίτητες για την συντακτική εγκυρότητα της. Αν δεν υπάρχουν όμως τότε θα εκτελεστεί και το επόμενο μπλοκ εντολών μετά το πρώτο ταίριασμα της μεταβλητής.

2. Η πρόταση default δεν είναι επίσης απαραίτητη. Ωστόσο συνιστάται η χρήση της προκειμένου να εξασφαλίσουμε πως τουλάχιστον μια ομάδα εντολών θα εκτελεστεί αλλά και πως έχουμε καλύψει όλες τις πιθανές περιπτώσεις για την τιμή της μεταβλητής.

```
<?php
$mood = "sad";
switch ($mood) {
case "happy":
echo "Hooray, I'm in a good mood!";
break;
case "sad":
echo "Awww. Don't be down!";
break;
default:
echo "Neither happy nor sad but $mood.";
break;
}
?>
```

Η πρώτη case ελέγχει την τιμή happy έναντι της τιμής της \$mood για ισότητα. Επειδή οι δύο τιμές δεν είναι ίσες, η εκτέλεση του script προχωρά στην δεύτερη case. Επειδή το αλφαριθμητικό sad είναι ισοδύναμο με την τιμή της \$mood, αυτό το τμήμα κώδικα εκτελείται. Η εντολή break στην τερματίζει την διαδικασία. Οι επόμενες γραμμές παρέχουν την προκαθορισμένη ενέργεια, για τις περιπτώσεις, στις οποίες καμία από τις συνθήκες δεν αποτιμάται ως αληθής.

### 2.6.2.γ Εντολή While

Ο απλούστερος τρόπος για να γράψουμε μία επανάληψη είναι με τη χρήση της εντολής while. Η εντολή αυτή δείχνει παρόμοια με την δομή μιας απλής εντολής if αλλά έχει την δυνατότητα δημιουργίας βρόχου:

```
while (condition)
{
group;
}
```

Δηλαδή όσο ισχύει η συνθήκη εκτελούνται η εντολές μέσα στον βρόχο.

```
<?php
$countner = 1;
```

```
while ($counter <= 12) {  
  echo $counter." times 2 is " .($counter * 2)."<br/>";  
  $counter++;  
}  
?>
```

Σ' αυτό το παράδειγμα αρχικοποιούμε μία μεταβλητή την \$counter. Η εντολή while ελέγχει την μεταβλητή και εφόσον ο ακέραιος που περιέχει είναι μικρότερος ή ίσος με 12, ο βρόχος συνεχίζει να επαναλαμβάνεται. Μέσα στο τμήμα κώδικα της while, η τιμή που περιέχει η μεταβλητή πολλαπλασιάζεται επί δύο και το αποτέλεσμα εκτυπώνεται. Και το αποτέλεσμα θα είναι:

```
1 times 2 is 2  
2 times 2 is 4  
3 times 2 is 6  
4 times 2 is 8  
5 times 2 is 10  
6 times 2 is 12  
7 times 2 is 14  
8 times 2 is 16  
9 times 2 is 18  
10 times 2 is 20  
11 times 2 is 22  
12 times 2 is 24
```

#### 2.6.2.δ Η Εντολή do ... while

Ο δεύτερος βρόχος που θα εξετάσουμε είναι ο βρόχος do ... while. Η σύνταξή του έχει ως ακολούθως:

```
do  
{  
  group;  
}while(condition);
```

Εδώ βλέπουμε πως το σύνολο εντολών θα εκτελεστεί και ύστερα θα ελέγξουμε αν η συνθήκη είναι αληθής ή ψευδής. Η διαφορά με τον προηγούμενο βρόχο είναι ότι με το do ... while οι εντολές θα εκτελεστούν τουλάχιστον μία φορά, όπως άλλωστε φαίνεται και στο επόμενο παράδειγμα:

```
<?php
```

```
$num = 1;
do {
echo "The number is: ".$num."<br/>";
$num++;
} while (($num > 200) && ($num < 400));
?>
```

Η εντολή `do...while` ελέγχει αν η μεταβλή `$num` περιέχει μια τιμή μεγαλύτερη από 200 και μικρότερη από 400. Στην γραμμή 2 αρχικοποιούμε την μεταβλητή σε τιμή 1, συνεπώς η έκφραση ελέγχουν επιστρέφει λάθος. Ωστόσο, το τμήμα του κώδικα εκτελείται πριν από την αποτίμηση της έκφρασης. Θα παραχθεί η έξοδος:

```
The number is: 1
```

### 2.6.2.ε Εντολή for

Το τελευταίο είδος επαναλήψεως είναι αυτό με την χρήση της εντολής `for`.

```
for (int; check; update)
{
group;
}
```

Ένα απλό παράδειγμα ακολουθεί:

```
<?php
for ($counter=1; $counter<=12; $counter++) {
echo $counter." times 2 is " . ($counter * 2). "<br/>";
}
?>
```

Θα παράγει την ακόλουθη έξοδο:

```
1 times 2 is 2
2 times 2 is 4
3 times 2 is 6
4 times 2 is 8
5 times 2 is 10
6 times 2 is 12
7 times 2 is 14
8 times 2 is 16
9 times 2 is 18
10 times 2 is 20
```



11 times 2 is 22

12 times 2 is 24

### 2.6.3 Χρήση & Διαχείριση Φορμών

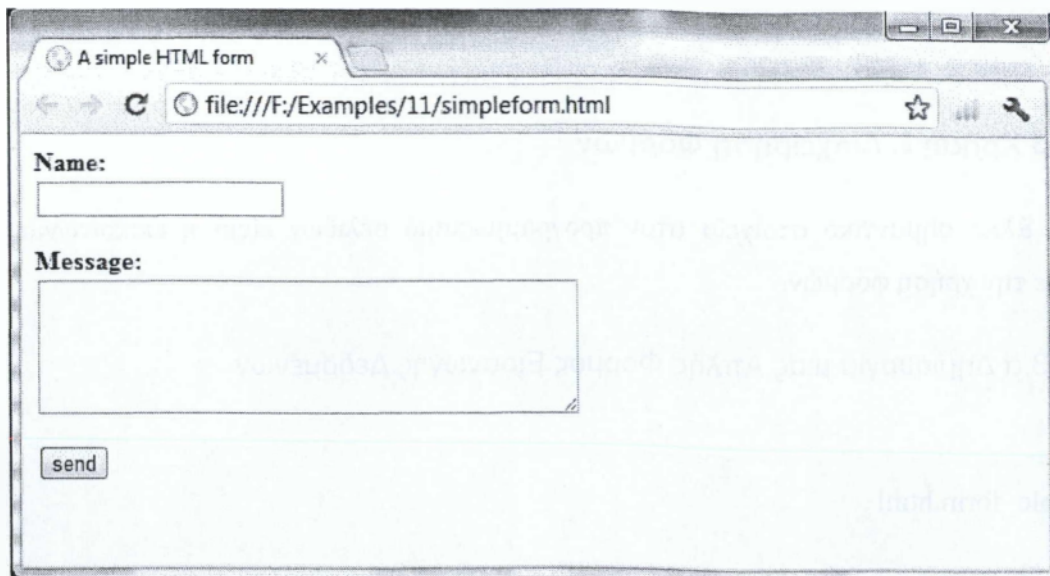
Ένα άλλο σημαντικό στοιχείο στον προγραμματισμό σελιδών είναι η επικοινωνία μεταξύ σελιδών με την χρήση φορμών.

#### 2.6.3.α Δημιουργία μιας Απλής Φόρμας Εισαγωγής Δεδομένων

##### simple\_form.html

```
<html>
<head>
<title>A simple HTML form</title>
</head>
<body>
<form action="send_simpleform.php" method="POST">
<p><strong>Name:</strong><br/>
<input type="text" name="user"></p>
<p><strong>Message:</strong><br/>
<textarea name="message" rows="5" cols="40"/></textarea></p>
<p><input type="submit" value="send"/></p>
</form>
</body>
</html>
```

Ο παραπάνω κώδικας ορίζει μία φόρμα η οποία περιλαμβάνει ένα πεδίο εισαγωγής κειμένου με όνομα users, ένα μεγαλύτερο πλαίσιο εισαγωγής κειμένου με το όνομα message και ένα κουμπί για την υποβολή. Το όρισμα action δείχνει σε ένα αρχείο με όνομα send\_simpleform.php το οποίο επεξεργάζεται τα δεδομένα που εισάγει ο χρήστης στη φόρμα.



#### send\_simpleform.php

```
<?php
echo "<p>Welcome <b>".$_POST["user"]."</b>!"</p>";
echo "<p>Your message is:<br/><b>".$_POST["message"]."</b></p>";
?>
```

Το script καλείται όταν ο χρήστης ολοκληρώσει την συμπλήρωση της φόρμας.

#### 2.6.3.β Αποστολή Ηλεκτρονικού Μηνύματος κατά την Υποβολή της φόρμας

Δημιουργούμε τον κώδικα που απαιτείται για την φόρμα. Η ενέργεια action που ορίζεται γι' αυτήν την φόρμα είναι sendmail.php

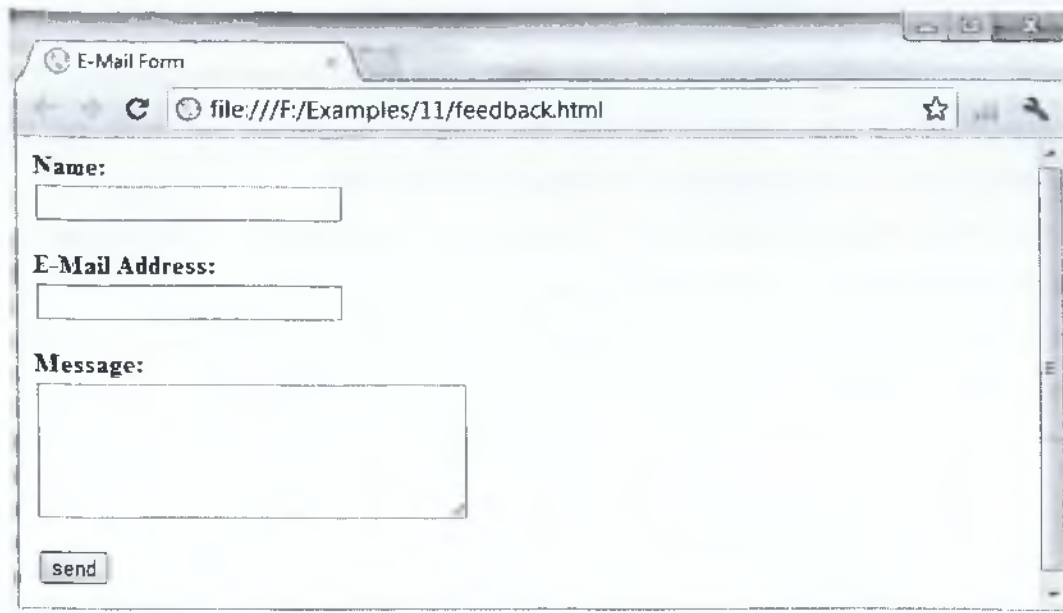
#### form.html

```
<html>
<head>
<title>E-Mail Form</title>
</head>
<body>
<form action="sendmail.php" method="POST">
<p><strong>Name:</strong><br/> <input type="text" size="25"
name="name"/></p>
<p><strong>E-Mail Address:</strong><br/> <input type="text" size="25"
name="email"/></p>
<p><strong>Message:</strong><br/>
<textarea name="message" cols="30" rows="5"></textarea></p>
<p><input type="submit" value="send"/></p>
```

```

</form>
</body>
</html>

```



#### sendmail.php

```

<html>
<head>
<title>Sending the Simple Feedback Form - HTML Version</title>
</head>
<body>
<?php
echo "<p>Thank you, <b>".$_POST["name"]."</b>, for your message!</p>";
echo "<p>Your e-mail address is: <b>".$_POST["email"]."</b>.</p>";
echo "<p>Your message was:<br />";
echo $_POST["message"]."</p>";
//start building the mail string
$msg = "<p><strong>Name:</strong> ".$_POST["name"]."</p>";
$msg .= "<p><strong>E-Mail:</strong> ".$_POST["email"]."</p>";
$msg .= "<p><strong>Message:</strong> ".$_POST["message"]."</p>";
//set up the mail
$recipient = "you@yourdomain.com";
$subject = "Form Submission Results";
$mailheaders = "MIME-Version: 1.0\r\n";
$mailheaders .= "Content-type: text/html; charset=ISO-8859-1\r\n";
$mailheaders .= "From: My Web Site <defaultaddress@yourdomain.com> \n";
$mailheaders .= "Reply-To: ".$_POST["email"];
//send the mail

```

```
mail($recipient, $subject, $msg, $mailheaders);
?>
</body>
</html>
```

Το αλφαριθμητικό \$msg περιέχει τις τιμές που εισήγαγε ο χρήστης στα πεδία της φόρμας. Αυτό το αλφαριθμητικό θα σταλεί μέσω ηλεκτρονικού ταχυδρομείου.

Οι επόμενες γραμμές περιλαμβάνουν ενσωματωμένα στον κώδικα τα στοιχεία για την διεύθυνση του παραλήπτη και το θέμα του μηνύματος.

Η συνάρτηση mail() δέχεται τέσσερις παραμέτρους: τον παραλήπτη, το θέμα, το μήνυμα, και οποιεσδήποτε επιπλέον κεφαλίδες (headers).

### 2.6.3.γ Μεταφορά Αρχείων

Όλοι οι δημοφιλείς web browser παρέχουν την δυνατότητα μεταφοράς αρχείων και το ίδιο φυσικά ισχύει για την PHP που διαθέτει λειτουργίες μεταφοράς αρχείων μέσω φορμών.

Οι πληροφορίες για το μεταφερόμενο αρχείο γίνονται διαθέσιμες μέσω της προκαθορισμένης γενικής μεταβλητής \$\_FILES, η οποία είναι μία διάταξη του πεδίου (ή των πεδίων) της φόρμας που χρησιμοποιείται για την μεταφορά. Το fileupload είναι το όνομα που χρησιμοποιούμε για το πεδίο της φόρμας, στο οποίο καθορίζεται το αρχείο.

Στοιχείο	Περιέχει
\$_FILES["fileupload"]["name"]	Όνομα του αρχείου πριν από την μεταφορά
\$_FILES["fileupload"]["temp_name"]	Διαδρομή προσωρινού αρχείου
\$_FILES["fileupload"]["size"]	Μέγεθος του αρχείου
\$_FILES["fileupload"]["type"]	Ο τύπος MIME του αρχείου

Ακολουθεί μια φόρμα HTML για τον χειρισμό της διαδικασίας αποστολής. Οι HTML φόρμες που περιλαμβάνουν πεδία για αποστολή αρχείων πρέπει επίσης να περιλαμβάνουν ένα όρισμα ENCTYPE:

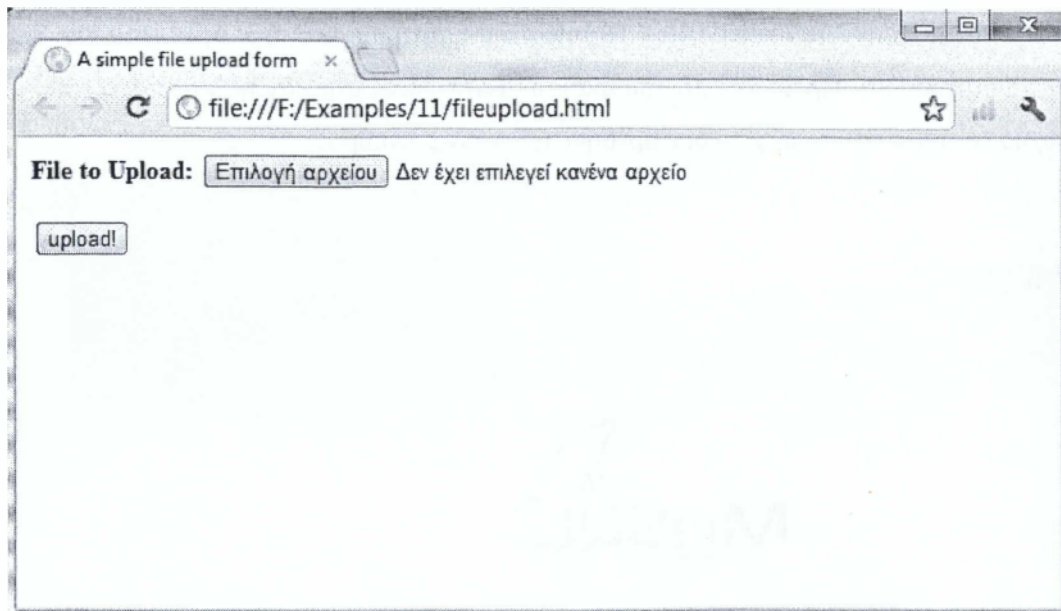
```
ENCTYPE = "multipart/form-data"
```

**fileupload.html**

```

<html>
<head>
<title>A simple file upload form</title>
</head>
<body>
<form action="do_upload.php" enctype="multipart/form-data" method="POST">
<input type="hidden" name="MAX_FILE_SIZE" value="51200"/>
<p><strong>File to Upload:</strong> <input type="file"
name="fileupload"/></p>
<p><input type="submit" value="upload!"></p>
</form>
</body>
</html>

```

**Δημιουργία του script Αποστολής Αρχείων****do\_upload.php**

```

<?php
$file_dir = "/path/to/upload/directory/";
foreach($_FILES as $file_name => $file_array) {
echo "path: ".$file_array["tmp_name"]."<br/>\n";
echo "name: ".$file_array["name"]."<br/>\n";
echo "type: ".$file_array["type"]."<br/>\n";
echo "size: ".$file_array["size"]."<br/>\n";
}

```

```
if (is_uploaded_file($file_array["tmp_name"])) {  
    move_uploaded_file($file_array["tmp_name"], "$file_dir/".$file_array["name"]);  
    or die ("Couldn't copy");  
    echo "file was moved!<br/>";  
}  
}  
?>
```

Δημιουργούμε την μεταβλητή `$file_dir` για την αποθήκευση της διαδρομής. Χρησιμοποιούμε έναν βρόχο, έτσι ώστε να μπορεί να επεκταθεί σε ευρύτερη κλίμακα εάν χρειαστεί να χειρίζεται πολλαπλές αποστολές αρχείων στην ίδια σελίδα. Το όνομα του αρχείου αποθηκεύεται στην μεταβλητή `$file_name` και τις πληροφορίες για το αρχείο στην μεταβλητή `$file_array`.

Χρησιμοποιώντας την συνάρτηση `is_upload_file()` ελέγχουμε ότι το αρχείο υπάρχει. Ύστερα το αρχείο νατιγράφεται από την προσωρινή του θέση σε ένα νέο κατάλογο. Χρησιμοποιούμε μία άλλη συνάρτηση την `move_uploaded_file()`. Η συνάρτηση αυτή απαιτεί σαν ορίσματα την διαδρομή προς το πηγαίο αρχείο και την διαδρομή προορισμού. Επιστρέφει `true` εάν η μετακίνηση του αρχείου είναι επιτυχής και `false` εάν το αρχείο δεν βρέθηκε ή δεν είναι έγκυρο.

## 2.7 MySQL



### 2.7.1 Σπουδαιότητα της Καλής Σχεδίασης μιας Βάσης Δεδομένων

Η καλή σχεδίαση μιας βάσης δεδομένων είναι ζωτικής σημασίας για την ανάπτυξη μιας εφαρμογής με υψηλή απόδοση. Εάν μία βάση δεδομένων δεν έχει βελτιστοποιημένες σχέσεις μεταξύ των πινάκων – κάτι το οποίο αποκαλείται κανονικοποίηση (normalization) – δεν θα μπει να λειτουργεί με τη μέγιστη δυνατή αποτελεσματικότητα.

Εκτός από την απόδοση υπάρχει και το θέμα της συντήρησης. Η βάση δεδομένων θα πρέπει να είναι εύκολη στην συντήρηση. Αυτό σημαίνει ότι θα πρέπει να αποθηκεύει όσο το δυνατόν λιγότερα επαναλαμβανόμενα δεδομένα. Εάν υπάρχουν πολλά επαναλαμβανόμενα δεδομένα στην βάση δεδομένων και γίνει μία αλλαγή σε κάποιο από αυτά, θα πρέπει να γίνει η ίδια αλλαγή σε όλες τις

υπόλοιπες εμφανίσεις του. για να εξαλειφθεί η επανάληψη και να βελτιωθούν οι δυνατότητες συντήρησης πρέπει να δημιουργηθεί ένας πίνακας τιμών και να χρησιμοποιηθεί ένα κλειδί για να αναφορά στην τιμή. Έτσι εάν αλλάει το όνομα της τιμής, η αλλαγή αυτή γίνεται μόνο μία φορά – στον κύριο πίνακα. Οι αναφορές παραμένουν ίδιες σε όλους τους άλλους πίνακες.

Το μεγαλύτερο πρόβλημα κατά την σχεδίαση μιας εφαρμογής είναι η έλλειψη αρχικής μελέτης. Η διαδικασία σχεδίασης πρέπει να περιλαμβάνει μία σχολαστική ανάλυση της βάσης δεδομένων – τι πρέπει να αποθηκεύει, πως σχετίζονται τα δεδομένα μεταξύ τους, και το πιο σημαντικό, εάν έχει δυνατότητα επέκτασης σε ευρύτερη κλίμακα.

Τα γενικά βήματα που πρέπει να ακολουθούνται κατά την διαδικασία σχεδίασης είναι:

- ορισμός του στόχου
- σχεδίαση των δομών δεδομένων (πίνακες, πεδία)
- εύρεση των σχέσεων
- ορισμός και υλοποίηση των επιχειρησιακών κανόνων
- δημιουργία της εφαρμογής

## 2.7.2 Τύποι Δεδομένων

Ο σωστός ορισμός των πεδίων ενός πίνακα είναι σημαντικός για την συνολική βελτιστοποίηση μιας βάση δεδομένων. Θα πρέπει να χρησιμοποιείται ο τύπος δεδομένων και το μέγεθος που χρειάζεται κάθε πεδίο.

Η MySQL χρησιμοποιεί πολλούς διαφορετικούς τύπους δεδομένα, οι οποίοι ταξινομούνται σε τρεις κατηγορίες: αριθμητικοί, ημερομηνία/ώρας και αλφαριθμητικών. Ο ορισμός του σωστού τύπου δεδομένων είναι πιο σημαντικός από οποιαδήποτε άλλη άποψη της διαδικασίας δημιουργίας ενός πίνακα.

### 2.7.2.α Αριθμητικοί Τύποι

- `Int`. ένας ακέραιος κανονικού μεγέθους
- `tinyint`. Ένας πολύ μικρός ακέραιος
- `smallint`. Ένας μικρός ακέραιος
- `mediumint`. Ένας ακέραιος μεσαίου μεγέθους
- `bigint`. Ένα μεγάλος ακέραιος
- `float(m, d)`. ένας αριθμός κινητής υποδιαστολής

- `double(m, d)`. ένας αριθμός κινητής υποδιαστολής διπλής ακρίβειας
- `decimal(m, d)`. ένας μη – συμπιεσμένος αριθμός κινητής υποδιαστολής.

#### 2.4.2.β Τύποι Δεδομένων για τιμές Ημερομηνίας/Ωρας

- `Date`. Μία τιμή ημερομηνίας με την μορφή `EEEE-MM-HH`
- `datetime`. Ένας συνδυασμός τιμών ημερομηνίας και ώρας με την μορφή `EEEE-MM-HH`

#### ΩΩ:ΛΛ:ΔΔ

- `timestamp`. Μία χρονική ένδειξη ανάμεσα στα μεσάνυχτα της 1ης ιανουαρίου 1970 και στο έτος 2037.
- `time`. Αποθηκεύει την ώρα σε μορφή `ΩΩ:ΛΛ:ΔΔ`
- `year(m)`. αποθηκεύει ένα έτος σε διψήφια ή τετραψήφια μορφή.

#### 2.7.2.γ Τύποι Αλφαριθμητικών

- `Char`. Ένα αλφαριθμητικό σταθερού μεγέθους
- `varchar`. Ένα αλφαριθμητικό μεταβλητού μεγέθους
- `text`. Ένα πεδίο με μέγεθος 65.535 χαρακτήρες
- `enum`. Μία απαρίθμηση, δηλαδή, μία λίστα.

#### 2.7.3 Δημιουργία Πινάκων

Η εντολή για τη δημιουργία ενός πίνακα απαιτεί τα ακόλουθα στοιχεία:

- Το όνομα του πίνακα
- Τα ονόματα των πεδίων
- Ένα ορισμό για κάθε πεδίο

Η γενική σύνταξη για την εντολή δημιουργίας πίνακα είναι:

```
CREATE TABLE table_name (col_name, col_type);
```

Το ακόλουθο παράδειγμα δημιουργεί έναν πίνακα:

```
mysql> CREATE TABLE example_autoincrement (
    id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
    data VARCHAR(100)
```



```
);
Query OK, 0 rows affected (0.01 sec)
```

### 2.7.4 Χρήση εντολών INSERT, SELECT

Αφού δημιουργηθούν οι πίνακες θα εισάγουμε δεδομένα με την εντολή INSERT τη SQL. Η βασική σύνταξη της εντολής είναι:

```
INSERT INTO table_name (column1, column2, column3,...)
VALUES (value1, value2, value3,...)
```

Στη λίστα των τιμών μέσα στις παρενθέσεις τα αλφαριθμητικά θα πρέπει να περικλείονται σε εισαγωγικά.

Η SELECT είναι η εντολή που χρησιμοποιείται για την ανάκτηση εγγραφών από έναν πίνακα. Η σύνταξη αυτής της εντολής είναι:

```
SELECT column_name(s)
FROM table_name
```



## 2.8 CSS Γλώσσα

Η CSS (Cascading Style Sheets – Διαδοχικά Φύλλα Στυλ) ή (αλληλουχία φύλλων στυλ) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστοτόπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στυλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.

Μορφή μιας εντολής CSS: Επιλογέας {ιδιότητα:Τιμή;}

Παράδειγμα: h1 {color:blue;}

Επιλογέας (selector) = στοιχείο του εγγράφου (π.χ. h1, p,...)

Ιδιότητα (attribute) = π.χ. color

Τιμή = π.χ. red

Η χρήση CSS γίνεται με 3 τρόπους:

- Κατευθείαν μέσα στις ετικέτες με την εντολή "style=..."
- Στο Head του εγγράφου με κανόνες style (Internal Style Sheet):

```
<style type="text/css"> body {background-color: blue; }
</style>
```

Ο παρακάτω ορισμός θα εμφανίζει κεντραρισμένες και με κόκκινα γράμματα όλες τις παραγράφους (<p>):

```
<title>Μορφοποίηση παραγράφου</title>
<style type="text/css">
p {
font-family: "Courier New";
color: #FF3333;
text-align: center;
} </style>
```

Στο παρακάτω παράδειγμα ορίζουμε σαν φόντο μια εικόνα η οποία θα παραμένει σταθερή στο κέντρο της οθόνης:

```
<style type="text/css">
body {
background: #00ff00 url('only_logo.jpg') no-repeat fixed center;
}
</style>
```

Έχουμε επίσης τη δυνατότητα να ορίσουμε ίδιο στυλ σε περισσότερες ετικέτες:

```
p,td,th {
font-family: "Courier New";
color: #FF3333;
text-align: center;
}
```

Η μπορούμε να ορίσουμε διαφορετικά στυλ για μια ετικέτα, χρησιμοποιώντας το όνομά της και μια τιμή για την ιδιότητα class της ετικέτας:

```
<style type="text/css">
em.idiotites {
font-size: 12px;
font-family: arial, helvetica, sans-serif;
color: #228B22;
font-weight: bold;
text-decoration: none;
}

em.times {
```

```
font-size: 12px;
font-family: Arial, Helvetica, sans-serif;
color: #FFA07A;
font-weight: bold;
text-decoration: none;
)
</style>
```

Η χρήση θα γίνει με `<em class="idiotites"> ... </em>`

- Με εξωτερικό αρχείο CSS, που το ορίζουμε στο Head:

```
<link rel="stylesheet" type="text/css" href="mystyle.css">
```

```
ή <link href="/mystyle.css" rel="stylesheet" type="text/css" media="all">
```

Επιλογές box (box είναι ένα μπλοκ με στοιχεία...)

**margin** = η απόσταση του box από άλλα γειτονικά...

**padding** = >> των περιεχομένων του box απ' το περίγραμμά του

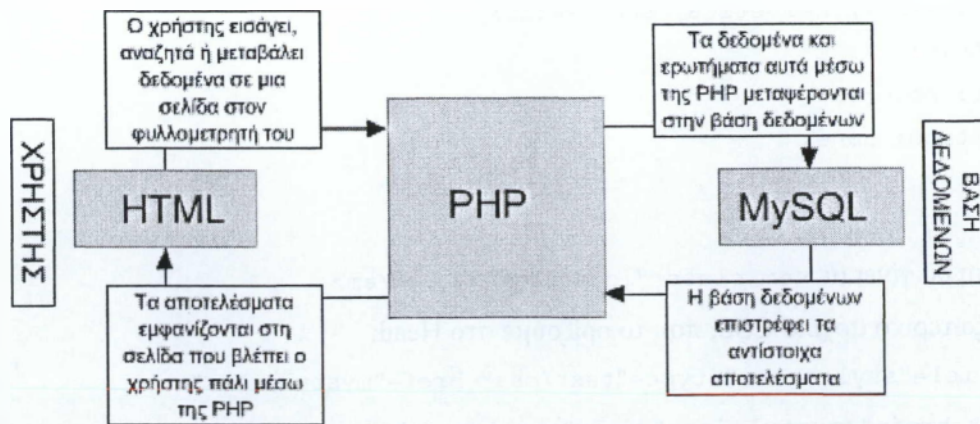
**Width, Heigh** = Οι διαστάσεις του box

**float** = επιτρέπει να χειριζόμαστε το μπλοκ σαν εικόνα και το τοποθετεί δεξιά ή αριστερά...

## 2.9 Ενοποίηση των γλωσσών PHP & MySQL

Για την διαχείριση μιας βάσης δεδομένων η οποία είναι γραμμένη σε MySQL κάποιος θα πρέπει να γνωρίζει τις εντολές και τη σύνταξη της γλώσσας. Ένα από τα μεγάλα πλεονεκτήματα της PHP είναι ότι μέσω αυτής μπορούμε να επικοινωνήσουμε με βάσεις δεδομένων γραμμένες σε MySQL. Η PHP λειτουργεί σαν αγωγός, στέλνοντας εντολές στη MySQL και λαμβάνοντας αποτελέσματα από αυτή.

Αυτό, σε συνδυασμό με την συνεργασία μεταξύ PHP και HTML, έχει σαν τελικό αποτέλεσμα να μπορεί κάποιος συμπληρώνοντας μια απλή HTML φόρμα, να μπορεί να εισάγει δεδομένα σε μια βάση δεδομένων ή ακόμα και με το πάτημα ενός κουμπιού να αλλάζει το περιεχόμενό της. Με απλά λόγια, η HTML, η PHP και η MySQL μπορούν να συνεργαστούν ώστε ο χρήστης μέσω μιας ιστοσελίδας να μπορεί να διαχειριστεί βάσεις δεδομένων. Μια απλή απεικόνιση αυτής της διαδικασίας ακολουθεί στο παρακάτω σχήμα:



Επικοινωνία μιας σελίδας σε φυλλομετρητή με μια βάση δεδομένων μέσω της PHP.

Έτσι ο χρήστης δεν χρειάζεται να γνωρίζει MySQL για να μπορεί να διαχειριστεί μια βάση δεδομένων. Το μόνο που χρειάζεται είναι μια σελίδα HTML η οποία να περιέχει κώδικα PHP κατάλληλο ώστε να επικοινωνεί με τη MySQL. Οι σελίδες της παρούσας εφαρμογής λειτουργούν ακριβώς με αυτό τον τρόπο.

### 2.9.1 Σύνδεση στην MySQL με την PHP

#### Δημιουργία μια Σύνδεσης

Η βασική σύνταξη για μια σύνδεση στην MySQL είναι:

```
$conn = mysql_connect( "hostname" , "username" , "password" );
```

Επιλογή βάσης δεδομένων:

```
mysql_select_db ( "Dbname" , $conn );
```

Εκτέλεση ερωτήματος:

```
$sql = " [μια εντολή MySQL] ";
```

```
mysql_query ( $sql , $conn );
```

Στο παράδειγμα που θα δούμε ο κώδικας δημιουργεί μια νέα σύνδεση και μετά ελέγχει για να δει αν συνέβη κάποιο σφάλμα. Αν συνέβη σφάλμα, εκτυπώνεται ένα μήνυμα σφάλματος και χρησιμοποιείται η συνάρτηση `mysql_connect_error()` ώστε να εκτυπώσει το μήνυμα. Αν δεν υπάρχουν σφάλματα γίνεται η σύνδεση.

```
mysqlconnect.php
```

```
<?php
```

```
$link = mysql_connect('localhost', 'mysql_user', 'mysql_password');
if (!$link) {
die('Could not connect: ' . mysql_error());
}
echo 'Connected successfully';
mysql_close($link);
?>
```

## 2.9.2 Δουλεύοντας με δεδομένα της MySQL

Η εισαγωγή, η ενημέρωση, η διαγραφή και η ανάκτηση δεδομένων είναι εργασίες οι οποίες βασίζονται στην χρήση της συνάρτησης `mysql_query()` για την εκτέλεση ερωτημάτων SQL.

### 2.9.2.α Εισαγωγή Δεδομένων σε έναν Πίνακα μέσω PHP

Η ευκολότερη μέθοδος για εισαγωγή δεδομένων σε έναν πίνακα μέσω της PHP είναι η ενσωμάτωση των δεδομένων στην εντολή INSERT.

`mysqlinsert.php`

```
<?php
$mysql = mysql_connect("localhost", "joeuser", "somepass", "testDB");

if (mysql_connect_error()) {
printf("Connect failed: %s\n", mysql_connect_error());
exit();
} else {
$sql = "INSERT INTO testTable (testField) VALUES ('some value')";
$res = mysql_query($mysql, $sql);

if ($res === TRUE) {
echo "A record has been inserted.";
} else {
printf("Could not insert record: %s\n", mysql_error($mysql));
}

mysql_close($mysql);
?>
```

### 2.9.2.β Ανάκτηση Δεδομένων με την PHP

Για να μπορέσουμε να κάνουμε ανάκτηση δεδομένων από έναν πίνακα θα γράψουμε ένα script το οποίο θα εκτελεί ένα ερώτημα επιλογής (select).

**select.php**

```
<?php
$mysql = mysql_connect("localhost", "joeuser", "somepass", "testDB");

if (mysql_connect_error()) {
printf("Connect failed: %s\n", mysql_connect_error());
exit();
} else {
$sql = "SELECT * FROM testTable";
$res = mysql_query($mysql, $sql);

if ($res) {
$number_of_rows = mysql_num_rows($res);
printf("Result set has %d rows.\n", $number_of_rows);
} else {
printf("Could not retrieve records: %s\n", mysql_error($mysql));
}

mysql_free_result($res);
mysql_close($mysql);
}
?>
```

### 3 Κεφάλαιο: Δημιουργία ιστοσελίδας Φοιτητικών Υπηρεσιών

Σε αυτό το κεφάλαιο θα περιγραφεί με λεπτομέρεια η λειτουργία του ιστοτόπου της εφαρμογής μας. Θα γίνει ανάλυση τόσο του κώδικα της εφαρμογής όσο και της διεπαφής που προκύπτει από την εκτέλεση αυτού.

#### 3.1 Σχεδιασμός και Δημιουργία των Πινάκων της Βάσης Δεδομένων

Ο παρακάτω πίνακας παρουσιάζει τα ονόματα των πινάκων και των πεδίων που θα χρησιμοποιήσουμε για την υλοποίηση του ιστοτόπου.

Πίνακας	Όνομα Πεδίων
Users	username, password, email, a_m, f_name, l_name, semester, direction
per_residence	a_m, street, z.c, city, country, res_per_phone
temp_residence	a_m, street, z.c, city, country, cell_phone, temp_phone
Signup	a_m, inyear, signup_way, period

Με τις παρακάτω εντολές θα δημιουργήσουμε τους πίνακες που χρειαζόμαστε:

```
CREATE TABLE `users` (
  `username` varchar(45) NOT NULL,
  `password` varchar(45) NOT NULL,
  `email` varchar(50) NOT NULL,
  `A_M` varchar(45) NOT NULL,
  `f_name` varchar(45) NOT NULL,
  `l_name` varchar(45) NOT NULL,
  `semester` enum('A','B','Γ','Δ','Ε','ΣΤ','Ζ','Η','ΠΤΥΧΙΟ') NOT NULL,
  `direction` enum('','ΠΑΝΡΟΦΟΡΙΚΗ','ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ') NOT NULL,
  PRIMARY KEY (`A_M`),
  UNIQUE KEY `Index_2` (`username`)
) ENGINE=InnoDB DEFAULT CHARSET=greek;
```

Αρχικά δημιουργούμε τον πίνακα users με πρωτεύων κλειδί το A\_M και κάνοντας το username UNIQUE ώστε να είναι μοναδικό για κάθε χρήστη.

Έπειτα δημιουργούμε τον πίνακα με τα στοιχεία της μόνιμης κατοικίας των χρηστών.

```
CREATE TABLE `per_residence` (
  `A_M` varchar(45) NOT NULL,
  `street` varchar(45) NOT NULL,
  `Z_C` varchar(45) NOT NULL,
  `city` varchar(45) NOT NULL,
  `country` varchar(45) NOT NULL,
  `res_per_phone` varchar(45) NOT NULL,
  PRIMARY KEY (`A_M`),
  CONSTRAINT `FK_per_residence_1` FOREIGN KEY (`A_M`) REFERENCES `users`
(`A_M`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=greek;
```

Ακολουθεί ο πίνακας προσωρινής κατοικίας των χρηστών και ο πίνακας με τον τρόπο εγγραφής τους στο TEI.

```
CREATE TABLE `temp_residence` (
  `A_M` varchar(45) NOT NULL,
  `street` varchar(45) NOT NULL,
  `Z_C` varchar(45) NOT NULL,
  `city` varchar(45) NOT NULL,
  `country` varchar(45) NOT NULL,
  `cell_phone` varchar(45) NOT NULL,
  `temp_phone` varchar(45) NOT NULL,
  PRIMARY KEY (`A_M`),
  CONSTRAINT `FK_temp_residence_1` FOREIGN KEY (`A_M`) REFERENCES `users`
(`A_M`) ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=greek;
```

```
CREATE TABLE `signup` (
  `A_M` varchar(45) NOT NULL,
  `in_year` date NOT NULL,
  `signup_way` varchar(45) NOT NULL,
  `period` enum('ΧΕΙΜΕΡΙΝΟ','ΕΑΡΙΝΟ') NOT NULL,
```



```

`semester` enum('A','B','Γ','Δ','Ε','ΕΤ','2','H','ΠΤΥΧΙΟ') NOT NULL,
PRIMARY KEY (`A_M`),
CONSTRAINT `FK_signup_1` FOREIGN KEY (`A_M`) REFERENCES `users` (`A_M`) ON
UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=greek;

```

### 3.2 Δημιουργία Μενού

Ο ιστότοπος θα πρέπει να δίνει την δυνατότητα στους χρήστες να εκτελούν αρκετές ενέργειες. Είναι συνεπώς λογικό να δημιουργήσουμε ένα μενού με τις σχετικές συνδέσεις. Η παρακάτω λίστα δημιουργεί το μενού με το οποίο θα πλοηγείται ο χρήστης.

```

<ul class="menu">
  <li class="top" <a href="index.html"
class="top_link"><span>Αρχική</span></a>
  <ul class="sub">
    <li><a href="anakoinoseis.html">Ανακοινώσεις</a></li>
    <li><a href="entipa.html">Ενημερωτικά Έντυπα</a></li>
    <li><a href="programma_spydon.html">Πρόγραμμα Σπουδών</a></li></ul></li>
  <li class="top" <a href="profil.php"
class="top_link"><span>Προφίλ</span></a></li>
  <li class="top" <a href="pistopoihtika.html"
class="top_link"><span>Πιστοποιητικά</span></a>
  <ul class="sub">
    <li><a href="aitisi_fitisis.html">Αίτηση Φοίτησης</a></li>
    <li><a href="aitisi_anaolis_stratefsis.html">Αναβολή Στράτευσης</a></li>
    <li><a href="diakopi_anaolis_stratefsis.html">Διακοπή Αναβολής
Στράτευσης</a></li>
    <li><a href="aitisi_fititikou_epidomatos.html">Αίτηση Φοιτητικού
Επιδόματος</a></li></ul></li>
  <li class="top" <a href="veveoseis.html"
class="top_link"><span>Βεβαιώσεις</span></a>
  <ul class="sub">
    <li><a href="veveosi_spydon.html">Βεβαίωση Σπουδών</a></li>
    <li><a href="veveosi_eggrafis.html">Βεβαίωση Εγγραφής</a></li>
    <li><a href="veveosi_apofitisis.html">Βεβαίωση Αποφοίτησης</a></li>
    <li><a href="veveosi_prosagogis_ejetazomenou.html">Βεβαίωση
Προσαγωγής</a></li></ul></li>

```

```

<li class="top"><a href="praktiki_askisi.html"
class="top_link"><span>Πρακτική Άσκηση</span></a>
  <ul class="sub">
    <li><a href="aitisi_praktikis_askisis.html">Αίτηση Πρακτικής
Άσκησης</a></li></ul></li>
  <li class="top"><a href="ptixiaki_ergasia.html"
class="top_link"><span>Πτυχιακή Εργασία</span></a>
  <ul class="sub">
    <li><a href="katoxirosi_thematos.html">Αίτηση Κατοχύρωσης Θέματος</a></li>
    <li><a href="katathesi_ptixiakis.html">Αίτηση Κατάθεσης
Πτυχιακής</a></li></ul></li>
  <li class="top"><a href="alles_aitiseis.html" class="top_link"><span>Άλλες
Αιτήσεις</span></a>
  <ul class="sub">
    <li><a href="aitisi_sitisis.html">Αίτηση Σίτισης</a></li>
    <li><a href="aitisi_pasou.html">Αίτηση Πάσου</a></li>
    <li><a href="aitisi_metavolis_stixion.html">Αίτηση Μεταβολής
Στοιχείων</a></li>
    <li><a href="aitisi_metafrasis_ptixiou.html">Αίτηση Μετάφρασης
Πτυχίου</a></li>
    <li><a href="aitisi_stegasis.html">Αίτηση Στέγασης</a></li>
    <li><a href="aitisi_analitikis_bathmologias.html">Αίτηση Αναλυτικής
Βαθμολογίας</a></li>
    <li><a href="aitisi_ipotrofias.html">Αίτηση Υποτροφίας</a></li>
    <li><a href="aitisi_erasmus.html">Erasmus</a></li>
    <li><a href="programa_leonardo.html">Πρόγραμμα Leonardo Da
Vinci</a></li></ul></li></ul></div>

```

Αρχή Προφίλ Πιστοποιητικό Βεβαιώσεις Πρακτική Άσκηση Πτυχιακή Εργασία Άλλες Αιτήσεις

Και έτσι δημιουργήθηκε το κύριο μενού. Υπάρχουν και άλλα υπομενού που θα εξετάσουμε στον πιο κάτω κώδικα.

Υπομενού Βεβαιώσεις:

```

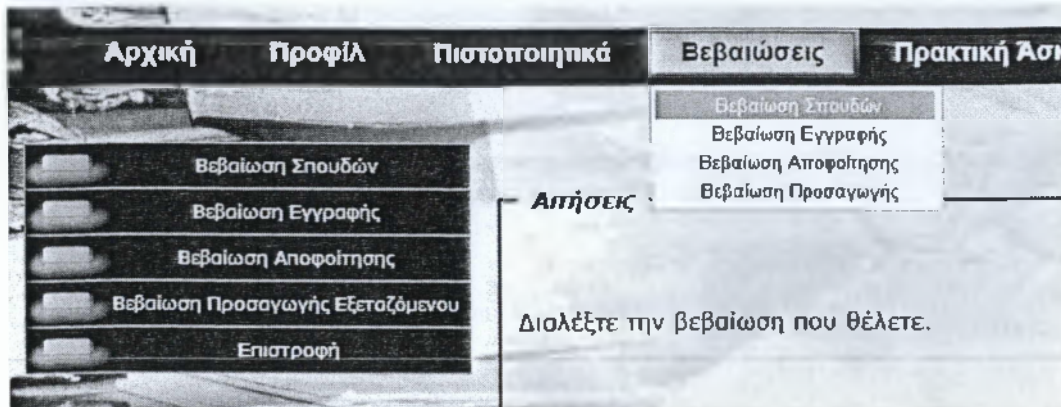
<div id="leftcol">
<div id="menu2">
<ul>
<li><a href="veveosi_spoynon.html">Βεβαίωση Σπουδών</a></li>
<li><a href="veveosi_eggrafis.html">Βεβαίωση Εγγραφής</a></li>
<li><a href="veveosi_apofitisis.html">Βεβαίωση Αποφοίτησης</a></li>

```

```

<li><a href="veveosi_prosagogis_ejetazomenou.html">Βεβαίωση Προσαγωγής
Εξεταζόμενου</a></li>
<li><a href="aitiseis.html">Επιστροφή</a></li></ul></div>
</div>

```

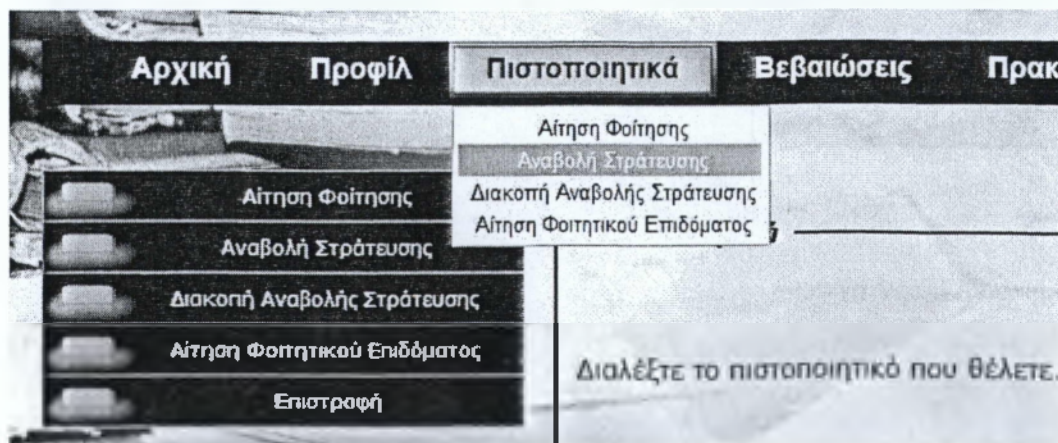


**Υπομενού Πιστοποιητικά:**

```

<div id="leftcol">
<div id="menu2">
<ul>
<li><a href="aitisi_fitisis.html">Αίτηση Φοίτησης</a></li>
<li><a href="aitisi_anavolis_stratefsisis.html">Αναβολή Στράτευσης</a></li>
<li><a href="diakopi_anavolis_stratefsisis.html">Διακοπή Αναβολής
Στράτευσης</a></li>
<li><a href="aitisi_fititikou_epidomatos.html">Αίτηση Φοιτητικού
Επιδόματος</a></li>
</ul><a href="aitiseis.html">Επιστροφή</a></li></ul></div>
</div>

```

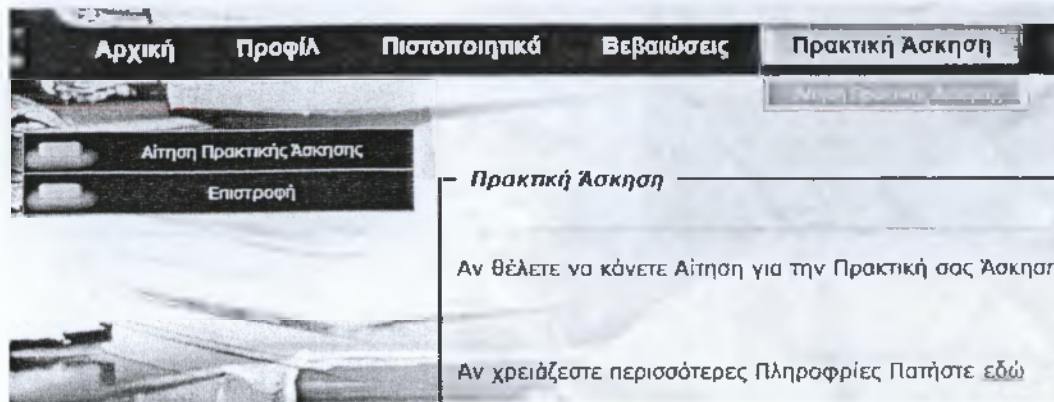


**Υπομενού Αιτήσεις Πρακτικής Άσκησης:**

```

<div id="leftcol">
<div id="menu2">
<ul>
<li><a href="aitisi_praktikis_askisis.html">Αίτηση Πρακτικής
Άσκησης</a></li>
<li><a href="aitiseis.html">Επιστροφή</a></li></ul></div>
</div>

```

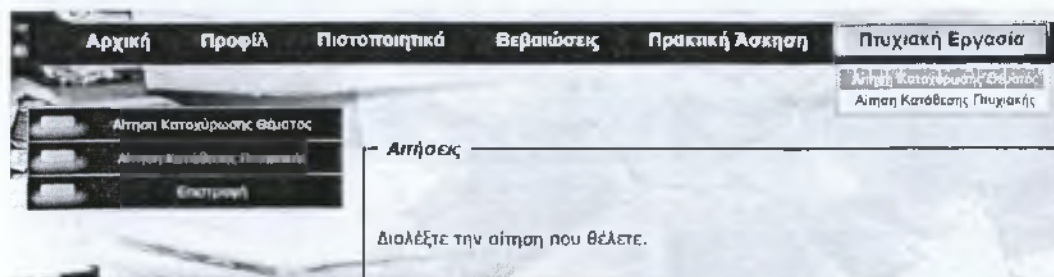


#### Υπομενού Αιτήσεις Πτυχιακής Εργασίας:

```

<div id="leftcol">
<div id="menu2">
<ul>
<li><a href="katoxirosi_thematos.html">Αίτηση Κατοχύρωσης Θέματος</a></li>
<li><a href="katathesi_ptixiakis.html">Αίτηση Κατάθεσης Πτυχιακής</a></li>
<li><a href="aitiseis.html">Επιστροφή</a></li></ul></div>
</div>

```



#### Υπομενού Άλλες Αιτήσεις:

```

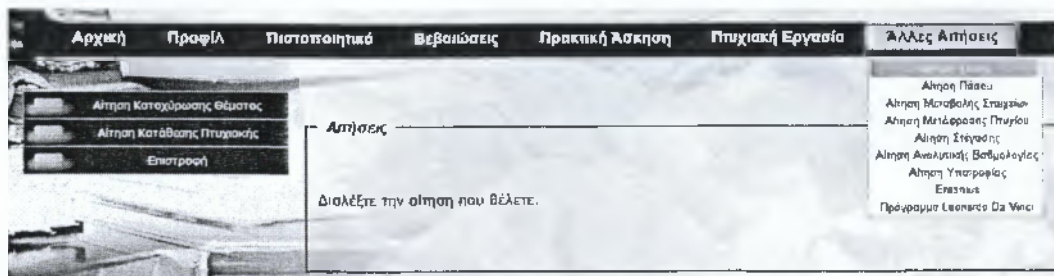
<div id="leftcol">
<div id="menu2">
<ul>
<li><a href="aitisi_sitisis.html">Αίτηση Σίτισης</a></li>
<li><a href="aitisi_pasou.html">Αίτηση Πάσου</a></li>

```

```

<li><a href="aitisi_metavolis_stixion.html">Αίτηση Μεταβολής
Στοιχείων</a></li>
<li><a href="aitisi_metafrasis_ptixiou.html">Αίτηση Μετάφρασης
Πτυχίου</a></li>
<li><a href="aitisi_stegasis.html">Αίτηση Στέγασης (φοιτητική
εστία)</a></li>
<li><a href="aitisi_analitikis_bathmologias.html">Αίτηση Αναλυτικής
Βαθμολογίας</a></li>
<li><a href="aitisi_ipotrofias.html">Αίτηση Υποτροφίας</a></li>
<li><a href="aitisi_erasmus.html">Προγράμματα Ανταλλαγών (erasmus)</a></li>
<li><a href="programa_leonardo.html">Πρόγραμμα Leonardo Da Vinci</a></li>
<li><a href="aitiseis.html">Επιστροφή</a></li></ul></div>
</div>

```



Το css μενού που χρησιμοποιήθηκε για το στυλ της σελίδας είναι:

```

body {
background-image: url("background/reading-book1.jpg");
margin: 0px;
padding: 0px;
text-align: center;
font: 100% Verdana, Arial, Helvetica, sans-serif;
}
// το ωφέλιμο πλάτος όλης της σελίδας
div#container {
margin: auto;
width: 1200px;
}

h1 {
font-size:20px;
}

h2 {
font-size:22px;
}

```

```
}

h3 {
font-size: 20px;
font-weight:bold;
margin: 10px 0 10px 0;
}

hr {
border: 1px dotted maroon;
margin-top: 20px;
margin-bottom: 20px;
padding: 0;
}

//αριστερό div για το κάθετο μενού
div#leftcol {
width: 320px;
padding-top: 40px;
height: 20px;
}

//δεξιό div για την εισαγωγή δεδομένων

div#rightcol {
width:880px;
float: right;
}

Το στυλ που χρησιμοποιήθηκε για το μενού:
/* δημιουργία μενού */

.bg {
background: url(menu_style/button4.gif);}
.menu {
padding:0 0 0 32px;
margin:0;
list-style:none;
height:40px;
background:black repeat-x;
position:relative;
font-family:arial, verdana, sans-serif; }
.menu li.top {
```

```
display:block;
float:left;
position:relative;}
.menu li a.top_link {
display:block;
float:left;
height:50px;
line-height:33px;
color:white;
text-decoration:none;
font-size:18px;
font-weight:bold;
padding:0 0 0 12px;
cursor:pointer;}
.menu li a.top_link span {
float:left;
font-weight:bold;
display:block;
padding:0 24px 0 12px;
height:50px;}
.menu li a.top_link span.down {
float:left;
display:block;
padding:0 24px 0 12px;
height:40px;
background:url(menu_style/down.gif) no-repeat right top;}
.menu li a.top_link:hover {
color:#000;
background:url(menu_style/button4.gif) no-repeat;}
.menu li a.top_link:hover span {
background:url(menu_style/button4.gif) no-repeat right top;}
.menu li a.top_link:hover span.down {
background:url(menu_style/button4a.gif) no-repeat right top;}

.menu li:hover > a.top_link {
color:#000;
background:url(menu_style/button4.gif) no-repeat;}
.menu li:hover > a.top_link span {
background:url(menu_style/button4.gif) no-repeat right top;}
.menu li:hover > a.top_link span.down {
background:url(menu_style/button4a.gif) no-repeat right top;}
```

```
.menu table {
border-collapse:collapse;
width:0;
height:0;
position:absolute;
top:0;
left:0;}

.menu a:hover {
visibility:visible;}
.menu li:hover {
position:relative;
z-index:200;}

.menu ul,
.menu :hover ul ul,
.menu :hover ul :hover ul ul,
.menu :hover ul :hover ul :hover ul ul,
.menu :hover ul :hover ul :hover ul :hover ul ul {
position:absolute;
left:-9999px;
top:-9999px;
width:0;
height:0;
margin:0;
padding:0;
list-style:none;}

.menu :hover ul.sub {
left:2px;
top:40px;
right:2px;
background: #fff;
padding:3px 0;
border:1px solid #999999;
white-space:nowrap;
width:200px;
height:auto;}
.menu :hover ul.sub li {
display:block;
```



```
height:20px;
position:relative;
float:left;
width:250px;}
.menu :hover ul.sub li a {
font-weight:normal;
display:block;
font-size:13px;
height:20px;
width:192px;
line-height:20px;
text-indent:5px;
color:#000;
text-decoration:none;
border:3px solid #fff;
border-width:0 0 0 3px;}
.menu :hover ul.sub li a.fly {
background:#fff url(menu_style/arrow.gif) 80px 7px no-repeat;}
.menu :hover ul.sub li a:hover {
background:#999999; color:#fff;}
.menu :hover ul.sub li a.fly:hover {
background:#999999 url(menu_style/arrow_over.gif) 80px 7px no-repeat;
color:#fff;}
.menu :hover ul li:hover > a.fly {
background:#999999 url(menu_style/arrow_over.gif) 80px 7px no-repeat;
color:#fff;}

.menu :hover ul :hover ul,
.menu :hover ul :hover ul :hover ul,
.menu :hover ul :hover ul :hover ul :hover ul,
.menu :hover ul :hover ul :hover ul :hover ul :hover ul {
left:90px;
top:-4px;
background: #fff;
padding:3px 0;
border:1px solid 999999;
white-space:nowrap;
width:93px;
z-index:200;
height:auto;}
```

Το στυλ που δημιούργησε το κάθετο μενού:

```
#menu2 {
width: 300px;
border-style: solid solid none solid;
border-color: #94AA74;
border-width: 1px;

}
#menu2 ul{
list-style:none;
margin:0px;
padding:0px;
}
#menu2 li a {
height: 32px;
voice-family: "\"\");\"";
voice-family: inherit;
height: 24px;
text-decoration: none;
font-weight:normal;
font-size: 12px;
color: white;
}
#menu2 li a:link, #menu li a:visited {
color: white;
display: block;
background: url(dropdown_menu/v7.gif);
background-repeat:no-repeat;
padding: 8px 0 0 50px;
}
#menu2 li a:hover {
color: white;
background: url(dropdown_menu/v7.gif) 0 -32px;
background-repeat:no-repeat;
padding: 8px 0 0 50px;
}
#menu2 li a:active {
color: white;
background: url(dropdown_menu/v7.gif) 0 -64px;
background-repeat:no-repeat;
```

```
padding: 8px 0 0 50px;
}
```

### 3.2.1 Δημιουργία Ιστοσελίδων & Υλοποίηση του Μηχανισμού Αποστολής Φορμών

Οι παραπάνω γλώσσες που αναδείχθηκαν αποτελούν πολύ δυνατά εργαλεία, κάθε ένα στον τομέα του. Ο συνδυασμός τους δίνει τρομερές δυνατότητες στον σχεδιαστή για δημιουργία δυναμικού web περιεχομένου. Η συνεργασία τους ουσιαστικά περιλαμβάνει την μεσολάβηση της PHP ώστε να επικοινωνεί ο χρήστης με μια βάση δεδομένων μέσα από τον φυλλομετρητή του.

Στην ενότητα αυτή θα δούμε τον συναδυασμό των γλωσσών αυτών στις ιστοσελίδες του ιστοτόπου.

Κάθε αίτηση είναι ένα ξεχωριστό html έγγραφο το οποίο περιέχει μια φόρμα. Πατώντας το κουμπί υποβολής κάθε στοιχείο της φόρμας αποστέλεται μέσω ηλεκτρονικού ταχυδρομείου στην διεύθυνση που εμείς θέλουμε.

Παρακάτω παρουσιάζονται επιλεκτικά κάποιες από τις φόρμες.

#### aitisi\_analitikis\_bathmologias.html

```
<div id="rightcol">
<form action="send_aitisi_analitikis_bathmologias.php" method="get">
<fieldset class="questions" >
<legend>Αίτηση Αναλυτικής Βαθμολογίας</legend>
<div id="innerleftcol" >
<br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Προσωπικά στοιχεία</legend>
<table id="info" border="1" >
<tr><td><label for="surname">Επώνυμο:</label></td>
<td><input type="text" class="textfield" name="surname" id="surname"
size="15" tabindex="1" /></td></tr>
<tr><td><label for="name">Όνομα:</label></td>
<td><input type="text" class="textfield" name="name" id="name" size="15"
tabindex="2" /></td></tr>
<tr><td><label for="fname">Πατρώνυμο:</label></td>
<td><input type="text" class="textfield" name="fname" id="fname" size="15"
tabindex="3" /></td></tr>
<tr><td><label for="mname">Μητρώνυμο:</label></td>
<td><input type="text" class="textfield" name="mname" id="mname" size="15"
tabindex="4" /></td></tr>
```

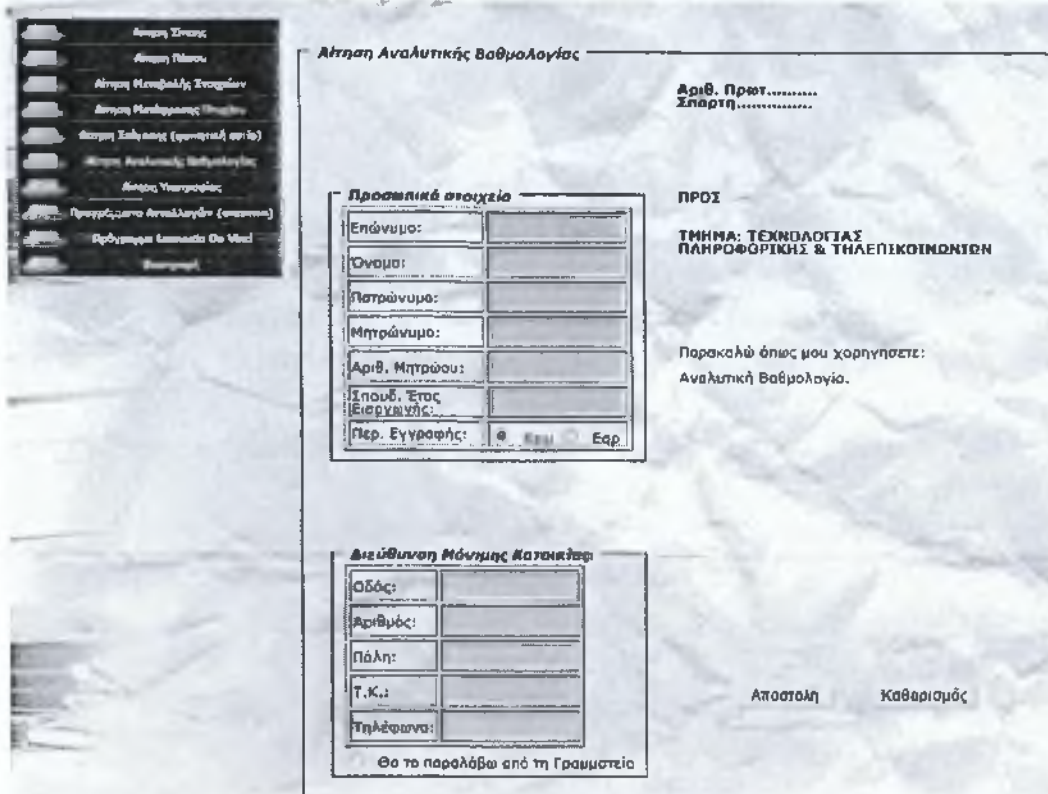
```

<tr><td><label for="am">Αριθ. Μητρώου:</label></td>
<td><input type="text" class="textfield" name="am" id="am" size="15"
tabindex="5" /></td></tr>
<tr><td><label for="address">Σπουδ. Έτος Εισαγωγής:</label></td>
<td><input type="text" class="textfield" name="year" id="year" size="15"
tabindex="6" /></td></tr>
<tr><td><label for="eggrafi">Περ. Εγγραφής:</label></td>
<td><input type="radio" name="eggrafi" id="eggrafi" value="X"
checked="checked" tabindex="7" /> &nbsp;&nbsp;&nbsp;Χειμ
<input type="radio" name="eggrafi" id="eggrafi" value="E" tabindex="5" />
&nbsp;&nbsp;&nbsp;Εαρ</td></tr></table></fieldset><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Διεύθυνση Μόνιμης Κατοικίας:</legend>
<table id="info" border="1" >
<tr><td><label for="road">Οδός:</label></td>
<td><input type="text" class="textfield" name="road" id="road" size="15"
tabindex="1" /></td></tr>
<tr><td><label for="number">Αριθμός:</label></td>
<td><input type="text" class="textfield" name="number" id="number" size="15"
tabindex="2" /></td></tr>
<tr><td><label for="city">Πόλη:</label></td>
<td><input type="text" class="textfield" name="city" id="city" size="15"
tabindex="3" /></td></tr>
<tr><td><label for="t.k">Τ.Κ.:</label></td>
<td><input type="text" class="textfield" name="tk" id="tk" size="15"
tabindex="4" /></td></tr>
<tr><td><label for="phone">Τηλέφωνο:</label></td>
<td><input type="text" class="textfield" name="phone" id="phone" size="15"
tabindex="5" /></td></tr></table>
<input type="radio" name="paralavi" id="paralavi" tabindex="6" /> &nbsp;&nbsp;&nbsp;Θα
το παραλάβω από τη Γραμματεία
</fieldset>
</div>
<div id="innerrightcol" >
<h4>Αριθ. Πρωτ.....<br /> Σπάρτη.....</h4><br /><br /><br />
<b>ΠΡΟΣ</b><br /><br />
<p><b>ΤΜΗΜΑ: ΤΕΧΝΟΛΟΓΙΑΣ <br /> ΠΛΗΡΟΦΟΡΙΚΗΣ &#38;
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ</b></p><br /><br /><br /><br /><br /><br />
Παρακαλώ όπως μου χορηγήσετε:<br /><br />

```







veveosi\_prosagogis\_ejetazomenou.html

```

<div id="rightcol">
  <form id="pe" action="send_veveosis_prosagogis.php" method="get"
onsubmit="return validatePE()" >
  <div id="vpe">
  <fieldset class="questions" >
  <legend>Βεβαίωση Προσαγωγής Εξεταζόμενου</legend><br /><br /><br />
  
  <h4>ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ <br /> ΥΠΟΥΡΓΕΙΟΥ ΠΑΙΔΕΙΑΣ <br /><small>ΔΙΑ ΒΙΟΥ
ΜΑΘΗΣΗΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ</small></h4>
  <br />
  <h4>ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ <br /> ΤΕΙ ΚΑΛΑΜΑΤΑΣ <br />ΠΑΡΑΡΤΗΜΑ
ΣΠΑΡΤΗΣ <br />
  ΤΜΗΜΑ: ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ </h4>
  <p></p><br /><br />
  <p style="text-align: center; font-size: 20px;" > ΒΕΒΑΙΩΣΗ </p><p></p>
  <p>Βεβαιώνεται ότι ο/η
  <input type="text" class="textfield" name="name" id="name" size="40"
tabindex="1" /> του
  <input type="text" class="textfield" name="fname" id="fname" size="12"
tabindex="2" /> σπουδαστής/ σπουδάστριά του παραπάνω Τμήματος,
  
```

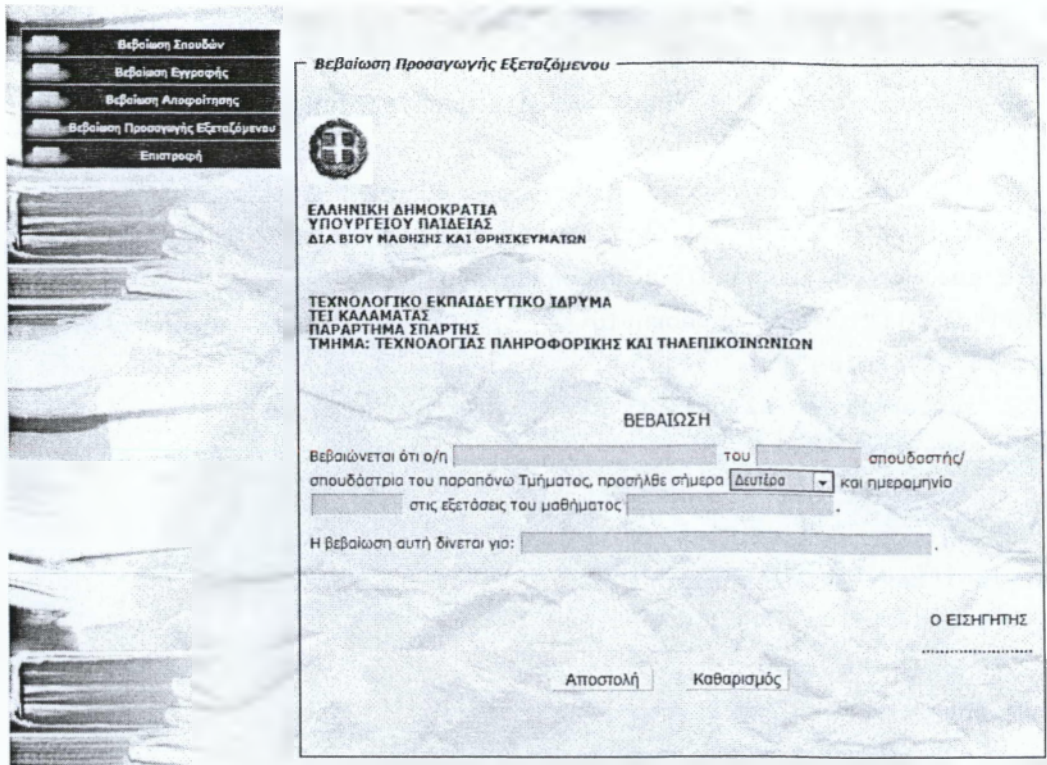




```

<h4>ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ <br /> ΤΕΙ ΚΑΛΑΜΑΤΑΣ <br /> ΠΑΡΑΡΤΗΜΑ
ΣΠΑΡΤΗΣ <br />
ΤΜΗΜΑ: ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ </h4>
<p></p><br />
<p style="text-align: center; font-size: 20px;">ΒΕΒΑΙΩΣΗ <p></p>
<p>Βεβαιώνεται ότι ο/η " $_GET["name"]." του " $_GET["fname"]." σπουδαστής/
σπουδάστριά του παραπάνω Τμήματος, προσήλθε σήμερα"
    $_GET["day"]." και ημερομηνία " $_GET["date"]." στις εξετάσεις του
μαθήματος" $_GET["clas"].".
    <p style="text-align:left;"> Η βεβαίωση αυτή δίνεται
για:" $_GET["cause"]."<br /><br />
    <p style="text-align:right;">Ο ΕΙΣΗΓΗΤΗΣ <br /><br
/>.....</p>
<p style="text-align:center;">
</div>";
$msg .="<br /><br />";
$recipient="fyspart@gmail.com";
$subject="Βεβαίωση Προσαγωγής Εξεταζόμενου";
$mailheaders="MIME-Version: 1.0\r\n";
$mailheaders .="Content-Type: text/html; charset=ISO-8859-1\r\n";
$mailheaders .="Reply to: " $_GET["email"];
mail($recipient, $subject, $msg, $mailheaders);
?>
</body>
</html>

```



katoxirosi\_thematos.html

```

<div id="rightcol">
<form action="send_katoxirosis_thematos.php" method="get">
<fieldset class="questions" >
<legend>Αίτηση Κατοχύρωσης Θέματος</legend>
<div id="innerleftcol" >
<br /><br /><br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Προσωπικά Στοιχεία</legend>
<table id="info" border="1" >
<tr><td><label for="surname">Επώνυμο:</label></td>
<td><input type="text" class="textfield" name="surname" id="surname"
size="20" tabindex="1" /></td></tr>
<tr><td><label for="name">Όνομα:</label></td>
<td><input type="text" class="textfield" name="name" id="name" size="10"
tabindex="2" /></td></tr>
<tr><td><label for="fname">Όνομα Πατρός:</label></td>
<td><input type="text" class="textfield" name="fname" id="fname" size="10"
tabindex="3" /></td></tr>
<tr><td><label for="mname">Όνομα Μητρός:</label></td>
<td><input type="text" class="textfield" name="mname" id="mname" size="10"
tabindex="4" /></td></tr>

```

```

<tr><td><label for="birpl">Τόπος Γέννησης:</label></td>
<td><input type="text" class="textfield" name="birpl" id="birpl" size="20"
tabindex="5" /></td></tr>
<tr><td><label for="adt">Α.Δ.Τ.:</label></td>
<td><input type="text" class="textfield" name="adt" id="adt" size="6"
tabindex="6" /></td></tr>
<tr><td><label for="katoikia">Τόπος Κατοικίας:</label></td>
<td><input type="text" class="textfield" name="katoikia" id="katoikia"
size="10" tabindex="7" /></td></tr>
<tr><td><label for="road">Οδός:</label></td>
<td><input type="text" class="textfield" name="road" id="road" size="15"
tabindex="8" /></td></tr>
<tr><td><label for="code">Ταχ. Κώδικας:</label></td>
<td><input type="text" class="textfield" name="code" id="code" size="5"
tabindex="9" /></td></tr>
<tr><td><label for="phone">Τηλέφωνο:</label></td>
<td><input type="text" class="textfield" name="phone" id="phone" size="10"
tabindex="10" /></td></tr></table></fieldset><br /><br />
<b>Θέμα:</b> Ανάθεση Πτυχιακής Εργασίας
<p><b>Συνημμένα:</b> Αναλυτική Βαθμολογία </p><br /><br /><br /><br />
<b>Ημερομηνία:</b>.../.../.... <br /><br /><br /><br /><br />
</div>
<div id="innerrightcol" >
<h4>ΠΡΟΣ <br /><br />ΤΕΙ ΚΑΛΑΜΑΤΑΣ ΠΑΡΑΡΤΗΜΑ ΣΠΑΡΤΗΣ</h4>
<br /><br /><br />
Παρακαλώ όπως εγκρίνεται την πτυχιακή μου εργασία με θέμα:<br />
<textarea id="theme" name="theme" rows="4" cols="50"
tabindex="11"></textarea><br /><br />
που οφείλω να παραδώσω ολοκληρωμένη εντός έτους από την ημερομηνία έγκρισης
της αίτησής μου με δυνατότητα παράτασης (υπό προϋποθέσεις) κατά ένα εξάμηνο.<br />
<br /><br /><br /><br />
<b>Δηλώνω υπεύθυνα ότι έχω ολοκληρώσει επιτυχώς το μάθημα &laquo;Σεμινάριο
Τελειοφοίτων&raquo; και ότι πληρώ τις λοιπές προϋποθέσεις για την έναρξη
εκπόνησης πτυχιακής εργασίας. </b><br /><br /><br /><br />
Ο/Η ΑΙΤΩΝ/ΟΥΣΑ <input type="text" class="textfield" name="aiton" id="aiton"
size="20" tabindex="12" /><br /><br /><br /><br />
Ο ΕΙΣΗΓΗΤΗΣ <input type="text" class="textfield" name="isigitis"
id="isigitis" size="20" tabindex="13" /><br /><br /><br /><br />
ΠΛΗΡΕΙ ΤΙΣ ΠΡΟΥΠΟΘΕΣΕΙΣ <br />(ΑΠΟ ΤΗ ΓΡΑΜΜΑΤΕΙΑ)<br /><br /><br />
.....<br /><br /><br /><br />

```



```

        <tr><td>Τηλέφωνο:</td><td>"
.$_GET["phone"]."</td></tr></table></fieldset><br /><br /><br /><br /><br />
        <b>Θέμα:</b> Ανάθεση Πτυχιακής Εργασίας
        <p><b>Συνημμένα:</b> Αναλυτική Βαθμολογία </p><br /><br /><br /><br />
        <b>Ημερομηνία:</b>.../.../.... <br /><br /><br /><br /><br /></div>
        <div style="width: 370px; text-align: justify; font-size: 16px; padding-
left: 20px; padding-right: 20px; float: left;">
        <h4>ΠΡΟΣ <br /><br />ΤΕΙ ΚΑΛΑΜΑΤΑΣ ΠΑΡΑΡΤΗΜΑ ΣΠΑΡΤΗΣ</h4>
        <br /><br /><br />
        Παρακαλώ όπως εγκρίνεται την πτυχιακή μου εργασία με θέμα:<br
/>".$_GET["theme"]."<br /><br />
        που οφείλω να παραδώσω ολοκληρωμένη εντός έτους από την ημερομηνία έγκρισης
της αίτησής μου με δυνατότητα παράτασης (υπό προϋποθέσεις) κατά ένα εξάμηνο.<br
/><br /><br /><br />
        <b>Δηλώνω υπεύθυνα ότι έχω ολοκληρώσει επιτυχώς το μάθημα &laquo;Σεμινάριο
Τελειοφοίτων&raquo; και ότι πληρώ τις λοιπές προϋποθέσεις για την έναρξη
εκπόνησης πτυχιακής εργασίας. </b><br /><br /><br /><br />
        Ο/Η ΑΙΤΩΝ/ΟΥΣΑ ".$_GET["aiton"]."<br /><br /><br /><br />
        Ο ΕΙΣΗΓΗΤΗΣ ".$_GET["isigitis"]."<br /><br /><br /><br />
        ΠΛΗΡΕΙ ΤΙΣ ΠΡΟΫΠΟΘΕΣΕΙΣ <br />(ΑΠΟ ΤΗ ΓΡΑΜΜΑΤΕΙΑ)<br /><br /><br />
        .....<br /><br /><br /><br />
        Ο ΠΡΟΪΣΤΑΜΕΝΟΣ <br /><br /><br />..... <br /><br /><br /><br />
/>
        <br /><br /><br /><br /><br />

</div>";
$msg .="<br /><br />";
$recipient="fysparti@gmail.com";
$subject="Αίτηση Κατοχύρωσης Θέματος Πτυχιακής";
$mailheaders="MIME-Version: 1.0\r\n";
$mailheaders .= "Content-Type: text/html; charset=ISO-8859-1\r\n";
$mailheaders .= "Reply to: ".$_GET["email"];
mail($recipient, $subject, $msg, $mailheaders);
?>
</body>
</html>

```

Αίτηση Κατάθεσης Πτυχιακής

Επιστροφή

**Αίτηση Καταχώρησης Θέματος**

**Προσωπικά Στοιχεία**

Επώνυμο:	<input type="text"/>
Όνομα:	<input type="text"/>
Όνομα Πατρός:	<input type="text"/>
Όνομα Μητρός:	<input type="text"/>
Τόπος Γέννησης:	<input type="text"/>
Α.Δ.Τ.:	<input type="text"/>
Τόπος Κατοικίας:	<input type="text"/>
Οδός:	<input type="text"/>
Ταχ. Κωδικός:	<input type="text"/>
Τηλέφωνο:	<input type="text"/>

**Θέμα:** Ανάθεση Πτυχιακής Εργασίας  
**Συνημμένα:** Αναλυτική Βαθμολογία

**Ημερομηνία:**.../.../....

**ΠΡΟΣ**  
**ΤΕΤΙ ΚΑΛΑΜΑΤΑΣ ΠΑΡΑΡΤΗΜΑ**  
**ΣΠΑΡΤΗΣ**

Παρακαλώ όπως εγκρίνεται την πτυχιακή μου εργασία με θέμα:

που σφείλω να παραδώσω ολοκληρωμένη εντός έτους από την ημερομηνία έγκρισης της αίτησής μου με δυνατότητα παράτασης (υπό προϋποθέσεις) κατά ένα εξάμηνο.

**Δηλώνω υπεύθυνα ότι έχω ολοκληρώσει επιτυχώς το μάθημα «Σεμινάριο Τελειοφοιτών» και ότι πληρώ τις λοιπές προϋποθέσεις για την έναρξη εκπόνησης πτυχιακής εργασίας.**

O/H ΑΙΤΩΝ/ΟΥΣΑ

O ΕΙΣΗΓΗΤΗΣ

ΠΑΝΡΕΙ ΤΙΣ ΠΡΟΥΠΟΘΕΣΕΙΣ (ΑΠΟ ΤΗ ΓΡΑΜΜΑΤΕΙΑ)

.....

O ΠΡΟΙΣΤΑΜΕΝΟΣ

.....

Αποστολή      Καθορισμός

aitisi\_metavolis\_stixion.html

```

<div id="rightcol">
<form action="send_aitisi_metavolis_stixion.php" method="get">
<fieldset class="questions" >
<legend>Αίτηση Μεταβολής Στοιχείων</legend>
<div id="innerleftcol" >
<br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Προσωπικά στοιχεία</legend>
<table id="info" border="1" >
<tr><td><label for="surname">Επώνυμο:</label></td>
<td><input type="text" class="textfield" name="surname" id="surname"
size="15" tabindex="1" /></td></tr>
<tr><td><label for="name">Όνομα:</label></td>
<td><input type="text" class="textfield" name="name" id="name" size="15"
tabindex="2" /></td></tr>
<tr><td><label for="fname">Πατρώνυμο:</label></td>

```

```

    <td><input type="text" class="textfield" name="fname" id="fname" size="15"
tabindex="3" /></td></tr>
    <tr><td><label for="mname">Μητρώνυμο:</label></td>
    <td><input type="text" class="textfield" name="mname" id="mname" size="15"
tabindex="4" /></td></tr>
    <tr><td><label for="date">Ημ/νία Γέννησης:</label></td>
    <td><input type="text" class="textfield" name="date" id="date" size="15"
tabindex="5" /></td></tr>
    <tr><td><label for="address">Τόπος Γέννησης:</label></td>
    <td><input type="text" class="textfield" name="year" id="year" size="15"
tabindex="6" /></td></tr>
    <tr><td><label for="adt">Α.Α.Τ.:</label></td>
    <td><input type="text" class="textfield" name="adt" id="adt" size="15"
tabindex="7" /></td></tr>
</table></fieldset><br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Διεύθυνση Μόνιμης Κατοικίας:</legend>
<table id="info" border="1" >
<tr><td><label for="road">Τόπος Κατοικίας:</label></td>
<td><input type="text" class="textfield" name="road" id="road" size="15"
tabindex="8" /></td></tr>
<tr><td><label for="number">Οδός&Αριθμός:</label></td>
<td><input type="text" class="textfield" name="number" id="number" size="15"
tabindex="9" /></td></tr>
<tr><td><label for="t.k">Τ.Κ.:</label></td>
<td><input type="text" class="textfield" name="tk" id="tk" size="15"
tabindex="10" /></td></tr>
<tr><td><label for="phone">Τηλέφωνο:</label></td>
<td><input type="text" class="textfield" name="phone" id="phone" size="15"
tabindex="11" /></td></tr></table>
</fieldset>
</div>
<div id="innerrightcol" >
<h4>Αριθ. Πρωτ.....<br /> Σπάρτη.....</h4><br /><br /><br />
<b>ΠΡΟΣ</b><br /><br />
<p><b>ΤΜΗΜΑ: ΤΕΧΝΟΛΟΓΙΑΣ <br /> ΠΛΗΡΟΦΟΡΙΚΗΣ &#38;
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ</b></p><br /><br /><br /><br /><br /><br /><br />
    Παρακαλώ περιγράψτε αναλυτικά τις αλλαγές που θέλετε να κάνετε.
<br />

```

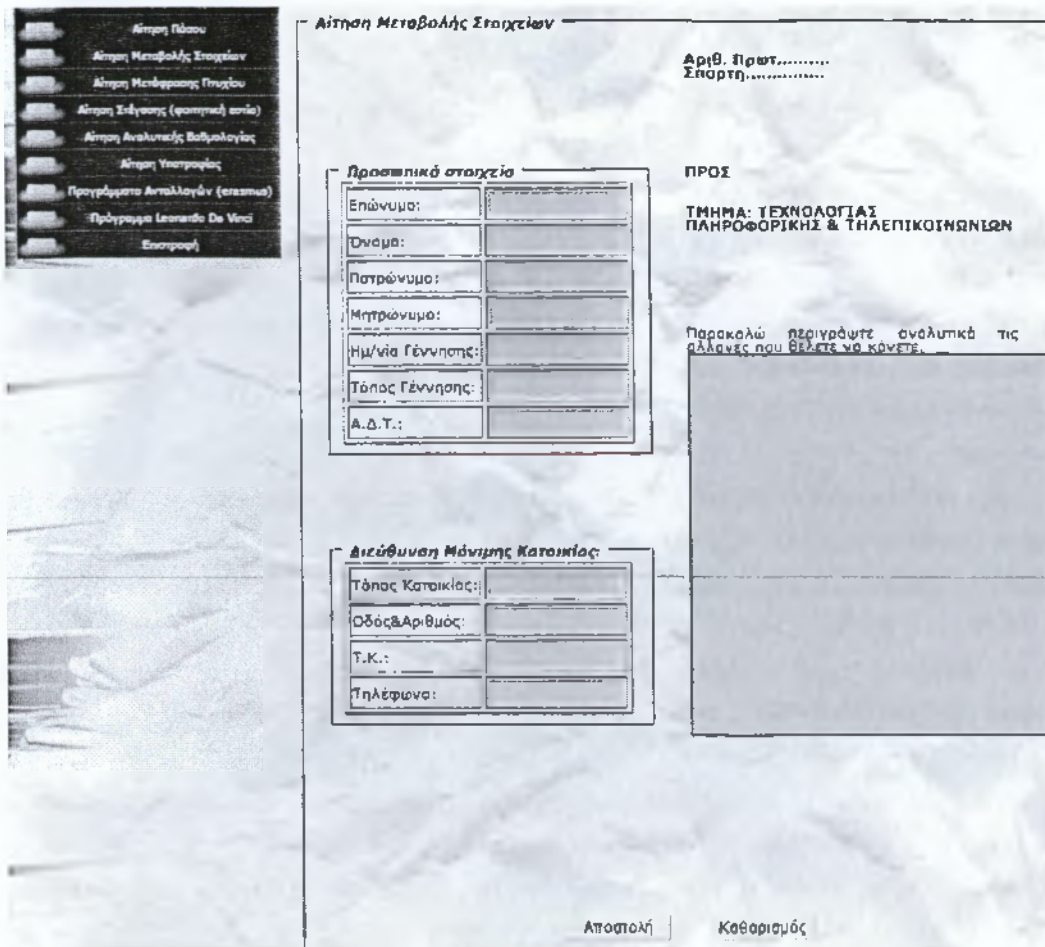




```

        <tr><td>Α.Δ.Τ.:</td><td>"      .$_GET["adt"]."</td></tr></table></fieldset><br
/><br /><br /><br />
        <fieldset style="margin: 12px 0 12px 0; text-align: left; border-color:
black; line-height: 16px;">
        <legend>Διεύθυνση Μόνιμης Κατοικίας:</legend>
        <table style="border: 1px;">
        <tr><td>Τόπος Κατοικίας:</td><td>" .$_GET["road"]."</td></tr>
        <tr><td>Οδός&Αριθμός:</td><td>" .$_GET["number"]."</td></tr>
        <tr><td>Τ.Κ.:</td><td>" .$_GET["tk"]."</td></tr>
        <tr><td>Τηλέφωνο:</td><td>"
.$_GET["phone"]."</td></tr></table></fieldset></div>
        <div style="width: 370px; text-align: justify; font-size: 16px; padding-
left: 20px; padding-right: 20px; float: left;">
        <h4>Αριθ. Πρωτ.....<br /> Σπάρτη.....</h4><br /><br />
        <h4>ΠΡΟΣ <br /><br /></h4>
        <h4><p style="text-align: center;" >Τμήμα Τεχνολογίας Πληροφορικής και
Τηλεπικοινωνιών</p></h4><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
/><br /></div>
        Παρακαλώ περιγράψτε αναλυτικά τις αλλαγές που θέλετε να κάνετε.<br />
        <textarea>" .$_GET["skoli"]."</textarea>
        <br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
/><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
        </div>";
        $msg .="<br /><br />";
        $recipient="fyspart@gmail.com";
        $subject="Αίτηση Μεταβολής Στοιχείων";
        $mailheaders="MIME-Version: 1.0\r\n";
        $mailheaders .="Content-Type: text/html; charset=ISO-8859-1\r\n";
        $mailheaders .="Reply to: " .$_GET["email"];
        mail($recipient, $subject, $msg, $mailheaders);
        ?>
    </body>
</html>

```



profil.php

```

<div id="rightcol">
<fieldset class="questions" >
<legend>Προφίλ</legend>
<div id="innerleftcol" >
<?php
if ($_GET["username"] != "") {
//σύνδεση στη βάση
$con = mysql_connect("mysql8.000webhost.com", "a9873928_fyspart", "sophial")
or die("Η σύνδεση με τη βάση απέτυχε <br />")
mysql_select_db("a9873928_fyspart", $con) or die("Η σύνδεση με τη βάση
απέτυχε <br />")
//query
$query1 = mysql_query("SELECT * FROM users WHERE
username='".$_GET["username"] . "'");
$query2 = "SELECT * FROM yearsignup y, users u WHERE y.A_M = u.A_M";
$query3 = "SELECT * FROM per_residence p, users u WHERE p.A_M = u.A_M";
$query4 = "SELECT * FROM temp_residence t, users u WHERE t.A_M = u.A_M";

```

```

mysql_query("SET NAME 'utf8';", $con);
mysql_query("SET CHARACTER SET 'utf8';", $con);
//επιτέλεση ερωτημάτων
$result1 = mysql_query ($query1, $con);
$result2 = mysql_query ($query2, $con);
$result3 = mysql_query ($query3, $con);
$result4 = mysql_query ($query4, $con);
$num1 = mysql_num_rows($result1);
$num2 = mysql_num_rows($result2);
$num3 = mysql_num_rows($result3);
$num4 = mysql_num_rows($result4);
$row1 = mysql_fetch_array($result1);
$row2 = mysql_fetch_array($result2);
$row3 = mysql_fetch_array($result3);
$row4 = mysql_fetch_array($result4);
?>
<br /><br /><br /><br />
<fieldset class="questions" >
<legend>Προσωπικά στοιχεία</legend>
<table id="info" border="1" >
<tr><td><label for="username">Username:</label></td>
<td><?php echo $row["username"]; ?></td></tr>
<tr><td><label for="surname">Επώνυμο:</label></td>
<td><?php echo $row["surname"]; ?></td></tr>
<tr><td><label for="name">Όνομα:</label></td>
<td><?php echo $row["f_name"]; ?></td></tr>
<tr><td><label for="am">Αριθ. Μητρώου:</label></td>
<td><?php echo $row["A_M"]; ?></td></tr>
<tr><td><label for="semester">Εξάμηνο:</label></td>
<td><?php echo $row["semester"]; ?></td></tr>
<tr><td><label for="direction">Κατεύθυνση:</label></td>
<td><?php echo $row["direction"]; ?></td></tr></table></fieldset><br /><br />
<fieldset class="questions" >
<legend>Στοιχεία Εγγραφής:</legend>
<table id="info" border="1" >
<tr><td><label for="in_year">Ακαδημαϊκό Έτος:</label></td>
<td><?php echo $row["in_year"]; ?></td></tr>
<tr><td><label for="period">Περίοδος:</label></td>
<td><?php echo $row["period"]; ?></td></tr>
<tr><td><label for="city">Τρόπος Εγγραφής:</label></td>

```

```

<td><?php echo $row["singup_way"];?></td></tr>
<tr><td><label for="t.k">Εξάμηνο:</label></td>
<td><?php echo $row["s_semester"];?></td></tr></table></fieldset></div>
<div id="innerrightcol" >
<br /><br /><br /><br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Διεύθυνση Μόνιμης Κατοικίας:</legend>
<table id="info" border="1" >
<tr><td><label for="street">Οδός:</label></td>
<td><?php echo $row["street"];?></td></tr>
<tr><td><label for="city">Πόλη:</label></td>
<td><?php echo $row["city"];?></td></tr>
<tr><td><label for="t.k">Τ.Κ.:</label></td>
<td><?php echo $row["Z.C."];?></td></tr>
<tr><td><label for="country">Χώρα:</label></td>
<td><?php echo $row["country"]; ?></td></tr>
<tr><td><label for="res_per_phone">Τηλέφωνο:</label></td>
<td><?php echo $row["res_per_phone"]; ?></td></tr></table>
</fieldset>
<br /><br /><br /><br /><br />
<fieldset class="questions" >
<legend>Διεύθυνση Προσωρινής Κατοικίας:</legend>
<table id="info" border="1" >
<tr><td><label for="street">Οδός:</label></td>
<td><?php echo $row["street"];?></td></tr>
<tr><td><label for="city">Πόλη:</label></td>
<td><?php echo $row["city"];?></td></tr>
<tr><td><label for="t.k">Τ.Κ.:</label></td>
<td><?php echo $row["Z.C."];?></td></tr>
<tr><td><label for="country">Χώρα:</label></td>
<td><?php echo $row["country"];?></td></tr>
<tr><td><label for="temp_phone">Σταθερό Τηλέφωνο:</label></td>
<td><?php echo $row["temp_phone"];?></td></tr>
<tr><td><label for="cell_phone">Κινητό Τηλέφωνο:</label></td>
<td><?php echo $row["cell_phone"];?></td></tr></table>
</fieldset></div>
<?php
)
mysql_close($con);
)
?>

```

```

    <br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
    /><br /><br /><br /> <br /><br /><br /><br /><br /><br /><br /><br /><br /> <br /><br /><br /><br /><br /><br /><br /><br /><br />
    /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
    /><br /> <br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br /><br />
    /><br /><br /><br /><br />
    <p>Αν θέλετε Μεταβολή των Στοιχείων πατήστε <a
href="aitisi_metavolis_stixion.html" title="Μεταβολή Στοιχείων">εδώ!</a> </p>
</fieldset>
</div>

```

Στην αρχή του κώδικα πραγματοποιείται σύνδεση με την βάση δεδομένων. Με την χρήση Sql ερωτημάτων διαλέγουμε από την βάση τα στοιχεία που θέλουμε να εμφανίσουμε στην σελίδα μας. Ακολουθεί το css που χρησιμοποιήθηκε για το στυλ των σελίδων.

Η κλάση textfield δημιουργήθηκε για το στυλ που θα έχει το κείμενο συμπλήρωσης της φόρμας.

```

.textfield {
font-size: 16px;
background-color: silver;
color: black;
border: 1px solid inset;
}

```

Η κλάση textarea δημιουργήθηκε για το στυλ που θα έχει το κείμενο για την περιοχή σχολίων.

```

.textarea {
font-size: 16px;
background-color: silver;
color: black;
}

```

Στυλ για το dropdown menu της φόρμας.

```

select {
font-size: 16px;
background-color: silver;
color: black;
border: 1px solid inset;
min-width: 70px;
}

```

```
}
```

Η κλάση `question` δημιουργήθηκε για το στυλ της ετικέτας `fieldset`.

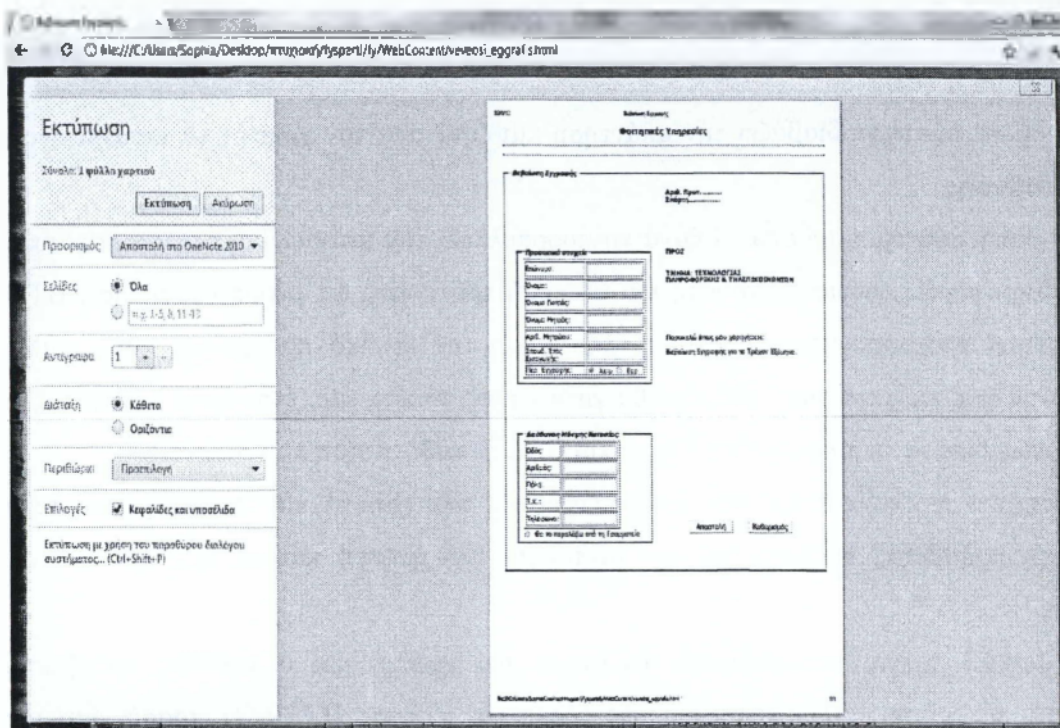
```
.questions {  
margin: 12px 0 12px 0;  
text-align: left;  
border-color: black;  
line-height: 16px;  
}
```

```
legend {  
padding: 0 10px 0 10px;  
color: black;  
font-weight: bold;  
font-style: italic;  
}
```

Επίσης στο CSS μπορούμε να ελέγξουμε πιο `div` θα τυπώσουμε. Στην περίπτωση αυτή μπορεί ο χρήστης να θέλει να τυπώσει μόνο την αίτηση. Για το λόγο αυτό θα πρέπει να τυπώνεται μόνο το `div` που βρίσκεται η αίτηση. Ο κώδικας για να το πετύχουμε αυτό ακολουθεί παρακατώ:

```
@media print {  
#container{display: table-row-group;}  
#ifl{display: none;}  
#ifr{display: none;}  
#leftcol{display: none;}  
#rightcol{display: block;}  
}  
  
}  
  
@media screen, print {  
#rightcol {display: block; margin: auto; width: 1200px;}  
  
@media print {  
body {font-family:arial, verdana, sans-serif;}
```

```
.menu {display: none;}
.bg {display: none;}
}
```



### 3.3 Έλεγχος της πρόσβασης

Για τον ιστότοπο θα δημιουργήσουμε ένα μηχανισμό για τον έλεγχο της ταυτότητας των χρηστών. Η πιστοποίηση (authentication) και η εξουσιοδότηση (authorization) είναι δύο κοινές απαιτήσεις για πολλές εφαρμογές στο διαδίκτυο.

Η πιστοποίηση χρησιμοποιείται ως ένας μηχανισμός για την επιβεβαίωση της ταυτότητας των συμβαλλόμενων μερών σε μία σύνοδο επικοινωνίας. Στην συγκεκριμένη εφαρμογή θα γίνει πιστοποίηση των χρηστών με το όνομα χρήστη και τον κωδικό πρόσβασης.

Η εξουσιοδότηση ασχολείται με τον έλεγχο της πρόσβασης σε διάφορους πόρους. Μπορεί να δοθεί εξουσιοδότηση βάση της διεύθυνσης IP του χρήστη, την εφαρμογή browser του χρήστη, το περιεχόμενο που προσπαθεί να προσπελάσει ο χρήστης ή η ταυτότητα του χρήστη την οποία χρησιμοποιούμε εδώ.

Η πιστοποίηση των χρηστών χρησιμοποιείται για σκοπούς παρακολούθησης ή ελέγχου πρόσβασης. Η προδιαγραφή του πρωτοκόλλου HTTP παρέχει δύο μηχανισμούς πιστοποίησης: βασικό και σύνοψης. Και στις δύο περιπτώσεις η διαδικασία που ακολουθείται είναι η εξής:

• Ένα client σύστημα προσπαθεί να προσπελάσει προστατευμένο περιεχόμενο στον Web Server.

• Ο Web Server ελέγχει εάν το client σύστημα παρέχει όνομα χρήστη και κωδικό πρόσβασης. Εάν όχι, επιστρέφει τον κωδικό κατάστασης της HTTP 401 ο οποίος υποδεικνύει ότι απαιτείται πιστοποίηση του χρήστη.

• Το client σύστημα διαβάζει την απάντηση και ζητά από τον χρήστη να εισάγει όνομα και κωδικό πρόσβασης.

• Το client σύστημα προσπαθεί ξανά να προσπελάσει την ιστοσελίδα, στέλνοντας αυτήν την φορά τις πληροφορίες ονόματος χρήστη και κωδικού πρόσβασης ως μέρος της αίτησης HTTP. Το client σύστημα απομνημονεύει το όνομα χρήστη και τον κωδικό πρόσβασης και μεταδίδει τα στοιχεία αυτά στις επόμενες αιτήσεις που θα γίνουν προς το ίδιο site, έτσι ώστε ο χρήστης να μην είναι υποχρεωμένος να τα πληκτρολογεί ξανά και ξανά σε κάθε αίτηση που στέλνει.

• Ελέγχεται η εγκυρότητα των διαπιστευτηρίων του χρήστη και παραχωρεί πρόσβαση ή αρνείται την πρόσβαση ανάλογα με την ταυτότητα του χρήστη και άλλους κανόνες ελέγχου πρόσβασης.

Στο βασικό σχήμα πιστοποίησης, το όνομα του χρήστη και ο κωδικός πρόσβασής του μεταδίδονται σε μορφή απλού κειμένου, ως μέρος της αίτησης HTTP. Η προσέγγιση αυτή θα μπορούσε να είναι επικίνδυνη από την άποψη της ασφάλειας, επειδή ένας εισβολέας θα μπορούσε πολύ εύκολα να υποκλέψει την επικοινωνία που διαμειβεται μεταξύ του server και της εφαρμογής browser, να μάθει το όνομα χρήστη και τον κωδικό πρόσβασης και να τα χρησιμοποιήσει για οποιονδήποτε κακό σκοπό.

Το δεύτερο σχήμα πιστοποίησης παρέχει αυξημένη ασφάλεια επειδή μεταδίδει μάλιστα συνοψη αντί για τον κωδικό πρόσβασης σε μορφή απλού κειμένου. Η συνοψη βασίζεται σ' έναν συνδυασμό αρκετών παραμέτρων, συμπεριλαμβανομένου του ονόματος χρήστη, του κωδικού πρόσβασης και της μεθόδου της αίτησης. Εξετάζοντας την συνοψη, ο server μπορεί να ελέγξει ότι το client σύστημα γνωρίζει τον κωδικό πρόσβασης, ακόμη κι αν ο κωδικός πρόσβασης δεν μεταδίδεται μέσω του δικτύου. Μόνο οι πολύ πρόσφατες εφαρμογές browser υποστηρίζουν πιστοποίηση με την χρήση ενός αλγόριθμου συνοψης.

### 3.3.1 Έλεγχος της πρόσβασης με βάση τις τιμές των Cookies

Στην ενότητα αυτή θα δημιουργήσουμε τη φόρμα εισαγωγής στοιχείων σύνδεσης του χρήστη, η οποία θα ελέγχει τις τιμές που εισάγουν οι χρήστες έναντι μιας βάσης δεδομένων. Εάν ο χρήστης



πιστοποιηθεί, στέλνεται ένα cookie, το οποίο υποδηλώνει αυτό το γεγονός. Κατόπιν, για όλες τις σελίδες, στις οποίες θέλετε να έχουν πρόσβαση μόνο οι πιστοποιημένοι χρήστες, ελέγχουμε την ύπαρξη αυτού του συγκεκριμένου cookie. Εάν το cookie υπάρχει, ο χρήστης θα μπορεί να δει την σελίδα. Εάν το cookie δεν υπάρχει, μπορείτε να επαναφέρετε τον χρήστη στην φόρμα εισαγωγής των στοιχείων σύνδεσης ή να εμφανιστεί ένα μήνυμα σχετιζόμενο με τους περιορισμούς πρόσβασης στην οθόνη του.

Όταν ενσωματώνουμε λογαριασμούς χρηστών σε μία βασιζόμενη στο web εφαρμογή, μία κοινή πρακτική είναι η αποθήκευση των πληροφοριών για τους χρήστες σε έναν πίνακα μιας βάσης δεδομένων. Οι πληροφορίες αυτού του πίνακα μπορούν κατόπιν να χρησιμοποιούνται για την εξουσιοδότηση των χρηστών, παρέχοντας τους πρόσβαση στους τομείς του site που έχουμε δημιουργήσει ειδικά γι' αυτούς τους χρήστες.

Η ακόλουθη εντολή δημιουργεί τον πίνακα με όνομα users στην βάση δεδομένων MySQL που χρησιμοποιούμε, με πεδία για τον κωδικό πρόσβασης, το username, το e-mail του χρήστη, το όνομα, το επώνυμο, το εξάμηνο και την κατεύθυνση:

```
CREATE TABLE `users` (
  `username` varchar(45) NOT NULL,
  `password` varchar(45) NOT NULL,
  `email` varchar(150) NOT NULL,
  `A_M` int(10) unsigned NOT NULL,
  `f_name` varchar(45) NOT NULL,
  `l_name` varchar(45) NOT NULL,
  `semester` enum('A','B','Γ','Δ','Ε','ΣΤ','Ζ','Η','ΠΤΥΧΙΟ') NOT NULL,
  `direction` enum('ΠΑΗΡΟΦΟΡΙΚΗ','ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ') NOT NULL,
  PRIMARY KEY (`username`,`password`,`A_M`) USING BTREE,
) ENGINE=InnoDB DEFAULT CHARSET=greek;
```

Αφού καταχωρήσουμε τους εξουσιοδοτημένους χρήστες στον πίνακα της βάσης δεδομένων, θα πρέπει να τους παρέχουμε έναν μηχανισμό για να αποδεικνύουν την ταυτότητά τους. Στην περίπτωση μας αρκεί μια απλή φόρμα με δύο πεδία. Ο κώδικας της φόρμας φαίνεται πιο κάτω:

#### loginform.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
```

```

<title>&#934;&#959;&#953;&#964;&#951;&#964;&#953;&#954;&#941;&#962;
&#933;&#960;&#951;&#961;&#949;&#963;&#943;&#949;&#962;</title>
<link rel="stylesheet" href="style.css" type="text/css" />
<link rel="stylesheet" href="menu_style.css" type="text/css" />
<link rel="stylesheet" href="style.css" type="text/css" />
<link rel="stylesheet" href="dropdown_menu.css" type="text/css" />
<!-- [if IE]>
<link href="ie.css" rel="stylesheet" type="text/css" />
<![endif] -->
</head>
<body>
<div id="container">
<h3>&#934;&#959;&#953;&#964;&#951;&#964;&#953;&#954;&#941;&#962;
&#933;&#960;&#951;&#961;&#949;&#963;&#943;&#949;&#962;</h3><hr></hr>

<hr /><br /><br /><br /><br /><br /><br /><br /><br />

<table width="300" border="4" align="center" cellpadding="0" cellspacing="1"
bgcolor="#CCCCCC">
<tr>
<td>
<form name="form1" method="post" action="userlogin.php">
<td>
<table width="100%" border="0" cellpadding="3" cellspacing="1"
bgcolor="#FFFFFF">
<tr>
<td colspan="3" ><strong>&#917;&#943;&#963;&#959;&#948;&#959;&#962;
&#934;&#959;&#953;&#964;&#951;&#964;&#942; </strong></td>
</tr>
<tr>
<td width="78">Username</td>
<td width="6">:</td>
<td width="294"><input name="username" type="text" id="username" /></td>
</tr>
<tr>
<td>Password</td>
<td>:</td>
<td><input name="password" type="text" id="password" /></td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;</td>

```

```

<td><input type="submit" name="Submit" value="Login" /></td>
</tr>
</table>
</td>
</form>
</tr>
</table>
</body>
</html>

```



Ύστερα θα δημιουργήσουμε το κατάλληλο script ώστε να ελέγχει αν έχουν δοθεί τα δύο πεδία στην φόρμα. Εάν οποιοδήποτε από αυτά τα δύο πεδία δεν υπάρχει ο χρήστης θα πηγαίνει σε μία σελίδα λάθους. Στην οποία πατώντας τον κατάλληλο σύνδεσμο θα επανέρχεται στην σελίδα της φόρμας. Το script θα υλοποιεί σύνδεση με την βάση δεδομένων που θα χρησιμοποιηθεί για την εκτέλεση του ερωτήματος sql το οποίο ελέγχει την ταυτότητα του χρήστη.

userlogin.php

```

<?php
$host="mysql8.000webhost.com"; // Host name
$username="a9873928_fyspart"; // Mysql username
$password="sophial"; // Mysql password
$db_name="a9873928_fyspart"; // Database name

```

```
$tbl_name="users"; // Table name

// Connect to server and select databse.

// mysql_connect("$host", "$username", "$password")or die("cannot connect");
$link = mysql_connect('mysql8.000webhost.com', 'a9873928_fyspart',
'sophial');
if (!$link) {
    die('Could not connect: ' . mysql_error());
}
echo ' ';

mysql_select_db("$db_name")or die("cannot select DB");

// username and password sent from form
$username=$_POST['username'];
$password=$_POST['password'];

// To protect MySQL injection (more detail about MySQL injection)
$myusername = stripslashes($username);
$mypassword = stripslashes($password);
$myusername = mysql_real_escape_string($username);
$mypassword = mysql_real_escape_string($password);

$sql="SELECT * FROM $tbl_name WHERE username='$username' and
password='$password'";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);
// If result matched $myusername and $mypassword, table row must be 1 row

if($count==1){
// Register $myusername, $mypassword and redirect to file "index.php"
session_register("username");
session_register("password");
header("location:index.php");
}
else {
echo "Wrong Username or Password <br />";
echo "Press <a href=\"loginform.php\">Here</a> To Go Back";
```

```
}
?>
```

Εάν η σύνδεση είναι επιτυχής το script θα μας πάει στην αρχική σελίδα του ιστότοπου index.php όπου σύμφωνα με το cookie ξεκινά η σύνοδος του χρήστη. Όντα ο χρήστης κλείσει την εφαρμογή browser, το cookie θα λήξει.

index.php

```
<?
```

```
    session_start();
    if(!session_is_registered(username)){
        header("location:loginform.php");
    }
?>
```

```
}
```

```
?>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
```

```
"http://www.w3.org/TR/xhtml2/DTD/xhtml11-strict.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
```

```
<head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
```

```
    <title>Φοιτητικές Υπηρεσίες</title>
```

```
    <link rel="stylesheet" href="style.css" type="text/css" />
```

```
        <link rel="stylesheet" href="menu_style.css" type="text/css" />
```

```
        <link rel="stylesheet" href="style.css" type="text/css" />
```

```
        <link rel="stylesheet" href="dropdown_menu.css" type="text/css" />
```

```
</head>
```

```
<body>
```

```
<div id="container">
```

```
    <h3>Φοιτητικές Υπηρεσίες</h3><hr></hr>
```

```
    <div id="ifl">
```

```
        
```

```
    </div>
```

```
    <div id="ifr">
```

```
        <input type="button" class="button" value="log out" />
```

```
    </div>
```

```
    <div style="clear: both"></div>
```

```
    <hr />
```

```
    <div>
```

```
        <ul class="menu">
```

```
            <li class="top"><a href="index.html"
```

```
class="top_link"><span>Αρχική</span></a>
```

```
            <ul class="sub">
```

```
                <li><a href="anakoinoseis.html">Ανακοινώσεις</a></li>
```

```
                <li><a href="entipa.html">Ενημερωτικά Έντυπα</a></li>
```

```

        <li><a href="programma_spydon.html">Πρόγραμμα
Σπουδών</a></li></ul></li>
        <li class="top"><a href="profil.php"
class="top_link"><span>Προφίλ</span></a></li>
        <li class="top"><a href="pistopoihtika.html"
class="top_link"><span>Πιστοποιητικά</span></a>
        <ul class="sub">
            <li><a href="aitisi_fitisis.html">Αίτηση
Φοίτησης</a></li>
            <li><a href="aitisi_anavolis_stratefsis.html">Αναβολή
Στράτευσης</a></li>
            <li><a href="diakopi_anavolis_stratefsis.html">Διακοπή
Αναβολής Στράτευσης</a></li>
            <li><a href="aitisi_fititikou_epidomatou.html">Αίτηση
Φοιτητικού Επιδόματος</a></li></ul></li>
        <li class="top"><a href="veveoseis.html"
class="top_link"><span>Βεβαιώσεις</span></a>
        <ul class="sub">
            <li><a href="veveosi_spydon.html">Βεβαίωση
Σπουδών</a></li>
            <li><a href="veveosi_eggrafis.html">Βεβαίωση
Εγγραφής</a></li>
            <li><a href="veveosi_apofitisis.html">Βεβαίωση
Αποφοίτησης</a></li>
            <li><a
href="veveosi_prosagogis_ejetazomenou.html">Βεβαίωση
Προσαγωγής</a></li></ul></li>
        <li class="top"><a href="praktiki_askisi.html"
class="top_link"><span>Πρακτική Άσκηση</span></a>
        <ul class="sub">
            <li><a href="aitisi_praktikis_askisis.html">Αίτηση
Πρακτικής Άσκησης</a></li></ul></li>
        <li class="top"><a href="ptixiaki_ergasia.html"
class="top_link"><span>Πτυχιακή Εργασία</span></a>
        <ul class="sub">
            <li><a href="katoxirosi_thematos.html">Αίτηση
Κατοχύρωσης Θεματός</a></li>
            <li><a href="katathesi_ptixiakis.html">Αίτηση
Κατάθεσης Πτυχιακής</a></li></ul></li>
        <li class="top"><a href="alles_aitiseis.html"
class="top_link"><span>Άλλες Αιτήσεις</span></a>
        <ul class="sub">
            <li><a href="aitisi_sitisis.html">Αίτηση
Σίτισης</a></li>
            <li><a href="aitisi_pasou.html">Αίτηση
Πάσου</a></li>
            <li><a href="aitisi_metavolis_stixion.html">Αίτηση
Μεταβολής Στοιχείων</a></li>
            <li><a href="aitisi_metafrasis_ptixiou.html">Αίτηση
Μετάφρασης Πτυχίου</a></li>
            <li><a href="aitisi_stegasis.html">Αίτηση
Στέγασης</a></li>
            <li><a
href="aitisi_analitikis_bathmologias.html">Αίτηση Αναλυτικής Βαθμολογίας</a></li>
            <li><a href="aitisi_ipotrofias.html">Αίτηση
Υποτροφίας</a></li>
            <li><a href="aitisi_erasmus.html">Erasmus</a></li>
            <li><a href="programa_leonardo.html">Πρόγραμμα
Leonardo Da Vinci</a></li></ul></li>
    </ul></div>
    <div id="leftcol">
    <div id="menu2">

```

```
<ul>
    <li><a
href="anakoinoseis.html">Ανακοινώσεις</a></li>
    <li><a href="#">Ενημερωτικά Έντυπα</a></li>
    <li><a href="programma_spydon.html">Πρόγραμμα
Σπουδών</a></li></ul></div>
</div>

<div id="rightcol">

</div>
</div>
</body>
</html>
```

#### 4 Μελλοντικές Βελτιώσεις

Είναι γεγονός ότι ζούμε, πλέον, σε μια εποχή όπου οι εξελίξεις στις τεχνολογίες πληροφορικής και στην κατασκευή ιστοσελίδων αυξάνονται θεαματικά. Στόχος μας ήταν η κατασκευή ενός ιστότοπου και η ανάδειξη των εφαρμογών της πληροφορικής όσον αφορά την κατασκευή ιστοσελίδων και πως κάνουν τις ιστοσελίδες πιο εύκολες στην περιήγησή τους.

Στην παρούσα ιστοσελίδα έχει δοθεί έμφαση στη λειτουργικότητα της εφαρμογής, έτσι ώστε να μειωθεί ο χρόνος απόκρισης της γραμματείας ως προς τους φοιτητές. Όσο τώρα, αφορά στην εμφάνιση αλλά και στις επιλογές πλοήγησης που έχει ο επισκέπτης σε κάθε σελίδα θα μπορούσαν να γίνουν αρκετές βελτιώσεις.

Ακόμα θα μπορούσαν να γίνουν βελτιώσεις όσον αφορά την πιστοποίηση των χρηστών. Και το σύστημα να γίνει πιο ασφαλές. Κάνοντας πιο ασφαλές τον κώδικα μας να ελέγξουμε τα λάθη των χρηστών τα οποία θα γίνουν από αμέλεια ή επιτιδυμένα ώστε τα δεδομένα των χρηστών να είναι ασφαλή.



## 5 Βιβλιογραφία

### Βιβλία:

1. Welling Luke, Ανάπτυξη Web Εφαρμογών με PHP και MySQL, 2005 Γκιούρδας
2. Μάθετε PHP MySQL και Apache σε ένα, 2004 Γκιούρδας
3. Μάθετε Ajax, Javascript και PHP

### Πηγές:

1. MySQL Documentation: MySQL Reference Manuals, <http://dev.mysql.com/doc/>
2. PHP Manual: online documentation, <http://php.net/manual/en/index.php>
3. [www.wikipedia.org](http://www.wikipedia.org) Ηλεκτρονική Βιβλιοθήκη
4. [www.w3schools.com](http://www.w3schools.com) Tutorial γλωσσών προγραμματισμού

### Εργαλεία:

1. Eclipse Version: Helios Service Release 1
2. MySQL Server 5.1