



**ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ**  
**ΕΦΑΡΜΟΓΕΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΣΤΗ ΔΙΟΙΚΗΣΗ ΚΑΙ**  
**ΟΙΚΟΝΟΜΙΑ**

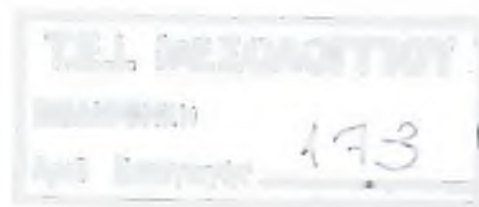
**ΤΙΤΛΟΣ ΠΤΥΧΙΑΚΗΣ**

**ΔΙΚΤΥΑΚΟΣ ΤΟΠΟΣ ΠΑΡΟΥΣΙΑΣΗΣ**  
**ΞΕΝΟΔΟΧΕΙΑΚΩΝ ΜΟΝΑΔΩΝ ΕΥΡΥΤΕΡΗΣ**  
**ΠΕΡΙΟΧΗΣ ΤΟΥ ΠΗΛΙΟΥ**

**Υπεύθυνος καθηγητής**  
**Αθανάσιος Σταυρακούδης**

**Αναστασία Θεοχαροπούλου**  
**ΑΜ: 9245**

**Ανδρέας Σοφαντζής**  
**ΑΜ: 9668**



## Ευχαριστίες...

Θα θέλαμε να ευχαριστήσουμε θερμά τον καθηγητή κ. Αθανάσιο Σταυρακούδη για τη σωστή καθοδήγηση και τη πολύτιμη βοήθειά του, καθώς επίσης και για τη συχνή επικοινωνία που είχαμε. Παράλληλα θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας για τη συμπαράσταση και την κατανόηση που έδειξαν καθ' όλη τη διάρκεια των σπουδών μας και ειδικότερα τη περίοδο που μας απασχόλησε η πραγματοποίηση της πτυχιακής μας εργασίας.

Θεοχαροπούλου Αναστασία  
Σοφαντζής Ανδρέας

1.	ΕΙΣΑΓΩΓΗ.....	4
2.	ΕΓΚΑΤΑΣΤΑΣΗ ΑΠΑΡΑΙΤΗΤΩΝ ΣΤΟΙΧΕΙΩΝ.....	7
2.1.	ΕΓΚΑΤΑΣΤΑΣΗ ΑΡΑΧΕ 1.33 .....	7
2.2.	ΕΓΚΑΤΑΣΤΑΣΗ ΜΥSQL 4.0.26 .....	9
2.3.	ΕΓΚΑΤΑΣΤΑΣΗ ΡΗΡ .....	9
2.4.	ΣΥΝΔΕΣΗ ΑΡΑΧΕ - ΡΗΡ.....	10
2.5.	ΕΓΚΑΤΑΣΤΑΣΗ ΡΗΡΜΥADMIN .....	10
3.	SQL .....	11
3.1.	ΟΙ ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ ΤΗΣ ΜΥSQL.....	11
3.2.	ΒΑΣΙΚΕΣ ΕΝΤΟΛΕΣ ΤΗΣ SQL.....	14
4.	ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ ΡΗΡ .....	22
4.1.	ΜΕΤΑΒΛΗΤΕΣ.....	22
4.2.	ΓΕΝΙΚΕΣ ΚΑΙ ΥΠΕΡ-ΓΕΝΙΚΕΣ ΜΕΤΑΒΛΗΤΕΣ.....	23
4.3.	ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ .....	24
5.	CSS (CASCADING STYLE SHEETS).....	27
5.1.	ΤΟ ΧΑΡΑΚΤΗΡΙΣΤΙΚΟ (ATTRIBUTE) CLASS .....	27
5.2.	ΤΟ ΚΕΙΜΕΝΟ ΤΟΥ CSS .....	27
5.3.	ΟΙ ΓΡΑΜΜΑΤΟΣΕΙΡΕΣ ΤΟΥ CSS.....	28
5.4.	ΤΑ ΠΕΡΙΓΡΑΜΜΑΤΑ ΤΟΥ CSS.....	29
5.5.	ΤΑ ΠΕΡΙΘΩΡΙΑ ΤΟΥ CSS .....	30
6.	ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ.....	32
7.	ADMINISTRATION PAGES.....	38
7.1.	ΤΙ ΕΙΝΑΙ .....	38
7.2.	ΠΩΣ ΓΙΝΕΤΑΙ .....	38
7.3.	ΠΙΝΑΚΑΣ ΕΛΕΓΧΟΥ.....	41
7.3.1.	Ξενοδοχεία. Εισαγωγή.....	41
7.3.2.	Ξενοδοχεία. Ενημέρωση / Διαγραφή .....	42
8.	ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ .....	45
8.1.	ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ .....	45
8.2.	ΔΗΜΙΟΥΡΓΙΑ ΠΙΝΑΚΩΝ.....	47
8.3.	ΑΝΑΚΤΗΣΗ ΔΕΔΟΜΕΝΩΝ .....	49
8.4.	ΕΙΣΑΓΩΓΗ ΔΕΔΟΜΕΝΩΝ .....	50
8.5.	SESSION & COOKIES.....	54
9.	ΒΙΒΛΙΟΓΡΑΦΙΑ .....	55

## 1. ΕΙΣΑΓΩΓΗ

Αναλαμβάνοντας την πραγματοποίηση πτυχιακής με τίτλο 'ΔΙΚΤΥΑΚΟΣ ΤΟΠΟΣ ΠΑΡΟΥΣΙΑΣΗΣ ΞΕΝΟΔΟΧΕΙΑΚΩΝ ΜΟΝΑΔΩΝ ΕΥΡΥΤΕΡΗΣ ΠΕΡΙΟΧΗΣ ΤΟΥ ΠΗΛΙΟΥ' κάναμε αρκετές έρευνες έως ότου καταλήξουμε στην σωστή δομή και σχεδιασμό της βάσης μας .Προσπαθήσαμε να μπούμε στη θέση κάποιου που θέλει να πάει διακοπές ή απλά από περιέργεια θέλει να δει το site μας και να βάλουμε στη σελίδα μας όλα αυτά που προφανώς χρειάζεται να γνωρίζει. Στην συνέχεια δημιουργήσαμε την έκθεση αυτή στην οποία υπάρχουν πολλές συγκεντρωμένες πληροφορίες που σχετίζονται με την εργασία μας αλλά και ότι χρησιμοποιήσαμε για τη δημιουργία της, ενώ παράλληλα παρέχουμε και ένα ειδικό εγχειρίδιο με οδηγίες που μπορούν να βοηθήσουν κάποιον να εγκαταστήσει τα προγράμματα που χρησιμοποιηθήκανε. Πιο συγκεκριμένα για την κατασκευή της ιστοσελίδας μας χρησιμοποιηθήκανε τα εξής προγράμματα:

- Dreamweaver MX 2004
- PHPMyAdmin
- Photoshop 8
- MySQL
- Microsoft Visio

Επίσης παραθέτουμε πληθώρα παραδειγμάτων βασισμένα πάντα στο θέμα της εργασία μας, με κύριο στόχο την παρουσίαση της σταδιακής εξέλιξης της εργασίας αλλά και την ουσιαστική κατανόηση της λειτουργίας της βάσης δεδομένων μας και άρα κατ' επέκταση και της ηλεκτρονικής μας σελίδας.

Ο κύριος λόγος δημιουργίας της σελίδας μας, είναι να μπορεί οποιοσδήποτε χρήστης χρησιμοποιώντας το site μας να μαζεύει πληροφορίες για τα ξενοδοχεία της Μαγνησίας και κυριότερα του Πηλίου. Παράλληλα έχει τη δυνατότητα να δει το χάρτη του Πηλίου, να βρει συγκεντρωμένα χρήσιμα τηλέφωνα και να ενημερωθεί για ενοικιάσεις αυτοκινήτων και για εστιατόρια.

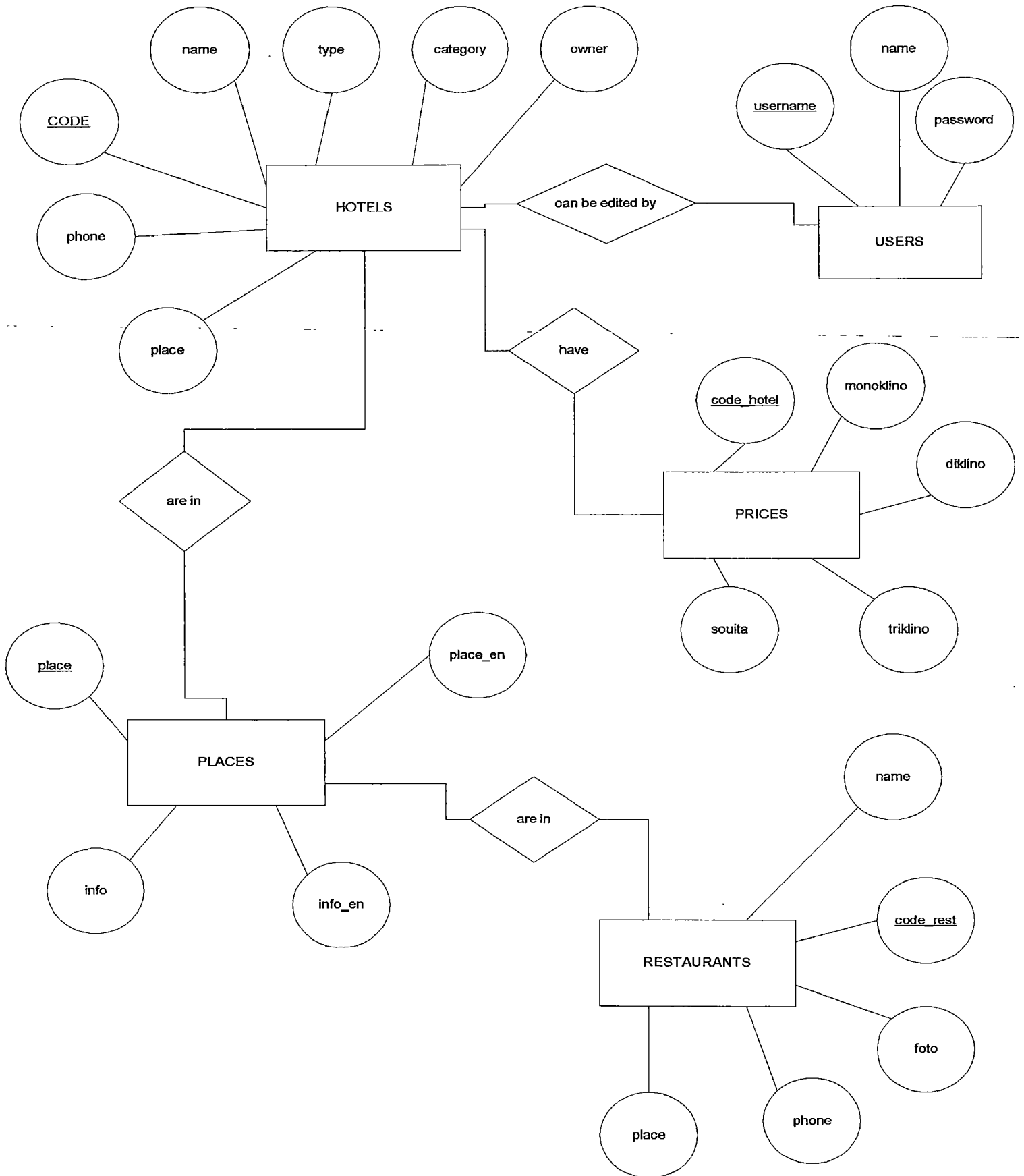
Ένας άλλος λόγος κατασκευής της ιστοσελίδας μας είναι να υπάρχει η δυνατότητα οι διαχειριστές του site να ανανεώνουν τις πληροφορίες, τη δομή, αλλά και τη λειτουργικότητα της σελίδας, όπως φυσικά και αν κριθεί απαραίτητο να μπορούν να γίνουν κάποιοι συμβιβασμοί ως προς το περιεχόμενο.

Θέλοντας να έχουμε μια αντικειμενική άποψη για το site μας, αλλά και για να μπορούμε να διορθώσουμε ή να καλύψουμε τυχόν ελλείψεις, το θέσαμε υπό δοκιμή σε περίπου 15 χρήστες του κοντινού μας περιβάλλοντος .Κατά γενική ομολογία βρήκαν ικανοποιητικό και λειτουργικό το site μας, ωστόσο υπήρχαν κάποιες σημαντικές παρατηρήσεις τις οποίες λάβαμε σοβαρά υπόψιν μας και προσπαθήσαμε να τις διορθώσουμε.

Προσπαθήσαμε και πιστεύουμε ότι στην έκθεση μας, καταφέραμε να έχουμε μία εκτενή παρουσίαση της πτυχιακής μας ,ώστε να μπορεί να γίνει κατανοητή και από κάποιον που δεν έχει ασχοληθεί ιδιαίτερα με τον προγραμματισμό και τις βάσεις δεδομένων.

Πριν ξεκινήσουμε την εργασία μας σχεδιάσαμε το σχήμα της βάσης μας με το μοντέλο οντοτήτων – συσχετίσεων. Αυτό μας βοήθησε στο να μην πηγαίνουμε στα τυφλά και να ξέρουμε ακριβώς πως θα είναι η βάση μας έτσι ώστε η υλοποίησή της να είναι απλή υπόθεση. Βέβαια στην πορεία υπήρξαν κάποιες αλλαγές αλλά το τελικό σχήμα δεν έχει μεγάλη διαφορά από το αρχικό. Στο παρακάτω σχήμα φαίνονται ακριβώς όλοι οι πίνακες, τα πεδία τους, τα αρχικά κλειδιά τους και όλες οι σχέσεις μεταξύ τους.

ΔΙΑΓΡΑΜΜΑ ΟΝΤΟΤΗΤΩΝ ΣΥΣΧΕΤΙΣΕΩΝ



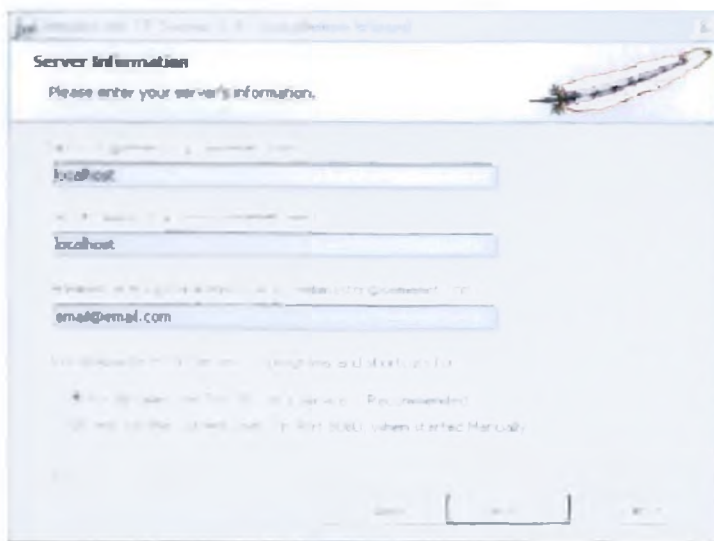
## 2. ΕΓΚΑΤΑΣΤΑΣΗ ΑΠΑΡΑΙΤΗΤΩΝ ΣΤΟΙΧΕΙΩΝ

Σ' αυτήν την ενότητα θα ασχοληθούμε με την εγκατάσταση του **APACHE 1.33** **MYSQL 4.0.26**, τη σύνδεση **PHP-APACHE** αλλά και την εγκατάσταση **PHPMYADMIN**.

Αρχικά να πούμε ότι πριν εγκαταστήσει κάποιος το APACHE θα πρέπει να διασφαλίσει ότι δεν τρέχει έναν Web server στο σύστημά του (π.χ μια προηγούμενη έκδοση του Apache, τον Microsoft Internet Information Server, ή τον Microsoft Personal Web Server). Καλό θα είναι λοιπόν να καταργήσει την εγκατάσταση των υπαρχόντων servers, ή να τους απενεργοποιήσει με οποιοδήποτε άλλο τρόπο. Μπορούμε να τρέχουμε πολλαπλούς Web servers, αλλά σε διαφορετικούς συνδυασμούς διευθύνσεων και θυρών.

### 2.1. Εγκατάσταση APACHE 1.33

Κατά την εγκατάσταση του APACHE 1.33 εμφανίζεται η παρακάτω μάσκα :



Εικόνα 1.1

στα πεδία της οποίας βάζουμε τις πληροφορίες όπως φαίνονται στην εικόνα 1.1

**Network Domain:** localhost

**Server Domain:** localhost

**Administrator's Email:** ένα απλό email (δεν έχει σημασία)

Η επόμενη μάσκα που εμφανίζεται είναι η ακόλουθη:



Εικόνα 1.2

Στην οποία επιλέγουμε Custom setup και σα φάκελο εγκατάστασης επιλέγουμε το c:\ όπως στην εικόνα 1.3 (αυτόματα θα εγκατασταθεί στο c:\apache)



Εικόνα 1.3



## 2.2. Εγκατάσταση MySQL 4.0.26

Κατά την διάρκεια εγκατάστασης MySQL 4.0.26 εμφανίζεται το εξής παράθυρο:



Εγκαθιστούμε στο c:\mysql και μετά επεξεργαζόμαστε το αρχείο my.ini στο c:\windows και προσθέτουμε τις παρακάτω γραμμές:

```
[server]
character-sets-dir=C:/mysql/share/charsets
default-character-set=greek
[client]
character-sets-dir=C:/mysql/share/charsets
default-character-set=greek
```

αυτήν την ενέργεια την κάνουμε για να μπορούμε να χρησιμοποιούμε τα ελληνικά χωρίς κανένα πρόβλημα.

## 2.3. Εγκατάσταση PHP

Για την εγκατάσταση της PHP κάνουμε τις παρακάτω ενέργειες:

- αντιγραφή φακέλου PHP στο c:\php
- αντιγράφουμε το αρχείο php4ts.dll στο c:\windows\system32
- αντιγράφουμε όλα τα αρχεία από το φάκελο dlls στο c:\windows\system32

- μετονομάζουμε το αρχείο `php.ini-recommended` σε `php.ini` και το αντιγράφουμε στο `c:\windows`
- κάνουμε edit το αρχείο `php.ini` και αλλάζουμε την ρύθμιση `'extension_dir'` σε `c:\php\extensions\` (προσοχή στην τελευταία slash)
- από τον κατάλογο `c:\php` αντιγράφουμε όλο το περιεχόμενο του φακέλου `mibs` στο `c:\usr\mibs`

## 2.4. Σύνδεση APACHE - PHP

Για την σύνδεση PHP-APACHE τώρα ακολουθούμε τα εξής βήματα:

- ανοίγουμε το αρχείο `httpd.conf` που βρίσκετε στον κατάλογο του `c:\apache\apache\conf`
- προσθέτουμε τις εξής γραμμές:

```
scriptalias /php/ "c:/php/"
```

```
addtype application/x-httpd-php .php .html
```

```
action application/x-httpd-php "/php/php.exe"
loadmodule php4_module c:/php/sapi/php4apache.dll
addtype application/x-httpd-php .php .html
```

## 2.5. Εγκατάσταση phpMyAdmin

Αντιγράφουμε το φάκελο `phpMyAdmin` στο `c:\apache\apache\htdocs`

στο φάκελο `phpMyAdmin` ανοίγουμε το αρχείο `config.inc.php` και βάζουμε στο `$cfg['PmaAbsoluteUri'] = ''`; την διεύθυνση που βρίσκεται το `phpMyAdmin` δηλαδή `$cfg['PmaAbsoluteUri'] = 'localhost/phpMyAdmin'`;

πάμε στο φάκελο `c:\mysql\bin` και τρέχουμε το `winmysqladmin.exe` μετά πάμε στον browser μας και τρέχουμε `localhost/phpMyAdmin/index.php` και ξεκινάμε να φτιάχνουμε τις βάσεις μας.

### 3. SQL

#### 3.1. Οι Τύποι Δεδομένων της MySQL

Ο σωστός ορισμός των πεδίων ενός πίνακα είναι σημαντικός για την συνολική βελτιστοποίηση μιας βάσης δεδομένων .Θα πρέπει πάντα να χρησιμοποιούμε τον τύπο δεδομένων και το μέγεθος που χρειαζόμαστε πραγματικά για κάθε επίπεδο , δηλαδή δε θα πρέπει να ορίζουμε ένα πεδίο με μέγεθος 10 χαρακτήρες εάν ξέρουμε ότι θα το χρησιμοποιούμε πάντα για την αποθήκευση τιμών 2 χαρακτήρων .Αυτοί οι τύποι πεδίων αναφέρονται σαν τύποι δεδομένων (data types) ,επειδή ορίζουν τα είδη των δεδομένων που μπορούμε να αποθηκεύουμε στα πεδία.

Η MySQL χρησιμοποιεί πολλούς διαφορετικούς τύπους δεδομένων , οι οποίοι ταξινομούνται σε τρεις κατηγορίες: αριθμητικοί ,ημερομηνίας /ώρας και αλφαριθμητικών .Θα πρέπει να δίνουμε ιδιαίτερη προσοχή στο θέμα των τύπων δεδομένων ,επειδή ο ορισμός του σωστού τύπου δεδομένων είναι πιο σημαντικός από οποιαδήποτε άλλη άποψη της διαδικασίας δημιουργίας ενός πίνακα.

#### Αριθμητικοί Τύποι

Η MySQL χρησιμοποιεί όλους τους στάνταρ αριθμητικούς τύπους δεδομένων της ANSI SQL.Συνεπώς εάν ερχόμαστε στην MySQL από ένα διαφορετικό σύστημα βάσεων δεδομένων ,οι ακόλουθοι ορισμοί είναι σχετικά γνωστοί .Στην ακόλουθη λίστα παρουσιάζονται οι πιο κοινοί τύποι δεδομένων ,μαζί με τις περιγραφές τους.

- INT : Ένας ακέραιος κανονικού μεγέθους ,ο οποίος μπορεί να είναι είτε προσημασμένος ,είτε μη-προσημασμένος .Εάν είναι προσημασμένος ,το υποστηριζόμενο πεδίο τιμών είναι από -2147483648 έως 2147483648 .Εάν είναι μη-προσημασμένος το υποστηριζόμενο πεδίο τιμών είναι από 0 έως 4294967295 .Το μέγιστο μήκος του είναι 11 ψηφία.
- TINYINT :Ένας πολύ μικρός ακέραιος ,ο οποίος μπορεί να είναι προσημασμένος ή μη-προσημασμένος .Εάν είναι προσημασμένος ,το υποστηριζόμενο πεδίο τιμών είναι από -128 έως 127 .Εάν είναι μη-προσημασμένος ,το υποστηριζόμενο πεδίο τιμών είναι από 0 έως 255 .Το μέγιστο μήκος του είναι 4 ψηφία.
- SMALLINT:Ένας μικρός ακέραιος ,ο οποίος μπορεί να είναι προσημασμένος ή μη-προσημασμένος.Εάν φέρει πρόσημο ,το υποστηριζόμενο πεδίο τιμών είναι από -32768 έως 32767 .Εάν δεν φέρει πρόσημο ,το υποστηριζόμενο πεδίο τιμών είναι από 0 έως 65535.Το μέγιστο μήκος του είναι 5 ψηφία.
- MEDIUMINT:Ένας μεσαίου μεγέθους ακέραιος ο οποίος μπορεί να είναι προσημασμένος ή μη-προσημασμένος .Εάν φέρει πρόσημο ,το υποστηριζόμενο πεδίο τιμών είναι από -8388608 έως 8388607.Εάν δεν φέρει πρόσημο ,το υποστηριζόμενο πεδίο τιμών είναι από 0 έως 16777215 .Το μέγιστο μήκος του είναι 9 ψηφία.
- BIGINT:Ένας μεγάλος ακέραιος ,ο οποίος μπορεί να είναι προσημασμένος ή μη-προσημασμένος .Εάν φέρει πρόσημο ,το υποστηριζόμενο πεδίο τιμών είναι από -9223372036854775808 έως

9223372036854775807 .Εάν δεν φέρει πρόσημο το υποστηριζόμενο πεδίο τιμών είναι από 0 έως 18446744073709551615 .Το μέγιστο μήκος του είναι 11 ψηφία.

- **FLOAT(M,D)**:Ένας αριθμός κινητής υποδιαστολής ,ο οποίος δεν μπορεί να είναι μη-προσημασμένος .Μπορούμε να ορίσουμε το μέγεθος εμφάνισης(M) και το πλήθος δεκαδικών (D) .Αυτές οι παράμετροι δεν είναι απαραίτητες εάν δεν τις καθορίσουμε ,χρησιμοποιούνται οι τιμές 10,2, όπου το 2 είναι ο αριθμός των δεκαδικών και το 10 είναι ο συνολικός αριθμός ψηφίων (συμπεριλαμβανομένων των δεκαδικών) .Η ακρίβεια μιας τιμής τύπου **FLOAT** μπορεί να φτάσει τα 24 δεκαδικά ψηφία.
- **DOUBLE(M,D)**:Ένας αριθμός κινητής υποδιαστολής διπλής ακρίβειας ,ο οποίος δεν μπορεί να είναι μη-προσημασμένος .Μπορείτε να ορίσουμε το μέγεθος εμφάνισης(M)και το πλήθος των δεκαδικών(D) .Αυτές οι παράμετροι δεν είναι απαραίτητες εάν δεν τις καθορίσουμε ,χρησιμοποιούνται οι τιμές 16,4 ,όπου το 4 είναι ο αριθμός των δεκαδικών .Η ακρίβεια μιας τιμής τύπου **DOUBLE** μπορεί να φτάσει τα 53 δεκαδικά ψηφία .Ο όρος **REAL** είναι συνώνυμος του όρου **DOUBLE**.
- **DECIMAL(M,D)**:Ένας μη-συμπιεσμένος αριθμός κινητής υποδιαστολής ,ο οποίος δεν μπορεί να είναι μη-προσημασμένος .Στους μη-συμπιεσμένους αριθμούς τύπου **DECIMAL** ,κάθε δεκαδικό ψηφίο αντιστοιχεί σε ένα byte . Είναι υποχρεωτικό να ορίσουμε το μέγεθος εμφάνισης (M) και το πλήθος των δεκαδικών (D) .Το **NUMERIC** είναι ένα συνώνυμο για τον τύπο **DECIMAL**.

Απ' όλους τους τύπους που υποστηρίζει η MySQL , θα χρησιμοποιούμε πιο συχνά τον **INT** .Εάν ορίσουμε τα πεδία μας σε μικρότερο μέγεθος απ' όσο πραγματικά χρειαζόμαστε ,μπορεί να συναντήσουμε διάφορα προβλήματα ,για παράδειγμα ,εάν ορίσουμε ένα πεδίο αποθήκευσης κωδικών με τύπο δεδομένων **TINYINT** (μη-προσημασμένο) ,δεν θα μπορέσουμε να προσθέσουμε την 256 εγγραφή στον πίνακά μας εάν το πεδίο αυτό είναι πρωτεύον κλειδί (και ,κατά συνέπεια απαιτούμενο).

**Σημείωση:** Κατά την παρουσίαση των αριθμητικών τύπων δεδομένων χρησιμοποιήθηκαν οι όροι προσημασμένος και μη-προσημασμένος .Όπως θυμόμαστε από την άλγεβρα του γυμνασίου ,ένας προσημασμένος ακέραιος είναι ένας θετικός ή αρνητικός ακέραιος αριθμός (δηλαδή φέρει πρόσημο) ,ενώ ένας μη-προσημασμένος ακέραιος είναι ένας μη-αρνητικός ακέραιος αριθμός.

## Ημερομηνίας /Ωρας

Η MySQL υποστηρίζει αρκετούς τύπους δεδομένων για την αποθήκευση τιμών ημερομηνίας /ώρας και αυτοί οι τύποι δεδομένων είναι ιδιαίτερα ευέλικτοι όσον αφορά την είσοδο δεδομένων από τον χρήστη. Με άλλα λόγια μπορούμε να εισάγουμε ημερομηνίες οι οποίες δεν είναι πραγματικές ,όπως για παράδειγμα η 30 Φεβρουαρίου .Μπορούμε επίσης να αποθηκεύουμε ημιτελείς τιμές ημερομηνίας-τιμές από τις οποίες λείπουν κάποια στοιχεία .Η ευελιξία των τύπων δεδομένων ημερομηνίας /ώρας της MySQL σημαίνει επίσης ότι όλη η ευθύνη για τον έλεγχο των ημερομηνιών επαφίεται στον

δημιουργό της εφαρμογής . Η MySQL ελέγχει μόνο την εγκυρότητα δύο στοιχείων :ότι η τιμή για τον μήνα ανήκει στο πεδίο τιμών 0-12 και η τιμή για την ημέρα ανήκει στο πεδίο τιμών 0-31.

Η MySQL δεν ελέγχει αυτόματα εάν η 30<sup>η</sup> ημέρα του δεύτερου μήνα (Φεβρουαρίου) είναι μία έγκυρη ή μη-έγκυρη ημερομηνία.

Οι τύποι δεδομένων που υποστηρίζει η MySQL για τιμές ημερομηνίας /ώρας είναι:

- **DATE:**Μία τιμή ημερομηνίας με την μορφή ΕΕΕΕ-MM-ΗΗ ,μεταξύ 1000-01-01 και 9999-12-31 .Για παράδειγμα , η 30<sup>η</sup> Δεκεμβρίου 1973 θα αποθηκευτεί σαν 1973-12-30.
- **DATETIME:**Ένας συνδυασμός τιμών ημερομηνίας και ώρας με την μορφή ΕΕΕΕ-MM-ΗΗ ΩΩ:ΛΛ:ΔΔ μεταξύ 1000-01-01 00:00:00 και 9999-12-31 23:59:59 .Για παράδειγμα ,η ώρα 3:30 το μεσημέρι της 30<sup>ης</sup> Δεκεμβρίου 1973 θα αποθηκευτεί σαν 1973-12-30 15:30:00.
- **TIMESTAMP:**Μία χρονική ένδειξη ανάμεσα στα μεσάνυχτα της 1<sup>ης</sup> Ιανουαρίου 1970 και στο έτος 2037 .Μπορείτε να ορίζετε πολλαπλά μήκη στο πεδίο **TIMESTAMP** ,το οποίο συσχετίζεται άμεσα με την οποιαδήποτε τιμή αποθηκεύεται σ' αυτό .Το προεπιλεγμένο μήκος για τον τύπο δεδομένων **TIMESTAMP** είναι 14 και αποθηκεύει τιμές της μορφής ΕΕΕΕΜΜΗΗΩΩΛΛΔΔ .

Η μορφή αυτή δείχνει παρόμοια με την μορφή του τύπου δεδομένων **DATETIME** , χωρίς τις παύλες ανάμεσα στους αριθμούς η ώρα 3:30 το μεσημέρι της 30<sup>ης</sup> Δεκεμβρίου του 1973 αποθηκεύεται σαν 19731230153000 .Άλλα μεγέθη διαθέσιμα για τον τύπο δεδομένων **TIMESTAMP** είναι τα 12 (ΕΕΜΜΗΗΩΩΛΛΔΔ) , 8(ΕΕΕΕΜΜΗΗ) ΚΑΙ 6(ΕΕΜΜΗΗ).

- **TIME:**Αποθηκεύει την ώρα σε μορφή ΩΩ:ΛΛ:ΔΔ.
- **YEAR(M):**Αποθηκεύει ένα έτος σε διψήφια ή τετραψήφια μορφή.Εάν η παράμετρος μεγέθους(M)οριστεί σε 2 (π .χ YEAR(2)), το έτος μπορεί να είναι από 1970 έως 2069(70 έως 69).Εάν η παράμετρος μεγέθους οριστεί σε 4, το έτος μπορεί να είναι από 1901 έως 2155.Το προεπιλεγμένο μέγεθος είναι 4.

Κατά πάσα πιθανότητα θα χρησιμοποιούμε τους τύπους δεδομένων **DATETIME** ή **DATE** πιο συχνά από οποιονδήποτε άλλο σχετιζόμενο με ημερομηνίες /ώρες τύπο δεδομένων.

## Τύποι Αλφαριθμητικών

Αν και οι τύποι δεδομένων για τους αριθμούς και τις τιμές ημερομηνίας /ώρας είναι χρήσιμοι ,τα περισσότερα δεδομένα που αποθηκεύουμε θα είναι σε μορφή αλφαριθμητικών .Η ακόλουθη λίστα περιγράφει τους πιο κοινούς τύπους δεδομένων για αλφαριθμητικά που υποστηρίζει η MySQL .

- **CHAR(M):**Ένα αλφαριθμητικό σταθερού μεγέθους ,με μέγεθος μεταξύ 1 και 255 χαρακτήρες(για παράδειγμα ,CHAR(5)) ,στο οποίο προστίθενται κενά διαστήματα στα δεξιά μέχρι να φτάσει στο προσδιοριζόμενο μήκος .Ο ορισμός μήκους δεν είναι απαραίτητος , αλλά το προεπιλεγμένο μήκος είναι 1.
- **VARCHAR(M):**Ένα αλφαριθμητικό μεταβλητού μεγέθους ,με μήκος από 1 έως 255 χαρακτήρες(για παράδειγμα ,VARCHAR(25)) .Θα

πρέπει να ορίζουμε την παράμετρο μήκους όταν δημιουργούμε ένα πεδίο VARCHAR.

- **BLOB or TEXT:** Ένα πεδίο με μέγιστο μέγεθος 65535 χαρακτήρες .Ο όρος BLOB είναι ακρωνύμιο του Binary Large Objects ,που σημαίνει μεγάλα αντικείμενα δυαδικής μορφής .Ο τύπος BLOB χρησιμοποιείται για την αποθήκευση μεγάλων ποσοτήτων δυαδικών δεδομένων ,όπως για παράδειγμα εικόνες ή άλλοι τύποι αρχείων .Τα πεδία που ορίζονται με τύπο TEXT αποθηκεύουν επίσης μεγάλες ποσότητες δεδομένων .Η διαφορά μεταξύ των δύο είναι ότι οι λειτουργίες ταξινόμησης και σύγκρισης που εκτελούνται στα αποθηκευμένα δεδομένα κάνουν διάκριση μεταξύ κεφαλαίων /πεζών χαρακτήρων για τα δεδομένα τύπου BLOB ,ενώ αυτό δεν ισχύει για τα πεδία με τύπο δεδομένων TEXT .Οι τύποι δεδομένων BLOB και TEXT δεν απαιτούν τον καθορισμό μεγέθους τους.
- **TINYBLOB or TINYTEXT:** Ένα πεδίο τύπου BLOB ή TEXT με μέγιστο μέγεθος 255 χαρακτήρες .Για τους τύπους TINYBLOB ή TINYTEXT δεν καθορίζεται μέγεθος.
- **MEDIUMBLOB or MEDIUMTEXT:** Ένα πεδίο τύπου BLOB ή TEXT με μέγιστο μέγεθος 16777215 χαρακτήρες .Για τους τύπους MEDIUMBLOB ή MEDIUMTEXT δεν καθορίζεται μέγεθος.
- **LOB or LONGTEXT:** Ένα πεδίο τύπου BLOB ή TEXT με μέγιστο μέγεθος 4294967295 χαρακτήρες .Για τους τύπους LOB ή LONGTEXT δεν καθορίζεται μέγεθος.
- **ENUM:** Μία απαρίθμηση ,δηλαδή μία λίστα .Όταν ορίζουμε έναν τύπο ENUM ουσιαστικά δημιουργούμε μία λίστα στοιχείων από τα οποία πρέπει να επιλεγεί μία τιμή (ή μπορεί να είναι NULL) .Για παράδειγμα εάν θέλουμε το πεδίο μας να περιέχει μία από τις τιμές "A" "B" ή "C" ,θα έπρεπε να το ορίσουμε σαν ENUM('A', 'B', 'C') ,έτσι ώστε μόνο αυτές οι τιμές (ή η τιμή NULL) να μπορούν να εισαχθούν στο πεδίο .Ένας τύπος ENUM μπορεί να περιέχει έως 65535 διαφορετικές τιμές .Ο τύπος ENUM χρησιμοποιεί έναν δείκτη για την αποθήκευση των στοιχείων.

Σημείωση: Ο τύπος SET(σύνολο) είναι παρόμοιος με τον ENUM στο ότι ορίζεται σαν λίστα .Ωστόσο ,ο τύπος SET αποθηκεύεται σαν μία πλήρης τιμή και όχι σαν ένας δείκτης προς μία τιμή ,όπως ισχύει με τους τύπους απαρίθμησης.

### 3.2. Βασικές Εντολές της SQL

Σ 'αυτό το κεφάλαιο θα κάνουμε μια εισαγωγή στην σύνταξη της SQL ,της γλώσσας που χρησιμοποιούμε για να δημιουργήσουμε και για να χειριζόμαστε πίνακες βάσεων δεδομένων MySQL .Πιο συγκεκριμένα θα ασχοληθούμε με τα ακόλουθα:

- Χρήση της εντολής CREATE TABLE για την δημιουργία ενός πίνακα.
- Χρήση της εντολής INSERT για την εισαγωγή εγγραφών σε πίνακες.

- Χρήση της εντολής SELECT για την ανάκτηση εγγραφών από πίνακες βάσεων δεδομένων.
- Χρήση βασικών συναρτήσεων και των προτάσεων WHERE και GROUP BY σε εκφράσεις της εντολής SELECT.
- Χρήση των εντολών UPDATE και REPLACE για την τροποποίηση υπάρχουσων εγγραφών .
- Χρήση της εντολής DELETE για τη διαγραφή εγγραφών.
- Χρήση των εγγενών συναρτήσεων χειρισμού αλφαριθμητικών της MySQL.
- Χρήση των εγγενών συναρτήσεων χειρισμού τιμών ημερομηνίας/ώρας της MySQL.

## Η Εντολή για την Δημιουργία Πινάκων

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

1. Το όνομα του πίνακα
2. Τα ονόματα των πεδίων
3. Έναν ορισμό για κάθε πεδίο

Η γενική σύνταξη για την εντολή δημιουργίας πίνακα είναι:

**CREATE TABLE όνομα \_πίνακα (όνομα \_στήλης τύπος \_στήλης);**

Το όνομα του πίνακα είναι δικής μας επιλογής ,αλλά θα πρέπει να αντικατοπτρίζει την χρήση ή τον ρόλο του πίνακα .Για παράδειγμα ,εάν έχουμε έναν πίνακα για την παρακολούθηση των κρατήσεων ενός ξενοδοχείου ,δεν θα ήταν συνετό εκ μέρους μας να του δώσουμε το όνομα s . Ένα λογικό και περιγραφικό όνομα θα μπορούσε να είναι reservations\_hotel. Παρόμοια ,τα ονόματα των πεδίων θα πρέπει να είναι όσο το δυνατόν πιο περιγραφικά και σχετιζόμενα με τον ρόλο των πεδίων ή με τα δεδομένα που αποθηκεύουν .Για παράδειγμα ,για το πεδίο το οποίο θα αποθηκεύει τα ονόματα των πελατών ενός ξενοδοχείου ,ένα καλό όνομα είναι customers\_names.

Το ακόλουθο παράδειγμα δημιουργεί έναν γενικευμένο πίνακα

```
CREATE TABLE `hotels` (
  `code` int(6) NOT NULL auto_increment,
  `name` varchar(20) NOT NULL default '',
  `type` varchar(35) NOT NULL default '',
  `owner` varchar(40) NOT NULL default '',
  `phone` varchar(10) NOT NULL default '',
  `place` varchar(35) NOT NULL default '',
  `category` varchar(6) NOT NULL default '',
  `foto` varchar(10) default NULL,
  `info` varchar(50) default NULL,
  `last_change` varchar(10) default NULL,
  PRIMARY KEY (`code`)
)
```

Σημείωση: Το πεδίο ορίζεται σαν πρωτεύον κλειδί(primary key). Χρησιμοποιώντας το auto\_increment(αυτόματη αύξηση τιμής) σαν χαρακτηριστικό του πεδίου, λέμε στην MySQL να εισάγει αυτόματα τον επόμενο διαθέσιμο αριθμό στο πεδίο κάθε εγγραφής που εισάγουμε.

Ο MySQL server ανταποκρίνεται με το μήνυμα Query OK κάθε φορά που εκτελείται επιτυχώς ένα ερώτημα, ανεξαρτήτως του τύπου του. Σε κάθε άλλη περίπτωση εμφανίζεται ένα μήνυμα σφάλματος.

## Χρήση της εντολής SELECT

Η SELECT είναι η εντολή της SQL που χρησιμοποιείται για την ανάκτηση εγγράφων από έναν πίνακα. Η σύνταξη αυτής της εντολής μπορεί να είναι εντελώς απλή, ή εξαιρετικά πολύπλοκη. Καθώς θα εξοικειωνόμαστε όλο και περισσότερο με τον προγραμματισμό εφαρμογών βάσεων δεδομένων, θα μάθουμε πως να εμπλουτίζουμε τις εντολές SELECT σε τόσο μεγάλο βαθμό, έτσι ώστε τελικά ένα μεγάλο μέρος της απαιτούμενης εργασίας να γίνεται από την ίδια την βάση δεδομένων, χωρίς να επιβαρυνόμαστε με συγγραφή πρόσθετου κώδικα.

Στην απλούστερη μορφή της, η σύνταξή της είναι κάπως έτσι:

```
SELECT εκφράσεις _και_ στήλες FROM όνομα _πίνακα
[WHERE συνθήκη _αληθής]
[ORDER BY κάποια _στήλη [ ASC | DESC ] ]
[LIMIT μετάθεση , γραμμές]
```

Ας ξεκινήσουμε με την πρώτη γραμμή :

```
SELECT εκφράσεις _και_ στήλες FROM όνομα _πίνακα
```

Μία ιδιαίτερα βολική έκφραση είναι το σύμβολο \* το οποίο αντιπροσωπεύει 'τα πάντα'. Συνεπώς, για να επιλέξουμε 'τα πάντα' (όλες τις γραμμές και όλες τις στήλες) από τον πίνακα hotels η εντολή SELECT θα πρέπει να χει την ακόλουθη μορφή:

```
SELECT * FROM hotels ;
```

Εάν θέλουμε να επιλέξουμε μόνο συγκεκριμένες στήλες, αντικαταστήσουμε τον χαρακτήρα \* με τα ονόματα των επιθυμητών στηλών διαχωρισμένα με κόμματα.

Η ακόλουθη εντολή επιλέγει μόνο τα πεδία name από τον πίνακα hotels:

```
SELECT name FROM hotels;
```

Αυτή η σύνταξη θα μας δώσει σαν αποτέλεσμα τα ονόματα όλων των ξενοδοχείων.

## Ταξινόμηση Των Αποτελεσμάτων Που Επιστρέφει Η Εντολή SELECT



Κατά κύριο λόγο τα αποτελέσματα των ερωτημάτων που εκτελείτε με την εντολή SELECT σε έναν πίνακα μιας βάσης δεδομένων εμφανίζονται με τη σειρά με την οποία εισάγουμε τις εγγραφές στον πίνακα ,χωρίς κάποιο είδος ταξινόμησης .Εάν θέλουμε να ταξινομήσουμε τα αποτελέσματα με έναν συγκεκριμένο τρόπο ,όπως για παράδειγμα κατά ημερομηνία ,κωδικό ,όνομα κ . λ . π ,θα πρέπει να καθορίσουμε αυτή την απαίτησή μας σε στην εντολή χρησιμοποιώντας την πρόταση ORDER BY . Στο αποτέλεσμα της ακόλουθης εντολής ,οι εγγραφές θα εμφανιστούν ταξινομημένες κατά όνομα εστιατορίου.

```
SELECT * FROM restaurants ORDER BY name;
```

Και σαν έξοδο θα έχει:

**Εντολή SQL:** SELECT \* FROM `restaurants` ORDER BY name  
**Εγγραφές:** 5

code_rest	name	place	phone	foto
4	PANTHEON	ΜΑΚΡΥΝΙΤΣΑ	2428099143	pantheon.jpg
1	ΔΕΣΠΟΤΙΚΟ	ΠΟΡΤΑΡΙΑ	2428099020	despotiko.jpg
2	ΚΑΡΝΑΓΙΟ	ΚΑΤΩ ΛΕΧΩΝΙΑ	2428093393	karnagio.jpg
3	ΜΑΡΑΜΠΟΥ	ΑΦΗΣΣΟΣ	2423033402	marabou.jpg
5	ΣΑΛΚΙΜΙ	ΜΗΛΙΕΣ	2423086010	salkimi.jpg

Χρήσιμη Πληροφορία: Όταν επιλέγουμε εγγραφές από έναν πίνακα χωρίς να καθορίσουμε μία σειρά ταξινόμησης ,τα αποτελέσματα του ερωτήματος μας μπορεί να εμφανιστούν ταξινομημένα σύμφωνα με τη τιμή του πεδίου-κλειδιού αλλά μπορεί και όχι .Αυτό συμβαίνει επειδή η MySQL επαναχρησιμοποιεί τον χώρο που καταλάμβαναν οι διαγραμμένες γραμμές(εγγραφές) .Με άλλα λόγια εάν προσθέσουμε εγγραφές με κωδικούς 1 έως 5 ,διαγράψουμε την εγγραφή με κωδικό 4 και κατόπιν προσθέσουμε μία επόμενη εγγραφή (η οποία θα λάβει τον κωδικό 6) ,οι εγγραφές θα εμφανίζονται στον πίνακα με την εξής σειρά 1,2,3,6,5.

Η προεπιλεγμένη σειρά ταξινόμησης που χρησιμοποιεί η πρόταση ORDER BY είναι αύξουσα (ASC) τα αλφαριθμητικά ταξινομούνται αλφαβητικά ,από το Α ως το Ζ ,οι ακέραιοι αυξητικά ξεκινώντας από το 0 και οι ημερομηνίες από την παλαιότερη έως την νεότερη .Μπορούμε επίσης να καθορίσουμε φθίνουσα σειρά ταξινόμησης χρησιμοποιώντας την δεσμευμένη λέξη DESC. Δεν περιοριζόμαστε σε ταξινόμηση βάσει ενός μόνο πεδίου μπορούμε να καθορίσουμε όσα πεδία θέλουμε ,διαχωρίζοντας τα ονόματά τους με κόμματα .Η σειρά αύξησης αντικατοπτρίζει την σειρά με την οποία κατονομάζουμε τα πεδία .

### Χρήση της Πρότασης WHERE στα Ερωτήματά μας

Μέχρι τώρα είδαμε αρκετούς τρόπους για να ανακτούμε συγκεκριμένες στήλες από τους πίνακες της βάσης δεδομένων μας ,αλλά όχι συγκεκριμένες γραμμές(εγγραφές) .

Αυτή ακριβώς την ανάγκη έρχεται να καλύψει η πρόταση WHERE .Από την βασική σύνταξη της SELECT ,μπορούμε να δούμε ότι η πρόταση WHERE χρησιμοποιείται για τον καθορισμό μιας συγκεκριμένης συνθήκης:

SELECT εκφράσεις\_και\_στήλες FROM όνομα\_πίνακα  
[WHERE συνθήκη\_αληθείς]

SELECT \* FROM hotels WHERE place='ΑΦΗΣΣΟΣ';

Εντολή SQL: SELECT \* FROM hotels WHERE place='ΑΦΗΣΣΟΣ'  
Εγγραφές: 12

code	name	type	owner	phone	place	category	foto	info	last_change
1	ΜΑΪΣΤΡΑΛΙ	ΞΕΝΟΔΟΧΕΙΟ	Μ. ΓΙΑΜΑΚΟΥ	2423033472	ΑΦΗΣΣΟΣ	Γ	NULL		0000-00-00
6	ΚΑΤΙΑ	ΞΕΝΟΔΟΧΕΙΟ	Μ. ΚΡΑΤΗΡΑΣ	2423033293	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
5	ΓΑΛΗΝΗ	ΞΕΝΟΔΟΧΕΙΟ	Ζ. ΜΑΡΙΝΑΓΗΣ	2423033214	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
7	ΦΑΡΟΣ	ΞΕΝΟΔΟΧΕΙΟ	Δ. ΓΙΑΝΝΟΥΛΗ	2423033293	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
8	ΕΙΡΗΝΗ	ΞΕΝΟΔΟΧΕΙΟ	ΕΙΡ. ΚΟΥΤΣΟΦΛΙΝΗ	2423033287	ΑΦΗΣΣΟΣ	Α	NULL	NULL	0000-00-00
9	ΜΠΑΡΜΠΑ-ΘΥΜΙΟΣ	ΞΕΝΟΔΟΧΕΙΟ	Ε. ΣΩΤΗΡΙΟΥ	2423033401	ΑΦΗΣΣΟΣ	Α	NULL	NULL	0000-00-00
11	ΑΦΗΣΣΟΣ		Μ. ΓΑΛΑΝΟΥ	2423033438	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
12	ΕΥΗ		Γ. ΚΑΡΑΜΠΑΣΗΣ	2423033340	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
13	ΚΑΛΛΙΠΤΕΡΗ		Γ. ΣΚΥΤΟΥΔΗ	2423033302	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
14	ΟΛΓΑ		Κ.ΓΑΛΑΝΟΥ	2423033310	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
15	ΠΑΝΟΡΑΜΑ		Ν.ΑΙΒΑΖΗΣ	2423033373	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00
16	ΡΕΝΑ		Δ.ΦΡΑΓΚΟΓΕΩΡΓΗ	2423033439	ΑΦΗΣΣΟΣ	Β	NULL	NULL	0000-00-00

## Χρήση των Τελεστών σε Προτάσεις WHERE

Σε προηγούμενα παραδείγματα χρησιμοποιούσαμε τον τελεστή ίσον(=) σε προτάσεις WHERE για να καθορίσουμε την αλήθεια μιας συνθήκης – ένα πράγμα είναι ίσο με ένα άλλο. Μπορούμε να χρησιμοποιούμε πολλούς τύπους τελεστών οι δημοφιλέστεροι είναι οι τελεστές σύγκρισης και οι λογικοί τελεστές. Οι τελεστές σύγκρισης, παρουσιάζονται στον παρακάτω πίνακα.

Τελεστής	Σημασία
=	Ίσο
!=	Άνισο
<=	Μικρότερο από ή ίσο με
<	Μικρότερο
>=	Μεγαλύτερο από ή ίσο με
>	Μεγαλύτερο

Υπάρχει επίσης ένας βολικός τελεστής με όνομα BETWEEN, ο οποίος είναι χρήσιμος για συγκρίσεις ακέραιων και άλλων δεδομένων επειδή αναζητά τα αποτελέσματα του μεταξύ μιας ελάχιστης και μιας μέγιστης τιμής.

Υπάρχουν επίσης οι λογικοί τελεστές οι οποίοι μας επιτρέπουν να χρησιμοποιούμε πολλαπλές συγκρίσεις στην πρόταση WHERE. Οι βασικοί λογικοί τελεστές είναι οι AND (σύζευξη) και OR (διάζευξη). Όταν χρησιμοποιούμε τον τελεστή AND, όλες οι συγκρίσεις που περιλαμβάνουμε στην πρόταση WHERE πρέπει να αποτιμηθούν σαν αληθείς. Από την άλλη, ο

τελεστής OR απαιτεί να είναι αληθής τουλάχιστον μία από τις συγκρίσεις της πρότασης WHERE .Επίσης ,μπορούμε να χρησιμοποιούμε τον τελεστή IN για να καθορίσουμε μία λίστα των στοιχείων που θέλουμε να ταιριάξουμε.

## Επιλογή Δεδομένων από Πολλαπλούς Πίνακες

Δεν περιοριζόμαστε στην επιλογή δεδομένων μόνο από έναν πίνακα τη φορά. Εάν ίσχυε ένας τέτοιος περιορισμός, ο προγραμματισμός εφαρμογών θα ήταν μία ιδιαίτερα επίπονη και κουραστική εργασία. Όταν επιλέγουμε δεδομένα από περισσότερους του ενός πίνακα με μία εντολή SELECT ,οι πίνακες ενώνονται(join) για την παραγωγή του αποτελέσματος.

Για παράδειγμα στην εργασία μας επιλέγουμε δύο από τους πίνακες μας prices(τιμές) και hotels(ξενοδοχεία) για να δείξουμε πως χρησιμοποιώντας δύο ξεχωριστές εντολές SELECT μπορούμε να επιλέξουμε όλες τις γραμμές από τους δύο πίνακες.

```
select * from prices;
select * from hotels;
```

Όταν θέλουμε να επιλέξουμε δεδομένα ταυτόχρονα από δύο πίνακες , υπάρχουν ορισμένες διαφορές στην σύνταξη της εντολής SELECT. Κατ' αρχήν θα πρέπει να διασφαλίσουμε ότι όλοι οι πίνακες από τους οποίους θέλουμε να ανακτήσουμε δεδομένα περιλαμβάνονται στην πρόταση FROM της εντολής SELECT.

Χρησιμοποιώντας σαν παράδειγμα τους πίνακες prices και hotels ,αν επιθυμούμε απλά να επιλέξουμε όλα τα πεδία και όλες τις εγγραφές από τους δύο πίνακες, τότε πρέπει να χρησιμοποιήσουμε την ακόλουθη εντολή SELECT:

```
select * from prices, hotels;
```

## Χρήση της Εντολής INSERT

Αφού δημιουργήσουμε μερικούς πίνακες ,θα χρησιμοποιούμε την εντολή INSERT της SQL για να προσθέτουμε νέες εγγραφές σ' αυτούς .Η βασική σύνταξη της εντολής INSERT είναι:

**INSERT INTO όνομα \_πίνακα (λίστα στηλών) VALUES (τιμές στηλών);**

Στην λίστα των τιμών μέσα στις παρενθέσεις τα αλφαριθμητικά θα πρέπει να περικλείονται σε εισαγωγικά. Η SQL χρησιμοποιεί εξ ορισμού αποστρόφους, αλλά η MySQL μας επιτρέπει να επιλέξουμε εάν θα χρησιμοποιούμε αποστρόφους ή εισαγωγικά. Επίσης θα πρέπει να χρησιμοποιούμε το χαρακτήρα \ πριν από τα εισαγωγικά που περιλαμβάνονται στο ίδιο το αλφαριθμητικό .

Χρήσιμη Πληροφορία: Όταν εισάγουμε ακέραιες τιμές δεν χρειάζεται να τις περικλείουμε σε εισαγωγικά ή αποστρόφους .

## Χρήση της Εντολής UPDATE

Η UPDATE είναι μία εντολή της SQL η οποία χρησιμοποιείται για την τροποποίηση του περιεχομένου μιας ή περισσότερων στηλών σε μία υπάρχουσα εγγραφή ενός πίνακα .Στην απλούστερη μορφή της η σύνταξη της εντολής UPDATE είναι:

```
UPDATE όνομα _ πίνακα
SET στήλη1= 'νέα τιμή' ,
στήλη2= 'νέα τιμή2'
[WHERE συνθήκη _ αληθής]
```

Η γενική διαδικασία για την ενημέρωση μιας εγγραφής είναι παρόμοια με αυτή που χρησιμοποιείται για την εισαγωγή μιας εγγραφής δηλαδή, τα δεδομένα που εισάγουμε πρέπει να είναι κατάλληλα για τον τύπο δεδομένων του πεδίου που τροποποιούμε και θα πρέπει επίσης να περικλείουμε τα αλφαριθμητικά μας σε αποστρόφους ή εισαγωγικά ,χρησιμοποιώντας χαρακτήρες \ αν είναι απαραίτητο.

## Χρήση της Εντολής DELETE

Η βασική σύνταξη για την εντολή DELETE είναι:

```
DELETE FROM όνομα _ πίνακα
[WHERE συνθήκη _ αληθής]
[LIMIT γραμμές]
```

Παρατηρούμε ότι δεν προβλέπεται προδιαγραφή για τις στήλες στην εντολή DELETE δηλαδή διαγράφει ολόκληρες εγγραφές .Θα πρέπει λοιπόν να είμαστε πολύ προσεκτικοί όταν τη χρησιμοποιούμε .Για παράδειγμα η ακόλουθη εντολή θα διαγράψει όλες τις εγγραφές του πίνακα:

```
Mysql> delete from prices;
```

Μπορούμε να επιβεβαιώσουμε το γεγονός της διαγραφής εκτελώντας μία εντολή SELECT για να επιλέξουμε δεδομένα από τον πίνακα:

```
Mysql> select * from prices;
```

Το αποτέλεσμα θα είναι empty set γιατί οι εγγραφές όλων των τιμών έχουν διαγραφεί.

Μία εντολή DELETE η οποία εκτελείται υπό συνθήκες όμοια με μία εντολή SELECT ή UPDATE η οποία εκτελείται υπό συνθήκες ,χρησιμοποιεί μία

πρόταση WHERE για την επιλογή συγκεκριμένων εγγραφών .Έχουμε στη διάθεσή μας ολόκληρη την γκάμα των τελεστών σύγκρισης και των λογικών τελεστών για να διατυπώσουμε την συνθήκη βάσει της οποίας θα επιλεχθούν οι εγγραφές που θέλουμε να διαγράψουμε. Για τους χρήστες της MYSQL 4.0(ή μεταγενέστερης έκδοσης) μπορούμε επίσης να χρησιμοποιούμε την πρόταση ORDER BY σε μία εντολή DELETE .Ακολουθεί η σύνταξη της εντολής DELETE με την προσθήκη της πρότασης ORDER BY :

```
DELETE FROM table _name  
[WHERE some _ condition _ is _ true]  
[ORDER BY some _column [ASC | DESC]]  
[LIMIT rows]
```

Με την πρώτη ματιά είναι λογικό να σκεφτούμε ότι 'τι σημασία έχει η ταξινόμηση για τις εγγραφές που πρόκειται να διαγράψουμε ;'.Η πρόταση ORDER BY δεν καθορίζει την σειρά διαγραφής αλλά χρησιμοποιείται για την ταξινόμηση των εγγραφών

## 4. ΔΟΜΙΚΑ ΣΤΟΙΧΕΙΑ ΤΗΣ PHP

Στην ενότητα αυτή θα ασχοληθούμε με τα δομικά στοιχεία της PHP,θα προσπαθήσουμε δηλαδή να καλύψουμε βασικές έννοιες ,αρχές και ειδικές λειτουργίες της PHP.

Πιο συγκεκριμένα ,θα ασχοληθούμε με τα ακόλουθα:

- Ορισμός μεταβλητών ,χρησιμότητα ,διαχείριση
- Προσπέλαση μεταβλητών
- Τύποι δεδομένων
- Κοινοί τελεστές
- Δημιουργία εκφράσεων με χρήση τελεστών
- Ορισμός και χρήση σταθερών

### 4.1.Μεταβλητές

Μία μεταβλητή(variable) είναι μια ειδική θέση την οποία μπορείτε να ορίσουμε για την αποθήκευση μιας τιμής. Οι μεταβλητές είναι ζωτικής σημασίας για τον προγραμματισμό καθώς χωρίς τις μεταβλητές, θα ήμασταν υποχρεωμένοι να ενσωματώσουμε στον κώδικα των script όλες τις τιμές που χρειαζόμαστε.

Οι μεταβλητές μας δίνουν την δυνατότητα να δημιουργούμε 'πρότυπα' για την εκτέλεση διάφορων ενεργειών(π.χ την πρόσθεση ή την αφαίρεση δύο αριθμών) χωρίς να ανησυχούμε για τις τιμές που περιέχουν οι μεταβλητές .Οι τιμές ανατίθενται στις μεταβλητές κατά την εκτέλεση του script – πιθανώς από τον ίδιο το χρήστη ,μέσω ενός ερωτήματος το οποίο εκτελείται σε μια βάση δεδομένων, ή πιθανώς από το αποτέλεσμα κάποιας προηγούμενης ενέργειας του script.Πρέπει να χρησιμοποιούνται μεταβλητές οποτεδήποτε τα δεδομένα που χρησιμοποιούνται για την εκτέλεση μιας λειτουργίας στο script μας υπόκεινται σε αλλαγή από την μία εκτέλεση του script στην επόμενη, ή κατά την διάρκεια ζωής του script.Μία μεταβλητή αποτελείται από ένα όνομα της μας, το οποίο έχει σαν πρόθεμα ένα σύμβολο δολαρίου(\$).Τα ονόματα των μεταβλητών μπορούν γράμματα, αριθμούς και τον χαρακτήρα της κάτω παύλας(\_).Τα ονόματα των μεταβλητών όμως δεν μπορούν να περιλαμβάνουν κενά διαστήματα. Πρέπει πάντα να ξεκινούν με ένα γράμμα, ή με τον χαρακτήρα της κάτω παύλας. Ακολουθούν ορισμένες έγκυρες μεταβλητές:

```
$a;
$a_1ongish_variable_name;
$2452;
$sleepyzzzz;
```

Το ελληνικό ερωτηματικό(;) γνωστό και ως χαρακτήρας τερματισμού εντολών χρησιμοποιείται για τον τερματισμό μιας εντολής της PHP.Οι χαρακτήρες ελληνικού ερωτηματικού στο παραπάνω απόσπασμα κώδικα δεν είναι μέρος των ονομάτων των μεταβλητών, αλλά χρησιμοποιούνται για να σηματοδοτήσουν το σημείο τέλους των εντολών που ορίζουν τις μεταβλητές.

Είναι λοιπόν φανερό ότι έχουμε στη διάθεσή μας άφθονες επιλογές για την ονομασία των μεταβλητών. Για να δηλώσουμε μία μεταβλητή, το μόνο που χρειάζεται είναι να την συμπεριλάβουμε στο script μας. Όταν δηλώνουμε μία μεταβλητή, συνήθως θα αναθέτουμε μία τιμή σ' αυτήν, στην ίδια εντολή, όπως βλέπετε στην συνέχεια:

```
$num1=13;
```

Τύπος	Περιγραφή
Resource	Αναφορά σε έναν πόρο ενός τρίτου κατασκευαστή(π.χ μία βάση δεδομένων)
Null	Μία μη-αρχικοποιημένη μεταβλητή (μία μεταβλητή της οποίας η τιμή είναι απροσδιόριστη)

Οι τύποι πόρων επιστρέφονται συχνά από συναρτήσεις οι οποίες ασχολούνται με εξωτερικές εφαρμογές ή αρχεία .Ο τύπος Null δεσμεύεται για τις μεταβλητές που δεν έχουν αρχικοποιηθεί(δηλαδή, τις μεταβλητές στις οποίες δεν έχετε αναθέσει ακόμη μία τιμή).

## Τελεστές και εκφράσεις

Μία γλώσσα προγραμματισμού δεν μπορεί να είναι χρήσιμη εάν δεν παρέχει κάποιο τρόπο για να χειριζόμαστε τα δεδομένα που αποθηκεύουμε. Οι τελεστές (operators) είναι σύμβολα τα οποία μας δίνουν τη δυνατότητα να χρησιμοποιούμε μία ή περισσότερες τιμές για να παράγουμε μία νέα τιμή. Μία τιμή στην οποία επιδρά ένας τελεστής αναφέρεται με τον όρο τελεστέος (operand).

## Ο Τελεστής Εκχώρησης Τιμής

Ο τελεστής εκχώρησης τιμής αντιπροσωπεύεται από τον χαρακτήρα =.Ο τελεστής εκχώρησης τιμής παίρνει την τιμή του δεξιού τελεστέου του και την εκχωρεί στον αριστερό τελεστέο του:

```
$name= " matt";
```

Η μεταβλητή \$name περιέχει τώρα το αλφαριθμητικό "matt".Το ενδιαφέρον είναι ότι αυτή η δομή αποτελεί μία έκφραση .Με την πρώτη ματιά ίσως νομίζουμε ότι τελεστής εκχώρησης τιμής αλλάζει απλώς την τιμή της μεταβλητής \$name χωρίς να παράγει μία τιμή, αλλά στην πραγματικότητα μία εντολή η οποία χρησιμοποιεί τον τελεστή εκχώρησης αποτιμάται πάντα σε ένα αντίγραφο της τιμής του δεξιού τελεστέου της. Συνεπώς , η γραμμή:

```
echo $name= "matt";
```

Θα εκτυπώσει το αλφαριθμητικό "matt" στο παράθυρο της εφαρμογής browser , επιπρόσθετα με την εκχώρηση του αλφαριθμητικού "matt" στην μεταβλητή \$name.

## Αριθμητικοί Τελεστές

Οι αριθμητικοί τελεστές λειτουργούν με τον γνωστό και αναμενόμενο τρόπο-εκτελούν αριθμητικές πράξεις .Ο παρακάτω πίνακας παρουσιάζει τους αριθμητικούς τελεστές .Ο τελεστής της πρόσθεσης προσθέτει τον δεξιό με τον αριστερό τελεστέο του .Ο τελεστής της αφαίρεσης αφαιρεί τον δεξιό τελεστέο από τον αριστερό .Ο τελεστής της διαίρεσης διαιρεί τον αριστερό τελεστέο με τον δεξιό .Ο τελεστής πολλαπλασιασμού πολλαπλασιάζει τον αριστερό τελεστέο με τον δεξιό .Ο τελεστής ακέрайου υπολοίπου επιστρέφει το ακέрайο υπόλοιπο της διαίρεσης του αριστερού τελεστέου με τον δεξιό.

Τελεστής	Όνομα	Παράδειγμα	Αποτέλεσμα
----------	-------	------------	------------

+	Πρόσθεση	$10+3$	13
-	Αφαίρεση	$10-3$	7
/	Διαίρεση	$10/3$	3.33333333
*	Πολλαπλασιασμός	$10*3$	30
%	Ακέραιο υπόλοιπο	$10\%3$	1



## 5. CSS (Cascading Style Sheets)

Τα CSS styles δίνουν στους σχεδιαστές μεγαλύτερο έλεγχο στο σχεδιασμό ιστοσελίδων ενώ προσφέρουν δυνατότητες για μεγαλύτερη ευελιξία αλλά και μικρότερο μέγεθος αρχείων. Με τα CSS μπορούμε να διαμορφώσουμε όπως θέλουμε το κείμενο, τη μορφή της σελίδας, τους χρωματισμούς και τις στοιχίσεις.

Η σύνταξη των CSS αποτελείται από τρία μέρη : έναν επιλογέα (selector), μια ιδιότητα (property) και μια τιμή (value) :

```
επιλογέας {ιδιότητα: τιμή}
```

```
επιλογέας {ιδιότητα: τιμή}
```

Ο **επιλογέας** είναι συνήθως το στοιχείο που θέλουμε να ορίσουμε, η **ιδιότητα** είναι το χαρακτηριστικό που θέλουμε να αλλάξουμε και η κάθε ιδιότητα μπορεί να πάρει μια **τιμή**. Η ιδιότητα και η τιμή ξεχωρίζουν από τον χαρακτήρα : και περικλείονται από τους χαρακτήρες { }, ως εξής :

```
επιλογέας {ιδιότητα: τιμή}
```

Αν η τιμή αποτελείται από πολλές λέξεις, πρέπει να τοποθετήσουμε εισαγωγικά :

```
p {font-family: "serif serif"}
```

Αν θέλουμε να ορίσουμε περισσότερες από μία ιδιότητες, πρέπει να ξεχωρίσουμε την κάθε ιδιότητα με τον χαρακτήρα ;. Το παρακάτω παράδειγμα δείχνει πώς μπορούμε να ορίσουμε μια κεντραρισμένη παράγραφο με χρώμα κειμένου κόκκινο :

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

### 5.1. Το Χαρακτηριστικό (Attribute) Class

Με το χαρακτηριστικό class μπορούμε να ορίσουμε διαφορετικά στυλ για το ίδιο στοιχείο (element). Ας υποθέσουμε ότι θέλουμε να έχουμε δύο είδη παραγράφων στο έγγραφό μας : μια δεξιά στοιχισμένη παράγραφο και μια κεντραρισμένη παράγραφο.

Να πώς μπορούμε να το κάνουμε αυτό με τα στυλ :

```
p.δεξιά {text-align: right}
```

```
p.κεντρία {text-align: center}
```

### 5.2. Το Κείμενο του CSS

Οι ιδιότητες κειμένου (text properties) μάς δίνουν τη δυνατότητα να ελέγξουμε την εμφάνιση του κειμένου. Είναι δυνατό να αλλάξουμε το χρώμα ενός κειμένου, να αυξήσουμε ή να ελαττώσουμε το διάστημα ανάμεσα στους χαρακτήρες ενός κειμένου, να ευθυγραμμίσουμε ένα κείμενο, να διακοσμήσουμε ένα κείμενο, να δημιουργήσουμε εσοχή στην πρώτη γραμμή ενός κειμένου κ.ά.

Property	Description	Value
color	Ορίζει το χρώμα του κειμένου	color
direction	Ορίζει την κατεύθυνση του κειμένου	Ltr, rtl
letter-spacing	Αυξάνει ή ελαττώνει το διάστημα ανάμεσα στους χαρακτήρες	Normal, length
text-align	Ευθυγραμμίζει το κείμενο σ' ένα στοιχείο	Left, right, center, justify
text-decoration	Προσθέτει διακόσμηση στο κείμενο	None, underline, overline, line-through, blink
text-indent	Δημιουργεί εσοχή στην πρώτη γραμμή του κειμένου ενός στοιχείου	Length, %
text-shadow		None, color, length
text-transform	Ελέγχει τα γράμματα σ' ένα στοιχείο	None, capitalize, uppercase, lowercase
unicode-bidi		Normal, embed, bidi-override
white-space	Ορίζει το πώς θα αντιμετωπίζεται το λευκό κενό (white space) μέσα σ' ένα στοιχείο	Normal, pre, nowrap
word-spacing	Αυξάνει ή ελαττώνει το διάστημα ανάμεσα στις λέξεις	Normal, length

### 5.3. Οι Γραμματοσειρές του CSS

Οι ιδιότητες γραμματοσειρών μάς δίνουν τη δυνατότητα να αλλάξουμε την οικογένεια της γραμματοσειράς (font family), το έντονο (boldness), το μέγεθος (size) και το στυλ (style) ενός κειμένου.

Property	Description	Value
Font	Ορίζει όλες τις ιδιότητες μιας γραμματοσειράς σε μια δήλωση	font-style, font-variant, font-weight, font-size, line-height, font-family, caption, icon, menu, message-box, small-caption, status-bar

font-family	Μια λίστα προτεραιότητας οικογενειών γραμματοσειράς και/ή generic ονομάτων οικογένειας για ένα στοιχείο	family-name, generic-family
font-size	Ορίζει το μέγεθος μιας γραμματοσειράς	xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, length, %
font-size-adjust	Καθορίζει μια τιμή άποψης (aspect value) για ένα στοιχείο που θα διατηρήσει το x-height της πρώτης επιλεγμένης γραμματοσειράς	None, number
font-stretch	Συμπυκνώνει ή επεκτείνει την τρέχουσα οικογένεια γραμματοσειράς	Normal, wider, narrower, ultra-condensed, extra-condensed, condensed, semi-condensed, semi-expanded, expanded, extra-expanded, ultra-expanded
font-style	Ορίζει το στυλ της γραμματοσειράς	Normal, italic, oblique
font-variant	Εμφανίζει το κείμενο με μικρά κεφαλαία (small-caps) ή κανονικά (normal)	Normal, small-caps
font-weight	Ορίζει το βάρος (weight) μιας γραμματοσειράς	Normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900

#### 5.4. Τα Περιγράμματα του CSS

Οι ιδιότητες περιγραμμάτων μάς δίνουν τη δυνατότητα να καθορίσουμε το στυλ, το χρώμα και το πλάτος του περιγράμματος ενός στοιχείου. Στην HTML χρησιμοποιούμε πίνακες για να δημιουργούμε περιγράμματα γύρω από ένα κείμενο, αλλά με τις ιδιότητες περιγραμμάτων μπορούμε να δημιουργήσουμε περιγράμματα με ωραία εφέ και τα οποία μπορούν να εφαρμοσθούν σ' ένα οποιοδήποτε στοιχείο.

Ιδιότητα	Περιγραφή	Τύπος
border	Ορίζει όλες τις ιδιότητες για τα τέσσερα περιγράμματα σε μια δήλωση	border-width, border-style, border-color
border-bottom	Ορίζει όλες τις ιδιότητες του κάτω περιγράμματος σε μια δήλωση	border-bottom-width, border-style, border-color
border-bottom-color	Ορίζει το χρώμα του κάτω περιγράμματος	border-color
border-bottom-style	Ορίζει το στυλ του κάτω περιγράμματος	border-style
border-bottom-width	Ορίζει το πλάτος του κάτω περιγράμματος	Thin, medium, thick, length
border-color	Ορίζει το χρώμα των τεσσάρων περιγραμμάτων και μπορεί να έχει	color

	τιμή από ένα έως τέσσερα χρώματα	
border-left	Ορίζει όλες τις ιδιότητες του αριστερού περιγράμματος σε μια δήλωση	border-left-width, border-style, border-color
border-left-color	Ορίζει το χρώμα του αριστερού περιγράμματος	border-color
border-left-style	Ορίζει το στυλ του αριστερού περιγράμματος	border-style
border-left-width	Ορίζει το πλάτος του αριστερού περιγράμματος	Thin, medium, thick, length
border-right	Ορίζει όλες τις ιδιότητες του δεξιού περιγράμματος σε μια δήλωση	border-right-width, border-style, border-color
border-right-color	Ορίζει το χρώμα του δεξιού περιγράμματος	border-color
border-right-style	Ορίζει το στυλ του δεξιού περιγράμματος	border-style
border-right-width	Ορίζει το πλάτος του δεξιού περιγράμματος	Thin, medium, thick, length
border-style	Ορίζει το στυλ των τεσσάρων περιγραμμάτων και μπορεί να έχει τιμή από ένα έως τέσσερα στυλ	None, hidden, dotted, dashed, solid, double, groove, ridge, inset, outset,
border-top	Ορίζει όλες τις ιδιότητες του πάνω περιγράμματος σε μια δήλωση	border-top-width, border-style, border-color
border-top-color	Ορίζει το χρώμα του πάνω περιγράμματος	border-color
border-top-style	Ορίζει το στυλ του πάνω περιγράμματος	border-style
border-top-width	Ορίζει το πλάτος του πάνω περιγράμματος	Thin, medium, thick, length
border-width	Ορίζει το πλάτος των τεσσάρων περιγραμμάτων σε μια δήλωση και μπορεί να έχει από μία έως τέσσερις τιμές	Thin, medium, thick, length

### 5.5. Τα Περιθώρια του CSS

Οι ιδιότητες περιθωρίου ορίζουν το διάστημα γύρω από τα στοιχεία. Μπορούμε να χρησιμοποιήσουμε αρνητικές τιμές για να επικαλύψουμε (overlap) το περιεχόμενο.

Το πάνω, δεξιά, κάτω και αριστερό περιθώριο μπορούν να αλλάξουν ανεξάρτητα χρησιμοποιώντας ξεχωριστές ιδιότητες. Μπορούμε να χρησιμοποιήσουμε την ιδιότητα margin για να αλλάξουμε όλα τα περιθώρια μονομιάς.

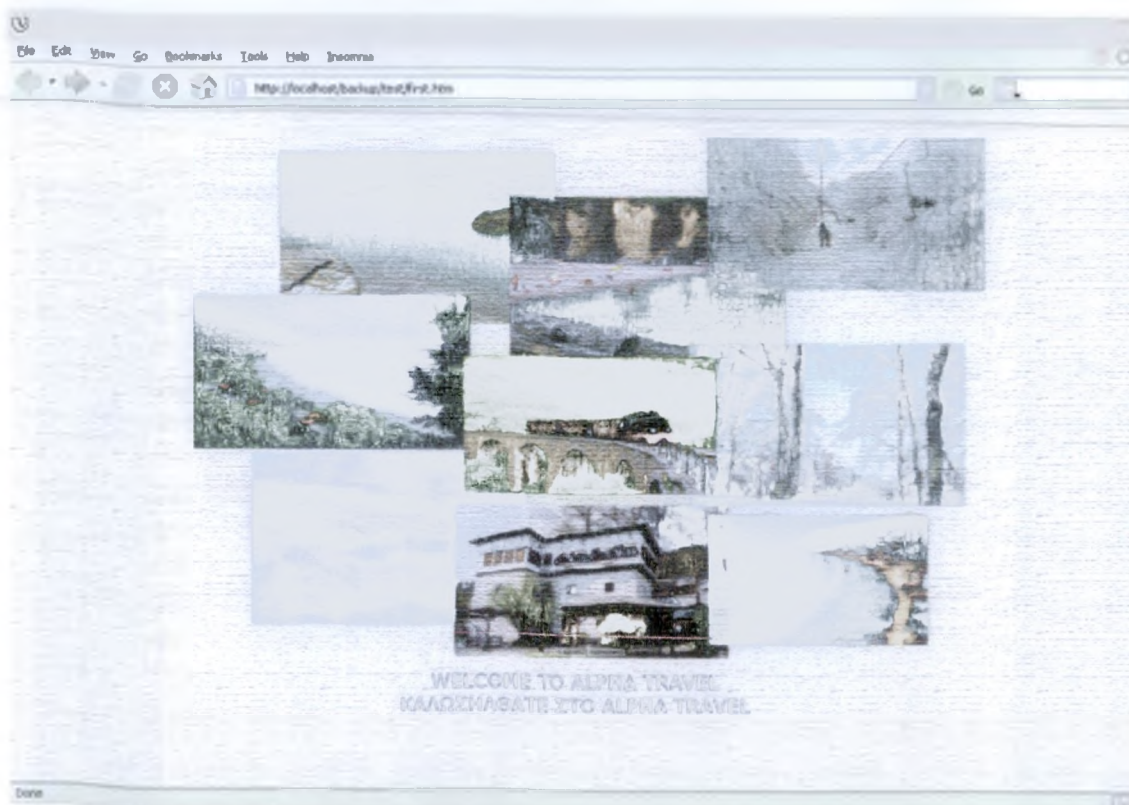
margin	Ορίζει τις ιδιότητες περιθωρίου σε μια δήλωση	margin-top, margin-right, margin-bottom, margin-left
margin-	Ορίζει το κάτω περιθώριο ενός	Auto, length, %

bottom	στοιχείου	
margin-left	Ορίζει το αριστερό περιθώριο ενός στοιχείου	Auto, length, %
margin-right	Ορίζει το δεξιό περιθώριο ενός στοιχείου	Auto, length, %
margin-top	Ορίζει το πάνω περιθώριο ενός στοιχείου	Auto, length, %

## 6. ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ

Κύριος σκοπός της σελίδας μας είναι να είναι όσο το δυνατόν πιο λιτή. Ο χρήστης να μπορεί να βρίσκει αμέσως αυτό που ψάχνει χωρίς να χρειάζεται να περάσει από όλους τους συνδέσμους. Το περιβάλλον να είναι φιλικό ως προς αυτόν για να κάνει πιο ευχάριστο το διάστημα που θα χρειαστεί να βρει τις πληροφορίες που θέλει. Η σελίδα προτείνεται να φορτώνεται σε οθόνη με ανάλυση 1024 x 768 αλλά και 800 x 600.

Μπαίνοντας στο site μας ο επισκέπτης θα κληθεί να διαλέξει σε ποια γλώσσα να δει τις πληροφορίες, ανάμεσα σε Ελληνικά και Αγγλικά.



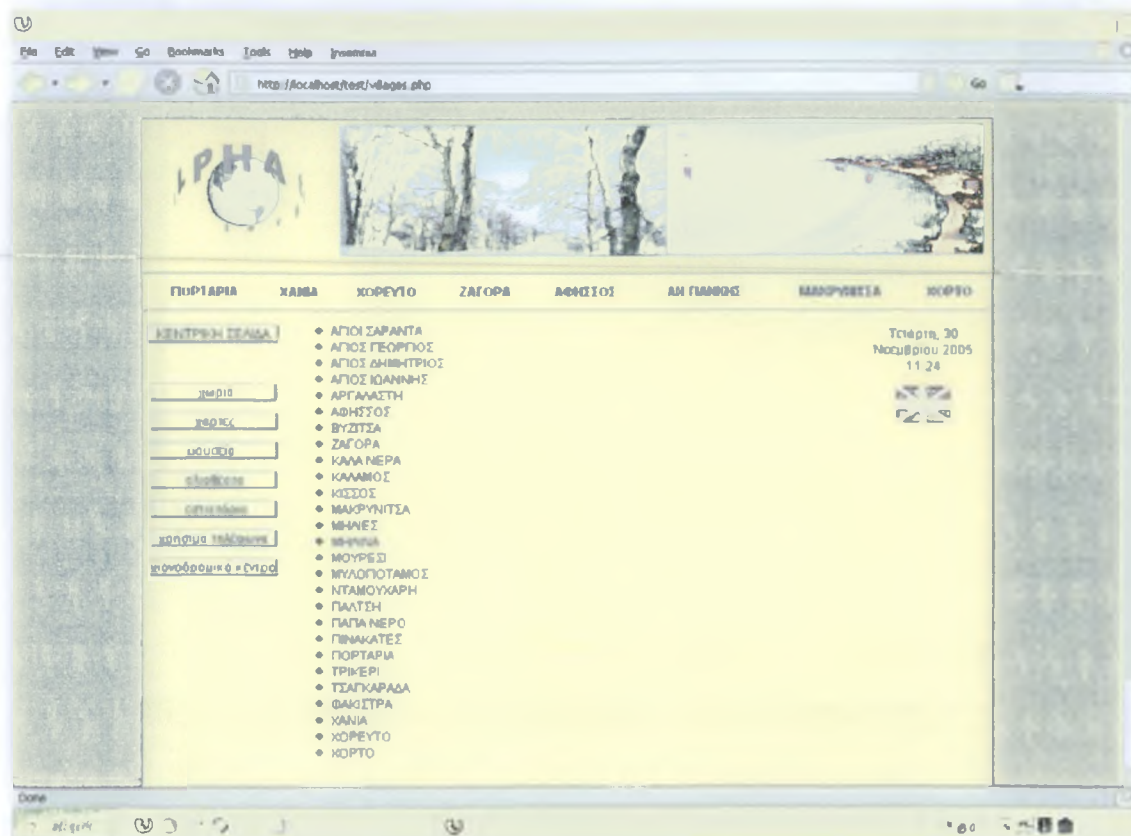
Στην κεντρική σελίδα ο χρήστης θα βρει κάποιες γενικές πληροφορίες για το Πήλιο καθώς και κάποια ιστορικά στοιχεία. από το αριστερό μενού θα μπορεί να περιηγηθεί στο κομμάτι του site που θα τον ενδιαφέρει περισσότερο. Οι επιλογές που έχει είναι οι εξής

- Χωριά
- Χάρτες
- Μουσεία
- Αξιοθέατα
- Εστιατόρια
- Χρήσιμα τηλέφωνα
- Χιονοδρομικό κέντρο



## ΧΩΡΙΑ

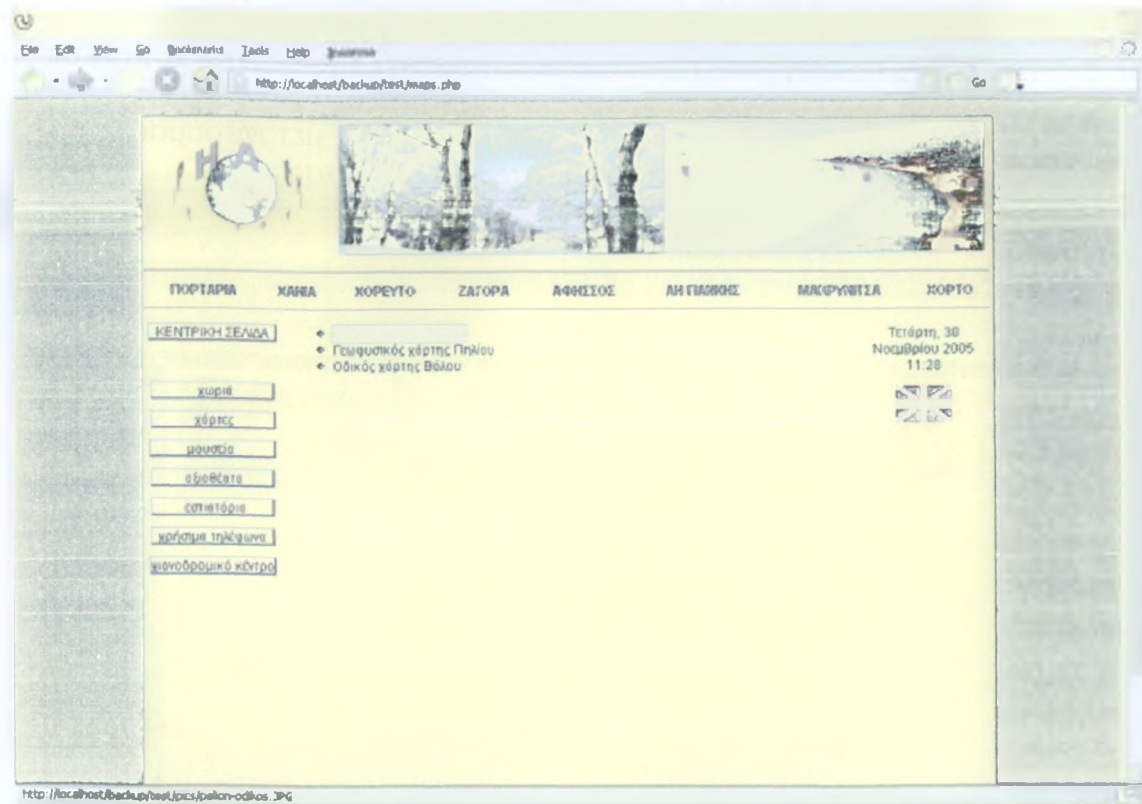
από αυτόν τον σύνδεσμο ο επισκέπτης έχει τη δυνατότητα να δει όλα τα χωριά του Πηλίου (σε αυτά που υπάρχουν ξενοδοχεία δηλαδή) και να ψάχνει τα ξενοδοχεία ανάλογα με την τοποθεσία. Αυτός ο τομέας της σελίδας είναι δυναμικός. Από τις σελίδες διαχείρισης (βλ. Administration Pages) έχει τη δυνατότητα ο διαχειριστής του συστήματος να εισάγει καινούργιες τοποθεσίες αλλά και να διαγράψει κάποιες από αυτές. Οποιαδήποτε αλλαγή θα επηρεάσει αυτή τη σελίδα ανάλογα. Πατώντας την αγγλική σημαία μεταφερόμαστε στην ίδια ακριβώς περιοχή της σελίδας μας στην αγγλική γλώσσα.





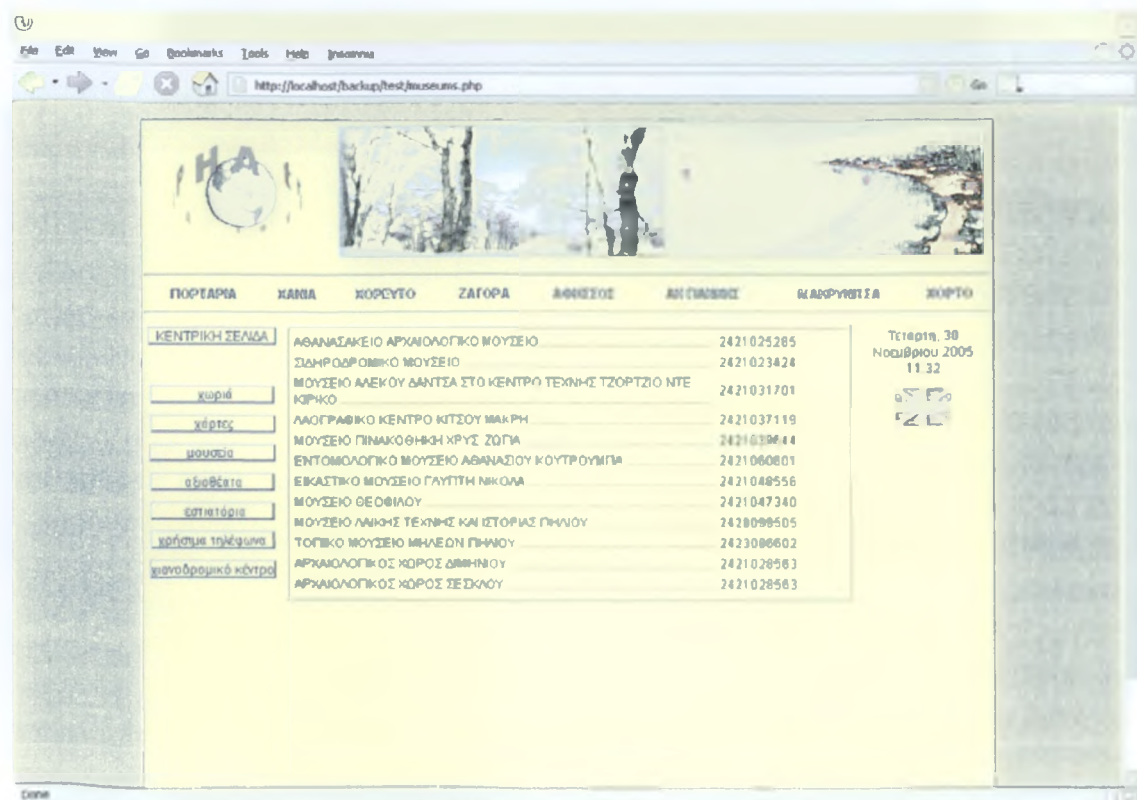
## ΧΑΡΤΕΣ

Εδώ ο χρήστης μπορεί να δει χρήσιμους χάρτες του Πηλίου και του Βόλου



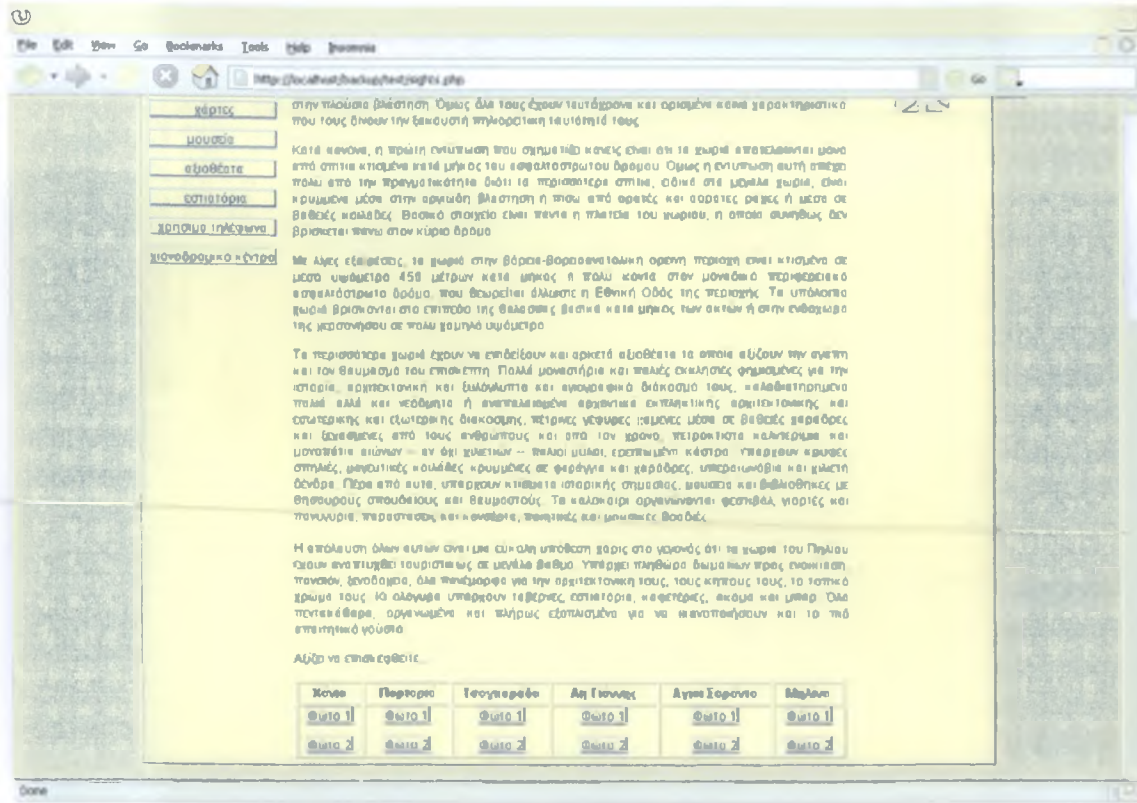
## ΜΟΥΣΕΙΑ

Τα κυριότερα μουσεία του Βόλου και του Πηλίου.



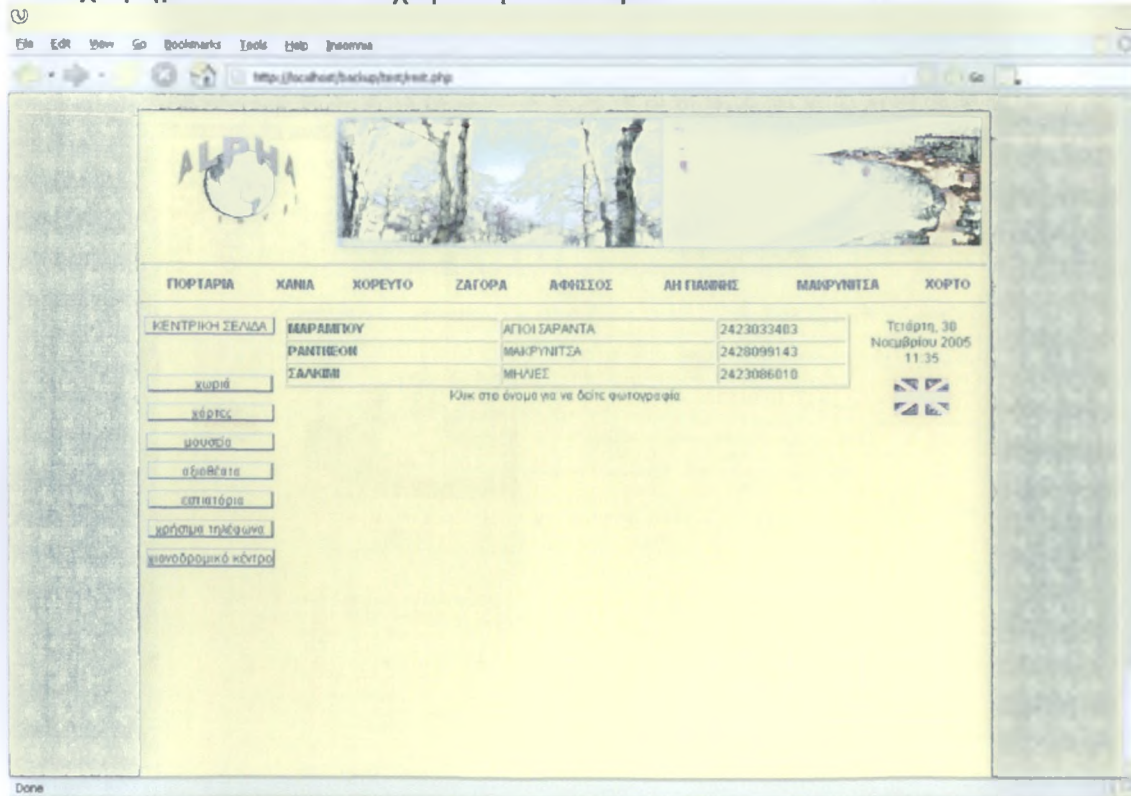
## ΑΞΙΟΘΕΑΤΑ

Εδώ προτείνουμε εμείς κάποιες τοποθεσίες του Πηλίου που πιστεύουμε ότι αξίζουν να τις επισκεφθεί κάποιος και παραθέτουμε ορισμένες φωτογραφίες.



## ΕΣΤΙΑΤΟΡΙΑ

Καταχωρημένα από το διαχειριστή εστιατόρια



## ΧΡΗΣΙΜΑ ΤΗΛΕΦΩΝΑ

Σε περίπτωση που ο επισκέπτης αποφασίσει να επισκεφθεί το Πήλιο σίγουρα θα πρέπει να έχει υπ' όψιν του αυτά τα τηλέφωνα

The screenshot shows a web browser window with the URL <http://localhost/backup/test/phones.php>. The page features a navigation menu with the following items: ΠΟΡΤΑΡΙΑ, ΧΑΜΑ, ΚΟΡΕΥΤΟ, ΖΑΓΟΡΑ, ΛΟΦΙΣΣΟΣ, ΑΝΤΙΤΑΒΕΣ, ΜΑΛΦΟΥΣΙΣΤΑ, and ΧΟΡΤΟ. Below the menu is a table titled "ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ" containing a list of phone numbers for various locations. On the right side of the page, there is a date and time stamp: "Τετάρτη, 30 Νοεμβρίου 2005 11:37".

ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ	ΤΗΛΕΦΩΝΟ
Αεροδρόμιο Αργείου	(24200) 76006
"Περίμετρο Δόρυχοι" Βερασίου	(24210) 39700 / 39707
Λυκαονικό Βόλου	20908
Σταθμός Λαμναροίων ΚΤΕΛ Βόλου	33253 / 25527
Σταθμός Λαμναροίων ΚΤΕΛ Αθήνας	(210) 8317186
Σταθμός Λαμναροίων ΚΤΕΛ Θεσσαλονίκης	(2310) 543087
Σταθμός Λαμναροίων ΚΤΕΛ Πάτρας	(2610) 222164
Οργανισμός Σιδηροδρόμων Ελλάδος (ΟΣΕ)	29555 / 24056
Σιδηροδρομικός σταθμός ΟΣΕ Αθήνας	(210) 524 0648
Σιδηροδρομικός σταθμός ΟΣΕ Θεσσαλονίκης	(2310) 517517
Ελληνικός Οργανισμός Τουρισμού (ΕΟΤ)	24915 / 36233
Νοσοκομείο Μάκρηντας	70951 / 70932
Δημοτικό Βόλου	21111
Τουριστική Αστυνομία	72420 / 72421
Γενικό Νοσοκομείο Βόλου	27531 / 39225 / 20012

## ΧΙΟΝΟΔΡΟΜΙΚΟ ΚΕΝΤΡΟ

Για τους λάτρεις του χιονιού πληροφορίες για το χιονοδρομικό κέντρο Πηλίου

The screenshot shows a web browser window with the URL <http://localhost/backup/test/chiwon.php>. The page features a navigation menu with the following items: ΠΟΡΤΑΡΙΑ, ΧΑΜΑ, ΚΟΡΕΥΤΟ, ΖΑΓΟΡΑ, ΛΟΦΙΣΣΟΣ, ΑΝΤΙΤΑΒΕΣ, ΜΑΛΦΟΥΣΙΣΤΑ, and ΧΟΡΤΟ. Below the menu is a section titled "ΚΕΝΤΡΙΚΗ ΣΕΛΙΔΑ" containing text about the ski center. On the right side of the page, there is a date and time stamp: "Τετάρτη, 30 Νοεμβρίου 2005 11:38".

Το χιονοδρομικό κέντρο είναι από τον Βόλο 27 χμ και μόλις 12 χμ από την Πορταριά κόντος τη θάλασσα για την διεύθυνση σας κατά την διάρκεια της χειμερινής σας επίσκεψης.

Από το χιονοδρομικό κέντρο Αφιδεμικής μπορείτε να απολαύσετε την μοναδική θέα που προφέρεται, προς τον Πενταστημό κόλπο και το Αγιο Πέτρο με το χιονισμένο τοπίο των βουνών του Πηλίου.

Το χιονοδρομικό κέντρο Αφιδεμικής είναι πλήρως εξοπλισμένο και διαθέτει τέσσερα σκαμνίσια. Διαθέτει ποικίλες πίστες για όλα, συνολικού μήκους 5000 μ. Επίσης για τους αρχάριους υπάρχουν ειδική πιστά αρχαρίων και εκπαιδευτές χι, όπου μπορούν να σας οδηγήσουν στις πρώτες σας προσπάθειες.

Παράλληλα με το ski στις πίστες, μπορείτε να εσχαλήσετε και με άλλα χιονοδρομικά υάρπη όπως snowboarding και ορειβασία.

Το πρώτο χιονιά κέντρο όπου βρίσκεται κοντά στο χιονοδρομικό κέντρο, είναι ιδιαίτερα δημοφές και εκτός της χειμερινής σεζόν. Καθ' όλη την διάρκεια της σεζόν και του καλοκαιριού, πραγματοποιούνται αγώνες μηχανών ενδίο οθεδρομής 4x4 καθώς και πεζοπορίες εξερεύνησης αυτής της τόσο εντυπωσιακής περιοχής.

Για περισσότερες πληροφορίες παρακαλώ επικοινωνήστε μαζί μας στα τηλέφωνα 24210 92600, 2428073719. Ήμε περίπτωση έκτακτης ανάγκης στο 6948504307.

## 7. ADMINISTRATION PAGES

### 7.1. Τι είναι

Εκτός από τις σελίδες που προσφέρονται για το κοινό, το site μας δίνει κάποια παραπάνω δικαιώματα σε χρήστες οι οποίοι είναι αρμόδιοι για τη σωστή λειτουργία του. Συγκεκριμένα οι διαχειριστές θα μπορούν να

- Εισάγουν / Ενημερώνουν / Διαγράφουν ξενοδοχεία από τη βάση
- Εισάγουν / Ενημερώνουν / Διαγράφουν εστιατόρια από τη βάση

Φυσικά δε θα θέλαμε σε καμία περίπτωση οποιοσδήποτε να μπορούσε να αλλάξει αυτές τις πληροφορίες. Αυτό το καταφέρνουμε δίνοντας πρόσβαση μόνο σε όσους έχουν κάποιο username και το αντίστοιχο password.

A screenshot of a web form for user authentication. It features two input fields: the first is labeled 'username' and the second is labeled 'password'. Below these fields is a button labeled 'Submit'.

### 7.2. Πώς γίνεται

Η όλη διαδικασία αποτελείται από 2 αρχεία. Το check.php και το login.php. Το πρώτο είναι ο έλεγχος που γίνεται σε κάθε σελίδα που θέλουμε να προστατέψουμε κι αν τελικά ο χρήστης δεν είναι συνδεδεμένος τότε εκτελείται το login.php.

Σε κάθε σελίδα που θέλουμε να προστατέψουμε θα κάνουμε include το check.php στην αντίστοιχη σελίδα.

Για να το πετύχουμε αυτό χρησιμοποιήσαμε sessions. Τα sessions είναι ένας τρόπος για να κρατάμε δεδομένα σε διαφορετικές σελίδες. Κάθε φορά που ανοίγουμε τον browser μας το session παίρνει μια μοναδική τιμή (session\_id). Μετά χρησιμοποιώντας αυτή την τιμή μπορούμε να ανακτήσουμε διάφορες μεταβλητές που έχουμε ορίσει προηγουμένως. Η διαδικασία αυτή ξεκινάει με την εντολή session\_start() και τερματίζει με την εντολή session\_unset() ή όταν κλείσουμε τον browser.

#### Check.php

```

1. <?php
2. session_start();
3. if (isset($_SESSION['username']))
4. {
5.     header("Location: * . "login.php" );

```

- 7. }
- 8. ?>

Αφού ξεκινήσουμε το session περνάμε στη μεταβλητή url το όνομα της σελίδας από την οποία γίνεται include το αρχείο. Στην 4<sup>η</sup> γραμμή γίνεται ο έλεγχος για το κατά πόσο έχει οριστεί μέσα στο session η μεταβλητή user (η οποία ορίζεται στο αρχείο login.php). Αν έχει οριστεί τότε συνεχίζει κανονικά και εκτελείται ο κώδικας της σελίδας. Αν όχι (πράγμα που σημαίνει ότι δεν έχει συνδεθεί ο χρήστης) τότε του λέμε να τρέξει το login.php.

### Login.php

```

1. <?php require_once('../Connections/conne.php'); ?>
2. <?php
3. mysql_select_db($database_conne, $conne);
4. $query_Recordset1 = "SELECT * FROM users";
5. $Recordset1 = mysql_query($query_Recordset1, $conne) or
   die(mysql_error());
6. $row_Recordset1 = mysql_fetch_assoc($Recordset1);
7. $totalRows_Recordset1 = mysql_num_rows($Recordset1);
8. ?>
9. <?php
10. // *** validate request to login to this site.
11. if (!isset($_SESSION)) {
12.     session_start();
13. }
14. //εκsetazoume an o xristis erxete apo kapoia alli selida
   i kateuthelian apo ti login.php
15. if (!isset($_SESSION['url']))
16. {
17.     $LoginSuccess = "admin.php";
18. }
19. else
20. {
21.     $LoginSuccess = $_SESSION['url'];
22. }
23. //εksetazoume an exoume idi syndethei
24. if (isset($_POST['user'])) {
25.     $_SESSION['user']=$_POST['user'];
26.     $loginUsername=$_POST['user'];
27.     $password=$_POST['pass'];
28.     $loginFailed = "fail.php";
29.     mysql_select_db($database_conne, $conne);
30.     $loginRS__query=sprintf("SELECT username, password FROM
       users WHERE username='%s' AND password='%s'",
31.     get_magic_quotes_gpc() ? $loginUsername:
       addslashes($loginUsername), get_magic_quotes_gpc() ?
       $password : addslashes($password));
32.     $loginRS = mysql_query($loginRS__query, $conne) or
       die(mysql_error());
33.     $loginFoundUser = mysql_num_rows($loginRS);
34.     if ($loginFoundUser) (//synthiki gia to an brethike
       sostos syndyasmos user,pass
35.     header ("Location: # . $LoginSuccess );
36.     }
37.     else {

```

```

38. header ("Location: ". $loginFailed );
39. }
40. }
41. ?>

42. <style type="text/css">
43. <!--
44. @import url("../hotels.css");
45. -->
46. </style>
47. <body>
48. <form name="form1" method="POST" action="">
49. <p align="center">username
50. <input name="user" type="text" id="user">
51. </p>
52. <p align="center">password
53. <input name="pass" type="password" id="pass">
54. </p>
55. <p align="center">
56. <input type="submit" name="Submit" value="Submit">
57. </p>
58. </form>
59. </body>
60. <?php
61. mysql_free_result($Recordset1);
62. ?>

```

**(11-12)** εξετάζουμε το κατά πόσο έχει ξεκινήσει κάποιο session. Αν δεν έχει ξεκινήσει τότε το ξεκινάμε εμείς

**(15-22)** εξετάζουμε αν υπάρχει ήδη η μεταβλητή url. Αν υπάρχει σημαίνει ότι ήρθαμε από κάποια άλλη σελίδα (και όχι από τη (login.php). Αν δεν έχουμε έρθει από κάποια άλλη σελίδα (άρα ήρθαμε από τη (login.php) τότε η *\$LoginSuccess* θα πάρει την τιμή admin.php που είναι και η κεντρική σελίδα των διαχειριστών., αλλιώς αν ήρθαμε από άλλη σελίδα, μετά τη σύνδεσή μας θα μεταφερθούμε στην αντίστοιχη σελίδα

**(24-28)** εξετάζουμε αν ο χρήστης είναι ήδη συνδεδεμένο. Αν δεν είναι ορίζουμε τις μεταβλητές loginUsername και password με τα αντίστοιχα username και password που έχει εισάγει ο χρήστης. Η loginFailed είναι μια σελίδα που θα μας εμφανίζεται σε περίπτωση που η εισαγωγή username/ password δεν είναι έγκυρη.

**(30-41)** γίνεται ο έλεγχος αν ο συνδυασμός username/password είναι σωστός και ανάλογα εκτελεί κώδικα.

**(42-60)** είναι απλός HTML κώδικας για τις φόρμες.

### 7.3. Πίνακας ελέγχου

ΞΕΝΟΔΟΧΕΙΑ	ΕΣΤΙΑΤΟΡΙΑ
<ul style="list-style-type: none"> <li>• Εισαγωγή</li> <li>• Ενημέρωση/Διαγραφή</li> </ul>	<ul style="list-style-type: none"> <li>• Εισαγωγή</li> <li>• Ενημέρωση/Διαγραφή</li> </ul>

από αυτό το menu ο διαχειριστής θα μπορεί να διαλέξει τις πληροφορίες που θέλει να αλλάξει.

#### 7.3.1. Ξενοδοχεία. Εισαγωγή

ΟΝΟΜΑ ΞΕΝΟΔΟΧΕΙΟΥ

ΤΥΠΟΣ

ΙΔΙΟΚΤΗΤΗΣ

ΤΗΛΕΦΩΝΟ

ΤΟΠΟΘΕΣΙΑ

ΚΑΤΗΓΟΡΙΑ

ΦΩΤΟΓΡΑΦΙΑ

ΙΣΤΟΣΕΛΙΔΑ

μέσα από μια φόρμα θα εισάγει τις πληροφορίες που θέλει. Τον τύπο και την κατηγορία θα τις διαλέγει μέσα από ένα μενού με στατικά στοιχεία ενώ την τοποθεσία θα την διαλέγει από μενού το οποίο θα διαβάζει τον πίνακα places.

## 7.3.2. Ξενοδοχεία. Ενημέρωση / Διαγραφή

Μέσα από αυτή τη σελίδα ο διαχειριστής έχει τη δυνατότητα να βλέπει όλα τα ξενοδοχεία με αλφαβητική σειρά και αναλόγως να διαλέγει αν θέλει να το ανανεώσει ή να το διαγράψει

ΚΑΤΗΓΟΡΙΑ	ΤΥΠΟΣ	ΙΔΙΟΚΤΗΤΗΣ	Τ.Φ.Π.Α.Σ.Α.	ΤΟΠΟΘΕΣΙΑ	ΚΑΤΗΓΟΡΙΑ	ΤΕΛΕΥΤΑΙΑ ΕΠΗΜΕΡΩΣΗ	ΕΠΙΣΤΑΣΗ	LOGOUT
ΑΙΓΥΪ	ΕΠΙΧΑΡΑΚΤΟ ΞΕΝΟΔΟΧΕΙΟ	ΕΤΣΙΟΠΟΥΛΗ	2420022770	ΑΓΙΟΙ ΣΑΡΑΝΤΑ	A	03/12/2005		end
ΑΙΘΥΣ	ΕΠΙΧΑΡΑΚΤΟ ΞΕΝΟΔΟΧΕΙΟ	ΣΟΦΑΝΤΖΗΣ	2420022910	ΑΡΧΗΣΣΟΣ	A	01/10/2005		
ΑΙΘΥΣ	ΕΠΙΧΑΡΑΚΤΟ ΞΕΝΟΔΟΧΕΙΟ	ΗΛΙΑΝΤΖΑΦΕΡΗΣ	2420023100	ΑΓΙΟΣ ΙΩΑΝΝΗΣ	A	20/08/2005		
ΑΙΘΥΣ	ΞΕΝΟΔΟΧΕΙΟ	ΕΙΤΖΑΒΕΡΑ	2420022904	ΑΓΙΟΙ ΣΑΡΑΝΤΑ	A	03/12/2005		end
ΑΙΘΥΣ	ΕΠΙΧΑΡΑΚΤΟ ΞΕΝΟΔΟΧΕΙΟ	ΠΑΝΑΪΗΣΟΥ	2420031241	ΑΓΙΟΣ ΙΩΑΝΝΗΣ	A	25/05/2005		
ΑΙΘΥΣ		ΜΗΡΟΠΟΥΛΟΣ	2420031223	ΑΓΙΟΣ ΙΩΑΝΝΗΣ	B	0000-00-00		
ΑΙΘΥΣ		ΕΛΤΟΙΩΤΗ	2420077573	ΑΓΙΟΙ ΠΕΤΡΟΣ	B	0000-00-00		
ΑΙΘΥΣ		ΑΝΤΩΝΑΡΑΤΗΣ	2422092003	ΑΡΧΗΣΣΟΣ	B	0000-00-00		
ΑΙΘΥΣ	ΞΕΝΟΔΟΧΕΙΟ	ΠΑΠΑΝΥΦΩΝΟΥ	2423022271	ΚΟΝΙΑΝΕΡΑ	A	0000-00-00		
ΑΙΘΥΣ		ΒΑΣΙΛΕΥΚΟΝΗ	2420031422	ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ	B	0000-00-00		
ΑΙΘΥΣ		ΑΝΤΩΝΗΣ	2420031291	ΑΡΧΗΣΣΟΣ	B	0000-00-00		
ΑΙΘΥΣ		Γ. ΓΑΛΟΥ	2420033100	ΑΡΧΗΣΣΟΣ	B	0000-00-00		
ΑΙΘΥΣ		Π.ΚΥΡΙΑ	2420000008	ΑΓΙΑ	B	0000-00-00		
ΑΙΘΥΣ	ΞΕΝΟΔΟΧΕΙΟ	ΑΔΑΜΑΝΤΩΣΤΟΥ	2420022210	ΚΟΝΙΑΝΕΡΑ	A	0000-00-00		
ΑΙΘΥΣ	ΕΠΙΧΑΡΑΚΤΟ ΞΕΝΟΔΟΧΕΙΟ	ΜΠΑΡΟΣ	2420000100	ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ ΓΡΗΓΟΡΗΣ	A	0000-00-00		
ΑΙΘΥΣ	ΞΕΝΟΔΟΧΕΙΟ	ΧΛΑΣΙΟΣ	2420022951	ΚΟΡΕΥΤΟ	B	0000-00-00		
ΑΙΘΥΣ		ΓΕΡΥΧΩΝΗΣ	2420031391	ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ	B	0000-00-00		



ΕΝΗΜΕΡΩΣΗ

ΟΝΟΜΑ ΞΕΝΟΔΟΧΕΙΟΥ	ΑΙΟΛΟΣ
ΤΥΠΟΣ	ΕΠΙΠΛΩΜΕΝΟ ΔΙΑΜΕΡΙΣΜΑ
ΙΔΙΟΚΤΗΤΗΣ	ΣΟΦ.ΚΙΤΣΙΟΣ
ΥΠΟΘΕΚΑ	2426022910
ΤΥΠΟΣ ΒΕΔΩ	ΑΦΗΣΣΟΣ ΑΠΟΙ ΣΑΡΑΝΤΑ
ΚΑΤΗΓΟΡΙΑ	A
ΦΩΤΟΓΡΑΦΙΑ	
ΙΣΤΟΡΙΑ	
ΜΟΝΟΚΑΜΙΟ	80
ΔΙΑΜΕΤΡΟ	110
ΥΨΟΣ	150
ΣΩΜΤΑ	220
03/12/2005 and Update	Επεξεργασία

μπορεί να εισάγει τιμές για το ξενοδοχείο αλλά και να αλλάξει οποιοδήποτε άλλο στοιχείο θέλει. Για λόγους ασφαλείας υπάρχει ένα πεδίο με την ημερομηνία και το username του διαχειριστή τα οποία προστίθενται αυτόματα χωρίς να υπάρχει δυνατότητα επεξεργασίας. Αυτομάτως με το που πατήσουμε update η ημερομηνία και το username καταχωρούνται στον πίνακα με όλα τα ξενοδοχεία για να γνωρίζουμε πότε έγινε η τελευταία επεξεργασία και από ποιόν.

ΔΙΑΓΡΑΦΗ

## ΔΙΑΓΡΑΦΗ ΞΕΝΟΔΟΧΕΙΟΥ

ΟΝΟΜΑ	ΑΓΝΑΝΤΙ
ΤΥΠΟΣ	ΕΠΙΠΛΩΜΕΝΟ ΔΙΑΜΕΡΙΣΜΑ
ΙΔΙΟΚΤΗΤΗΣ	Τ.ΤΡΙΑΝΤΑΦΥΛΛΟΥ
ΤΗΛΕΦΩΝΟ	2126023114
ΤΟΠΟΘΕΣΙΑ	ΑΓΙΟΣ ΙΩΑΝΝΗΣ
ΚΑΤΗΓΟΡΙΑ	A

Διαγραφή

ένα παράθυρο επιβεβαίωσης εμφανίζεται και όταν πατήσουμε διαγραφή, διαγράφεται το ξενοδοχείο από τη βάση.

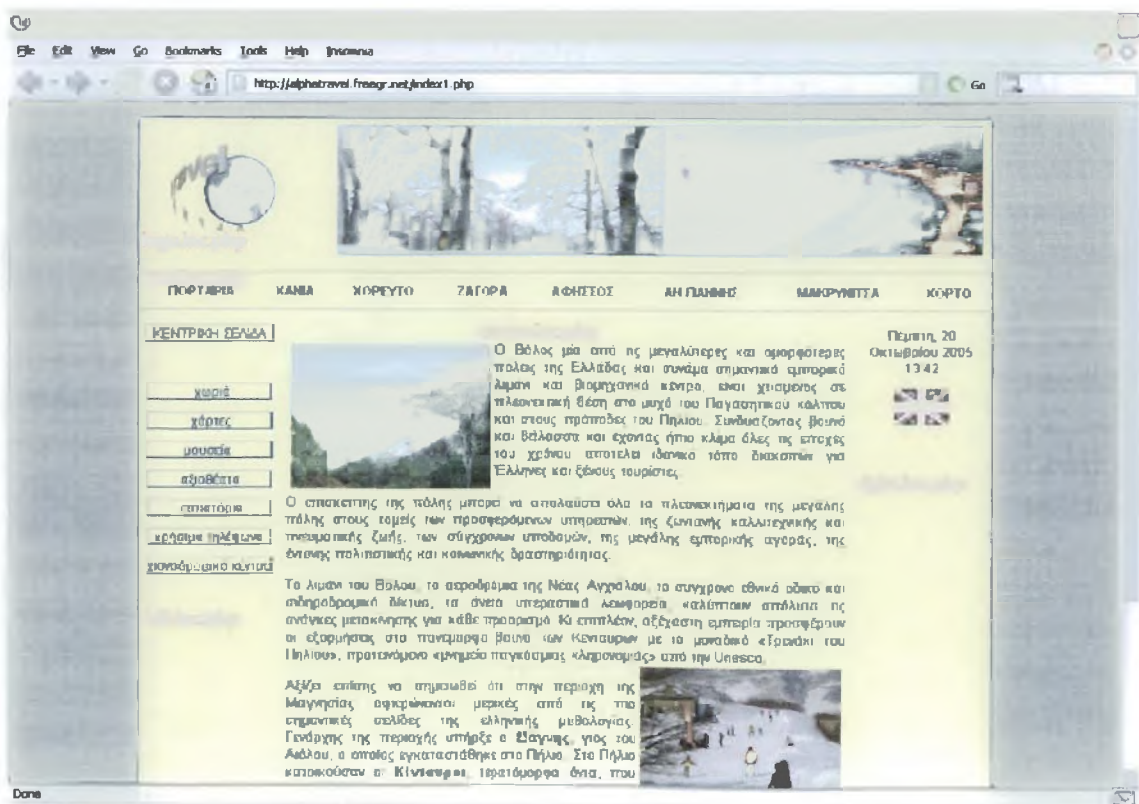
## 8. ΑΝΑΛΥΣΗ ΚΩΔΙΚΑ

Σε αυτή την ενότητα παραθέτουμε τα σημαντικότερα κομμάτια κώδικα τα οποία χρησιμοποιήθηκαν στην εργασία μας. Κάποια από αυτά έχουν προστεθεί αυτόματα από το Dreamweaver τα οποία όμως θα εξηγήσουμε τη λειτουργία τους, και κάποια άλλα τα έχουμε γράψει εμείς για μεγαλύτερη ευελιξία.

### 8.1. Κεντρική σελίδα

Η κεντρική σελίδα αποτελείται συνολικά από 5 υποσελίδες τις οποίες με την εντολή include της ενώνουμε σε μία. Αυτές είναι:

- Logo.inc.php
- Tabs.inc.php
- Left.inc.php
- Center.inc.php
- Right.inc.php



Ο παρακάτω κώδικας είναι ο κώδικας για τη δημιουργία της κεντρικής σελίδας (index1.php).

```
<style type="text/css">
<!--
body {
    background-color: #333366;
}
-->
</style><title>...: HOTELS :...</title>
<?php
session_start();
$_SESSION['page']= basename($_SERVER['PHP_SELF']);
?>

<table width="765" align="center" bgcolor="ffffcc">
  <tr>
    <td colspan="3">
<?php include("logo.inc.php");?>
    </td>
  </tr>
  <tr>
    <td colspan="3">
<?php include("tabs.inc.php");?>
    </td>
  </tr>
  <tr valign="top" bgcolor="#FFFFFF">
    <td valign="top">
<?php include("left.inc.php");?>
    <td valign="top">
<?php include("center.inc.php");?>
    <td valign="top">
<?php include("right.inc.php");?>
    </td>
  </tr>
</table>
</body>
</html>
```

αρχικά δημιουργούμε έναν πίνακα 3 σειρών και 3 στηλών. Στην 1<sup>η</sup> γραμμή στην 1<sup>η</sup> στήλη με την εντολή include βάζουμε τη σελίδα logo.inc.php. Επειδή όμως δε θέλουμε να βάλουμε κάτι άλλο στις υπόλοιπες 2 στήλες και θέλουμε η 1<sup>η</sup> να πιάνει όλο το μήκος των στηλών χρησιμοποιούμε την εντολή colspan.

Στη 2<sup>η</sup> σειρά στην 1<sup>η</sup> στήλη βάζουμε με την εντολή Include την υποσελίδα tabs.inc.php και πάλι χρησιμοποιούμε την εντολή colspan.

Στην 3<sup>η</sup> σειρά στην πρώτη στήλη αντίστοιχα με την εντολή include βάζουμε της σελίδες left.inc.php, center.inc.php, right.inc.php αντίστοιχα στην 1<sup>η</sup>, 2<sup>η</sup>, 3<sup>η</sup> στήλη.

## 8.2. Δημιουργία Πινάκων

Για τη δημιουργία των πινάκων της βάσης μας χρησιμοποίησαμε τον ακόλουθο κώδικα:

```
--
-- Δομή Πίνακα για τον Πίνακα `hotels`
--
CREATE TABLE `hotels` (
  `code` int(6) NOT NULL auto_increment,
  `name` varchar(20) NOT NULL default '',
  `type` varchar(35) NOT NULL default '',
  `owner` varchar(40) NOT NULL default '',
  `phone` varchar(10) NOT NULL default '',
  `place` varchar(35) NOT NULL default '',
  `category` varchar(6) NOT NULL default '',
  `foto` varchar(10) default NULL,
  `info` varchar(50) default NULL,
  `last_change` varchar(10) default NULL,
  PRIMARY KEY (`code`)
) TYPE=MyISAM AUTO_INCREMENT=154 ;

-----

--
--
--
-- Δομή Πίνακα για τον Πίνακα `places`
--
CREATE TABLE `places` (
  `place` varchar(20) NOT NULL default '',
  `place_en` varchar(20) NOT NULL default '',
  `info` text NOT NULL,
  `info_en` text,
  PRIMARY KEY (`place`)
) TYPE=MyISAM;

-----

--
--
--
-- Δομή Πίνακα για τον Πίνακα `prices`
--
CREATE TABLE `prices` (
  `code_hotel` int(6) NOT NULL default '0',
  `monoklino` double NOT NULL default '0',
  `diklino` double NOT NULL default '0',
  `triklino` double NOT NULL default '0',
  `souita` double NOT NULL default '0',
  PRIMARY KEY (`code_hotel`)
```

```
) TYPE=MyISAM;
```

```
-----
```

```
--
```

```
--
```

```
-- Δομή Πίνακα για τον Πίνακα `restaurants`
```

```
--
```

```
CREATE TABLE `restaurants` (
  `code_rest` int(11) NOT NULL auto_increment,
  `name` varchar(40) NOT NULL default '',
  `place` varchar(20) NOT NULL default '',
  `phone` varchar(10) NOT NULL default '',
  `foto` varchar(50) default NULL,
  PRIMARY KEY (`code_rest`)
) TYPE=MyISAM AUTO_INCREMENT=11 ;
```

```
-----
```

```
--
```

```
--
```

```
-- Δομή Πίνακα για τον Πίνακα `users`
```

```
--
```

```
CREATE TABLE `users` (
  `username` varchar(10) NOT NULL default '',
  `password` varchar(15) NOT NULL default '',
  `name` varchar(50) NOT NULL default '',
  PRIMARY KEY (`username`)
) TYPE=MyISAM;
```

Σε κάθε πίνακα ορίζουμε κάθε πρωτεύον κλειδί με την εντολή PRIMARY KEY ('πεδίο').

### 8.3. Ανάκτηση Δεδομένων

Το σημαντικότερο αρχείο της εργασίας ίσως είναι το `results.php` καθώς είναι αυτό το οποίο μας δίνει αποτελέσματα σε κάθε ερώτημά μας. Όταν θέλουμε να ψάξουμε για ξενοδοχεία σε συγκεκριμένο μέρος τρέχουμε το αρχείο `results.php` βάζοντας σαν όρισμα την τοποθεσία που θέλουμε. Το τμήμα του κώδικα που κάνει αυτή τη δουλειά στο `results.php` είναι:

```
if (isset($_GET['place'])) {
    $colname_results = (get_magic_quotes_gpc()) ?
    $_GET['place'] : addslashes($_GET['place']);
}
mysql_select_db($database_conne, $conne);
$query_results = sprintf("SELECT * FROM hotels WHERE
place = '%s' ORDER BY name ASC", $colname_results);
$results = mysql_query($query_results, $conne) or
die(mysql_error());
$row_results = mysql_fetch_assoc($results);
$totalRows_results = mysql_num_rows($results);
```

αρχικά ο κώδικας ελέγχει αν έχει ήδη περαστεί στη μεταβλητή `place` (η τοποθεσία που ψάχνουμε) κάποια τιμή.

#### ΠΑΡΑΤΗΡΗΣΗ:

Επειδή υπάρχουν κάποιοι χαρακτήρες οι οποίοι δεν πρέπει να μπουν σε μια βάση δεδομένων γιατί θα δημιουργήσουν πολλά προβλήματα χρησιμοποιούμε κάποιες εντολές για να είμαστε σίγουροι ότι κατά την εισαγωγή δεδομένων από το χρήστη δε θα παρουσιαστεί κανένα πρόβλημα. Αυτοί οι χαρακτήρες είναι τα μονά εισαγωγικά (`'`), τα διπλά εισαγωγικά (`"`), το backslash (`\`) και το NUL (NULL). Η εντολή `addslashes()` όπου βρει κάποια λέξη που περιέχει κάποιον από τους παραπάνω χαρακτήρες προσθέτει στον αμέσως προηγούμενο χαρακτήρα το `\`.

Ο κώδικας εξετάζει αν έχει ήδη οριστεί η μεταβλητή `place` και την τρέχει με την εντολή `addslashes` που εξηγήσαμε παραπάνω. Αμέσως μετά εκτελεί το ερώτημα που χρειάζεται για να πάρουμε τα αποτελέσματα που θέλουμε. Δηλαδή με την εντολή `SELECT * FROM hotels WHERE place = '%s' ORDER BY name ASC` του λέμε να ψάξει όλον πίνακα `hotels` όπου το πεδίο `place` είναι ίσο με το όρισμα που έχουμε βάλει στη σελίδα `results.php`. Με την εντολή `$results = mysql_query($query_results, $conne) or die(mysql_error());` βάζουμε όλα μας τα αποτελέσματα στη μεταβλητή `$results` και τέλος με την εντολή `$row_results = mysql_fetch_assoc($results);` βάζουμε όλα τα αποτελέσματα της `$results` σε ένα πίνακα για να μπορούμε να τα επεξεργαστούμε πιο εύκολα.

## 8.4. Εισαγωγή Δεδομένων

Εισαγωγή δεδομένων κάνουμε με την εντολή INSERT VALUES ... INTO ...

Για παράδειγμα στον πίνακα hotels όλες οι εγγραφές που έχουμε εισάγει είναι οι εξής:

```

INSERT INTO `hotels` VALUES (1, 'ΜΑΪΣΤΡΑΛΙ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'Μ. ΓΙΑΜΑΚΟΥ', '2423033472', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ', 'Α',
NULL, 'www.mai.com', '30/09/2005');
INSERT INTO `hotels` VALUES (6, 'ΚΑΤΙΑ', 'ΞΕΝΟΔΟΧΕΙΟ', 'Μ.
ΚΡΑΤΗΡΑΣ', '2423033293', 'ΑΦΗΣΣΟΣ', 'Β', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (5, 'ΓΑΛΗΝΗ', 'ΞΕΝΟΔΟΧΕΙΟ', 'Ζ.
ΜΑΡΙΝΑΓΗΣ', '2423033214', 'ΑΦΗΣΣΟΣ', 'Β', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (7, 'ΦΑΡΟΣ', 'ΞΕΝΟΔΟΧΕΙΟ', 'Δ.
ΓΙΑΝΝΟΥΛΗ', '2423033293', 'ΑΦΗΣΣΟΣ', 'Β', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (8, 'ΕΙΡΗΝΗ', 'ΞΕΝΟΔΟΧΕΙΟ', 'ΕΙΡ.
ΚΟΥΤΣΟΦΛΙΝΗ', '2423033287', 'ΑΦΗΣΣΟΣ', 'Α', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (9, 'ΜΠΑΡΜΠΑ-ΘΥΜΙΟΣ', 'ΞΕΝΟΔΟΧΕΙΟ',
'Ε. ΣΩΤΗΡΙΟΥ', '2423033401', 'ΑΦΗΣΣΟΣ', 'Α', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (11, 'ΑΦΗΣΣΟΣ', '', 'Μ. ΓΑΛΑΝΟΥ',
'2423033438', 'ΑΦΗΣΣΟΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (12, 'ΕΥΗ', '', 'Γ. ΚΑΡΑΜΠΑΣΗΣ',
'2423033340', 'ΑΦΗΣΣΟΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (17, 'ΕΛΥΤΗΣ', '', 'Α. ΠΟΥΡΝΑΡΑ',
'2423054482', 'ΕΥΝΟΒΡΥΣΗ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (18, 'ΕΡΩΒΙΟΣ', '', 'Α. ΜΠΙΛΙΑΔΗΣ',
'2423054210', 'ΠΑΛΤΗ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (19, 'ΣΠΑΛΑΘΡΑ', '',
'Γ. ΠΑΠΑΛΕΞΑΝΔΡΟΥ', '2423065326', 'ΧΟΡΤΟ', 'Β', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (20, 'ΚΑΛΑΜΑ', '', 'Α. ΚΩΣΤΑΚΗ',
'2423054200', 'ΚΑΛΑΜΟΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (22, 'ΛΗΔΑ', '', 'Π. ΚΟΝΙΟΡΔΟΣ',
'2423065398', 'ΧΟΡΤΟ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (23, 'ΛΑΥΚΟΣ', 'ΠΑΡΑΔΟΣΙΑΚΟ ΚΑΤΑΛΥΜΑ',
'ΑΛ. ΧΑΛΚΙΑΣ', '2423065017', 'ΛΑΥΚΟΣ', 'Α', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (24, 'ΠΑΡΙΣΗΣ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'ΧΡ. ΘΕΟΧΑΡΟΠΟΥΛΟΣ', '2423065856', 'ΛΑΥΚΟΣ', 'Α', NULL,
NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (26, 'ΑΛΟΗ', 'ΕΠΙΠΛΩΜΕΝΟ ΔΙΑΜΕΡΙΣΜΑ',
'ΠΑΝ. ΜΗΤΣΙΟΥ', '2423031241', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'Α', NULL, NULL,
'25/09/2005');
INSERT INTO `hotels` VALUES (27, 'ΜΑΡΩ', '', 'Κ. ΚΑΗΜΕΝΤΟΠΟΥΛΟΣ',
'2426031282', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'Γ', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (28, 'ΑΤΡΑΠΟΣ', '', 'ΑΝ. ΑΛΤΙΝΗΣ',
'2426031291', 'ΚΙΣΣΟΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (29, 'ΑΝΕΣΙΣ', '', 'Μ. ΡΗΓΟΠΟΥΛΟΣ',
'2426031223', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (30, 'ΒΡΥΩΝΗΣ', '', 'Γ. ΒΡΥΩΝΗΣ',
'2426031301', 'ΑΓΙΟΣ ΔΗΜΗΤΡΙΟΣ', 'Β', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (31, 'ΓΑΛΗΝΗ', '', 'Κ. ΛΕΒΕΝΤΗΣ',
'2426031234', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'Β', NULL, NULL, '0000-00-00');

```



```

INSERT INTO `hotels` VALUES (32, 'ΕΔΕΜ', '', 'Δ. ΚΟΝΤΟΠΟΥΛΟΣ',
'2426031163', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', '', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (33, 'ΕΛΕΑΝΑ', 'ΞΕΝΟΔΟΧΕΙΟ', 'ΕΛ.
ΤΑΣΙΟΥ', '2426031097', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'B', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (34, 'ΑΡΧΟΝΤΙΚΟ ΕΛΝΤΑ', '', 'ΕΛ.
ΤΕΙΚΡΙΚΩΝΗ', '2426031422', 'ΑΓΙΟΣ ΔΗΜΗΤΡΙΟΣ', 'B', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (35, 'ΕΥΤΥΧΙΑ', '', 'Χ. ΕΥΘΥΜΙΟΥ',
'2426031150', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (40, 'ΜΑΙΡΟΥΛΑΣ', '', 'ΕΙΡ.ΚΟΠΑΝΑ',
'2426031478', 'ΑΓΙΟΣ ΔΗΜΗΤΡΙΟΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (41, 'ΝΤΑΜΟΥΧΑΡΗ', '',
'ΑΠ.ΒΑΙΝΟΠΟΥΛΟΣ', '2426049840', 'ΝΤΑΜΟΥΧΑΡΗ', 'B', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (47, 'ΒΑΡΒΑΡΑ', '', 'Π.ΚΑΡΚΑΛΑ',
'2428092946', 'ΑΡΡΙΑ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (48, 'ΚΕΝΤΑΥΡΟΣ', '', 'ΑΠ.ΜΑΧΑΙΡΑΣ',
'2428096224', 'ΑΓΙΟΣ ΛΑΥΡΕΝΤΙΟΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (49, 'ΜΠΙΛΛΑΣ', '', 'Α.ΜΠΟΚΟΥ',
'2428092926', 'ΑΡΡΙΑ', 'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (50, 'ΛΑΟΔΑΜΕΙΑ', '', 'ΧΡ.ΑΣΔΕΡΑΚΗΣ',
'2428078800', 'ΝΕΑ ΑΓΧΙΑΛΟΣ', 'Γ', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (52, 'ΑΠΟΛΛΩΝ', '', 'ΕΛ.ΤΟΚΑΛΗ',
'2428077578', 'ΝΕΑ ΑΓΧΙΑΛΟΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (53, 'ΑΡΑΠΙΤΣΕΑΣ', '', 'ΔΗΜ.ΑΡΑΠΙΤΣΕΑΣ',
'2422092083', 'ΑΧΙΛΛΕΙΟ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (54, 'ΜΕΓΑΣ ΑΛΕΞΑΝΔΡΟΣ', '',
'ΑΝ.ΑΝΑΓΝΩΣΤΟΥ', '2428076752', 'ΝΕΑ ΑΓΧΙΑΛΟΣ', 'B', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (55, 'ΘΩΜΑΣ', '', 'Β.ΔΗΜΗΤΡΙΟΥ',
'2422022077', 'ΑΛΜΥΡΟΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (56, 'ΠΡΩΤΕΣΣΙΑΛΟΣ', '',
'ΧΡ.ΑΣΔΕΡΑΚΗΣ', '2428076310', 'ΝΕΑ ΑΓΧΙΑΛΟΣ', 'B', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (58, 'ΠΥΡΑΣΣΟΣ', '', 'Ε.ΑΣΔΕΡΑΚΗ',
'2428076256', 'ΝΕΑ ΑΓΧΙΑΛΟΣ', 'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (59, 'ΘΑΣΙΣ', '', 'Γ.ΙΑΚΩΒΙΔΗΣ',
'2422091297', 'ΑΜΑΛΙΑΠΟΛΗ', 'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (60, 'ΤΟΞΟΤΕΣ', '', 'Α.ΣΙΑΦΑΚΑΣ',
'2428096600', 'ΧΑΝΙΑ', 'Δ', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (61, 'ΧΑΝΙΑ', '', 'Α.ΖΑΧΑΡΗΣ',
'2428096421', 'ΧΑΝΙΑ', 'Γ', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (62, 'ΜΑΝΘΟΣ', '', 'Κ.ΚΕΦΑΛΑΣ',
'2428096402', 'ΧΑΝΙΑ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (63, 'ΤΑΣΙΑ', 'ΕΠΙΠΛΩΜΕΝΟ ΔΙΑΜΕΡΙΣΜΑ',
'ΑΝ.ΚΟΚΚΙΝΗ', '2428096410', 'ΑΦΗΣΣΟΣ', 'A', NULL, NULL,
'09/10/2005');
INSERT INTO `hotels` VALUES (64, 'ΧΑΝΙ ΤΟΥ ΖΗΕΗ', '',
'Ζ.ΣΤΑΜΑΤΗΣ', '2428096414', 'ΧΑΝΙΑ', 'B', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (65, 'ΠΗΛΕΑΣ', '', 'Α.ΠΑΤΙΔΗΣ',
'2428096431', 'ΧΑΝΙΑ', 'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (67, 'ΒΟΤΣΑΛΑ', 'ΞΕΝΟΔΟΧΕΙΟ',
'Χ.ΛΑΣΚΟΣ', '2426022001', 'ΧΟΡΕΥΤΟ', 'B', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (68, 'ΕΡΑΤΩ', 'ΞΕΝΟΔΟΧΕΙΟ', 'Ε.ΡΗΝΟΥ',
'2426022445', 'ΧΟΡΕΥΤΟ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (72, 'ΑΙΓΕΥΣ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'Ε.ΤΣΙΜΠΟΥΚΗ', '2426022778', 'ΑΓΙΟΙ ΣΑΡΑΝΤΑ', 'A',
NULL, NULL, '02/12/2005');

```

```

INSERT INTO `hotels` VALUES (74, 'ΧΑΡΑΥΓΗ', 'ΞΕΝΟΔΟΧΕΙΟ',
'Κ.ΘΑΡΣΙΤΗΣ', '2426022814', 'ΖΑΓΟΡΑ', 'A', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (75, 'ΑΙΟΛΟΣ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΣΟΦ.ΚΙΤΣΙΟΣ', '2426022910', 'ΑΦΗΣΣΟΣ', 'A', NULL,
NULL, '04/10/2005');
INSERT INTO `hotels` VALUES (77, 'ΦΛΑΜΙΝΓΚΟ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΣΤ.ΜΠΑΡΟΥΤΑΣ', '2426022815', 'ΧΟΡΕΥΤΟ', 'B', NULL,
NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (78, 'ΧΑΓΙΑΤΙ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'Τ.ΤΡΙΑΝΤΑΦΥΛΛΟΥ', '2426023540', 'ΧΟΡΕΥΤΟ', 'B',
NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (79, 'ΓΚΑΓΙΑΝΝΗ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'Ν.ΓΚΑΓΙΑΝΝΗΣ', '2426023391', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ', 'A',
NULL, 'www.gka.net', '30/09/2005');
INSERT INTO `hotels` VALUES (80, 'ΔΡΑΚΟΠΟΥΛΟΥ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'Μ.ΣΤΑΜΟΥ', '2426023566', 'ΖΑΓΟΡΑ', 'A', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (81, 'ΠΡΙΓΚΟΥ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'Ι.ΠΡΙΓΚΟΣ', '2426023627', 'ΖΑΓΟΡΑ', 'A', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (82, 'ΣΤΑΜΟΥ', 'TRADITIONAL HOUSE',
'ΚΑΤ.ΣΤΑΜΟΥ', '2426023880', 'ΖΑΓΟΡΑ', 'A', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (83, 'ΑΚΡΟΓΙΑΛΙ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΗΛ.ΜΑΝΤΖΑΦΛΕΡΗΣ', '2423023146', 'ΑΓΙΟΣ ΙΩΑΝΝΗΣ',
'A', NULL, NULL, '29/09/2005');
INSERT INTO `hotels` VALUES (84, 'ΑΛΚΥΩΝ', 'ΞΕΝΟΔΟΧΕΙΟ',
'M.ΤΖΑΒΕΛΛΑ', '2423022364', 'ΚΑΛΑ ΝΕΡΑ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (85, 'ΙΖΕΛΑ', 'ΞΕΝΟΔΟΧΕΙΟ',
'A.ΤΣΙΜΑΣ', '2423022379', 'ΚΑΛΑ ΝΕΡΑ', 'B', NULL, NULL, '0000-00-
00');
INSERT INTO `hotels` VALUES (86, 'ΚΑΣΣΑΝΔΡΑ', 'ΞΕΝΟΔΟΧΕΙΟ',
'Φ.ΦΑΝΑΡΙΩΤΟΥ', '2423022764', 'ΚΑΛΑ ΝΕΡΑ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (87, 'ΠΗΓΑΣΟΣ', 'ΞΕΝΟΔΟΧΕΙΟ',
'I.ΒΑΣΔΕΚΗΣ', '2423023111', 'ΚΑΛΑ ΝΕΡΑ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (88, 'ΡΟΔΙΑ', 'ΞΕΝΟΔΟΧΕΙΟ',
'Δ.ΑΝΑΓΝΩΣΤΟΥ', '2423022566', 'ΚΑΛΑ ΝΕΡΑ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (89, 'ΣΕΡΑΝΙΔΕΣ', 'ΞΕΝΟΔΟΧΕΙΟ',
'ΑΓ.ΜΠΛΑΝΑ', '2423022164', 'ΚΑΤΩ ΓΑΤΖΕΑ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (91, 'ΑΡΓΩ', 'ΞΕΝΟΔΟΧΕΙΟ',
'ΠΑ.ΠΟΛΥΧΡΟΝΟΥ', '2423022271', 'ΚΑΛΑ ΝΕΡΑ', 'A', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (92, 'ΒΙΚΤΩΡΙΑ', 'ΞΕΝΟΔΟΧΕΙΟ',
'ΑΙΚ.ΑΝΑΓΝΩΣΤΟΥ', '2423022219', 'ΚΑΛΑ ΝΕΡΑ', 'A', NULL, NULL,
'0000-00-00');
INSERT INTO `hotels` VALUES (95, 'ΙΑΤΡΙΑΗ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'ΣΤ.ΙΑΤΡΙΑΗΣ', '2426049477', 'ΜΟΥΡΕΣΙ', 'B', NULL,
NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (96, 'ΟΛΓΑ', 'ΠΑΡΑΔΟΣΙΑΚΟ ΚΑΤΑΛΥΜΑ',
'ΑΓΓ.ΑΥΓΕΡΙΝΟΥ', '2426049651', 'ΜΟΥΡΕΣΙ', 'B', NULL, NULL, '0000-
00-00');
INSERT INTO `hotels` VALUES (97, 'ΝΤΕΛΚΗΣ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΠΑΝ.ΝΤΕΛΚΗΣ', '2428092047', 'ΑΓΡΙΑ', 'B', NULL,
NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (98, 'ΠΛΑΤΑΝΙΔΙΑ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΕΛ.ΓΚΟΒΑΡΗ', '2428093625', 'ΠΛΑΤΑΝΙΔΙΑ', 'B', NULL,
NULL, '0000-00-00');

```

```
INSERT INTO `hotels` VALUES (101, 'ΒΟΓΙΑΤΖΟΠΟΥΛΟΥ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'Μ.ΣΠΑΝΟΣ', '2428093135', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ ΝΗΛΕΙΑΣ',
'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (102, 'ΧΡΥΣΗ ΝΕΦΕΛΗ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'ΠΑΝ.ΔΗΜΗΤΡΙΟΥ', '2103728028', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ
ΝΗΛΕΙΑΣ', 'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (103, 'ΔΕΡΕΛΗ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'Μ.ΔΕΡΕΛΗ', '2428093163', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ ΝΗΛΕΙΑΣ',
'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (104, 'ΤΖΩΡΤΖΗ', 'ΠΑΡΑΔΟΣΙΑΚΟ
ΚΑΤΑΛΥΜΑ', 'Ε.ΠΑΤΣΙΑΔΑ', '2428094252', 'ΑΓΙΟΣ ΓΕΩΡΓΙΟΣ ΝΗΛΕΙΑΣ',
'B', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (105, 'ΕΛΙΣΑΒΕΤ', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'Χ.ΠΑΡΑΣΚΕΥΟΠΟΥΛΟΣ', '2428076234', 'ΝΕΑ ΑΓΧΙΑΛΟΣ',
'A', NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (106, 'NANBY', 'ΕΠΙΠΛΩΜΕΝΟ
ΔΙΑΜΕΡΙΣΜΑ', 'ΣΤ.ΠΡΙΟΝΙΣΤΗΣ', '2422091227', 'ΑΜΑΛΙΑΠΟΛΗ', 'A',
NULL, NULL, '0000-00-00');
INSERT INTO `hotels` VALUES (110, 'NATAΣΑ APARTMENTS', '', 'NATAΣΑ
ΘΕΟΧΑΡΟΠΟΥΛΟΥ', '6947637544', 'ΔΙΜΗΝΙ', 'A', NULL, NULL, '0000-00-
00');
```

## 8.5. Session & cookies

Session και cookies είναι δύο διαφορετικές μέθοδοι για το πέρασμα τιμών μεταξύ των διάφορων σελίδων ενός site.

### SESSIONS

- ✓ αποθηκεύονται στο server (server-side)
- ✓ η πληροφορία την οποία αποθηκεύουν είναι ενεργή όσο ο browser είναι ανοιχτός, όταν ο browser κλείσει οι πληροφορίες πρέπει να ξαναεισαχθούν από το χρήστη.
- ✓ Το γεγονός ότι τα sessions αποθηκεύονται στον server(χρησιμοποιούν resources του server), μπορεί να επιφέρουν μεγάλη επιβάρυνση στο server.
- ✓ είναι πιο ασφαλείς σε σχέση με τα cookies

### COOKIES

- ✓ αποθηκεύονται στον client(στο pc μας)
- ✓ η πληροφορία την οποία αποθηκεύουν είναι ενεργή μέχρι να λήξει η διάρκεια ζωής που είχε οριστεί κατά την δημιουργία τους, ακόμη και ο browser να κλείσει, στην επόμενη επίσκεψη μας στο site η πληροφορία θα είναι ενεργή (δεν θα χρειαστεί να ξανακάνουμε Login)εκτός και εάν έχει λήξει το cookie
- ✓ Πολλοί χρήστες έχουν απενεργοποιημένα τα cookies στον υπολογιστή τους οπότε σε αυτή την περίπτωση δεν είναι δυνατόν να αποθηκευτεί η πληροφορία του χρήστη

Για την εργασία μας αυτό που ζητάγαμε είναι να μπορεί ο διαχειριστής να μεταβαίνει σε όσες σελίδες ο ίδιος θέλει για να αλλάξει τα δεδομένα (ξενοδοχεία / εστιατόρια) χωρίς να χρειάζεται κάθε φορά να πιστοποιεί την ταυτότητά του. Αν χρησιμοποιούσαμε cookies αυτό θα ήταν εφικτό αλλά την επόμενη φορά που θα προσπαθούσε να ξαναμπεί σε αυτές τις σελίδες δε θα του ζηταγε ούτε username ούτε password κάτι το οποίο βεβαίως είναι ανασφαλές. Κυρίως για το λόγο αυτό χρησιμοποιήσαμε sessions.

#### Βασικές εντολές session

```
Session_start()
```

Με αυτή την εντολή δημιουργούμε ένα session.

```
Session_unset()
```

Ακυρώνει όλες τις μεταβλητές του session

```
Session_destroy()
```

Κλείνει ένα session

```
$_SESSION['user']=natasa
```

με αυτόν τον τρόπο περνάμε τιμές σε μεταβλητές session. Πχ τώρα περνάει στη session μεταβλητή user την τιμή natasa την οποία θα μπορούμε να προσπελάσουμε σε οποιαδήποτε σελίδα έχουμε ξεκινήσει session.

Παράδειγμα αναλύεται στην ενότητα 7.2.

## 9. ΒΙΒΛΙΟΓΡΑΦΙΑ

- ✓ ΜΑΘΕΤΕ PHP MYSQL ΚΑΙ APACHE ΟΛΑ ΣΕ ΕΝΑ (Mellonie Julie, εκδόσεις ΓΚΙΟΥΡΔΑΣ)
- ✓ Sams Teach Yourself MySQL In 24 Hours
- ✓ Εγχειρίδιο php

### Δικτυακοί τόποι

- ✓ [www.mysql.gr](http://www.mysql.gr)
- ✓ [www.php.gr](http://www.php.gr)

### Forum με οδηγίες

- ✓ [www.mysql.gr](http://www.mysql.gr)
- ✓ [www.php.gr](http://www.php.gr)