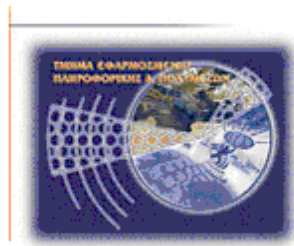




Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων



Πτυχιακή Εργασία

"Σχεδίαση και Υλοποίηση Ιστοσελίδας Τουριστικού Οδηγού της Ρόδου"

Ελευθέριος Νάκκας (ΑΜ: 2171)

Γεώργιος Σταυρής (ΑΜ: 2351)

Επιβλέπων Καθηγητής: Νικόλαος Παπαδάκης

ΗΡΑΚΛΕΙΟ 2013

Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε τον καθηγητή κ. Ν. Παπαδάκη για την πολύτιμη βοήθεια που μας προσέφερε κατά τη διάρκεια της εκπόνησης την πτυχιακής εργασίας.

Abstract

The aim of this report is to implement a website that provides all the necessary information to the visitor on Rhodes and auxiliary functions for a pleasant stay on the island. This application is very useful for visitors, as it offers all necessary function for navigating the island.

The system is based on a database, which contains information about sites, restaurants, and photographs of Rhodes, and all the necessary data to be processed, such as booking a hotel room and ticket booking. Apart from the operation of reservation may make and cancel it.

The implementation of the current system was based on the technology of SQL, for creating and managing the database, as well as the technology of HTML, PHP and JavaScript, for developing the website.

Σύνοψη

Σκοπός της εργασίας αυτής είναι η υλοποίηση μιας ιστοσελίδας που παρέχει όλες τις απαραίτητες πληροφορίες στον επισκέπτη της Ρόδου καθώς και βοηθητικές λειτουργίες για μια ευχάριστη διαμονή στο νησί. Η εφαρμογή αυτή είναι πολύ χρήσιμη για τον επισκέπτη, αφού προσφέρει όλες τις απαραίτητες λειτουργίες για την πλοήγηση του στο νησί.

Το σύστημα στηρίζεται σε μία βάση δεδομένων, που περιέχει την πληροφορία αξιοθέατων, εστιατορίων και φωτογραφιών της Ρόδου, καθώς και όλα τα απαραίτητα δεδομένα για την διεκπεραίωση λειτουργιών, όπως την κράτηση δωματίου σε ξενοδοχείο και την κράτηση εισιτηρίου. Εκτός από τη λειτουργία της κράτησης μπορεί να γίνει και ακύρωση της.

Οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίηση του παραπάνω συστήματος είναι η SQL για τη δημιουργία και διαχείριση της βάσης δεδομένων, καθώς επίσης και η HTML, PHP και η JavaScript για την ανάπτυξη της ιστοσελίδας.

Πίνακας Περιεχομένων

ΕΥΧΑΡΙΣΤΙΕΣ	3
ABSTRACT	5
ΣΥΝΟΨΗ	7
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	9
ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ	11
ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ	12
ΠΕΡΙΛΗΨΗ	12
ΚΙΝΗΤΡΟ ΓΙΑ ΤΗ ΔΙΕΞΑΓΩΓΗ ΤΗΣ ΕΡΓΑΣΙΑΣ	12
ΣΚΟΠΟΣ ΚΑΙ ΣΤΟΧΟΙ ΤΗΣ ΕΡΓΑΣΙΑΣ	12
ΔΟΜΗ ΤΗΣ ΕΡΓΑΣΙΑΣ	13
ΚΕΦΑΛΑΙΟ 2. ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ	14
2.1 ΣΤΑΤΙΚΕΣ ΙΣΤΟΣΕΛΙΔΕΣ	14
2.2 ΔΥΝΑΜΙΚΕΣ ΙΣΤΟΣΕΛΙΔΕΣ	14
2.3 HTML	15
2.3.1 ΣΤΟΙΧΕΙΑ	16
2.3.2 ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ	17
2.3.3 ΔΗΛΩΣΗ ΤΥΠΟΥ ΕΓΓΡΑΦΟΥ	17
2.4 CSS	17
2.5 PHP	18
2.5.1 ΣΧΟΛΙΑ	19
2.5.2 ΜΕΤΑΒΛΗΤΕΣ	19
2.5.3 ΣΤΑΘΕΡΕΣ	19
2.5.4 ΤΕΛΕΣΤΕΣ	19
2.5.5 GET ΚΑΙ POST	20
2.5.6 INCLUDES	21
2.5.7 HTML ΚΑΙ PHP	22
2.5.8 PHP ΚΑΙ JAVASCRIPT	23
2.6 JAVASCRIPT	24
2.7 ΕΡΓΑΛΕΙΑ ΑΝΑΠΤΥΞΗΣ ΔΥΝΑΜΙΚΩΝ ΕΦΑΡΜΟΓΩΝ	24

2.7.1 EASYRHP	24
2.7.2 ΡΗΡΜΥADMIN	28
2.7.3 ΕΞΥΠΗΡΕΤΗΤΗΣ ΑΡΑΧΕ	29
2.7.4 DREAMWEAVER	30

ΚΕΦΑΛΑΙΟ 3. ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ **34**

3.1 ΣΧΕΔΙΑΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	34
3.1.1 ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΑ ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΤΟΥΣ	34
3.1.2 ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ	35
3.2 ΔΟΜΗΜΕΝΗ ΓΛΩΣΣΑ ΕΠΕΡΩΤΗΣΕΩΝ (STRUCTURED QUERY LANGUAGE, SQL)	36
3.2.1 ΒΑΣΙΚΗ ΔΟΜΗ	37
3.2.2 ΠΡΑΞΕΙΣ ΣΥΝΟΛΩΝ	38
3.2.3 ΣΥΝΟΠΤΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ	39
3.2.4 ΚΕΝΕΣ ΤΙΜΕΣ	40
3.2.5 ΤΡΟΠΟΠΟΙΗΣΗ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	41

ΚΕΦΑΛΑΙΟ 4. ΑΝΑΛΥΣΗ ΣΥΣΤΗΜΑΤΟΣ **44**

4.1 ΠΕΡΙΓΡΑΦΗ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	44
4.1.1 ΔΙΑΓΡΑΜΜΑ ΟΝΤΟΤΗΤΩΝ ΣΧΕΣΕΩΝ	44
4.1.2 ΣΧΕΣΕΙΣ ΚΑΙ ΑΝΑΛΥΣΗ ΤΩΝ ΣΧΕΣΕΩΝ	45
4.1.3 ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ	47
4.2 ΠΕΡΙΓΡΑΦΗ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	49
4.3 ΥΛΟΠΟΙΗΣΗ ΚΑΙ ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ ΤΗΣ ΕΦΑΡΜΟΓΗΣ	59
4.3.1 ΔΗΜΙΟΥΡΓΙΑ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	59
4.3.2 SQL ΕΠΕΡΩΤΗΣΕΙΣ ΣΤΗ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ ΡΗΡ	67

ΚΕΦΑΛΑΙΟ 5. ΑΠΟΤΕΛΕΣΜΑΤΑ **72**

5.1 ΣΥΜΠΕΡΑΣΜΑΤΑ	72
5.2 ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ ΚΑΙ ΕΠΕΚΤΑΣΕΙΣ	72

ΒΙΒΛΙΟΓΡΑΦΙΑ **73**

ΠΑΡΑΡΤΗΜΑ: ΔΙΑΦΑΝΕΙΕΣ ΠΑΡΟΥΣΙΑΣΗΣ **74**

Πίνακας Εικόνων

ΕΙΚΟΝΑ 1 - ΕΓΚΑΤΑΣΤΑΣΗ EASYRHP	25
ΕΙΚΟΝΑ 2 - ΕΠΙΛΟΓΗ ΦΑΚΕΛΟΥ ΕΓΚΑΤΑΣΤΑΣΗΣ	25
ΕΙΚΟΝΑ 3 - ΣΥΝΤΟΜΕΥΣΗ EASYRHP	25
ΕΙΚΟΝΑ 4 - ΔΙΑΧΕΙΡΙΣΗ ΕΡΓΑΣΙΩΝ	26
ΕΙΚΟΝΑ 5 - ΠΑΡΑΘΥΡΟ ΕΡΓΑΣΙΩΝ EASYRHP (Α)	26
ΕΙΚΟΝΑ 6 - ΠΑΡΑΘΥΡΟ ΕΡΓΑΣΙΩΝ EASYRHP (Β)	27
ΕΙΚΟΝΑ 7 - ΦΑΚΕΛΟΣ WWW	27
ΕΙΚΟΝΑ 8 - ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΔΙΑΧΕΙΡΙΣΗΣ EASYRHP	28
ΕΙΚΟΝΑ 9 - ΑΡΧΙΚΗ ΣΕΛΙΔΑ ΤΟΥ DREAMWEAVER	32
ΕΙΚΟΝΑ 10 - ΔΙΕΠΑΦΗ ΕΡΓΑΣΙΑΣ ΤΟΥ DREAMWEAVER	33
ΕΙΚΟΝΑ 11 - ΑΡΧΙΚΗ ΣΕΛΙΔΑ	49
ΕΙΚΟΝΑ 12 - ΠΡΟΒΟΛΗ ΛΙΣΤΑΣ ΚΑΤΑΧΩΡΗΜΕΝΩΝ ΞΕΝΟΔΟΧΕΙΩΝ	50
ΕΙΚΟΝΑ 13 - ΠΡΟΒΟΛΗ ΦΩΤΟΓΡΑΦΙΩΝ ΞΕΝΟΔΟΧΕΙΟΥ	50
ΕΙΚΟΝΑ 14 - ΠΡΟΒΟΛΗ ΦΩΤΟΓΡΑΦΙΑΣ	51
ΕΙΚΟΝΑ 15 - ΠΡΟΒΟΛΗ ΑΞΙΟΘΕΑΤΩΝ ΡΟΔΟΥ	51
ΕΙΚΟΝΑ 16 - ΠΡΟΒΟΛΗ ΛΙΣΤΑΣ ΚΑΤΑΧΩΡΗΜΕΝΩΝ ΕΣΤΙΑΤΟΡΙΩΝ	52
ΕΙΚΟΝΑ 17 - ΠΡΟΒΟΛΗ ΧΑΡΤΗ ΤΗΣ ΡΟΔΟΥ ΜΕΣΩ GOOGLE MAPS	53
ΕΙΚΟΝΑ 18 - ΠΡΟΒΟΛΗ ΓΡΑΦΕΙΩΝ	53
ΕΙΚΟΝΑ 19 - ΠΡΟΒΟΛΗ ΓΡΑΦΕΙΩΝ ΕΝΟΙΚΙΑΖΟΜΕΝΩΝ ΑΥΤΟΚΙΝΗΤΩΝ ΚΑΙ ΜΗΧΑΝΩΝ	54
ΕΙΚΟΝΑ 20 - ΠΡΟΒΟΛΗ ΔΙΑΘΕΣΙΜΩΝ ΟΧΗΜΑΤΩΝ ΓΡΑΦΕΙΟΥ ΕΝΟΙΚΙΑΖΟΜΕΝΩΝ	54
ΕΙΚΟΝΑ 21 - ΠΡΟΒΟΛΗ ΚΑΤΑΧΩΡΗΜΕΝΩΝ ΤΑΞΙΔΙΩΤΙΚΩΝ ΓΡΑΦΕΙΩΝ	55
ΕΙΚΟΝΑ 22 - ΠΡΟΒΟΛΗ ΔΙΑΘΕΣΙΜΩΝ ΤΑΞΙΔΙΩΤΙΚΩΝ ΠΑΚΕΤΩΝ ΣΥΓΚΕΚΡΙΜΕΝΟΥ ΤΑΞΙΔΙΩΤΙΚΟΥ ΓΡΑΦΕΙΟΥ	55
ΕΙΚΟΝΑ 23 - ΦΟΡΜΑ ΣΥΜΠΛΗΡΩΣΗΣ ΣΤΟΧΕΙΩΝ ΓΙΑ ΚΡΑΤΗΣΗ	56
ΕΙΚΟΝΑ 24 - ΠΡΟΒΟΛΗ ΔΙΑΘΕΣΙΜΩΝ ΔΩΜΑΤΙΩΝ ΓΙΑ ΚΡΑΤΗΣΗ	57
ΕΙΚΟΝΑ 25 - ΠΡΟΒΟΛΗ ΚΩΔΙΚΟΥ ΑΡΙΘΜΟΥ ΚΡΑΤΗΣΗΣ	57
ΕΙΚΟΝΑ 26 - ΠΡΟΒΟΛΗ ΦΩΤΟΓΡΑΦΙΩΝ	58
ΕΙΚΟΝΑ 27 - ΠΡΟΒΟΛΗ ΦΩΤΟΓΡΑΦΙΑΣ	58
ΕΙΚΟΝΑ 28 - ΠΙΝΑΚΑΣ ΚΣΕΝΟΔΟΧΕΙΟ	59
ΕΙΚΟΝΑ 29 - ΠΙΝΑΚΑΣ ΚΡΑΤΗΣΗ	60
ΕΙΚΟΝΑ 30 - ΠΙΝΑΚΑΣ ΔΩΜΑΤΙΟ	60
ΕΙΚΟΝΑ 31 - ΠΙΝΑΚΑΣ ΕΙΚΟΝΕΣ_ΚΣΕΝ	61
ΕΙΚΟΝΑ 32 - ΠΙΝΑΚΑΣ ΧΑΡΤΗΣ	61
ΕΙΚΟΝΑ 33 - ΠΙΝΑΚΑΣ ΑΚΣΙΟΕΑΤΟ	62
ΕΙΚΟΝΑ 34 - ΠΙΝΑΚΑΣ ΔΡΟΜΟΛΟΓΙΟ	63
ΕΙΚΟΝΑ 35 - ΠΙΝΑΚΑΣ ΕΣΤΙΑΤΟΡΙΟ	63
ΕΙΚΟΝΑ 36 - ΠΙΝΑΚΑΣ ΓΡΑΦΕΙΟ_ΕΝΟΙΚ	64
ΕΙΚΟΝΑ 37 - ΠΙΝΑΚΑΣ ΑΥΤΟΚΙΝΗΤΟ	65
ΕΙΚΟΝΑ 38 - ΠΙΝΑΚΑΣ ΜΗΧΑΝΑΚΙ	65
ΕΙΚΟΝΑ 39 - ΠΙΝΑΚΑΣ ΕΚΔΡΟΜΙΚΟ_GRAF	66
ΕΙΚΟΝΑ 40 - ΠΙΝΑΚΑΣ ΕΚΔΡΟΜΗ	67

Κεφάλαιο 1. Εισαγωγή

Περίληψη

Σε αυτήν την πτυχιακή εργασία σχεδιάστηκε και υλοποιήθηκε ένας ηλεκτρονικός ταξιδιωτικός οδηγός της Ρόδου. Το σύστημα αυτό έχει στόχο να ενημερώσει και να εξυπηρετήσει τον επισκέπτη της Ρόδου με πολλούς τρόπους. Επιγραμματικά, παρέχεται η δυνατότητα προβολής των διαθέσιμων καταλυμάτων, γραφείων ενοικιάσεων, δρομολόγια μέσω μαζικής μεταφοράς, η ηλεκτρονική κράτηση δωματίου ή εισιτηρίου για κάποιο μέσο μεταφοράς καθώς επίσης και η προβολή πληροφορίας σχετική με εστιατόρια, αξιοθέατα κλπ.

Για την υλοποίηση της εφαρμογής, χρησιμοποιήθηκε η γλώσσα σχεσιακών συστημάτων MySQL, το εργαλείο XAMPP και η scripting γλώσσα PHP.

Κίνητρο για τη Διεξαγωγή της Εργασίας

Η Ρόδος, ή το νησί των ιπποτών, αποτελεί τουριστικό προορισμό Ελλήνων και ξένων. Η ύπαρξη ενός τουριστικού οδηγού που πληροφορεί τον επισκέπτη για τη διαμονή, τη δρομολόγια, τις παραλίες και τα αξιοθέατα κρίνεται απαραίτητη. Ένας ηλεκτρονικός τουριστικός οδηγός όχι μόνο ενημερώνει τον επισκέπτη του νησιού για τις επιλογές που προσφέρονται αλλά αποτελεί διαφήμιση και προώθηση του σε μελλοντικούς επισκέπτες. Τα τελευταία χρόνια, όλο και περισσότεροι ιστότοποι λειτουργούν στον Παγκόσμιο Ιστό, με βασικό σκοπό την προβολή κάποιου τουριστικού προορισμού.

Στα πλαίσια των παραπάνω αναγκών, δημιουργήθηκε η συγκεκριμένη ιστοσελίδα. Ο συγκεκριμένος τουριστικός οδηγός παρέχει τη δυνατότητα της ηλεκτρονικής κράτησης δωματίων και εισιτηρίων στους χρήστες του. Επίσης, προτείνει εστιατόρια, αξιοθέατα και παραλίες του νησιού. Τέλος, παρέχεται φωτογραφικό υλικό της Ρόδου και πληροφορία για γραφεία ενοικιάσεων και εκδρομικά γραφεία.

Σκοπός και Στόχοι της Εργασίας

Σκοπός της εργασίας αυτής είναι η υλοποίηση μιας ιστοσελίδας που παρέχει όλες τις απαραίτητες πληροφορίες στον επισκέπτη της Ρόδου καθώς και βοηθητικές λειτουργίες για μια ευχάριστη διαμονή στο νησί.

Παρακάτω ακολουθούν συνοπτικά οι λειτουργίες που παρέχει το σύστημα αυτό στο χρήστη του:

1. Προβολή ξενοδοχείων του νησιού
2. Προβολή γραφείων ενοικιάσεων του νησιού
3. Προβολή αξιοθέατων και παραλιών του νησιού
4. Προβολή δρομολογίων μέσω μαζικής μεταφοράς
5. Προβολή εκδρομικών γραφείων του νησιού
6. Προβολή εστιατορίων του νησιού
7. Προβολή φωτογραφικού υλικού και χάρτη της Ρόδου
8. Αναζήτηση και κράτηση δωματίου με συγκεκριμένα κριτήρια
9. Αναζήτηση και κράτηση εισιτηρίου αεροπλάνου/πλοίου/λεωφορείου με συγκεκριμένα κριτήρια
10. Ακύρωση κράτησης

Δομή της Εργασίας

Η αναφορά αποτελείται από τα επόμενα κεφάλαια.

Στο Κεφάλαιο 2, περιγράφεται το θεωρητικό πλαίσιο στο οποίο βασίστηκε η δημιουργία της ιστοσελίδας. Στο Κεφάλαιο 3, περιγράφονται τα εργαλεία που αξιοποιήθηκαν για την υλοποίηση του συστήματος. Το Κεφάλαιο 4 περιέχει σενάρια χρήσης του συστήματος καθώς και ενδεικτικά στιγμιότυπα από αυτά (screenshots). Τέλος, το Κεφάλαιο 5 περιλαμβάνει τα συμπεράσματα από την υλοποίηση και εκτέλεση του συστήματος και πιθανές μελλοντικές επεκτάσεις.

Κεφάλαιο 2. Θεωρητικό Πλαίσιο

2.1 Στατικές Ιστοσελίδες

Στατική ιστοσελίδα (static web-page ή flat web-page) ονομάζεται μια ιστοσελίδα της οποίας το περιεχόμενο μεταφέρεται στον χρήστη ακριβώς στην μορφή που είναι αποθηκευμένο στον εξυπηρετητή ιστοσελίδων (web server), σε αντίθεση με τις δυναμική ιστοσελίδα όπου το περιεχόμενο δημιουργείται από μια εφαρμογή η οποία εκτελείται στον εξυπηρετητή ιστοσελίδων.

Τα περιεχόμενα μιας στατικής ιστοσελίδας εμφανίζονται με την ίδια μορφή σε όλους του χρήστες με την μορφή που είναι αποθηκευμένα στο σύστημα αρχείων του εξυπηρετητή ιστοσελίδων. Οι στατικές ιστοσελίδες είναι αποθηκευμένες συνήθως σε μορφή HTML και μεταφέρονται χρησιμοποιώντας το πρωτόκολλο HTTP.

Στα πλεονεκτήματα των στατικών σελίδων περιλαμβάνεται το γεγονός ότι η σελίδα μπορεί να βρίσκεται με εύκολο τρόπο κλωνοποιημένη σε περισσότερους από ένα εξυπηρετητές. Επίσης δε χρειάζεται ειδικό λογισμικό στον εξυπηρετητή ιστοσελίδων για τη δημοσίευση στατικών σελίδων. Τέλος, η σελίδα μπορεί να είναι διαθέσιμη στον φυλλομετρητή κατευθείαν από ένα αποθηκευτικό μέσο όπως ένα CD-ROM ή ένα USB stick μνήμης χωρίς να χρειάζεται να διαμεσολαβήσει ένας εξυπηρετητής ιστοσελίδων με κατάλληλο λογισμικό χειρισμό ιστοσελίδων (για παράδειγμα οι δυναμικές σελίδες σε γλώσσα PHP θέλουν ένα διεργασμένο PHP ο οποίος τρέχει στον εξυπηρετητή ιστοσελίδων και δημιουργεί δυναμικά τη σελίδα πριν αυτή αποσταλεί στον φυλλομετρητή του χρήστη).

Στον αντίποδα, ένα χαρακτηριστικό μειονέκτημα των στατικών ιστοσελίδων είναι ότι δεν είναι εύκολη η διαδραστικότητα με τον χρήστη και η διαχείριση μεγάλου αριθμού στατικών ιστοσελίδων δεν είναι εύκολη χωρίς αυτόματα εργαλεία.

2.2 Δυναμικές Ιστοσελίδες

Δυναμική ιστοσελίδα (dynamic web page) είναι μια ιστοσελίδα η οποία δημιουργείται δυναμικά την στιγμή της πρόσβασης σε αυτή ή την στιγμή που ο χρήστης αλληλεπιδρά με τον εξυπηρετητή ιστοσελίδων. Οι δυναμικές ιστοσελίδες θεωρούνται δομικό στοιχείο της νέας γενιάς του παγκόσμιου ιστού (Web 2.0) όπου η πληροφορία διαμοιράζεται σε πολλαπλές ιστοσελίδες.

Η δυναμική ιστοσελίδα μπορεί να δημιουργείται δυναμικά από ένα σενάριο εντολών, το οποίο εκτελείται τοπικά στο πελάτη ή στον εξυπηρετητή ή και στον πελάτη και στον εξυπηρετητή.

Είναι δύσκολο να πει κάποιος με ακρίβεια πότε οι "δυναμικές ιστοσελίδες" ξεκίνησαν αλλά η γενικότερη ιδέα ξεκίνησε μετά την εξάπλωση και ανάπτυξη των ιστοσελίδων. Το πρωτόκολλο HTTP άρχισε να χρησιμοποιείται από το 1990 και η γλώσσα HTML ξεκίνησε να χρησιμοποιείται για τον σχεδιασμό ιστοσελίδων από το 1996. Η εξάπλωση χρήσης των φυλλομετρητών ξεκίνησε το 1993 με τον φυλλομετρητή Mosaic. Η ιδέα της δημιουργίας δυναμικών ιστοσελίδων υπήρχε και πριν την ανάπτυξη των ιστοσελίδων. Για παράδειγμα πριν το 1990 υπήρχαν "Βάσεις Δεδομένων" (Bulletin Board Systems) στα οποία οι χρήστες συνδέονταν μέσω μόντεμ και το περιεχόμενο που έβλεπαν δημιουργούνταν δυναμικά.

Η σελίδα συνήθως περιέχει ένα σενάριο εντολών το οποίο ενσωματώνει μια διαδραστικότητα με τον χρήστη (η διαδραστικότητα γίνεται με χρήση πληκτρολογίου - ποντικιού). Το περιεχόμενο της ιστοσελίδας δημιουργείται δυναμικά στον τοπικό υπολογιστή μετά την εκτέλεση του σεναρίου εντολών το οποίο λαμβάνεται από τον απομακρυσμένο εξυπηρετητή.

Για παράδειγμα η γλώσσα σεναρίου εντολών JavaScript ή ActionScript (γλώσσα σεναρίων της Macromedia - σήμερα είναι η εταιρία Adobe Systems) χρησιμοποιείται για την τοπική δημιουργία δυναμικών ιστοσελίδων τεχνολογίας DHTML ή Flash οι οποίες συχνά χρησιμοποιούνται για την ενσωμάτωση ήχων, κινούμενων εικόνων ή αλλαγές και διαμόρφωση του κείμενου της σελίδας. Η γλώσσα σεναρίου η οποία εκτελείται τοπικά μπορεί να καλεί απομακρυσμένα σενάρια εντολών στον εξυπηρετητή ιστοσελίδων χρησιμοποιώντας τεχνολογίες όπως hidden Frame, XMLHttpRequest ή άλλες υπηρεσίες ιστού (web services).

Η πρώτη "ευρέως διαδεδομένη" έκδοση της γλώσσας σεναρίου εντολών JavaScript ενσωματώθηκε το 1996 στο φυλλομετρητή Netscape 3.

Μια δυναμική ιστοσελίδα συνήθως δημιουργείται από ένα λογισμικό το οποίο τρέχει στον απομακρυσμένο εξυπηρετητή ιστοσελίδων. Ο χρήστης-πελάτης επισκέπτεται τον εξυπηρετητή ιστοσελίδων ο οποίος δημιουργεί δυναμικά την ιστοσελίδα την οποία επιστρέφει σε μορφή HTML στον πελάτη. Τέτοιες τεχνολογίες για την δημιουργία δυναμικών ιστοσελίδων στον εξυπηρετητή είναι η τεχνολογία ASP, ColdFusion, Perl, PHP, WebDNA και άλλες. Οι γλώσσες αυτές συνήθως (με εξαίρεση την ASP.NET και την JSP) χρησιμοποιούν ένα σενάριο εντολών CGI κατά την δυναμική δημιουργία των σελίδων.

Πολλές φορές οι σελίδες είναι αποθηκευμένες στατικά σε κρυφή-προσωρινή μνήμη και αποστέλλονται κατευθείαν στο πελάτη. Με αυτό το τρόπο αποφεύγεται η υπερφόρτωση του εξυπηρετητή ιστοσελίδων και καλείται η μηχανή δημιουργία δυναμικών ιστοσελίδων μόνο όταν το αποθηκευμένο περιεχόμενο έχει αλλάξει.

Η τεχνολογία AJAX χρησιμοποιεί σενάρια εντολών και στον πελάτη και στον απομακρυσμένο εξυπηρετητή για την δημιουργία της δυναμικής ιστοσελίδας. Στην τεχνολογία AJAX γίνεται ανταλλαγή περιεχομένου και δεδομένων από τον υπολογιστή του πελάτη και του απομακρυσμένου εξυπηρετητή και η σελίδα διαμορφώνεται από το σενάριο εντολών το οποίο τρέχει τοπικά στο πελάτη. Το πλεονέκτημα της τεχνολογίας αυτής είναι ότι περιορίζεται η χρήση του εξυπηρετητή ιστοσελίδων και αποστέλλεται μόνο το περιεχόμενο. Για παράδειγμα οι χάρτες της Google είναι ένα παράδειγμα δικτυακής εφαρμογής η οποία χρησιμοποιεί την τεχνολογία AJAX.

2.3 HTML

Η HTML (HyperText Markup Language, 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, οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ο σκοπός ενός web browser είναι να διαβάσει τα έγγραφα HTML και 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, και 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.

2.3.1 Στοιχεία

Τα έγγραφα HTML αποτελούνται από στοιχεία HTML τα οποία στην πιο γενική μορφή τους έχουν τρία συστατικά: ένα ζεύγος από ετικέτες, την «ετικέτα εκκίνησης» και την «ετικέτα τερματισμού», μερικές ιδιότητες μέσα στην ετικέτα εκκίνησης, και τέλος το κείμενο ή το γραφικό περιεχόμενο μεταξύ των ετικετών, το οποίο μπορεί να περιλαμβάνει και άλλα στοιχεία εμφωλευμένα μέσα του. Το στοιχείο HTML μπορεί να είναι οτιδήποτε ανάμεσα στις ετικέτες εκκίνησης και τερματισμού. Τέλος, κάθε ετικέτα περικλείεται σε σύμβολα «μεγαλύτερο από» και «μικρότερο από», δηλαδή < και >.

Επομένως, η γενική μορφή ενός στοιχείου HTML είναι: `<tag attribute1="value1" attribute2="value2">content</tag>`. Μερικά στοιχεία HTML περιγράφονται ως *άδεια στοιχεία*, έχουν τη μορφή `<tag attribute1="value1" attribute2="value2" >`, και δεν έχουν καθόλου περιεχόμενο. Το όνομα κάθε στοιχείου HTML είναι το ίδιο όνομα που χρησιμοποιείται στις αντίστοιχες ετικέτες. Το όνομα της ετικέτας τερματισμού ξεκινά με μια κάθετο «/», η οποία παραλείπεται στα άδεια στοιχεία. Τέλος, αν δεν αναφέρονται ρητά οι ιδιότητες ενός στοιχείου, τότε χρησιμοποιούνται οι προεπιλογές σε κάθε περίπτωση.

Υπάρχουν διάφοροι τύπου στοιχείων στην HTML.

Η δομική σήμανση περιγράφει τον σκοπό του κειμένου

Για παράδειγμα, το `<h2>Golf</h2>` σημαίνει ότι η λέξη «Golf» θα είναι μια επικεφαλίδα δεύτερου επιπέδου. Η δομική σήμανση δεν εμπεριέχει συγκεκριμένο τρόπο εμφάνισης, αλλά οι περισσότεροι browser έχουν δικό τους προκαθορισμένο τρόπο για τη μορφοποίηση όλων των στοιχείων. Το περιεχόμενο μπορεί να μορφοποιηθεί οπτικά με τη χρήση των CSS.

Η σήμανση οπτικής μορφοποίησης περιγράφει την εμφάνιση του κειμένου, άσχετα από τον σκοπό του.

Για παράδειγμα το `έντονο κείμενο` υποδεικνύει ότι οι συσκευές που θα εμφανίσουν το κείμενο θα το κάνουν έντονο, αλλά δεν λέει τι θα κάνουν με το κείμενο οι συσκευές που δεν μπορούν να το εμφανίσουν, όπως για παράδειγμα οι συσκευές φωνητικής ανάγνωσης σελίδων. Τόσο στην περίπτωση του `έντονο` όσο και του `<i>πλάγιο</i>`, υπάρχουν άλλα στοιχεία τα οποία μπορεί να οπτικά να εμφανίζονται ίδια, αλλά είναι πιο *σημαντικά*, όπως τα `σημαντικό κείμενο` και `κείμενο με έμφαση` αντίστοιχα. Είναι ευκολότερα να ερμηνεύσει ένας φωνητικός browser τα τελευταία δύο στοιχεία. Ωστόσο, δεν είναι ισότιμα με τα αντίστοιχα στοιχεία οπτικής μορφοποίησης. Για παράδειγμα ένα πρόγραμμα φωνητικής ανάγνωσης της σελίδας δεν θα πρέπει να προφέρει με έμφαση τον τίτλο ενός βιβλίου, αλλά όταν το ίδιο περιεχόμενο εμφανίζεται σε μια οθόνη τότε ο τίτλος θα είναι με πλάγια γράμματα. Τα περισσότερα στοιχεία οπτικής μορφοποίησης έχουν θεωρούνται ξεπερασμένα μετά την προδιαγραφή HTML 4.0, και έχουν αντικατασταθεί από τα CSS.

Η σήμανση υπερκειμένου κάνει μερικά τμήματα ενός εγγράφου να συνδέουν με άλλα έγγραφα.

Τα στοιχεία `anchor` δημιουργεί έναν υπερσύνδεσμο στο έγγραφο, και η ιδιότητα `href` ορίζει τον στόχο του συνδέσμου. Για παράδειγμα, η σήμανση HTML `Βικιπαίδεια` εμφανίζει το κείμενο «Wikipedia» ως υπερσύνδεσμο. Για την εμφάνιση μιας εικόνας ως συνδέσμου, μπορεί να εισαχθεί ένα στοιχείο `img` ως περιεχόμενο του στοιχείου `a`. Όπως και το `br`, το `img` είναι άδειο στοιχείο. Έχει ιδιότητες αλλά δεν έχει περιεχόμενο, ούτε ετικέτα τερματισμού: ``.

2.3.2 Τύποι δεδομένων

Στην HTML ορίζονται μερικοί τύποι δεδομένων για το περιεχόμενο των στοιχείων, όπως σενάρια εντολών ή stylesheet, και μια πληθώρα τύπων για τις τιμές των ιδιοτήτων, συμπεριλαμβανομένων των ID, των name, των URI, διαφόρων αριθμών και μονάδων μήκους, γλωσσών, τύπων αρχείων πολυμέσων, χρωμάτων, κωδικοποιήσεων χαρακτήρων, ημερομηνιών κλπ.

2.3.3 Δήλωση τύπου εγγράφου

Τα έγγραφα HTML πρέπει να αρχίζουν με μια Δήλωση τύπου εγγράφου (Document Type Declaration, ανεπίσημα λέγεται και «doctype»). Αυτή η δήλωση βοηθά τους browser να καταλάβουν πώς πρέπει να διαβάσουν το περιεχόμενο του εγγράφου και πώς να το παρουσιάσουν μετά, και ιδιαίτερα όταν χρησιμοποιείται το quirks mode.

Ο αρχικός σκοπός του doctype ήταν να επιτρέπει την ανάλυση και επιβεβαίωση των εγγράφων HTML από εργαλεία SGML τα οποία ήταν βασισμένα στο Document Type Definition (DTD). Το DTD στο οποίο αναφέρεται το DOCTYPE περιέχει μια γραμματική σε γλώσσα μηχανής, η οποία καθορίζει τι επιτρέπεται και τι απαγορεύεται να υπάρχει μέσα στο έγγραφο. Οι browser, από την άλλη, δεν υλοποιούν την HTML ως εφαρμογή της SGML και συνεπώς δεν διαβάζουν το DTD. Στην HTML5 δεν ορίζεται κανένα DTD, λόγω εγγενών τεχνολογικών περιορισμών. Έτσι το doctype<!doctype html> δεν αναφέρεται σε κανένα.

2.4 CSS

Η CSS (*Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ*) ή (αλληλουχία φύλλων στύλ) 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 δηλαδή 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 ιστότοπου. 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.

Η αλληλουχία εφαρμογής των φύλλων στύλ

Για ένα έγγραφο πχ xhtml θα υπάρχουν παραπάνω από ένα φύλλα στυλ τα οποία περιέχουν δηλώσεις για την εμφάνιση ενός συγκεκριμένου στοιχείου. Το Φύλλο στύλ που εφαρμόζεται σε ένα έγγραφο μπορεί να προέρχεται από :

- το συγγραφέα μιας ιστοσελίδας
- το χρήστη του πλοηγού
- τον ίδιο τον πλοηγό, αν έχει το δικό του προκαθορισμένο φύλλο στυλ .

Συνεπώς για ένα xhtml στοιχείο θα υπάρχουν παραπάνω από μια δηλώσεις που πιθανόν να είναι συγκρουόμενες. Το πρότυπο css για να επιλύσει παρόμοιες συγκρούσεις έχει καθορίσει μια αλληλουχία-σειρά στην οποία θα μπουν αυτές οι δηλώσεις και με βάση την οποία θα επιλεγεί πχ η δήλωση που είναι πρώτη στη σειρά.

Ο αλγόριθμος δημιουργίας αυτής της σειράς-αλληλουχίας είναι ο ακόλουθος:

1. Βρες όλες τις δηλώσεις που εφαρμόζονται στο στοιχείο που μας ενδιαφέρει. Οι δηλώσεις εφαρμόζονται στο στοιχείο αν ο επιλογέας του το επιλέξει (ταιριάζει με αυτό).
2. Ταξινόμησε με βάση τη σημασία (κανονική ή σημαντική) και προέλευση (συγγραφέας , χρήστη ή πλοηγός χρήστη). Με αύξουσα σειρά προτεραιότητας:
 - a. Δηλώσεις πλοηγού χρήστη

- b. Κανονικές δηλώσεις χρήστη
- c. Κανονικές δηλώσεις συγγραφέα
- d. Σημαντικές δηλώσεις συγγραφέα
- e. Σημαντικές δηλώσεις χρήστη

3. Ταξινόμησε τις δηλώσεις ίδιας σημασίας και προέλευσης με κριτήριο την εξειδίκευση του επιλογέα: οι πιο εξειδικευμένοι επιλογείς υπερσχύουν των πιο γενικών. Τα ψευδό-στοιχεία και οι ψευδο-κλάσεις λογαριάζονται σαν κανονικά στοιχεία και κλάσεις αντίστοιχα.
4. Τέλος ταξινόμησε ανάλογα με τη σειρά καθορισμού: αν δύο δηλώσεις έχουν το ίδιο βάρος , προέλευση και εξειδίκευση , αυτή που προσδιορίστηκε τελευταία επικρατεί. Οι δηλώσεις σε εισαγόμενα φύλλα στυλ θεωρούνται ότι δηλώνονται πριν από τις δηλώσεις στο ίδιο το φύλλο στυλ .

Αφού λοιπόν προκύψει μια σειρά-αλληλουχία κανόνων εμφάνισης που αφορούν το ίδιο στοιχείο θα επιλεγεί προς εφαρμογή (για την αποφυγή συγκρούσεων) η δήλωση που θα είναι τελευταία στην σειρά που αναλύθηκε πιο πάνω.

2.5 PHP

Γλώσσα προγραμματισμού για τη δημιουργία σελίδων web με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία από συμβατό web server (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 κώδικα PHP.

Η ιστορία της PHP ξεκινά από το 1995, 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, βασιζόμενη αυτή τη φορά στη γλώσσα C και αριθμώντας περισσότερα από 50.000 web sites που τη χρησιμοποιούσαν, ενώ αργότερα την ίδια χρονιά οι Andi Gutmans και Zeev Suraski ξαναέγραψαν τη γλώσσα από την αρχή, βασιζόμενοι όμως αρκετά στην PHP/FI 2.0. Έτσι η PHP έφθασε στην έκδοση 3.0 η οποία θύμιζε περισσότερο τη σημερινή μορφή της. Στη συνέχεια, οι Zeev και Andi δημιούργησαν την εταιρεία Zend (από τα αρχικά των ονομάτων τους), η οποία συνεχίζει μέχρι και σήμερα την ανάπτυξη και εξέλιξη της γλώσσας PHP. Ακολούθησε το 1998 η έκδοση 4 της PHP, τον Ιούλιο του 2004 διατέθηκε η έκδοση 5, ενώ αυτή τη στιγμή έχουν ήδη διατεθεί και τα πρώτα snapshots της επερχόμενης PHP 6 , για οποιονδήποτε προγραμματιστή θέλει να τη χρησιμοποιήσει.

Σήμερα περισσότερα από 16.000.000 web sites, ποσοστό μεγαλύτερο από το 35% των ιστοσελίδων του Διαδικτύου, χρησιμοποιούν scripts γραμμένα με τη γλώσσα PHP, ενώ το υπόλοιπο 65% το μοιράζονται στατικές σελίδες HTML και όλες οι άλλες γλώσσες προγραμματισμού. Πρόκειται για μια εξέλιξη που ο ίδιος ο Rasmus Lerdorf σε πρόσφατη συνέντευξή του δήλωσε ότι δεν περίμενε όταν, πριν από 10 χρόνια, δημιούργουσε

τις πρώτες γραμμές κώδικα PHP. Τόνισε όμως ότι η PHP δεν θα είχε γίνει τόσο δημοφιλής αν η εξέλιξή της είχε παραμείνει προσωπική του προσπάθεια και δεν είχε βοηθηθεί από τους Andi Gutmans, Zeev Suraski και την εθελοντική συμμετοχή προγραμματιστών από ολόκληρο τον κόσμο. Τα περισσότερα web sites επί του παρόντος χρησιμοποιούν κυρίως τις εκδόσεις 4 και 5 της PHP.

Με τις γλώσσες Perl και C/C++ στις οποίες έχει τις ρίζες της, η PHP έχει εξαιρετική ομοιότητα ως προς τον τρόπο σύνταξης, αλλά και πολλές εντολές της.

2.5.1 Σχόλια

Όταν ο προχωρημένος προγραμματιστής φτάσει στο σημείο να δημιουργεί ολοκληρωμένες εφαρμογές PHP, το πιθανότερο είναι ότι θα χρειαστεί να τοποθετήσει σχόλια σε διάφορα σημεία του περίπλοκου κώδικα, ώστε να θυμάται τη λειτουργία που εκτελούν.

Σχόλια μπορούν να εισαχθούν είτε με δύο χαρακτήρες καθέτου // στην αρχή κάθε γραμμής, είτε με /* και */ στην αρχή και στο τέλος του σχολίου αντίστοιχα (ανεξάρτητα από τον αριθμό των γραμμών του). Το κείμενο που σημειώνεται με αυτό τον τρόπο, αγνοείται κατά την επεξεργασία του κώδικα PHP από τον web server.

2.5.2 Μεταβλητές

Όπως οι περισσότερες γλώσσες προγραμματισμού, έτσι και η PHP υποστηρίζει μεταβλητές. Στην PHP οι μεταβλητές φέρουν ως πρόθεμα τον χαρακτήρα του δολλαρίου (\$).

Είναι εφικτό επίσης μια μεταβλητή να επεκταθεί με περισσότερες τιμές, χρησιμοποιώντας τον χαρακτήρα .= (προσέξτε την τελεία πριν το "ίσον"). Τη διαφορά .= (με τελεία) και = (χωρίς τελεία) τη βλέπουμε στα δύο επόμενα παραδείγματα.

Η PHP επιτρέπει την τμηματική αντικατάσταση περιεχομένων σε strings με το function str_replace(). Το function str_replace() είναι case sensitive, κάτι που σημαίνει ότι παίζει ρόλο η διαφορά πεζών και κεφαλαίων γραμμάτων στην αντικατάσταση κειμένου. Για την αντικατάσταση κειμένου χωρίς περιορισμούς μεταξύ πεζών και κεφαλαίων, μπορεί να χρησιμοποιηθεί το function str_ireplace() (case insensitive string replace), το οποίο όμως υποστηρίζεται μόνο από την έκδοση 5.0 και νεότερη της γλώσσας PHP.

2.5.3 Σταθερές

Οι constants δεν φέρουν πρόθεμα και ορίζονται με την εντολή define.

2.5.4 Τελεστές

Οι operators θέτουν προϋποθέσεις για την εκτέλεση ενός τμήματος κώδικα. Οι operators διατυπώνονται με τις εντολές if, elseif, else ή switch/case σε συνδυασμό με τα σύμβολα !=, ==, >, <, <=, >=.

Αναλυτικά:

Operators σύγκρισης

$\$a == \b Το $\$a$ είναι ίσο με το $\$b$

$\$a != \b Το $\$a$ δεν είναι ίσο με το $\$b$

$\$a > \b Το $\$a$ είναι μεγαλύτερο του $\$b$

$\$a >= \b Το $\$a$ είναι μεγαλύτερο ή ίσο του $\$b$

Operators μαθηματικών πράξεων

$\$a + \b Πρόσθεση $\$a$ συν $\$b$

$\$a - \b Αφαίρεση $\$a - \b

$\$a * \b Πολλαπλασιασμός του $\$a$ επί το $\$b$

$\$a / \b Διαίρεση $\$a$ προς $\$b$

$\$a++$ Αύξηση του $\$a$ κατά 1 μονάδα

$\$a--$ Μείωση του $\$a$ κατά 1 μονάδα

Λογικοί operators

$\&\&$ και

$\|\|$ ή

$!$ όχι

Ακολουθούν μερικά δείγματα κώδικα όπου γίνεται χρήση των παραπάνω operators και η "μετάφρασή" τους.

```
if ($a > $b)
```

Αν το $\$a$ είναι μεγαλύτερο από το $\$b$

```
if ($a > $b && $c < $d)
```

Αν το $\$a$ είναι μεγαλύτερο από το $\$b$ και το $\$c$ είναι μικρότερο από το $\$d$

```
if ($text1 == "Καλημέρα" || $text2 == "Καλησπέρα")
```

Αν το $\$text1$ είναι "Καλημέρα" ή το $\$text2$ είναι "Καλησπέρα"

```
if ($text1 != "Καλησπέρα")
```

Αν το $\$text1$ δεν είναι "Καλησπέρα"

Προσέξτε τον διπλό χαρακτήρα "ίσον" `==` στους παραπάνω operators. Κατά τον ορισμό προϋποθέσεων για τις τιμές μεταβλητών στην PHP με τους operators `if`, `elseif`, `else` πρέπει απαραίτητα να χρησιμοποιείται το διπλό "ίσον". Αν χρησιμοποιούσαμε μόνο "ίσον" στον παραπάνω κώδικα (δηλαδή π.χ. `if ($mynumber = 6)` αντί για `if ($mynumber == 6)`), η PHP θα απέδιδε στη μεταβλητή `$mynumber` τον αριθμό 6, κάτι που θα προκαλούσε προβλήματα στην εκτέλεση του υπόλοιπου κώδικα.

2.5.5 GET και POST

Αν μέχρι τώρα έχετε επισκεφθεί πολλά sites με σελίδες php, σίγουρα θα έχετε παρατηρήσει ότι συχνά στο URL τους υπάρχει ο χαρακτήρας του αγγλικού ερωτηματικού ? ακολουθούμενο από άλλες παραμέτρους. Π.χ. κάτι σαν `article.php?id=256&page=news&t=25432&something=other`

Οι παράμετροι αυτές λαμβάνονται από τον web server και αφού ενσωματωθούν ως μεταβλητές στο αρχείο PHP, υφίστανται επεξεργασία ώστε η σελίδα να εμφανίσει συγκεκριμένες πληροφορίες. Ακολουθεί ένα παράδειγμα κώδικα όπου τα GET data χρησιμοποιούνται για να εμφανίσουν διαφορετικό κείμενο:

```
<?php
```

```
    $id = $_GET['id'];
```

```
    if ($id == 1) {
```

```
        print "Κείμενο 1";
```

```
    }
```

```
elseif ($id == 2) {  
    print "Δεύτερο κείμενο";  
}  
else {  
    print "Άλλο κείμενο";  
}
```

?>

Υποθέτουμε ότι το αρχείο με τον παραπάνω κώδικα ονομάζεται page.php

Ο επισκέπτης που θα ανοίξει τη σελίδα με τη μορφή page.php?id=1 θα δει το κείμενο "Κείμενο 1". Αν ανοίξει τη σελίδα σαν page.php?id=2 θα δει στην οθόνη του τη φράση "Δεύτερο κείμενο", ενώ για οποιοδήποτε άλλο id, θα δει τη φράση "Άλλο κείμενο".

Αν ο αριθμός του id πρόκειται να σταλεί μέσω μια φόρμας που χρησιμοποιεί τη μέθοδο POST, αρκεί να αντικατασταθεί η γραμμή:

```
$id = $_GET['id'];
```

με την ακόλουθη:

```
$id = $_POST['id'];
```

Σε περίπτωση που το \$id πρόκειται να χρησιμοποιηθεί σε ερωτήματα βάσεων δεδομένων, έχοντας πάντα αριθμητική τιμή, προτείνεται η χρήση του function intval() για την αποφυγή απώλειας δεδομένων από τον server μέσω SQL injections. Με τη χρήση του intval, αποκόπτονται οι μη αριθμητικοί χαρακτήρες από μια μεταβλητή.

Η λειτουργία του εξηγείται στο ακόλουθο παράδειγμα:

```
<?php
```

```
    $text = "something223344";  
    $text = intval($text);  
    print "$text";
```

```
?>
```

που θα έχει σαν αποτέλεσμα την εμφάνιση του αριθμού 223344

Ο παλαιότερος τρόπος λήψης μεταβλητών μέσω POST και GET data ήταν με τη χρήση των long arrays \$HTTP_POST_VARS και \$HTTP_GET_VARS αντίστοιχα.

Οι προγραμματιστές ιστοσελίδων θα πρέπει να αντικαταστήσουν το συντομότερο δυνατό στις εφαρμογές PHP που δημιουργούν, τα \$HTTP_POST_VARS και \$HTTP_GET_VARS με τα \$_POST και \$_GET αντίστοιχα.

Τα \$HTTP_POST_VARS και \$HTTP_GET_VARS καταργούνται και ήδη δεν είναι συμβατά με την νέα έκδοση 6 της PHP. Αυτό ισχύει ακόμη και αν στις ρυθμίσεις της PHP 6 ενεργοποιηθούν τα long arrays (register_long_arrays = On).

2.5.6 Includes

Σε πολλές περιπτώσεις θα χρειαστεί να δημιουργηθεί μια web εφαρμογή, η οποία σε ορισμένα σημεία θα περιέχει κοινά στοιχεία κώδικα PHP, όπως για παράδειγμα ίδιος header σε όλες τις σελίδες της.

Για να μην αναγκάζεται ο προγραμματιστής να γράφει και να τροποποιεί τον ίδιο κώδικα σε όλα τα αρχεία php

της εφαρμογής, μπορεί να χρησιμοποιήσει τη μέθοδο `include`. Ακολουθεί ένα παράδειγμα: Θέλουμε να εμφανίσουμε ένα μήνυμα καλωσορίσματος στην αρχή όλων των σελίδων.

Μέρος 1ο: Δημιουργούμε ένα αρχείο `header.php` με τον εξής κώδικα:

```
<?php  
    print "Καλωσήρθατε στη σελίδα μου!";  
?>
```

και το αποθηκεύουμε στον ίδιο φάκελο με τα άλλα αρχεία της εφαρμογής.

Μέρος 2ο: Στην αρχή των άλλων σελίδων `php` τοποθετούμε τον εξής κώδικα:

```
<?php  
    include "header.php";  
?>
```

Έτσι, το μήνυμα "Καλωσήρθατε στη σελίδα μου!" θα εμφανίζεται τώρα σε όλες τις σελίδες της εφαρμογής. Όταν χρειαστεί να τροποποιήσουμε αυτό το μήνυμα καλωσορίσματος, δεν θα χρειαστεί να ανοίξουμε μία-μία όλες τις σελίδες της εφαρμογής, αλλά κάνουμε την τροποποίηση μόνο στο `header.php`. Η αλλαγή που κάναμε στο `header.php` θα εμφανιστεί αυτόματα σε όλες τις σελίδες.

2.5.7 HTML και PHP

Η PHP και η HTML είναι δύο γλώσσες προγραμματισμού που συνεργάζονται απόλυτα μεταξύ τους. Όπως ήδη αναφέρθηκε προηγουμένως, ο `web server` επεξεργάζεται τον κώδικα PHP, ώστε να παράγει "on the fly" κώδικα HTML που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών. Πολλοί νέοι προγραμματιστές συχνά θέτουν το ερώτημα: "Τι να μάθω πρώτα; PHP ή HTML;"

Η απάντηση είναι ότι πρώτα πρέπει να μάθετε HTML και μετά PHP. Κάποιος που δεν ξέρει HTML δεν θα μπορέσει να δημιουργήσει σελίδες χρησιμοποιώντας PHP. Κι αυτό γιατί δεν θα γνωρίζει με ποιο τρόπο θα προγραμματίσει τις σελίδες PHP ώστε να παραγουν "on the fly" σωστό περιεχόμενο HTML που θα εμφανιστεί στους `web browsers` των επισκεπτών.

Σε ένα τμήμα κειμένου που παράγεται μέσω PHP μπορείτε να χρησιμοποιήσετε οποιοδήποτε `tag` της γλώσσας HTML.

Για παράδειγμα ο ακόλουθος κώδικας PHP:

```
<?php  
    echo "<p><b>Hello!</b></p>  
        <p><i>Welcome to my page</i></p>";  
?>
```

Θα δώσει:
Hello!

Welcome to my page

Πρέπει όμως να προσέξετε και την περίπτωση που ο κώδικας HTML περιέχει εισαγωγικά. Συγκεκριμένα, το παρακάτω δείγμα κώδικα δεν θα λειτουργήσει και θα εμφανίσει σφάλμα επεξεργασίας (`parse error`):

```
<?php
// Δείγμα λανθασμένου κώδικα PHP!
echo "<a href='mypage2.html'>Page 2</a>";
?>
```

Ο λόγος είναι ότι στον κώδικα HTML υπάρχουν εισαγωγικά, χωρίς χαρακτήρες διαφυγής (escape characters). Έτσι, ο web server "μπερδεύεται" και αντιλαμβάνεται τα εισαγωγικά της HTML ως σημεία τερματισμού της εντολής PHP echo, με αποτέλεσμα να μην αναγνωρίζει σωστά ότι ακολουθεί μετά από αυτά.

Η χρήση χαρακτήρων διαφυγής με ανάποδη κάθετο (\) είναι η μία λύση γι' αυτό το πρόβλημα. Με τους χαρακτήρες διαφυγής τα εισαγωγικά της HTML αντικαθίστανται από \" όταν περιέχονται σε εντολές PHP.

Επομένως, ο παραπάνω κώδικας θα λειτουργήσει σωστά αν διορθωθεί ως εξής:

```
<?php
echo "<a href='\"mypage2.html\"'>Page 2</a>";
?>
```

Εναλλακτικά, μπορείτε να χρησιμοποιήσετε τον παρακάτω κώδικα, χωρίς να χρειαστούν χαρακτήρες διαφυγής.

```
<?php
print <<<EOF
<a href="mypage2.html">Page 2</a>
EOF;
?>
```

2.5.8 PHP και Javascript

Παρόλο που οι PHP και [Javascript](#) είναι δύο γλώσσες που έχουν σαν κοινό στοιχείο την παραγωγή δυναμικού περιεχομένου, έχουν την εξής σημαντική διαφορά: η πρώτη είναι server side, δηλαδή ο κώδικάς της εκτελείται στον διακομιστή ιστοσελίδων όπου παράγεται το περιεχόμενο HTML, ενώ η δεύτερη είναι client site, δηλαδή εκτελείται από τα προγράμματα περιήγησης των επισκεπτών.

Παρόλα αυτά, οι δύο γλώσσες μπορούν να συνεργαστούν μεταξύ τους. Ακολουθεί ένα παράδειγμα όπου προβάλλεται ένα Javascript alert box με δεδομένα που έχουν προκύψει κατόπιν επεξεργασίας κώδικα PHP:

```
<?php
$cid = $_GET['id'];
if ($cid == 4){
    $text = "Καλημέρα!";
}
else {
    $text = "Καλησπέρα!";
}
print "<script language=javascript>
alert('$text');
</script>";
?>
```

Ας υποθέσουμε ότι η σελίδα αυτή αποθηκεύεται στον web server με το όνομα page.php

Ο επισκέπτης που θα ανοίξει τη σελίδα με τη μορφή page.php?id=4 θα δει τον browser του να εμφανίζει ένα

Javascript alert box με το κείμενο:

Καλημέρα!

ενώ αν την ανοίξει με τη μορφή `page.php?id=1` ή γενικά οποιοδήποτε άλλο id, θα δει ένα alert box με το κείμενο:

Καλησπέρα!

2.6 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 (multiparadigm), υποστηρίζοντας αντικειμενοστρεφές, προστακτικό και συναρτησιακό στυλ προγραμματισμού.

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

2.7 Εργαλεία ανάπτυξης δυναμικών εφαρμογών

Ίσως ο πιο απλός και βολικός τρόπος για να πειραματιστούμε με την PHP και τη MySQL, χωρίς να μπλέξουμε με web hosts, είναι να στήσουμε ένα πακέτο WAMP (Windows, Apache, MySQL & PHP) στον υπολογιστή μας. Ένα από τα πλέον διαδεδομένα και πλήρη πακέτα WAMP είναι το EasyPHP, το οποίο πέρα από το καθιερωμένο τρίπτυχο Apache, MySQL & PHP, προσφέρει και το εργαλείο phpMyAdmin για την εύκολη διαχείριση βάσεων δεδομένων MySQL. Μπορούμε φυσικά να εγκαταστήσουμε ξεχωριστά τις εφαρμογές που περιέχει το EasyPHP, αλλά θα χάσουμε έτσι ορισμένα από τα πλεονεκτήματα που μας προσφέρει. Ένα από αυτά είναι η βεβαιότητα ότι οι εκδόσεις των εφαρμογών είναι συμβατές μεταξύ τους, κάτι για το οποίο δεν μπορούμε να είμαστε πάντα σίγουροι όταν τις εγκαθιστούμε ξεχωριστά. Ας δούμε λοιπόν τη διαδικασία εγκατάστασης σε περιβάλλον Windows 7 (αν θέλετε κάτι παρόμοιο για Linux ή Mac, τσεκάρτε το XAMPP και το MAMP αντίστοιχα).

2.7.1 EasyPHP

Η EasyPHP είναι ένα πακέτο λογισμικού για τα Windows που επιτρέπει στην PHP και στην MySQL να αναπτυχθεί στον προσωπικό υπολογιστή ενός προγραμματιστή. Ξεκίνησε το 1999 και ήταν η πρώτη δέσμη λογισμικού WAMP. Η EasyPHP εγκαθιστά υπηρεσίες ενός web server στα Windows του υπολογιστή και

επιτρέπει την γρήγορη και εύκολη ανάπτυξη της PHP και MySQL σε localhost (γνωστό και ως 127.0.0.1). Το πακέτο περιλαμβάνει έναν Apache server, MySQL βάση δεδομένων , και PHP επέκταση. Το πακέτο αυτό είναι ικανό να αναπτυχθεί είτε σε ένα PC είτε σε ένα USB drive. Η διαμόρφωση γίνεται αυτόματα από την εφαρμογή χωρίς αλληλεπίδραση από τον χρήστη.

Εγκατάσταση και βασικές επιλογές του EasyPHP

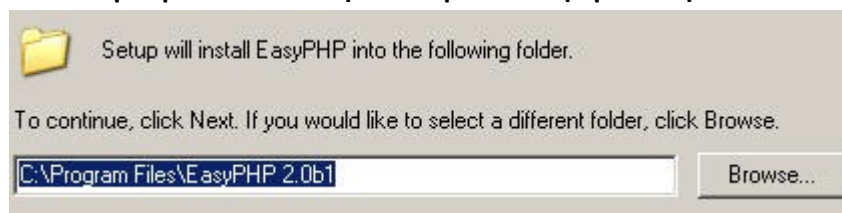
Παρακάτω ακολουθούν τα βήματα της εγκατάστασης του EasyPHP

1. Αρχικά, αποθηκεύουμε το αρχείο εγκατάστασης που βρίσκεται στη θέση: <http://www.easyphp.org/save-easyphp-devservervc9-latest.php>
2. Εκτελούμε διπλό κλικ πάνω στο αρχείο εγκατάστασης και ακολουθούμε τα παρακάτω βήματα:



Εικόνα 1 - Εγκατάσταση EasyPHP

3. Επιλέγουμε Ok -> Next -> Επιλέγουμε την επιλογή I Accept the Agreement -> Next -> Next
4. Στην περίπτωση που επιθυμούμε να το αποθηκεύσουμε σε διαφορετικό φάκελο επιλέγουμε Browse:



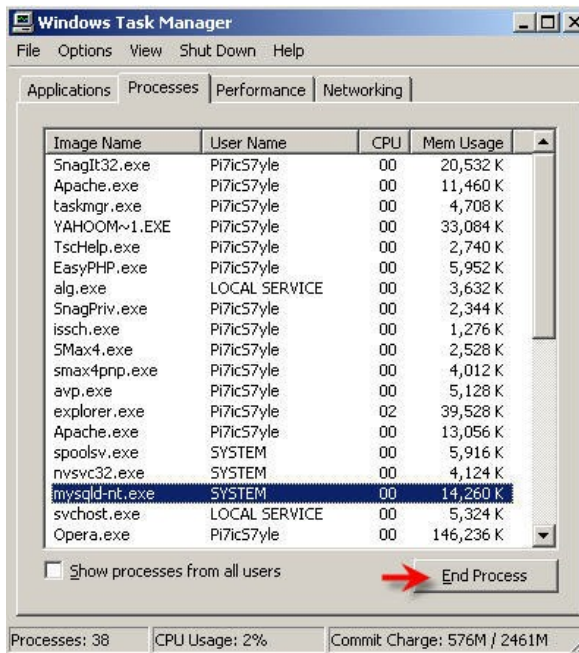
Εικόνα 2 - Επιλογή φακέλου εγκατάστασης

5. Επιλέγουμε Next -> Next -> Install -> Finish
6. Τώρα το EasyPHP έχει εγκατασταθεί και ένα εικονίδιο έχει εμφανιστεί στη γραμμή εργασιών.



Εικόνα 3 - Συντόμευση EasyPHP

7. Εκτελούμε διπλό κλικ στο παραπάνω εικονίδιο και ο εξυπηρετητής βρίσκεται σε λειτουργία. Στην περίπτωση που έχουμε εγκαταστήσει παλαιότερα την MySQL, ο εξυπηρετητής της MySQL δεν θα ξεκινήσει. Γι' αυτό το λόγο, θα πρέπει να τερματίσουμε τη διεργασία mysql-nt.exe.

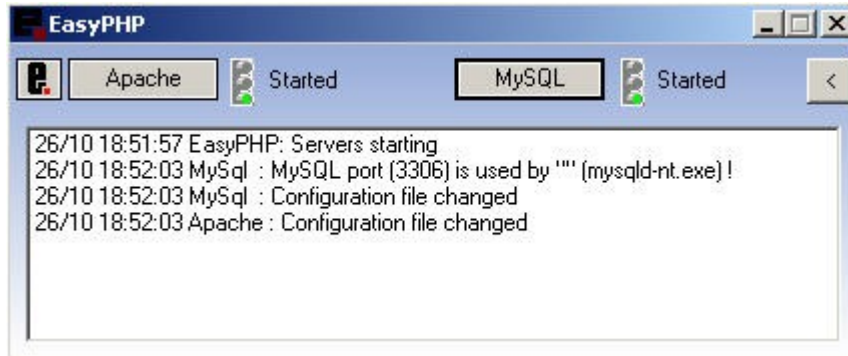


Εικόνα 4 - Διαχείριση εργασιών

8. Τώρα, με διπλό κλικ στο εικονίδιο της EasyPHP επιλέγουμε την έναρξη της MySQL

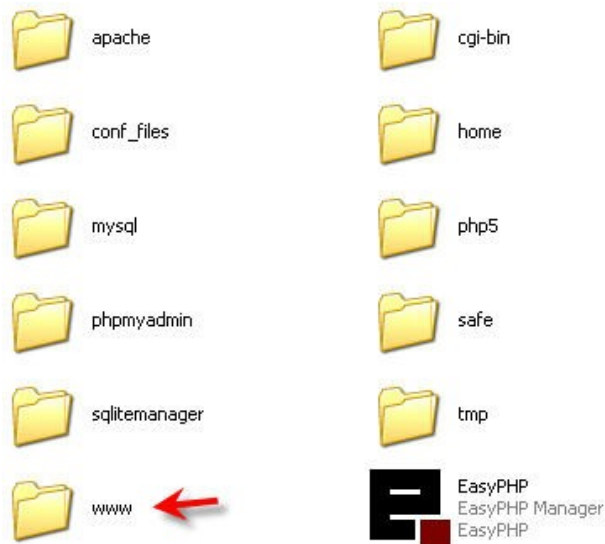


Εικόνα 5 - Παράθυρο εργασιών EasyPHP (a)



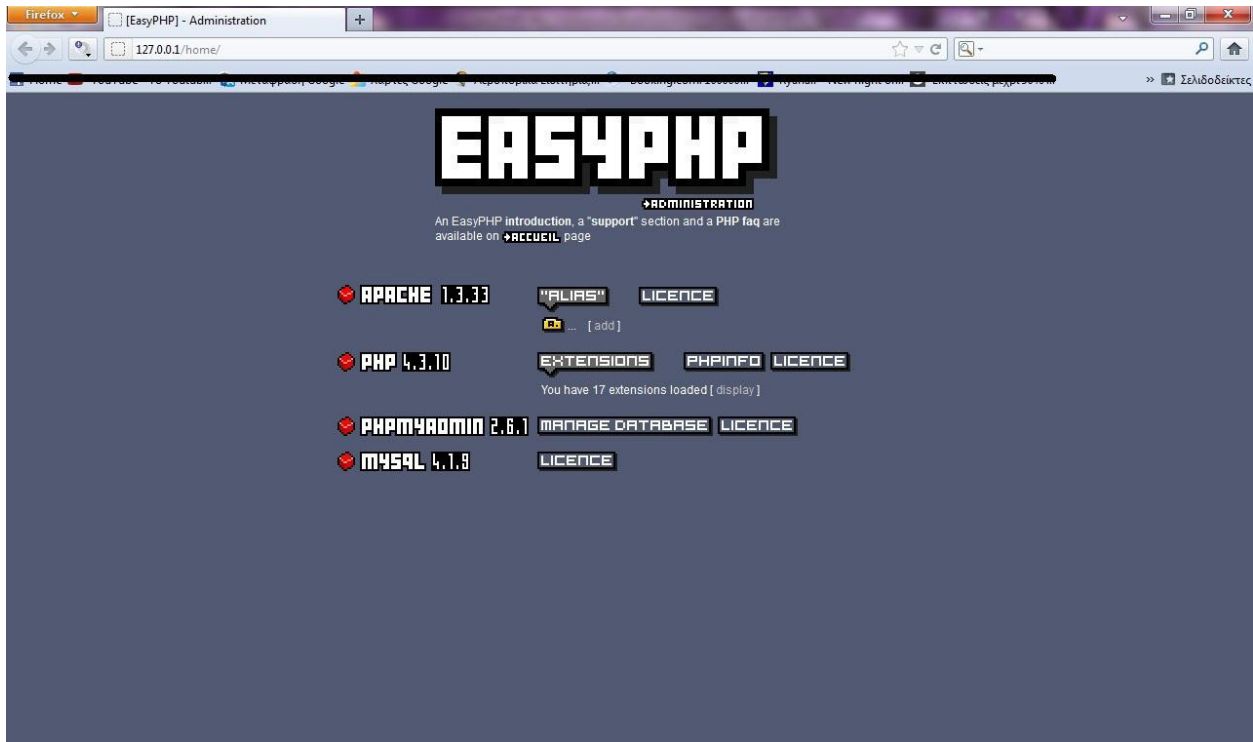
Εικόνα 6 - Παράθυρο εργασιών EasyPHP (b)

9. Για να εκτελεστούν τα αρχεία μας θα πρέπει να τα αντιγράψουμε στο φάκελο **www**



Εικόνα 7 - Φάκελος **www**

10. Τέλος, αν γράψουμε στο φυλλομετρητή μας <http://localhost> θα εμφανιστεί η κεντρική σελίδα της EasyPHP που θα είναι κάπως έτσι:



Εικόνα 8 - Αρχική σελίδα διαχείρισης EasyPHP

2.7.2 PhpMyAdmin

Το εργαλείο phpMyAdmin παρέχει εκτεταμένη διαχείριση βάσεων δεδομένων MySQL στο Διαδίκτυο. Υποστηρίζει μια ευρεία λίστα λειτουργιών μέσω γραφικής διεπαφής, όπως σχετικά με πίνακες, πεδία, συσχετίσεις, ευρετήρια, χρήστες και δικαιώματα. Παράλληλα, είναι δυνατή και η εκτέλεση οποιασδήποτε εντολής SQL.

Υποστηριζόμενα χαρακτηριστικά

Το phpMyAdmin μπορεί μέχρι στιγμής να κάνει:

- να περιηγηθεί και να διαγράψει βάσεις δεδομένων, πίνακες, προβολές στήλες και ευρετήρια
- να προβάλει σύνολα πολλαπλών αποτελεσμάτων μέσω αποθηκευμένων διεργασιών ή ερωτημάτων
- να δημιουργήσει, αντιγράψει, διαγράψει, μετονομάσει και αλλάξει βάσεις δεδομένων, πίνακες, στήλες και ευρετήρια
- να συντηρεί το διακομιστή, τις βάσεις δεδομένων και τους πίνακες με προτάσεις για τις ρυθμίσεις του διακομιστή
- να εκτελεί, επεξεργάζεται και σελιδοποιεί κάθε δήλωση :όρο:«SQL», καθώς και δέσμες ερωτημάτων
- να φορτώσει αρχεία κειμένου σε πίνακες
- να δημιουργήσει και να διαβάσει απορρίμματα πινάκων
- εξαγωγή δεδομένων σε διάφορες μορφές: *CSV*, *XML*, *PDF*, *ISO/IEC 26300 - OpenDocument* Κείμενο και Λογιστικό φύλλο, *Microsoft Word 2000*, και μορφές *LATEX*
- import data and *MySQL* structures from *OpenDocument* spreadsheets, as well as *XML*, *CSV*, and *SQL* files
- να διαχειριστεί πολλούς διακομιστές

- να διαχειριστεί χρήστες MySQL και τα δικαιώματά τους
- να ελέγξει την ακεραιότητα συσχέτισης σε πίνακες MyISAM
- να χρησιμοποιεί Ερώτημα από παράδειγμα (QBE), να δημιουργεί περίπλοκα ερωτήματα που συνδέουν αυτόματα τους απαιτούμενους πίνακες
- δημιουργία γραφικών PDF της διάταξης της βάσης δεδομένων σας
- να αναζητεί καθολικά σε μια βάση δεδομένων ή μέρος αυτής
- να μεταφέρει αποθηκευμένα δεδομένα σε οποιαδήποτε μορφή χρησιμοποιώντας ένα σύνολο από προκαθορισμένες συναρτήσεις, όπως η εμφάνιση δεδομένων BLOB όπως εικόνα ή σύνδεσμο λήψης
- να ανιχνεύει αλλαγές σε βάσεις δεδομένων, πίνακες και προβολές
- να επικοινωνεί σε 62 διαφορετικές γλώσσες

2.7.3 Εξυπηρετητής Apache

Ο Apache Web Server είναι αυτό ακριβώς που δηλώνει το όνομά του. Πρόκειται δηλαδή για έναν εξυπηρετητή (server) του παγκόσμιου Ιστού (Web). Με τον όρο server το μυαλό μας πηγαίνει ίσως σε ηλεκτρονικούς υπολογιστές που φιλοξενούν ιστοσελίδες και όχι άδικα. Ο όρος αυτός χρησιμοποιείται και για το μηχάνημα εξυπηρετητή (hardware) αλλά και για το πρόγραμμα (software). Στο άρθρο αυτό θα ασχοληθούμε μόνο με το software και συγκεκριμένα με τον Apache.

Ο Apache εγκαθίσταται σε έναν υπολογιστή ο οποίος μπορεί να χρησιμοποιεί διάφορα λειτουργικά συστήματα όπως Linux, Unix, Microsoft Windows, GNU, FreeBSD, Solaris, Novell NetWare, Mac OS X, OS/2, TPF. Ο ρόλος του Apache είναι να αναμένει αιτήσεις από διάφορα προγράμματα – χρήστες (clients) όπως είναι ένας ο φυλλομετρητής (browser) ενός χρήστη και στη συνέχεια να εξυπηρετεί αυτές τις αιτήσεις “σερβίροντας” τις σελίδες που ζητούν είτε απευθείας μέσω μιας ηλεκτρονικής διεύθυνσης (URL), είτε μέσω ενός συνδέσμου (link). Ο τρόπος με τον οποίο ο Apache εξυπηρετεί αυτές τις αιτήσεις, είναι σύμφωνος με τα πρότυπα που ορίζει το πρωτόκολλο HTTP (Hypertext Transfer Protocol).

Το ξεκίνημα και η εξέλιξη του Apache Web Server

Η περίοδος έναρξης της δημιουργίας του προγράμματος χρονολογείται στις αρχές του 1990, όταν άρχισε να αναπτύσσεται από τον Robert McCool, ως ένα project του National Center for Supercomputing Applications (NCSA) με το όνομα HTTPd (HTTP daemon). Το 1994 ο Robert McCool αποχώρησε από το NCSA με αποτέλεσμα το NCSA HTTPd να μείνει σχεδόν εγκαταλειμμένο, πέρα από κάποιες διορθώσεις (patches) που ανέπτυσαν και διένειμαν εκτός από τον McCool και άλλοι προγραμματιστές,. Το 1995 ανέλαβε το πρόγραμμα το Ίδρυμα Λογισμικού Apache (Apache Software Foundation), το οποίο διατηρεί την εποπτεία του έως και σήμερα.

Ο Apache σήμερα

Ο Apache HTTP αναπτύσσεται από την “Κοινότητα Ανοικτού Λογισμικού” και η εποπτεία, υποστήριξη, και διάθεση του προγράμματος γίνεται από το Apache Software Foundation. Το πρόγραμμα είναι ανοιχτού κώδικα (open source), κάτι που σημαίνει ότι σύμφωνα με την άδεια χρήσης του (license), διατίθεται δωρεάν και μπορούν να γίνουν ελεύθερα από το χρήστη προσθήκες και τροποποιήσεις στον κώδικα του.

Η προέλευση του ονόματος του

Υπάρχουν δύο εκδοχές σχετικά με την προέλευση του ονόματος του. Η πρώτη εντοπίζεται στα πρώτα χρόνια της δημιουργίας του, τότε που ως NCSA HTTPd έπρεπε να αναπτύσσονται συνεχώς διορθώσεις (patches) για να ενσωματωθούν στον αρχικό του κώδικα με αποτέλεσμα να του δοθεί το όνομα a patchy server. Η δεύτερη εκδοχή σύμφωνα με το ίδρυμα Apache, αναφέρει ότι το όνομα αυτό δόθηκε προς τιμήν των ιθαγενών Ινδιάνων της Αμερικής και συμβολίζει το μαχητικό πνεύμα και την αντοχή.

Χαρακτηριστικά και λειτουργίες του Apache HTTP

Ο Apache διαθέτει ποικιλία χαρακτηριστικών και μπορεί να υποστηρίξει μια μεγάλη γκάμα εφαρμογών με τις οποίες και συνεργάζεται. Οι δυνατότητες του προγράμματος αυτού καθαυτού και τα χαρακτηριστικά του δεν είναι και τόσο πολλά. Ένα από τα βασικότερα χαρακτηριστικά του όμως, το οποίο και του δίνει μεγάλες δυνατότητες, είναι ότι μπορεί να προσαρμόσει επάνω του πολλές προσθήκες προγραμμάτων (modules), τα οποία με τη σειρά τους παρέχουν διαφορετικές λειτουργίες. Μερικά από τα πιο γνωστά modules του Apache HTTP είναι τα modules πιστοποίησης, όπως για παράδειγμα τα mod_access, mod_auth, mod_digest κ.λπ. Παρέχει επίσης SSL σε TLS μέσω των (mod_ssl), και proxy module (mod_proxy), πραγματοποιεί ανακατευθύνσεις διευθύνσεων (URL rewrites) μέσω του mod_rewrite, καταγραφές συνδέσεων μέσω του mod_log_config, συμπίεση αρχείων μέσω του mod_gzip και πολλά άλλα modules τα οποία διατίθενται είτε απ'ο το Apache Software Foundation, είτε από τρίτες εταιρίες λογισμικού.

Ένα άλλο χαρακτηριστικό – δυνατότητα του Apache HTTP, όπως έχω αναφέρω πιο πάνω, είναι ότι μπορεί να εγκατασταθεί σε διάφορα λειτουργικά συστήματα. Ο Apache HTTP υποστηρίζει επίσης αρκετές διάσημες εφαρμογές και γλώσσες προγραμματισμού όπως MySQL, PHP, Perl, Python κ.λπ.

Αυτά είναι μερικά από τα χαρακτηριστικά και τις λειτουργίες του που κάνουν τον Apache τον πιο δημοφιλή Web Server από το 1996 έως τις μέρες μας. Περισσότερο από το 50% των ιστοχώρων του παγκόσμιου ιστού, χρησιμοποιεί τον Apache ως εξυπηρετητή. Το υπόλοιπο ποσοστό καλύπτουν αντίστοιχα προγράμματα, όπως το Microsoft Internet Information Services (IIS), ο Sun Java System Web Server, ο Zeus Web Server κ.

2.7.4 Dreamweaver

Το Dreamweaver είναι ένα κορυφαίο πρόγραμμα επεξεργασίας HTML και εργαλεία ανάπτυξης εφαρμογών για το web. Είτε χρησιμοποιείτε το Mac OS X είτε τα Windows ,θα έχετε στην διάθεσή σας ένα δυνατό σύνολο χαρακτηριστικών από το Dreamweaver .Οι διασυνδέσεις να παράγετε τις ίδιες όμορφες ,λειτουργικές ιστοθέσεις και με τις δύο εκδοσεις . Το Dreamweaver είναι εξαίρετο για την δημιουργία στα γρήγορα ελκυστικών ιστοσελίδων, που περιλαμβάνουν μορφοποιημένο κείμενο, γραφικά, φόρμες πλαίσια, πίνακες και άλλα αντικείμενα. Το Dreamweaver είναι καλύτερο από την DHTML (δυναμική HTML),και επιτρέπει κίνηση γραμμής χρόνου ,απόλυτη τοποθέτηση και συγγραφή script στις οποίες κάνετε απλώς κλικ για να προσθέσετε ένα συγκεκριμένο αντικείμενο. Το Dreamweaver σας επιτρέπει επίσης να δημιουργήσετε και να εφαρμόσετε πρότητα διαδοχικά φύλλα στυλ(cascading style sheets,css)για να μορφοποιείτε την εμφάνιση των ιστοσελίδων σας μέσα στο πρόγραμμα περιήγησης.Το Dreamweaver σας δίνει την δυνατότητα να χρησιμοποιείτε css για να μορφοποιείτε τις ιστοσελίδες 34σας.Μπορείτε να χρησιμοποιείτε επίσης εργαλεία του Dreamweaver για να σχεδιάζετε διατάξεις σελίδων χρησιμοποιώντας css

Σε Ποιον Απευθύνεται και τι μπορείτε να κάνετε με το Dreamweaver

Είτε δημιουργείτε την πρώτη σας ιστοσελίδα ,είτε έχετε αποφασίσει πρόσφατα να δοκιμάσετε λογισμικό επεξεργασίας web αφού κωδικοποιούσατε με το χέρι για πολλά χρόνια , το Dreamweaver θα σας αρέσει ιδιαίτερα.Το Dreamweaver σας δίνει την ευελιξία να δημιουργήσετε την δική σας προσωπική ιστοσελίδα ή μια ολόκληρη ιστοθέση στο εταιρικό σας δίκτυο.

Το Dreamweaver είναι το πιο δημοφιλές επαγγελματικό εργαλείο δημιουργίας ιστοθέσεων για το web.Ο αριθμός των χρηστών του είναι τεράστιος.Πολλες μεγάλες εταιρίες το χρησιμοποιούν σαν πρότυπο για ανάπτυξη για το web.Το Dreamweaver είναι επίσης ένα δημοφιλές εργαλείο για σχεδιαστές του web,οι οποίοι εργάζονται σε ιστοθέσεις για τους πελάτες τους.Το Dreamweaver είναι ελκυστικό σε μια μεγάλη ποικιλία χρηστών ,που περιλαμβάνουν πεπειραμένους και αρχάριους σ σχεδιαστές ιστοθέσεων προγραμματιστές

(αυτούς που θέλουν να γράφουν HTML) σχεδιαστές ειδικεύονται στην ανάπτυξη εφαρμογών με γλώσσες σαν τις Coldfusion και ASP.Net και σε εκείνους που δεν έχουν ακούσει τίποτε για αυτές τις γλώσσες εφαρμογών .Το Dreamweaver εργάζεται παρόμοια σε υπολογιστές Macintosh και windows και είναι δημοφιλές και στις δύο πλατφόρμες.

Η βάση του Dreamweaver είναι η γλώσσα σήμανσης υπερκειμένου (hypertext Markup Language,HTML) μια γλώσσα που χρησιμοποιείται για διαμόρφωση κειμένου, έτσι ώστε να μπορείτε να το δείτε στο worldwide Web με λογισμικό που ονομάζεται πρόγραμμα περιήγησης στο Web. Αλλά δεν χρειάζεται να ξέρετε HTML για να αρχίσετε να χρησιμοποιείτε το Dream weaver.

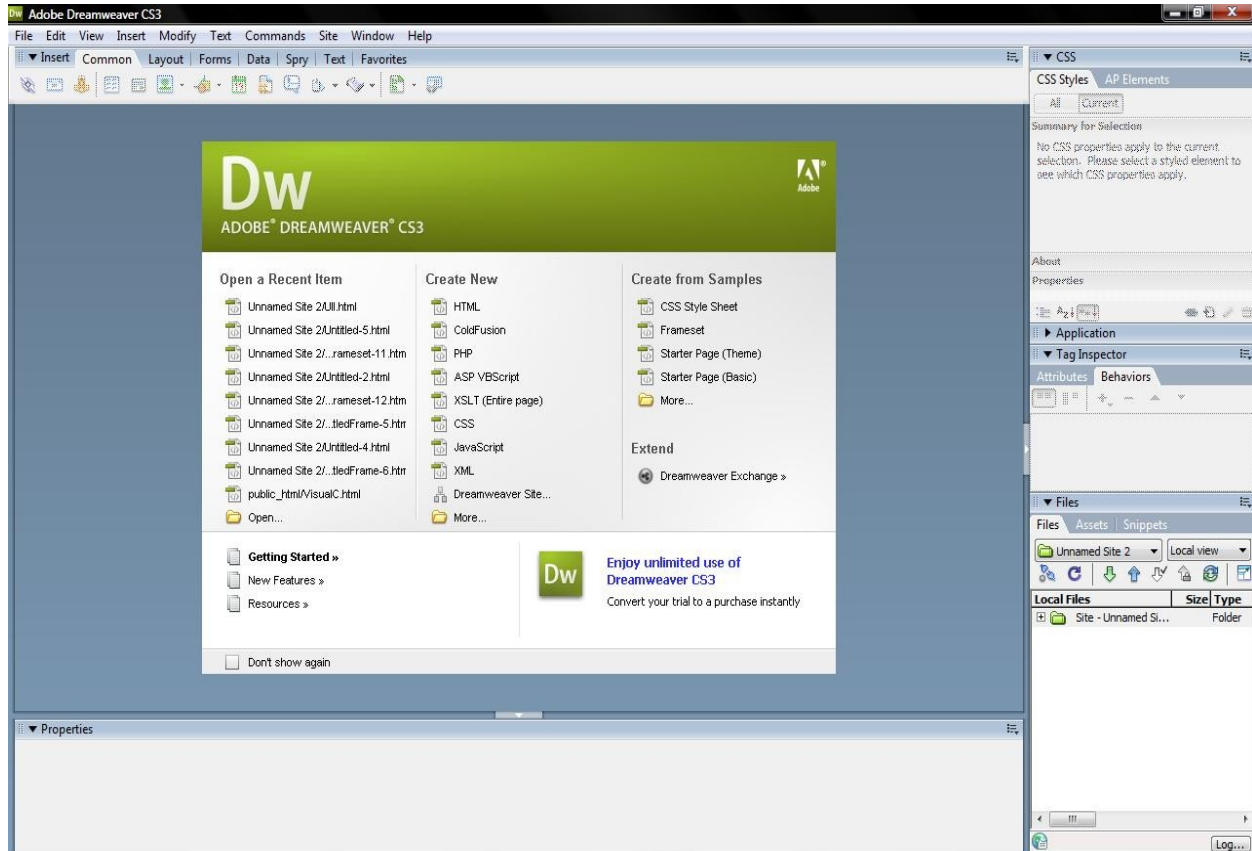
Γνωρίστε το Dreamweaver

Το Dreamweaver είναι ένα πλήρες περιβάλλον ανάπτυξης για το web ένας επεξεργαστής HTML (Hypertext Markup Language) ένα εργαλείο συγγραφής ,και ένα εργαλείο δυναμικής ανάπτυξης ιστοθέσεων και ένα εργαλείο διαχείρισης ιστοθέσεων όλα μέσα σε ένα πακέτο.Οι ιστοθέσεις δημιουργούνται με χρήση HTML ,αλλά μπορείτε να κάνετε πολλά πράγματα ,χωρίς να δείτε ποτέ καθόλου HTML .Αν θέλετε να παράγετε 35ποιοτικά επαγγελματικές ιστοσελίδες ,περιλαμβανομένης και της δυνατότητας συγγραφής σκριπτ ,το Dreamweaver σας διευκολύνει.

Το Dreamweaver είναι ένας επεξεργαστής ιστοσελίδων μορφής WYSIWYG(what you seeiw what you get,αυτό που βλέπετε, αυτό πέρνετε)που είναι ιδιαίτερα δυναμικός και ταυτόχρονα πολύ εύκολος στη χρήση.Μπορείτε να δημιουργήσετε νέες ιστοθέσεις με το Dream weaver,και μπορείτε να εισάγετε και να επεξεργαστείτε υπάρχουσες ιστοθέσεις. Το Dreamweaver δεν θα αλλάξει και δεν θα αναδιατάξει τον κώδικά σας HTML.Ένα από τα πιο δημοφιλή χαρακτηριστικά του Dreamweaver είναι από παλιά, ότι αφήνει ανέπαφες τις υπάρχουσες ιστοθέσεις .Οι άνθρωποι της Macromedia της εταιρίας πο δημιούργησε το Dreamweaver ,ονομάζουν αυτό το χαρακτηριστικό Roundtrip HTML Το Dreamweaver είναι επίσης ένα εργαλείο συγγραφής .Τι σημαίνει εργαλείο συγγραφής; Τα εργαλεία συγγραφής σας επιτρέπουν να δημιουργήσετε μια πλήρη εφαρμογή, που περιλαμβάνει διαδραστικότητα.Αν και το Dreamweaver μπορεί να χρησιμοποιηθεί σαν ένας επεξεργαστής HTML, μπορεί επίσης να χρησιμοποιηθεί για δημιουργία εφαρμογών πολυμέσων.Μπορείτε φυσικά να επεξεργαστείτε απλώς HTML με το Dreamweaver,ή μπορείτε να συγγράψετε κάτι που θέλετε για τους χρήστες σας,με ήχο, κίνηση, βίντεο και δραστικότητα.

Εγκατάσταση του λογισμικού του Dreamweaver

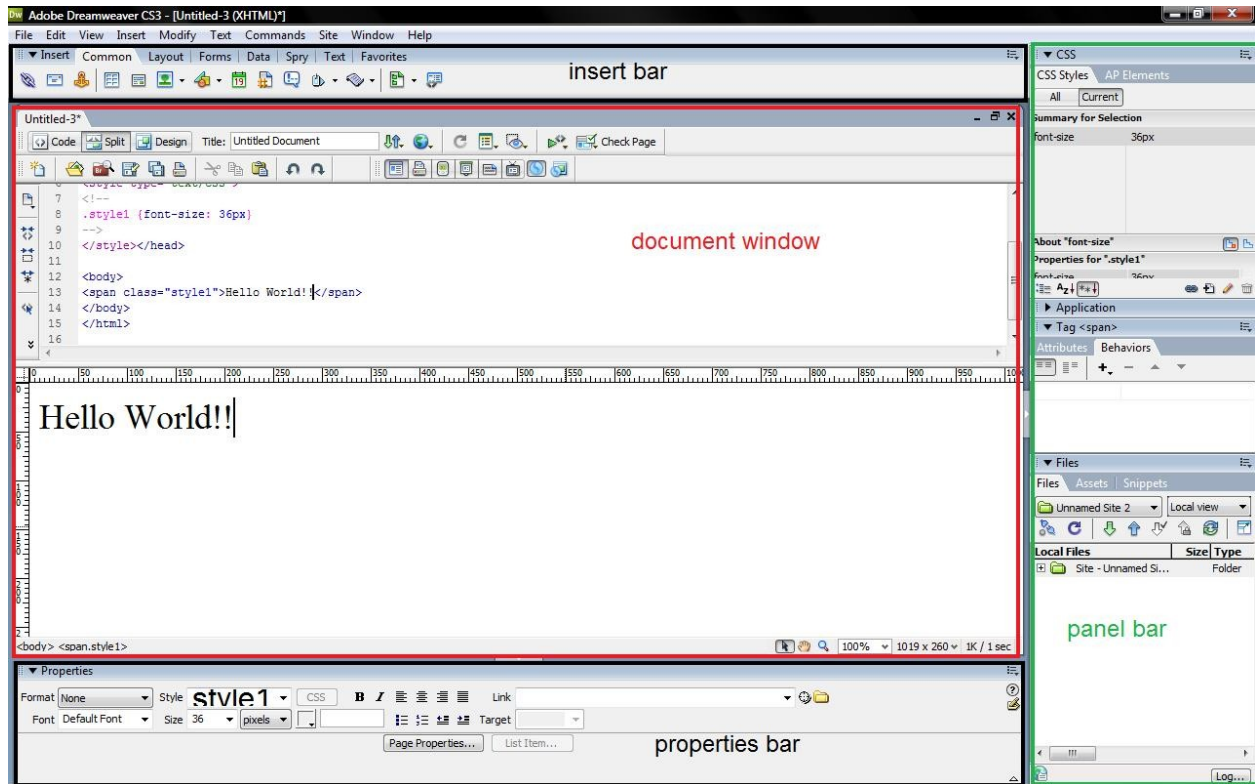
Το Dreamweaver, εγκαθίσταται από ένα πρότυπο πρόγραμμα εγκατάστασης για windows ή Macintosh.Το πρόγραμμα εγκατάστασης δημιουργεί όλους τους απαραίτητους καταλόγους και αρχεία στον σκληρό σας δίσκο που χρειάζονται για εκτέλεση του Dreamweaver.



Εικόνα 9 - Αρχική σελίδα του Dreamweaver

Η περιοχή εργασίας του Dreamweaver

Όταν ανοίγετε το Dreamweaver, για πρώτη φορά στα windows σας δίνετε η δυνατότητα να επιλέξετε αν θα χρησιμοποιήσετε την περιοχή εργασίας Designer ή την περιοχή εργασίας coder. Η περιοχή εργασίας coder κλειδώνει όλες τις παλέτες του Dreamweaver, στην αριστερή πλευρά της οθόνης, ενώ η περιοχή εργασίας Designer κλειδώνει όλες τις παλέτες στην δεξιά πλευρά της οθόνης. Η έκδοση Macintosh του Dreamweaver, σας εισάγει αυτόματα στην περιοχή εργασίας Designer. Αν επιλέξετε αρχικά την περιοχή εργασίας coder και μετά θέλετε να αλλάξετε την περιοχή εργασίας Designer, μπορείτε να το κάνετε επιλέγοντας window, workspace Layout, Designer.



Εικόνα 10 - Διεπαφή εργασίας του Dreamweaver

Κεφάλαιο 3. Προγραμματιστικά Εργαλεία

3.1 Σχεδίαση Βάσης Δεδομένων

3.1.1 Βάσεις Δεδομένων και τα πλεονεκτήματά τους

Μια βάση δεδομένων αποτελείται από κάποια συλλογή μόνιμων δεδομένων που χρησιμοποιούνται από τα συστήματα των εφαρμογών μιας δεδομένης επιχείρησης.

Μια βάση δεδομένων για να θεωρηθεί σωστά σχεδιασμένη και να μπορεί να λειτουργήσει με αποτελεσματικό τρόπο, θα πρέπει να διαθέτει ένα πλήθος χαρακτηριστικών τα πιο σημαντικά εν των οποίων είναι τα ακόλουθα:

- Δεν θα πρέπει να περιλαμβάνει περιττά δεδομένα (*redundant data*). Αυτό σημαίνει ότι τα ίδια δεδομένα δεν θα πρέπει να καταχωρούνται στη βάση δύο φορές. Εάν συμβαίνει κάτι τέτοιο, τότε ανακύπτουν δυο σοβαρά προβλήματα. Το πρώτο και προφανές πρόβλημα, είναι ότι σπαταλούμε άσκοπα αποθηκευτικό χώρο στο σκληρό δίσκο, αφού την ίδια πληροφορία την αποθηκεύουμε δύο φορές. Το δεύτερο και σοβαρότερο πρόβλημα, είναι ότι υπάρχει ο κίνδυνος δημιουργίας ασυνεπών δεδομένων (*inconsistent data*). Πράγματι, εάν κρατάμε δύο φορές την ίδια πληροφορία και η πληροφορία αυτή σε κάποια χρονική στιγμή υποστεί κάποιο είδος επεξεργασίας – για παράδειγμα την τροποποιήσουμε ή τη διαγράψουμε – τότε η επεξεργασία αυτή θα πρέπει να εφαρμοσθεί και στις δύο καταχωρήσεις που αφορούν το ίδιο δεδομένο, διότι διαφορετικά, η βάση θα περιέχει δεδομένα που δεν είναι συνεπή. Για το λόγο αυτό, ένας από τους πρώτους ελέγχους που πραγματοποιούμε στη βάση αμέσως μετά το σχεδιασμό της, είναι ο έλεγχος παρουσίας επαναλαμβανόμενων πεδίων, και η απομάκρυνσή τους, εφ' όσον υπάρχουν.
- Η βάση θα πρέπει να είναι σχεδιασμένη έτσι ώστε τα δεδομένα που περιλαμβάνει να ανακτώνται εύκολα και γρήγορα. Μια κακή σχεδίαση της δομής της βάσης, οδηγεί στη δημιουργία μιας βάσης η οποία είναι εξαιρετικά δυσκίνητη και αναποτελεσματική. Η σωστή σχεδίαση μιας βάσης δεδομένων, επιτυγχάνεται με εφαρμογή επί της δομής της βάσης, μιας τεχνικής, η οποία ονομάζεται κανονικοποίηση (*normalization*).
- Η βάση θα πρέπει να χαρακτηρίζεται από σύστημα ασφάλειας (*security system*) που να απαγορεύει την πρόσβαση στα δεδομένα μη εξουσιοδοτημένων ατόμων. Αυτό ισχύει κυρίως όπως θα δούμε, σε μεγάλες βάσεις δεδομένων με πολλούς χρήστες, και η τεχνική που συνήθως εφαρμόζεται, συνίσταται στον καθορισμό ομάδων χρηστών (*user groups*) με διαφορετικά δικαιώματα πρόσβασης στον καθένα από αυτούς. Ο κάθε χρήστης λαμβάνει ένα κωδικό πρόσβασης (*password*) και τα καθήκοντα που μπορεί να επιτελέσει είναι εντελώς συγκεκριμένα και καθορισμένα εκ των προτέρων.
- Η βάση θα πρέπει να μπορεί να διαχειρίζεται ταυτόχρονες προσπελάσεις πάνω στα ίδια δεδομένα (*concurrency control*). Ένα σύστημα κράτησης αεροπορικών εισιτηρίων που δέχεται ταυτόχρονες κλήσεις για κρατήσεις θέσεων, θα πρέπει να έχει τη δυνατότητα να αποτρέψει την καταχώρηση της ίδιας θέσης σε δύο άτομα που την έχουν ζητήσει ταυτόχρονα.
- Η βάση θα πρέπει να διαθέτει σύστημα δημιουργίας αντιγράφων ασφαλείας των δεδομένων που είναι καταχωρημένα σε αυτή (*backups*). Η ταυτόχρονη αποθήκευση των δεδομένων σε περισσότερους από ένα δίσκους, είναι μια εργασία επιβεβλημένη, προκειμένου να είναι δυνατή η ανάκτησή τους σε περιπτώσεις κατάρρευσης της βάσης για οποιοδήποτε λόγο.

Τα πλεονεκτήματα που προκύπτουν από τον κεντρικό έλεγχο των δεδομένων και κατ' επέκταση από τη χρήση των βάσεων δεδομένων συνοψίζονται στα παρακάτω:

- Ο πλεονασμός (redundancy) μπορεί να μειωθεί στο ελάχιστο
- Η ασυνέπεια μπορεί να αποφευχθεί (ως ένα βαθμό)
- Τα δεδομένα μπορούν να είναι κοινόχρηστα
- Μπορούν να επιβάλλονται πρότυπα
- Μπορούν να εφαρμόζονται περιορισμοί ασφάλειας
- Μπορεί να διατηρείται η ακεραιότητα
- Οι αντικρουόμενες απαιτήσεις μπορούν να εξισορροπούνται

3.1.2 Σχεσιακές Βάσεις Δεδομένων

Με τον όρο σχεσιακή βάση δεδομένων, ονομάζουμε τις βάσεις δεδομένων στις οποίες τα δεδομένα είναι αποθηκευμένα σε πίνακες (tables), οι οποίοι σχετίζονται (συνδέονται) με ένα συγκεκριμένο τρόπο μεταξύ τους. Στις σχεσιακές βάσεις δεδομένων, λόγω της σχέσης (σύνδεσης) που υπάρχει μεταξύ των πινάκων αποφεύγεται η διπλή εισαγωγή των ίδιων δεδομένων από περισσότερους από έναν πίνακες.

Πίνακες

Οι σχεσιακές βάσεις δεδομένων αποτελούνται από σχέσεις, που συνήθως ονομάζονται πίνακες (tables). Ένας πίνακας είναι ακριβώς αυτό που αποδεικνύει το όνομά του, ένας πίνακας με δεδομένα.

Στήλες

Κάθε στήλη του πίνακα έχει μοναδικό όνομα και περιέχει διαφορετικά δεδομένα. Κάθε στήλη έχει ένα σχετικό τύπο δεδομένων.

Γραμμές

Κάθε γραμμή του πίνακα αντιπροσωπεύει ένα διαφορετικό πελάτη. Εξ αιτίας της μορφής ενός πίνακα, όλες οι γραμμές έχουν τις ίδιες ιδιότητες. Οι γραμμές ονομάζονται επίσης εγγραφές.

Τιμές

Κάθε γραμμή του πίνακα αποτελείται αντιπροσωπεύει από ένα σύνολο ξεχωριστών τιμών που αντιστοιχούν σε στήλες. Κάθε τιμή πρέπει να έχει έναν τύπο δεδομένων ο οποίος καθορίζεται από την στήλη της.

Πρωτεύον κλειδί

Οι βάσεις δεδομένων συνήθως αποτελούνται από πολλούς πίνακες και χρησιμοποιούν ένα κλειδί για συσχέτιση ενός πίνακα με έναν άλλον. Πρωτεύον κλειδί ενός πίνακα είναι το πεδίο (ή το σύνολο των πεδίων) που προσδιορίζει μοναδικά κάθε εγγραφή του πίνακα.

Από την στιγμή που θα καθορίσουμε το πρωτεύον κλειδί σε έναν πίνακα, δεν θα επιτρέπεται να καταχωρηθούν σε αυτό διπλότυπες τιμές ή κενές τιμές (NULL) στο πεδίο (ή στα πεδία) του πρωτεύοντος κλειδιού. Κάθε πίνακας μιας βάσης δεδομένων μπορεί να έχει ένα και μόνο πρωτεύον κλειδί.

Πρέπει να είμαστε προσεκτικοί κατά την επιλογή του πεδίου που θα ορίσουμε ως πρωτεύον κλειδί του πίνακα. Αν για παράδειγμα στον πίνακα 'πελάτες' της Εικόνας 1.1 επιλέξουμε το πεδίο Επώνυμο ως πρωτεύον κλειδί του πίνακα, δεν θα επιτρέπεται η εισαγωγή στον πίνακα δύο πελάτες (δύο εγγραφές) με το ίδιο επώνυμο. Μπορούμε όμως να ορίσουμε ως πρωτεύον κλειδί στον πίνακα, το πεδίο ΑΔΤ, μια και είναι βέβαιο ότι κάθε πελάτης θα έχει μοναδικό αριθμό αστυνομικής ταυτότητας. Αυτό βέβαια με την προϋπόθεση ότι ο πίνακας περιέχει το πεδίο ΑΔΤ. Εάν δεν το έχει θα πρέπει να διαλέξουμε κάποιο άλλο πεδίο.

Σχήματα

Η πλήρης σχεδίαση των πινάκων μιας βάσης δεδομένων ονομάζεται σχήμα (schema). Ένα σχήμα θα πρέπει να δείχνει τους πίνακες μαζί με τις στήλες τους, τους τύπους δεδομένων των στηλών και να υποδεικνύει το πρωτεύον κλειδί κάθε πίνακα και τα ξένα κλειδιά τους.

Σχέσεις

Τα ξένα κλειδιά αντιπροσωπεύουν μία σχέση μεταξύ των δεδομένων δύο πινάκων. Για παράδειγμα, η σύνδεση από τον πίνακα 'Δρομολόγιο' στον πίνακα 'Πελάτες' αντιπροσωπεύει μια σχέση μεταξύ μιας γραμμής του πίνακα 'Δρομολόγιο' και μιας γραμμής του πίνακα 'Πελάτες'. Υπάρχουν τρία βασικά είδη σχέσεων σε μια σχεσιακή βάση δεδομένων:

1. Σχέση ένα-προς ένα: υπάρχει ένα στοιχείο μόνο σε κάθε πλευρά της σχέσης.
2. Σχέση ένα-προς πολλά: μία γραμμή ενός πίνακα είναι συνδεδεμένη με πολλές γραμμές του άλλου πίνακα. Σε αυτές τις σχέσεις, ο πίνακας που περιέχει τις πολλές γραμμές θα έχει ένα ξένο κλειδί που τον συνδέει με τον πίνακα με την μία γραμμή.
3. Σχέση πολλά-προς πολλά: μία γραμμή ενός πίνακα είναι συνδεδεμένη με πολλές γραμμές ενός δεύτερου πίνακα και μια γραμμή του δεύτερου πίνακα είναι συνδεδεμένη με πολλές γραμμές του πρώτου πίνακα.

3.2 Δομημένη Γλώσσα Επερωτήσεων (Structured Query Language, SQL)

Η SQL είναι η πιο ευρέως χρησιμοποιούμενη εμπορικά γλώσσα ερωτημάτων και χρησιμοποιεί ένα συνδυασμό από σχεσιακή άλγεβρα και σχεσιακό λογισμό. Αν και αναφερόμαστε στην SQL ως "γλώσσα ερωτημάτων", μπορεί ωστόσο να κάνει περισσότερα από ένα απλό ερώτημα σε μια βάση δεδομένων. Μπορεί να ορίσει τη δομή των δεδομένων, να τροποποιήσει τα δεδομένα της βάσης δεδομένων και να καθορίσει περιορισμούς ασφάλειας.

Η γλώσσα SQL έχει διάφορα μέρη:

- **Data Definition Language (DDL):** Το SQL DDL παρέχει εντολές για ορισμό σχεσιακών σχημάτων, διαγραφή σχέσεων και τροποποίηση σχέσεων σχημάτων.
- **Interactive data-manipulation language (DML):** Το SQL DML περιλαμβάνει μια γλώσσα ερωτημάτων που βασίζεται στη σχεσιακή άλγεβρα και στο σχεσιακό λογισμό. Περιλαμβάνει επίσης εντολές για εισαγωγή εγγραφών, διαγραφή εγγραφών και τροποποίηση εγγραφών της βάσης δεδομένων.
- **Προβολή ορισμού:** Το SQL DDL περιλαμβάνει εντολές για ορισμό προβολών.
- **Έλεγχος συναλλαγών:** Η SQL περιλαμβάνει εντολές για καθορισμό της αρχής και του τέλους των συναλλαγών.

- Ενσωματωμένη SQL και δυναμική SQL: Η ενσωματωμένη και η δυναμική SQL ορίζουν πως μπορούν να ενσωματωθούν οι SQL εντολές μέσα σε γενικές γλώσσες προγραμματισμού, όπως οι C, C++, Java, PL/1, Cobol, Pascal και Fortran.
- Ακεραιότητα: Το SQL DDL περιλαμβάνει εντολές για καθορισμό περιορισμών ακεραιότητας που πρέπει να ικανοποιούν τα δεδομένα που είναι αποθηκευμένα στη βάση δεδομένων. Δεν επιτρέπονται ενημερώσεις που παραβιάζουν τους περιορισμούς ακεραιότητας.
- Πιστοποίηση: Το SQL DDL περιλαμβάνει εντολές για καθορισμό δικαιωμάτων για σχέσεις και προβολές.

3.2.1 Βασική Δομή

Μια σχεσιακή βάση δεδομένων αποτελείται από ένα σύνολο από σχέσεις, κάθε μια εκ των οποίων έχει ένα μοναδικό όνομα. Η SQL επιτρέπει τη χρήση κενών τιμών για να δείξει ότι η τιμή είτε είναι άγνωστη είτε δεν υπάρχει. Η βασική δομή μιας παράστασης SQL αποτελείται από τρεις όρους: `select`, `from` και `where`.

- Ο όρος `select` αντιστοιχεί στην πράξη προβολής (projection) της σχεσιακής άλγεβρας. Χρησιμοποιείται για να εμφανίζει τις ιδιότητες που είναι επιθυμητές στο αποτέλεσμα ενός ερωτήματος. Ας δούμε ένα απλό ερώτημα χρησιμοποιώντας το παράδειγμα της τράπεζας "Βρείτε τα ονόματα όλων των υποκαταστημάτων στη σχέση `loan`"

```
select branch-name  
from loan
```

Στις περιπτώσεις που θέλουμε να απαλείψουμε τα διπλότυπα, εισάγουμε τη λέξη κλειδί `distinct` μετά τη `select`. Μπορούμε να ξαναγράψουμε το προηγούμενο ερώτημα ως

```
select distinct branch-name  
from loan
```

αν θέλουμε να αφαιρεθούν τα διπλότυπα.

Η SQL μας επιτρέπει να χρησιμοποιούμε τη λέξη κλειδί `all` αν θέλουμε να καθορίσουμε άμεσα ότι δεν αφαιρούνται τα διπλότυπα:

```
select all branch-name  
from loan
```

Το σύμβολο του αστερίσκου "*" μπορεί να χρησιμοποιηθεί για να δηλώσει "όλες τις ιδιότητες". Έτσι, η χρήση του `loan.*` στην προηγούμενη `select` θα έδειχνε ότι όλες οι ιδιότητες του `loan` θα επιλεγθούν. Μια `select` της μορφής `select *` δείχνει ότι όλες οι ιδιότητες όλων των σχέσεων που εμφανίζονται στην εντολή `from` επιλέγονται.

- Ο όρος `from` αντιστοιχεί στην πράξη του καρτεσιανού γινομένου της σχεσιακής άλγεβρας. Εμφανίζει τις σχέσεις που θα σαρωθούν για τον υπολογισμό της παράστασης. Αφού ο φυσικός σύνδεσμος ορίζεται σε σχέση με ένα καρτεσιανό γινόμενο, μια επιλογή και μια προβολή, είναι σχετικά από θέμα να γράψουμε μια SQL παράσταση για το φυσικό σύνδεσμο. Το ερώτημα "Για όλους τους πελάτες που έχουν ένα δάνειο από την τράπεζα, βρείτε τα ονόματά τους, τους αριθμούς δανείων και το ποσό τους", στην SQL μπορεί να γραφεί ως

```
select customer-name, borrower.loan-number, amount  
from borrower, loan
```

where borrower.loan-number = loan.loan-number

- Ο όρος where αντιστοιχεί στο κατηγορήμα επιλογής της σχεσιακής άλγεβρας. Αποτελείται από ένα κατηγορήμα που περιλαμβάνει ιδιότητες των σχέσεων που εμφανίζονται στον όρο from. Ας δούμε τη χρήση της where στην SQL. Σκεφτείτε τώρα το: "Βρείτε όλους τους αριθμούς δανείων για τα δάνεια που έγιναν στο υποκατάστημα Perryridge με ποσά μεγαλύτερα από 1200€". Αυτό το ερώτημα μπορεί να γραφεί στην SQL ως εξής:

```
select loan-number
```

```
from loan
```

```
where branch-name = 'Perryridge' and amount > 1200
```

Η SQL χρησιμοποιεί στην where τους λογικούς συνδέσμους and, or και not, αντί για τα μαθηματικά σύμβολα \wedge , \vee και \neg . Οι τελεστές των λογικών συνδέσμων μπορεί να είναι παραστάσεις που περιλαμβάνουν τους τελεστές σύγκρισης $<$, $<=$, $>$, $>=$, $=$ και $<>$. Η SQL μας επιτρέπει να χρησιμοποιούμε τους τελεστές σύγκρισης για να συγκρίνουμε συμβολοσειρές και αριθμητικές παραστάσεις, όπως επίσης και ειδικούς τύπους, όπως τύπους ημερομηνιών.

Η SQL περιλαμβάνει έναν τελεστή σύγκρισης between για να απλοποιήσει τις where που καθορίζουν ότι μια τιμή είναι μικρότερη ή ίση με κάποια τιμή και μεγαλύτερη ή ίση με κάποια άλλη. Αν θέλουμε να δούμε τον αριθμό δανείου των δανείων με ποσά μεταξύ 90.000€ και 100.000€, μπορούμε να χρησιμοποιήσουμε τη σύγκριση between για να γράψουμε

```
select loan-number
```

```
from loan
```

```
where amount between 90000 and 100000
```

Παρόμοια, μπορούμε να χρησιμοποιήσουμε τον τελεστή not between.

Η SQL παρέχει ένα μηχανισμό για τη μετονομασία σχέσεων και ιδιοτήτων. Χρησιμοποιεί τον όρο as, που παίρνει τη μορφή:

```
old-name as new-name
```

Η as μπορεί να εμφανιστεί στις select και from. Για παράδειγμα, αν θέλουμε το όνομα της ιδιότητας loan-number να αντικατασταθεί με το loan-id, μπορούμε να ξαναγράψουμε το προηγούμενο ερώτημα ως

```
select customer-name, borrower.loan-number as loan-id, amount
```

```
from borrower, loan
```

```
where borrower.loan-number = loan.loan-number
```

3.2.2 Πράξεις Συνόλων

Οι SQL τελεστές Union, intersect και except λειτουργούν πάνω σε σχέσεις και αντιστοιχούν με τους τελεστές της σχεσιακής άλγεβρας \cup , \cap και $-$. Όπως και η ένωση, η τομή και η διαφορά στη σχεσιακή άλγεβρα, έτσι και οι σχέσεις που συμμετέχουν στις πράξεις πρέπει να είναι συμβατές, δηλαδή, πρέπει να έχουν το ίδιο σύνολο ιδιοτήτων.

Ο Τελεστής Ένωσης

Για να βρούμε όλους τους πελάτες που έχουν πάρει ένα δάνειο, ένα λογαριασμό ή και τα δύο από την τράπεζα, γράφουμε

```
(select customer-name  
from depositor)  
union  
(select customer-name  
from borrower)
```

Η πράξη union αυτόματα απαλείφει τα διπλότυπα, αντίθετα με τη select. Έτσι, στο προηγούμενο ερώτημα, αν ένας πελάτης, ας πούμε ο Jones, έχει διάφορους λογαριασμούς ή δάνεια (ή και τα δύο) στην τράπεζα, τότε ο Jones θα εμφανιστεί μόνο μια φορά στο αποτέλεσμα.

Ο Τελεστής Τομής

Για να βρούμε όλους τους πελάτες που έχουν ένα δάνειο και ένα λογαριασμό στην τράπεζα, γράφουμε

```
(select customer-name  
from depositor)  
intersect  
(select customer-name  
from borrower)
```

Η πράξη intersect απαλείφει αυτόματα τα διπλότυπα. Έτσι, στο προηγούμενο ερώτημα, αν ένας πελάτης, ας πούμε ο Jones, έχει διάφορους λογαριασμούς και δάνεια με την τράπεζα, τότε ο Jones θα εμφανίζεται μια φορά το αποτέλεσμα.

Ο Τελεστής Εξαίρεσης

Για να βρούμε όλους τους πελάτες που έχουν έναν λογαριασμό αλλά όχι δάνειο στην τράπεζα, γράφουμε

```
(select distinct customer-name  
from depositor)  
except  
(select customer-name  
from borrower)
```

Στην πράξη except απαλείφονται αυτόματα τα διπλότυπα. Έτσι, στο προηγούμενο ερώτημα, θα εμφανιστεί μια εγγραφή με το όνομα πελάτη Jones (ακριβώς μια) στο αποτέλεσμα, μόνο αν ο Jones έχει ένα λογαριασμό στην τράπεζα αλλά όχι δάνειο.

3.2.3 Συνοπτικές Συναρτήσεις

Οι συνοπτικές συναρτήσεις είναι συναρτήσεις που παίρνουν μια συλλογή (ένα σύνολο ή πολλαπλό σύνολο) από τιμές ως είσοδο και επιστρέφουν μια μόνο τιμή. Η SQL προσφέρει πέντε ενσωματωμένες συνοπτικές συναρτήσεις:

- Μέσος όρος: avg

- Ελάχιστο: min
- Μέγιστο: max
- Άθροισμα: sum
- Καταμέτρηση: count

Η είσοδος στο sum και στο avg πρέπει να είναι μια συλλογή από αριθμούς, αλλά οι άλλοι τελεστές μπορούν να λειτουργήσουν σε συλλογές μη αριθμητικών τύπων δεδομένων, όπως σε συμβολοσειρές.

Φανταστείτε το ερώτημα "Βρείτε το μέσο όρο των υπόλοιπων των λογαριασμών του υποκαταστήματος Perryridge". Γράφουμε αυτό το ερώτημα ως εξής:

```
select avg(balance)
from account
where branch-name = 'Perryridge'
```

Το αποτέλεσμα αυτού του ερωτήματος είναι μια σχέση με μια μόνο ιδιότητα, που περιέχει μια εγγραφή με μια αριθμητική τιμή που αντιστοιχεί στο μέσο όρο των ποσών του υποκαταστήματος. Προαιρετικά, μπορούμε να δώσουμε ένα όνομα στην ιδιότητα της σχέσης του αποτελέσματος, χρησιμοποιώντας την as.

Υπάρχουν περιπτώσεις που θα θέλαμε να εφαρμόσουμε τη συνοπτική συνάρτηση όχι μόνο σε ένα σύνολο από εγγραφές, αλλά επίσης και σε μια ομάδα από σύνολα εγγραφών. Το καθορίζουμε αυτό στην SQL χρησιμοποιώντας το group by. Η ιδιότητα ή οι ιδιότητες που δίνονται στο group by χρησιμοποιούνται για να σχηματίσουν ομάδες. Οι εγγραφές με την ίδια τιμή σε όλες τις ιδιότητες του group by τοποθετούνται σε μια ομάδα.

Φανταστείτε το ερώτημα "Βρείτε το μέσο όρο του υπόλοιπου κάθε υποκαταστήματος". Γράφουμε αυτό το ερώτημα ως εξής:

```
select branch-name, avg(balance)
from account
group by branch-name
```

3.2.4 Κενές Τιμές

Η SQL επιτρέπει τη χρήση κενών τιμών για να δείξει απουσία πληροφοριών στην τιμή μιας ιδιότητας.

Μπορούμε να χρησιμοποιήσουμε την ειδική λέξη κλειδί null σε ένα κατηγορημα για να ελέγξουμε για κενή τιμή. Έτσι, για να βρούμε όλους τους αριθμούς δανείων που εμφανίζονται στη σχέση loan με κενές τιμές για το ποσό, γράφουμε

```
select loan-number
from loan
where amount is null
```

Το κατηγορημα is not null ελέγχει για την απουσία μιας κενής τιμής.

3.2.5 Τροποποίηση της Βάσης Δεδομένων

Έχουμε περιορίσει την προσοχή μας μέχρι τώρα στην εξαγωγή πληροφοριών από την βάση δεδομένων. Τώρα, δείχνουμε πως να προσθέτετε, να αφαιρείτε ή να αλλάζετε πληροφορίες με την SQL.

Διαγραφή

Μια αίτηση διαγραφής εκφράζεται με τον ίδιο τρόπο όπως ένα ερώτημα. Μπορούμε να διαγράψουμε μόνο ολόκληρη εγγραφή και δεν μπορούμε να διαγράψουμε τιμές μόνο σε συγκεκριμένες ιδιότητες. Η SQL εκφράζει μια διαγραφή με το

```
delete from r  
where P
```

όπου το P αντιπροσωπεύει ένα κατηγορημα και το r μια σχέση. Η εντολή delete βρίσκει πρώτα όλες τις εγγραφές t στο r για τις οποίες το P(t) είναι αληθές και μετά τις διαγράφει από το r. Ο όρος where μπορεί να παραλειφθεί, οπότε τότε διαγράφονται όλες οι εγγραφές στο r.

Παρατηρήστε ότι μια εντολή delete λειτουργεί σε μια μόνο σχέση. Αν θέλουμε να διαγράψουμε εγγραφές από διάφορες σχέσεις, μπορούμε να χρησιμοποιήσουμε μια εντολή delete για κάθε σχέση.

Το κατηγορημα στην where μπορεί να είναι περίπλοκο, όπως μια εντολή where της select. Από την άλλη άκρη, η where μπορεί να είναι κενή. Η αίτηση

```
delete from loan
```

διαγράφει όλες τις εγγραφές από τη σχέση loan. Εδώ υπάρχουν παραδείγματα των SQL αιτήσεων διαγραφής:

- Διαγραφή όλων των λογαριασμών στο υποκατάστημα Perryridge

```
delete from account  
where branch-name = 'Perryridge'
```

- Διαγραφή όλων των δανείων με ποσά μεταξύ 1300€ και 1500€

```
delete from loan  
where amount between 1300 and 1500
```

- Διαγραφή όλων των λογαριασμών σε κάποιο υποκατάστημα που βρίσκεται Needham

```
delete from account
where branch-name in (select branch-name
                      from branch
                      where branch-city = 'Needham')
```

Εισαγωγή

Για να εισάγουμε δεδομένα σε μια σχέση, καθορίζουμε είτε την εγγραφή που θα εισαχθεί είτε γράφουμε ένα ερώτημα του οποίου το αποτέλεσμα είναι ένα σύνολο από εγγραφές που θα εισαχθούν. Προφανώς, οι τιμές των ιδιοτήτων για τις εγγραφές που εισάγονται πρέπει να είναι μέλη του πεδίου τιμών της ιδιότητας. Παρόμοια, οι εγγραφές που εισάγονται πρέπει να έχουν την σωστή τάξη.

Η απλούστερη εντολή insert είναι μια αίτηση να εισαχθεί μια εγγραφή. Υποθέστε ότι θέλουμε να εισάγουμε το γεγονός ότι υπάρχει ένας λογαριασμός A-9732 στο υποκατάστημα Perryridge και ότι έχει υπόλοιπο 1200€. Γράφουμε

```
insert into account
values ('A-9732', 'Perryridge ', 1200)
```

Σε αυτό το παράδειγμα, οι τιμές καθορίζονται με τη σειρά, με την οποίαν εμφανίζονται οι αντίστοιχες ιδιότητες στο σχήμα. Προς χάρη των χρηστών που μπορεί να θυμούνται την σειρά των ιδιοτήτων, η SQL επιτρέπει να καθοριστούν οι ιδιότητες ως μέρος της εντολής insert. Για παράδειγμα, οι παρακάτω SQL εντολές Insert είναι ίδιες σε λειτουργία με την προηγούμενη:

```
insert into account (account-number, branch-name, balance)
values ('A-9732', 'Perryridge ', 1200)
```

```
insert into account (branch-name , account-number, balance)
values ('Perryridge ', 'A-9732', 1200)
```

Ενημερώσεις

Σε κάποιες περιπτώσεις, μπορεί να θέλουμε να αλλάξουμε την τιμή μιας εγγραφής χωρίς να αλλάξουμε όλες τις τιμές της. Για το σκοπό αυτό, μπορεί να χρησιμοποιηθεί η εντολή update. Όπως μπορούμε για την insert και την delete, έτσι μπορούμε να επιλέξουμε και εγγραφές που θα ενημερωθούν, χρησιμοποιώντας ένα ερώτημα.

Υποθέστε ότι γίνονται οι ετήσιες πληρωμές του επιτοκίου και όλα τα υπόλοιπα αυξάνονται κατά 5%. Γράφουμε

```
update account
set balance = balance * 1.05
```

Η προηγούμενη εντολή update εφαρμόζεται μόνο μια φορά σε κάθε μια από τις εγγραφές στη σχέση account.

Αν το επιτόκιο θα πληρωθεί μόνο σε λογαριασμούς με υπόλοιπο 1000€ ή περισσότερο, μπορούμε να γράψουμε

```
update account  
set balance = balance * 1.05  
where balance >= 1000
```


4.1.2 Σχέσεις και Ανάλυση των Σχέσεων

- **Ξενοδοχείο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα ξενοδοχεία της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα
 2. Διεύθυνση
 3. Κατηγορία
 4. Τηλέφωνο
 5. E-mail
- **Κράτηση.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τις κρατήσεις των δωματίων. Είναι ασθενής οντότητα γιατί η ύπαρξη της εξαρτάται από την ύπαρξη ενός ξενοδοχείου. Έχει τα εξής χαρακτηριστικά:
 1. Κωδικός αριθμός κράτησης
 2. Όνομα Ξενοδοχείου
 3. Είδος (Μονόκλινο/Δίκλινο/Τρίκλινο)
 4. Ημερομηνία_άφιξης
 5. Ημερομηνία_αναχώρησης
 6. Κατηγορία ξενοδοχείου
 7. Μέσο μεταφοράς
 8. Τόπος αναχώρησης
 9. Τόπος άφιξης
 10. Ημερομηνία αναχώρησης
 11. Ημερομηνία επιστροφής
- **Δωμάτιο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με το δωμάτιο ενός ξενοδοχείου. Είναι ασθενής οντότητα γιατί η ύπαρξη της εξαρτάται από την ύπαρξη ενός ξενοδοχείου. Έχει τα εξής χαρακτηριστικά:
 12. Όνομα Ξενοδοχείου
 13. Κωδικός αριθμός δωματίου
 14. Είδος (Μονόκλινο/Δίκλινο/Τρίκλινο)
 15. Τιμή
- **Εικόνες_Ξεν.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τη φωτογραφία ενός ξενοδοχείου. Είναι πλειότιμο πεδίο γιατί ένα ξενοδοχείο μπορεί να προβάλλει περισσότερες από μια φωτογραφίες. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα ξενοδοχείου
 2. Όνομα αρχείου/εικόνας
- **Χάρτης.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τους χάρτες της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα περιοχής
 2. Όνομα αρχείου/εικόνας
- **Αξιοθέατο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα αξιοθέατα της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα

2. Διεύθυνση
 3. Περιγραφή
 4. Όνομα αρχείου/εικόνας
- **Δρομολόγιο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα δρομολόγια από, προς και εντός της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Κωδικός αριθμός δρομολογίου
 2. Μέσο μεταφοράς
 3. Τιμή
 4. Εταιρία
 5. Ημερομηνία αναχώρησης
 6. Ημερομηνία άφιξης
 7. Σημείο αναχώρησης
 8. Σημείο άφιξης
 9. Θέσεις
 - **Εστιατόριο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα εστιατόρια της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα
 2. Είδος
 3. Διεύθυνση
 4. Τηλέφωνο
 5. Μέση Τιμή
 - **Γραφείο_ενοικ.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα γραφεία ενοικιάσεως της Ρόδου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα
 2. Διεύθυνση
 3. Τηλέφωνο
 - **Αυτοκίνητο.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα αυτοκίνητα που μπορεί να ενοικιάζει κάποια γραφείο ενοικιαζόμενων οχημάτων. Είναι ασθενής οντότητα γιατί η ύπαρξη της εξαρτάται από την ύπαρξη ενός γραφείου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα γραφείου
 2. Αριθμός κυκλοφορίας αυτοκινήτου
 3. Εταιρία
 4. Κυβικά
 5. Χρώμα
 6. Χρέωση
 7. Κατηγορία
 8. Θέσεις
 - **Μηχανάκι.** Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα μηχανάκια που μπορεί να ενοικιάζει κάποια γραφείο ενοικιαζόμενων οχημάτων. Είναι ασθενής οντότητα γιατί η ύπαρξη της εξαρτάται από την ύπαρξη ενός γραφείου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα γραφείου

2. Αριθμός κυκλοφορίας μηχανής
 3. Κυβικά
 4. Χρέωση
 5. Κατηγορία
- Εκδρομικό_γραφ. Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τα εκδρομικά γραφεία που δραστηριοποιούνται στη Ρόδο. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα
 2. Τηλέφωνο
 3. Διεύθυνση
 - Εκδρομή. Είναι ο πίνακας που κρατάει την πληροφορία που είναι σχετική με τις εκδρομές που μπορεί να προσφέρει ένα ταξιδιωτικό γραφείο. Είναι ασθενής οντότητα γιατί η ύπαρξη της εξαρτάται από την ύπαρξη ενός τέτοιου γραφείου. Έχει τα εξής χαρακτηριστικά:
 1. Όνομα γραφείου
 2. Κωδικός αριθμός εκδρομής
 3. Προορισμός
 4. Αφετηρία
 5. Ημερομηνία αναχώρησης
 6. Ημερομηνία άφιξης
 7. Άτομα

4.1.3 Σχεσιακό Μοντέλο

Από το παραπάνω μοντέλο οντοτήτων-σχέσεων, προκύπτουν οι ακόλουθοι πίνακες της Βάσης Δεδομένων.

Τα πρωτεύοντα κλειδιά του κάθε πίνακα είναι υπογραμμισμένα με απλή υπογράμμιση (πχ. Πίνακας Εκδήλωση: Όνομα), ενώ τα δευτερεύοντα με διπλή (πχ. Πίνακας Παρουσιάζει: Όνομα Αξιοθέατου και Όνομα Εκδήλωσης). Δευτερεύοντα είναι τα κλειδιά που είναι πρωτεύοντα σε άλλους πίνακες.

ΤΕΙ Κρήτης, Εφαρμοσμένη Πληροφορική και Πολυμέσα

Ξενοδοχείο				
<u>Όνομα</u>	Διεύθυνση	Κατηγορία	Τηλέφωνο	email

Κράτηση										
<u>id</u>	<u>Όνομα</u>	τύπος	Ημερ_αναχώρησης	Ημερ_άφιξης	Κατηγορία	Μέσο	Από	Προς	Αναχώρηση	Επιστροφή

Δωμάτιο			
<u>Όνομα_Ξεν</u>	Κωδ.Αριθμός	Είδος	Τιμή

Εικόνες_Ξεν	
<u>Όνομα_Ξεν</u>	Εικόνα

Χάρτης	
<u>Όνομα</u>	Εικόνα

Αξιοθέατο			
<u>Όνομα</u>	Διεύθυνση	Περιγραφή	Εικόνα

Δρομολόγιο								
<u>id</u>	Μέσο_Μεταφοράς	Τιμή	Εταιρία	Ημερ_αναχώρησης	Ημερ_άφιξης	Σ_αναχ	Σ_αφ	Θέσεις

Εστιατόριο				
<u>Όνομα</u>	Είδος	Διεύθυνση	Τηλέφωνο	Μέση_Τιμή

Γραφείο_ενοικ		
<u>Όνομα</u>	Διεύθυνση	Τηλέφωνο

Αυτοκίνητο							
<u>Όνομα_γραφείου</u>	<u>Αριθμ_Κυκλ</u>	Εταιρία	Κυβικά	Χρώμα	Χρέωση	Κατηγορία	Θέσεις

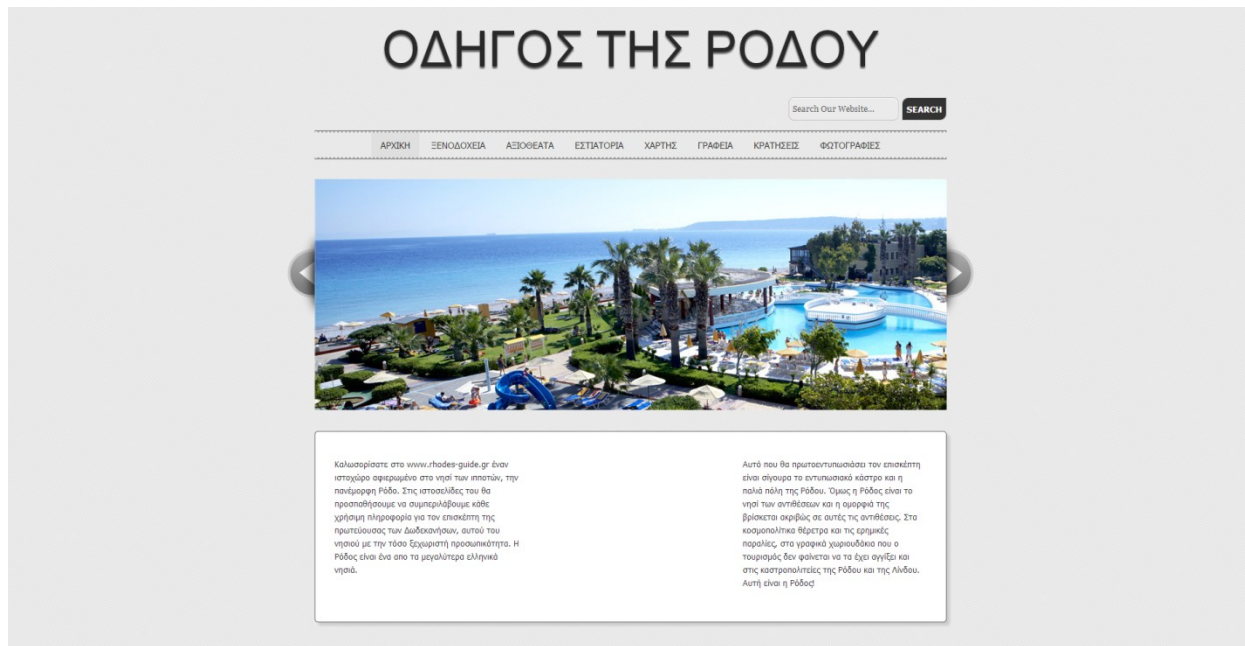
Μηχανάκι				
<u>Όνομα_γραφείου</u>	<u>Αριθμ_Κυκλ</u>	Κυβικά	Χρέωση	Κατηγορία

Εκδρομικό_γραφ		
<u>Όνομα</u>	Τηλέφωνο	Διεύθυνση

Εκδρομή						
<u>Όνομα_γραφ</u>	<u>id</u>	Προορισμός	Αφετηρία	Ημερ_αναχ	Ημερ_αφ	άτομα

4.2 Περιγραφή της Εφαρμογής

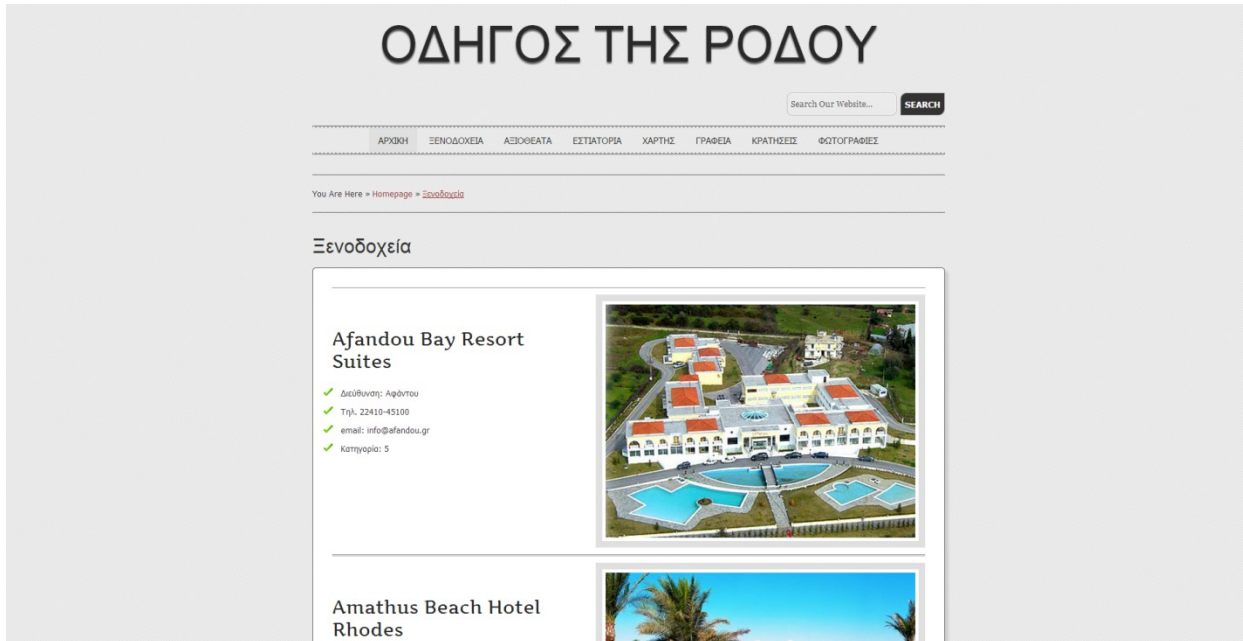
- Αρχική



Εικόνα 11 - Αρχική σελίδα

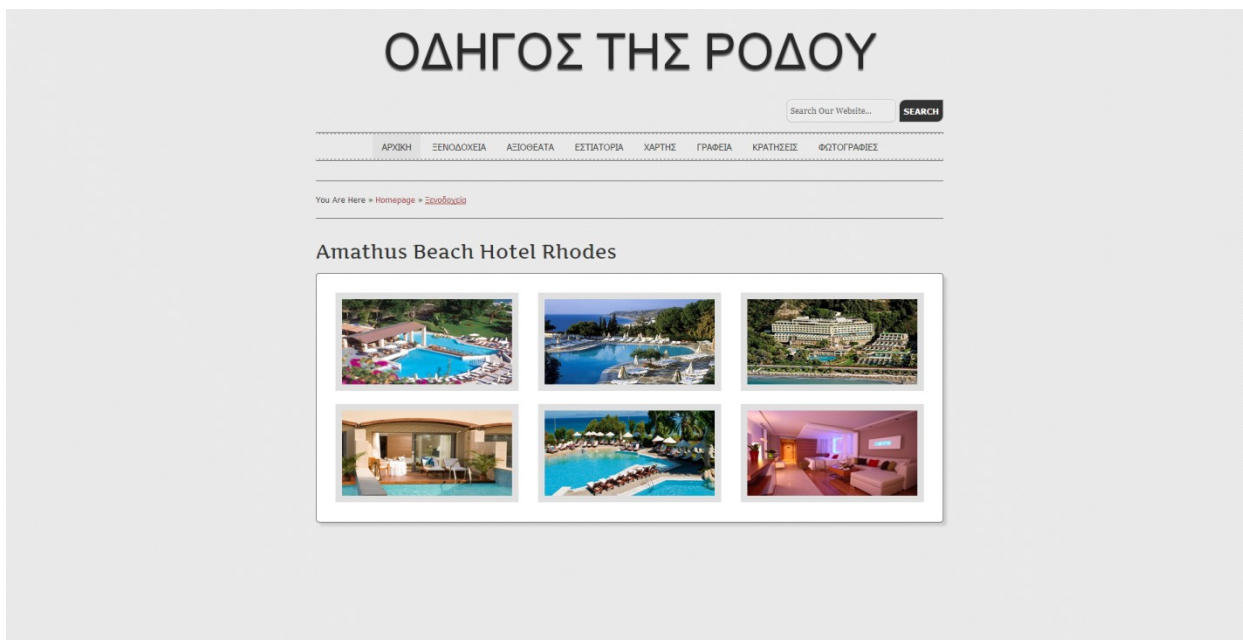
Στην Αρχική σελίδα ο χρήστης του συστήματος μπορεί να αποκτήσει μια σύντομη πρώτη εικόνα για το νησί της Ρόδου. Αποτελείται από το λογότυπο της ιστοσελίδας, το μενού επιλογών, μερικές χαρακτηριστικές φωτογραφίες από τα αξιοθέατα της Ρόδου και μια σύντομη περιγραφή του νησιού.

- Ξενοδοχεία



Εικόνα 12 - Προβολή λίστας καταχωρημένων ξενοδοχείων

Στη σελίδα των Ξενοδοχείων ο χρήστης μπορεί να βρει πληροφορία σχετική με τα ξενοδοχεία της Ρόδου κάθε κατηγορίας. Προβάλλεται μια λίστα με τα ονόματα των ξενοδοχείων, τη διεύθυνση τους, τρόπους επικοινωνίας (τηλέφωνο και e-mail) καθώς επίσης και την κατηγορία του κάθε ξενοδοχείου.



Εικόνα 13 - Προβολή φωτογραφιών ξενοδοχείου



Εικόνα 14 - Προβολή φωτογραφίας

Ο χρήστης μπορεί να επιλέξει το όνομα κάποιου ξενοδοχείου ώστε να πλοηγηθεί ανάμεσα στις φωτογραφίες του.

- **Αξιοθέατα**

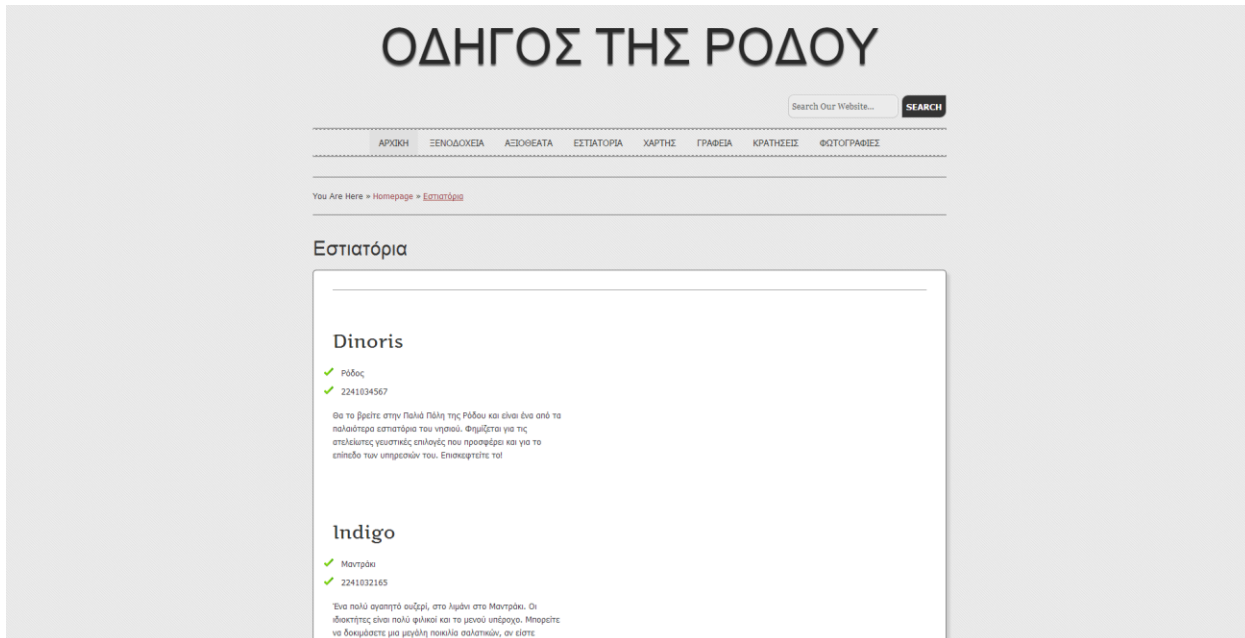


Εικόνα 15 - Προβολή αξιοθέατων Ρόδου

Η επιλογή "Αξιοθέατα" προβάλλει μια λίστα με τις φωτογραφίες των αξιοθέατων της Ρόδου. Το κάθε αξιοθέατο αντιπροσωπεύεται από μια φωτογραφία και προβάλλονται σε τριάδες. Επιλέγοντας ο χρήστης τη φωτογραφία

μπορεί να τη δει σε μεγέθυνση, ενώ επιλέγοντας το σύνδεσμο View More Information οδηγείται σε μια σελίδα που περιέχει μια αναλυτική περιγραφή του αξιοθέατου. Η περιγραφή αυτή αφορά μια ιστορική αναδρομή, την τοποθεσία και άλλα απαραίτητα στοιχεία για την ενημέρωση του επισκέπτη.

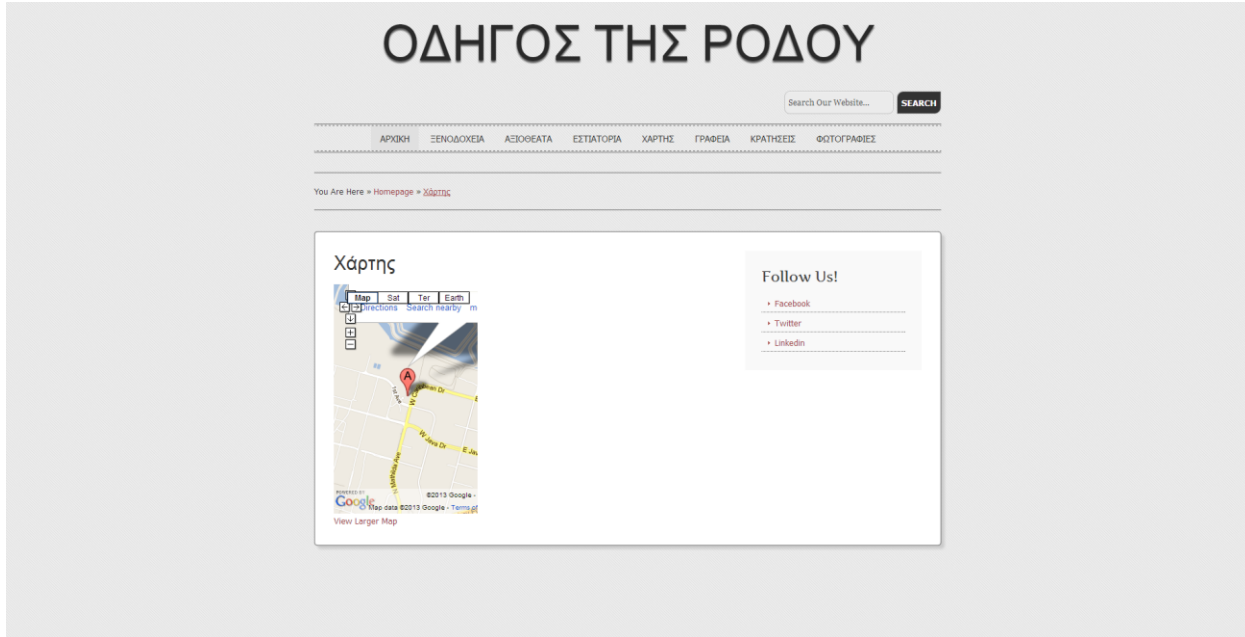
- **Εστιατόρια**



Εικόνα 16 - Προβολή λίστας καταχωρημένων εστιατορίων

Η επιλογή "Εστιατόρια" προβάλλει τη λίστα των εστιατορίων της Ρόδου με τη διεύθυνση στην οποία βρίσκονται, ένα τηλέφωνο επικοινωνίας και μια σύντομη περιγραφή για την κουζίνα του κάθε εστιατορίου.

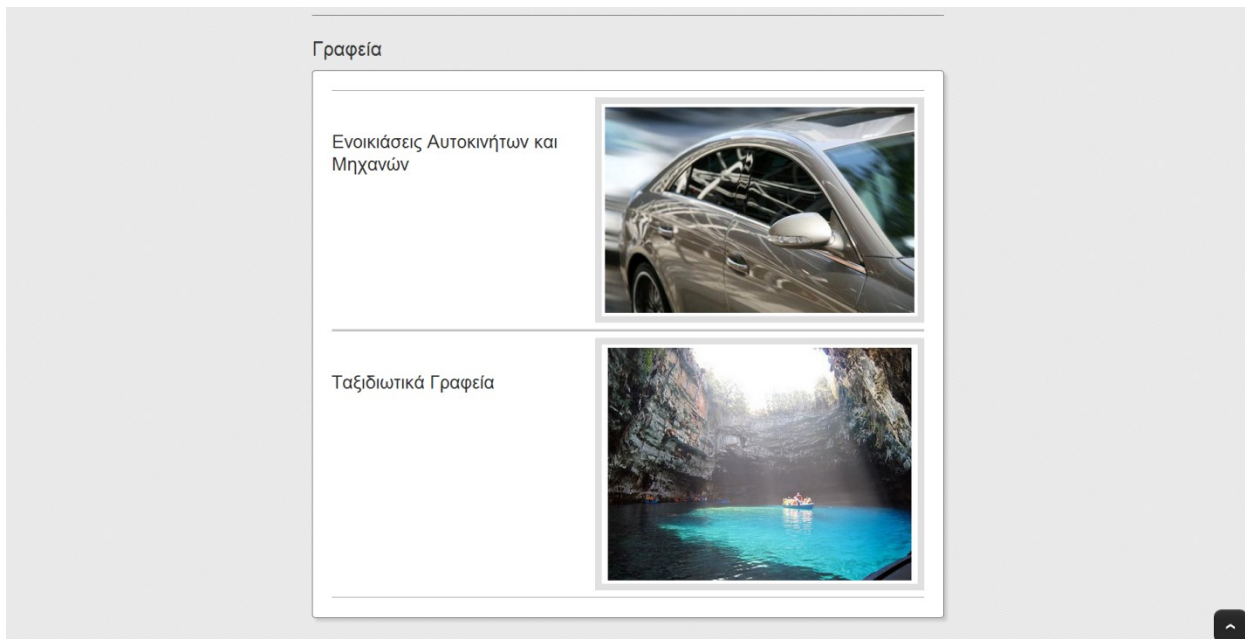
- **Χάρτης**



Εικόνα 17 - Προβολή χάρτη της Ρόδου μέσω Google Maps

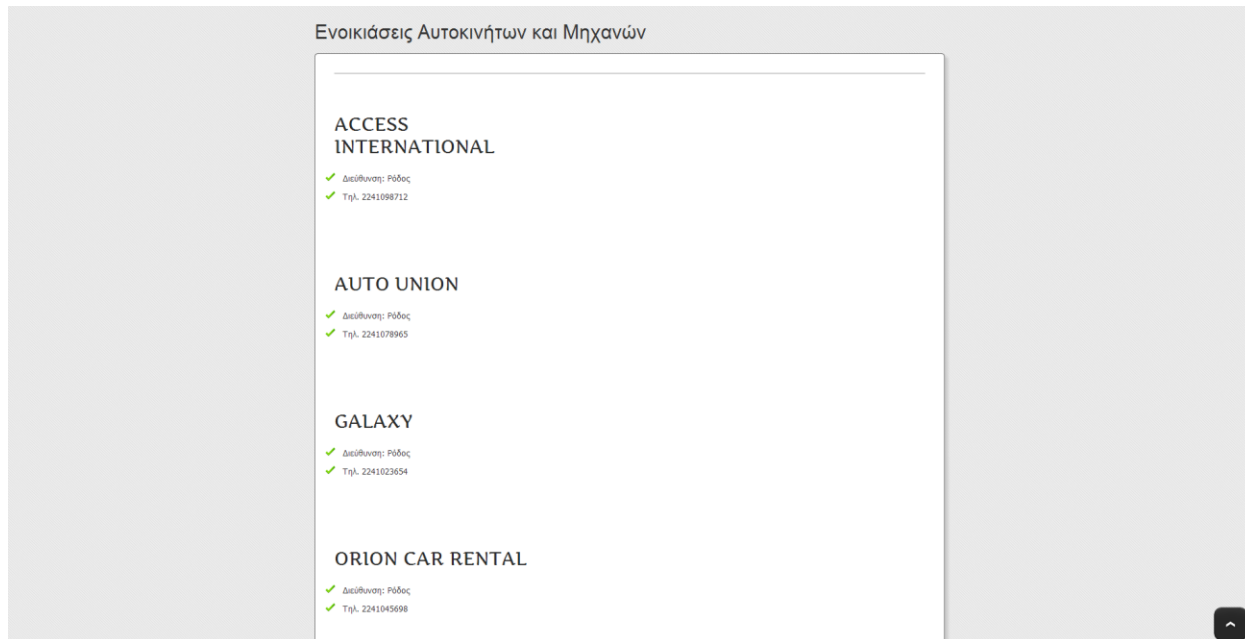
Η επιλογή του Χάρτη, οδηγεί στην εμφάνιση του χάρτη της Google Map.

- **Γραφεία**



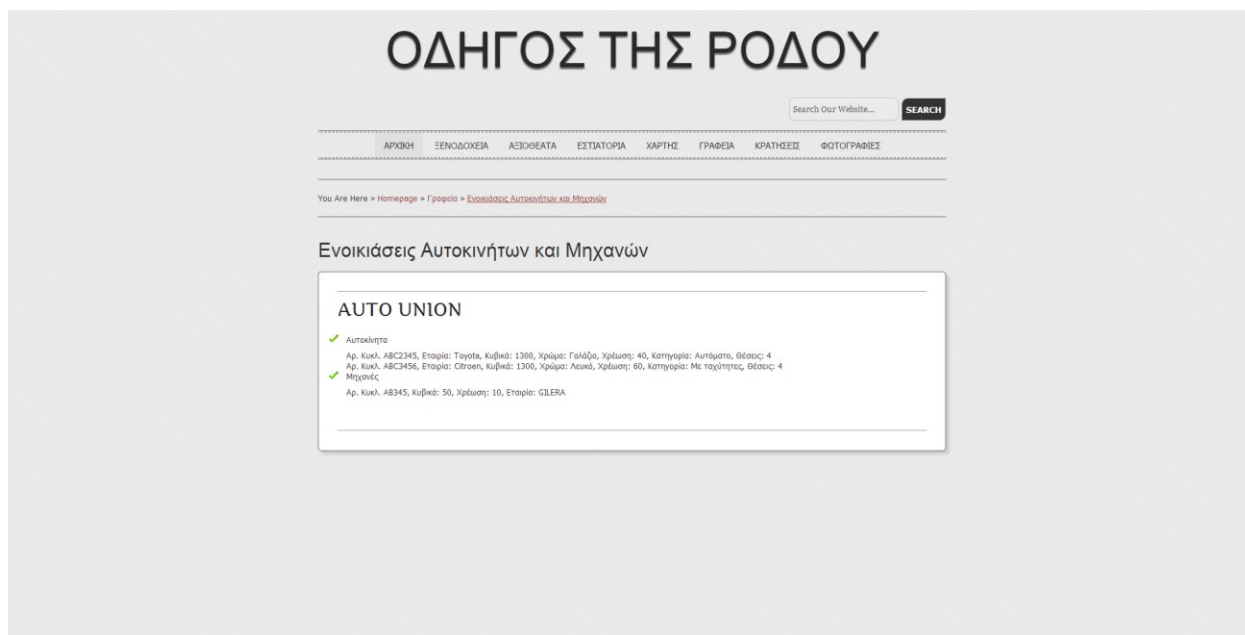
Εικόνα 18 - Προβολή Γραφείων

Επιλέγοντας ο χρήστης την έκτη επιλογή του μενού, "Γραφεία", οδηγείται ανάμεσα στις δύο επιλογές ανάμεσα στα γραφεία ενοικίασης αυτοκινήτων και μηχανών και στα ταξιδιωτικά γραφεία της Ρόδου.



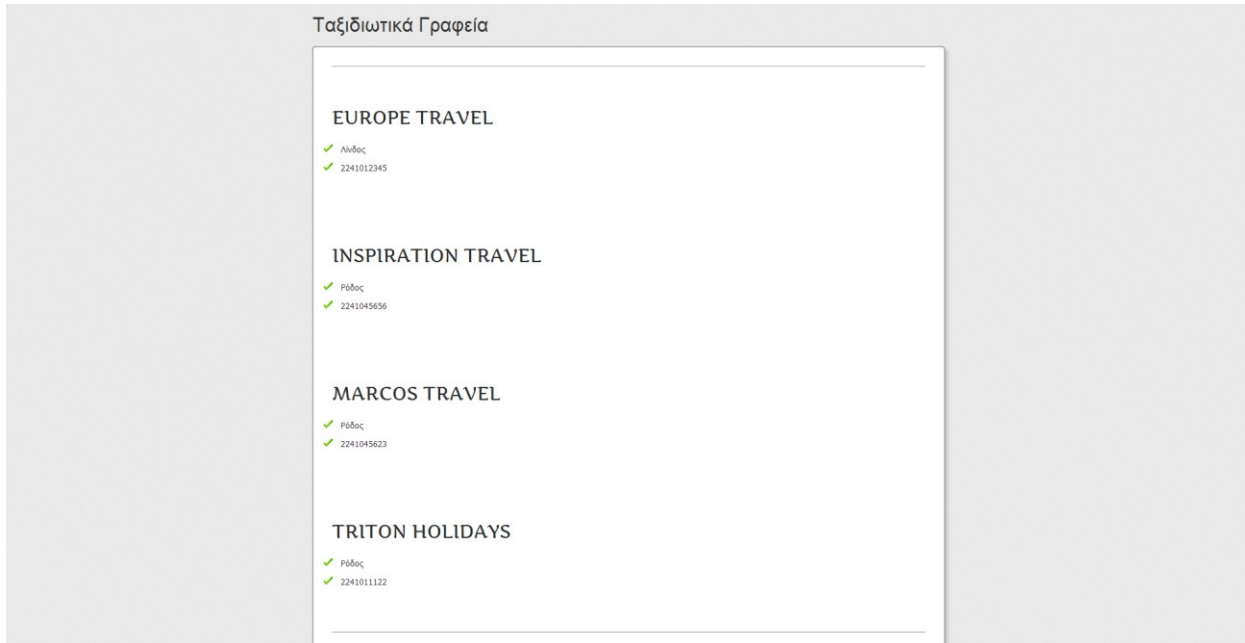
Εικόνα 19 - Προβολή γραφείων ενοικιαζόμενων αυτοκινήτων και μηχανών

Η πρώτη επιλογή "Ενοικιάσεις Αυτοκινήτων και Μηχανών" οδηγεί στην προβολή της λίστας των γραφείων ενοικιαζόμενων οχημάτων. Κάθε καταχώρηση στη λίστα αποτελείται από τον τίτλο του γραφείου, τη διεύθυνση του και ένα τηλέφωνο επικοινωνίας.



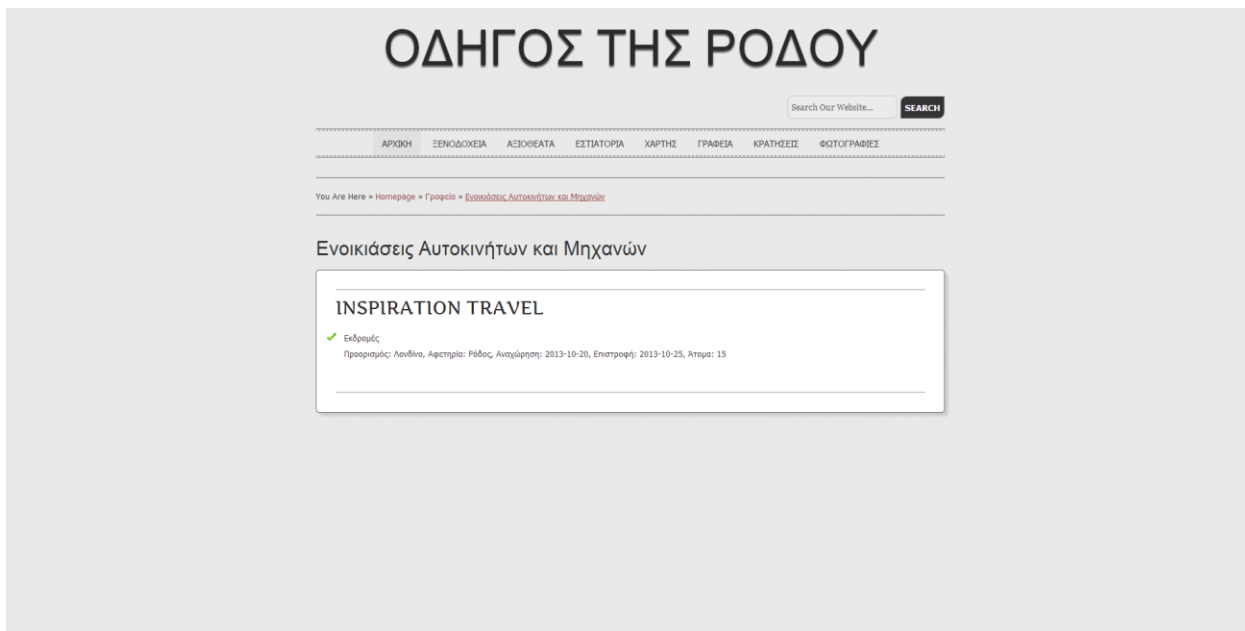
Εικόνα 20 - Προβολή διαθέσιμων οχημάτων γραφείου ενοικιαζόμενων

Η επιλογή του ονόματος του κάθε γραφείου οδηγεί σε μια σελίδα που περιέχει το όνομα του γραφείου, μια λίστα με τα αυτοκίνητα που διαθέτει καθώς επίσης και με τις μηχανές προς ενοικίαση.



Εικόνα 21 - Προβολή καταχωρημένων ταξιδιωτικών γραφείων

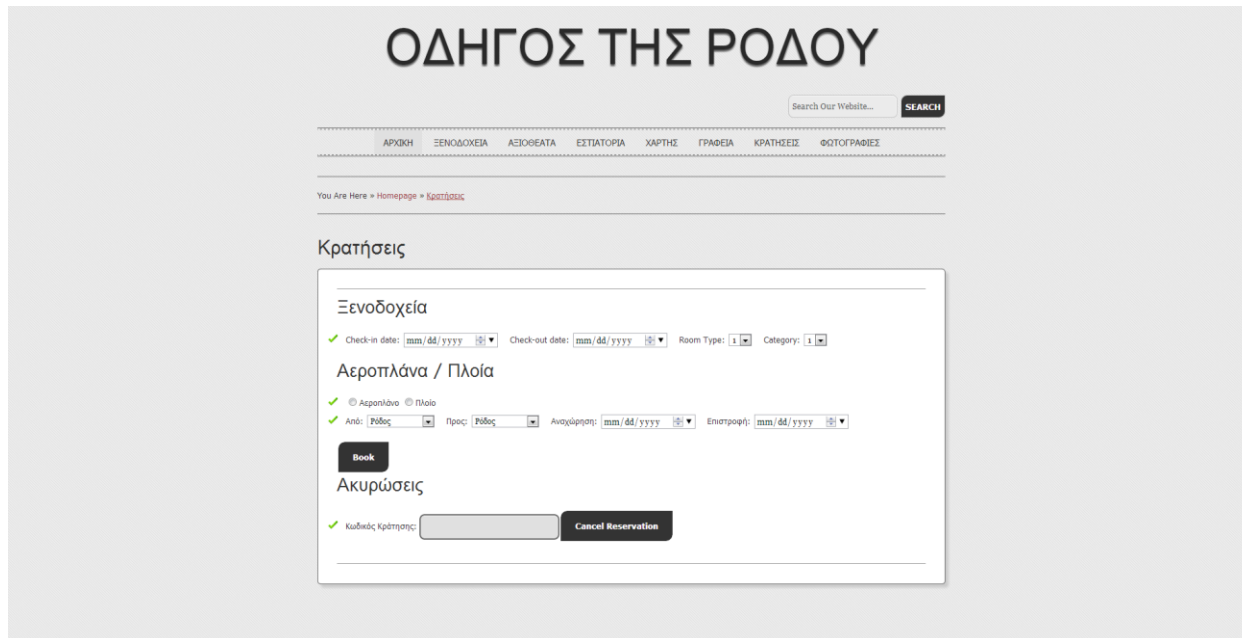
Η δεύτερη επιλογή "Ταξιδιωτικά Γραφεία" οδηγεί στην προβολή της λίστας των ταξιδιωτικών γραφείων του νησιού. Κάθε καταχώρηση στη λίστα αποτελείται από τον τίτλο του γραφείου, τη διεύθυνση του και ένα τηλέφωνο επικοινωνίας.



Εικόνα 22 - Προβολή διαθέσιμων ταξιδιωτικών πακέτων συγκεκριμένου ταξιδιωτικού γραφείου

Η επιλογή του ονόματος του κάθε γραφείου οδηγεί σε μια σελίδα που περιέχει το όνομα του γραφείου και μια λίστα με τα εκδρομικά πακέτα που διαθέτει.

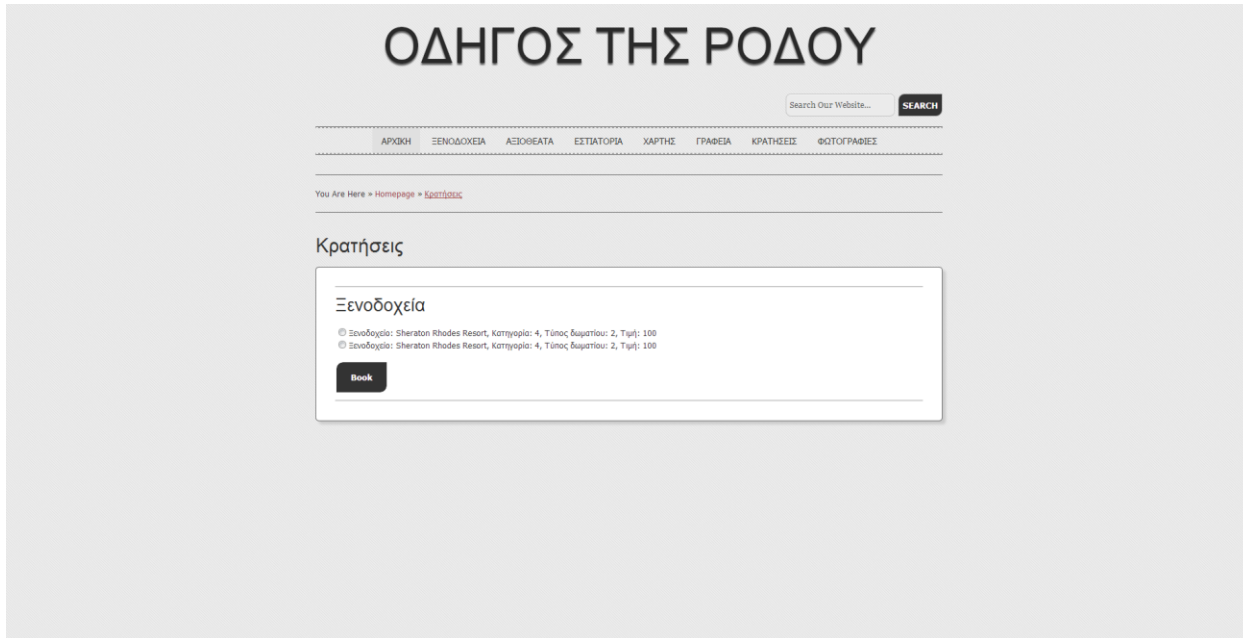
- Κρατήσεις



The screenshot shows a web page titled "ΟΔΗΓΟΣ ΤΗΣ ΡΟΔΟΥ" (Rhodes Guide). At the top, there is a search bar with the text "Search Our Website..." and a "SEARCH" button. Below the search bar is a navigation menu with links: ΑΡΧΗ, ΞΕΝΟΔΟΧΕΙΑ, ΑΞΙΟΣΕΑΤΑ, ΕΣΤΙΑΤΟΡΙΑ, ΧΑΡΤΗΣ, ΓΡΑΦΕΙΑ, ΚΡΑΤΗΣΕΙΣ, ΦΩΤΟΓΡΑΦΙΕΣ. Below the navigation menu, there is a breadcrumb trail: "You Are Here » Homepage » Κρατήσεις". The main content area is titled "Κρατήσεις" (Reservations) and contains a form for booking. The form has three sections: 1. "Ξενοδοχεία" (Hotels) with fields for "Check-in date" (mm/dd/yyyy), "Check-out date" (mm/dd/yyyy), "Room Type" (1), and "Category" (1). 2. "Αεροπλάνα / Πλοία" (Airlines / Ships) with radio buttons for "Αεροπλάνα" (selected) and "Πλοία", and fields for "Από:" (Rhodos), "Προς:" (Rhodos), "Αναχώρηση:" (mm/dd/yyyy), and "Επιστροφή:" (mm/dd/yyyy). 3. "Ακυρώσεις" (Cancellations) with a "Κωδικός κράτησης:" field and a "Cancel Reservation" button. A "Book" button is also present in the first section.

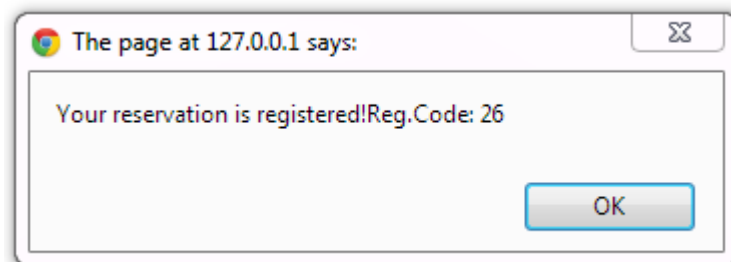
Εικόνα 23 - Φόρμα συμπλήρωσης στοιχείων για κράτηση

Η επιλογή "Κρατήσεις" ανοίγει τη σελίδα των κρατήσεων σε δωμάτιο, κρατήσεων εισιτηρίων ή την ακύρωση κρατήσεων. Πιο αναλυτικά, στο πρώτο τμήμα, ο επισκέπτης της ιστοσελίδας μπορεί να επιλέξει την ημέρα εισόδου στο ξενοδοχείο, την ημέρα εξόδου από αυτό, τον τύπο του δωματίου (μονόκλινο/ δίκλινο / τρίκλινο) και την κατηγορία του ξενοδοχείου. Στη συνέχεια, στο δεύτερο τμήμα, ο χρήστης μπορεί να επιλέξει το μεταφορικό μέσο, ανάμεσα σε αεροπλάνο και πλοίο, και πληροφορία σχετική με το ταξίδι, όπως σημείο αναχώρησης και προορισμό και ημερομηνία αναχώρησης και επιστροφής. Το κουμπί Book, έχει ως αποτέλεσμα την καταχώρηση της κράτησης στη βάση δεδομένων. Η διαδικασία της κράτησης μπορεί να μην περιλαμβάνει ένα από τα δύο παραπάνω τμήματα. Το τρίτο και τελευταίο τμήμα δίνει τη δυνατότητα στο χρήστη να ακυρώσει την κράτηση του, καταχωρώντας τον κωδικό κράτησης και πατώντας το κουμπί "Cancel Reservation".



Εικόνα 24 - Προβολή διαθέσιμων δωματίων για κράτηση

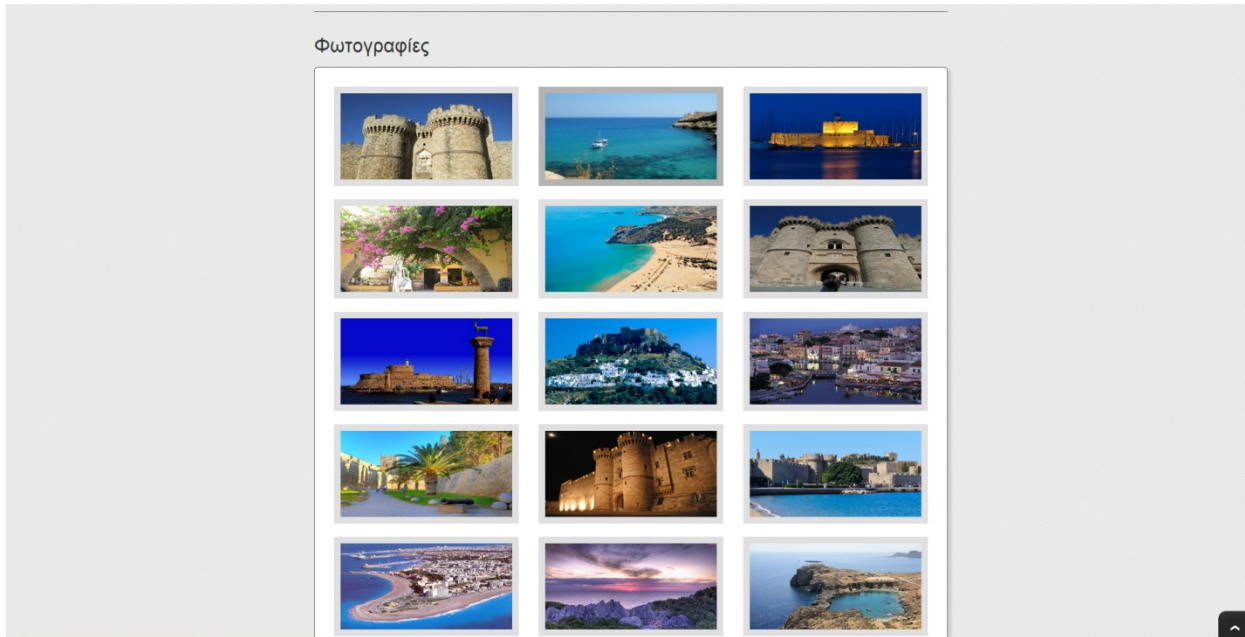
Συμπληρώνοντας τη φόρμα των κρατήσεων, ο χρήστης καθορίζει τα κριτήρια με τα οποία αναζητά δωμάτιο και το σύστημα του επιστρέφει μια λίστα με τα δωμάτια που ικανοποιούν τα κριτήρια του. Έτσι, ο χρήστης μπορεί να επιλέξει κάποιο από τα προτεινόμενα δωμάτια και στη συνέχεια να επιλέξει το κουμπί Book για την καταχώρηση της κράτησης στη βάση δεδομένων.



Εικόνα 25 - Προβολή κωδικού αριθμού κράτησης

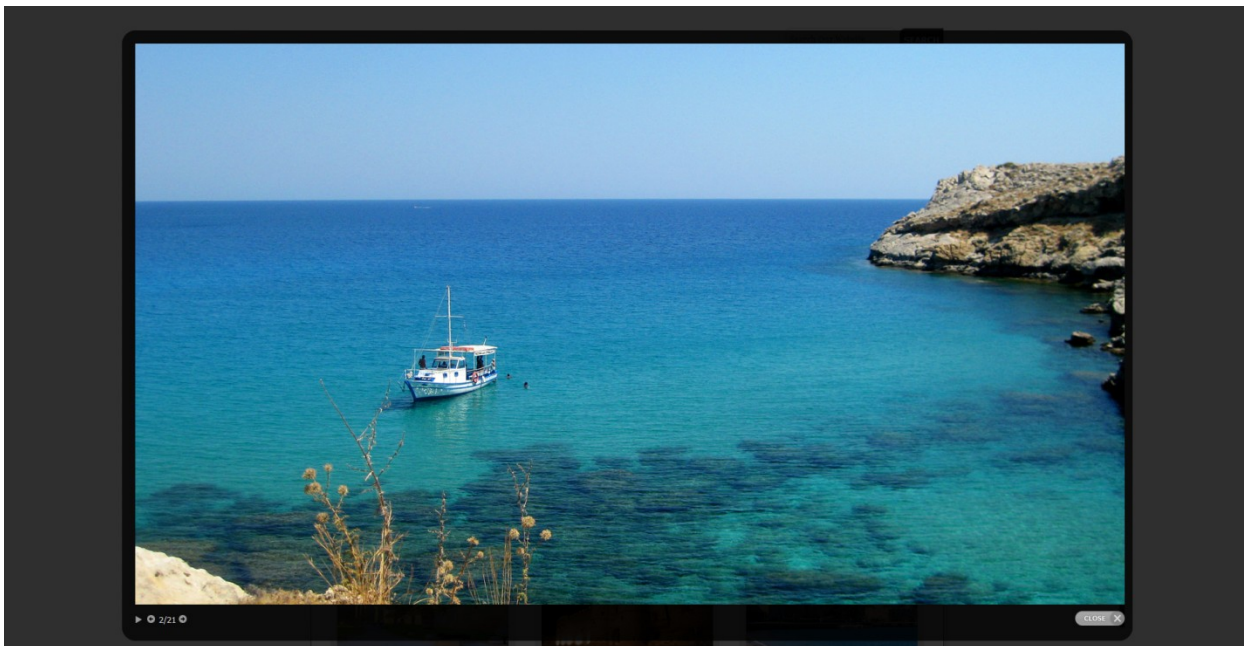
Κατά την ολοκλήρωση της διαδικασίας της κράτησης, εμφανίζεται ένα παράθυρο που ενημερώνει τον χρήστη με τον κωδικό της κράτησης του, που είναι απαραίτητη στην περίπτωση της ακύρωσης της.

- Φωτογραφίες



Εικόνα 26 - Προβολή φωτογραφιών

Η τελευταία επιλογή του μενού οδηγεί στην πλήρη λίστα των φωτογραφιών της ιστοσελίδας.



Εικόνα 27 - Προβολή φωτογραφίας

Επιλέγοντας ο χρήστης την κάθε φωτογραφία, μπορεί να δει την προβολή της σε μεγέθυνση.

4.3 Υλοποίηση και Ανάλυση Κώδικα της Εφαρμογής

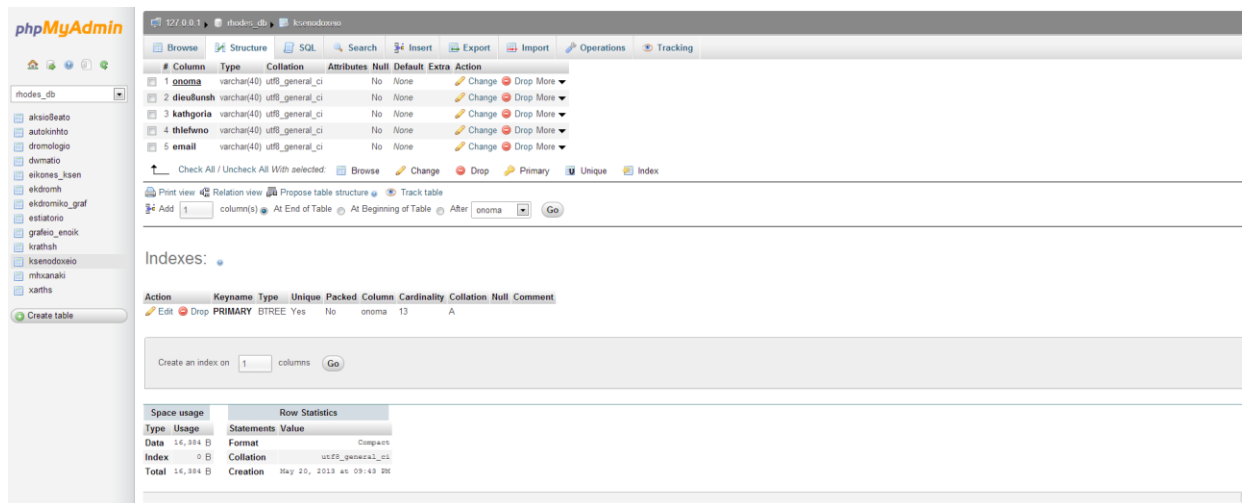
4.3.1 Δημιουργία Βάσης Δεδομένων

Στη συνέχεια, παρατίθενται οι εντολές για τη δημιουργία των παραπάνω πινάκων σε MySQL για τη δημιουργία της Βάσης Δεδομένων.

CREATE TABLE ksenodoxeio(

```

onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
diou8unsh varchar(40) COLLATE utf8_general_ci NOT NULL,
kathgoria varchar(40) COLLATE utf8_general_ci NOT NULL,
thlefwno varchar(40) COLLATE utf8_general_ci NOT NULL,
email varchar(40) COLLATE utf8_general_ci NOT NULL,
PRIMARY KEY(onoma));
    
```



Εικόνα 28 - Πίνακας ksenodoxeio

CREATE TABLE krathsh(

```

id int NOT NULL,
onoma varchar(40) COLLATE utf8_general_ci,
room_type int NOT NULL,
hmer_afikshs DATE,
hmer_anax DATE,
category int NOT NULL,
meso varchar(40) COLLATE utf8_general_ci,
apo varchar(40) COLLATE utf8_general_ci,
pros varchar(40) COLLATE utf8_general_ci,
anaxwrhsh date,
epistrofh date,
PRIMARY KEY(id),
FOREIGN KEY (onoma) REFERENCES ksenodoxeio(onoma));
    
```

Table structure for `rhodes_db.krathsh`:

#	Column	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop More
2	onoma	varchar(40)	utf8_general_ci		Yes	NULL		Change Drop More
3	room_type	int(11)			No	None		Change Drop More
4	hmer_eklishs	date			Yes	NULL		Change Drop More
5	hmer_anax	date			Yes	NULL		Change Drop More
6	category	int(11)			No	None		Change Drop More
7	meso	varchar(40)	utf8_general_ci		Yes	NULL		Change Drop More
8	apo	varchar(40)	utf8_general_ci		Yes	NULL		Change Drop More
9	pros	varchar(40)	utf8_general_ci		Yes	NULL		Change Drop More
10	anaxwarhah	date			Yes	NULL		Change Drop More
11	epietrofh	date			Yes	NULL		Change Drop More

Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	id	23	A		
Edit Drop	onoma	BTREE	No	No	onoma	4	A	YES	

Εικόνα 29 - Πίνακας krathsh

**CREATE TABLE dwmatio(
onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
id int NOT NULL,
eidoss int NOT NULL,
timh float,
PRIMARY KEY(onoma, id),
FOREIGN KEY (onoma) REFERENCES ksenodoxeio(onoma));**

Table structure for `rhodes_db.dwmatio`:

#	Column	Type	Collation	Attributes	Null	Default	Extra	Action
1	onoma	varchar(40)	utf8_general_ci		No	None		Change Drop More
2	id	int(11)			No	None		Change Drop More
3	eidoss	int(11)			No	None		Change Drop More
4	timh	float			Yes	NULL		Change Drop More

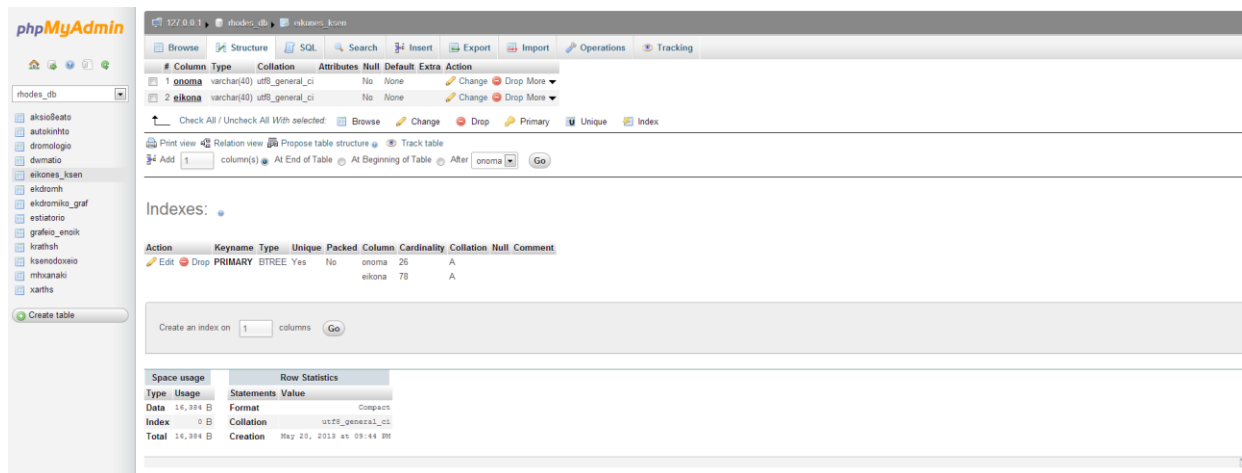
Indexes:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	PRIMARY	BTREE	Yes	No	onoma	17	A		
					id	35	A		

Εικόνα 30 - Πίνακας dwmatio

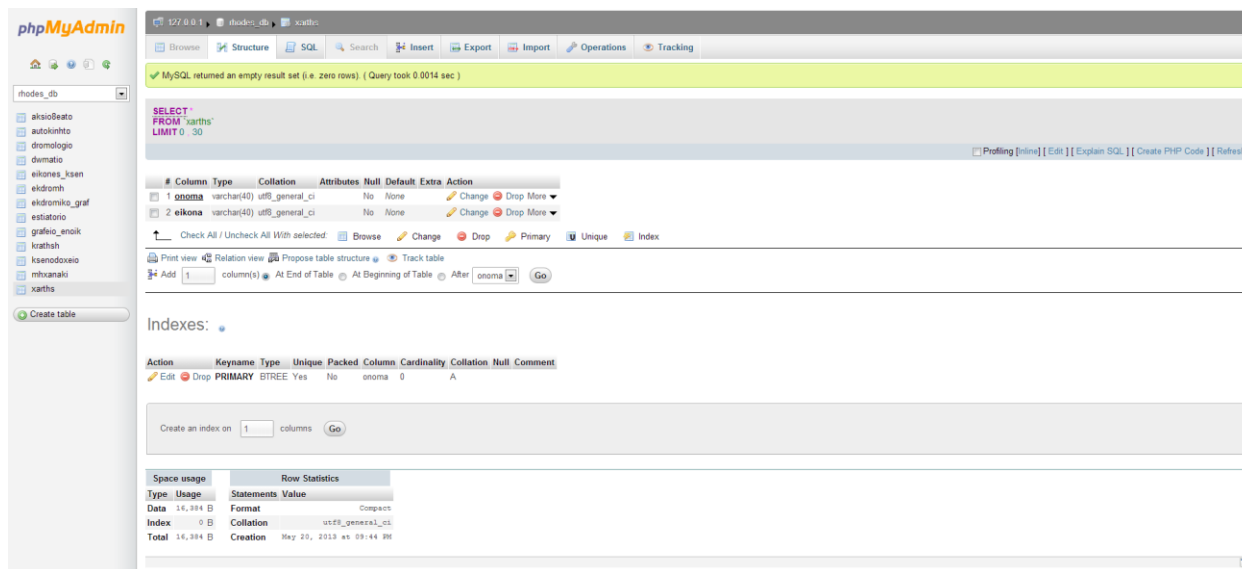
**CREATE TABLE eikones_ksen(
onoma varchar(40) COLLATE utf8_general_ci NOT NULL,**

**eikona varchar(40) COLLATE utf8_general_ci NOT NULL,
PRIMARY KEY(onoma, eikona),
FOREIGN KEY (onoma) REFERENCES ksenodoxeio(onoma));**



Εικόνα 31 - Πίνακας eikones_ksen

**CREATE TABLE xarths(
onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
eikona varchar(40) COLLATE utf8_general_ci NOT NULL,
PRIMARY KEY(onoma)
);**

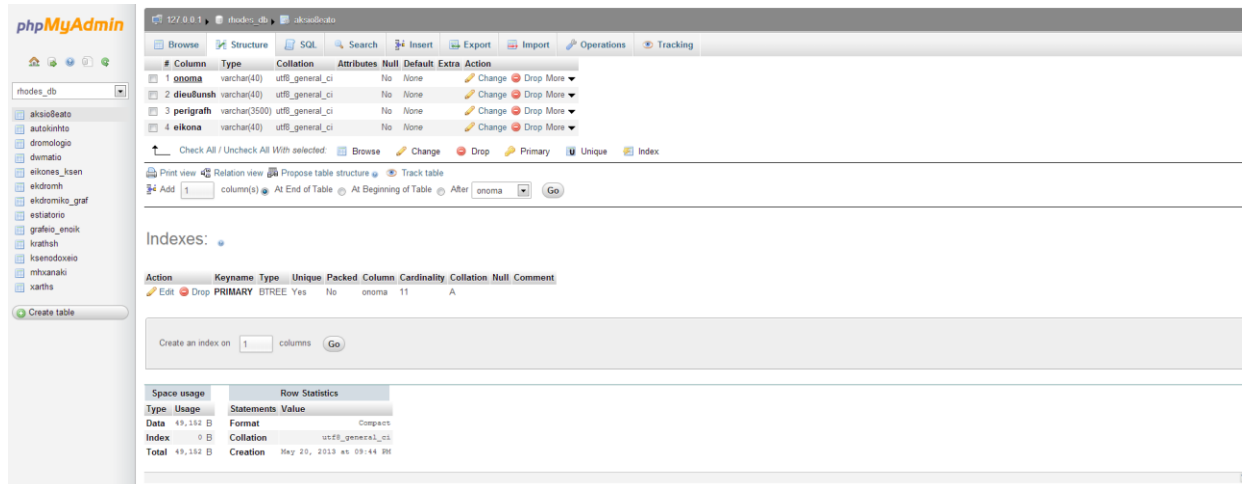


Εικόνα 32 - Πίνακας xarths

**CREATE TABLE aksio8eato(
onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
dieu8unsh varchar(40) COLLATE utf8_general_ci NOT NULL,**

```
perigraph varchar(500) COLLATE utf8_general_ci NOT NULL,  
eikona varchar(40) COLLATE utf8_general_ci NOT NULL,  
PRIMARY KEY(ονομα)
```

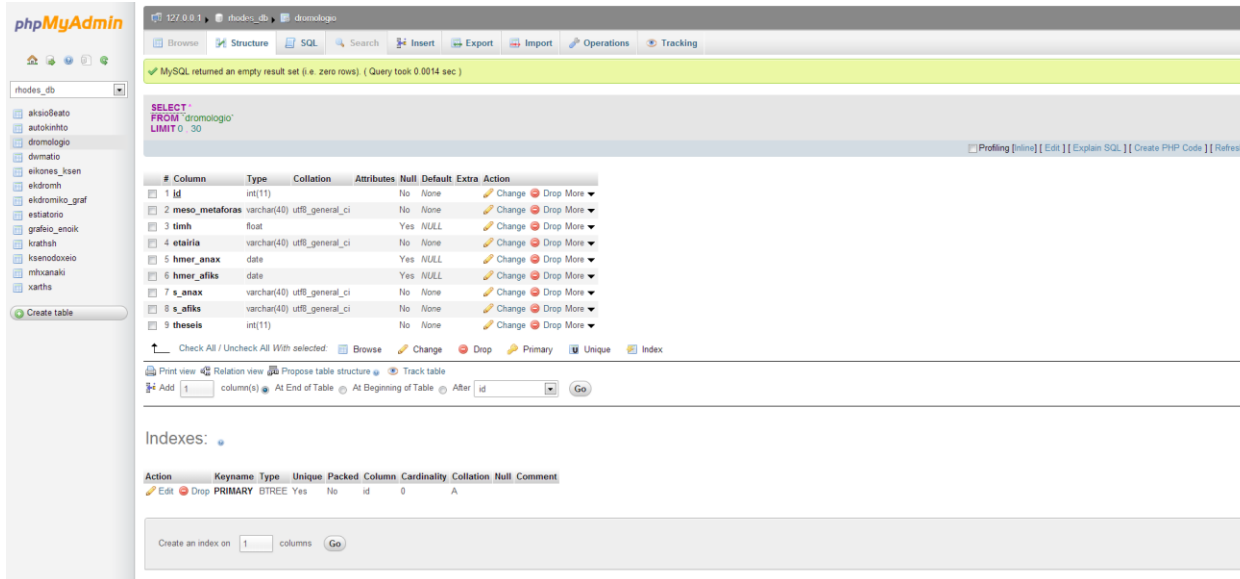
```
);
```



Εικόνα 33 - Πίνακας aksioeato

```
CREATE TABLE dromologio(  
id int NOT NULL,  
meso_metaforas varchar(40) COLLATE utf8_general_ci NOT NULL,  
timh float,  
etairia varchar(40) COLLATE utf8_general_ci NOT NULL,  
hmer_anax date,  
hmer_afiks date,  
s_anax varchar(40) COLLATE utf8_general_ci NOT NULL,  
s_afiks varchar(40) COLLATE utf8_general_ci NOT NULL,  
theseis int NOT NULL,  
PRIMARY KEY(id)
```

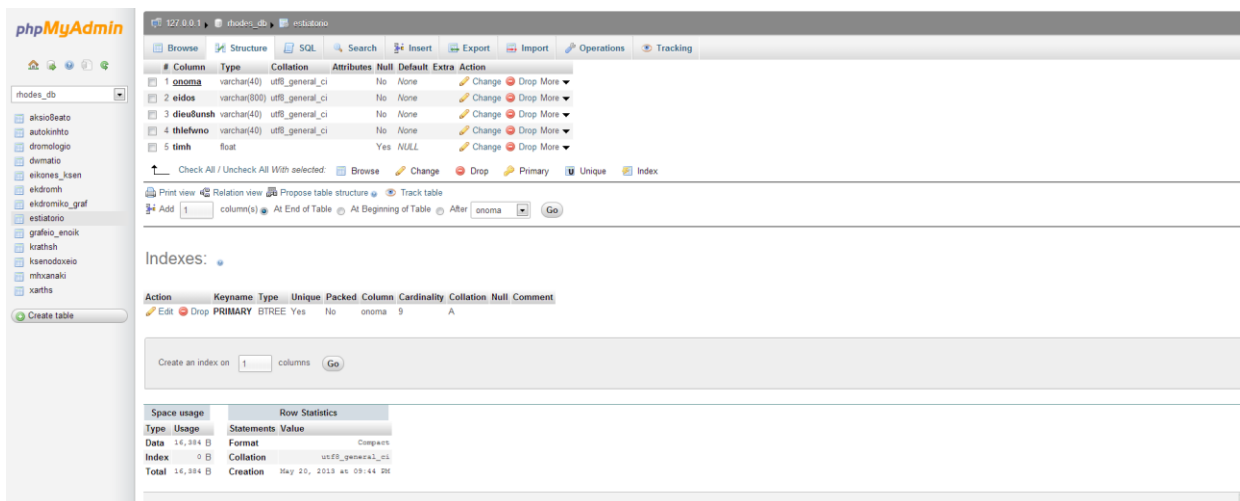
```
);
```

Εικόνα 34 - Πίνακας dromologio

```

CREATE TABLE estiatorio(
    onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
    eidos varchar(40) COLLATE utf8_general_ci NOT NULL,
    dieu8unsh varchar(40) COLLATE utf8_general_ci NOT NULL,
    thlefwno varchar(40) COLLATE utf8_general_ci NOT NULL,
    timh float,
    PRIMARY KEY(onoma)
);
    
```



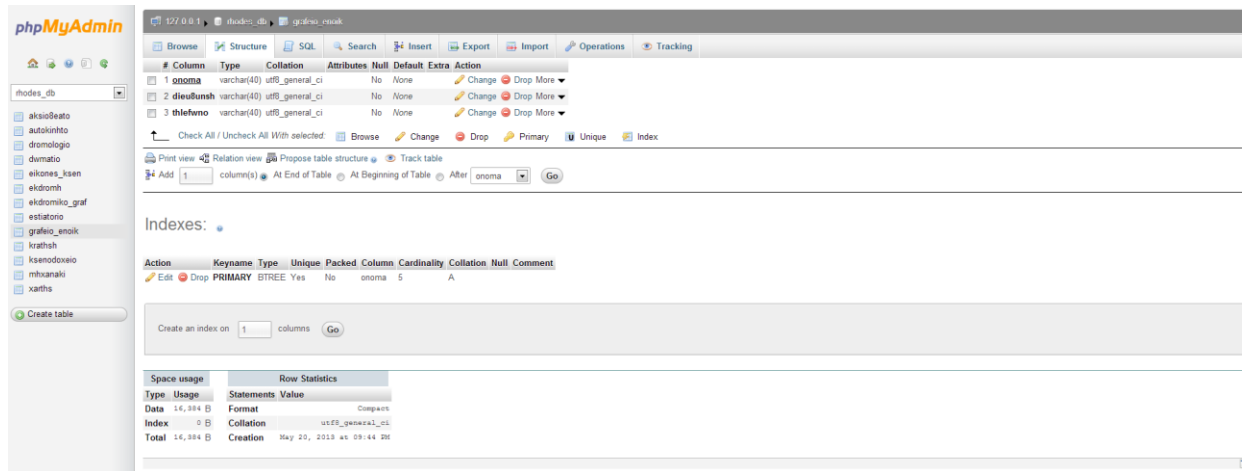
Εικόνα 35 - Πίνακας estiatorio

```

CREATE TABLE grafeio_enoik(
    onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
    dieu8unsh varchar(40) COLLATE utf8_general_ci NOT NULL,
    );
    
```

**thlefwno varchar(40) COLLATE utf8_general_ci NOT NULL,
PRIMARY KEY(onoma)**

);

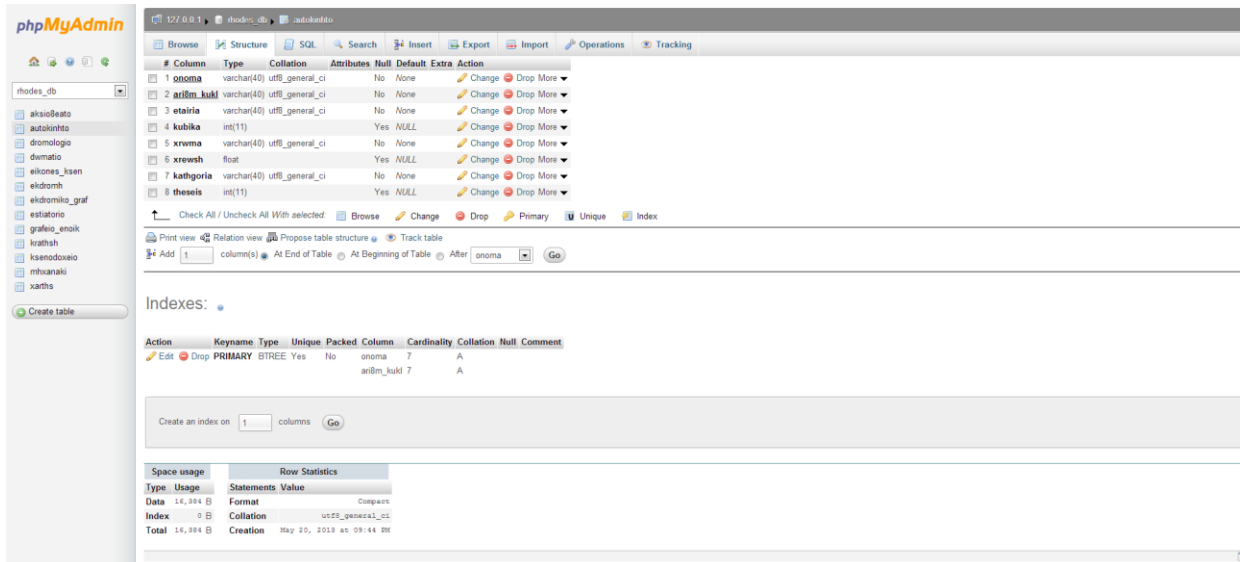


Εικόνα 36 - Πίνακας grafeio_enoik

CREATE TABLE autokinhto(

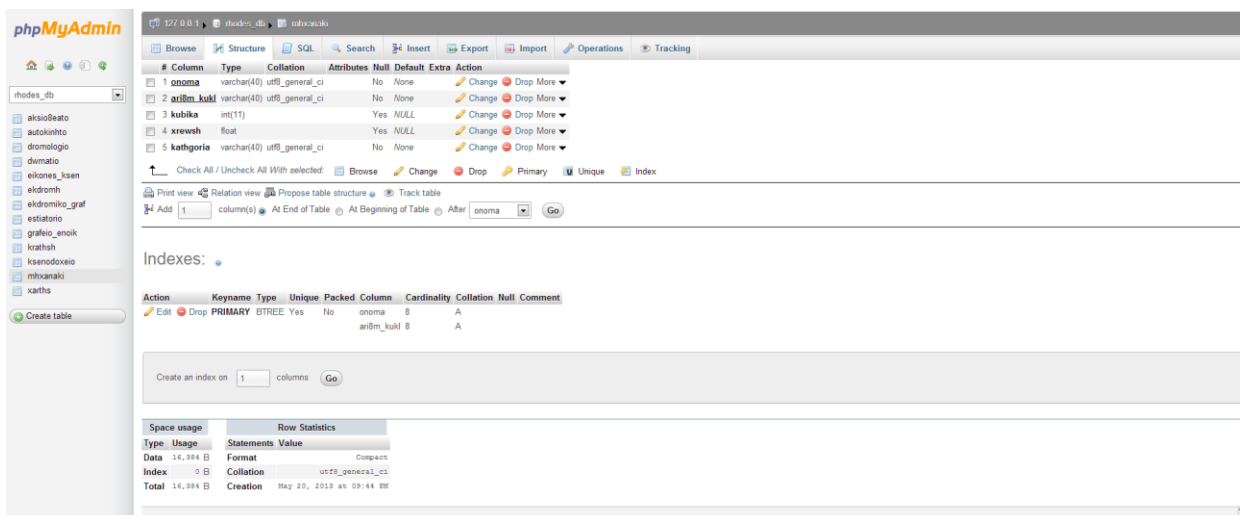
**onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
ari8m_kukl varchar(40) COLLATE utf8_general_ci NOT NULL,
etairia varchar(40) COLLATE utf8_general_ci NOT NULL,
kubika int,
xrwma varchar(40) COLLATE utf8_general_ci NOT NULL,
xrewsh float,
kathgoria varchar(40) COLLATE utf8_general_ci NOT NULL,
theseis int,
PRIMARY KEY(onoma, ari8m_kukl),
FOREIGN KEY (onoma) REFERENCES grafeio_enoik(onoma)**

);



Εικόνα 37 - Πίνακας autokinhto

```
CREATE TABLE mhxanaki(
    onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
    ari8m_kukl varchar(40) COLLATE utf8_general_ci NOT NULL,
    kubika int,
    xrewsh float,
    kathgoria varchar(40) COLLATE utf8_general_ci NOT NULL,
    PRIMARY KEY(onoma, ari8m_kukl),
    FOREIGN KEY (onoma) REFERENCES grafeio_enoik(onoma)
);
```

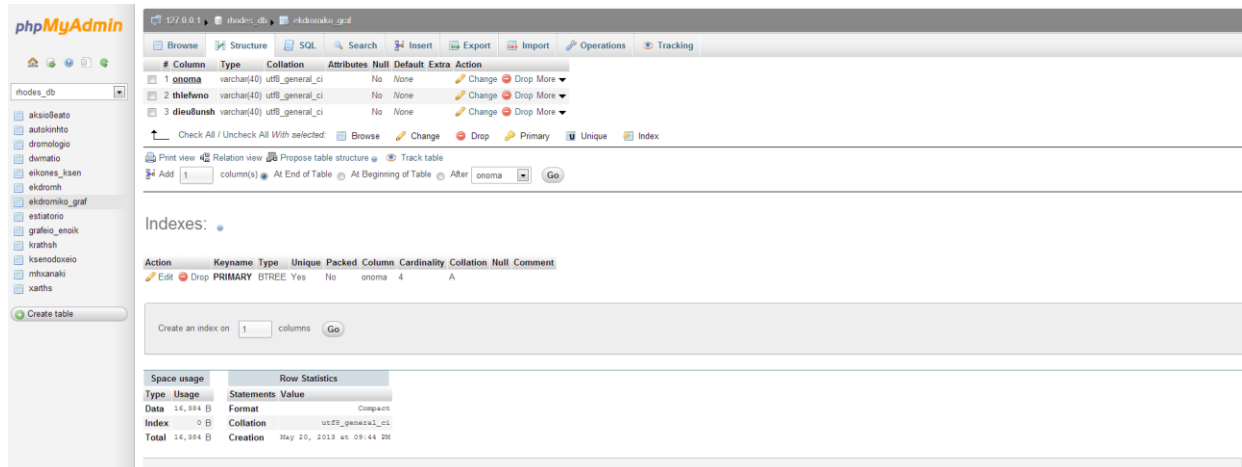


Εικόνα 38 - Πίνακας mhxanaki

```
CREATE TABLE ekdromiko_graf(
    onoma varchar(40) COLLATE utf8_general_ci NOT NULL,
```

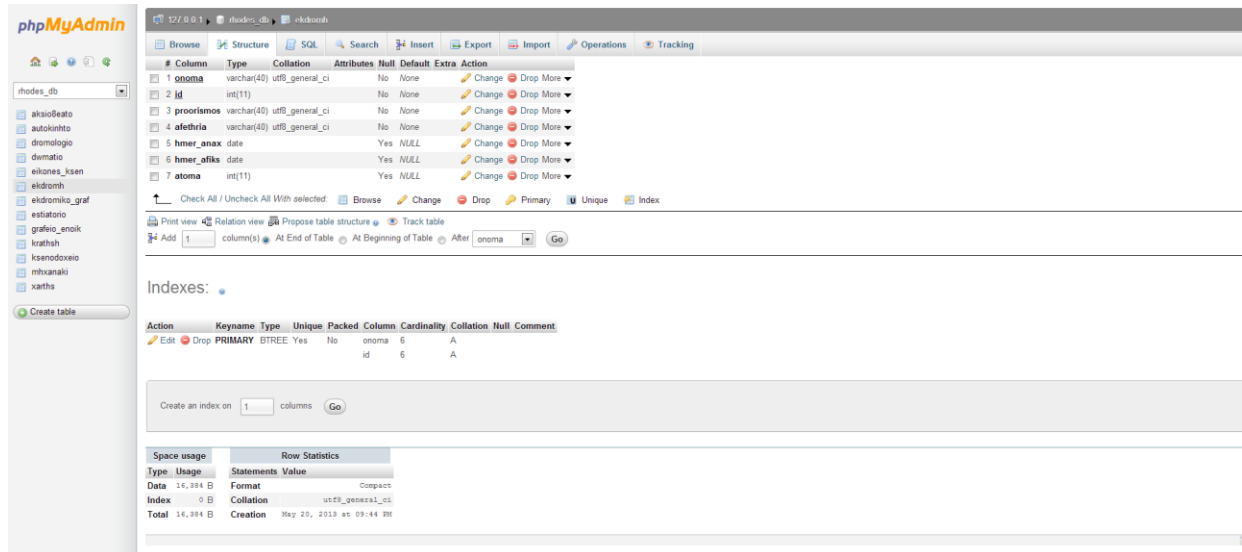
```
thlefwno varchar(40) COLLATE utf8_general_ci NOT NULL,  
dieu8unsh varchar(40) COLLATE utf8_general_ci NOT NULL,  
PRIMARY KEY(onoma)
```

```
);
```



Εικόνα 39 - Πίνακας `ekdromiko_graf`

```
CREATE TABLE ekdromh(  
  onoma varchar(40) COLLATE utf8_general_ci NOT NULL,  
  id int NOT NULL,  
  proorismos varchar(40) COLLATE utf8_general_ci NOT NULL,  
  afethria varchar(40) COLLATE utf8_general_ci NOT NULL,  
  hmer_anax date,  
  hmer_afiks date,  
  atoma int,  
  PRIMARY KEY(onoma, id),  
  FOREIGN KEY (onoma) REFERENCES ekdromiko_graf(onoma)  
);
```



Εικόνα 40 - Πίνακας ekdromh

4.3.2 SQL Επερωτήσεις στη Βάση Δεδομένων και Ανάλυση Κώδικα PHP

Παρακάτω περιγράφονται τα τμήματα κώδικα που εκτελούν τις επερωτήσεις στη βάση δεδομένων καθώς και την υλοποίηση του στην PHP.

Ξενοδοχεία

```
<?php
require "config.php";

$db_data = new db_data();
MYSQL_QUERY("SET NAMES 'utf8'");
$query = MYSQL_QUERY("SELECT * FROM ksenodoxeio");
?>
```

Αρχικά, δημιουργείται μια σύνδεση με τη βάση και εκτελείται η επερώτηση SET NAMES 'utf8' ώστε τα δεδομένα που θα επιστραφούν από την επερώτηση να είναι σε κωδικοποίηση utf8. Στη συνέχεια εκτελείται η επερώτηση SELECT * FROM ksenodoxeio που επιστρέφει όλα τα στοιχεία του πίνακα ksenodoxeio. Τα αποτελέσματα της επερώτησης αυτής εκχωρούνται στη μεταβλητή \$query.

```
<?php
$i = 0;
while($row = mysql_fetch_array($query)){
    $i++;
    ?>
    <div class="portfolio">
    <div class="tagline_left">
        <h2><a href="hotelpic.php?name=<?php echo $row[0];?>">
            <?php echo $row[0];?></a>
        </h2>
        <ul id="list1">
        <li>Διεύθυνση: <?php echo $row[1];?></li>
```

```
<li>Τηλ. <?php echo $row[3];?></li>
    <li>email: <?php echo $row[4];?></li>
    <li>Κατηγορία: <?php echo $row[2];?></li>
</ul>
</div>
    <div class="tagline_right"><a href="images/hotels/<?php echo $i;?>.jpg" data-
gal="prettyPhoto" title="Title"></a></div>
    <div class="clear"></div>
</div>
<?php }?>
```

Με τη βοήθεια της `while` η μεταβλητή `$row` περιέχει τα αποτελέσματα της μεταβλητής `$query` ανά γραμμή και έτσι αναπαριστώνται τα ξενοδοχεία που είναι καταχωρημένα στη βάση δεδομένων καθώς και η σχετική με αυτά πληροφορία. Στην περίπτωση που ο χρήστης του συστήματος επιλέξει κάποιο ξενοδοχείο εκτελείται ο ακόλουθος κώδικας με παράμετρο το όνομα του ξενοδοχείου.

```
$db_data = new db_data();
$name = $_GET['name'];
$query = MYSQL_QUERY("SELECT * FROM ksenodoxeio WHERE onoma='$name'");
$row = mysql_fetch_array($query);
```

Δημιουργείται μια σύνδεση με τη βάση και στη συνέχεια ορίζεται ως `$name` η παράμετρος από το προηγούμενο `.php` αρχείο, δηλαδή το όνομα του ξενοδοχείου που επέλεξε ο χρήστης. Εκτελείται η επερώτηση `SELECT * FROM ksenodoxeio WHERE onoma='$name'` που επιστρέφει όλη την σχετική πληροφορία από τον πίνακα `ksenodoxeio` με το συγκεκριμένο όνομα. Τέλος, η μεταβλητή `$row` περιέχει όλη την σχετική με αυτό το ξενοδοχείο πληροφορία και την αναπαριστά λεπτομερώς.

Αξιοθέατα

Επιλέγοντας ο χρήστης του συστήματος την επιλογή Αξιοθέατα από το κεντρικό μενού, οδηγείται σε μια λίστα με τα αξιοθέατα της Ρόδου. Από εκεί μπορεί να επιλέξει ένα αξιοθέατο και να εκτελεστεί το παρακάτω τμήμα κώδικα.

```
$db_data = new db_data();
$name = $_GET['name'];

MYSQL_QUERY("SET NAMES 'utf8'");
$query = MYSQL_QUERY("SELECT * FROM aksioθεato WHERE eikona='$name'");

$row = mysql_fetch_array($query);
```

Στο αρχείο αυτό ανοίγει μια σύνδεση με τη βάση δεδομένων, αποθηκεύεται στη μεταβλητή `$name` το όνομα της εικόνας του αξιοθέατου που επέλεξε ο χρήστης και εκτελείται η επερώτηση που καθορίζει τη μορφή των αποτελεσμάτων σε κωδικοποίηση `utf8`. Στη συνέχεια, εκτελείται η επερώτηση που επιστρέφει από τον πίνακα `aksioθεato` τα δεδομένα που αντιστοιχούν στο αξιοθέατο που το όνομα της εικόνας είναι ίσο με την τιμή της μεταβλητής `$name`.

Εστιατόρια

```
$db_data = new db_data();

MYSQL_QUERY("SET NAMES 'utf8'");
```

```
$query = MYSQL_QUERY("SELECT * FROM estiatorio");
```

Αρχικά, δημιουργείται μια σύνδεση με τη βάση και εκτελείται η επερώτηση SET NAMES 'utf8' ώστε τα δεδομένα που θα επιστραφούν από την επερώτηση να είναι σε κωδικοποίηση utf8. Στη συνέχεια εκτελείται η επερώτηση SELECT * FROM estiatorio που επιστρέφει όλα τα στοιχεία του πίνακα estiatorio. Τα αποτελέσματα της επερώτησης αυτής εκχωρούνται στη μεταβλητή \$query.

```
while($row = mysql_fetch_array($query)){
```

Με τη χρήση της επαναληπτικής δομής while και της μεταβλητής \$row προβάλλονται τα ονόματα των εστιατορίων της Ρόδου που είναι καταχωρημένα στη βάση καθώς και η σχετική πληροφορία (περιγραφή, τηλέφωνο επικοινωνίας κλπ.).

Γραφεία

Η επιλογή Γραφεία του κεντρικού μενού οδηγεί σε μια σελίδα με δύο συνδέσμους, για γραφεία ενοικιάσεως οχημάτων και για ταξιδιωτικά γραφεία.

```
$db_data = new db_data();
```

```
MYSQL_QUERY("SET NAMES 'utf8'");
```

```
$query = MYSQL_QUERY("SELECT * FROM grafeio_enoik");
```

Στην περίπτωση που ο χρήστης του συστήματος επιλέξει το σύνδεσμο με τα γραφεία ενοικιάσεως, εκτελείται η επερώτηση που επιστρέφει όλα τα δεδομένα του πίνακα grafeio_enoik. Έτσι, προβάλλεται μια λίστα με τα διαθέσιμα γραφεία ενοικίασης αυτοκινήτων, εκ των οποίων ο χρήστης μπορεί να επιλέξει κάποιο και να εκτελεστεί ο ακόλουθος κώδικας.

```
$db_data = new db_data();
```

```
MYSQL_QUERY("SET NAMES 'utf8'");
```

```
$name = $_GET['name'];
```

```
$query = MYSQL_QUERY("SELECT * FROM grafeio_enoik WHERE onoma = '$name'");
```

```
$row = mysql_fetch_array($query);
```

Στο παραπάνω τμήμα εντολών εκτελείται μια επερώτηση που επιστρέφει στη μεταβλητή \$query τη σχετική πληροφορία του γραφείου ενοικιάσεως με το όνομα που δίνεται ως παράμετρος.

```
</li>Αυτοκίνητα</li>
```

```
<?php
```

```
    $query2 = MYSQL_QUERY("SELECT * FROM autokinhto WHERE onoma = '$name'");
```

```
    while($row2 = mysql_fetch_array($query2)){
```

```
?>
```

```
<p><?php echo "Αρ. Κυκλ. ", $row2[1], ", Εταιρία: ", $row2[2], ", Κυβικά: ", $row2[3], ", Χρώμα: ", $row2[4], ", Χρέωση: ", $row2[5], ", Κατηγορία: ", $row2[6], ", Θέσεις: ", $row2[7];?></p>
```

```
<?php
```

```
    }
```

```
?>
```

Η παραπάνω επερώτηση επιστρέφει όλη τη σχετική πληροφορία με τα διαθέσιμα αυτοκίνητα του συγκεκριμένου γραφείου ενοικιαζόμενων αυτοκινήτων.

```
<li>Μηχανές</li>
<?php
    $query3 = MYSQL_QUERY("SELECT * FROM mhchanaki WHERE onoma = '$name'");
    while($row3 = mysql_fetch_array($query3)){
?>
<p><?php echo "Αρ. Κυκλ. ", $row3[1], ", Κυβικά: ", $row3[2], ", Χρέωση: ", $row3[3], ", Εταιρία: ",
$row3[4];?></p>
<?php
    }
?>
```

Η παραπάνω επερώτηση επιστρέφει όλη τη σχετική πληροφορία με τις διαθέσιμες μηχανές του συγκεκριμένου γραφείου ενοικιαζόμενων αυτοκινήτων.

```
$db_data = new db_data();

MYSQL_QUERY("SET NAMES 'utf8'");
$query = MYSQL_QUERY("SELECT * FROM ekdromiko_graf");
```

Στην περίπτωση που ο χρήστης του συστήματος επιλέξει το σύνδεσμο με τα ταξιδιωτικά γραφεία ενοικίασεως, εκτελείται η επερώτηση που επιστρέφει όλα τα δεδομένα του πίνακα ekdromiko_graf. Έτσι, προβάλλεται μια λίστα με τα διαθέσιμα ταξιδιωτικά γραφεία, εκ των οποίων ο χρήστης μπορεί να επιλέξει κάποιο και να εκτελεστεί ο ακόλουθος κώδικας.

```
<li>Εκδρομές</li>
<?php
    $query2 = MYSQL_QUERY("SELECT * FROM ekdromh WHERE onoma = '$name'");
    while($row2 = mysql_fetch_array($query2)){
?>
<p><?php echo "Προορισμός: ", $row2[2], ", Αφετηρία: ", $row2[3], ", Αναχώρηση: ", $row2[4], ",
Επιστροφή: ", $row2[5], ", Άτομα: ", $row2[6];?></p>
<?php
    }
?>
```

Η παραπάνω επερώτηση επιστρέφει όλη τη σχετική πληροφορία με τις διαθέσιμες εκδρομές του συγκεκριμένου ταξιδιωτικού γραφείου.

Κρατήσεις

Η επιλογή των Κρατήσεων οδηγεί σε μια φόρμα όπου ο χρήστης μπορεί να συμπληρώσει τα στοιχεία κράτησης σε ξενοδοχείο ή τα στοιχεία κράτησης εισιτηρίου ή και τα δύο. Η καταχώρηση της φόρμας οδηγεί στην εκτέλεση του κώδικα στο αρχείο booking.php.

```
$db_data = new db_data();

$check_in = $_POST['check_in'];
$check_out = $_POST['check_out'];
$room_type = $_POST['room_type'];
$category = $_POST['category'];

$apo = $_POST['apo'];
$pros = $_POST['pros'];
$anax = $_POST['anax'];
$epistr = $_POST['epistr'];

mysql_query("INSERT INTO krathsh (room_type, hmer_afikshs, hmer_anax, category, apo, pros,
anaxwrhsh, epistrofh) VALUES('$room_type', '$check_in', '$check_out', '$category', '$apo', '$pros',
'$anax', '$epistr')");
```

Με την παραπάνω επερώτηση καταχωρείται στον πίνακα krathsh η πληροφορία που συμπλήρωσε ο χρήστης του συστήματος.

```
$q = mysql_query("SELECT DISTINCT * FROM ksenodoxeio k, dwmatio d WHERE kathgoria='$category'
AND eidos='$room_type' AND k.onoma = d.onoma");
```

Με αυτή την επερώτηση το σύστημα επιστρέφει στον χρήστη τις δυνατές επιλογές του μεταξύ δωματίων ξενοδοχείου με κριτήριο την κατηγορία, το όνομα και το είδος.

```
$db_data = new db_data();

$value = $_POST['reserve'];
$hotel = strtok($value, "/");
$id = strtok("/");

$sql = "SELECT MAX(id) FROM krathsh";
$q1 = mysql_query($sql);
$r1 = mysql_fetch_array($q1);

$sql2 = "UPDATE krathsh SET onoma = '$hotel' WHERE id = '$r1[0]'";
$q2 = mysql_query($sql2);
```

Το παραπάνω τμήμα κώδικα επιστρέφει τον κωδικό της τελευταίας κράτησης ώστε ο χρήστης να μπορεί να τον χρησιμοποιήσει στην περίπτωση ακύρωσης της.

```
$db_data = new db_data();
$name = $_POST['reservation_code'];

$sql = "DELETE FROM krathsh WHERE id = '$name'";
mysql_query($sql);
```

Το αρχείο delete_reservation.php δίνει τη δυνατότητα στο χρήστη να συμπληρώσει τον κωδικό κράτησης και στη συνέχεια να διαγράψει από τον πίνακα krathsh την εγγραφή με τον συγκεκριμένο κωδικό.

Κεφάλαιο 5. Αποτελέσματα

5.1 Συμπεράσματα

Ο τουριστικός οδηγός της Ρόδου που υλοποιήθηκε και περιγράφηκε παραπάνω αποτελεί μια ολοκληρωμένη πρόταση στον επισκέπτη του νησιού. Αποτελεί ένα σημαντικό βοηθητικό εργαλείο αφού μπορεί να ενημερώσει τον χρήστη του για τα αξιοθέατα και τα εστιατόρια καθώς επίσης παρέχει διάφορες υπηρεσίες κράτησης εισιτηρίων, οχημάτων και δωματίων. Με αυτόν τον τρόπο, ο χρήστης έχει στη διάθεση του ένα ολοκληρωμένο εργαλείο μετακίνησης στο νησί της Ρόδου και σε σύντομο χρονικό διάστημα μπορεί να καλύψει κάθε πιθανή ανάγκη που μπορεί να δημιουργηθεί στον επισκέπτη του νησιού.

5.2 Μελλοντική Εργασία και Επεκτάσεις

Μακροπρόθεσμα θα μπορούσε να προστεθούν περισσότερα κριτήρια στη διαδικασία αναζήτησης δωματίων. Έτσι, ο χρήστης του συστήματος θα μπορούσε να επιλέξει ξενοδοχείο με βάση την περιοχή στην οποία βρίσκεται, ή την απόσταση από κάποιο σημείο του νησιού κλπ.

Επιπλέον, η χρήση του είναι δυνατόν να επεκταθεί και να καλύψει περισσότερα νησιά των Δωδεκανήσων. Αυτό θα ευνοούσε την διοργάνωση ταξιδιών μεταξύ των νησιών.

Τέλος, το συγκεκριμένο σύστημα θα μπορούσε να μετατραπεί σε εφαρμογή κινητής συσκευής (κινητό τηλέφωνο ή tablet pc), ώστε η χρήση του να γίνεται σε πραγματικό χρόνο, χωρίς να είναι απαραίτητη η πρόσβαση του επισκέπτη σε σταθερό υπολογιστή.

Βιβλιογραφία

1. C. J. Date , "Εισαγωγή στα συστήματα Βάσεων Δεδομένων", Κλειδάριθμος, 1996
2. Luke Welling Laura Thomson, "Ανάπτυξη Web εφαρμογών με Php και MySQL", Γκιούρδας, 2011
3. <http://el.wikipedia.org>
4. <http://www.techteam.gr/wiki/PHP>
5. <http://www.easyphp.org/>
6. <https://phpmyadmin-greek.readthedocs.org/en/latest/>
7. <http://www.php.net>
8. <http://www.mysql.com/>
9. <http://www.w3schools.com/>
10. <http://www.lib.teicrete.gr>
11. <http://www.sqlschool.gr/>
12. <http://www.w3.org/>

Παράρτημα: Διαφάνειες Παρουσίασης



ΣΧΕΔΙΑΣΗ ΚΑΙ ΥΛΟΠΟΙΗΣΗ
ΙΣΤΟΣΕΛΔΑΣ
ΤΟΥΡΙΣΤΙΚΟΥ ΟΔΗΓΟΥ ΤΗΣ ΡΟΔΟΥ
Ελευθέριος Νάκκας (ΑΜ: 2171)
Γεώργιος Σταυρής (ΑΜ: 2351)
Επιβλέπων Καθηγητής: Νικόλαος Παπαδάκης

ΠΕΡΙΕΧΟΜΕΝΑ

- Εισαγωγή
- Θεωρητικό πλαίσιο και προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν
- Ανάλυση και Παρουσίαση της Εφαρμογής
- Συμπεράσματα – Μελλοντική εργασία και επεκτάσεις

ΕΙΣΑΓΩΓΗ

- Κίνητρο
 - Η ενημέρωση του επισκέπτη και η διαφήμιση του νησιού σε μελλοντικούς επισκέπτες
- Στόχος
 - Παροχή πληροφοριών στον επισκέπτη της Ρόδου σχετικών με αξιοθέατα, διαμονή, διατροφή και μέσα μεταφοράς

ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ – ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΑ ΕΡΓΑΛΕΙΑ

- Στατικές – Δυναμικές Ιστοσελίδες
- HTML
- CSS
- PHP
- JavaScript
- EasyPHP, PHPMyAdmin, Apache, Dreamweaver
- SQL

Αρχική
σελίδα

ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ



ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

5

Προβολή
ξενοδοχείων

ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ



ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

6



Προβολή
αξιοθέατου

ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ. Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

7

Προβολή
εστιατορίων

ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ. Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

8

Προβολή
Χάρτη



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

Προβολή
Χάρτη



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

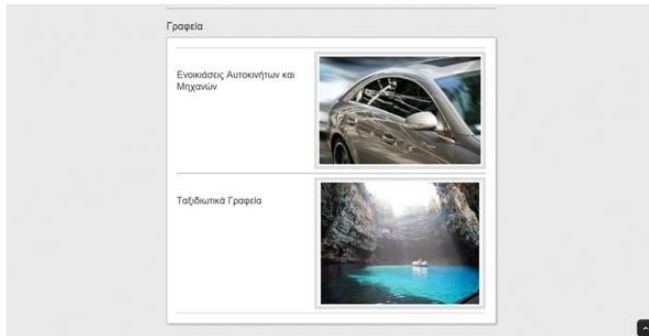
ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

9

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

10

Προβολή
Γραφείων

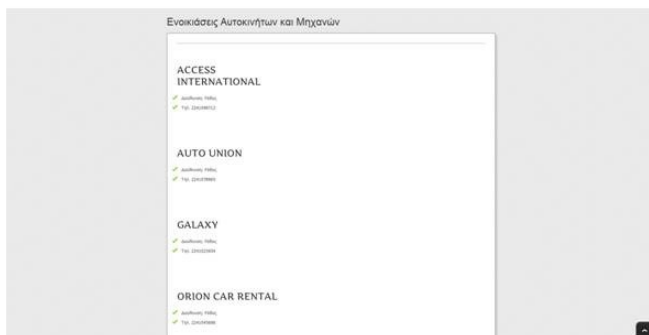


ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

11

Προβολή
γραφείων
ενοικιαζόμενων
οχημάτων



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

12



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

Φόρμα συμπλήρωσης στοιχείων για κράτηση δωματίου ή εισητηρίου ή και των δύο

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων

13

ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

Προβολή διαθέσιμων δωματίων για κράτηση

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης Πληροφορικής & Πολυμέσων

14

Προβολή
κωδικού
αριθμού
κράτησης



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

15

Προβολή
φωτογραφίας



ΑΝΑΛΥΣΗ ΚΑΙ ΠΑΡΟΥΣΙΑΣΗ ΕΦΑΡΜΟΓΗΣ

ΤΕΙ Κρήτης ΣΤΕΦ, Τμήμα Εφαρμοσμένης
Πληροφορικής & Πολυμέσων

16

ΣΥΜΠΕΡΑΣΜΑΤΑ - ΕΠΕΚΤΑΣΕΙΣ

○ Συμπεράσματα

- Πλήρης ενημέρωση για όλη την απαραίτητη πληροφορία (αξιοθέατα, εστιατόρια, ξενοδοχεία, μέσα μεταφοράς)
- Δυνατότητα κράτησης δωματίου ή εισιτηρίου
- Δυνατότητα ακύρωσης κράτησης

○ Επεκτάσεις

- Περισσότερα κριτήρια στις κρατήσεις
- Εφαρμογή σε περισσότερα νησιά των Δωδεκανήσων
- Υλοποίηση στη μορφή εφαρμογής για κινητές συσκευές