



ΤΕΙ ΗΠΕΙΡΟΥ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΤΕ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Διαδικτυακή πλατφόρμα εκπαίδευσης
των: Html, Php & Sql**

ΣΤΑΥΡΟΥ ΕΛΕΑΝΑ (Α.Μ. 8311)

Επιβλέπων : Χαριλόγης Βασίλειος

ΑΡΤΑ, 2015

Στην αγαπημένη μου φίλη, Έλλη

ΔΗΛΩΣΗ ΠΝΕΥΜΑΤΙΚΗΣ ΙΔΙΟΚΤΗΣΙΑΣ

Η παρούσα εργασία αποτελεί προϊόν αποκλειστικά δικής μου προσπάθειας. Όλες οι πηγές που χρησιμοποιήθηκαν περιλαμβάνονται στη βιβλιογραφία και γίνεται ρητή αναφορά σε αυτές μέσα στο κείμενο όπου έχουν χρησιμοποιηθεί.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΕΦΑΛΑΙΟ 1

1.1 ΕΙΣΑΓΩΓΗ.....	7
1.2 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΔΙΑΔΙΚΤΥΑΚΗΣ ΕΚΠΑΙΔΕΥΣΗΣ.....	7
1.3 ΣΚΟΠΟΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ.....	8
1.4 ΕΡΓΑΛΕΙΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ.....	8
1.4.1 HTML5.....	8
1.4.2 CSS3.....	9
1.4.3 JAVASCRIPT.....	10
1.4.4 PHP.....	11
1.4.5 SQL.....	12

ΚΕΦΑΛΑΙΟ 2

2.1 ΠΡΟΓΡΑΜΜΑΤΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ.....	13
2.1.1 XAMPP.....	13
2.1.2 NOTEPAD++.....	20
2.1.3 GIMP 2.8.....	20

ΚΕΦΑΛΑΙΟ 3

3.1 Η ΒΑΣΙΚΗ ΔΟΜΗ ΤΗΣ HTML5.....	22
3.2 ΒΑΣΙΚΑ HTML ΣΤΟΙΧΕΙΑ.....	26
3.3 ΠΕΡΙΣΣΟΤΕΡΑ HTML ΣΤΟΙΧΕΙΑ.....	27

ΚΕΦΑΛΑΙΟ 4

4.1 CSS – Cascading Style Sheets.....	30
4.2 ΜΟΡΦΟΠΟΙΗΣΗ ΚΕΙΜΕΝΟΥ ΜΕ ΣΤΥΛ.....	33
4.3 ΔΙΑΤΑΞΗ ΣΕΛΙΔΩΝ ΜΕ ΣΤΥΛ.....	35

ΚΕΦΑΛΑΙΟ 5

5.1 Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ.....	39
5.2 ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ.....	40
5.3 ΣΥΝΔΕΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ PHP.....	42

ΚΕΦΑΛΑΙΟ 6

6.1 ΠΑΡΟΥΣΙΑΣΗ ΠΛΑΤΦΟΡΜΑΣ.....	45
6.2 JAVASCRIPT ΚΑΙ HTML.....	47
6.3 ΠΑΡΑΔΕΙΓΜΑΤΑ.....	53
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	58

ΠΕΡΙΛΗΨΗ

Το αντικείμενο της παρούσας πτυχιακής εργασίας είναι η κατασκευή διαδικτυακής εκπαιδευτικής πλατφόρμας των Html, Php και MySql. Πιο συγκεκριμένα, για τη δημιουργία της έγινε χρήση των γλωσσών HTML, CSS, Javascript, PHP και για τη δημιουργία, αποθήκευση και την εμφάνιση πληροφοριών χρησιμοποιήθηκε η γλώσσα βάσεων δεδομένων SQL.

Η πλατφόρμα παρέχει στον επισκέπτη τη δυνατότητα εκμάθησης των γλωσσών μέσα από:

- Θεωρητικό κομμάτι για κάθε γλώσσα
- Παραδείγματα κώδικα
- Δυνατότητα επεξεργασίας του κώδικα

Τα προγράμματα που χρησιμοποιήθηκαν για τη δημιουργία της πλατφόρμας είναι το XAMPP - που περιέχει Apache,PHP και MySql - το Notepad++ και το GIMP.

ΚΕΦΑΛΑΙΟ 1

1.1 ΕΙΣΑΓΩΓΗ

Το διαδίκτυο είναι ένα επικοινωνιακό δίκτυο που επιτρέπει την ανταλλαγή δεδομένων μεταξύ οποιουδήποτε διασυνδεδεμένου υπολογιστή. Η τεχνολογία του είναι κυρίως βασισμένη στην διασύνδεση επιμέρους δικτύων ανά τον κόσμο και πολυάριθμα πρωτόκολλα επικοινωνίας. Στην πιο εξειδικευμένη και περισσότερο χρησιμοποιούμενη μορφή του, με τον όρο διαδίκτυο, περιγράφεται το παγκόσμιο πλέγμα διασυνδεδεμένων υπολογιστών.

Το διαδίκτυο, σε συνδυασμό με την ολοένα αναπτυσσόμενη ψηφιακή τεχνολογία, έχει δημιουργήσει μία τεράστια αγορά γνώσεων/πληροφοριών. Παραδοσιακές μορφές τέχνης (όπως για παράδειγμα ο κινηματογράφος και η μουσική) μέσω της ψηφιακής τεχνολογίας παίρνουν την ίδια μορφή (αρχείων δεδομένων) με αντικείμενα που εκ πρώτης όψεως είναι εντελώς διαφορετικά (όπως για παράδειγμα η ιατρική επιστήμη ή κάποιο πρόγραμμα λογισμικού). Παρατηρείται λοιπόν μία συγκέντρωση γνώσης ή, αν είναι δυνατό να λεχθεί, πολιτιστικής κληρονομιάς, που σχετίζεται άμεσα με το Internet. (Βικιπαίδεια, 2011)

Σήμερα, η εξ αποστάσεως εκπαίδευση υλοποιείται σχεδόν αποκλειστικά με την υποστήριξη του υπολογιστή, και πιο συγκεκριμένα σε διαδικτυακό περιβάλλον. Για το λόγο αυτό, τείνει να είναι ταυτόσημη με τις έννοιες ηλεκτρονική μάθηση (e-learning), μάθηση υποβοηθούμενη από υπολογιστή (computer assisted learning), μάθηση μέσω διαδικτύου (online learning), διαδικτυακή εκπαίδευση (online education), εκπαίδευση βασισμένη στο διαδίκτυο (web-based education). Η διαφορά στη σημασία των όρων αυτών αρχίζει να υποβαθμίζεται και ο διαχωρισμός γίνεται όλο και πιο δύσκολος τόσο για αρχάριους όσο και για έμπειρους. (Βικιπαίδεια, 2015)

1.2 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΔΙΑΔΙΚΤΥΑΚΗΣ ΕΚΠΑΙΔΕΥΣΗΣ

Τα πλεονεκτήματα της διαδικτυακής εκπαίδευσης είναι:

- Το εκπαιδευτικό υλικό είναι πάντα διαθέσιμο, αρκεί να υπάρχει σύνδεση με το διαδίκτυο.
- Είναι παντού διαθέσιμο.
- Είναι διαθέσιμο σε όλους που έχουν στη διάθεση τους απλά μέσα, όπως PC και σύνδεση με το διαδίκτυο ενώ δεν απαιτεί οργανωμένους χώρους εκπαίδευσης.
- Είναι πλούσιο σε περιεχόμενο, περιεκτικό και δεν κουράζει τον εκπαιδευόμενο.
- Είναι εξαιρετικά αποτελεσματικό, όταν γίνεται με προηγμένο τρόπο παρουσίασης: πολυμέσα, βίντεο, ήχος, κείμενα, εικόνες, παραστάσεις, ομιλία, διαλογική συνεργασία.
- Εξασφαλίζεται συνεχής βελτίωση του περιεχομένου και της αποτελεσματικότητας. (Μουρλετίδης, 2010)

1.3 ΣΚΟΠΟΣ ΤΗΣ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Στόχος της παρούσας πτυχιακής εργασίας είναι ο σχεδιασμός και η ανάπτυξη μιας διαδικτυακής πλατφόρμας εκπαίδευσης των Html, Php και MySql, όπου ο επισκέπτης της ιστοσελίδας θα είναι σε θέση να μελετήσει σε βάθος τον κώδικά τους. Η εκπαιδευτική πλατφόρμα παρέχει στον επισκέπτη το θεωρητικό κομμάτι της κάθε γλώσσας, παραδείγματα με επεξήγηση του κώδικα καθώς και τη δυνατότητα επεξεργασίας του κώδικα για την καλύτερη δυνατή εκμάθησή του. Στη συνέχεια θα παρουσιαστεί ο τρόπος με τον οποίο η πλατφόρμα δημιουργήθηκε (τα προγράμματα και τα εργαλεία που χρησιμοποιήθηκαν, ο κώδικας και η επεξήγησή του) καθώς και η ιστοσελίδα μέσα από εικόνες.

1.4 ΕΡΓΑΛΕΙΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ

1.4.1 HTML5

Είναι χρήσιμο να αναφερθούν κάποια βασικά πράγματα σχετικά με την προέλευση της HTML, έτσι ώστε να γίνει κατανοητή η HTML5. Η HTML ξεκίνησε στις αρχές της δεκαετίας του 1990 ως ένα σύντομο έγγραφο που παρέθετε κάποια στοιχεία τα οποία χρησιμοποιούνταν για την κατασκευή ιστοσελίδων. Πολλά από τα στοιχεία αυτά αφορούσαν την περιγραφή περιεχομένου ιστοσελίδων όπως οι επικεφαλίδες, οι παράγραφοι και οι λίστες. Ο αριθμός έκδοσης της HTML μεγάλωνε καθώς η γλώσσα εξελισσόταν με την εισαγωγή νέων στοιχείων και προσαρμογών στους κανόνες της. Η τελευταία έκδοση είναι η HTML5.

Η HTML5 αποτελεί φυσική εξέλιξη των προηγούμενων εκδόσεων της HTML και προσπαθεί να καλύψει τις ανάγκες των υπάρχοντων και των μελλοντικών ιστότοπων. Κληρονομεί τη συντριπτική πλειονότητα των λειτουργιών από τους προκατόχους της, γεγονός που σημαίνει ότι αν κάποιος έχει γράψει κώδικα HTML πριν από την έλευση της HTML5, γνωρίζει ήδη αρκετά για την HTML5. Αυτό σημαίνει επίσης ότι μεγάλο μέρος της HTML5 λειτουργεί τόσο σε παλιούς όσο και σε νέους φυλλομετρητές. Η προς τα πίσω συμβατότητα αποτελεί βασική αρχή στον σχεδιασμό της HTML5.

Η HTML (αρχικοποίηση του αγγλικού **HyperText Markup Language**, ελλ. Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων. Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.



Ο σκοπός ενός web browser είναι να διαβάζει τα έγγραφα HTML και τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας.

Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους του ιστότοπους. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, ενώ μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Οι Web browsers μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου. (Βικιπαίδεια, 2015)

1.4.2 CSS3

Ενώ η HTML καθορίζει το νόημα του περιεχομένου και προσδίδει στις ιστοσελίδες τη βασική τους δομή, τα επάλληλα φύλλα στυλ (Cascading Style Sheets, CSS) καθορίζουν την εμφάνιση.

Το φύλλο στυλ (style sheet) είναι απλώς ένα αρχείο κειμένου το οποίο περιλαμβάνει έναν ή περισσότερους κανόνες που προσδιορίζουν – μέσω ιδιοτήτων (properties) και τιμών (values) – με ποιον τρόπο θα πρέπει να απεικονιστούν τα συγκεκριμένα στοιχεία των ιστοσελίδων.

Υπάρχουν ιδιότητες CSS για τον έλεγχο της βασικής μορφοποίησης (όπως είναι το μέγεθος της γραμματοσειράς και το χρώμα), των ιδιοτήτων της διάταξης της σελίδας (όπως είναι η θέση και η ροή του κειμένου) και του χειρισμού της εκτύπωσης (όπως είναι ο καθορισμός του σημείου αλλαγής σελίδας όταν την τυπώνει ο χρήστης). Η τεχνολογία CSS διαθέτει επίσης πολλές δυναμικές ιδιότητες που επιτρέπουν σε αντικείμενα να εμφανίζονται και να εξαφανίζονται, και οι οποίες είναι χρήσιμες για τη δημιουργία πτυσσόμενων μενού και άλλων αλληλεπιδραστικών στοιχείων.

Αυτός ο διαχωρισμός της μορφοποίησης και περιεχομένου δίνει τη δυνατότητα να παρουσιαστεί η ίδια σελίδα σήμανσης σε διαφορετικές μορφές για την απόδοση διαφορετικών μεθόδων, όπως στην οθόνη, σε έντυπη μορφή, από τη φωνή (όταν διαβάζεται από μια μηχανή αναζήτησης που βασίζεται σε ομιλία ή αναγνώστη οθόνης) και στις συσκευές που βασίζονται σε Braille. Μπορεί επίσης να χρησιμοποιηθεί για να εμφανίσει την ιστοσελίδα διαφορετικά, ανάλογα με το μέγεθος της οθόνης ή η συσκευή στην οποία προβάλλεται το αντικείμενο. Αν και ο συντάκτης μιας ιστοσελίδας, συνήθως,



συνδέει ένα CSS αρχείο μέσα στο αρχείο σήμανσης, οι αναγνώστες για να καθορίσουν ένα διαφορετικό φύλλο στυλ, όπως ένα CSS αρχείο που είναι αποθηκευμένο στο δικό τους υπολογιστή, πρέπει να παρακάμψουν αυτό που έχει καθορίσει ο συγγραφέας. Αν ο συγγραφέας ή ο αναγνώστης δεν συνδέσουν το έγγραφο σε ένα φύλλο στυλ, θα εφαρμόζεται το προεπιλεγμένο στυλ του προγράμματος περιήγησης. Ένα άλλο πλεονέκτημα των CSS είναι ότι οι αισθητικές αλλαγές στο γραφικό σχεδιασμό ενός εγγράφου (ή εκατοντάδες έγγραφα) μπορούν να εφαρμοστούν γρήγορα και εύκολα, τροποποιώντας μερικές γραμμές σε ένα αρχείο, και όχι από μια διαδικασία επίπονη (και κατά συνέπεια ακριβή) και όχι πάνω από το έγγραφο γραμμή προς γραμμή για να αλλάξουν την σήμανση. (Βικιπαίδεια, 2015)

1.4.3 JAVASCRIPT

Η JavaScript (JS) είναι διερμηνευμένη γλώσσα προγραμματισμού για ηλεκτρονικούς υπολογιστές. Αρχικά αποτέλεσε μέρος της υλοποίησης των φυλλομετρητών Ιστού, ώστε τα σενάρια από την πλευρά του πελάτη (client-side scripts) να μπορούν να επικοινωνούν με τον χρήστη, να ανταλλάσσουν δεδομένα ασύγχρονα και να αλλάζουν δυναμικά το περιεχόμενο του εγγράφου που εμφανίζεται.



Η JavaScript είναι μια γλώσσα σεναρίων που βασίζεται στα πρωτότυπα (prototype-based), είναι δυναμική, με ασθενείς τύπους και έχει συναρτήσεις ως αντικείμενα πρώτης τάξης. Η σύνταξή της είναι επηρεασμένη από τη C. Η JavaScript αντιγράφει πολλά ονόματα και συμβάσεις ονοματοδοσίας από τη Java, αλλά γενικά οι δύο αυτές γλώσσες δε σχετίζονται και έχουν πολύ διαφορετική σημασιολογία. Οι βασικές αρχές σχεδιασμού της JavaScript προέρχονται από τις γλώσσες προγραμματισμού Self και Scheme. Είναι γλώσσα βασισμένη σε διαφορετικά προγραμματιστικά παραδείγματα (multi-paradigm), υποστηρίζοντας αντικειμενοστρεφές και συναρτησιακό στυλ προγραμματισμού.

Η JavaScript χρησιμοποιείται και σε εφαρμογές εκτός ιστοσελίδων — τέτοια παραδείγματα είναι τα έγγραφα PDF, οι εξειδικευμένοι φυλλομετρητές (site-specific browsers) και οι μικρές εφαρμογές της επιφάνειας εργασίας (desktop widgets). Οι νεότερες εικονικές μηχανές και πλαίσια ανάπτυξης για JavaScript (όπως το Node.js) έχουν επίσης κάνει τη JavaScript πιο δημοφιλή για την ανάπτυξη εφαρμογών Ιστού στην πλευρά του διακομιστή (server-side).

Η JavaScript έχει γίνει μία από τις πιο δημοφιλείς γλώσσες προγραμματισμού ηλεκτρονικών υπολογιστών στον Παγκόσμιο Ιστό (Web). Αρχικά, όμως, πολλοί επαγγελματίες προγραμματιστές υποτίμησαν τη γλώσσα διότι το κοινό της ήταν ερασιτέχνες συγγραφείς ιστοσελίδων και όχι επαγγελματίες προγραμματιστές (και μεταξύ άλλων λόγων). Με τη χρήση της τεχνολογίας Ajax, η JavaScript γλώσσα επέστρεψε στο προσκήνιο και έφερε πιο επαγγελματική προσοχή προγραμματισμού. Το αποτέλεσμα ήταν ένα καινοτόμο αντίκτυπο στην εξάπλωση των πλαισίων και των βιβλιοθηκών, τη βελτίωση προγραμματισμού με JavaScript, καθώς και αυξημένη χρήση της JavaScript έξω από τα προγράμματα περιήγησης στο Web.

Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού). (Βικιπαίδεια,2015)

1.4.4 PHP

Η PHP είναι μια γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML.



Ένα αρχείο με κώδικα PHP θα πρέπει να έχει την κατάλληλη επέκταση (π.χ. *.php, *.php4, *.phtml κ.ά.). Η ενσωμάτωση κώδικα σε ένα αρχείο επέκτασης .html δεν θα λειτουργήσει και θα εμφανίσει στον browser τον κώδικα χωρίς καμία επεξεργασία, εκτός αν έχει γίνει η κατάλληλη ρύθμιση στα MIME types του server. Επίσης ακόμη κι όταν ένα αρχείο έχει την επέκταση .php, θα πρέπει ο server να είναι ρυθμισμένος για να επεξεργάζεται και να μεταγλωττίζει τον κώδικα PHP σε HTML που καταλαβαίνει το πρόγραμμα πελάτη. Ο διακομιστής Apache, που χρησιμοποιείται σήμερα ευρέως σε συστήματα με τα λειτουργικά συστήματα GNU/Linux, Microsoft Windows, Mac OS X υποστηρίζει εξ ορισμού την εκτέλεση κώδικα PHP, είτε με την χρήση ενός πρόσθετου (mod_php) ή με την αποστολή του κώδικα προς εκτέλεση σε εξωτερική διεργασία CGI ή FCGI ή με την έλευση της php5.4 υποστηρίζεται η εκτέλεση σε πολυάσχολους ιστοχώρους, FastCGI Process Manager (FPM).

Ο συνδυασμός Linux/Apache/MySQL/PHP, που είναι η πιο δημοφιλής πλατφόρμα εκτέλεσης ιστοσελίδων είναι γνωστός και με το ακρωνύμιο LAMP. Παρόμοια, ο συνδυασμός */Apache/MySQL/PHP ονομάζεται *AMP, όπου το πρώτο αρχικό αντιστοιχεί στην πλατφόρμα, στην οποία εγκαθίστανται ο Apache, η MySQL και η PHP (π.χ. Windows, Mac OS X).

Η ιστορία της PHP ξεκινά από το 1994, όταν ένας φοιτητής, ο Rasmus Lerdorf δημιούργησε χρησιμοποιώντας τη γλώσσα προγραμματισμού Perl ένα απλό script με όνομα php.cgi, για προσωπική χρήση. Το script αυτό είχε σαν σκοπό να διατηρεί μια λίστα στατιστικών για τα άτομα που έβλεπαν το online βιογραφικό του σημείωμα. Αργότερα αυτό το script το διέθεσε και σε φίλους του, οι οποίοι άρχισαν να του ζητούν να προσθέσει περισσότερες δυνατότητες. Η γλώσσα τότε ονομαζόταν PHP/FI από τα αρχικά Personal Home Page/Form Interpreter. Το 1997 η PHP/FI έφθασε στην έκδοση 2.0, βασισμένη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερους από 50.000 ιστότοπους που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασισμένοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend

(από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και οι πρώτες δοκιμαστικές εκδόσεις της επερχόμενης PHP 6, για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει. Οι περισσότεροι ιστότοποι επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP. (Βικιπαίδεια, 2015)

1.4.5 SQL

Η SQL (αγγλ. αρκτ. από το Structured Query Language) είναι μία γλώσσα υπολογιστών στις βάσεις δεδομένων, που σχεδιάστηκε για τη διαχείριση δεδομένων, σε ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System, RDBMS) και η οποία, αρχικά, βασίστηκε στη σχεσιακή άλγεβρα. Η γλώσσα περιλαμβάνει δυνατότητες ανάκτησης και ενημέρωσης δεδομένων, δημιουργίας και τροποποίησης σχημάτων και σχεσιακών πινάκων, αλλά και ελέγχου πρόσβασης στα δεδομένα. Η SQL ήταν μία από τις πρώτες γλώσσες για το σχεσιακό μοντέλο του Edgar F. Codd, στο σημαντικό άρθρο του το 1970, και έγινε η πιο ευρέως χρησιμοποιούμενη γλώσσα για τις σχεσιακές βάσεις δεδομένων.



Η SQL αναπτύχθηκε στην IBM από τους Andrew Richardson, Donald C. Messerly και Raymond F. Boyce, στις αρχές της δεκαετίας του 1970. Αυτή η έκδοση, αποκαλούμενη αρχικά SEQUEL, είχε ως σκοπό να χειριστεί και να ανακτήσει τα στοιχεία που αποθηκεύτηκαν στο πρώτο RDBMS της IBM, το System R.

Το πρώτο σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) ήταν το RDMBS που αναπτύχθηκε στο MIT, στις αρχές της δεκαετίας του 1970 και η Ingres, που αναπτύχθηκε το 1974 στο Πανεπιστήμιο Μπέρκλεϋ. Η Ingres εφάρμοσε μία γλώσσα διατύπωσης ερωτήσεων γνωστή ως QUEL, το οποίο αντικαταστάθηκε αργότερα στην αγορά από την SQL. Προς το τέλος της δεκαετίας του 70 η Relational Software (τώρα Oracle Corporation) είδε τη δυνατότητα αυτών που περιγράφηκαν από Codd, Chamberlin, και Boyce και ανέπτυξε την SQL βασισμένο στο RDBMS, με τις φιλοδοξίες πώλησης του στο Αμερικανικό ναυτικό, την Κεντρική Υπηρεσία Πληροφοριών και άλλες Αμερικανικές Υπηρεσίες. Το καλοκαίρι του 1979, η Relational Software εισήγαγε την πρώτη διαθέσιμη στο εμπόριο εφαρμογή του SQL και νίκησε την IBM με τη διάθεση του πρώτου εμπορικού RDBMS για μερικές εβδομάδες. (Βικιπαίδεια, 2014)

ΚΕΦΑΛΑΙΟ 2

2.1 ΠΡΟΓΡΑΜΜΑΤΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΗΘΗΚΑΝ

2.1.1 XAMPP

Το XAMPP είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού, λογισμικού ανοικτού κώδικα και ανεξαρτήτου πλατφόρμας το οποίο περιέχει το εξυπηρετητή ιστοσελίδων http Apache, την βάση δεδομένων MySQL και ένα διερμηνέα για κώδικα γραμμένο σε γλώσσες προγραμματισμού PHP και Perl.

Το XAMPP είναι ένα ελεύθερο λογισμικό το οποίο περιέχει ένα εξυπηρετητή ιστοσελίδων το οποίο μπορεί να εξυπηρετεί και δυναμικές ιστοσελίδες τεχνολογίας PHP/MySQL. Είναι ανεξάρτητο πλατφόρμας και τρέχει σε Microsoft Windows, Linux, Solaris, and Mac OS X και χρησιμοποιείται ως πλατφόρμα για την σχεδίαση και ανάπτυξη ιστοσελίδων με την τεχνολογίες όπως PHP, JSP και Servlets. Παρακάτω θα δούμε την εγκατάστασή του βήμα – βήμα. (Βικιπαίδεια, 2015)



Εγκατάσταση του XAMPP στα Windows

Αρχικά ζητείται η επιλογή γλώσσας. Η προεπιλεγμένη επιλογή είναι Αγγλικά (English). Αφού επιλεγθεί η γλώσσα επιλέγεται το κουμπί OK και στη συνέχεια εμφανίζεται στην οθόνη ένα μήνυμα καλωσορίσματος του προγράμματος εγκατάστασης, όπως φαίνεται στην Εικόνα 2.1:



Εικόνα 2.1 Κύρια οθόνη εγκατάστασης του XAMPP

Κάνοντας κλικ στο κουμπί Next συνεχίζεται η διεργασία εγκατάστασης. Όπως συμβαίνει με τις περισσότερες εγκαταστάσεις μέσω οδηγού, ζητείται να επιλεγθεί μια θέση

εγκατάστασης και ορισμένες επιλογές. Η εγκατάσταση του XAMPP δεν διαφέρει από τις άλλες εγκαταστάσεις. Αφήνοντας τις προεπιλεγμένες θέσεις εγκατάστασης και επιλογές, συνεχίζεται σταδιακά η διαδικασία της εγκατάστασης. Σε αυτό το σημείο, γίνεται η διεργασία εγκατάστασης, όπως φαίνεται στην Εικόνα 2.2.



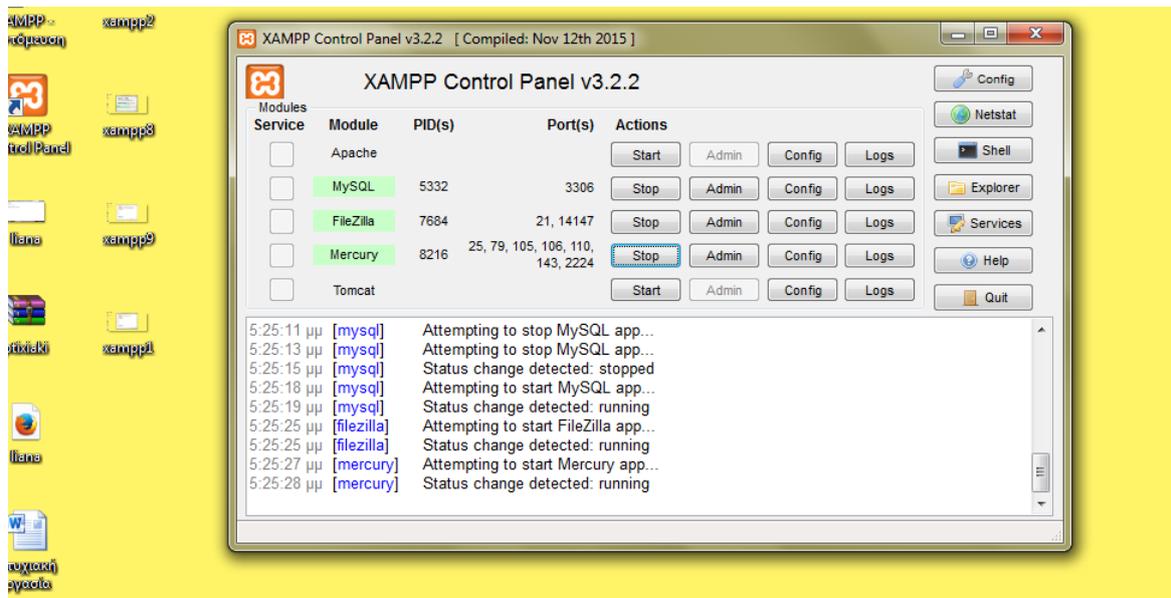
Εικόνα 2.2 Συνέχεια εγκατάστασης του XAMPP

Όταν τελειώσει η διεργασία εγκατάστασης, εμφανίζεται ειδοποίηση του προγράμματος εγκατάστασης. Κάνοντας κλικ στο κουμπί Finish ολοκληρώνεται η διεργασία. Πριν τερματίσει τελικά η διεργασία εγκατάστασης, τίθεται το ερώτημα εκκίνησης του πίνακα ελέγχου, Control Panel, ώστε να διαχειριστούν οι εγκατεστημένες υπηρεσίες, όπως φαίνεται στην Εικόνα 2.3.



Εικόνα 2.3 Ολοκλήρωση εγκατάστασης

Το XAMPP Control Panel, όπως φαίνεται στην Εικόνα 2.4, παρέχει προσπέλαση με ένα κλικ, για την εκκίνηση και τον τερματισμό των διεργασιών του Apache και του MySQL server, που εκτελούνται στον υπολογιστή. Αν εκτελούν αυτές οι διεργασίες διακομιστή σε έναν τοπικό υπολογιστή, μόνο για σκοπούς ανάπτυξης, ενεργοποιούνται μόνο όταν χρειάζονται. Το Control Panel παρέχει γρήγορη προσπέλαση για αυτήν την εργασία.



Εικόνα 2.4 Το XAMPP Control Panel

Για τον έλεγχο της εκτέλεσης του web διακομιστή, γίνεται είσοδος σε ένα πρόγραμμα πλοήγησης και εισαγωγή της διεύθυνσης **http://localhost/xampp.php**.

Αυτό ήταν όλο. Το XAMPP εγκατέστησε τον Apache, την PHP και την MySQL στον υπολογιστή και η εγκατάσταση των υπηρεσιών εμφανίζεται στην αριστερή στήλη, προβάλλοντας την <http://localhost/xampp>. Η εγκατάσταση των προγραμμάτων που περιέχονται στο XAMPP μπορεί να γίνει ξεχωριστά. Ο τρόπος θα περιγραφεί στην εγκατάσταση του Apache παρακάτω:

Εγκατάσταση του Apache στα Windows

Ο Apache HTTP γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Όποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache είναι ένας από τους δημοφιλέστερους εξυπηρετητές ιστού, εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation). Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής



συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL. (Βικιπαίδεια, 2015)

Πριν την εγκατάσταση του Apache θα πρέπει να διασφαλιστεί ότι δεν εκτελείται ένας Web server στο σύστημά (π.χ. μία προηγούμενη έκδοση του Apache, τον Microsoft Internet Information Server ή τον Microsoft Personal Web Server). Καλό θα είναι να καταργηθεί η εγκατάσταση των υπαρχόντων servers ή η απενεργοποίησή τους με οποιονδήποτε άλλο τρόπο. Μπορούν να εκτελούνται πολλαπλοί Web servers, αλλά σε διαφορετικούς συνδυασμούς διευθύνσεων και θυρών.

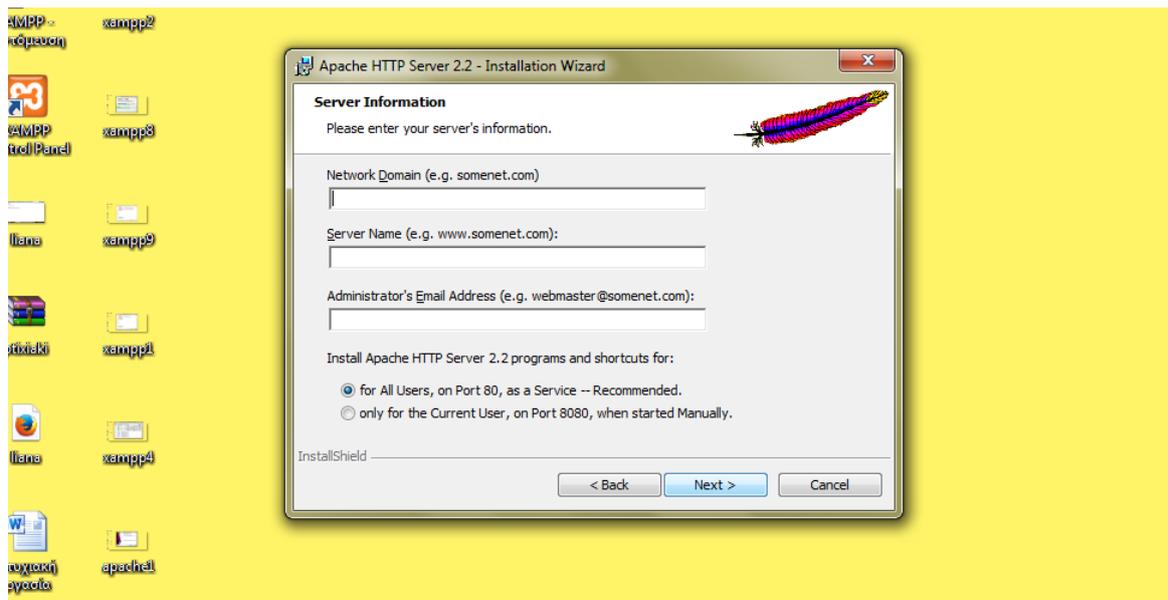
Πριν να ξεκινήσει η εγκατάσταση, γίνεται αναζήτηση της σύνδεσης που φέρει τον τίτλο Win32 Binary (MSI Installer). Αφού κατέβει το αρχείο εγκατάστασης, κάνοντας διπλό κλικ πάνω του θα ξεκινήσει ο οδηγός εγκατάστασης (Installation Wizard). Θα εμφανιστεί μία οθόνη καλωσορίσματος, η οποία παρουσιάζεται στην Εικόνα 2.5.

Κάνοντας κλικ στο Next θα συνεχιστεί η διαδικασία εγκατάστασης. Θα ζητηθεί η αποδοχή της συμφωνία άδειας χρήσης του Apache. Ουσιαστικά, η συμφωνία άδειας χρήσης αναφέρει ότι μπορούν να γίνουν προσαρμογές στο λογισμικό. Θα πρέπει να διαβαστεί η άδεια χρήσης για να κατανοηθούν πλήρως οι όροι της.



Εικόνα 2.5 Η οθόνη καλωσορίσματος του οδηγού εγκατάστασης του Apache στα Windows

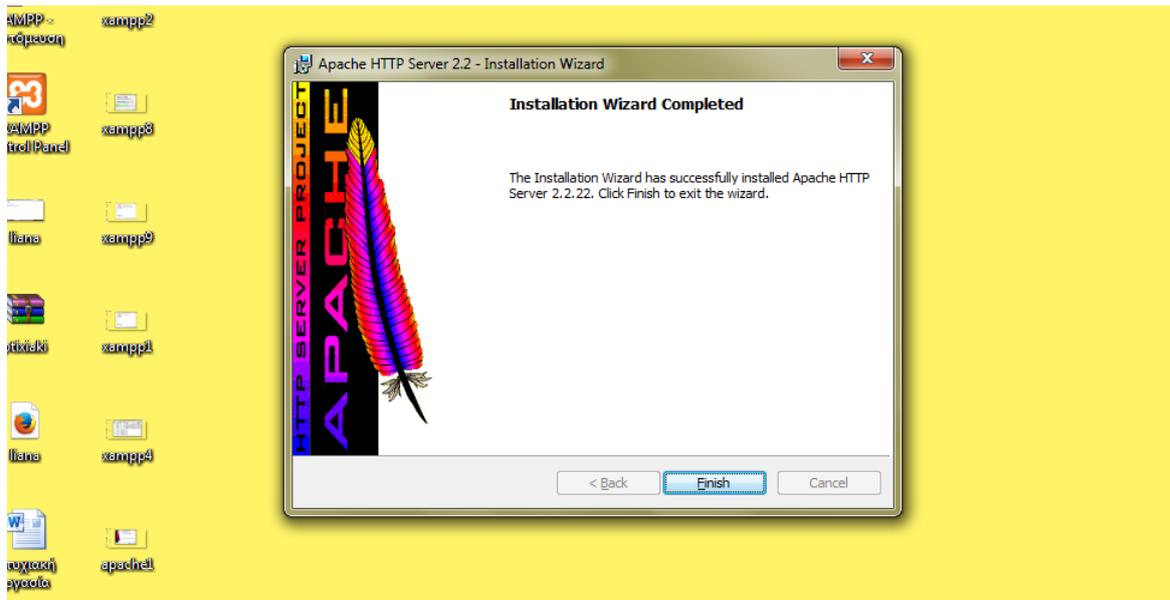
Αφού γίνει αποδοχή των όρων της συμφωνίας άδειας χρήσης, ο οδηγός παρουσιάζει μία σύντομη εισαγωγή στον Apache. Στη συνέχεια ζητείται η παροχή βασικών πληροφοριών του υπολογιστή, όπως στην εικόνα 2.6. Σ'αυτές περιλαμβάνονται η πλήρης διεύθυνση δικτύου για τον server – το όνομα domain και όνομα του server (π.χ. mycomputer.mydomain.com) – καθώς και η διεύθυνση ηλεκτρονικού ταχυδρομείου του επόπτη (administrator) του server. Το όνομα του server θα είναι το όνομα που θα χρησιμοποιούν οι client υπολογιστές ώστε να προσπελάσουν τον server και η διεύθυνση ηλεκτρονικού ταχυδρομείου του επόπτη θα προστίθεται στα μηνύματα σφάλματος, έτσι ώστε οι επισκέπτες να ξέρουν πώς να επικοινωνήσουν όταν προκύπτουν προβλήματα.



Εικόνα 2.6 Η οθόνη εισαγωγής των βασικών πληροφοριών

Στην ίδια οθόνη μπορεί να επιλεγεί αν ο Apache θα εγκατασταθεί ως μία υπηρεσία (service) στο σύστημα ή αν θα απαιτείται η "χειροκίνητη" εκκίνησή του. Η εγκατάσταση του Apache ως μία υπηρεσία έχει ως αποτέλεσμα ο server να εκκινεί κάθε φορά που εκκινούν τα Windows και μπορεί να γίνει έλεγχος με τα γνωστά εργαλεία διαχείρισης υπηρεσιών, που παρέχουν τα ίδια τα Windows. Επιλέγεται αυτή η μέθοδος εάν ο Apache θα εκτελείται σε ένα περιβάλλον παραγωγής ή σε οποιοδήποτε άλλο περιβάλλον, το οποίο απαιτεί τη συνεχή λειτουργία του server. Εάν γίνει εγκατάσταση του Apache για τον τρέχοντα χρήστη θα πρέπει να ακολουθηθεί η "χειροκίνητη" διαδικασία εκκίνησης και να ορίζεται ως προεπιλεγμένη θύρα (default port), την οποία ακροάζεται ο Apache για αιτήσεις, η 8080. Επιλέγεται αυτή η μέθοδος εάν χρησιμοποιείται ο Apache για σκοπούς δοκιμών ή αν υπάρχει ήδη ένας Web server, ο οποίος εκτελείται στη θύρα 80 και επιλέγεται το κουμπί Next. Εάν το σύστημά δεν έχει πλήρη διεύθυνση δικτύου, χρησιμοποιείται το όνομα localhost ή η διεύθυνση IP 127.0.0.1 σαν όνομα server (στο πεδίο Server Name).

Η επόμενη οθόνη επιτρέπει την επιλογή του τύπου της εγκατάστασης. Με την τυπική (Typical) εγκατάσταση εγκαθίστανται τα δυαδικά αρχεία του Apache και τα αρχεία πληροφοριών τεκμηρίωσης, αλλά δεν εγκαθίστανται τα header αρχεία και οι βιβλιοθήκες. Η εξειδικευμένη (Custom) εγκατάσταση επιτρέπει την επιλογή της εγκατάστασης των header αρχείων ή των αρχείων πληροφοριών τεκμηρίωσης. Αφού επιλεγεί ο κατάλογος εγκατάστασης, το πρόγραμμα θα προχωρήσει στην καθαυτό διαδικασία εγκατάστασης των αρχείων. Εάν όλα πάνε καλά, θα παρουσιάσει την τελική οθόνη που φαίνεται στην Εικόνα 2.7.



Εικόνα 2.7 Επιτυχής ολοκλήρωση της εγκατάστασης

Εγκατάσταση της PHP στα Windows

Με παρόμοιο τρόπο γίνεται και η εγκατάσταση της PHP και για τον έλεγχο της εγκατάστασης γίνεται άνοιγμα ενός προγράμματος πλοήγησης και εισαγωγή του **http://localhost/phpinfo.php**. Το αποτέλεσμα θα πρέπει να είναι παρόμοιο με αυτό της Εικόνας 2.8:

PHP Version 5.6.15 	
System	
Build Date	Oct 29 2015 12:34:05
Compiler	MSVC11 (Visual C++ 2012)
Architecture	x86
Configure Command	cscrip /nologo configure.js "--enable-snapshot-build"--disable-isapi"--enable-debug-pack"--without-mssql"--without-pdo-mssql"--without-pi3web"--with-pdo-oci=c:\php-sdk\oraclew86\instantclient_12_1\sdk\shared"--with-oci8-12c=c:\php-sdk\oraclew86\instantclient_12_1\sdk\shared"--enable-object-out-dir=.obj"--enable-com-dotnet=shared"--with-mcrypt=static"--without-analyzer"--with-pgo"
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\Windows
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,TS,VC11
PHP Extension Build	API20131226,TS,VC11
Debug Build	no
Thread Safety	enabled

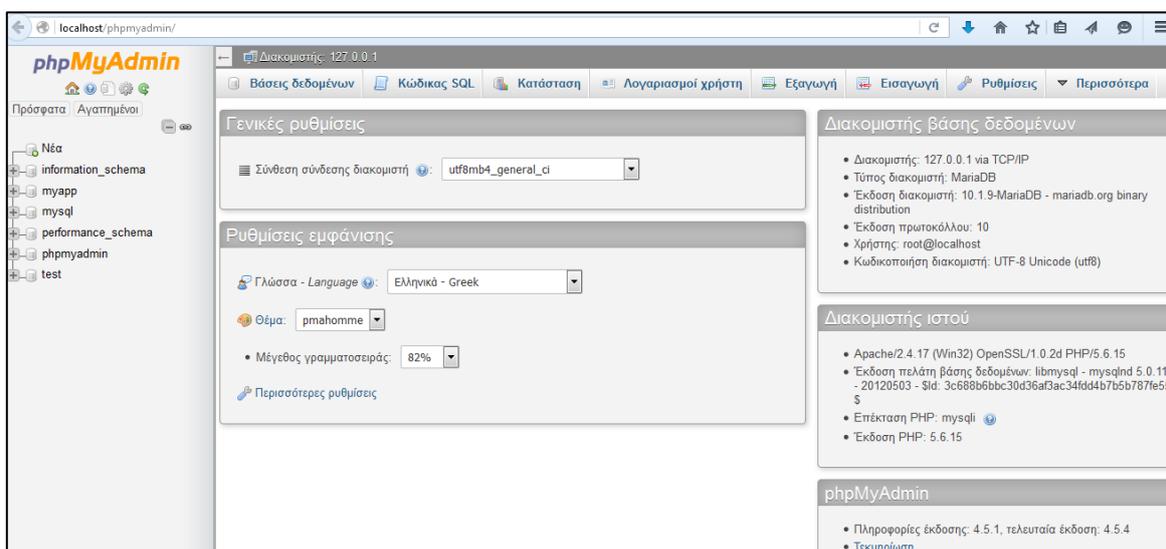
Εικόνα 2.8 Επιτυχής εγκατάσταση της PHP

phpMyAdmin



Το phpMyAdmin είναι ένα ελεύθερο, ανοιχτού κώδικα εργαλείο γραμμένο σε PHP το οποίο χρησιμοποιείται για την διαχείριση της MySQL βάσης δεδομένων μέσω φυλλομετρητή. Μπορεί να εκτελέσει διάφορες ενέργειες όπως την δημιουργία, την επεξεργασία ή την διαγραφή βάσεων δεδομένων, πινάκων, πεδίων ή γραμμών, την εκτέλεση SQL ερωτημάτων, την διαχείριση των χρηστών καθώς και των δικαιωμάτων τους. Για να εμφανιστεί η phpMyAdmin γίνεται εισαγωγή του: **http://localhost/phpmyadmin/** σε ένα πρόγραμμα περιήγησης ή πηγαίνοντας στο XAMPP Control Panel και επιλέγοντας Admin στη MySQL:

Θα πρέπει να εμφανιστεί η οθόνη της εικόνας 2.9:



Εικόνα 2.9 Αρχική οθόνη της phpMyAdmin

2.1.2 Notepad++



Το Notepad++ είναι ένας εύχρηστος αλλά πλούσιος σε λειτουργίες επεξεργαστής κειμένου, με δυνατότητες όπως άνοιγμα πολλών αρχείων σε καρτέλες, αυτόματος χρωματισμός κώδικα για όλες τις γλώσσες προγραμματισμού και πολλά ακόμα χαρακτηριστικά.

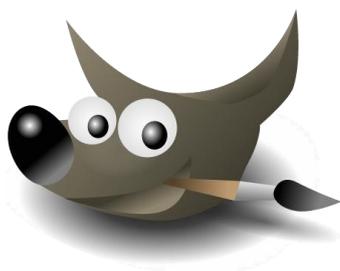
Στην εικόνα 2.10 παρουσιάζεται κομμάτι κώδικα ιστοσελίδας :

```
36 <p>Τα στοιχεία HTML γράφονται με μια <strong>ετικέτα έναρξης</strong>, μια <strong>ετικέτα τέλους
37 τους:</p>
38 <p><strong>&lt;tagname&gt;</strong>περιεχόμενο<strong>&lt;/tagname&gt;</strong></p>
39 <p><strong>&lt;p&gt;</strong>Η πρώτη μου παράγραφος<strong>&lt;/p&gt;</strong></p>
40
41 <table class="version">
42   <tr class="dif">
43     <th scope="col">Ετικέτα αρχής</th>
44     <th scope="col">Περιεχόμενο στοιχείου</th>
45     <th scope="col">Ετικέτα τέλους</th>
46   </tr>
47   <tr>
48     <td&lt;h1&gt;</td>
49     <td>Η πρώτη μου επικεφαλίδα</td>
50     <td&lt;/h1&gt;</td>
51   </tr>
52   <tr class="dif">
53     <td&lt;p&gt;</td>
54     <td>Η πρώτη μου παράγραφος</td>
55     <td&lt;/p&gt;</td>
```

Εικόνα 2.10 Γραμμές κώδικα στο Notepad++

2.1.3 GIMP 2.8

Το Gimp (GNU Image Manipulation Program) είναι ένα δωρεάν και ελεύθερο λογισμικό επεξεργασίας γραφικών τύπου raster. Είναι εργαλείο που επικεντρώνεται κυρίως στη διαμόρφωση και την επεξεργασία εικόνας και είναι ελεύθερα διαθέσιμο σε εκδόσεις προσαρμοσμένες για τα πιο δημοφιλή λειτουργικά συστήματα όπως τα Microsoft Windows, το Mac OS X, και το GNU/Linux.



Εκτός από τη λεπτομερή επιδιόρθωση εικόνων και τη σχεδίαση ελεύθερης μορφής, το GIMP μπορεί να φέρει εις πέρας και άλλες βασικές εργασίες επεξεργασίας εικόνας, όπως η αλλαγή μεγέθους, η

επεξεργασία και η «καλλιέργεια» φωτογραφιών, το φωτομοντάζ συνδυάζοντας πολλαπλές εικόνες και η μετατροπή μεταξύ διαφορετικών μορφών εικόνας. Επίσης μπορεί να χρησιμοποιηθεί για να δημιουργήσει κινούμενες εικόνες σε πολλές μορφές, όπως GIF και MPEG μέσω του Animation plug-in.

Το όραμα του προϊόντος είναι ότι το GIMP είναι ένα δωρεάν λογισμικό για απαιτητικούς χρήστες που βασίζεται στην εφαρμογή γραφικών, στην επεξεργασία και δημιουργία πρωτότυπων εικόνων, στα γραφικά στοιχεία των ιστοσελίδων και στην τέχνη μετατροπής των στοιχείων μιας διεπαφής που έχει ως επίκεντρο τον χρήστη. (Βικιπαίδεια, 2015)



Εικόνα 2.11 Οθόνη εκκίνησης του GIMP 2.8

ΚΕΦΑΛΑΙΟ 3

3.1 Η ΒΑΣΙΚΗ ΔΟΜΗ ΤΗΣ HTML5

Η HTML έχει τρία κύρια συστατικά σήμανσης: τα **στοιχεία** (elements), τις **ιδιότητες** ή **χαρακτηριστικά** (attributes) και τις **τιμές** (values). Τα *στοιχεία* μοιάζουν με μικρές ετικέτες που περιγράφουν τα διάφορα μέρη μιας ιστοσελίδας. Τα στοιχεία μπορεί να περιέχουν κείμενο και άλλα στοιχεία, ή να είναι κενά. Οι *ιδιότητες* περιλαμβάνουν πληροφορίες σχετικά με το περιεχόμενο του εγγράφου, σε αντίθεση με το ίδιο το περιεχόμενο.

Κάποια στοιχεία έχουν μία ή περισσότερες ιδιότητες, που περιγράφουν περαιτέρω τον σκοπό και το περιεχόμενο (αν υπάρχει) του στοιχείου. Κάποιες ιδιότητες μπορούν να δεχτούν οποιαδήποτε *τιμή*, ενώ άλλες είναι πιο περιορισμένες. Για παράδειγμα, οι ιδιότητες **href** και **src** που αναφέρονται σε άλλα αρχεία θα πρέπει να περιέχουν τιμές με τη μορφή διευθύνσεων URL (Uniform Resource Locator, Ενιαίος Εντοπιστής Πόρων) , που είναι μια μοναδική διεύθυνση αρχείου στον Ιστό.

Οι σελίδες HTML χωρίζονται σε δύο ενότητες: την κεφαλίδα (**head**) και το σώμα (**body**). Η ενότητα **head** του εγγράφου είναι το σημείο όπου ορίζεται ο τίτλος της σελίδας, συμπεριλαμβάνονται πληροφορίες σχετικά με τη σελίδα για τις μηχανές αναζήτησης όπως το Google, φορτώνονται τα φύλλα στυλ και περιστασιακά φορτώνονται αρχεία JavaScript (αν και για λόγους επιδόσεων τις περισσότερες φορές τα αρχεία JavaScript είναι καλύτερο να φορτώνονται ακριβώς πριν από την τελική ετικέτα `</body>` στο τέλος της σελίδας. Με εξαίρεση την ετικέτα `title`, το περιεχόμενο της ενότητας `head` δεν είναι ορατό στους χρήστες που επισκέπτονται την ιστοσελίδα.

Το στοιχείο **body** περικλείει το περιεχόμενο της σελίδας, στο οποίο συμπεριλαμβάνεται το κείμενο, οι εικόνες, οι φόρμες, τα αρχεία ήχου και βίντεο και το όποιο άλλο αλληλεπιδραστικό περιεχόμενο. Με άλλα λόγια, αυτά είναι ό,τι βλέπουν οι επισκέπτες της σελίδας. (Castro & Hyslop, 2012)

Παρακάτω παρουσιάζονται τα βασικά στοιχεία των ενότητων `head` και `body`, καθώς και η ερμηνεία τους:

```
<!DOCTYPE html >
```

```

< html lang = "el" >

< head >

< meta charset = " UTF – 8 " / >

< title > < / title >

< / head >

< body >

< / body >

< / html >

```

- **< ! DOCTYPE html >** : δηλώνει τη σελίδα ως έγγραφο HTML5.
- **<html lang = “ κωδικός – γλώσσας “ >** : ο κωδικός γλώσσας είναι ο κωδικός που αντιστοιχεί στην προεπιλεγμένη γλώσσα του περιεχομένου της σελίδας. Για παράδειγμα, ο κωδικός < html = “el “ > είναι για Ελληνικά, ο κωδικός < html = “ fr “ > είναι για Γαλλικά.
- **< head >** : δηλώνει την κεφαλίδα του εγγράφου για τη σελίδα.
- **< meta charset = “ UTF – 8 “ / >** : δηλώνει την κωδικοποίηση χαρακτήρων του εγγράφου ως UTF – 8. Το διάστημα και η κάθετος είναι προαιρετικά, έτσι η εντολή < meta charset = “ UTF – 8 “ > θα λειτουργήσει εξίσου καλά.
- **< title > < / title >** : περιέχει τον τίτλο της σελίδας.
- **< / head >** : τερματίζει την κεφαλίδα του εγγράφου για τη σελίδα
- **< body >** : ξεκινάει το σώμα της σελίδας, είναι το σημείο όπου θα μπει το περιεχόμενο
- **< / body >** : τερματίζει το σώμα
- **< / html >** : τερματίζει τη σελίδα

Αφού αναλύθηκαν τα στοιχεία της σελίδας και της ενότητας head, θα παρουσιαστούν τα στοιχεία της ενότητας body τα οποία περικλείουν το περιεχόμενο της σελίδας που προβάλλεται στον επισκέπτη.

Η συνηθισμένη *διάταξη* μιας σελίδας αποτελείται από τέσσερα βασικά συστατικά στοιχεία: μια κύρια **κεφαλίδα** με στοιχεία πλοήγησης, ένα **άρθρο** στην κύρια περιοχή περιεχομένου, ένα πλευρικό **πλαίσιο** με εφαπτόμενες πληροφορίες και ένα **υποσέλιδο**.

Η διάταξη αυτή παρουσιάζεται παρακάτω:

Κεφαλίδα με στοιχεία πλοήγησης	
Κύριο περιεχόμενο σελίδας	Σχετιζόμενο, εφαπτόμενο κείμενο
Υποσέλιδο	

Πίνακας 3.1 Δομή HTML σελίδας

Ο HTML κώδικας της σελίδας που φαίνεται παραπάνω θα είναι:

.....

< body >

< header >

< nav >

< / nav >

< / header >

< article >

< h1 > < / h1 >

< section >

< h2 > < / h2 >

< / section >

< section >

< h2 > < /h2 >

< / section >

< / article >

< aside >

< h1 > < / h1 >

< / aside >

< footer >

< p > < / p >

< / footer >

< / body >

< / html >

Τα HTML στοιχεία αναλύονται στη συνέχεια:

- **< header >** : χαρακτηρίζει μια ομάδα εισαγωγικού περιεχομένου ή περιεχομένου πλοήγησης. Η σελίδα μπορεί να έχει οποιοδήποτε πλήθος στοιχείων header και η σημασία τους μπορεί να ποικίλλει ανάλογα με το θεματικό πλαίσιο. Συχνά η κεφαλίδα της σελίδας περιέχει το λογότυπο της τοποθεσίας, τα κύρια στοιχεία πλοήγησης, άλλους καθολικούς συνδέσμους, και ακόμα και ένα πλαίσιο αναζήτησης.
- **< article >** : αναπαριστάνει μια αυτοτελή σύνθεση σε ένα έγγραφο, σελίδα, εφαρμογή ή τοποθεσία και είναι αυτόνομο διανεμόμενο ή επαναχρησιμοποιήσιμο, π.χ. σε συγκεντρωτικές δημοσιεύσεις. Αυτές μπορεί να είναι δημοσίευση σε φόρουμ, άρθρο εφημερίδας ή περιοδικού, καταχώρηση ιστολογίου, σχόλιο κάποιου χρήστη, αλληλεπιδραστική μικροεφαρμογή ή μικροεργαλείο, ή οποιοδήποτε άλλο ανεξάρτητο στοιχείο περιεχομένου.
- **< aside >** : συνήθως ορίζει ένα πλευρικό πλαίσιο, όμως μπορεί να τοποθετηθεί σε διάφορα σημεία της σελίδας, ανάλογα με το θεματικό πλαίσιο. Μπορεί να είναι ένα πλαίσιο μέσα στο ίδιο το κύριο περιεχόμενο, να βρίσκεται στην ίδια στήλη αλλά να μην είναι ένθετο στο κύριο περιεχόμενο, ή να βρίσκεται σε μια δευτερεύουσα στήλη ως πλευρικό πλαίσιο. Παραδείγματα στοιχείου aside είναι ένα απόφθεγμα, ένα πλευρικό πλαίσιο, ένα πλαίσιο με συνδέσμους για συναφή

άρθρα σε τοποθεσίες ειδήσεων, διαφημίσεις, μια τροφοδοσία Twitter, ή μια λίστα συναφών προϊόντων σε μια εμπορική τοποθεσία.

- **< footer >** : αντιπροσωπεύει ένα υποσέλιδο για το πλησιέστερο στοιχείο article, aside, ή body στο οποίο είναι ένθετο. Θα είναι το υποσέλιδο για ολόκληρη τη σελίδα μόνο όταν ο πλησιέστερος πρόγονός του είναι το στοιχείο body. Αν το στοιχείο footer περικλείει όλο το περιεχόμενο της ενότητάς του (για παράδειγμα ένα στοιχείο article), τότε αντιπροσωπεύει το ισοδύναμο ενός παραρτήματος, ευρετηρίου, ή μακροσκελούς άδειας χρήσης, ανάλογα με το θεματικό πλαίσιο.

3.2 ΒΑΣΙΚΑ HTML ΣΤΟΙΧΕΙΑ

Επικεφαλίδες

Η HTML παρέχει έξι επίπεδα επικεφαλίδων για τη διαμόρφωση της ιεραρχίας πληροφοριών στις σελίδες. Η κάθε επικεφαλίδα χαρακτηρίζεται με κάποιο από τα στοιχεία **h1** – **h6**, όπου το h1 είναι η επικεφαλίδα κορυφαίου επιπέδου, το h2 είναι υποεπικεφαλίδα του h1, το h3 είναι υποεπικεφαλίδα του h2, κ.ο.κ. Οι επικεφαλίδες είναι από τα σημαντικότερα στοιχεία HTML σε οποιαδήποτε σελίδα.

Ενότητες

Το στοιχείο **section** αντιπροσωπεύει μια γενική ενότητα του εγγράφου ή της εφαρμογής. Σε αυτό το πλαίσιο, η ενότητα είναι μια θεματική ομαδοποίηση του περιεχομένου, συνήθως με μια επικεφαλίδα. Παραδείγματα ενότητων είναι τα κεφάλαια, οι διάφορες καρτέλες σελίδων σε πλαίσια διαλόγου με πολλές καρτέλες, ή οι αριθμημένες ενότητες μιας εργασίας. Η αρχική σελίδα ενός ιστότοπου θα μπορούσε να χωριστεί σε ενότητες για την εισαγωγή, τα στοιχεία ειδήσεων και τις πληροφορίες επικοινωνίας.

Σήμανση πλοήγησης

Το στοιχείο **nav** αναπαριστά ρητά μια ενότητα με κύριους συνδέσμους πλοήγησης. Οι σύνδεσμοι που υπάρχουν σε ένα στοιχείο nav μπορεί να δείχνουν σε περιεχόμενο της ίδιας σελίδας, σε άλλες σελίδες ή πόρους, ή και στα δύο αυτά. Όποια κι αν είναι η περίπτωση, το στοιχείο nav θα πρέπει να χρησιμοποιείται μόνο για τις σημαντικότερες ομάδες συνδέσμου του εγγράφου και όχι για όλες.

Γενικοί αποδέκτες

Κάποιες φορές, για την εφαρμογή κάποιας μορφοποίησης με φύλλα CSS ή κάποιου εφέ με JavaScript, κανένα από τα article, section, aside, nav ή τα άλλα στοιχεία δεν είναι σημασιολογικά κατάλληλο. Αυτό που χρειάζεται είναι ένας γενικός αποδέκτης, ο οποίος δεν έχει απολύτως καμία σημασιολογική ερμηνεία. Ο αποδέκτης αυτός είναι το στοιχείο **div** (“υποδιαίρεση”, division). Αφού τοποθετηθεί το στοιχείο div, μπορεί να εφαρμοστεί το επιθυμητό στυλ ή εφέ JavaScript σε αυτό.

Ονομασία στοιχείων με κλάση ή αναγνωριστικό

Τα HTML στοιχεία μπορούν να καθοριστούν με ένα μοναδικό *αναγνωριστικό*, να αντιστοιχιστούν σε μια συγκεκριμένη *κλάση* (ή κλάσεις), ή και τα δύο. Αφού γίνει αυτό,

μπορεί να εφαρμοστεί στυλ σε όλα τα στοιχεία με το συγκεκριμένο αναγνωριστικό (id) ή όνομα κλάσης (class).

Κάθε αναγνωριστικό (**id**) σε ένα έγγραφο HTML πρέπει να είναι μοναδικό. Με άλλα λόγια, δύο στοιχεία της ίδιας σελίδας δε μπορούν να έχουν το ίδιο id, και το κάθε στοιχείο θα πρέπει να έχει μόνο ένα id. Όμως το ίδιο id μπορεί να εμφανίζεται σε πολλές σελίδες και δεν είναι υποχρεωτικό να αντιστοιχίζεται κάθε φορά στο ίδιο στοιχείο, αν και αυτό γίνεται συνήθως. Αντιθέτως, το κάθε όνομα κλάσης (**class**) μπορεί να αντιστοιχιστεί σε οποιοδήποτε πλήθος στοιχείων σε μια σελίδα, και σε κάθε στοιχείο μπορούν να αντιστοιχιστούν περισσότερα από ένα ονόματα class.

Ρόλοι οροσήμου

Οι ρόλοι οροσήμου προσδιορίζουν ένα σύνολο περιοχών της σελίδας για να βελτιώσουν την προσπελασιμότητά τους:

role = " banner " : περιλαμβάνει συνήθως στοιχεία όπως το λογότυπο ή την ταυτότητα του χορηγού της τοποθεσίας

role = " navigation " : συλλογή αντικειμένων πλοήγησης για την πλοήγηση στο έγγραφο ή σε συναφή έγγραφα

role = " main " : το κύριο περιεχόμενο του εγγράφου

role = " complementary " : υποστηρικτική ενότητα του εγγράφου

3.3 ΠΕΡΙΣΣΟΤΕΡΑ HTML ΣΤΟΙΧΕΙΑ

Παρακάτω παρουσιάζονται τα στοιχεία που χρησιμοποιούνται συνήθως σε μια σελίδα HTML:

- ✚ < p > : ξεκίνημα νέας παραγράφου
- ✚ < address > : ορίζει τις πληροφορίες επικοινωνίας για τον συγγραφέα, τους ανθρώπους ή τον οργανισμό που σχετίζεται με μια σελίδα HTML, ή με ένα μέρος της σελίδας όπως με μια αναφορά ή με ένα άρθρο ειδήσεων
- ✚ < figure > : δημιουργία σχήματος
- ✚ < time > : καθορισμός ώρας
- ✚ < strong > : δηλώνει σημαντικό κείμενο
- ✚ < em > : εκφράζει έμφαση
- ✚ < cite > : καθορισμός παραπομπής ή αναφοράς
- ✚ < mark > : επισήμανση κειμένου
- ✚ < abbr > : επεξήγηση συντομογραφιών
- ✚ < dfn > : καθορισμός όρου
- ✚ < sup > : δημιουργία εκθέτη
- ✚ < sub > : δημιουργία δείκτη
- ✚ < code > : χαρακτηρισμός κώδικα

- ✚ < **pre** > : χρήση προμορφοποιημένου κειμένου
- ✚ < **small** > : καθορισμός ψιλών γραμμμάτων
- ✚ < **br** > : αλλαγή γραμμής
- ✚ < **span** > : το στοιχείο αυτό δεν έχει καμία σημασιολογική ερμηνεία και χρησιμοποιείται όταν κανένα άλλο στοιχείο δεν είναι κατάλληλο
- ✚ < **li** > : δημιουργία λίστας (χρησιμοποιούμε < **ol** > για διατεταγμένη λίστα και < **ul** > για μη διατεταγμένη λίστα)
- ✚ < **table** > : δημιουργία πίνακα (χρησιμοποιείται το < **tr** > για δημιουργία γραμμής και < **td** > για τη δημιουργία κελιού δεδομένων).

Εικόνες στην html

Το στοιχείο που χρησιμοποιείται για την εισαγωγή μίας εικόνας σε μια σελίδα είναι < **img src = " θέση εικόνας " / >** , όπου η θέση εικόνας δηλώνει τη θέση του αρχείου της εικόνας στον διακομιστή. Επίσης, προσθέτοντας την ιδιότητα **alt** ορίζεται το κείμενο που θα εμφανιστεί αν για κάποιο λόγο δεν εμφανιστεί η εικόνα. Οι ιδιότητες **width** και **height** ορίζουν το ύψος και το πλάτος της εικόνας αντίστοιχα.

Σύνδεσμοι στην html

Η δημιουργία συνδέσμου προς μια ιστοσελίδα γίνεται με το στοιχείο < **a href = " σελίδα. html" > < / a >** όπου σελίδα.html είναι η διεύθυνση URL της σελίδας προορισμού.

Φόρμες στην html

Οι φόρμες ορίζονται με το στοιχείο < **form** >. Το στοιχείο < **input** >, που είναι και το πιο σημαντικό στοιχείο της φόρμας, καθορίζει τον τύπο των δεδομένων εισόδου:

< **input type = " text " >** : ορίζει είσοδο κειμένου

< **input type = " radio " >** : ορίζει κουμπί μίας μόνο επιλογής (ανάμεσα σε πολλές)

< **input type = " submit " >** : ορίζει κουμπί υποβολής

< **input type = " password " >** : ορίζει ένα πεδίο κωδικού πρόσβασης

< **input type = " checkbox " >** : ορίζει κουμπί μηδέν ή περισσότερων επιλογών

(ανάμεσα σε πολλές)

< **input type = " button " >** : ορίζει ένα κουμπί

< **input type = " number " >** : χρησιμοποιείται για πεδία εισόδου που πρέπει να

περιέχουν μια αριθμητική τιμή

< input type = " date " > : χρησιμοποιείται για πεδία εισόδου που πρέπει να περιέχουν μια ημερομηνία

Κάποιες από τις *ιδιότητες* του στοιχείου < form > είναι οι ακόλουθες:

value : ορίζει την αρχική τιμή για ένα πεδίο εισαγωγής

size : ορίζει το μέγεθος (σε χαρακτήρες) για το πεδίο εισαγωγής

maxlength : ορίζει το μέγιστο επιτρεπόμενο μήκος για το πεδίο εισαγωγής

ΚΕΦΑΛΑΙΟ 4

4.1 CSS – Cascading Style Sheets

Ένα αρχείο φύλλου στυλ εφαρμόζει στυλ CSS σε πολλές ιστοσελίδες (ή ακόμα και σε έναν ολόκληρο ιστότοπο), σε μια μεμονωμένη σελίδα, ή σε ένα μεμονωμένο στοιχείο HTML. Αυτό επιτυγχάνεται με τρεις μεθόδους: τα **εξωτερικά φύλλα στυλ** (που είναι η προτιμότερη επιλογή), τα **ενσωματωμένα φύλλα στυλ**, και τα **εμβόλιμα στυλ** (που είναι η λιγότερο επιθυμητή επιλογή).

- Τα **εξωτερικά (external) φύλλα στυλ** είναι ιδανικά για να εφαρμοστεί σε όλες τις σελίδες μιας τοποθεσίας μια συνεπής εμφάνιση. Όλα τα στυλ μπορούν να οριστούν σε ένα εξωτερικό φύλλο στυλ και κατόπιν ζητείται από κάθε σελίδα της τοποθεσίας να φορτώνει αυτό το εξωτερικό φύλλο, εξασφαλίζοντας με αυτό τον τρόπο ότι όλες οι σελίδες θα έχουν τις ίδιες ρυθμίσεις.
- Τα **ενσωματωμένα φύλλα στυλ** (embedded style sheets) είναι η δεύτερη μέθοδος για την εφαρμογή στυλ CSS σε μια σελίδα. Με τη μέθοδο αυτή τα στυλ ορίζονται απευθείας στο έγγραφο HTML το οποίο επηρεάζουν. Αφού τα στυλ υπάρχουν μόνο μέσα στο αρχείο HTML, ο κώδικας CSS δεν θα εφαρμόζεται σε άλλες σελίδες όπως θα συνέβαινε με τα εξωτερικά φύλλα στυλ.
- Τα **εμβόλιμα στυλ (inline styles)** είναι η τρίτη επιλογή για την εφαρμογή στυλ CSS σε σελίδες HTML. Είναι όμως με διαφορά η λιγότερο επιθυμητή επιλογή, επειδή διαπλέκει έντονα το περιεχόμενο (HTML) και την παρουσίαση (CSS), γεγονός που αποτελεί παραβίαση των βέλτιστων πρακτικών. Το εμβόλιμο στυλ επηρεάζει μόνο ένα στοιχείο, έτσι χάνεται ένα από τα βασικά οφέλη που παρέχουν τα εξωτερικά φύλλα στυλ: ότι γράφεται ο κώδικας μία φορά και εμφανίζεται παντού. Αν φανταστεί κανείς τον κόπο που θα χρειαζόταν για να διορθωθούν δεκάδες σελίδες HTML μόνο και μόνο για να γίνει μια απλή αλλαγή γραμματοσειράς, θα καταλάβαινε αμέσως γιατί τα εμβόλιμα στυλ δεν προορίζονται για τακτική χρήση. (Castro & Hyslop, 2012)

Κατασκευή επιλογών

Ο επιλογέας προσδιορίζει τα στοιχεία στα οποία θα εφαρμοστεί ο κανόνας στυλ. Για παράδειγμα, για να μορφοποιηθούν όλα τα στοιχεία **p** με τη γραμματοσειρά Times και ύψος 12 πίκσελ, θα πρέπει να δημιουργηθεί ένας επιλογέας που να προσδιορίζει μόνο τα στοιχεία **p** και να αφήνει όλα τα άλλα στοιχεία του κώδικα ανέπαφα.

Επιλογή στοιχείων με βάση το όνομα

Το πιο συχνά χρησιμοποιούμενο κριτήριο για την επιλογή των στοιχείων που θα μορφοποιηθούν είναι πιθανότατα το όνομα ή ο τύπος του στοιχείου:

```
h2 {  
color : red ;  
}
```

Επιλογή στοιχείων με βάση την κλάση ή το αναγνωριστικό

Αν έχουν χαρακτηριστεί στοιχεία με κλάση (**class**) ή αναγνωριστικό (**id**) μπορεί να χρησιμοποιηθεί στον επιλογέα αυτό το κριτήριο έτσι ώστε να μορφοποιηθούν μόνο τα στοιχεία εκείνα που έχουν τον αντίστοιχο χαρακτηρισμό (chapter είναι το όνομα της κλάσης) :

```
.chapter {  
color : red ;  
}
```

Επιλογή συνδέσμων με βάση την κατάστασή τους

Τα φύλλα CSS επιτρέπουν την εφαρμογή μορφοποίησης σε συνδέσμους με βάση την τρέχουσα κατάστασή τους (state) – δηλαδή αν έχει μεταβεί σε αυτούς ο επισκέπτης, αν έχει τον δείκτη του ποντικιού του επάνω τους, ή οτιδήποτε άλλο. Αυτό επιτυγχάνεται με μια σειρά από ψευδοκλάσεις :

```
a : link {  
color : red ;  
}
```

link : αλλάζει την εμφάνιση των συνδέσμων στους οποίους δεν έχει πατήσει ακόμη ο επισκέπτης και στους οποίους δεν δείχνει ή δεν τους έχει ενεργοποιήσει τη δεδομένη στιγμή.

```
a : visited {  
color : orange ;  
}
```

visited : αλλάζει την εμφάνιση των συνδέσμων τους οποίους έχει ήδη ενεργοποιήσει ο επισκέπτης

```
a : focus {  
color : purple ;  
}
```

focus : αλλάζει την εμφάνιση κάθε συνδέσμου που έχει επιλέξει ο επισκέπτης με το πληκτρολόγιο και είναι έτοιμος να τον ενεργοποιήσει

```
a : hover {  
color : green ;  
}
```

hover : αλλάζει την εμφάνιση των συνδέσμων στους οποίους δείχνει ο επισκέπτης

```
a : active {  
color : blue ;  
}
```

active : αλλάζει την εμφάνιση των συνδέσμων όταν ενεργοποιούνται

Επιλογή στοιχείων με βάση τις ιδιότητες

Μπορούν επίσης να εφαρμοστούν μορφοποιήσεις μόνο στα στοιχεία που έχουν μια συγκεκριμένη ιδιότητα ή τιμή ιδιότητας:

```
section [class] {  
color : red ;  
}
```

Οι αγκύλες περικλείουν την επιθυμητή ιδιότητα και την τυχόν επιθυμητή τιμή. Στην περίπτωση αυτή παραλείπεται η τιμή έτσι ώστε να επιλεγούν όλες οι ενότητες που έχουν ιδιότητα **class**.

Προσδιορισμός ομάδων στοιχείων

Συχνά χρειάζεται να εφαρμοστούν οι ίδιοι κανόνες στυλ σε περισσότερα από ένα στοιχεία. Σε μια τέτοια περίπτωση μπορούν είτε να επαναληφθούν οι κανόνες για κάθε στοιχείο, είτε να συνδυαστούν επιλογείς και να εφαρμοστούν οι κανόνες με μία μόνο κίνηση. Όπως είναι φυσικό, η δεύτερη προσέγγιση είναι πιο αποδοτική και κάνει γενικά ευκολότερη τη συντήρηση των φύλλων στυλ.

```
h1,  
h2 {  
color : red ;  
}
```

Μπορεί να χρησιμοποιηθεί οποιοδήποτε πλήθος μεμονωμένων επιλογέων (ανεξάρτητα από το αν περιλαμβάνουν ονόματα στοιχείων, αναγνωριστικά, ή κλάσεις), αρκεί να χωρίζονται μεταξύ τους με κόμματα. Ο κάθε επιλογέας δεν είναι υποχρεωτικό να βρίσκεται σε δική του γραμμή, όμως πολλοί σχεδιαστές χρησιμοποιούν αυτή τη σύμβαση για να είναι το φύλλο στυλ πιο ευανάγνωστο. Το όφελος είναι πιο προφανές όταν οι επιλογείς είναι πιο μακροσκελείς.

4.2 ΜΟΡΦΟΠΟΙΗΣΗ ΚΕΙΜΕΝΟΥ ΜΕ ΣΤΥΛ

Αν και πολλές από τις ιδιότητες που αναλύονται παρακάτω εφαρμόζονται κυρίως σε κείμενο, αυτό δε σημαίνει ότι λειτουργούν αποκλειστικά και μόνο με κείμενο. Πολλές από αυτές λειτουργούν εξίσου καλά και για άλλους τύπους περιεχομένου.

Επιλογή γραμματοσειράς

Μια από τις σημαντικότερες επιλογές που γίνονται για την ιστοσελίδα είναι η γραμματοσειρά για το σώμα κειμένου και τις επικεφαλίδες. Όμως, δεν υποστηρίζουν εξ ορισμού όλα τα συστήματα τις ίδιες γραμματοσειρές κι έτσι θα πρέπει να οριστούν εναλλακτικές γραμματοσειρές ως εναλλακτικές λύσεις.

```
body {  
  
font-family : " Palatino Linotype" ;  
  
}  
  
h1, h2 {  
  
font-family : "Arial Black" ;  
  
}
```

Καθορισμός μεγέθους γραμματοσειράς

```
h1 {  
  
font-size : 22px ;  
  
}  
  
h2 {  
  
font-size : 15px ;  
  
}
```

Καθορισμός του χρώματος

Το χρώμα των στοιχείων της ιστοσελίδας μπορεί επίσης να αλλάξει χρησιμοποιώντας ονόματα χρωμάτων, δεκαεξαδικές τιμές ή τιμές RGB :

```
h2 {  
color : #c3f ;  
}
```

Αλλαγή του φόντου κειμένου

Το φόντο μπορεί να αλλάξει για μεμονωμένα στοιχεία, για ολόκληρη τη σελίδα ή για ένα συνδυασμό των δύο. Έτσι μπορεί να γίνει αλλαγή μόνο για λίγες παραγράφους ή λέξεις, για τους συνδέσμους στις διάφορες καταστάσεις τους, για ενότητες περιεχομένου και για πολλά άλλα.

```
body {  
background : #eef ;  
}  
h1 {  
background : #ebc6f9 ;  
}
```

Ρύθμιση των αποστάσεων

Η απόσταση μεταξύ των λέξεων ή μεταξύ των γραμμμάτων μπορεί να αυξηθεί ή να μειωθεί:

```
h2 {  
word-spacing : 5px ;  
letter-spacing : 5px ;  
}
```

Στοίχιση κειμένου

Το κείμενο μπορεί να ρυθμιστεί έτσι ώστε να στοιχίζεται πάντα στα δεξιά, στα αριστερά, στο κέντρο ή πλήρως (και αριστερά και δεξιά).

- στοίχιση του κειμένου αριστερά : **left**
- στοίχιση του κειμένου δεξιά : **right**

- στοίχιση του κειμένου στο κέντρο : **center**
- πλήρης στοίχιση του κειμένου (και προς τα αριστερά και προς τα δεξιά) : **justify**

```
p {  
text-align : center ;  
}
```

4.3 ΔΙΑΤΑΞΗ ΣΕΛΙΔΩΝ ΜΕ ΣΤΥΛ

Τα φύλλα CSS μπορούν να δημιουργήσουν μια μεγάλη ποικιλία διατάξεων σελίδας, εκτός από το συνηθισμένο τύπο διάταξης σελίδας (μια κύρια επικεφαλίδα στην κορυφή, δύο στήλες περιεχομένου και ένα υποσέλιδο στο κάτω μέρος).

Καθορισμός του ύψους ή του πλάτους για ένα στοιχείο

```
#main {  
width : 70% ;  
}  
.photo {  
height : 75px ;  
}
```

Το πλάτος και το ύψος μπορούν να εκφραστούν είτε ως μήκος (με μονάδες όπως px ή em), είτε ως ποσοστό του γονικού στοιχείου.

Καθορισμός των περιθωρίων γύρω από ένα στοιχείο

Το περιθώριο είναι η ποσότητα διαφανούς χώρου ανάμεσα σε ένα στοιχείο και το επόμενο.

```
#container {  
margin : 20px ;  
}
```

Μπορεί επίσης να προστεθεί ένα από τα ακόλουθα επιθέματα στην ιδιότητα **margin** για να εφαρμοστεί περιθώριο σε μία μόνο πλευρά : **-top**, **-bottom**, **-left**, **-right**. Δεν πρέπει να υπάρχει κενό μετά τη λέξη margin

```
margin-top : 10px ;
```

Προσθήκη συμπλήρωσης γύρω από ένα στοιχείο

Η συμπλήρωση είναι πρόσθετος χώρος γύρω από τα περιεχόμενα ενός στοιχείου, μέσα όμως από το περίγραμμα.

```
#container {  
  
padding : 30px 10px 0 0 ;  
  
}
```

Όπως και με τα περιθώρια, αν καθοριστούν τέσσερις τιμές για την ιδιότητα padding, αυτές θα αντιστοιχιστούν με ωρολογιακή φορά (πάνω, δεξιά, κάτω και αριστερά).

Δημιουργία στοιχείων που επιπλέουν

Με την τεχνική αυτή το κείμενο μπορεί να αναδιπλώνεται γύρω από εικόνες ή σχήματα, να δημιουργηθούν πολύστηλες διατάξεις σελίδας και πολλά άλλα.

```
.photo {  
  
float : left ;  
  
}
```

- **left** : το στοιχείο βρίσκεται στα αριστερά και το υπόλοιπο περιεχόμενο ρέει στα δεξιά του
- **right** : το στοιχείο βρίσκεται στα δεξιά και το υπόλοιπο περιεχόμενο ρέει στα αριστερά του

Έλεγχος του σημείου όπου επιπλέουν τα στοιχεία

Μπορεί να ρυθμιστεί γύρω από ποια στοιχεία μπορεί να επιπλέει ένα στοιχείο και γύρω από ποια δεν μπορεί. Για να εμποδιστεί ένα στοιχείο να επιπλέει δίπλα σε κάτι που δεν πρέπει, χρησιμοποιείται η ιδιότητα **clear**.

- **left** : εμποδίζονται τα στοιχεία να επιπλέουν αριστερά από το στοιχείο που μορφοποιείται
- **right** : εμποδίζονται τα στοιχεία να επιπλέουν δεξιά από το στοιχείο που μορφοποιούμε
- **both** : εμποδίζονται τα στοιχεία να επιπλέουν και στις δύο πλευρές του στοιχείου που μορφοποιείται
- **none** : για να επιτραπεί στα στοιχεία να επιπλέουν σε οποιαδήποτε πλευρά του στοιχείου που μορφοποιείται

```
#footer {  
  
clear : both ;
```

}

Η ιδιότητα **clear** δηλώνει ότι το αντίστοιχο στοιχείο (στην περίπτωση αυτή το υποσέλιδο) δεν θα πρέπει να ρέει γύρω από το στοιχείο που επιπλέει, αλλά θα πρέπει να εμφανίζεται μετά από το στοιχείο αυτό.

Ρύθμιση του περιγράμματος

Περίγραμμα μπορεί να δημιουργηθεί γύρω από μεμονωμένες πλευρές ενός στοιχείου και κατόπιν να καθοριστεί το πάχος του, η μορφή του και το χρώμα του.

- **border-style** : ορίζει το στυλ του περιγράμματος (none, dotted, solid, dashed)
- **border-width** : ορίζει το πάχος του περιγράμματος
- **border-color** : ορίζει το χρώμα του περιγράμματος
-

```
#footer {
```

```
border-top : 2px dotted #b74e07 ;
```

```
}
```

Απόλυτη τοποθέτηση στοιχείων

Τα στοιχεία της ιστοσελίδας ρέουν κατά κανόνα με τη σειρά με την οποία εμφανίζονται στον πηγαίο κώδικα HTML. Με άλλα λόγια, αν το στοιχείο **img** βρίσκεται πριν από το στοιχείο **p**, η εικόνα θα εμφανίζεται πριν από την παράγραφο. Μπορούν όμως να αποσπαστούν στοιχεία από τη φυσική ροή και να τα τοποθετηθούν σε απόλυτες θέσεις, καθορίζοντας την ακριβή θέση τους ως προς το στοιχείο **body** ή το πλησιέστερο τοποθετημένο προγονικό στοιχείο.

```
#masthead form {
```

```
position : absolute ;
```

```
}
```

Χειρισμός της υπερχείλισης

Τα στοιχεία δεν περιορίζονται πάντα μέσα στα πλαίσιά τους. Κάποιες φορές το πλαίσιο απλώς δεν είναι αρκετά μεγάλο. Για παράδειγμα, μια εικόνα που είναι πιο πλατιά από τον αποδέκτη της θα περισσεύει έξω από αυτόν. Ή ίσως να έχει τοποθετηθεί το περιεχόμενο έξω από το πλαίσιο, είτε με αρνητικά περιθώρια είτε με απόλυτη τοποθέτηση. Ανεξάρτητα από την αιτία, η περιοχή έξω από το πλαίσιο του στοιχείου μπορεί να ελεγχθεί την με την ιδιότητα **overflow** (υπερχείλιση).

- **visible** : επεκτείνεται το πλαίσιο του στοιχείου έτσι ώστε να χωρά το περιεχόμενο (αυτή είναι η προεπιλεγμένη ρύθμιση)

- **hidden** : κρύβει όποιο περιεχόμενο δεν χωρά μέσα στο πλαίσιο του στοιχείου
- **scroll** : προστίθενται ράβδοι κύλισης στο στοιχείο, έτσι ώστε να μπορούν οι επισκέπτες να προσπελάσουν τα υπερχειλίζοντα στοιχεία αν το επιθυμούν
- **auto** : εμφανίζονται ράβδοι κύλισης μόνο όταν χρειάζεται

```
.thumbnails {
```

```
overflow : hidden ;
```

```
}
```

Εμφάνιση και απόκρυψη στοιχείων

Η ιδιότητα **display** είναι πολυπρόσωπη. Ο φυσικός τύπος εμφάνισης ενός στοιχείου μπορεί να ελεγχθεί, για παράδειγμα να αλλάξει από **inline** σε **block** ή το αντίστροφο. Η ιδιότητα **display** είναι επίσης χρήσιμη όταν πρέπει να αποτραπεί ένα στοιχείο και το περιεχόμενό του από το να καταλάβουν ζωτικό χώρο στη σελίδα.

Από την άλλη πλευρά, ο κύριος ρόλος της ιδιότητας **visibility** είναι προφανώς το να ελέγξει αν ένα στοιχείο θα είναι ορατό. Σε αντίθεση με την ιδιότητα **display**, όταν κρύβεται ένα στοιχείο με την ιδιότητα **visibility**, ο κενός χώρος δείχνει πού θα εμφανιζόταν διαφορετικά το στοιχείο και τα περιεχόμενά του.

```
img {
```

```
display : block ;
```

```
}
```

Αν οριστούν οι εικόνες με την ιδιότητα `display : block ;` αυτές θα εμφανίζονται πλέον σε ξεχωριστές γραμμές, όπως συμβαίνει και με τις παραγράφους και τα άλλα στοιχεία που εμφανίζονται εξ ορισμού ως στοιχεία επιπέδου μπλοκ. Αν όμως χρησιμοποιηθεί η ρύθμιση **display : none;** δεν θα παραμείνει κανένα ορατό ίχνος του στοιχείου στον φυλλομετρητή.

```
img {
```

```
visibility : hidden ;
```

```
}
```

ΚΕΦΑΛΑΙΟ 5

5.1 Η ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ

Η σωστή σχεδίαση μίας βάσης δεδομένων είναι ο μοναδικός τρόπος ο οποίος μπορεί να εγγυηθεί αποτελεσματικότητα, ευελιξία και δυνατότητα εύκολης διαχείρισης/συντήρησης μιας εφαρμογής. Η βάση δεδομένων θα πρέπει να είναι εύκολη στη συντήρηση και αυτό σημαίνει ότι θα πρέπει να αποθηκεύει όσο το δυνατόν λιγότερα επαναλαμβανόμενα δεδομένα. Εάν υπάρχουν πολλά επαναλαμβανόμενα δεδομένα στη βάση δεδομένων και γίνει μία αλλαγή σε κάποιο από αυτά, θα πρέπει να γίνει η ίδια αλλαγή σε όλες τις υπόλοιπες εμφανίσεις του στην βάση δεδομένων.

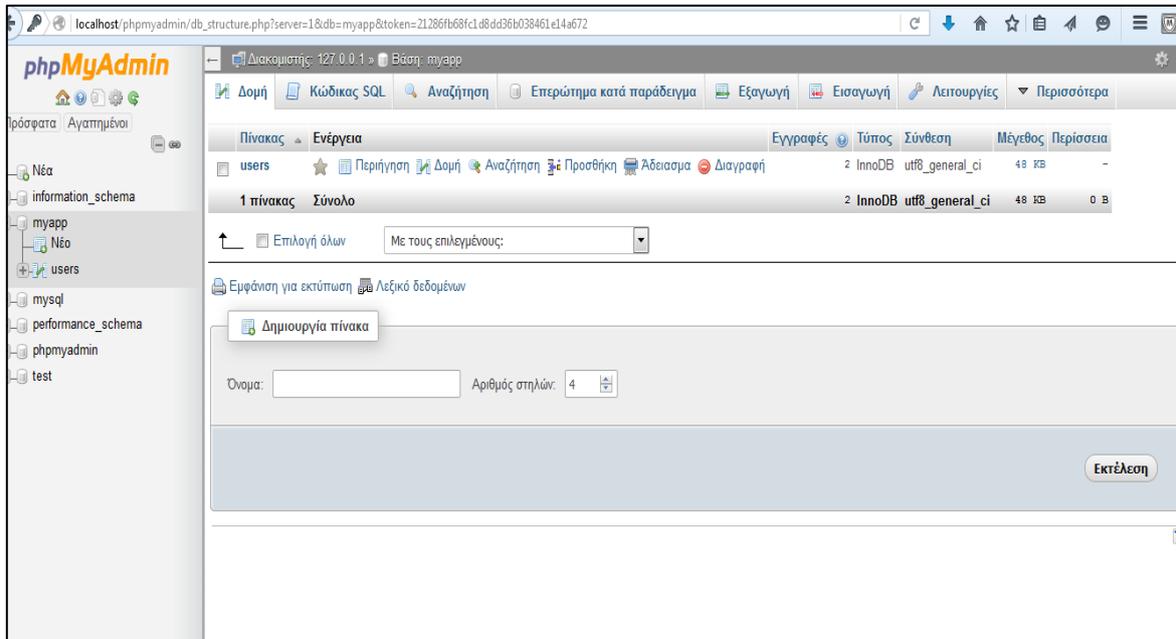
Τα πλεονεκτήματα που παρέχει μία καλοσχεδιασμένη βάση δεδομένων είναι πολλά. Όσο περισσότερη δουλειά γίνει εκ των προτέρων για το σωστό σχεδιασμό της βάσης δεδομένων, τόσο λιγότερη θα χρειαστεί να γίνει εκ των υστέρων. (Meloni, 2012)

Βασικές εντολές της SQL

- CREATE TABLE όνομα_πίνακα (όνομα_στήλης, τύπος στήλης) ; η εντολή για την δημιουργία ενός πίνακα
- INSERT INTO όνομα_πίνακα (λίστα στηλών) VALUES (τιμές στηλών) ; προσθήκη νέων εγγραφών σε πίνακα
- SELECT εκφράσεις_και_στήλες FROM όνομα_πίνακα ; ανάκτηση εγγραφών από ένα πίνακα
- UPDATE όνομα_πίνακα
SET στήλη1 = ' νέα τιμή ',
στήλη2 = ' νέα τιμή2 '
[WHERE συνθήκη_αληθής] ; τροποποίηση του περιεχομένου μίας ή περισσότερων στηλών σε μία υπάρχουσα εγγραφή ενός πίνακα
- REPLACE INTO όνομα_πίνακα (λίστα στηλών) VALUES (τιμές στηλών) ; τροποποίηση των εγγραφών ενός πίνακα
- DELETE FROM όνομα_πίνακα
[WHERE συνθήκη-αληθής]
[LIMIT γραμμές] ; διαγραφή ολόκληρων εγγραφών

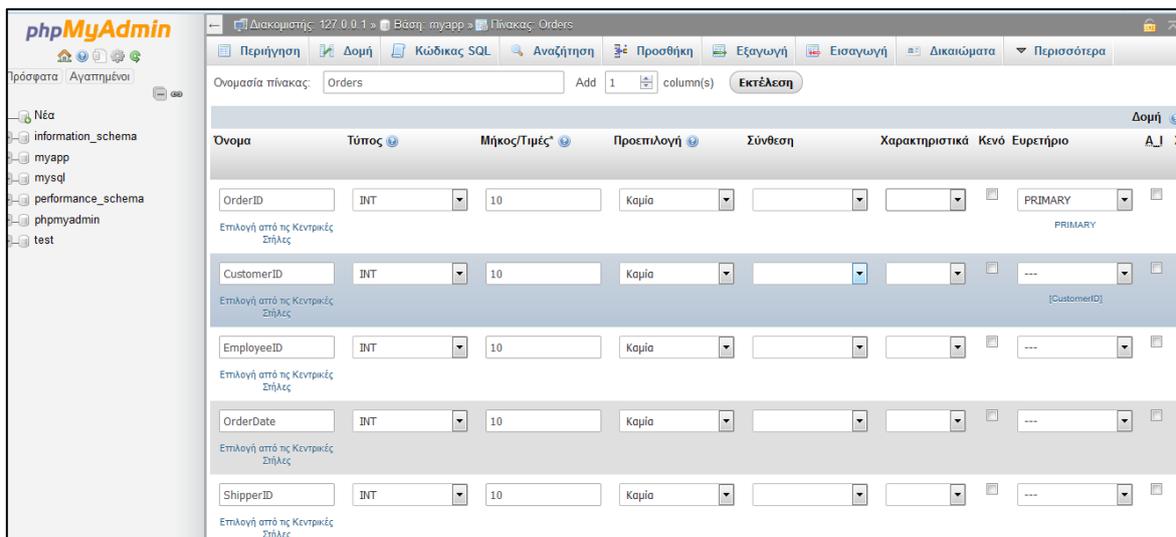
5.2 ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΑ

Στην παρούσα ενότητα θα περιγραφεί η διαδικασία δημιουργίας ενός πίνακα, καθώς οι πίνακες αποτελούν απαραίτητο κομμάτι στην υλοποίηση της εκπαιδευτικής πλατφόρμας της εργασίας. Η κατασκευή του πίνακα θα γίνει με την **phpMyAdmin** :



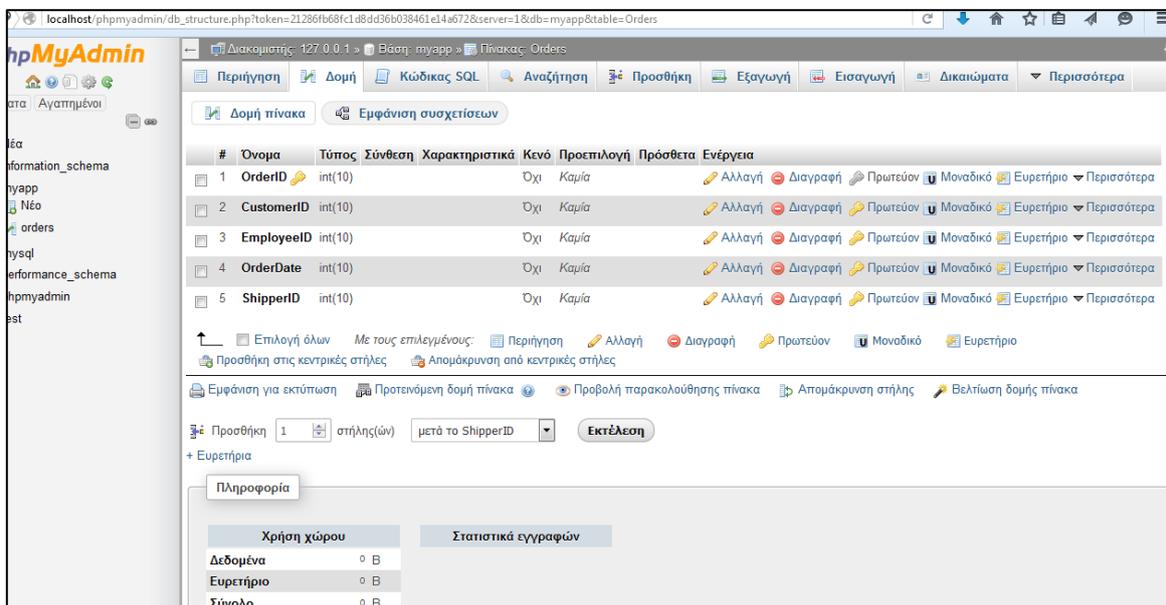
Εικόνα 5.1 Δημιουργία πίνακα

Αρχικά, θα πρέπει να γίνει εισαγωγή του ονόματος του πίνακα και ο αριθμός των στηλών και στη συνέχεια επιλογή του κουμπιού "Εκτέλεση" :



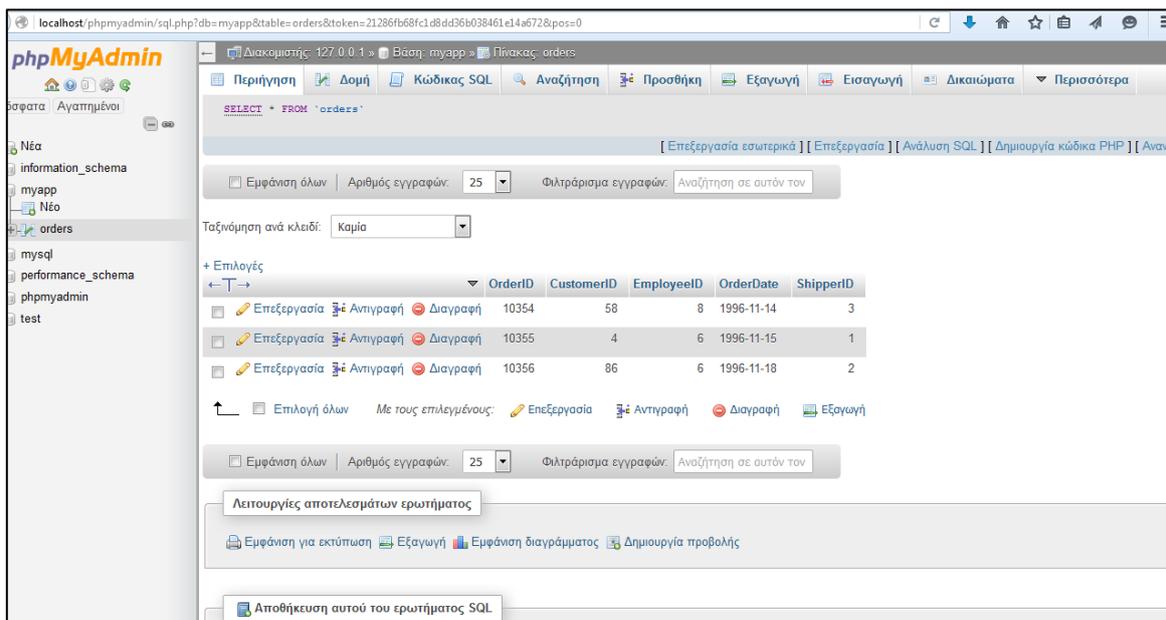
Εικόνα 5.2 Εισαγωγή πεδίων του πίνακα

Στην Εικόνα 5.2 εμφανίζονται τα πλαίσια όπου ορίζεται το κάθε πεδίο του πίνακα, εισάγοντας το όνομα του πεδίου, τον τύπο, κ.α. Αφού γίνει η εισαγωγή των στοιχείων για τα πεδία, επιλέγεται το κουμπί "Αποθήκευση".



Εικόνα 5.3 Επιτυχής δημιουργία των πεδίων

Όπως φαίνεται και στην παραπάνω εικόνα, τα πεδία του πίνακα 'Orders' έχουν δημιουργηθεί και κατόπιν θα γίνει η εισαγωγή των τιμών στα πεδία. Αυτό γίνεται επιλέγοντας 'Προσθήκη' και στη συνέχεια 'Περιήγηση' :



Εικόνα 5.4 Επιτυχής εισαγωγή εγγραφών στον πίνακα

Μετά την εισαγωγή και των τιμών στα πεδία, δημιουργείται ο παραπάνω πίνακας ο οποίος περιέχει πέντε πεδία (' OrderID ' , ' CustomerID ' , ' EmployeeID ' , ' OrderDate ' , ' ShipperID ') και τρεις εγγραφές στο πεδίο κάθε πίνακα. Η διαδικασία δημιουργίας ενός πίνακα σε βάση δεδομένων πραγματοποιήθηκε!

5.3 ΣΥΝΔΕΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΜΕ PHP

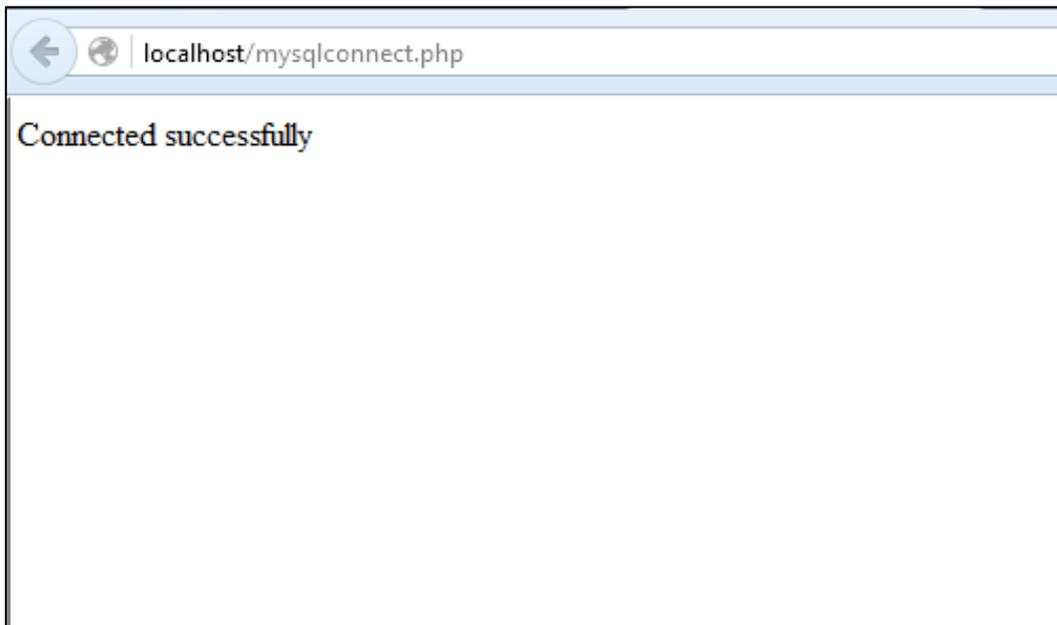
Για την επικοινωνία με την MySQL χρησιμοποιείται ένα λειτουργικό script σύνδεσης, το οποίο είναι το ακόλουθο :

```
<?php
$servername = "localhost";
$username = "myuser";
$password = "pass";

// Create connection
$conn = new mysqli ($servername, $username, $password);

// Check connection
if ($conn->connect_error) {
die ("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

Ο κώδικας δημιουργεί νέα σύνδεση και μετά ελέγχει για να δει αν συνέβη ένα σφάλμα. Αν συνέβη σφάλμα, εκτυπώνεται το μήνυμα σφάλματος " **Connection failed** ". Αν δεν υπάρχουν σφάλματα, εκτυπώνεται το εξής μήνυμα " **Connected successfully** " και η εμφάνισή του στην οθόνη θα μοιάζει με την Εικόνα 5.5 :



Εικόνα 5.5 Επιτυχής σύνδεση με βάση δεδομένων

Ανάκτηση Δεδομένων με την PHP

Τα περιεχόμενα ενός πίνακα μπορούν να ανακτηθούν και να εμφανιστούν στην οθόνη χρησιμοποιώντας ένα script :

```
<?php
```

```
$servername = "localhost";
```

```
$username = "myuser";
```

```
$password = "pass";
```

```
$dbname = "myapp";
```

```
// Create connection
```

```
$conn = new mysqli($servername, $username, $password, $dbname);
```

```
// Check connection
```

```
if ($conn->connect_error) {
```

```
die("Connection failed: " . $conn->connect_error);
```

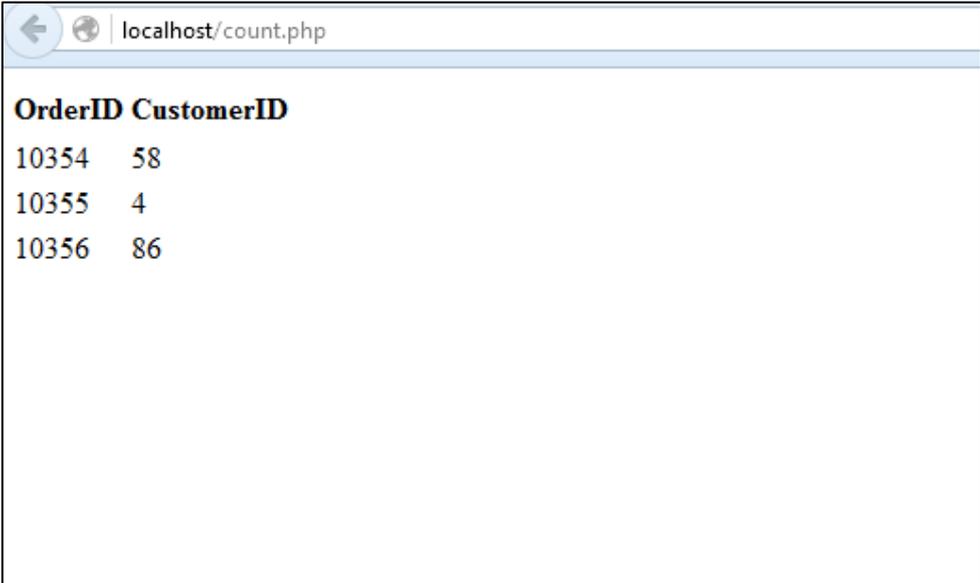
```
}
```

```

$sql = "SELECT OrderID, CustomerID FROM orders";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
echo "<table><tr><th>OrderID</th><th>CustomerID</th></tr><tr>";
// output data of each row
while($row = $result->fetch_assoc()) {
echo"<tr><td>".$row["OrderID"]."</td><td>".$row["CustomerID"]."
".$row["pass"]."</td></tr>";
}
echo "</table>";
} else {
echo "0 results";
}
$conn->close();
?>

```

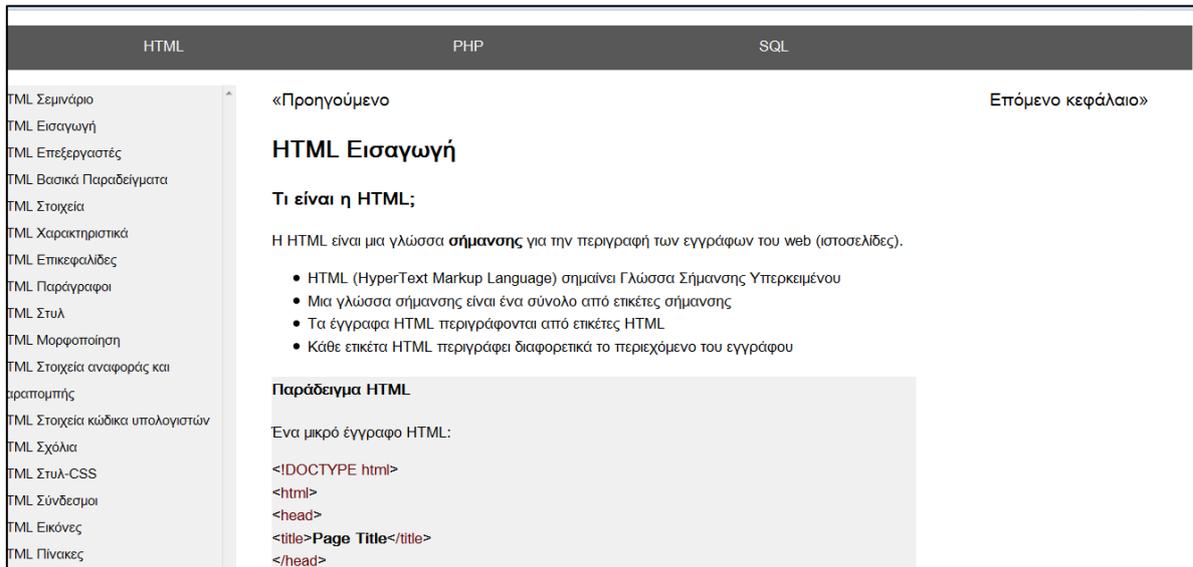


OrderID	CustomerID
10354	58
10355	4
10356	86

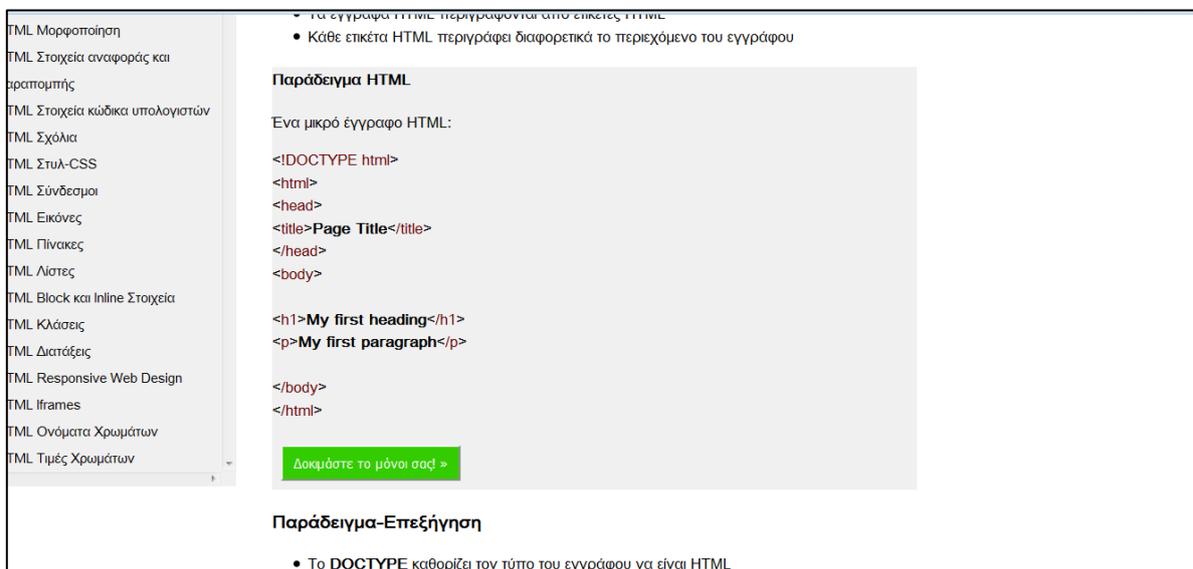
Εικόνα 5.6 Εμφάνιση εγγραφών

ΚΕΦΑΛΑΙΟ 6

6.1 ΠΑΡΟΥΣΙΑΣΗ ΠΛΑΤΦΟΡΜΑΣ



Εικόνα 6.1 Αρχική σελίδα πλατφόρμας



Εικόνα 6.2 Παράδειγμα κώδικα της HTML

Στις παραπάνω εικόνες εμφανίζεται η σελίδα που εισάγει βασικές έννοιες της **HTML**. Επάνω υπάρχει το **μενού** μέσω του οποίου γίνεται η περιήγηση στην HTML, την PHP ή στην SQL. Αριστερά υπάρχει το **πλαίσιο** με το μενού το οποίο οδηγεί στην κάθε ενότητα της γλώσσας.

Το **κύριο μέρος** της σελίδας περιλαμβάνει τον τίτλο της ενότητας, συνδέσμους μεταφοράς στην προηγούμενη ή την επόμενη ενότητα καθώς και θεωρία για τα στοιχεία της ενότητας αυτής.

Κάτω από τη θεωρία υπάρχει παράδειγμα με κομμάτι HTML κώδικα με ένα κουμπί για δοκιμή. Πατώντας το **κουμπί** αυτό θα ανοίξει μια νέα σελίδα, όπου γίνονται **τροποποιήσεις του κώδικα**. Αυτό είναι και το πιο σημαντικό στην εφαρμογή, γιατί γίνονται καλύτερα κατανοητά όσα παρουσιάζονται στη θεωρία τροποποιώντας τα κομμάτια του κώδικα.

Παράδειγμα-Επεξήγηση

- Το **DOCTYPE** καθορίζει τον τύπο του εγγράφου να είναι HTML
- Το κείμενο μεταξύ **<html>** και **</html>** περιγράφει ένα έγγραφο HTML
- Το κείμενο μεταξύ **<head>** και **</head>** παρέχει πληροφορίες σχετικά με το έγγραφο
- Το κείμενο μεταξύ των **<title>** και **</title>** προβλέπει έναν τίτλο για το έγγραφο
- Το κείμενο μεταξύ **<body>** και **</body>** περιγράφει το ορατό περιεχόμενο της σελίδας
- Το κείμενο μεταξύ **<h1>** και **</h1>** περιγράφει μια επικεφαλίδα
- Το κείμενο μεταξύ **<p>** και **</p>** περιγράφει μια παράγραφο

Χρησιμοποιώντας αυτήν την περιγραφή, ένας web browser μπορεί να εμφανίσει ένα έγγραφο με τίτλο και μια παράγραφο.

HTML Ετικέτες

Οι ετικέτες HTML είναι **λέξεις-κλειδιά** (ονόματα tag) που περιβάλλονται από **αγκύλες**:

<tagname> περιεχόμενο **</tagname>**

- Οι ετικέτες HTML συνήθως είναι **σε ζεύγη**, όπως **<p>** και **</p>**.
- Η πρώτη ετικέτα σε ένα ζευγάρι είναι η **ετικέτα έναρξης**, η δεύτερη ετικέτα είναι η **ετικέτα τέλους**
- Η ετικέτα τέλους είναι γραμμένη σαν την ετικέτα έναρξης, αλλά με μια **κάθετο** πριν από το όνομα ετικέτας

Η ετικέτα έναρξης συχνά αποκαλείται **ετικέτα ανοίγματος**. Η ετικέτα τέλους συχνά αποκαλείται **ετικέτα κλεισίματος**

Προνόμιατα περιήγησης στο Web

Εικόνα 6.3 Επεξήγηση παραδείγματος

Κάτω από το παράδειγμα υπάρχει επεξήγηση του κώδικα και ακολουθεί **πίνακας** με πληροφορίες για τη δομή μιας HTML σελίδας (Εικόνα 6.4) :

Δομή HTML σελίδας

Παρακάτω απεικονίζεται η δομή μιας σελίδας HTML:

```
<html>
  <head>
    <title>Τίτλος σελίδας</title>
  </head>
  <body>
    <h1>Αυτή είναι μια επικεφαλίδα</h1>
    <p>Αυτή είναι μια παράγραφος</p>
    <p>Αυτή είναι μια άλλη παράγραφος</p>
  </body>
</html>
```

Εικόνα 6.4 Δομικά HTML στοιχεία

Σε όλα τα παραδείγματα χρησιμοποιείται HTML5.

Εκδόσεις HTML

Από τις πρώτες ημέρες του Παγκόσμιου Ιστού, έχουν υπάρξει πολλές εκδόσεις της HTML:

Έκδοση	Έτος
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

«Προηγούμενο

Επόμενο κεφάλαιο»

Παραδείγματα HTML

Στοιχεία
Χαρακτηριστικά

Παραδείγματα PHP

Μεταβλητές
Τύποι δεδομένων

Παραδείγματα SQL

Select
Update
Delete

Εικόνα 6.5 Αναφορά παλαιότερων εκδόσεων

Το τελευταίο κομμάτι της σελίδας που αφορά την εισαγωγή στην html κλείνει με πίνακα των παλαιότερων εκδόσεων της html, συνδέσμους για μεταφορά στην προηγούμενη ή την επόμενη ενότητα και το υποσέλιδο. Το **υποσέλιδο** περιέχει παραδείγματα από κάθε γλώσσα ξεχωριστά.

6.2 JAVASCRIPT ΚΑΙ HTML

Όπως αναφέρθηκε και στην προηγούμενη ενότητα, το μεγάλο πλεονέκτημα αυτής της εφαρμογής είναι οι δοκιμές που γίνονται στον κώδικα της κάθε γλώσσας. Σε αυτή την ενότητα θα περιγραφεί με ποιο τρόπο γίνεται αυτό.

Παράδειγμα HTML

Ένα μικρό έγγραφο HTML:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

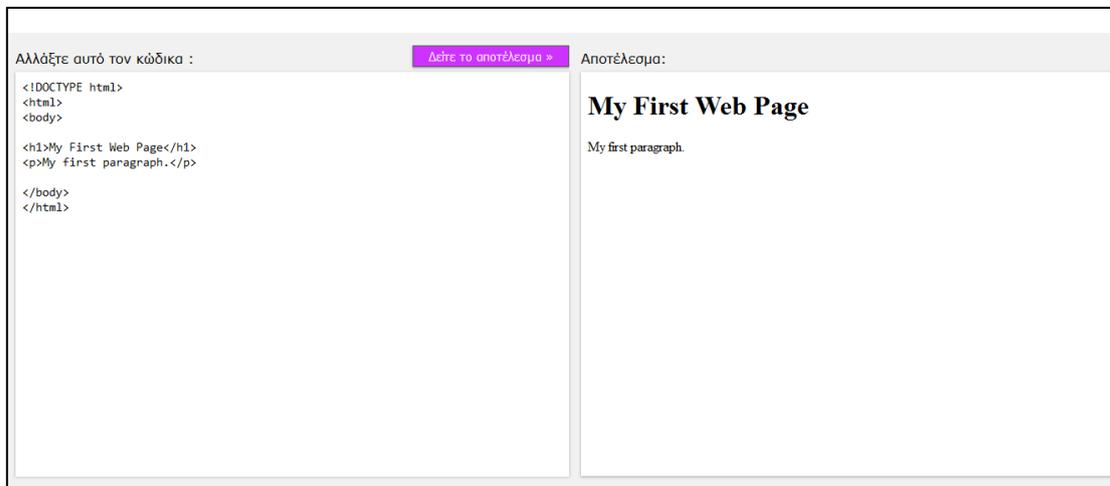
<h1>My first heading</h1>
<p>My first paragraph</p>

</body>
</html>
```

Δοκιμάστε το μόνοι σας! »

Εικόνα 6.6 Παράδειγμα κώδικα

Πατώντας το κουμπί «**Δοκιμάστε μόνοι σας!**» θα ανοίξει η παρακάτω σελίδα όπου μπορούν να γίνουν οι αλλαγές στον κώδικα.



Εικόνα 6.7 Παράδειγμα της εφαρμογής

Η **Javascript** χρησιμοποιείται για να δώσει **διαδραστικότητα** μεταξύ χρήστη και HTML εγγράφου, έτσι ενσωματώνεται σε HTML σελίδες. Για να κατασκευαστεί η παραπάνω σελίδα χρησιμοποιήθηκε μια φόρμα (στοιχείο **form**) και η Javascript χρησιμοποιήθηκε για να επεξεργαστεί το στοιχείο αυτό. Ο HTML κώδικας είναι :

```
<div class = "container" >
```

```
<div class = "textareaccontainer" >
```

```
<div class = "textarea" >
```

```
<div style = "overflow: auto;" >
```

```
<div class = "headerText" > Αλλάξτε αυτό τον κώδικα : </div>
```

```
<button class = "seeResult" type = "button" onclick = "submitTryit ( )" > Δείτε το
```

```
αποτέλεσμα
```

```
& raquo; </button>
```

```
</div>
```

```
<div class = "textareawrapper" >
```

```
<textarea autocomplete = "off" class = "code_input" id = "textareaCode" wrap = "logical"
```

```
spellcheck = "false" >
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1> My First Web Page </h1>
```

```
<p> My first paragraph </p>
```

```
</body>
```

```
</html>
```

```
</textarea>
```

```
<form autocomplete = "off" style = "margin: 0px; display: none;" >
```

```
<input type = "text" name = "code" id = "code" />
```

```
<input type = "text" id = "bt" name = "bt" />
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class = "iframecontainer" >
```

```
<div class = "iframe" >
```

```
<div style = "overflow: auto;" >
```

```
<div class = "headerText" > Αποτέλεσμα: </div>
```

```
</div>
```

```
<div id = "iframewrapper" class = "iframewrapper" >
```

```
</div>
```

```
</div>
```

```
</div
```

Η σελίδα χωρίζεται σε δύο μεγάλα πλαίσια μέσα σε στοιχεία **<div>**. Στο αριστερό πλαίσιο βρίσκεται το τμήμα του κώδικα στο οποίο γίνονται οι δοκιμές. Επάνω από το πλαίσιο αυτό βρίσκεται το κουμπί και με το πάτημά του παρουσιάζεται το αποτέλεσμα των δοκιμών. Το κουμπί αυτό δημιουργήθηκε με το στοιχείο **<button>**. Με την ιδιότητα **onclick** που περιέχεται σε αυτό, καλείται η **Javascript** για να συλλέξει και να επεξεργαστεί τα δεδομένα που εισάγονται στο στοιχείο **<textarea>** -αριστερό πλαίσιο- και να τα εμφανίσει στη δεξιά πλευρά. Παρακάτω παρουσιάζεται ο κώδικας της **Javascript**:

```
<script type = "text/javascript" >
function submitTryit( ) {
var text = document.getElementById("textareaCode").value;
var ifr = document.createElement ("iframe");
ifr.setAttribute ("frameborder", "0");
ifr.setAttribute ("id", "iframeResult");
document.getElementById ("iframewrapper") .innerHTML = "";
document.getElementById ("iframewrapper" ) .appendChild (ifr);
var ifrw = (ifr.contentWindow) ? ifr.contentWindow : (ifr.contentDocument.document)
? ifr.contentDocument.document : ifr.contentDocument;
ifrw.document.open ( );
ifrw.document.write (text);
ifrw.document.close ( );
}
</script>
```

Η **function** φορτώνεται στη μνήμη του browser και δε χρησιμοποιείται μέχρι να γραφτεί κάτι και να πατηθεί το κουμπί (**button**). Με το πάτημα του κουμπιού καλείται η function (μέσω της ιδιότητας **onclick** που περιέχει τη function **submitTryit ()**). Στη συνέχεια η function αποκτά πρόσβαση στα HTML στοιχεία με την εντολή **document.getElementById** χρησιμοποιώντας το **id** του στοιχείου, επεξεργάζεται τα δεδομένα με την **.innerHTML** και επιστρέφει το αποτέλεσμα.

Η μορφοποίηση των σελίδων για τις δοκιμές έγινε επίσης με CSS κώδικα, τμήμα του οποίου παρουσιάζεται παρακάτω:

```
.container {  
    background-color: #f1f1f1;  
    width: 100%;  
    overflow : auto;  
    position: absolute;  
    top: 100px;  
    bottom: 0;  
    height: auto;  
    min-height: 250px;  
    min-width: 550px;  
}
```

```
.textareacontainer,  
.iframecontainer {  
    float: left;  
    height: 100%;  
    width: 50%;  
}
```

```
.textarea, .iframe {  
    height: 100%;  
    width: 100%;  
    padding: 15px;  
}
```

```
.textarea {
```

```
padding-right: 7px;  
}
```

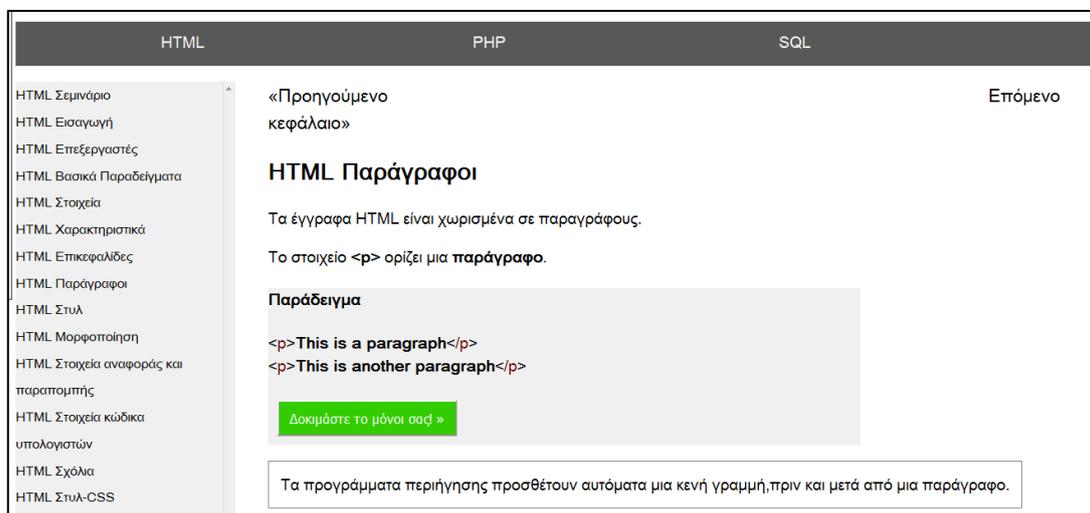
```
.iframe {  
    padding-left: 7px;  
}
```

```
.headerText {  
    width: auto;  
    float: left;  
    font-family: verdana;  
    font-size: 1em;  
    line-height: 2;  
}
```

```
.seeResult {  
    float: right;  
    font-size: 15px;  
    background-color: #c3f;  
    color: #ffffff;  
    border: 1px solid #555555;  
    padding: 0 15px;  
    line-height: 1.45;  
    width: auto;  
    cursor: pointer;  
}
```

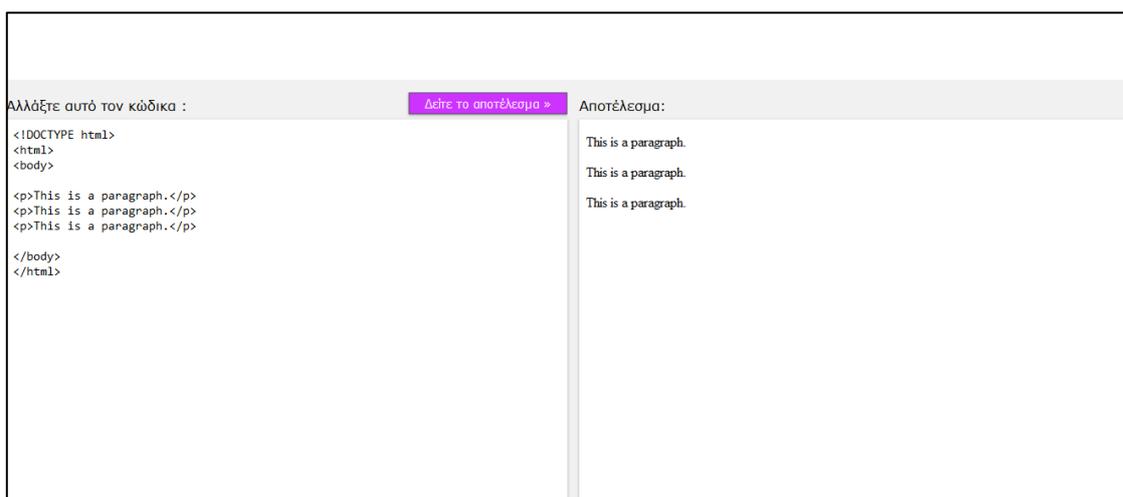
6.3 ΠΑΡΑΔΕΙΓΜΑΤΑ

Τέλος, αφού παρουσιάστηκε η υλοποίηση της εφαρμογής, θα ακολουθήσουν λίγα παραδείγματα των δυνατοτήτων που προσφέρει η εκπαιδευτική πλατφόρμα :



The screenshot shows a web interface for an HTML tutorial. At the top, there are three tabs: HTML, PHP, and SQL. The HTML tab is active. On the left, there is a sidebar with a list of navigation links: HTML Σεμινάριο, HTML Εισαγωγή, HTML Επεξεργαστές, HTML Βασικά Παραδείγματα, HTML Στοιχεία, HTML Χαρακτηριστικά, HTML Επικεφαλίδες, HTML Παράγραφοι, HTML Στυλ, HTML Μορφοποίηση, HTML Στοιχεία αναφοράς και παραπομπής, HTML Στοιχεία κώδικα υπολογιστών, HTML Σχόλια, and HTML Στυλ-CSS. The main content area is titled «Προηγούμενο κεφάλαιο» and «Επόμενο». The main heading is «HTML Παράγραφοι». Below the heading, there is text explaining that HTML characters are separated into paragraphs and that the <p> tag defines a paragraph. A code example shows <p>This is a paragraph</p> and <p>This is another paragraph</p>. A green button says «Δοκιμάστε το μόνοι σας! >». At the bottom, a note says «Τα προγράμματα περιήγησης προσθέτουν αυτόματα μια κενή γραμμή, πριν και μετά από μια παράγραφο.»

Εικόνα 6.8 Παράδειγμα HTML



The screenshot shows a code editor interface. On the left, there is a text area with the label «Αλλάξτε αυτό τον κώδικα :». Below it is the HTML code: <!DOCTYPE html>, <html>, <body>, <p>This is a paragraph.</p>, <p>This is a paragraph.</p>, <p>This is a paragraph.</p>, </body>, and </html>. A purple button says «Δείτε το αποτέλεσμα >». On the right, there is a text area with the label «Αποτέλεσμα:». Below it is the rendered output: This is a paragraph., This is a paragraph., and This is a paragraph.

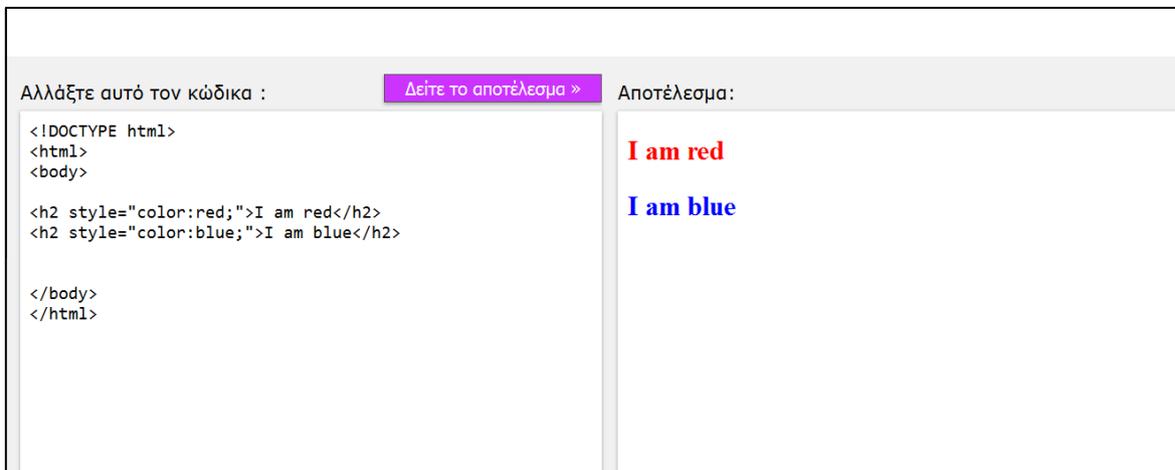
Εικόνα 6.9 Παράδειγμα επεξεργασίας κώδικα

Στις παραπάνω εικόνες περιγράφεται πώς μπορεί κάποιος να δει με ποιον τρόπο εμφανίζεται το στοιχείο της παραγράφου αλλάζοντας το κείμενο που περιέχεται στο στοιχείο < p > .



Εικόνα 6.10 Παράδειγμα HTML

Στην Εικόνα 6.11 παρουσιάζεται το στοιχείο της επικεφαλίδας **<h>** , το οποίο περιέχει την ιδιότητα **style = " color : red ;"** . Προσθέτοντας την ιδιότητα αυτή σε ένα μόνο στοιχείο **<h>** η μορφοποίηση θα γίνει μόνο σε αυτό και όχι στα υπόλοιπα στοιχεία.



Εικόνα 6.11 Παράδειγμα επεξεργασίας κώδικα με CSS inline στυλ



Εικόνα 6.12 Παράδειγμα PHP



Εικόνα 6.13 Παράδειγμα προβολής PHP κώδικα

Το στοιχείο που χρησιμοποιήθηκε δεν είναι το **<form>** αλλά ένα στοιχείο **<iframe>** όπως φαίνεται στο παράδειγμα που ακολουθεί :

`< div class = "container" >`

`< div class = "textareacontainer" >`

`< div class = "textarea" >`

`< div style = "overflow: auto;" >`

```
< div class = "headerText" > PHP κώδικας:</div>
< /div >
< div class = "iframewrapper" >
< iframe id = "iframeSource" frameborder = "0" src = "showphpcode.html " ></iframe>
</div>
</div>
</div>
< div class = "iframecontainer" >
< div class = "iframe" >
< div style = "overflow: auto;" >
< div class = "headerText" > Αποτέλεσμα: </div>
</div>
< div class = "iframewrapper" >
< iframe id = "iframeResult" frameborder = "0" src = "helloworld.html" ></iframe>
</div>
</div>
</div>
</div>
```

HTML PHP SQL

SQL Σεμινάριο
SQL Εισαγωγή
SQL Σύνταξη
SQL Select
SQL Distinct
SQL Where
SQL And & Or
SQL Order By
SQL Insert Into
SQL Update
SQL Delete
SQL Injection
SQL Select top
SQL Like
SQL Wildcards
SQL Between
SQL In

«Προηγούμενο

Επόμενο κεφάλαιο»

SQL SELECT Δήλωση

Η δήλωση SELECT χρησιμοποιείται για την επιλογή δεδομένων από μια βάση δεδομένων.

Η SQL δήλωση SELECT

Η δήλωση SELECT χρησιμοποιείται για την επιλογή δεδομένων από μια βάση δεδομένων.

Το αποτέλεσμα αποθηκεύεται σε έναν πίνακα αποτελεσμάτων (result-set)

SQL SELECT Σύνταξη

```
SELECT column_name,column_name
FROM table_name;
```

και

```
SELECT * FROM table_name;
```

Εικόνα 6.14 Παράδειγμα SQL

SQL Δήλωση:

```
SELECT * FROM Customers;
```

Τροποποιήστε την SQL δήλωση και πατήστε "SQL εκτέλεση" για να δείτε το αποτέλεσμα.

SQL εκτέλεση >

Αποτέλεσμα:

Βάση Δεδομένων

Tablenames	Records
Customers	91
Categories	8
Employees	10
OrderDetails	518
Orders	196
Products	77
Shippers	3
Suppliers	29

Εικόνα 6.15 Επεξεργασία εγγραφών βάσης δεδομένων

Η τελευταία ενότητα της εφαρμογής περιλαμβάνει πίνακες της **βάσης δεδομένων** και ένα πεδίο όπου μπορούν να δηλωθούν οι εντολές προς τη βάση. Μπορεί εύκολα κάποιος να καταλάβει τη λειτουργία της κάθε εντολής βλέποντας το αποτέλεσμα στην οθόνη.

ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ

Castro, E., Hyslop, B. (2012). HTML5 και CSS3 με εικόνες. Μετάφρ. Σαμαράς, Γ., Εκδόσεις Κλειδάριθμος, Αθήνα.

Meloni, J. (2012). Μάθετε PHP, MySQL και Apache. Μετάφρ. Σκουλαρίκης, Φ., Εκδόσεις Γκιούρδας, Αθήνα.

W3schools.com. The world's largest web developer site (1999-2016). *HTML. CSS. JavaScript. SQL. PHP. JQuery. W3.CSS.* Διαθέσιμο στο διαδικτυακό τόπο: <http://www.w3schools.com/>. Ανακτήθηκε στις: 27 Δεκεμβρίου 2015.

Βικιπαίδεια (2011). *Διαδίκτυο*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/Διαδίκτυο> Ανακτήθηκε στις: 20 Δεκεμβρίου 2015.

Βικιπαίδεια (2014). *SQL*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/SQL>. Ανακτήθηκε στις: 28 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *Apache HTTP Εξυπηρετητής* Διαθέσιμο στο διαδικτυακό τόπο: https://el.wikipedia.org/wiki/Apache_HTTP_Εξυπηρετητής .Ανακτήθηκε στις: 27 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *CSS*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/CSS> . Ανακτήθηκε στις: 21 Δεκεμβρίου 20015.

Βικιπαίδεια (2015). *GIMP*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/GIMP>.Ανακτήθηκε στις: 27 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *HTML*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/HTML> . Ανακτήθηκε στις: 21 Δεκεμβρίου 20015.

Βικιπαίδεια (2015). *Javascript*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/JavaScript>. Ανακτήθηκε στις: 23 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *PHP*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/PHP>. Ανακτήθηκε στις: 28 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *XAMPP*. Διαθέσιμο στο διαδικτυακό τόπο: <https://el.wikipedia.org/wiki/XAMPP>. Ανακτήθηκε στις: 23 Δεκεμβρίου 2015.

Βικιπαίδεια (2015). *Εξ αποστάσεως εκπαίδευση*. Διαθέσιμο στο διαδικτυακό τόπο: https://el.wikipedia.org/wiki/%CE%95%CE%BE_%CE%B1%CF%80%CE%BF%CF%83%CF%84%CE%AC%CF%83%CE%B5%CF%89%CF%82_%CE%B5%CE%BA%CF%80%CE%B1%CE%AF%CE%B4%CE%B5%CF%85%CF%83%CE%B7. Ανακτήθηκε στις: 20 Δεκεμβρίου 2015.

Μουρλετίδης, Κ.(2010). *Η Ηλεκτρονική Μάθηση (e-Learning)*. Διαθέσιμο στο διαδικτυακό τόπο: <http://reviews.in.gr/greece/education/article/?aid=1231053712>. Ανακτήθηκε στις: 20 Δεκεμβρίου 2015.