ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ

# Προγραμματισμός Διαδικτύου

# Ασκήσεις Εργαστηρίου

**Ενότητα: ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ No 09**

Δρ. Μηνάς Δασυγένης

mdasyg@ieee.org

**Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών**

Εργαστήριο Ψηφιακών Συστημάτων και Αρχιτεκτονικής Υπολογιστών

http:// arch.icte.uowm.gr/mdasyg

# Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.


Έκδοση Σεπτέμβριος 2011

# Περιεχόμενα

1. Σκοπός της άσκησης ..... 4
2. Ερωτήσεις/Ασκήσεις ..... 4

# 1. Σκοπός της άσκησης

Εργαστήριο PHP#5:

- Εμφάνιση αριθμού αξιολογήσεων.
- Απλή εγγραφή χρήστη.
- Ταυτοποίηση Χρήστη.
- Διατήρηση της κατάστασης με Συνοδούς (session), και Μπισκότα (Cookies).

# 2. Ερωτήσεις/Ασκήσεις

Δημιουργήστε ένα φάκελο με το όνομα lab09 μέσα στο htdocs του xampp και τοποθετήστε μέσα σε αυτόν τα αρχεία που είχαν δημιουργηθεί στο προηγούμενο εργαστήριο.

Επισκεφτείτε τη σελίδα http://localhost/lab09 και επιβεβαιώστε ότι εμφανίζεται σωστά ο ιστοχώρος.

### Εμφάνιση Αριθμού reviews στην αναζήτηση

Προκειμένου να κάνουμε πιο χρήσιμη την αναζήτηση αξιολογήσεων θα πρέπει να προβούμε στις κατάλληλες αλλαγές ώστε: (α) να εμφανίζεται ο αριθμός των αποτελεσμάτων δίπλα σε κάθε όνομα και (β) να μπορεί ο χρήστης να πατήσει πάνω σε μια αξιολόγηση και να βλέπει αναλυτικά όλες τις αξιολογήσεις.

Προκειμένου να έχουμε τον αριθμό των αποτελεσμάτων, η πιο καλή λύση είναι να προσθέσουμε μια στήλη που να κρατάει αυτόν τον αριθμό. Επίσης, επειδή ενδέχεται κάποια ονόματα που γράφονται διαφορετικά να αντιστοιχούν στο ίδιο αντικείμενο θα πρέπει να προσθέσουμε μια στήλη στην οποία θα καταγράφεται (αν υπάρχει) το reviewed_id που είναι συνώνυμο με αυτό.

**Προσθέστε** στον πίνακα reviewed_names 2 στήλες:

- nr_reviews, int, not null, προκαθορισμένο (As defined:) με τιμή 0
- alias, int,null ok

επίσης από το **εμφάνιση σχέσεων** να ορίσετε μια εσωτερική σχέση της στήλης alias με τη στήλη reviewed_id .

Το επόμενο βήμα είναι να τροποποιήσουμε τον κώδικα που προσθέτει αξιολογήσεις στη βάση.

- Ανοίξτε το αρχείο add_process.php
- Στο σημείο του κώδικα που βρίσκουμε το id του στοιχείου, δηλαδή στο:

```
/* Start inserting to database */
$sql = "select * from reviewed_names where reviewed_name='$elementname' Limit 1";
@ $result = mysql_query($sql);
if (mysql_numrows($result) >0 && $row = mysql_fetch_array($result))
        {
        $reviewed_id=$row[reviewed_id];
        echo "Βρέθηκε εγγραφή για την ονομασία $elementname με id=".$reviewed_id;
        }
```

θα πρέπει να τοποθετήσετε μέσα στο if (mysql_numrows($result)... τη γραμμή

**$nr_reviews=$row[nr_reviews];**

ώστε να διαβάζουμε και αυτήν τη στήλη μαζί με το id.

- Η παραπάνω μεταβλητή nr_reviews θα πρέπει να τοποθετηθεί και μέσα στο else που ακολουθεί ακριβώς από κάτω (δηλαδή στην περίπτωση που δεν εκτελεστεί το παραπάνω if) και να τις δώσουμε τιμή 0

**$nr_reviews=0;**

- Το επόμενο βήμα είναι πριν την SQL εντολή insert που βρίσκεται λίγες γραμμές παρακάτω και έξω από την else { }, να αυξήσουμε το nr_reviews κατά 1 (αφού προσθέτουμε μια νέα καταχώρηση για αυτό το στοιχείο), δηλαδή:

**$nr_reviews=$nr_reviews + 1;**

- Στη συνέχεια θα πρέπει εκτός από την εντολή SQL Insert που εκτελείται εκτελέσουμε μια ακόμη εντολή SQL update (ενημέρωσης) στον πίνακα reviewed_names. Σε αυτήν την εντολή θα ενημερώνουμε τη στήλη nr_reviews του πίνακα reviewed_names τη νέα τιμή που έχουμε υπολογίσει. (Αφού προσθέτουμε μια νέα καταχώρηση πρέπει να αυξήσουμε το nr_reviews κατά 1.) Η εντολή sql που πρέπει να εκτελέσετε ύστερα από τη εντολή SQL που υπάρχει (και κάτω από το if που ελέγχει αν εκτελέστηκε ή όχι) είναι η παρακάτω:

**$sql="update reviewed_names set nr_reviews='$nr_reviews' where reviewed_id='$reviewed_id'";**

Τοποθετήστε τον υπόλοιπο κώδικα που εκτελεί το ερώτημα που σας δίνεται και επίσης γίνεται ο έλεγχος για το αν εκτελέστηκε σωστά το ερώτημα ή όχι (με παρόμοιο τρόπο με τις προηγούμενες SQL εντολές).

- Επισκεφτείτε τη σελίδα http://localhost/lab09 και προσθέστε μια αξιολόγηση για να επιβεβαιώστε την ορθή λειτουργία του κώδικα που μόλις φτιάξατε.
- Προσθέστε μια ακόμη αξιολόγηση (με το ίδιο όνομα) και δείτε στο phpmyadmin αν ο μετρητής των αξιολογήσεων (στήλη nr_reviews στο reviewed_names) έχει πάρει την τιμή 2.
- Στη συνέχεια θα πρέπει να τροποποιήσουμε το αρχείο που αντιστοιχεί στην εύρεση αξιολογήσεων ώστε μετά το όνομα να εμφανίζει μέσα σε παρένθεση τον αριθμό των αξιολογήσεων που υπάρχουν).
- Ανοίξτε το αρχείο search.php
- Στο σημείο που γίνεται η εκτύπωση του ονόματος, αν βρεθεί, θέλουμε στην ίδια γραμμή μέσα σε παρένθεση να βρίσκεται και ο αριθμός των αξιολογήσεων. Θα πρέπει λοιπόν να τροποποιήσουμε τη γραμμή ώστε να συμπεριλάβουμε και τη στήλη nr_reviews ως εξής:

**echo "<li>$row[reviewed_name] ($row[nr_reviews])</li>";**

- Επιβεβαιώστε ότι λειτουργεί. Επισκεφτείτε τη σελίδα και ψάξτε να βρείτε μια καταχώρηση που υπάρχει στη βάση σας.
- Το επόμενο βήμα είναι να επιτρέψουμε στο χρήστη να επιλέξει (να πατήσει κλικ) κάποιο από τα αποτελέσματα και να ανοίξει μια καινούργια σελίδα στην οποία να φαίνονται αναλυτικά τα αποτελέσματα. Θα πρέπει λοιπόν να προσθέσουμε στον ανωτέρω κώδικα που εμφανίζει μια γραμμή για κάθε αποτέλεσμα κατάλληλη HTML μορφοποίηση ώστε να μπορεί ο επισκέπτης να πατήσει πάνω στο όνομα.

Επειδή θέλουμε όταν πατάει ο χρήστης σε ένα δεσμό να μεταφέρεται η πληροφορία για ποιο αντικείμενο θέλει να δει τις αξιολογήσεις, αυτό θα γίνει με αποστολή των στοιχείων μέσω GET . Αν τα τοποθετούσαμε τα στοιχεία μέσα σε φόρμα τότε θα μπορούσαμε να χρησιμοποιήσουμε και το POST (κάτι που δε μπορούμε να το κάνουμε τώρα, γιατί δεν έχουμε φόρμα).

Επίσης, χρησιμοποιούμε τη μορφή GET επειδή αυτό σημαίνει ότι μπορεί κάποιος από το internet να δει αμέσως τις αξιολογήσεις για ένα συγκεκριμένο site αν έχει κάνει bookmark το δεσμό. Ο δεσμός θα έχει τη μορφή

http://localhost/get_review.php?id=XXXXXX

όπου XXXXXX είναι το reviewed_id, το οποίο μπορούμε να το χρησιμοποιήσουμε μέσω $row[reviewed_id].

- Τροποποιήστε κατάλληλα τον κώδικα του βήματος με το echo (λίγες γραμμές πιο πριν), ώστε να τοποθετηθούν δεσμοί στις λέξεις $row[reviewed_name] που να παραπέμπουν στο get_review.php?id=$row[reviewed_id]

(δηλαδή να χρησιμοποιηθεί η δομή:
**<a href=get_review.php?id=$row[reviewed_id]> … </a>**

- Επιβεβαιώστε ότι λειτουργεί. Επισκεφτείτε τη σελίδα και ψάξτε να βρείτε μια καταχώρηση που υπάρχει στη βάση σας. Θα πρέπει να εμφανιστεί η καταχώρηση ως δεσμός, δηλαδή να μπορείτε να πατήσετε πάνω της. Ασφαλώς, το αρχείο get_review.php δεν υπάρχει, οπότε θα πρέπει να το δημιουργήσετε.
- Αντιγράψτε το αρχείο template.php στο get_review.php . Ανοίξτε το αρχείο get_review.php
- Μέσα στο τμήμα maincontent τοποθετήστε τα <?php και ?>
- Αρχικά θα πρέπει να εξάγουμε την πληροφορία που μας δίνεται στο $_GET[id]. Μάλιστα, προκειμένου να αποφύγουμε πιθανό πρόβλημα από κακόβουλους χρήστες επειδή γνωρίζουμε ότι είναι αριθμός, θα κάνουμε διπλή μετατροπή για αν αποφύγουμε το πρόβλημα. Δώστε, λοιπόν:

**$reviewed_id=strval(intval($_GET['id']));**

- Στη συνέχεια ανοίξτε τη σύνδεση με τη βάση δεδομένων και λίγο πριν το τέλος του τμήματος div, κλείστε τη σύνδεση.
- Τοποθετήστε στη μεταβλητή $sql το ερώτημα που θέλετε να στείλετε προς τη βάση. Το ερώτημα αυτό είναι να επιλεγούν όλες οι καταχωρήσεις (select *) από τον πίνακα reviewsgr (from reviewsgr) οι οποίες έχουν reviewed_id (where reviewed_id=$reviewed_id) ίσο με αυτό που βρήκαμε πιο πριν.
- Εκτελέστε το SQL ερώτημα.
- Αν το mysql_numrows του προηγούμενου βήματος επιστρέψει τιμή με πάνω από 0 αποτελέσματα (δηλαδή βρέθηκαν καταχωρήσεις) τότε να εμφανίσει το μήνυμα “Βρέθηκαν οι παρακάτω αξιολογήσεις”. Αν δε βρεθούν αξιολογήσεις να εμφανιστεί το μήνυμα “Δε βρέθηκε καμία αξιολόγηση”
- Επιβεβαιώστε ότι λειτουργεί. Επισκεφτείτε τη σελίδα και ψάξτε να βρείτε μια καταχώρηση που υπάρχει στη βάση σας. Θα πρέπει να εμφανιστεί η καταχώρηση ως δεσμός, δηλαδή να μπορείτε να πατήσετε πάνω της. Πατήστε πάνω στο δεσμό σε μια καταχώρηση που έχει αξιολογήσεις και δείτε αν εμφανίζεται το μήνυμα ότι έχουν βρεθεί αξιολογήσεις.
- Το επόμενο βήμα είναι να εμφανίζει τις αξιολογήσεις. Για να γίνει αυτό θα χρησιμοποιήσουμε μια δομή while() η οποία θα εκτελεί τη συνάρτηση mysql_fetch_array (ακριβώς όπως σε προηγούμενα παραδείγματα).
- Μέσα στη δομή επανάληψης θέλουμε να εμφανίζει το χρήστη, την αξιολόγηση που έδωσε και τους πρώτους 20 χαρακτήρες. Για να το κάνουμε αυτό θα χρησιμοποιήσουμε την παρακάτω δομή κώδικα:

```
echo "<li>Χρήστης $row[user_id] - Αξιολόγηση $row[review_rating]<br>" ;
echo substr($row[review_text],0,20);
echo "<a href='get_review.php?detailed_id=$row[reviews_id]'>Περισσότερα...</a>";
echo "</li>";
```

- Παρατηρήστε ότι αν ο χρήστης θέλει να δει περισσότερες πληροφορίες (π.χ. όλο το κείμενο και τις εικόνες) θα πρέπει να πατήσει σε ένα δεσμό, ο οποίος οδηγεί στο ίδιο αρχείο αλλά αυτή τη φορά η μεταβλητή στο GET που χρησιμοποιούμε είναι η detailed_id.
- Επιβεβαιώστε ότι λειτουργεί η σελίδα έως αυτό το σημείο.

- Θα πρέπει στη συνέχεια να διαχωρίσουμε τις περιπτώσεις στις οποίες έχουμε είτε το id είτε το detailed_id στο URL. Αρχικά θα ελέγχουμε αν υπάρχει η παράμετρος id, και τότε θα εκτελείται το παραπάνω κομμάτι κώδικα.
- Αν δεν υπάρχει η παράμετρος id ($_GET['id']) αλλά υπάρχει η παράμετρος detailed_id ($_GET['detailed_id']) τότε θα γίνεται το SQL ερώτημα που θα επιλέγει από το reviewsgr την καταχώρηση με reviews_id ίσο με detailed_id.
- Κατασκευάστε τη παραπάνω δομή if
- Στη συνέχεια θα εκτυπώνετε τα παρακάτω στοιχεία, δηλαδή το χρήστη, την αξιολόγηση και τις φωτογραφίες που ίσως έχουν ανέβει (σημείωση: Χρησιμοποιήστε τη μεταβλητή $userimages η οποία θα δείχνει στον κατάλογο που βρίσκονται οι εικόνες (δηλαδή στην πλήρη διαδρομή για το userimages):

```
echo "<h2>Χρήστης: $row[user_id] - Αξιολόγηση: $row[review_rating]</h2><br>" ;
echo $row[review_text];
if(file_exists("$userimages/$row[review_fileimage1]"))
{echo "<hr /><img src=\"userimages/$row[review_fileimage1]\" />";}
if(file_exists("$userimages/$row[review_fileimage2]"))
{echo "<hr /><img src=\"userimages/$row[review_fileimage2]\" />";}
if(file_exists("$userimages/$row[review_fileimage3]"))
{echo "<hr /><img src=\"userimages/$row[review_fileimage3]\" />";}
echo "";
```

- Σε οποιαδήποτε άλλη περίπτωση θα εμφανίζεται ένα μήνυμα ότι “Δε βρέθηκε καμία καταχώρηση”.
- Κατασκευάστε την παραπάνω δομή if { } ...elseif { ..} else {..} μαζί με τον κώδικα μέσα στις αγκύλες. Ο έλεγχος για το αν υπάρχει κάποια μεταβλητή γίνεται μέσω τη συνάρτησης isset();
- Επιβεβαιώστε ότι λειτουργεί η σελίδα έως αυτό το σημείο. Πατήστε το δεσμό Περισσότερα.. που εμφανίζεται όταν κάνουμε μια εύρεση αξιολόγησης.

**Σύνδεση & Αποσύνδεση των χρηστών**

Προκειμένου να επιτρέψουμε στους χρήστες να συνδέονται, θα πρέπει να κατασκευάσουμε:

- Τη φόρμα στην οποία θα γράφονται
- Τη φόρμα στην οποία ο διαχειριστής θα ενεργοποιεί τους χρήστες που έχουν κάνει εγγραφή. (η επιβεβαίωση μπορεί να γίνει και μέσω e-mail)
- Sessions, Cookies

## Τροποποίηση της βάσης δεδομένων

Τροποποιήστε τον πίνακα users και προσθέστε 2 πεδία:

- userlevel tinyint not null
- timestamp, int not null

Επίσης, επειδή θα θέλουμε να κρατάμε σε ένα πίνακα και τους χρήστες που έχουν συνδεθεί, δημιουργήστε ένα πίνακα active_users ως εξής:

```
CREATE TABLE active_users (
 username varchar(30) primary key,
 timestamp int(11) unsigned not null
);
```

Ακόμη, θα θέλουμε να έχουμε ένα πίνακα για τους ανώνυμους επισκέπτες, δηλαδή τους χρήστες που δεν έχουν κάνει login. Δημιουργήστε ένα πίνακα active_guests με:

```
CREATE TABLE active_guests (
 ip varchar(15) primary key,
 timestamp int(11) unsigned not null
);
```

Τέλος θα πρέπει να κατασκευάσουμε ένα πίνακα στον οποίο θα τοποθετούμε τους χρήστες που θα τους απαγορεύουμε προσωρινά να συνδεθούν (banned users) ως εξής:

```
CREATE TABLE banned_users (
 username varchar(30) primary key,
 timestamp int(11) unsigned not null
);
```

### Κατασκευή φόρμας εγγραφής χρήστη

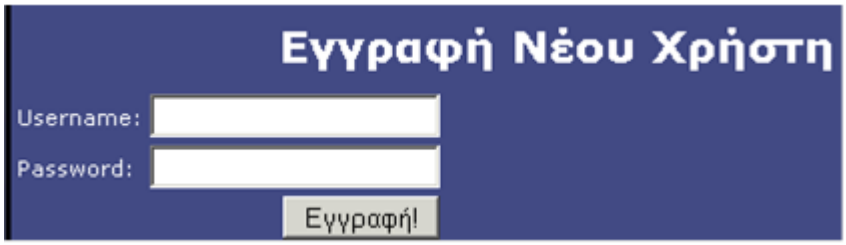
- Ανοίξτε το αρχείο left_column.php και πηγαίνετε στο σημείο που υπάρχει η φόρμα για login. Πριν κλείσει η φόρμα συμπληρώστε τις παρακάτω γραμμές οι οποίες θα εμφανίσουν το κουμπί Σύνδεση (αν το έχετε ήδη στη φόρμα παραλείψετε αυτή γραμμή) και το δεσμό για δωρεάν εγγραφή:

```
<input type="submit" name="submit" value="Σύνδεση" /><br />
<span><a href="signup.php"> Δωρεάν εγγραφή </a></span>
```

- Τροποποιήστε κατάλληλα το CSS (αυξήστε το ύψος του τμήματος login) ώστε να μην εμφανίζεται κανένα πρόβλημα.
- Στις πρώτες γραμμές του CSS (πριν την εγγραφή για το #login) προσθέστε τις παρακάτω γραμμές ώστε να είναι πιο εμφανές ο δεσμός για 'Δωρεάν εγγραφή':

```
#login span {
background-color:#FFFFFF;
}
```

- Αντιγράψτε το αρχείο template.php στο αρχείο signup.php και ανοίξτε το για επεξεργασία.
- Μέσα στο τμήμα maincontent τοποθετήστε τα <?php και ?>
- Ανοίξτε τα άγκυρα { } (τα οποία θα τα χρησιμοποιήσετε σε μερικά βήματα σε μια δομή if.
- Μέσα στα άγκυρα τοποθετήστε αυτά που αναφέρονται στο επόμενο βήμα:
- Τοποθετήστε τον κατάλληλο κώδικα, ώστε να δημιουργήστε την παρακάτω (απλή) φόρμα εγγραφής χρήστη χρησιμοποιώντας το php heredoc (Εικόνα 1). Δε θα τοποθετήσουμε σε αυτό το σημείο πιο εξεζητημένους ελέγχους και πεδία για να γίνει καλύτερα κατανοητή η λειτουργία του συστήματος.




Εικόνα 1- Εγγραφή νέου χρήστη

- Ονοματίστε τα πεδία της φόρμας ως username, password.
- Επειδή γίνεται αποστολή των στοιχείων της φόρμας στο ίδιο αρχείο php, θα πρέπει να διαχωρίσουμε τις δύο περιπτώσεις στις οποίες είτε έχουμε κάνει αποστολή στοιχείων είτε όχι, με μια δομή if ως εξης:

```
if (!empty($_POST['username']) && !empty($_POST['password']))
{/*...κώδικας χειρισμού φόρμας ...*/}
else
{/*...εκτύπωση φόρμας σύμφωνα με προηγούμενο βήμα*/}
```

- Επιβεβαιώστε την ορθή λειτουργία της φόρμας. Συμπληρώστε στο όνομα χρήστη user και στον κωδικό user123 και πατήστε Εγγραφή. Θα πρέπει να δείτε μια κενή σελίδα. Μόλις τη δείτε αφήστε το παράθυρο ανοιχτό και συνεχίστε με την επεξεργασία του κώδικα χειρισμού της φόρμας ως εξής:
- Τροποποιήστε αρχικά τη βάση δεδομένων στον πίνακα users το πεδίο password το οποίο θα πρέπει να είναι varchar με μέγεθος 64 χαρακτήρες. Αυτό γίνεται γιατί θα αποθηκεύουμε μια κρυπτογραφημένη μορφή του κωδικού η οποία θα έχει αρκετούς περισσότερους χαρακτήρες από αυτούς που θα δώσει ο χρήστης.
- Ανοίξτε τη σύνδεση με τη βάση δεδομένων αμέσως μετά το άγκιστρο του κώδικα χειρισμού φόρμας και πριν από το τέλος αυτού του group κλείστε τη σύνδεση με τη βάση δεδομένων.
- Θα χρησιμοποιήσουμε δύο μεταβλητές στις οποίες θα τοποθετήσουμε τα δύο στοιχεία που έρχονται μέσω $_POST. Παρατηρήστε ότι χρησιμοποιείται η συνάρτηση mysql_real_escape_string() για να φιλτράρουμε την είσοδο, ώστε να μην είναι επιρρεπής η σελίδα μας σε κακόβουλες χρήσεις, όπως SQL injection. Παρατηρήστε ότι το password περνάει μέσω της συνάρτησης sha1(), ώστε να αποθηκεύσουμε το κρυπτογραφημένο hash του κωδικού στη βάση δεδομένων και όχι του ίδιου του κωδικού. Δηλαδή αν ο χρήστης δώσει 'user123' ως κωδικό, δε θα αποθηκευτεί αυτό αλλά η τιμή “20f0b8dcb99e90675f13687f07d5de0e22a26bfa” η οποία είναι η sha1() τιμή. Από την sha1() τιμή δε μπορούμε να βρούμε τον κωδικό user123 (είναι συνάρτηση μιας κατεύθυνσης (one-way-function)).

```
$username=mysql_real_escape_string($_POST['username']);
$password=sha1(mysql_real_escape_string($_POST['password']));
```

- Τοποθετήστε στο παραπάνω κομμάτι κώδικα την κλήση της συνάρτησης trim() στις μεταβλητές που έρχονται μέσω $_POST ώστε αν ο χρήστης πληκτρολογήσει π.χ. “ XXXX “ (δηλαδή κενά πριν και μετά) αυτά να απομακρυνθούν.
- Το πρώτο στοιχείο που θα πρέπει να ελέγξουμε είναι η ύπαρξη ή όχι του συγκεκριμένου χρήστη. Αυτό θα επιτευχθεί με το να δημιουργήσουμε την παρακάτω συνάρτηση μέσα στο αρχείο database_functions.inc. Η παρακάτω συνάρτηση επιστρέφει 1 αν υπάρχει έστω ένας χρήστης με αυτό το όνομα, διαφορετικά (αν δεν υπάρχει χρήστης με αυτό το username) επιστρέφει 0

```
function usernameTaken($username){
  $sql = "select username from users where username = '$username'";
  $result = mysql_query($sql);
 return ((mysql_numrows($result) > 0)?1:0);
}
```

- Στο signup.php λοιπόν ελέγχουμε με μια δομή if την τιμή που επιστρέφει η συνάρτηση usernameTaken και αν είναι 1 τότε εμφανίζουμε το αντίστοιχο μήνυμα στο χρήστη και σταματάει η εκτέλεση διαφορετικά (else) συνεχίζει η εκτέλεση κανονικά.
- Στη συνέχεια κατασκευάστε τη συνάρτηση μέσα στο database_functions.inc

```
function addNewUser($username, $password){}
```

η οποία θα κάνει insert στη βάση δεδομένων στον αντίστοιχο πίνακα το όνομα χρήστη και τον κωδικό. Προσέξετε ότι επειδή έχουμε και άλλες στήλες στον αντίστοιχο πίνακα, για την ώρα θα τοποθετούμε ενδεικτικές τιμές (κάποια πεδία δε μπορούν να είναι NULL-βάλτε εσείς ότι τιμές θέλετε στο SQL insert). Η συνάρτηση αυτή θα επιστρέφει 1 αν γίνει με επιτυχία η εισαγωγή, διαφορετικά θα επιστρέφει 0.

- Τοποθετήστε το παρακάτω κομμάτι κώδικα μέσα στο else του προηγούμενου βήματος, το οποίο αναφέρει αν έγινε με επιτυχία ή όχι η προσθήκη του χρήστη:

```
/* Lets add a user */
if (addNewUser($username,$password))
{       echo "Ο χρήστης $username προστέθηκε με επιτυχία";       }
else
{       echo "Παρουσιάστηκε σφάλμα κατά την προσθήκη";}
```

- Επιβεβαιώστε την ορθή λειτουργία της σελίδας με το να δημιουργήσετε μέσω της ανωτέρω φόρμας ένα χρήστη και να δείτε το μήνυμα επιτυχίας (ότι ο χρήστης προστέθηκε με επιτυχία).
- Επιβεβαιώστε μέσω phpmyadmin ότι έχει εισαχθεί μια καταχώρηση στον πίνακα users με το όνομα χρήστη που έχετε δώσει.

## Ταυτοποίηση χρήστη (login)

1. Δημιουργήστε ένα χρήστη αν δεν το έχετε κάνει ήδη με username=user και κωδικό =user123
2. Η φόρμα login κάνει POST στο αρχείο login.php
3. Κατασκευάστε το αρχείο login.php με το να αντιγράψετε το αρχείο template.php στο αρχείο login.php. Τοποθετήστε μέσα στο κεντρικό τμήμα τις ετικέτες για τον κώδικα php που θα γράψετε προσεχώς.
4. Ανοίξτε τη σύνδεση με τη βάση δεδομένων αμέσως μετά το άγκιστρο του κώδικα χειρισμού φόρμας και πριν από το τέλος αυτού του group κλείστε τη σύνδεση με τη βάση δεδομένων.
5. Δημιουργήστε τις μεταβλητές username και password ακριβώς όπως στον κώδικα εγγραφής νέου χρήστη.
6. Κατασκευάστε μια συνάρτηση με όνομα η οποία θα κάνει select από τον πίνακα users τα πεδία που το username=$username και το password=$password. Αν βρεθεί καταχώρηση με αυτά τα στοιχεία, τότε το ζευγάρι (username,password) είναι έγκυρο οπότε η συνάρτηση θα πρέπει να επιστρέψει 1 (=επιτυχία) διαφορετικά θα πρέπει να επιστρέφει 0.

```
function confirmUser($username, $password){}
```

7. Χρησιμοποιήστε την παρακάτω δομή if/else για το login

```
/* Verify User*/
if (confirmUser($username,$password))
{
echo "Καλώς ορίσατε";
}
else
{
echo "Παρουσιάστηκε σφάλμα κατά την είσοδό";
}
```

8. Επιβεβαιώστε την ορθή λειτουργία με το να κάνετε login με ένα όνομα χρήστη και έναν κωδικό, ο οποίος δεν υπάρχει και στη συνέχεια με ένα όνομα χρήστη και κωδικό που υπάρχει.

## Ταυτοποίηση χρήστη (login) και διατήρηση της κατάστασης ταυτοποίησης (sessions)

1. Το πρόβλημα με τον προηγούμενή κατάσταση είναι ότι μπορεί ο χρήστης να κάνει login αλλά αυτό δεν επηρεάζει καθόλου τις υπόλοιπες σελίδες, αφού μόλις πατηθεί κάποιος άλλος δεσμός η κατάσταση της σύνδεσης χάνεται. Θα πρέπει λοιπόν να χρησιμοποιήσουμε μια τεχνική για να διατηρούμε την κατάσταση της σύνδεσης. Αυτό θα επιτευχθεί μέσω των Sessions.
2. Για να χρησιμοποιήσουμε τα sessions θα πρέπει στην αρχή κάθε σελίδας, πριν στείλουμε το οποιοδήποτε HTML να τα ενεργοποιήσουμε. Για να γίνει αυτό θα πρέπει να δημιουργήσουμε ένα αρχείο κενό με όνομα phpheaders.php και στην πρώτη γραμμή κάθε php αρχείου (εκτός από τα php αρχεία των συναρτήσεων και των αρχείων που γίνονται require/include) να κάνουμε include αυτό το αρχείο ως

```
<?php require(“phpheaders.php”); ?>
```

3. Μέσα στο αρχείο αυτό τοποθετήστε την κλήση της συνάρτησης session_start()
4. Επισκεφτείτε το site και επιβεβαιώστε ότι λειτουργεί σωστά (δηλαδή δεν εμφανίζεται κάποιο πρόβλημα)
5. Ενεργοποιώντας τα sessions αυτόματα δημιουργείται ένας πίνακας $_SESSION στον οποίο τοποθετούμε κάποιες τιμές. Οι τιμές που τοποθετούμε δεν είναι προσβάσιμες στον πελάτη και δε μπορεί να τις δει. Μπορεί να δει μόνο τον αριθμό του $_SESSION. Οπότε, σε αυτόν τον πίνακα μπορούμε να τοποθετήσουμε ότι θέλουμε.
6. Στον πίνακα SESSIONS θέλουμε να τοποθετήσουμε το όνομα χρήστη που έχει κάνει login και το επίπεδο χρήστη που είναι.
7. Δημιουργήστε μια συνάρτηση findUserLevel η οποία θα καλείται εφόσον ο χρήστης έχει κάνει login και θα επιστρέφει το userlevel του συγκεκριμένου username.

```
function findUserLevel($username)
```

8. Την τιμή userlevel θα την τοποθετήσετε στη μεταβλητή $_SESSION['userlevel'] ενώ την τιμή του $username θα την τοποθετήσετε στη μεταβλητή $_SESSION['username']. Ασφαλώς, αυτές οι εκχωρήσεις θα πρέπει να γίνουν στον κλάδο του if που έχει γίνει με επιτυχία το login.
9. Το πρώτο στοιχείο που θα πρέπει να κάνουμε, είναι ΑΝ έχουμε κάνει με επιτυχία login, δηλαδή αν υπάρχει η μεταβλητή $_SESSION['username'] τότε δε πρέπει να εμφανίζεται η φόρμα login, αλλά να μας αναφέρει 'Έχετε συνδεθεί ως ... Αποσύνδεση'.

10. Ανοίξτε το αρχείο left_column.html το οποίο περιέχει τη φόρμα login. Βρείτε το σημείο που βρίσκεται η φόρμα (δηλαδή μέσα στο τμήμα login).
11. Κατασκευάστε μια δομή if /else η οποία θα κοιτάει αν έχει τεθεί η μεταβλητή username (isset()) και αν ναι τότε ΔΕ θα εκτυπώνει τη φόρμα. Ασφαλώς η δομή θα πρέπει να βρίσκεται μέσα σε php tags.
12. Αν δεν υπάρχει αυτή η μεταβλητή, τότε θα εμφανίζει τη φόρμα.
13. Επιβεβαιώστε ότι αν κάνετε login εξαφανίζεται η φόρμα και μένει κενό το πλαίσιο (αφού δεν έχουμε ακόμη τοποθετήσει κανένα κώδικα).
14. Προσθέστε το κείμενο στην κατάσταση που έχει ανιχνευτεί ότι υπάρχει το $_SESSION['username'] το μήνυμα **Έχετε συνδεθεί ως <$username>. Αποσύνδεση.** Το πλήρες όνομα θα το εξάγουμε από τη βάση. Η λέξη αποσύνδεση θα είναι δεσμός προς το αρχείο logout.php
15. Κατασκευάστε το αρχείο logout.php αντιγράφοντας το αρχείο template.php. Μέσα στο κυρίως σώμα γράψτε το κείμενο "Η αποσύνδεση έγινε με επιτυχία". Το κείμενο αυτό θα εμφανίζεται αν ΔΕΝ υπάρχει η μεταβλητή $_SESSION['username']. Επίσης, σε αυτό το αρχείο θα πρέπει να τοποθετηθεί κάτω από το require(phpheaders.php) ο κώδικας php που θα καταστρέφει το session και τις μεταβλητές.

Συγκεκριμένα θα εκτελούνται οι συναρτήσεις:

**session_unset();**

**session_destroy();**

**session_start();**

**session_regenerate_id(); //για να δημιουργηθεί ένα καινούργιο session**

16. Επιβεβαιώστε την ορθή λειτουργία της σελίδας login, και της logout
17. Θα διαπιστώσετε ότι έχει ένα μικρό πρόβλημα. Το πρόβλημα είναι ότι πρώτα εμφανίζεται η φόρμα login και μετά γίνεται ο έλεγχος στο login.php αν είναι σωστό το όνομα χρήστη και ο κωδικός. Αυτό μπορεί να μπερδέψει το χρήστη και να νομίσει ότι δεν έχει κάνει login και να συμπληρώσει τα στοιχεία της φόρμας πάλι. Ένα work-arround είναι να εμφανίζουμε τη φόρμα μόνο αν δεν έχει τιμή το $_POST['username'] (empty($_POST['username']).Τροποποιήστε κατάλληλα τον κώδικα στο left_column.html ώστε η φόρμα να εμφανίζεται μόνο αν είναι κενό το παραπάνω πεδίο. Αν δεν είναι κενό, τότε δε θα εμφανίζεται, αφού σημαίνει ότι ο χρήστης είναι στη διαδικασία login. Ασφαλώς, μια πιο σωστή αντιμετώπιση είναι να κάνουμε τον έλεγχο login και να θέτουμε το $_SESSION στις πρώτες γραμμές του login.php και στη συνέχεια (αφού δηλαδή γίνει ο έλεγχος και τεθεί η μεταβλητή $_SESSION['username'] να γίνεται include το αρχείο left_column.php.

==> Επιλέξτε από το προηγούμενο βήμα όποια μέθοδο θέλετε, κατά προτίμηση να μπει ο έλεγχος στις πρώτες γραμμές του login.php (αυτό θα μας βοηθήσει στα Cookies παρακάτω)

## Ταυτοποίηση χρήστη (login) και μόνιμη διατήρηση της κατάστασης ταυτοποίησης (sessions) με χρήση μπισκότων (cookies)

1. Αφού επιβεβαιώσετε ότι λειτουργούν όλα σωστά ως αυτό το σημείο, δηλαδή μπορείτε να κάνετε login/logout το επόμενο στοιχείο είναι να χρησιμοποιήσουμε cookies.
2. Τροποποιήστε τη φόρμα login στο αρχείο left_column ώστε να περιέχει και ένα checkbox με τίτλο 'remember me' ως εξής:

**<label> Remember Me </label><input id="remember_me" name="remember_me" type="checkbox" value="1" />**

3. Τροποποιήστε κατάλληλα το CSS ώστε να εμφανίζεται όλη η φόρμα μέσα στο πλαίσιο (αυξήστε το ύψος του login)

4. Αν ο χρήστης ενεργοποιήσει το κουμπί remember me, τότε θα μεταφερθεί μια μεταβλητή remember_me με τιμή 1 στο $_POST. Πρέπει λοιπόν στο login.php, μόλις επιβεβαιώσουμε ότι ο χρήστης έχει ταυτοποιηθεί ορθά, να στείλουμε ένα cookie στο πελάτη με τα στοιχεία της σύνδεσης για μελλοντική χρήση.
5. Πρέπει στο login.php να έχουμε τον έλεγχο στις πρώτες γραμμές (πριν το doctype) ως εξής:

```
<?php
include_once("site_functions.php");
db_open();
$username=mysql_real_escape_string(trim($_POST['username']));
$password=sha1(mysql_real_escape_string(trim($_POST['password'])));
        /* Verify User*/
        if (confirmUser($username,$password))
        {
 $_SESSION['userlevel']=findUserLevel($username);
  $_SESSION['username']=$username;
        }
?>
```

6. επειδή θα έχουμε όμως ένα σφάλμα λόγω του ότι κάνουμε require τα site_functions.php σε δύο σημεία, θα πρέπει στο αρχείο left_column.php να αλλάξουμε το require σε include_once() (δηλαδή αν έχει ήδη χρησιμοποιηθεί αυτό το αρχείο, να αγνοηθεί αυτή η εντολή) ως εξής:

```
<?php include_once("site_functions.php"); ?>
```

7. Επιβεβαιώστε ότι λειτουργεί σωστά το login/logout
8. Μέσα στη δομή if που εκτελείται αν είναι σωστό το ζευγάρι $username,$password θα εξεταστεί αν ο χρήστης έχει επιλέξει να σταλεί cookie. Για να το κάνουμε αυτό θα δώσουμε αρχικά:

```
if(isset($_POST['remember_me']){$remember=mysql_real_escape_string($_POST['remember_
me']);}
```

9. Στη συνέχεια μέσα στη δομή if (confirmUser($username,$password)) θα εξετάσουμε την τιμή που έχει $remember. Αν είναι 1 τότε θα στείλουμε cookie διαφορετικά δε θα κάνουμε τίποτα:

```
if ($remember==1) {
setcookie('auth','yes',time()+3600);
setcookie('uid',"$uid",time()+3600);
setcookie('username',"$username",time()+3600);
setcookie('password',"$password",time()+3600);
}
```

10. Επίσης, όταν ο χρήστης κάνει logout θα πρέπει να σβήνεται το cookie. Θα πρέπει λοιπόν στις πρώτες γραμμές του logout.php να τοποθετήσετε το παρακάτω κομμάτι κώδικα.

```
setcookie('username','',0);
setcookie('password','',0);
unset($_COOKIE);
```

11. Το επόμενο βήμα που θα πρέπει να κάνουμε, είναι κάθε φορά που επισκέπτεται ο χρήστης τη σελίδα **και ΔΕΝ υπάρχει η μεταβλητή $_SESSION['username'] αλλά υπάρχει η**

**$_COOKIE['username']** να κάνουμε έλεγχο στη βάση δεδομένων αν το ζευγάρι $_COOKIE['username'] και $_COOKIE['password'] είναι έγκυρα. Ανοίξτε λοιπόν το αρχείο phpheaders.php . Ο έλεγχος θα πρέπει να γίνεται, γιατί ο χρήστης θα μπορούσε να κατασκευάσει μόνος του ένα cookie με μια οποιαδήποτε τιμή στο username και μια οποιαδήποτε τιμή στο password. Θα πρέπει να ρωτήσουμε τη βάση δεδομένων λοιπόν αν είναι έγκυρο ζευγάρι αυτό που μας έρχεται μέσω cookie. Επίσης, επειδή ο χρήστης μπορεί να τροποποιήσει τις μεταβλητές username και password θα πρέπει αυτές να φιλτράρονται μέσω της συνάρτησης addslashes():

```
if(!isset($_SESSION['username']) && isset($_COOKIE['username']))
{
include_once("site_functions.php");
$username=addslashes($_COOKIE['username']);
$password=addslashes($_COOKIE['password']);

db_open();

if (confirmUser($username,$password))
{
        $_SESSION['userlevel']=findUserLevel($username);
        $_SESSION['username']=$username;
}
else //Cookie is invalid-Must be removed
{
                setcookie('username','$username',time()-3600);



        setcookie('password','$password',time()-3600);
                unset($_COOKIE);
}

db_close();
}
```

12. Επιβεβαιώστε ότι λειτουργεί σωστά το login/logout με τα cookies με το να κλείσετε το φυλλομετρητή ενώ έχετε κάνει login και να επισκεφτείτε τη σελίδα http://localhost/lab09 και να διαπιστώσετε αν σας εμφανίζεται το μήνυμα ότι έχετε συνδεθεί.