



ΑΛΕΞΑΝΔΡΕΙΟ Τ. Ε. Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ  
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



Πτυχιακή εργασία

# Κατασκευή διαδικτυακής εφαρμογής αδειών-ρεπό για τη Διεύθυνση Προσωπικού

Των φοιτητών:  
Ζησόπουλο Θωμά  
ΑΜ: 02/1994  
Γεωργιάδη Ιωάννη  
ΑΜ: 02/1996

Επιβλέπων Καθηγητής  
Δασυγένης Μηνάς

Θεσσαλονίκη 2010

## ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ.....	5
ΠΕΡΙΛΗΨΗ.....	6
ΠΕΡΙΛΗΨΗ ΣΤΑ ΑΓΓΛΙΚΑ .....	6
ΕΙΣΑΓΩΓΗ .....	7
1ο ΚΕΦΑΛΑΙΟ .....	8
1.1 Εισαγωγή .....	9
1.2 Λίγα λόγια για την PHP και την MySQL.....	9
1.2.1 Τι είναι η PHP;.....	9
1.2.2 Τι είναι η MySQL;.....	9
1.2.3 Πλεονεκτήματα της PHP .....	10
1.2.4 Πλεονεκτήματα της MySQL.....	11
1.3 Επισκόπηση-Παράδειγμα σύνταξης γλώσσας php .....	12
2ο ΚΕΦΑΛΑΙΟ .....	37
2.1 Εισαγωγή .....	38
2.2 Καταγραφή των πινάκων της βάσης. ....	38
2.3 Πίνακες καταγραφής στατιστικών στοιχείων .....	43
2.4 Φόρμες εισαγωγής δεδομένων.....	44
2.5 Φόρμες Διαχειριστή.....	53
2.6 Λοιπές φόρμες.....	60
2.7 Επίλογος .....	64
3ο ΚΕΦΑΛΑΙΟ .....	65
3.1 Εισαγωγή .....	66
3.2 Η δομή των εταιρειών.....	66
3.3 Οργανόγραμμα.....	67
3.4 Ο χρήστης administrator-διαχειριστής.....	68
3.5 Ροή των αδειών .....	68
3.6 Λοιπές δυνατότητες του συστήματος .....	69
3.7 Επίλογος .....	70
4ο ΚΕΦΑΛΑΙΟ .....	71
4.1 Προβλήματα που αντιμετωπιστήκαν .....	72
4.1.1 Εισαγωγή .....	72
4.1.2 Προβλήματα που αφορούν στο σύνολο της εργασίας .....	72
4.1.3 Προβλήματα που αφορούν στο κώδικα .....	73
4.2 Ασφάλεια εφαρμογής .....	73

4.2.1	Εισαγωγή .....	73
4.2.2	Εισαγωγή των χρηστών στην εφαρμογή.....	73
4.2.3	Ορθή συμπλήρωση των ημερομηνιών αίτησης αδειών. ....	74
4.2.4	Ασφάλεια των επιλογών στο μενού του διαχειριστή.....	74
4.3	Επίλογος .....	76
ΠΑΡΑΡΤΗΜΑΤΑ.....		77
ΠΑΡΑΡΤΗΜΑ Α .....		78
ΠΑΡΑΡΤΗΜΑ Β .....		91
ΠΑΡΑΡΤΗΜΑ Γ.....		96
ΠΑΡΑΡΤΗΜΑ Δ .....		107

## ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ – ΣΧΗΜΑΤΩΝ

Εικόνα 1. 1 – Η φόρμα παραγγελίας .....	13
Εικόνα 1. 2 – Η φόρμα αποτελεσμάτων .....	16
Εικόνα 1. 3 – Βελτιωμένη φόρμα αποτελεσμάτων.....	17
Εικόνα 1. 4 – Εμφάνιση στοιχείων παραγγελίας.....	18
Εικόνα 1. 5 – Εμφάνιση στοιχείων και συνολικού ποσού παραγγελίας.....	21
Εικόνα 1. 6 – Εμφάνιση κόστους αποστολής προϊόντων .....	25
Εικόνα 1. 7 – Οι functions AddSlashes() και StripSlashes() .....	31
Εικόνα 1. 8 – Απλό αποτέλεσμα εφαρμογής PHP κώδικα .....	33
Εικόνα 1. 9 – Δημιουργία πίνακα μέσω της HTML .....	35
Εικόνα 1. 10 – Παράδειγμα χρήσης function msg() .....	36
Εικόνα 2. 1 – Δημιουργία πίνακα members .....	38
Εικόνα 2. 2 – Δημιουργία πίνακα vacation type.....	39
Εικόνα 2. 3 – Δημιουργία πίνακα departments .....	39
Εικόνα 2. 4 – Δημιουργία πίνακα employee .....	40
Εικόνα 2. 5 – Δημιουργία πίνακα vacation .....	40
Εικόνα 2. 6 – Δημιουργία πίνακα groups.....	41
Εικόνα 2. 7 – Δημιουργία πίνακα adeies .....	41
Εικόνα 2. 8 – Δημιουργία πίνακα adeies backup.....	43
Εικόνα 2. 9 – Φόρμα καταχώρησης άδειας .....	45
Εικόνα 2. 10 – Φόρμα επεξεργασίας λογαριασμού .....	47
Εικόνα 2. 11 – Φόρμα προσωπικού ιστορικού αδειών .....	48
Εικόνα 2. 12 – Φόρμα ιστορικού υπαλλήλων .....	499
Εικόνα 2. 13 – Φόρμα έγκρισης άδειας.....	50
Εικόνα 2. 14 – Φόρμα επεξεργασίας χρηστών .....	50
Εικόνα 2. 15 – Φόρμα επεξεργασίας χρηστών 2 .....	52
Εικόνα 2. 16 – Μήνυμα λάθους .....	52
Εικόνα 2. 17 – Φόρμα επεξεργασίας διευθύνσεων.....	53
Εικόνα 2. 18 – Φόρμα διαχείρισης της εφαρμογής.....	55
Εικόνα 2. 19 – Φόρμα καταχώρησης χρηστών.....	58
Εικόνα 2. 20 – Φόρμα εισόδου .....	60

Εικόνα 2. 21 – Μήνυμα λάθους κατα την είσοδο .....	60
Εικόνα 4. 1 – Στοιχεία Χρηστών .....	73
Εικόνα 4. 2 – Μήνυμα λάθους .....	74
Εικόνα 4. 3 – Μήνυμα επιβεβαίωσης.....	74

## **ΠΡΟΛΟΓΟΣ**

Το θέμα αυτής της πτυχιακής αν και απαιτητικό είναι ιδιαίτερα ενδιαφέρον. Με βάση αυτό το κριτήριο επιλέχθηκε καθώς και με βάση το γεγονός ότι αποτελεί σύγχρονο κομμάτι της επιστήμης της πληροφορικής. Οι δυσκολίες που αντιμετωπίσαμε και η επίλυσή τους μας βοήθησαν να αποκτήσουμε ορισμένες γνώσεις στη σχεδίαση βάσεων και διαδικτυακών εφαρμογών τις οποίες είτε αγνοούσαμε τελείως είτε δεν γνωρίζαμε επαρκώς. Σε συνεργασία με τον επιβλέπον καθηγητή μας καθώς και με την εταιρία που θα χρησιμοποιήσει το πρόγραμμα, υλοποιήσαμε την εφαρμογή και το αποτέλεσμα κρίνεται, τουλάχιστον από την πλευρά μας, επιτυχημένο.

## ΠΕΡΙΛΗΨΗ

Η πτυχιακή εργασία αυτή έχει θέμα την δημιουργία ενός προγράμματος αίτησης- παραχώρησης αδειών κάποιας εταιρίας. Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε η γλώσσα προγραμματισμού PHP και το εργαλείο διαχείρισης βάσεων δεδομένων MySQL .

Η εταιρία που θα χρησιμοποιήσει το προγράμματα είναι υπαρκτή και αποτελείται ουσιαστικά από τρεις διαφορετικές εταιρίες, μία μητρική και δύο θυγατρικές. Έχει τέσσερα επίπεδα απασχόλησης μέσω των οποίων περνάει μια άδεια προκειμένου να εγκριθεί. Οι υπάλληλοι, ανάλογα με το επίπεδο απασχόλησής τους , ζητούν κάποιον από τους διαθέσιμους τύπους άδειας και περιμένουν την έγκρισή της. Εάν η άδεια εγκριθεί, η εφαρμογή ενημερώνει τον υπάλληλο και καταχωρεί την άδειά του στη βάση δεδομένων. Έαν απορριφθεί ενημερώνεται αντίστοιχα με την αιτιολογία απόρριψης.

Τέλος η εφαρμογή δίνει τη δυνατότητα τήρησης στατιστικών στοιχείων για όλες τις άδειες που έχουν καταχωρηθεί. Κάτι τέτοιο δεν ήταν δυνατό να γίνει με το υπάρχον σύστημα που διέθετε η εταιρία. Επίσης δίνει τη δυνατότητα σε κάθε υπάλληλο ξεχωριστά να ελέγχει τα δικά του στατιστικά στοιχεία καθώς και την πορεία των αδειών που έχει ζητήσει.

## ΠΕΡΙΛΗΨΗ ΣΤΑ ΑΓΓΛΙΚΑ

The subject of this thesis is the writing of a day-off program for a specific company, using the PHP language and the MySQL database management tool. The company that will use the implementation is actually separated into three different companies, one parent company and two subsidiary companies. There are four avocation levels in every company, through which all the day-off application should be approved in order to be given. All employees, according to their avocation level, can ask for one of the available day-off types and they wait for it to be approved. The implementation informs the employees if their application is approved or declined. It also gives the reason for the application was declined.

Finally, the implementation gives the opportunity to the manager to keep statistical records for all the day-off types and all the employees. Additionally, all the employees are able to check their own statistical records and the status of any day-off being asked.

## ΕΙΣΑΓΩΓΗ

Στόχος της πτυχιακής εργασίας αυτής είναι η υλοποίηση ενός on-line προγράμματος αίτησης – χορήγησης αδειών για μία υπαρκτή εταιρία. Σκοπός του είναι η αντικατάσταση του υπάρχοντος προγράμματος, βάση του οποίου γινόταν διακίνηση αιτήσεων μεταξύ των υπαλλήλων. Ως αποτέλεσμα της υλοποίησης και εφαρμογής του προγράμματος είναι η κατά πολύ ταχύτερη και ασφαλέστερη διαδικασία αίτησης , και κατ' επέκταση έγκρισης, μίας άδειας που ζητήθηκε από κάποιο υπάλληλο.

Στο κείμενο που ακολουθεί δίνεται η πορεία υλοποίησης της πτυχιακής εργασίας καθώς και τα εργαλεία που χρησιμοποιήθηκαν για την εκπόνησή της. Το κείμενο είναι χωρισμένο σε τέσσερα κεφάλαια που αντιστοιχούν στα τέσσερα βασικά μέρη της εφαρμογής.

Στο πρώτο κεφάλαιο αναφέρονται τα εργαλεία που χρησιμοποιήθηκαν για να υλοποιηθεί η εφαρμογή. Αναλύεται διεξοδικά η γλώσσα προγραμματισμού που χρησιμοποιήθηκε καθώς και βασικές αρχές της. Επίσης αναλύεται το εργαλείο βάσεων δεδομένων που υποστηρίζει τους πίνακες της εφαρμογής και τέλος αναλύονται τα χαρακτηριστικά του server στον οποίο έχει στηθεί η εφαρμογή.

Στο δεύτερο κεφάλαιο περιγράφεται βηματικά ο σχεδιασμός των βάσεων δεδομένων καθώς και οι φόρμες εισαγωγής των δεδομένων στην εφαρμογή. Παρουσιάζονται και επεξηγούνται κάποιοι βασικοί κώδικες υλοποίησης ώστε να κατανοηθεί ευκολότερα η δομή της εφαρμογής και απεικονίζονται όλοι οι πίνακες δεδομένων του συνόλου της βάσης.

Στο τρίτο κεφάλαιο αναλύεται η γενική δομή της εταιρίας. Περιγράφονται τα επίπεδα απασχόλησης των υπαλλήλων και δίνεται σχηματικά το οργανόγραμμα της εταιρίας. Περιγράφονται τα δικαιώματά των υπαλλήλων ανάλογα με τη θέση που κατέχουν στην εταιρία καθώς και ο ρόλος και οι αρμοδιότητες του διαχειριστή της εφαρμογής. Τέλος αναλύεται η γενική ροή της εφαρμογής και ο τρόπος αίτησης και χορήγησης αδειών και αναφέρονται κάποιες επιπλέον δυνατότητες που παρέχει το πρόγραμμα στους υπαλλήλους.

Στο τέταρτο και τελευταίο κεφάλαιο καταγράφονται τα προβλήματα που παρουσιάστηκαν κατά τη διάρκεια υλοποίησης της πτυχιακής εργασίας και περιγράφεται ο τρόπος που επιλύθηκαν ή παρακάμφθηκαν. Τα προβλήματα αυτά χωρίζονται σε δύο κατηγορίες ανάλογα με τη φύση τους. Επίσης, στο κεφάλαιο αυτό αναφέρονται όλες οι ενέργειες που αφορούν την ασφάλεια και σωστή λειτουργία της εφαρμογής. Οι ενέργειες αυτές έχουν άμεση σχέση με τα προβλήματα που αντιμετωπίστηκαν καθώς η προσπάθεια προστασίας της εφαρμογής δημιούργησε διάφορα ζητήματα που χρειάστηκαν επίλυση.

## **1ο ΚΕΦΑΛΑΙΟ**

Ανάλυση των εργαλείων που χρησιμοποιήθηκαν



## 1.1 Εισαγωγή

Στο κεφάλαιο αυτό θα γίνει μια γενική αναφορά στα εργαλεία που χρησιμοποιήθηκαν κατά την εκπόνηση της εργασίας καθώς και τα πλεονεκτήματά τους. Επίσης θα δοθεί ένα βασικό και αρκετά διεξοδικό παράδειγμα για την κατανόηση της γλώσσας PHP, που αποτελεί και το βασικό εργαλείο της εφαρμογής.

## 1.2 Λίγα λόγια για την PHP και την MySQL

### 1.2.1 Τι είναι η PHP;

Η PHP είναι μια γλώσσα script από την πλευρά του διακομιστή, σχεδιασμένη ειδικά για το Web. Μέσα σε μια HTML σελίδα μπορείτε να ενσωματώσετε PHP κώδικα, που θα εκτελείται κάθε φορά που θα επισκέπτεστε τη σελίδα. Ο PHP κώδικας μεταφράζεται στο Web διακομιστή και δημιουργεί HTML ή άλλη έξοδο που θα δει ο επισκέπτης.

Η PHP δημιουργήθηκε το 1994 και ήταν αρχικά η δουλειά ενός ατόμου, του **Rasmus Lerdorf**. Υιοθετήθηκε και από άλλα ταλαντούχα άτομα και έχει περάσει από τρεις βασικές εκδόσεις. Τον Ιανουάριο του 2001 ήταν σε χρήση σχεδόν σε πέντε εκατομμύρια τομείς παγκόσμια και αυτός ο αριθμός μεγαλώνει γρήγορα.

Η PHP είναι ένα προϊόν ανοιχτού κώδικα. Υπάρχει πρόσβαση στον κώδικα προέλευσης. Μπορεί να χρησιμοποιηθεί, να αλλαχθεί και να αναδιανεμηθεί, χωρίς χρέωση.

Η PHP αρχικά σήμαινε *Personal Home Page* ( προσωπική αρχική σελίδα ), αλλά άλλαξε σύμφωνα με την σύμβαση GNU και τώρα σημαίνει *PHP Hypertext Preprocessor* ( προεπεξεργαστής κειμένου PHP ). Η τρέχουσα βασική έκδοση της PHP είναι η 4. Αυτή η έκδοση έχει μερικές μεγάλες βελτιώσεις στη γλώσσα.

### 1.2.2 Τι είναι η MySQL;

Η MySQL είναι ένα πολύ γρήγορο και δυνατό σύστημα *διαχείρισης βάσεων δεδομένων*. Μια βάση δεδομένων επιτρέπει να αποθηκεύονται, να αναζητούνται, να ταξινομούνται και να ανακαλούνται τα δεδομένα αποτελεσματικά. Ο MySQL διακομιστής ελέγχει την πρόσβαση στα δεδομένα, για να μπορούν να δουλεύουν πολλοί χρήστες ταυτόχρονα, για να παρέχει γρήγορη πρόσβαση και να διασφαλίζει ότι μόνο πιστοποιημένοι χρήστες μπορούν να έχουν πρόσβαση.

Συνεπώς η MySQL είναι ένας πολυνηματικός διακομιστής πολλαπλών χρηστών. Χρησιμοποιεί την SQL (Structured Query Language ) την τυπική γλώσσα ερωτημάτων για βάσεις δεδομένων, παγκόσμια. Η MySQL είναι διαθέσιμη από το 1996 αλλά η ιστορία της ξεκινά από το 1979 .

### 1.2.3 Πλεονεκτήματα της PHP

Κάποιοι από τους βασικούς ανταγωνιστές της PHP είναι ο Perl, Microsoft Active Server Pages (ASP), Java Server Pages ( JSP ) και Allaire Cold Fusion.

Σε σύγκριση με αυτά τα προϊόντα, η PHP έχει πολλά πλεονεκτήματα όπως :

- Υψηλή απόδοση
- Διασυνδέσεις με πολλά διαφορετικά συστήματα βάσεων δεδομένων
- Ενσωματωμένες βιβλιοθήκες για πολλές συνηθισμένες Web διαδικασίες
- Χαμηλό κόστος
- Ευκολία μάθησης και χρήσης
- Μεταφερσιμότητα
- Διαθεσιμότητα του κώδικα προέλευσης

Ακολουθεί μια πιο λεπτομερή περιγραφή αυτών των πλεονεκτημάτων.

#### Απόδοση

Η PHP είναι πολύ αποτελεσματική. Με ένα φθηνό διακομιστή μπορούν να εξυπηρετηθούν εκατομμύρια επισκέψεων καθημερινά. Οι δοκιμές που δημοσιεύθηκαν από την Zend Technologies ( <http://www.zend.com>), δείχνουν ότι η PHP ξεπερνά τους ανταγωνιστές της. Ολοκλήρωση με Βάσεις δεδομένων

Η PHP έχει εγγενείς συνδέσεις για πολλά συστήματα βάσεων δεδομένων. Εκτός από την MySQL, μπορεί κάποιος να συνδεθεί κατευθείαν με τις βάσεις δεδομένων PostgreSQL, mSQL, Oracle, dbm, filePro, Informix, InterBase, Sybase, μεταξύ άλλων.

Χρησιμοποιώντας το *Open Database Connectivity Standard (ODBC)* μπορεί κάποιος να συνδεθεί σε οποιαδήποτε βάση δεδομένων παρέχει ένα πρόγραμμα οδήγησης ODBC. Αυτό περιλαμβάνει και τα προϊόντα της Microsoft products, μεταξύ άλλων.

#### Ενσωματωμένες Βιβλιοθήκες

Επειδή η PHP σχεδιάστηκε για να χρησιμοποιείται στο Web, έχει πολλές ενσωματωμένες βιβλιοθήκες, που εκτελούν πολλές χρήσιμες λειτουργίες σχετικές με το Web. Μπορεί κάποιος να δημιουργηθεί εικόνες GIF δυναμικά, να συνδεθεί με άλλες υπηρεσίες δικτύων, να στείλει ηλεκτρονικά μηνύματα, να δουλέψει με cookies και να δημιουργήσει PDF έγγραφα: όλα αυτά με λίγες γραμμές κώδικα.

#### Κόστος

Η PHP είναι δωρεάν. Μπορεί κάποιος να κατεβάσει την τελευταία έκδοση από το <http://www.php.net>, χωρίς χρέωση.

#### Εκμάθηση της PHP

Η σύνταξη της PHP βασίζεται σε άλλες γλώσσες προγραμματισμού, βασικά στην C και στην Perl.

#### Μεταφερσιμότητα

Η PHP είναι διαθέσιμη για πολλά λειτουργικά συστήματα. Μπορεί να γραφεί κώδικας PHP για δωρεάν συστήματα τύπου Unix, όπως LINUX και FreeBSD, για εμπορικές εκδόσεις του UNIX, όπως το Solaris και το IRIX ή για διαφορετικές εκδόσεις των Microsoft Windows. Ο κώδικας συνήθως θα δουλεύει χωρίς αλλαγές στα συστήματα που τρέχουν την PHP.

### **Κώδικας προέλευσης**

Υπάρχει πρόσβαση στον κώδικα προέλευσης της PHP. Αντίθετα με εμπορικά, κλειστά προγράμματα, αν υπάρχει κάτι που πρέπει να αλλάχθει ή να προστεθεί στη γλώσσα, μπορεί να γίνει. Δεν χρειάζεται να περιμένει κάποιος τον κατασκευαστή να εμφανίσει διορθώσεις. Δεν υπάρχει ανησυχία αν ο κατασκευαστής θα σταματήσει να υπάρχει ή αν θα σταματήσει να υποστηρίζει το προϊόν.

#### **1.2.4 Πλεονεκτήματα της MySQL**

Μερικοί από τους κύριους ανταγωνιστές της MySQL είναι οι PostgreSQL , Microsoft SQL και Oracle. Η MySQL έχει πολλά πλεονεκτήματα, όπως χαμηλό κόστος, εύκολη διαμόρφωση και μάθηση και ο κώδικας προέλευσης είναι διαθέσιμος.

#### **Απόδοση**

Η MySQL είναι χωρίς αμφιβολία γρήγορη. Μπορεί κανείς να δει την σελίδα δοκιμών <http://web.mysql.com/benchmark.html> . Πολλές από αυτές τις δοκιμές δείχνουν ότι η MySQL είναι αρκετά πιο γρήγορη από τον ανταγωνισμό.

#### **Χαμηλό κόστος**

Η MySQL είναι διαθέσιμη δωρεάν, με άδεια ανοικτού κώδικα (Open Source) ή με χαμηλό κόστος, αν παρθεί εμπορική άδεια, αν απαιτείται από κάποια εφαρμογή .

#### **Ευκολία Χρήσης**

Οι περισσότερες μοντέρνες βάσεις δεδομένων χρησιμοποιούν SQL. Αν χρησιμοποιηθεί ένα άλλο σύστημα διαχείρισης βάσεων δεδομένων δεν θα υπάρξει πρόβλημα προσαρμογής σε αυτό.

#### **Μεταφερσιμότητα**

Η MySQL μπορεί να χρησιμοποιηθεί σε πολλά διαφορετικά συστήματα Unix όπως επίσης και στα Microsoft Windows .

#### **Κώδικας Προέλευσης**

Όπως και με την PHP , μπορεί εύκολα να τροποποιηθεί ο κώδικας προέλευσης της MySQL.

### 1.3 Επισκόπηση-Παράδειγμα σύνταξης γλώσσας php

Σε αυτό το μέρος θα γίνει μια γρήγορη επισκόπηση της σύνταξη της PHP και των εντολών της γλώσσας. Θα δωθεί ένα παράδειγμα μιας online φόρμας παραγγελιών για την εκμάθηση χρήσης των μεταβλητών , των τελεστών και των παραστάσεων της PHP και θα καλυφθούν οι τύποι των μεταβλητών . Θα δειχθεί πως είναι δυνατόν να προσπελαθούν οι μεταβλητές μιας φόρμας και ποιός ο χειρισμός τους ώστε να βρεθούν τα σύνολα και οι φόροι σε μια παραγγελία πελάτη. Μετά θα αναπτυχθεί η εφαρμογή χρησιμοποιώντας το PHP script για να επικυρωθούν τα δεδομένα εισόδου. Θα εξεταστούν οι λογικές μεταβλητές και θα δωθούν παραδείγματα για το πώς να χρησιμοποιείται η εντολή if else. Τέλος θα εξερευνηθούν οι βρόγχοι γράφοντας PHP για να δημιουργηθούν μεταβαλλόμενοι πίνακες.

Τα βασικά θέματα αυτού του μέρους είναι :

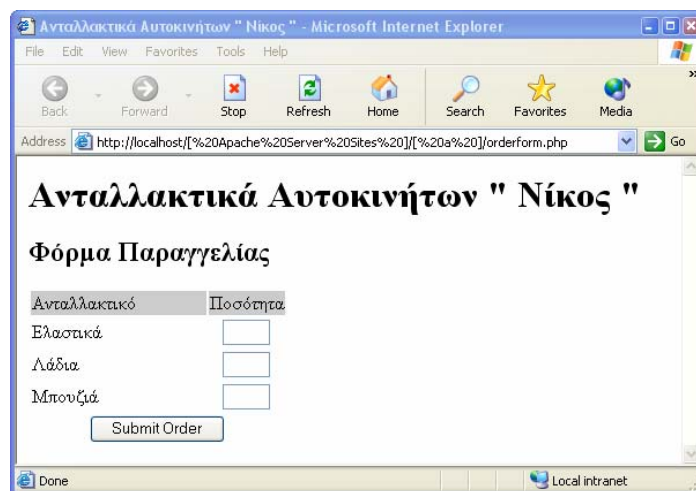
- Ενσωμάτωση PHP στην HTML
- Προσθέτοντας δυναμικά περιεχόμενα
- Προσπέλαση μεταβλητών φορμών
- Αναγνωριστικά
- Μεταβλητές που δηλώνονται από τον χρήστη
- Τύποι μεταβλητών
- Εκχώρηση τιμών σε μεταβλητές
- Αποφάσεις με τις if , else
- Επανάληψη : βρόγχου while , do και for
- Πίνακες
- Χειρισμός συμβολοσειρών
- Επαναχρησιμοποίηση κώδικα και σύνταξη συναρτήσεων

### **Εφαρμογή δείγμα : Τα ανταλλακτικά του Νίκου**

Μια από τις πιο συνηθισμένες εφαρμογές οποιασδήποτε γλώσσας script από την πλευρά του διακομιστή είναι η επεξεργασία HTML φορμών. Η εφαρμογή θα αρχίσει από την φόρμα παραγγελίας από το κατάστημα ανταλλακτικών του Νίκου, μια υποθετική εταιρία με ανταλλακτικά.

#### **Η φόρμα παραγγελία**

Αυτή τη στιγμή έχει φτιαχθεί η φόρμα παραγγελιών του Νίκου για τα ανταλλακτικά που πουλάει ο Νίκος. Η φόρμα φαίνεται στην εικόνα 1. 1 είναι μια σχετικά απλή φόρμα παραγγελίας παρόμοια με αυτές που υπάρχουν στις σελίδες του Internet. Το πρώτο πράγμα που θα ήθελε ο Νίκος να ξέρει , είναι , τι παρήγγειλε ο πελάτης του και ποιό είναι το σύνολο της παραγγελίας του καθώς και ο φόρος θα πληρωθεί.



**Εικόνα 1. 1 – Η φόρμα παραγγελία**

Η αρχική φόρμα παραγγελίας του Νίκου καταγράφει μόνο προϊόντα και ποσότητες.

### Κώδικας του αρχείου orderform.html

```
<html> <head> <title>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος &quot;</title>
<meta http-equiv=»Content-Type» content=»text/html; Charset=windows-1253»>
</head>
<body> <h1>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος
&quot;</h1> <h2>Φόρμα Παραγγελίας</h2> <form
action=»processorder. Php» method=post> <table border=0>
<tr bgcolor=#cccccc>
    <td width=150>Ανταλλακτικό</td>
    <td width=15>Ποσότητα</td> </tr> <tr>
    <td>Ελαστικά</td> <td align=center><input type=»text” name=»tireqty” size=3
maxlength=3></td> </tr> <tr> <td>Λάδια</td>
    <td align=center><input type=»text” name=»oilqty” size=3 maxlength=3></td>
</tr> <tr>
    <td>Μπουζιά</td>
    <td align=center><input type=»text” name=»sparkqty” size=3 maxlength=3></td>
</tr> <tr>
    <td colspan=2 align=center><input type=submit value=»Submit Order”></td>
</tr>
    </table>
    </form>
    </body>
    </html>
```

Υπάρχουν δυο σημαντικά πράγματα στον κώδικα. Το πρώτο πράγμα που παρατηρείται είναι ότι έχει οριστεί η ενέργεια της φόρμας να είναι το όνομα του PHP script που θα επεξεργαστεί την παραγγελία του πελάτη. Γενικά η τιμή της ιδιότητας ACTION είναι το URL που θα φορτωθεί όταν ο χρήστης πατήσει το κουμπί (submit). Τα δεδομένα που έχει πληκτρολογήσει ο χρήστης στη φόρμα θα σταλούν σε αυτό το URL, μέσω της μεθόδου που καθορίζεται στην ιδιότητα METHOD. Το δεύτερο σημαντικό πράγμα είναι τα ονόματα των πεδίων της φόρμας – tireqty, oilqty και sparkqty. Αυτά τα ονόματα θα χρησιμοποιηθούν ξανά στο PHP script. Εξ αιτίας αυτού, είναι σημαντικό να δοθούν στα πεδία της φόρμας ονόματα που θα αντιπροσωπεύουν τις έννοιες της φόρμας για να μπορούν να απομνημονευτούν εύκολα.

### Επεξεργασία της φόρμας

Για να γίνει η επεξεργασία της φόρμας θα πρέπει να δημιουργηθεί το script που αναφέρθηκε στην ιδιότητα ACTION της ετικέτας FORM, που ονομάζεται *processorder.Php*. Ανοίγοντας το Notepad πρέπει να δημιουργηθεί το παραπάνω αρχείο και εισαχθεί ο παρακάτω κώδικας:

---

```
<html> <head> <title>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος &quot;</title>  
<meta http-equiv=»Content-Type» content=»text/html; Charset=windows-1253»>  
</head> <body> <h1>Ανταλλακτικά Αυτοκινήτων</h1> <h2>Αποτελέσματα  
Παραγγελίας</h2> </body> </html>
```

---

Μέχρι στιγμής έχει γραφθεί ο απλός κώδικας HTML.

### Ενσωμάτωση PHP στην HTML

Κάτω από την επικεφαλίδα <h2> του αρχείου πρέπει να πληκτρολογηθεί η παρακάτω γραμμή:

---

```
<?  
    Echo “<p> Η παραγγελία επεξεργάστηκε. ”;  
?>
```

---

Στη συνέχεια πρέπει να αποθηκευτεί το αρχείο και να φορτωθεί στον Browser το αρχείο με την φόρμα παραγγελίας, να συμπληρωθεί η παραγγελία και να πατηθεί το Submit. Το αποτέλεσμα θα πρέπει να είναι παρόμοιο με την εικόνα 1. 2 .

Το κείμενο που πέρασε στην δομή της PHP εμφανίζεται στον browser. Βλέποντας τον κώδικα μέσα από τον browser, φαίνεται το παρακάτω.

```
<html> <head> <title>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος &quot;</title>  
<meta http-equiv=»Content-Type» content=»text/html; Char set=windows-1253»>  
</head> <body> <h1>Ανταλλακτικά Αυτοκινήτων</h1> <h2>Αποτελέσματα  
Παραγγελίας</h2> <p>Η παραγγελία επεξεργάστηκε.</p> </body> </html>
```

Δεν φαίνεται καθόλου ο απλός κώδικας. Αυτό συμβαίνει επειδή ο μεταφραστής έχει τρέξει το script και το έχει αντικαταστήσει με την έξοδο από το script. Αυτό σημαίνει ότι, από την PHP μπορεί να παραχθεί καθαρή HTML, που φαίνεται σε οποιονδήποτε browser – με αλλά λόγια, ο browser του χρήστη δεν χρειάζεται να καταλαβαίνει PHP. Αυτό δείχνει την ιδέα των script από την πλευρά του διακομιστή. Η PHP έχει μεταφραστεί και εκτελεστεί στο Web διακομιστή, ξεχωριστά από την JavaScript και άλλες τεχνολογίες από την πλευρά του πελάτη που μεταφράζονται και εκτελούνται σε ένα Web Browser στον υπολογιστή του χρήστη.

Ο κώδικας που υπάρχει τώρα σε αυτό το αρχείο αποτελείται από τέσσερα πράγματα

- HTML
- Ετικέτες PHP
- Εντολές PHP

### Χρησιμοποιώντας τις PHP ετικέτες

Ο κώδικας PHP στο προηγούμενο παράδειγμα, αρχίζει με <? Και τελειώνει με ?> Αυτό είναι παρόμοιο με τις ετικέτες HTML, επειδή όλες ξεκινάνε με ένα σύμβολο μικρότερο “<” και τελειώνουν με ένα σύμβολο μεγαλύτερο “>”. Αυτά τα σύμβολα ονομάζονται PHP ετικέτες και λένε στον Web διακομιστή πού ξεκινά ο PHP

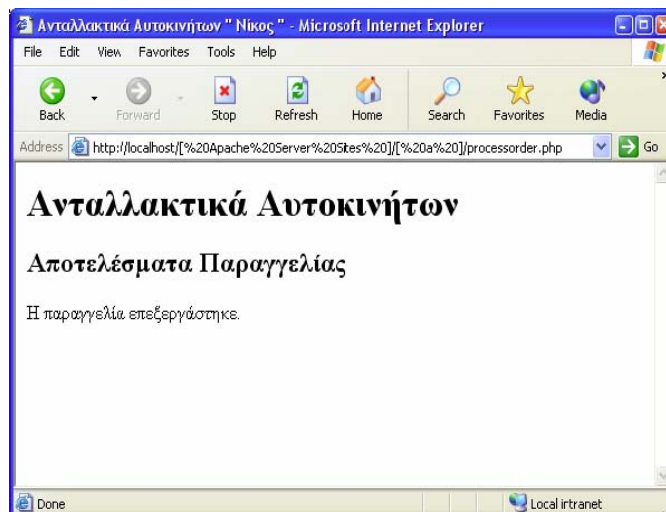
κώδικας και που τελειώνει. Το κενό μεταξύ των ετικετών θα μεταφραστεί σαν PHP. Το κείμενο έξω από αυτές τις ετικέτες θα αντιμετωπιστεί σαν κανονική HTML. Οι ετικέτες PHP επιτρέπουν τη διαφυγή από την HTML.

#### PHP εντολές

Κατευθύνουν τον μεταφραστή της PHP βάζοντας τις εντολές PHP μεταξύ της αρχικής και τελικής ετικέτας. Σε αυτό το παράδειγμα χρησιμοποιείται μόνο ένας τύπος εντολής:

**echo "<rp> Η παραγγελία επεξεργάστηκε.";**

Χρησιμοποιώντας την δομή echo παράγεται ένα πολύ απλό αποτέλεσμα: τυπώνεται η συμβολοσειρά που έχει περάσει σε αυτήν, στον browser. Στην εικόνα 1. 2 φαίνεται το αποτέλεσμα που είναι το κείμενο «Η παραγγελία επεξεργάστηκε. », που εμφανίζεται στο παράθυρο του browser.



Εικόνα 1. 2 – Η φόρμα αποτελεσμάτων



Όπως φαίνεται στο παράδειγμα, εμφανίζεται ένα ερωτηματικό στο τέλος της εντολής echo. Αυτό χρησιμοποιείται για να ξεχωρίζει εντολές της PHP, όπως μια τελεία χρησιμοποιείται για να ξεχωρίζει τις προτάσεις της γλώσσας. Είναι συνηθισμένο συντακτικό λάθος εάν παραληφθεί το ερωτηματικό.

#### Προσθέτοντας δυναμικά περιεχόμενα

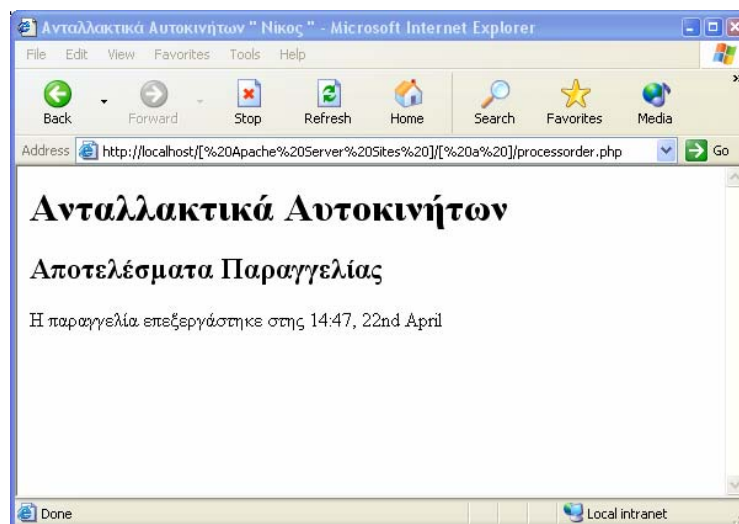
Μέχρι στιγμής, δεν έχει χρησιμοποιηθεί PHP για να γίνει κάτι που δεν έχει γίνει με απλή HTML. Ο λόγος χρήσης της γλώσσας script από την πλευρά του διακομιστή είναι για να μπορούν να παρέχονται δυναμικά περιεχόμενα στους χρήστες μιας τοποθεσίας. Αυτή είναι μια σημαντική εφαρμογή, επειδή τα περιεχόμενα αυτά αλλάζουν σύμφωνα με τις ανάγκες ενός χρήστη ή κάνουν τους χρήστες να επανέρχονται στην τοποθεσία. Η PHP το καθιστά αυτό ιδιαίτερα εύκολο.

Ξεκινώντας με κάτι απλό:

Αντικατάσταση της PHP στο processorder.php με τον παρακάτω κώδικα :

```
<? echo "<p>Η παραγγελία επεξεργάστηκε στις "; echo date("Hi, jS F");  
echo "<br>"; ?>
```

Σε αυτόν τον κώδικα χρησιμοποιείται η ενσωματωμένη συνάρτηση date() της PHP, για να ειπωθεί στον πελάτη η ημερομηνία και η ώρα που έγινε η επεξεργασία της παραγγελίας το . Αυτό είναι διαφορετικό κάθε φορά που θα τρέχει το script.



Εικόνα 1.3 – Βελτιωμένη φόρμα αποτελεσμάτων

### Προσπέλαση μεταβλητών φορμών

Η ουσία χρήσης μιας φόρμας παραγγελίας είναι η συλλογή της παραγγελίας του πελάτη. Στην PHP είναι πολύ εύκολη η συλλογή πληροφοριών που πληκτρολόγησε ο πελάτης. Μέσα στο PHP script, υπάρχει πρόσβαση σε κάθε ένα από τα πεδία της φόρμας σαν μια μεταβλητή με το ίδιο όνομα με το πεδίο της φόρμας. Παράδειγμα:

Προσθέτονται οι παρακάτω γραμμές στο κάτω μέρος του PHP script :

```
echo "<ρ>Η παραγγελία σας περιέχει τα εξής"; echo "<br>"; echo $tireqty. "
ελαστικά<br>"; echo $oilqty. " μπουκάλια λάδι<br>"; echo $sparkqty. "
μπουζιά<br>";
```

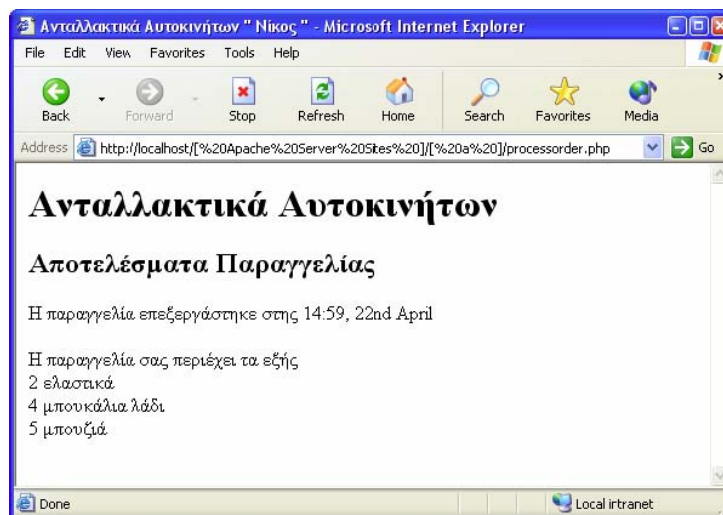
Συμπληρώνοντας την φόρμα πάλι στον browser, η έξοδος του script μοιάζει με αυτό που φαίνεται στην Εικόνα 1.4. Οι πραγματικές τιμές που θα φαίνονται φυσικά, θα εξαρτώνται από αυτό που πληκτρολογήθηκε στη φόρμα.

### Μεταβλητές φορμών

Τα δεδομένα του script καταλήγουν σε PHP

μεταβλητές.

Μπορούν εύκολα να αναγνωριστούν τα ονόματα των μεταβλητών στην PHP επειδή ξεκινούν με το \$ .



Εικόνα 1.4 – Εμφάνιση στοιχείων παραγγελίας

### Συνένωση συμβολοσειρών

Στο script, χρησιμοποιείται

το echo για να δωθεί η τιμή που πληκτρολόγησε ο χρήστης σε κάθε ένα από τα πεδία της φόρμας, ακολουθούμενη από κάποιο επεξηγηματικό κείμενο. Προσέχοντας καλύτερα την εντολή echo, φαίνεται ότι το όνομα της μεταβλητής και το επόμενο κείμενο έχουν μια τελεία μεταξύ τους ως εξής :

```
echo $tireqty. " ελαστικά<br>";
```

Αυτός είναι ο τελεστής συνένωσης συμβολοσειρών που χρησιμοποιείται για να προσθέτει συμβολοσειρές (κομμάτια κειμένου) μεταξύ τους.

### Μεταβλητές και σταθερές

Η μεταβλητή και η συμβολοσειρά που συνενώνονται σε κάθε μια από τις εντολές echo, είναι διαφορετικοί τύποι πραγμάτων. Οι μεταβλητές είναι σύμβολα για δεδομένα. Η συμβολοσειρά είναι η ίδια δεδομένα. Όταν χρησιμοποιούνται απλά δεδομένα σε ένα πρόγραμμα ,αυτό ονομάζεται σταθερά για να ξεχωρίζεται από

μια μεταβλητή. Η `$tireqty` είναι μια μεταβλητή, ένα σύμβολο που αντιπροσωπεύει τα δεδομένα που πληκτρολόγησε ο χρήστης. Από την άλλη το “ελαστικά” είναι μια σταθερά και μπορεί να χρησιμοποιηθεί ακριβώς όπως είναι.

#### Αναγνωριστικά

Τα αναγνωριστικά είναι τα ονόματα των μεταβλητών. Υπάρχουν μερικοί απλοί κανόνες για τα αναγνωριστικά :

- Τα αναγνωριστικά μπορεί να είναι οποιοδήποτε μήκους και μπορούν να αποτελούνται από γράμματα, αριθμούς, χαρακτήρες υπογράμμισης.
- Τα αναγνωριστικά δεν μπορούν να ξεκινούν με ψηφίου.
- Στην PHP, τα αναγνωριστικά είναι ευαίσθητα σε κεφαλαία /πεζά. Το `$tireqty` δεν είναι ίδιο με το `$Tireqty`.
- Τα αναγνωριστικά για τις μεταβλητές μπορούν να έχουν το ίδιο όνομα με μια ενσωματωμένη συνάρτηση.

#### Εκχώρηση τιμών σε μεταβλητές

Δίνονται τιμές σε μεταβλητές χρησιμοποιώντας τον τελεστή εκχώρησης, το ίσων (=). Στην τοποθεσία του ‘Νίκου’, δουλεύοντας για τον συνολικό αριθμό των ανταλλακτικών που παραγγέλθηκαν και το συνολικό ποσό, προσθέτονται οι παρακάτω γραμμές στο κάτω μέρος του script.

```
$totalqty=0; $totalamount =0.00;
```

Κάθε μια από αυτές τις γραμμές δημιουργεί μια μεταβλητή και της δίνει μια σταθερή τιμή.

#### Τύποι μεταβλητών

Η PHP υποστηρίζει τους παρακάτω τύπους δεδομένων:

- Integer (ακέραιος)
- Double(διπλής ακρίβειας)
- String(συμβολοσειράς)
- Array(πίνακα)
- Object(αντικείμενα)

## Τελεστές

Οι τελεστές είναι σύμβολα που μπορούν να χρησιμοποιηθούν για τον χειρισμό τιμών και μεταβλητών εκτελώντας μια λειτουργία πάνω τους.

### Αριθμητικοί τελεστές

Τελεστής	Όνομα	Παράδειγμα
+	Πρόσθεσης	$a+b$
-	Αφαίρεσης	$a-b$
*	Πολλαπλασιασμού	$a*b$
/	Διαίρεσης	$a/b$
%	Υπολοίπου	$a\%b$

### Τελεστές συμβολοσειρών

$a = \text{"Νίκος"}; b = \text{"Ανταλλακτικά"}; results = a.b;$

Η μεταβλητή  $results$  περιέχει τώρα την συμβολοσειρά "Νίκος Ανταλλακτικά".

### Τελεστές σύγκρισης

Οι τελεστές σύγκρισης χρησιμοποιούνται για να συγκριθούν δύο τιμές. Οι παραστάσεις που χρησιμοποιούν αυτούς τους τελεστές επιστρέφουν λογικές τιμές True – False, ανάλογα με το αποτέλεσμα σύγκρισης.

Τελεστής	Όνομα	Παράδειγμα
==	Ισότητα	$a == b$
===	Ταυτότητα	$a === b$
!=	Άνισο	$a != b$
<>	Άνισο	$a <> b$
<	Μικρότερο	$a < b$
>	Μεγαλύτερο	$a > b$
<=	Μικρότερο ή ίσον	$a <= b$
>=	Μεγαλύτερο ή ίσον	$a >= b$

### Λογικοί τελεστές

Οι λογικοί τελεστές χρησιμοποιούνται για να συνδυάζουν τα αποτελέσματα των λογικών συνθηκών. Για παράδειγμα μπορεί να ενδιαφέρεται ο χρήστης σε κάποια περίπτωση αν η τιμή μιας μεταβλητής, της  $a$  είναι μεταξύ 0 και 100. Θα πρέπει να ελεγχθούν οι συνθήκες  $a >= 0$  και  $a <= 100$ , χρησιμοποιώντας τον τελεστή AND ως εξής:  $a > 0 \ \&\& \ a <= 100$ .

Τελεστής	Όνομα	Χρήση	Αποτέλεσμα
!	NOT	$!b$	Επιστρέφει true αν το $b$ είναι false και το αντίστροφο
&&	AND	$a \ \&\& \ b$	Επιστρέφει true αν τα $a$ και $b$ είναι και τα δύο true
			διαφορετικά επιστρέφει false
	OR	$a \    \ b$	Επιστρέφει true αν το $a$ ή το $b$ είναι true/
And	AND	$a \ \text{and} \ b$	Το ίδιο με το &&
Or	OR	$a \ \text{or} \ b$	Το ίδιο με το

### Ο τελεστής μη εμφάνισης λαθών

Ο τελεστής @ μπορεί να χρησιμοποιηθεί εμπρός από οποιαδήποτε πρόταση . \$a = @(57/0); Χωρίς τον τελεστή @ , αυτή η γραμμή θα δημιουργήσει μια προειδοποίηση για διαίρεση με το 0. Με τον τελεστή , το λάθος δεν εμφανίζεται. Αν δεν εμφανίζονται τα λάθη θα πρέπει να συντάχθει κατάλληλος κώδικας για να χειρίζεται τα τυχόν σφάλματα στην εφαρμογή.

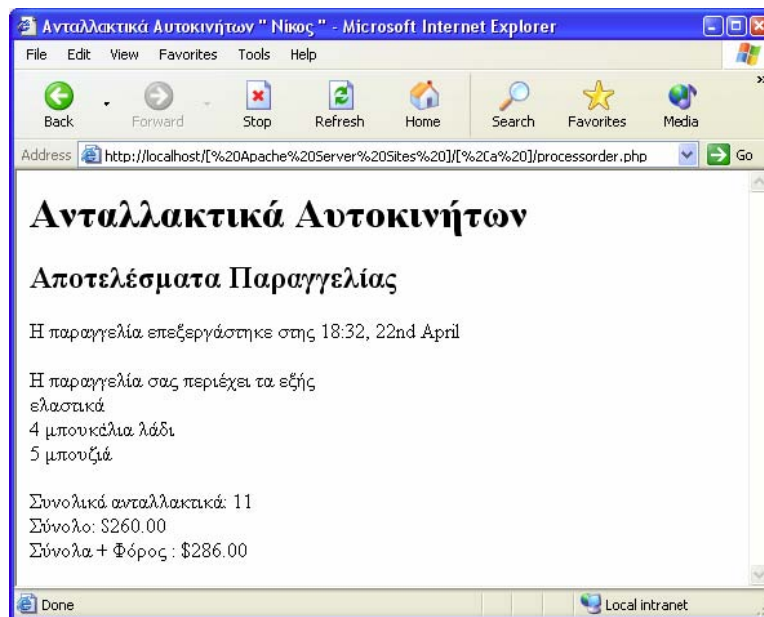
### Χρήση τελεστών: Βρίσκοντας τα σύνολα μια φόρμας

Γνωρίζοντας πλέον για τους τελεστές της PHP , θα υπολογιστούν τα σύνολα και ο φόρος στην φόρμα παραγγελίας του «Νίκου».

Πρόσθεση του παρακάτω κώδικα στο κάτω μέρος του PHP script :

```
$TYREPRICE=100; $OILPRICE=10; $SPARKPRICE= 4;
$totalqty = $tyreqty + $oilqty + $sparkqty; $totalamount = $tyreqty *
$TYREPRICE + $oilqty * $OILPRICE + $sparkqty *
$SPARKPRICE; echo "<br>\n"; echo "Συνολικά ανταλλακτικά: ". $totalqty.
"<br>\n"; echo "Σύνολο: $"; echo number_format($totalamount, 2); echo
"<br>\n";
$taxrate = 0. 10; // local sales tax is 10% $totalamount = $totalamount * (1
+ $taxrate); $totalamount = number_format($totalamount, 2); echo
"Σύνολο + Φόρος : $". $totalamount. "<br>\n";
```

Ανανεώνοντας το browser φαίνεται κάτι ανάλογο με την παρακάτω φόρμα:



Εικόνα 1. 5 - Εμφάνιση στοιχείων και συνολικού ποσού παραγγελίας

Όπως φαίνεται έχουν χρησιμοποιηθεί διαφορετικοί τελεστές σε αυτό το τμήμα του κώδικα. Έγινε χρήση της πρόσθεσης και του πολλαπλασιασμού για να βρεθούν τα ποσά και του τελεστή συνένωσης συμβολοσειρών για να διαμορφωθεί η έξοδος του browser. Χρησιμοποιήθηκε επίσης η συνάρτηση `number_format()` για να μορφοποιηθούν τα σύνολα ως συμβολοσειρές με δύο δεκαδικά ψηφία. Αυτή η συνάρτηση είναι από την βιβλιοθήκη `Math` της PHP.

### Αποφάσεις με συνθήκες υπό Όρους

Για λογική απόκριση στην είσοδο του χρήστη , ο κώδικας θα πρέπει να παίρνει λογικές αποφάσεις. Οι δομές που λένε στο πρόγραμμα μας να πάρει αποφάσεις ονομάζονται *εντολές υπό όρους*.

### Εντολές if

Μπορεί να χρησιμοποιηθεί μια εντολή `if` για τη λήψη μιας απόφασης. Θα πρέπει να δωθεί στην εντολή `if` μια συνθήκη για να μπορεί να χρησιμοποιηθεί. Αν η εντολή είναι `true` , θα εκτελεστεί το επόμενο τμήμα του κώδικα . Οι συνθήκες με εντολές `if` πρέπει να περιβάλλονται με παρενθέσεις. Όταν ο επισκέπτης δεν παραγγέλνει ανταλλακτικά , θα πρέπει να του εμφανίζεται μήνυμα ότι «Δεν παραγγείλατε τίποτα από την προηγούμενη σελίδα». Αυτό μπορεί να υλοποιηθεί εύκολα με την παρακάτω εντολή `if`:

```
if( $totalqty == 0 ) echo "Δεν παραγγείλατε τίποτα από την προηγούμενη σελίδα!<br>";
```

### Τμήματα κώδικα

Συνήθως υπάρχουν περισσότερες από μια εντολές που χρειάζεται να εκτελεστούν μέσα σε μια εντολή `if` . Δεν υπάρχει ανάγκη να τοποθετηθεί μια νέα εντολή `if` πριν από κάθε μια. Αντίθετα μπορούν να ομαδοποιούνται πολλές εντολές `if` σαν ένα μπλοκ. Για να δηλωθεί ένα τέτοιο τμήμα κώδικα πρέπει να περιλαμβάνεται μέσα σε άγκιστρα :

```
If ( $totalqty == 0 )  
{ echo "<font color=red>"; echo "Δεν παραγγείλατε τίποτα από την προηγούμενη σελίδα!<br>"; echo "</font>";  
}
```

Οι τρεις γραμμές κώδικα που περιβάλλονται στα άγκιστρα είναι τώρα ένα τμήμα κώδικα. Όταν η συνθήκη είναι `true` , εκτελούνται και οι τρεις γραμμές. Όταν είναι `false` αγνοούνται.

Εντολές else

Συνήθως πρέπει να αποφασιστεί, όχι μόνο αν πρέπει να εκτελεστεί μια ενέργεια αλλά επίσης ποιο σύνολο ενεργειών πρέπει να εκτελεστεί. Μια εντολή else επιτρέπει να οριστεί μια εναλλακτική ενέργεια, όταν η συνθήκη της εντολής if είναι false. Πρέπει να προειδοποιούνται οι πελάτες του Νίκου όταν δεν παραγγέλνουν τίποτα. Από την άλλη εάν κάνουν μια παραγγελία, αντί για μια προειδοποίηση, πρέπει να εμφανίζεται τι παρήγγειλαν.

```
if( $totalqty == 0 ) {
    echo "Δεν παραγγείλατε τίποτα από την προηγούμενη σελίδα!<br>"; }
else {
    echo $tireqty. " ελαστικά<br>";   echo $oilqty. " μπουκάλια λάδι<br>";
    echo $sparkqty. " μπουζί<br>";
}
```

Θα μπορούσαν να δημιουργηθούν πιο περίπλοκες λογικές διαδικασίες βάζοντας ένθετες τις εντολές if, την μια μέσα στην άλλη. Στον παρακάτω κώδικα, όχι μόνο θα εμφανιστεί το άθροισμα όταν η συνθήκη \$totalqty είναι αληθής, αλλά επίσης και κάθε γραμμή του αθροίσματος θα εμφανίζεται μόνο αν η δική της συνθήκη είναι αληθής.

```
if( $totalqty == 0 ) {
    echo "Δεν παραγγείλατε τίποτα από την προηγούμενη σελίδα!<br>"; }
else {
    if ( $tyreqty>0 ) echo $tireqty. "
        ελαστικά<br>"; if ( $oilqty>0 )
        echo $oilqty. " μπουκάλια λάδι<br>"; if ( $sparkqty>0 ) echo
        $sparkqty. " μπουζί<br>";
}
```

Εντολές else if

Για πολλές από τις αποφάσεις που παίρνονται, υπάρχουν περισσότερες από δύο επιλογές. Μπορεί να δημιουργηθεί μια σειρά από πολλές επιλογές χρησιμοποιώντας την εντολή else if. Η εντολή αυτή είναι συνδυασμός της else και της if. Παρέχοντας μια σειρά από συνθήκες, το πρόγραμμα μπορεί να ελέγχει κάθε μια, μέχρι να βρει αυτή που είναι αληθείς. Ο Νίκος δίνει μεγαλύτερη έκπτωση για μεγάλες παραγγελίες ελαστικών. Οι εκπτώσεις που δίνει είναι η εξής:

- Λιγότερα από 10 ελαστικά – χωρίς έκπτωση
- 10-49 ελαστικά – 5% έκπτωση
- 50-99 ελαστικά -10% έκπτωση
- 100 ή περισσότερα ελαστικά – 15% έκπτωση

Μπορεί να δημιουργηθεί κώδικας για να υπολογίζει την έκπτωση χρησιμοποιώντας συνθήκες και εντολές if και else if. Πρέπει να χρησιμοποιείται ο τελεστής AND (&&) για να συνδυαστούν δύο συνθήκες σε μία.

```
If ($tireqty <10) $discount = 0; else if ($tireqty >=10 && $tireqty <=49 ) $discount = 5; else if ($tireqty >=50 && $tireqty <=99 ) $discount = 10; else if ($tireqty >=100 ) $discount = 15;
```

#### Επανάληψη : Επαναλαμβάνοντας ενέργειες

Σε ένα πράγμα που οι υπολογιστές είναι πολύ καλοί είναι η αυτοματοποίηση επαναλαμβανόμενων διαδικασιών. Αν υπάρχει κάτι που πρέπει να γίνει με τον ίδιο τρόπο, πολλές φορές, μπορεί να χρησιμοποιηθεί ένας βρόγχος για την επανάληψη κάποιων μερών του προγράμματος.

Ο Νίκος θέλει ένα πίνακα που να εμφανίζει το κόστος μεταφοράς που θα προστεθεί στην παραγγελία ενός πελάτη. Με την μεταφορική εταιρία που χρησιμοποιεί ο Νίκος, το κόστος μεταφοράς εξαρτάται από την απόσταση που θα μεταφερθεί το πακέτο. Το κόστος μπορεί να βρεθεί με έναν απλό τύπο.

Θα πρέπει ο πίνακας με τα μεταφορικά να μοιάζει με τον πίνακα της Εικόνας 1. 6. Η λίστα δείχνει τον HTML που εμφανίζει αυτόν τον πίνακα. Φαίνεται ότι είναι μεγάλος και επαναλαμβανόμενος.

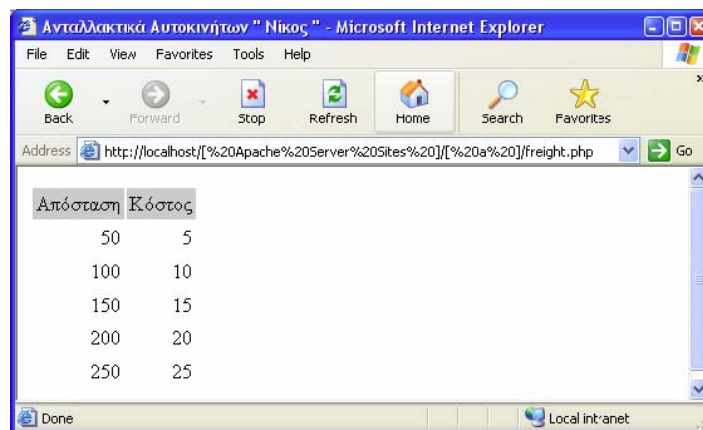
```
<html>
<head>
<title>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος &quot;</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
</head>
<body>
<table border = 0 cellpadding = 3>
<tr> <td bgcolor = "#CCCCCC" align = center>Απόσταση</td> <td bgcolor =
"#CCCCCC" align = center>Κόστος</td>
</tr>
<tr>
```



```

<td align = right>50</td>
  <td align = right>5</td> </tr> <tr>
  <td align = right>100</td>
  <td align = right>10</td> </tr> <tr>
  <td align = right>150</td>
  <td align = right>15</td> </tr> <tr>
  <td align = right>200</td>
  <td align = right>20</td> </tr> <tr>
  <td align = right>250</td>
  <td align = right>25</td> </tr> </table> </body>

```



Εικόνα 1. 6 – Εμφάνιση κόστους αποστολής προϊόντων

Θα βοηθούσε πολύ αν, αντί να ζητηθεί από ένα άτομο που βαριέται εύκολα και πρέπει να πληρωθεί για το χρόνο του να πληκτρολογήσει τον HTML κώδικα, να το κάνει ένας φθηνός και ακούραστος υπολογιστής. Οι εντολές βρόγχων λένε στην PHP να εκτελεί μια εντολή ή σύνολο εντολών, συνεχώς.

### Βρόγχοι While

Το απλούστερο είδος βρόγχων στην PHP είναι ο βρόγχος while. Όπως και μια εντολή if έτσι και αυτός βασίζεται σε μια συνθήκη. Η διαφορά μεταξύ ενός βρόγχου while και μιας εντολής if είναι ότι η εντολή if εκτελεί το επόμενο τμήμα του κώδικα μία φορά, εφόσον η συνθήκη είναι αληθής. Ο βρόγχος while εκτελεί το τμήμα επαναλαμβανόμενα, εφόσον η συνθήκη είναι αληθής. Γενικά χρησιμοποιούμε ένα βρόγχο while όταν δεν είναι γνωστό πόσες επαναλήψεις θα απαιτηθούν για να γίνει η συνθήκη αληθής. Αν απαιτείται σταθερός αριθμός επαναλήψεων, μπορεί να χρησιμοποιηθεί ένας βρόγχος for. Η βασική δομή ενός βρόγχου while είναι :

**while (condition) expression;**

Ο παρακάτω κώδικας θα εμφανίσει τους αριθμούς 1 έως το 5

```

$num
while ($num <= 5)
{ echo $num. "<br>"; $num++;
}

```

*Η εντολή \$num++; Ισοδυναμεί με \$num = \$num + 1 ; δηλαδή προσθέτει στον εαυτό του κάθε φορά το 1.*

Στην αρχή της κάθε επανάληψης, ελέγχει την συνθήκη. Αν είναι ψευδής, το τμήμα του δεν θα εκτελεστεί και ο βρόγχος θα τερματιστεί. Θα εκτελεστεί η αμέσως επόμενη εντολή μετά το βρόγχο.

Μπορεί να χρησιμοποιηθεί ένας βρόγχος while για να εμφανιστεί ο επαναλαμβανόμενο πίνακας μεταφορών της Εικόνας 1. 6 .

```
<html> <head> <title>Ανταλλακτικά Αυτοκινήτων &quot; Νίκος &quot;</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
</head> <body> <table border = 0 cellpadding = 3> <tr>
  <td bgcolor = "#CCCCCC" align = center>Απόσταση</td>
  <td bgcolor = "#CCCCCC" align = center>Κόστος</td> <? $distance =
50; while ($distance <= 250 ) {
  echo "<tr>\n <td align = right>$distance</td>\n"; echo " <td align =
right>".
  $distance / 10 . "</td>\n</tr>\n"; $distance += 50;
} ?> </table> </body> </html>
```

### Βρόγχοι for

Ο τρόπος που χρησιμοποιήθηκε ο βρόγχος while προηγουμένως είναι πολύ συνηθισμένος. Ορίζεται αρχικά ένας μετρητής. Πριν από κάθε επανάληψη, ελέγχεται ο μετρητής με μια συνθήκη. Στο τέλος της επανάληψης, τροποποιείται ο μετρητής. Μπορεί να γραφεί αυτό το στυλ βρόγχου με μια πιο συμπαγή μορφή, χρησιμοποιώντας ένα βρόγχο for. Η βασική δομή ενός βρόγχου for είναι :

```
for ( expression1; condition ; expresion2)
```

- Το expression1 εκτελείται μια φορά στην αρχή. Εδώ συνήθως ορίζουμε την αρχική τιμή ενός μετρητή.
- Το condition ελέγχεται πριν από κάθε επανάληψη. Εάν επιστρέψει false , η επόμενη επανάληψη θα σταματήσει. Εδώ συνήθως ελέγχουμε τον μετρητή ως προς ένα όριο.
- Το expression2 εκτελείται στο τέλος κάθε επανάληψης. Εδώ συνήθως ρυθμίζεται την τιμή του μετρητή

Μπορεί να ξαναγραφτεί ο βρόγχος while ως βρόχος for.

```
For ($distance =50 ; $distance <=250 ; $distance += 50) { echo
"<tr>\n <td align = right>$distance</td>\n"; echo " <td align =
right>". $distance / 10 . «</td>\n</tr>\n»; }
```

Και οι δύο εκδόσεις του while και του for είναι ισοδύναμες. Στη δεύτερη περίπτωση είναι λίγο πιο συμπαγής και εξοικονομούνται δύο γραμμές.

### Σπάζοντας μια δομή ελέγχου ή Script

Εάν χρειάζεται να σταματήσει η εκτέλεση ενός τμήματος του κώδικα, υπάρχουν τρεις προσεγγίσεις, ανάλογα με το αποτέλεσμα που πρέπει να επιτευχθεί. Αν πρέπει να σταματήσει η εκτέλεση ενός βρόγχου, μπορεί να χρησιμοποιηθεί η εντολή break. Για μεταπήδηση στην επόμενη επανάληψη του βρόγχου, μπορεί αντίθετα να χρησιμοποιηθεί η εντολή continue. Αν τέλος πρέπει να σταματήσει εξολοκλήρου το PHP script, πρέπει να χρησιμοποιηθεί η εντολή exit. Για παράδειγμα :

```
If ( $totalqty == 0 )
```

```
{ echo "Δεν συμπληρώσατε την παραγγελία σας στην προηγούμενη σελίδα";
  Exit;
}
```

Η κλήση για έξοδο εμποδίζει την PHP να εκτελέσει το υπόλοιπο script.

### Πίνακες

Ένας πίνακας (array) είναι μια μεταβλητή που περιέχει ένα σύνολο ή σειρά από τιμές. Ένας πίνακας μπορεί να έχει πολλά στοιχεία. Κάθε στοιχείο μπορεί να περιέχει μια τιμή, όπως κείμενο ή αριθμούς ή ένα άλλο πίνακα. Ένας πίνακας που περιέχει πίνακες, ονομάζεται πολυδιάστατος πίνακας. Η PHP υποστηρίζει πίνακες με αριθμητικούς δείκτες και συσχετιζόμενους πίνακες. Η λίστα προϊόντων του Νίκου είναι ένας πίνακας του παραδείγματος μας. Στην εικόνα 2.1 φαίνεται μια λίστα από τα τρία προϊόντα, αποθηκευμένα σε μορφή πίνακα και με μια μεταβλητή \$products, που περιέχει τις τρεις τιμές.

Αφού μπου οι πληροφορίες σε ένα πίνακα, μπορούν να γίνουν διάφορα χρήσιμα πράγματα με αυτόν. Χρησιμοποιώντας τους βρόγχους, μπορεί να εξοικονομηθεί κόπος εκτελώντας τις ίδιες ενέργειες σε κάθε τιμή του πίνακα. Με αυτόν τον τρόπο, με μια γραμμή κώδικα μπορούν να περάσουν όλες οι τιμές σε μια συνάρτηση. Οι τιμές που αποθηκεύονται σε έναν πίνακα ονομάζονται στοιχεία του πίνακα. Κάθε στοιχείο του πίνακα έχει ένα σχετικό δείκτη, που

Ελαστικά	Λάδια	Μπουζί
----------	-------	--------

Εικόνα 2.1

χρησιμοποιείται για πρόσβαση στα στοιχεία. Οι πίνακες στις περισσότερες γλώσσες προγραμματισμού, έχουν αριθμητικούς δείκτες. Η PHP

υποστηρίζει αυτούς τους τύπους πινάκων.

### Πίνακες με αριθμητικούς δείκτες

Για να δημιουργηθεί ο πίνακας της Εικόνας 2.1 χρησιμοποιείται η παρακάτω γραμμή κώδικα PHP:

```
$products = array ("Ελαστικά", "Λάδια", "Μπουζί");
```

Αυτό θα δημιουργήσει έναν πίνακα που ονομάζεται `products` και περιέχει τις τρεις αρχικές τιμές. «Ελαστικά , Λάδια , Μπουζί» .

#### Πρόσβαση στα περιεχόμενα του πίνακα

Για να υπάρχει πρόσβαση στα περιεχόμενα του πίνακα μιας μεταβλητής, χρησιμοποιείται το όνομά της. Πληκτρολογώντας:

```
$products[1] = "Μπαταρίες" ;
```

Εισάγεται στο πεδίο 1 του πίνακά μας την τιμή «Μπαταρίες». Αντιθέτως πληκτρολογώντας :

```
echo $products[1] ;
```

Θα τυπωθεί στην οθόνη η τιμή που περιέχεται μέσα στον πίνακα `products` που σε αυτή την περίπτωση είναι «Λάδια».

#### Χρησιμοποιώντας βρόγχους για πρόσβαση σε πίνακες

Επειδή ο πίνακας έχει δείκτη μια σειρά από αριθμούς, μπορεί να χρησιμοποιηθεί ένας βρόγχο `for` για πιο εύκολη εμφάνιση των περιεχομένων του :

```
for ( $i=0 ; $i <3 ; $i++ ) echo "$products[$i] " ;
```

Αυτός ο βρόγχος θα εμφανίσει τα στοιχεία του πίνακα, απαιτώντας λιγότερη πληκτρολόγηση.

#### Συσχετιζόμενοι πίνακες

Στον πίνακα προϊόντων η PHP δίνει σε κάθε στοιχείο έναν προκαθορισμένο δείκτη. Αυτό σημαίνει ότι το πρώτο στοιχείο που προστείνεται είναι το στοιχείο 0, το δεύτερο είναι το 1 κλπ. Η PHP υποστηρίζει επίσης συσχετιζόμενους πίνακες.

Σε ένα συσχετιζόμενο πίνακα, μπορεί να συσχετιστεί με κάθε τιμή ενός οποιοδήποτε πεδίου ή ενός δείκτη.

#### Αρχικοποιώντας ένα συσχετιζόμενο πίνακα

Ο παρακάτω κώδικας δημιουργεί ένα συσχετιζόμενο πίνακα, με τα ονόματα των προϊόντων ως πεδία και τις τιμές των προϊόντων ως τιμές.

```
$prices = array ("Ελαστικά"=>100, "Λάδια"=>10, "Μπουζί"=>4);
```

#### Πρόσβαση στα στοιχεία του πίνακα

Και πάλι μπορεί να υπάρχει πρόσβαση στα περιεχόμενα του πίνακα χρησιμοποιώντας το όνομα της μεταβλητής και ένα πεδίο, ώστε να μπορεί να υπάρχει πρόσβαση στις πληροφορίες που έχουν αποθηκευτεί στον πίνακα `prices` ως `$prices ["Ελαστικά"]`, `$prices ["Λάδια"]`, `$prices ["Μπουζί"]`.

Όπως και οι πίνακες με αριθμητικούς δείκτες, έτσι και οι συσχετιζόμενοι πίνακες μπορούν να δημιουργηθούν και να αρχικοποιηθούν, προσθέτοντας κάθε φορά ένα στοιχείο. Ο παρακάτω κώδικας θα δημιουργήσει τον ίδιο πίνακα \$prices. Αντί να δημιουργήσει ένα πίνακα με τρία στοιχεία, αυτός ο κώδικας, δημιουργεί ένα πίνακα μόνο με ένα στοιχείο και μετά προσθέτει δύο επιπλέον.

```
$prices = array("Ελαστικά"=>100); $prices["Λάδια"]=10; $prices["Μπουζί"]=4;
```

Εδώ είναι ένας άλλος, λίγο διαφορετικός, αλλά ισοδύναμος κώδικας. Σε αυτή την έκδοση δεν δημιουργείται άμεσα ένας πίνακας. Ο πίνακας δημιουργείται όταν προστεθεί το πρώτο στοιχείο σε αυτόν.

```
$prices["Ελαστικά"]=100; $prices["Λάδια"]=10; $prices["Μπουζί"]=4;
```

### **Μορφοποίηση συμβολοσειρών**

Αποκόπτοντας συμβολοσειρές :chop(), ltrim(), trim()

Το πρώτο βήμα για την μορφοποίηση είναι η αποκοπή των επιπλέον από μία συμβολοσειρά. Αν και αυτό δεν απαιτείται ποτέ, μπορεί να είναι χρήσιμο αν πρόκειται να αποθηκευτεί η συμβολοσειρά σε ένα αρχείο ή βάση δεδομένων ή αν πρόκειται να συγκριθεί με άλλες συμβολοσειρές.

```
$name = trim($name); $email =  
trim($email); $feedback =  
trim($feedback);
```

Η συνάρτηση trim() κόβει τον κενό χώρο από την αρχή και το τέλος της συμβολοσειράς και επιστρέφει την τελική συμβολοσειρά. Οι χαρακτήρες που αποκόπτεται είναι οι αλλαγές γραμμών και οι χαρακτήρες επιστροφής, οι οριζόντιοι και οι κατακόρυφοι στηλοθέτες, οι χαρακτήρες τέλους συμβολοσειράς και τα κενά.

Ανάλογα με το συγκεκριμένο σκοπό, μπορεί να χρησιμοποιηθεί τις συναρτήσεις ltrim() ή chop(). Είναι και οι δύο παρόμοιες με την trim(), παίρνοντας μια συμβολοσειρά ως παράμετρο και επιστρέφοντας την μορφοποιημένη συμβολοσειρά. Η διαφορά μεταξύ των τριών είναι ότι η trim() αφαιρεί τον κενό χώρο από την αρχή και το τέλος μιας συμβολοσειράς, η ltrim() μόνο από την αρχή και η chop() αφαιρεί τον κενό χώρο από το τέλος της.

Χρησιμοποιώντας HTML μορφοποίηση : Η συνάρτηση nl2br()

Η συνάρτηση nl2br() παίρνει μια συμβολοσειρά ως παράμετρο και αντικαθιστά όλες τις αλλαγές γραμμών με την ετικέτα της HTML <br>. Αυτό είναι χρήσιμο για να εμφανιστεί μια μεγάλη συμβολοσειρά στον browser.

Αλλάζοντας τα πεζά-κεφαλαία μιας συμβολοσειράς  
Μπορούμε επίσης να αλλάξουμε τα πεζό-κεφαλαία μιας συμβολοσειράς.

<i>Συνάρτηση</i>	<i>Περιγραφή</i>	<i>Χρήση</i> \$sub	<i>Τιμή</i> Καλημέρα παιδιά
Strtoupper()	Μετατρέπει την συμβολοσειρά σε κεφαλαία	Strtoupper(\$sub)	ΚΑΛΗΜΕΡΑ ΠΑΙΔΙΑ
Strtolower()	Μετατρέπει την συμβολοσειρά σε πεζά	strtolower(\$sub)	Καλημέρα παιδιά
Ucfirst()	Κάνει κεφαλαίο τον πρώτο χαρακτήρα της συμβολοσειράς αν είναι αλφαριθμητικός χαρακτήρας	ucfirst(\$sub)	Καλημέρα παιδιά
ucwords()	Κάνει κεφαλαίο τον πρώτο χαρακτήρα κάθε λέξης της συμβολοσειράς που αρχίζει με ένα αλφαριθμητικό χαρακτήρα	ucwords(\$sub)	Καλημέρα Παιδιά

#### Πίνακας συναρτήσεων πεζά-κεφαλαία

Μορφοποιώντας συμβολοσειρές για αποθήκευση: AddSlashes()  
και StripSlashes()

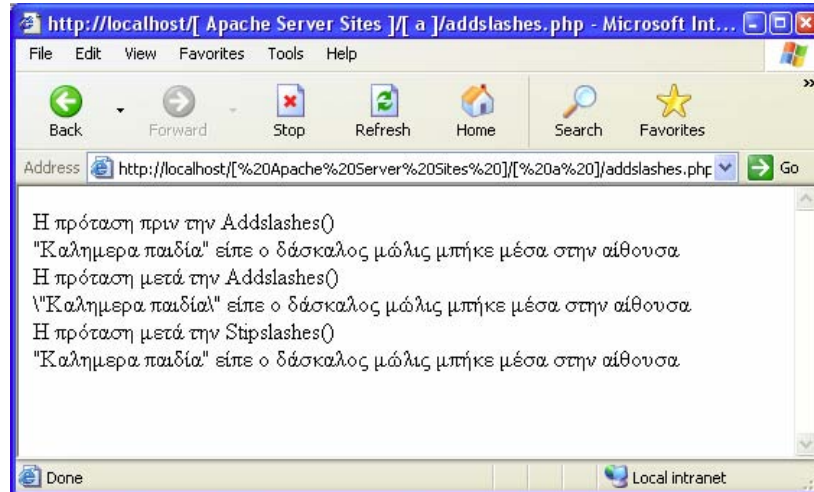
Όπως χρησιμοποιούνται συναρτήσεις συμβολοσειρών για να μορφοποιηθεί μια συμβολοσειρά οπτικά μπορούν να χρησιμοποιηθούν κάποιες από αυτές τις συναρτήσεις για να μορφοποιηθεί μια συμβολοσειρά για αποθήκευση σε μια βάση δεδομένων. Κάποιοι χαρακτήρες είναι έγκυροι όταν είναι μέρος μιας συμβολοσειράς αλλά μπορούν να δημιουργήσουν προβλήματα, ιδιαίτερα όταν εισάγονται δεδομένα σε μια βάση δεδομένων, επειδή η βάση δεδομένων θα μπορούσε να μεταφράσει αυτούς τους χαρακτήρες σαν χαρακτήρες ελέγχου. Οι προβληματικοί χαρακτήρες είναι τα εισαγωγικά (μονά και διπλά), οι ανάποδες κάθετοι και ο χαρακτήρας NULL. Πρέπει να βρεθεί ένας τρόπος να σημειώνονται αυτοί οι χαρακτήρες, ώστε οι βάσεις δεδομένων, όπως η MySQL να καταλάβουν τί υπονοείται όταν χρησιμοποιείται ένας σταθερός ειδικός χαρακτήρα, αντί για μια σειρά ελέγχου. Για να αποφεύγονται αυτοί οι χαρακτήρες προστίθεται μια ανάποδη κάθετος ( \ ) εμπρός από αυτούς. Για παράδειγμα τα διπλά εισαγωγικά “ γίνονται \” και η ανάποδος κάθετος γίνεται \\. Η PHP παρέχει δύο συναρτήσεις ειδικά σχεδιασμένες για χαρακτήρες διαφυγής. Πριν γραφτεί μια συμβολοσειρά σε μια βάση δεδομένων, θα πρέπει να μορφοποιηθεί με τις εντολές Addslashes() για παράδειγμα :

```
$feedback = addslashes($feedback);
```

Όταν χρησιμοποιείται η `AddSlashes()`, η συμβολοσειρά αποθηκεύεται με τις ανάποδους καθέτους. Όταν ανακαλείται η συμβολοσειρά, θα πρέπει απαραίτητα να αφαιρούνται οι καθέτες. Αυτό επιτυγχάνεται με την χρησιμοποίηση της εντολής `StripSlashes()`. Για παράδειγμα :

```
$feedback = StripSlashes($feedback);
```

Η εικόνα 2.2 δείχνει τα αποτελέσματα όταν χρησιμοποιούνται αυτές οι συναρτήσεις σε συμβολοσειρές.



Εικόνα 1.7 – Οι functions `AddSlashes()` και `StripSlashes()`

```
<html> <head> <meta http-equiv="Content-Type" content="text/html; charset=windows-1253"> </head> <body> <?
    $feed = "Καλημέρα παιδιά" είπε ο δάσκαλος μόλις μπήκε μέσα στην αίθουσα';
    echo "Η πρόταση πριν την addslashes()<br>"; echo $feed; echo "<br>Η πρόταση μετά την addslashes()<br>"; $feed = addslashes($feed); echo $feed; echo "<br>Η πρόταση μετά την stripslashes()<br>"; $feed = stripslashes($feed); echo $feed;
?>
</body> </html>
```

### Χρησιμοποιώντας την `Substr()`

Με την συνάρτηση `substr()` να δίνεται πρόσβαση σε μια δευτερεύουσα συμβολοσειρά, μεταξύ συγκεκριμένων αρχικών και τελικών σημείων μιας συμβολοσειράς. Η συνάρτηση `substr()` έχει το παρακάτω πρότυπο :

```
string substr(string , int start , int [length] );
```

Αυτή η συνάρτηση επιστρέφει μια δευτερεύουσα συμβολοσειρά, που έχει αντιγράψει μέσα από το string.

Παραδείγματα χρησιμοποιώντας την συμβολοσειρά `test` :

```
$test = "Your customer service is excellent";
```

### παραδείγματα

```
substr($test, 1); □ "our customer service is excellent" substr($test, -9); □ "excellent" substr($test, 0, 4); □ "Your" substr($test, 4, -13); □ "customer service"
```

### Ελέγχοντας το μήκος συμβολοσειράς με την `strlen()`

Μπορεί να ελέγχεται το μέγεθος μιας συμβολοσειράς με τη συνάρτηση `strlen()`.



Αν περάσει μια συμβολοσειρά , αυτή η συνάρτηση θα επιστρέψει το μήκος της. Για παράδειγμα , strlen(“hello”) επιστρέφει 5.

#### Επαναχρησιμοποίηση Κώδικα και σύνταξη συναρτήσεων

Ένας από τους στόχους των προγραμματιστών είναι να επαναχρησιμοποιούν κώδικα αντί να συντάσσουν νέο κώδικα. Η επαναχρησιμοποίηση υπάρχοντος κώδικα , περιορίζει το κόστος , αυξάνει την αξιοπιστία και βελτιώνει τη συνέπεια.

#### Χρησιμοποιώντας τις require() και include()

Η PHP παρέχει δύο πολύ απλές εντολές, για να μπορεί να ξαναχρησιμοποιηθεί οποιοδήποτε είδος κώδικα. Χρησιμοποιώντας μια εντολή require() ή include(), μπορεί να φορτωθεί ένα αρχείο σε ένα PHP script. Το αρχείο μπορεί να περιέχει οτιδήποτε που θα πληκτρολογηθεί σε ένα script, συμπεριλαμβανομένων PHP εντολών, κειμένου, HTML ετικετών, PHP συναρτήσεων.

#### Χρησιμοποιώντας την require()

Ο παρακάτω κώδικας αποθηκεύεται σε ένα αρχείο που ονομάζεται reusable. php:

```
<? echo “Αυτό είναι ένα απλό δείγμα PHP. <br>”; ?>
```

Ο παρακάτω κώδικας αποθηκεύεται σε ένα αρχείο που ονομάζεται main. php

```
<? echo “Αυτό είναι το κεντρικό αρχείο. <br>”; require (“reusable. php”); echo “Το script τελειώνει εδώ. <br>”;
```

```
?>
```

Αν φορτωθεί το reusable. php, θα εμφανιστεί το μήνυμα «είναι ένα απλό δείγμα PHP» στον browser. Αν φορτωθεί το αρχείο main. php τότε θα εμφανιστεί κάτι λίγο πιο ενδιαφέρον.





Αυτό καλεί μια συνάρτηση που ονομάζεται `function_name` και δεν απαιτεί παραμέτρους. Αυτή η γραμμή αγνοεί οποιαδήποτε τιμή μπορεί να επιστρέψει η συνάρτηση. Οι περισσότερες συναρτήσεις απαιτούν μια ή περισσότερες παραμέτρους, δηλαδή πληροφορίες που δίνονται σε μια συνάρτηση όταν καλείται και που επηρεάζουν το αποτέλεσμα της εκτέλεσης της συνάρτησης. Οι παράμετροι περνώνται τοποθετώντας τα δεδομένα ή το όνομα μιας μεταβλητής που περιέχει τα δεδομένα μέσα σε παρενθέσεις, μετά το όνομα της συνάρτησης. Μια κλήση με παράμετρο μοιάζει ως εξής :

```
Function_name("parameter");
```

Σε αυτή την περίπτωση η παράμετρος που χρησιμοποιήθηκε ήταν μια συμβολοσειρά που περιείχε μόνο την λέξη «parameter», αλλά οι παρακάτω κλήσεις είναι επίσης αποδεκτές ανάλογα με την συνάρτηση.

```
Function_name(2);
```

```
Function_name(6. 231);
```

```
Function_name("$variable");
```

#### Γιατί να ορίζονται προσωπικές συναρτήσεις

Οι συναρτήσεις που είναι ενσωματωμένες στην PHP επιτρέπουν την συνδιαλλαγή με αρχεία, τη χρησιμοποίηση βάσεις δεδομένων, τη δημιουργία γραφικών και τη σύνδεση σε άλλους διακομιστές. Παρόλα αυτά μπορούν να γραφτούν και οι μας συναρτήσεις κάποιου προγραμματιστή, για να εκτελεστεί οποιαδήποτε διαδικασία χρειαστεί.

#### Βασική δομή συναρτήσεων

Μια δήλωση συνάρτησης δημιουργεί ή δηλώνει μια νέα συνάρτηση. Η δήλωση ξεκινάει με την λέξη κλειδί **function**, δίνει το όνομα της συνάρτησης, τις απαιτούμενες παραμέτρους και περιέχει τον κώδικα που εκτελείται κάθε φορά που καλείται η συνάρτηση.

Παράδειγμα :

```
Function my_function() { echo "Η συνάρτηση μου κλήθηκε"; }
```

Για την κλήση της αρκεί να γραφτεί η εντολή `my_function()`;

#### Ονομάζοντας την συνάρτηση του προγραμματιστή

Το πιο σημαντικό θέμα που πρέπει να σκεφτεί κατά την ονομασία των συναρτήσεων είναι ότι το όνομα πρέπει να είναι σύντομο και περιγραφικό.

Μερικοί περιορισμοί είναι οι εξής:

- Η συνάρτηση δεν μπορεί να έχει το ίδιο όνομα με μια υπάρχουσα συνάρτηση
- Το όνομα της συνάρτησης μπορεί να περιέχει μόνο γράμματα, ψηφία και χαρακτηριστικές υπογράμμισης
- Το όνομα της συνάρτησης δεν μπορεί να ξεκινάει με ψηφίο

### Παράμετροι

Για να κάνουν την δουλειά τους οι περισσότερες συναρτήσεις απαιτούν μια ή δύο παραμέτρους. Μια παράμετρος επιτρέπει το πέρασμα δεδομένων σε μια συνάρτηση. Παρακάτω υπάρχουν δυο συναρτήσεις που απαιτούν μία παράμετρο. Στην πρώτη περίπτωση η παράμετρος της συναρτήσεως αντιστοιχεί στο πλήθος των πινάκων που χρειάζεται να σχεδιαστούν στην έξοδο του browser, ενώ στη δεύτερη περίπτωση η παράμετρος αντιστοιχεί στο πλήθος των επαναλήψεων μιας λέξης.

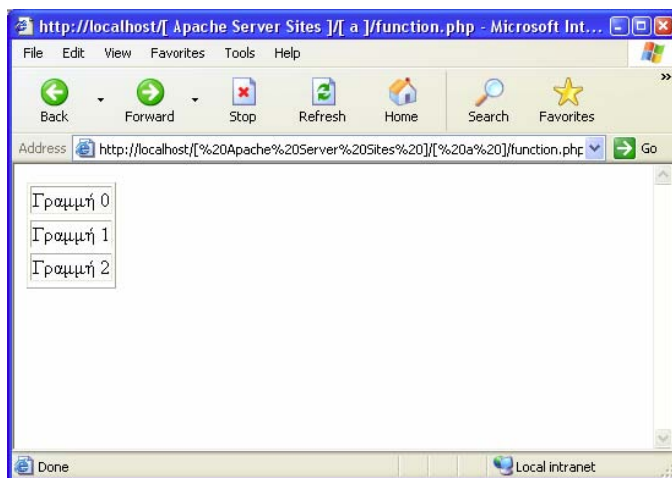
```
<?
```

```
function create_table($data)
{
    echo "<table border=1>";
    for ($i =1 ; $i <= $data ; $i++)
    {
        echo "<tr><td>Γραμμή $i</td><tr>\n";
    }
    echo "</table>";
}
```

Για την εκτέλεσή της πρέπει να γραφτεί η εντολή :

```
<? create_table(3) ?>
```

και το αποτέλεσμα στον browser θα είναι το εξής :



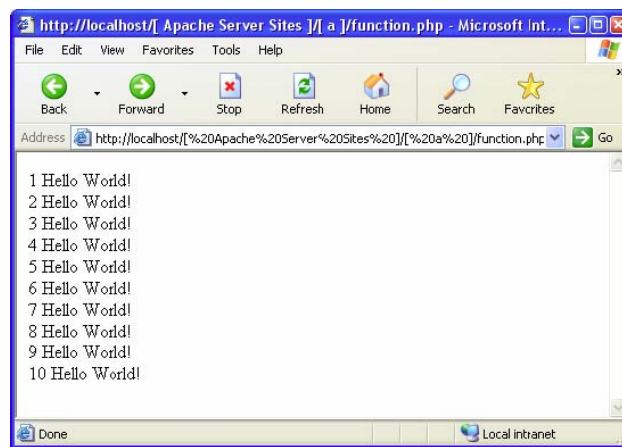
Εικόνα 1. 9 – Δημιουργία πίνακα μέσω της HTML

```
<?
function msg($data)
{
for ($i =1 ; $i <= $data ; $i++)
{
    echo $i. " Hello World!<br>";
}
}
?>
```

Για την εκτέλεσή της πρέπει να γραφτεί η εντολή :

```
<? msg(10); ?>
```

Το αποτέλεσμα στον browser είναι:



Εικόνα 1. 3 – Παράδειγμα χρήσης function msg()

## **2ο ΚΕΦΑΛΑΙΟ**

Σχεδιασμός των βάσεων – Φόρμες εισαγωγής






















## 2.1 Εισαγωγή

Στο κεφάλαιο αυτό θα αναλυθούν όλοι πίνακες που δημιουργήθηκαν κατά την εκπόνηση της εφαρμογής και χωρίζονται σε δύο κατηγορίες. Στους πίνακες καταγραφής δεδομένων και στους πίνακες τήρησης στατιστικών στοιχείων. Επίσης αναφέρονται οι φόρμες εισαγωγής στοιχείων, οι φόρμες που χειρίζεται αποκλειστικά ο διαχειριστής και οι λοιπές φόρμες, που είναι αυτές που δεν κατατάσσονται σε καμία από τις παραπάνω κατηγορίες.

## 2.2 Καταγραφή των πινάκων της βάσης.

Για το σχεδιασμό της βάσης της εταιρίας δημιουργήθηκαν επτά βασικοί πίνακες.






















- Ο πίνακας members που περιλαμβάνει το username και τον κωδικό του κάθε χρήστη.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>ID</u>	mediumint(5)			No	None	auto_increment	      
<input type="checkbox"/>	<u>username</u>	varchar(100)	utf8_general_ci		No	None		      
<input type="checkbox"/>	<u>user_password</u>	varchar(40)	utf8_general_ci		No	None		      

```
CREATE TABLE IF NOT EXISTS `members` (
  `ID` mediumint(5) NOT NULL AUTO_INCREMENT,
  `username` varchar(100) NOT NULL,
  `user_password` varchar(40) NOT NULL,
  PRIMARY KEY (`ID`, `username`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;
```

Εικόνα 2. 1 – Δημιουργία πίνακα members

- Ο πίνακας vacation\_type που περιλαμβάνει όλες τις άδειες που μπορούν να ζητηθούν από το προσωπικό της εταιρίας.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>ID</u>	mediumint(5)			No	None	auto_increment	      
<input type="checkbox"/>	<u>type</u>	varchar(100)	utf8_general_ci		No	None		      
<input type="checkbox"/>	<u>maxdays</u>	int(2)			No	None		      

```
CREATE TABLE IF NOT EXISTS `vacation_type` (
  `ID` mediumint(5) NOT NULL AUTO_INCREMENT,
  `type` varchar(100) NOT NULL,

```

```
`maxdays` int(2) NOT NULL,  
PRIMARY KEY (`ID`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=16 ;
```

Εικόνα 2. 2 – Δημιουργία πίνακα vacation\_type







































































- Ο πίνακας departments που περιλαμβάνει τα τμήματα στα οποία είναι χωρισμένη η εταιρία.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>ID</u>	mediumint(9)			No	None	auto_increment	      
<input type="checkbox"/>	name	varchar(50)	utf8_bin		No	None		      

```
CREATE TABLE IF NOT EXISTS `departments` (  
  `ID` mediumint(9) NOT NULL AUTO_INCREMENT,  
  `name` varchar(50) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,  
  PRIMARY KEY (`ID`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

Εικόνα 2. 3 – Δημιουργία πίνακα departments

- Ο πίνακας employee που περιλαμβάνει όλους τους υπαλλήλους που ανήκουν στο δυναμικό της εταιρίας.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>UserID</u>	mediumint(5)			No	None	auto_increment	      
<input type="checkbox"/>	name	varchar(30)	utf8_general_ci		No	None		      
<input type="checkbox"/>	surname	varchar(30)	utf8_general_ci		No	None		      
<input type="checkbox"/>	email	varchar(50)	utf8_general_ci		No	None		      
<input type="checkbox"/>	department	varchar(50)	utf8_general_ci		No	None		      
<input type="checkbox"/>	groups	int(10)			No	None		      
<input type="checkbox"/>	normaldays	int(2)			No	None		      
<input type="checkbox"/>	speciality	varchar(50)	utf8_general_ci		No	None		      
<input type="checkbox"/>	private_min	int(3)			No	180		      
<input type="checkbox"/>	previous_year	int(11)			No	None		      



























































































```
CREATE TABLE IF NOT EXISTS `employee` (  
  `UserID` mediumint(5) NOT NULL AUTO_INCREMENT,  
  `name` varchar(30) NOT NULL,  
  `surname` varchar(30) NOT NULL,  
  `email` varchar(50) NOT NULL,  
  `department` varchar(50) NOT NULL,  
  `groups` int(10) NOT NULL,  
  `normaldays` int(2) NOT NULL,
```

```

`speciality` varchar(50) NOT NULL,
`private_min` int(3) NOT NULL DEFAULT '180',
`previous_year` int(11) NOT NULL,
PRIMARY KEY (`UserID`, `name`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

```

- Ο πίνακας vacation που περιλαμβάνει όλες τις άδειες που έχουν καταχωρηθεί στη βάση. Μια [Εικόνα 2. 4 – Δημιουργία πίνακα employee](#) εγγραφή καταχωρείται στον πίνακα αυτόν όποτε ένας υπάλληλος ζητήσει οποιοδήποτε τύπο άδειας.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> ID	mediumint(10)			No	None	auto_increment	     
<input type="checkbox"/> Req_by	mediumint(5)			No	None		     
<input type="checkbox"/> type	varchar(100)	utf8_general_ci		No	None		     
<input type="checkbox"/> Start_day	date			No	None		     
<input type="checkbox"/> End_day	date			No	None		     
<input type="checkbox"/> Reason	varchar(500)	utf8_general_ci		No	None		     
<input type="checkbox"/> Rej_Reason	varchar(500)	utf8_general_ci		No	None		     
<input type="checkbox"/> Stage	int(2)			No	None		     
<input type="checkbox"/> Req_at	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP	     
<input type="checkbox"/> dayssum	int(2)			No	0		     
<input type="checkbox"/> hourssum	int(2)			No	0		     
<input type="checkbox"/> Start_hour	time			No	None		     
<input type="checkbox"/> End_hour	time			No	None		     
<input type="checkbox"/> Back_day	date			No	None		     
<input type="checkbox"/> days	int(11)			No	None		     

```

CREATE TABLE IF NOT EXISTS `vacation` (
  `ID` mediumint(10) NOT NULL AUTO_INCREMENT,
  `Req_by` mediumint(5) NOT NULL,
  `type` varchar(100) NOT NULL,
  `Start_day` date NOT NULL,
  `End_day` date NOT NULL,
  `Reason` varchar(500) NOT NULL,
  `Rej_Reason` varchar(500) NOT NULL,
  `Stage` int(2) NOT NULL,
  `Req_at` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `dayssum` int(2) NOT NULL DEFAULT '0',
  `hourssum` int(2) NOT NULL DEFAULT '0',
  `Start_hour` time NOT NULL,
  `End_hour` time NOT NULL,
  `Back_day` date NOT NULL,
  `days` int(11) NOT NULL,















```



```
PRIMARY KEY (`ID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

Εικόνα 2. 5 – Δημιουργία πίνακα vacation






















- Ο πίνακας groups που καταγράφει το επίπεδο απασχόλησης των υπαλλήλων της εταιρίας.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	<u>ID</u>	mediumint(5)			No	None		      
<input type="checkbox"/>	type	varchar(50)	utf8_bin		No	None		      

```
CREATE TABLE IF NOT EXISTS `groups` (
  `ID` mediumint(5) NOT NULL,
  `type` varchar(50) CHARACTER SET utf8 COLLATE utf8_bin NOT NULL,
  PRIMARY KEY (`ID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

Εικόνα 2. 6 – Δημιουργία πίνακα groups

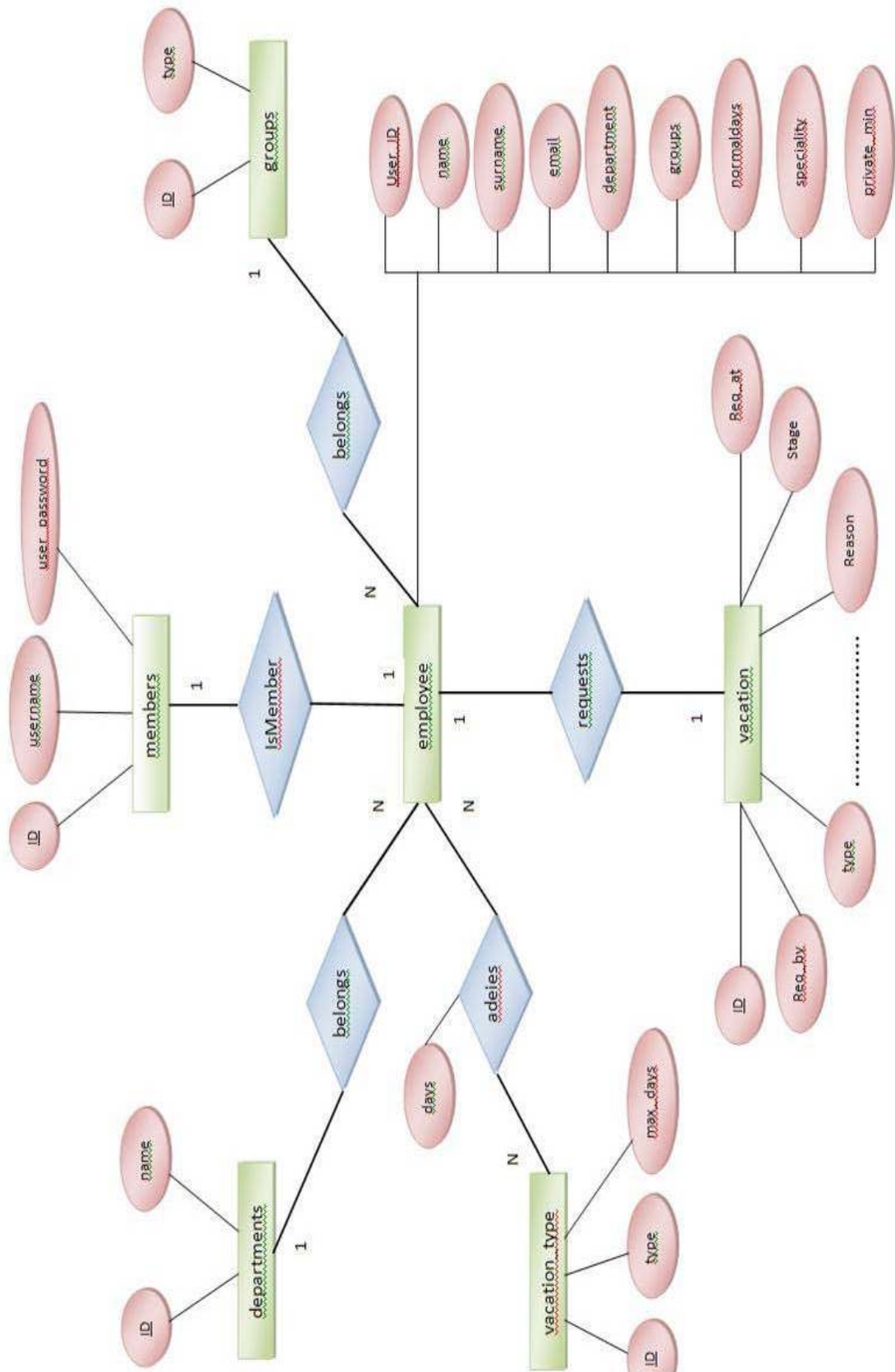
- Ο πίνακας adeies που περιλαμβάνει το σύνολο των ημερών που έχουν εγκριθεί για όλες τις άδειες και για κάθε υπάλληλο.

	Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/>	UserID	mediumint(5)			No	None		      
<input type="checkbox"/>	type	varchar(30)	utf8_general_ci		No	None		      
<input type="checkbox"/>	days	int(10)			No	0		      

```
CREATE TABLE IF NOT EXISTS `adeies` (
  `UserID` mediumint(5) NOT NULL,
  `type` varchar(30) NOT NULL,
  `days` int(10) NOT NULL DEFAULT '0'
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC;
```

Εικόνα 2. 7 – Δημιουργία πίνακα adeies

### 2.3 Διάγραμμα ER



## 2.4 Πίνακες καταγραφής στατιστικών στοιχείων

Εκτός από τους επτά βασικούς πίνακες δημιουργήθηκαν δύο επιπλέον. Ο πίνακας `adeies_backup` και ο πίνακας `backup`, που είναι πιστά αντίγραφα των πινάκων `adeies` και `vacation` αντίστοιχα. Σκοπός των πινάκων αυτών είναι:

### 1. Ταχύτερη προσπέλαση των βασικών πινάκων

Για να μην επιβαρύνονται οι βασικοί πίνακες της βάσης (`adeies`, `vacation`) ο χρήστης `administrator` έχει τη δυνατότητα διαγραφής όλων των εγγραφών τους στην αρχή κάθε έτους. Όλες αυτές οι εγγραφές, πριν διαγραφούν, μεταφέρονται στους δύο παραπάνω πίνακες.

### 2. Καταγραφή στατιστικών στοιχείων

Αντίθετα με τους βασικούς πίνακες που περιέχουν εγγραφές μόνο του τρέχοντος έτους, στους πίνακες-αντίγραφα καταχωρούνται όλες οι εγγραφές. Για το λόγο αυτό οι πίνακες-αντίγραφα χρησιμοποιούνται για να αντλούνται όλα τα πιθανά στατιστικά στοιχεία.

### 3. Ασφάλεια της βάσης

Χρησιμοποιώντας πίνακες-αντίγραφα δίνεται μια ικανοποιητική λύση στο πρόβλημα της απώλειας δεδομένων από τη βάση.

Πίνακας <code>adeies_backup</code>														
	Field	Type	Collation	Attributes	Null	Default	Extra	Action						
<input type="checkbox"/>	<b>UserID</b>	mediumint(5)			No	None								
<input type="checkbox"/>	<b>type</b>	varchar(30)	utf8_general_ci		No	None								
<input type="checkbox"/>	<b>days</b>	int(10)			No	0								

	Field	Type	Collation	Attributes	Null	Default	Extra	Action						
<input type="checkbox"/>	<b>ID</b>	mediumint(10)			No	None	auto_increment							
<input type="checkbox"/>	<b>Req_by</b>	mediumint(5)			No	None								
<input type="checkbox"/>	<b>type</b>	varchar(100)	utf8_general_ci		No	None								
<input type="checkbox"/>	<b>Start_day</b>	date			No	None								
<input type="checkbox"/>	<b>End_day</b>	date			No	None								
<input type="checkbox"/>	<b>Reason</b>	varchar(500)	utf8_general_ci		No	None								
<input type="checkbox"/>	<b>Rej_Reason</b>	varchar(500)	utf8_general_ci		No	None								
<input type="checkbox"/>	<b>Stage</b>	int(2)			No	None								
<input type="checkbox"/>	<b>Req_at</b>	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP							
<input type="checkbox"/>	<b>dayssum</b>	int(2)			No	0								
<input type="checkbox"/>	<b>hourssum</b>	int(2)			No	0								
<input type="checkbox"/>	<b>Start_hour</b>	time			No	None								
<input type="checkbox"/>	<b>End_hour</b>	time			No	None								
<input type="checkbox"/>	<b>Back_day</b>	date			No	None								
<input type="checkbox"/>	<b>days</b>	int(11)			No	None								

Εικόνα 2. 8 – Δημιουργία πίνακα `adeies backup`

## 2.5 Φόρμες εισαγωγής δεδομένων

Για την εισαγωγή δεδομένων στην εφαρμογή δημιουργήθηκαν έντεκα φόρμες. Το επίπεδο απασχόλησης και η ιδιότητα των υπαλλήλων καθορίζει την πρόσβασή τους στις ανάλογες φόρμες.

- **Η αρχική φόρμα**(home. php):

Η αρχική φόρμα της εφαρμογής , η οποία περιέχει οδηγίες και ανακοινώσεις προς τους χρήστες. Η φόρμα αυτή ειδοποιεί τον κατάλληλο χρήστη αν έχει άδειες προς έγκριση.

Ο κώδικας που την υλοποιεί είναι ο παρακάτω

---

```
<?php
$i=0;
$query=mysql_query("SELECT groups , department FROM employee
WHERE UserID='$member_ID'");
$row=mysql_fetch_assoc($query);
$department=$row['department'];
$group=$row['groups'];
$result=mysql_query("SELECT UserID FROM employee WHERE
department='$department'");
while($row=mysql_fetch_array($result)){
    if($group==3){
        $result2=mysql_query("SELECT ID FROM vacation WHERE
Stage='0' AND Req_by='$row[0]'");
        while($row=mysql_fetch_array($result2)){
            $i++;
        }
    }
}
}
while
if($group==2){
```

```

$result2=mysql_query("SELECT ID FROM vacation WHERE
Stage=1");
while($row=mysql_fetch_array($result2)){
    $i++;};//while
};//if
else if($group==1){
    $result2=mysql_query("SELECT ID FROM vacation WHERE
Stage=2");
    while($row=mysql_fetch_array($result2)){
        $i++;};//while
    };//else ?>

```

- **Η φόρμα καταχώρησης άδειας(adeia. php):**

Σε αυτή τη φόρμα ο κάθε υπάλληλος καταχωρεί την άδεια που επιθυμεί να του χορηγηθεί.

Εικόνα 2. 9 Φόρμα καταχώρησης άδειας

Ανάλογα με την άδεια που θα ζητηθεί μπορούν να γίνουν οι παρακάτω ενέργειες:

- Έλεγχος αν ο χρήστης έχει υπερβεί το όριο ημερών που του αναλογεί για την κάθε άδεια.

---

#### Κώδικας έλεγχου

```
else if($mins>$sum){
$check=false;?>
<script language="JavaScript">
alert ("Έχετε υπερβεί το όριο ωρών σας | Εισάγετε νέα ημερομηνία")
</script>
<?php }//if
```

---

- Εμφάνιση των κατάλληλων πεδίων ανάλογα με τον τύπο άδειας που θα ζητηθεί. Παράδειγμα η εμφάνιση του πεδίου αιτιολογίας στις άδειες Εργασίας ωρών και ρεπτό. (Κώδικας στο παράρτημα Α)
- Έλεγχος αν η αρχή της άδειας είναι 10 μέρες προγενέστερη της μέρας που θα καταχωρηθεί, κάτι το οποίο απαγορεύεται από το καταστατικό της εταιρίας.

---

#### Κώδικας έλεγχου

```
else if($difference2>10){
$check=false;?>
<script language="JavaScript">
alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα. Προσπαθήστε ξανά")
</script>
<?php }//if
```

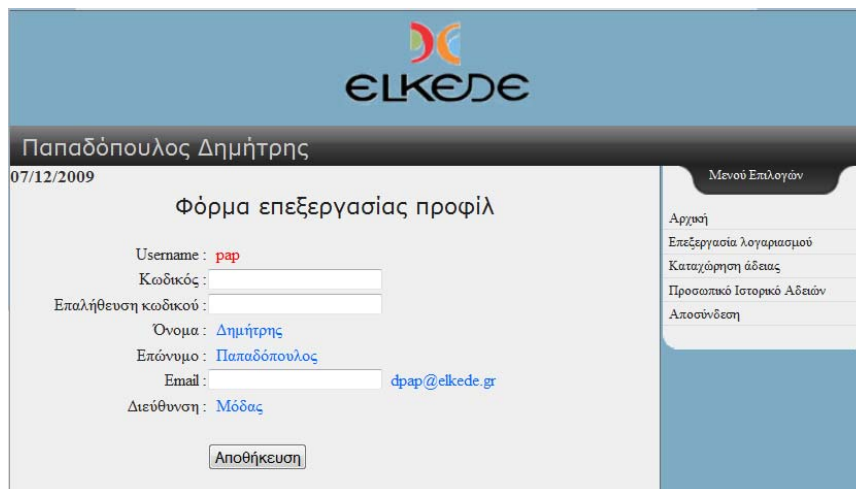
---

- **Η φόρμα επεξεργασίας λογαριασμού(profile. php):**

Σε αυτή τη φόρμα μπορεί ο κάθε χρήστης να επεξεργαστεί τον λογαριασμό του. Να αλλάξει τον κωδικό πρόσβασης στο σύστημα καθώς και το email του.

Για την αλλαγή του κωδικού πρόσβασης απαιτείται και επαλήθευση για την αποφυγή οποιουδήποτε σφάλματος κατά την εισαγωγή.

(Κώδικας στο παράρτημα Β)



Εικόνα 2. 10 Φόρμα επεξεργασίας λογαριασμού

- **Η φόρμα προσωπικού ιστορικού αδειών(history. php):**

Σε αυτή τη φόρμα υπάρχει η δυνατότητα , ο χρήστης να ελέγξει πόσες μέρες από την κάθε άδεια έχει χρησιμοποιήσει , πόσες ημέρες του απομένουν καθώς και το όριο για κάθε ένα τύπο άδειας. Με την επιλογή «αναλυτικά» ο χρήστης ενημερώνεται για την κατάσταση και τις λεπτομέρειες των αδειών που έχει ζητήσει. Αναφέρεται ο τύπος της άδειας , η ακριβής ημερομηνία έναρξης και λήξης καθώς και η διάρκεια της. Επίσης αναφέρεται εάν η άδεια έχει εγκριθεί , έχει απορριφθεί ή εάν βρίσκεται στο στάδιο της έγκρισης.



Παπαδόπουλος Δημήτρης

08/12/2009

Μενού Επιλογών

Αργωή  
Επεξεργασία λογαριασμού  
Καταχώρηση άδειας  
Προσωπικό Ιστορικό Αδειών  
Αποσύνδεση

### Προσωπικό Ιστορικό Αδειών

Απο το σύνολο των 1 αδειών που έχουν ζητηθεί. Οι 0 έχουν εγκριθεί ενώ 1 είναι σε αναμονή.  
Οι αιτήσεις σας

ΤΥΠΟΣ ΑΔΕΙΑΣ	ΟΡΙΟ ΗΜΕΡΩΝ	ΕΓΚΡΙΘΕΙΣΕΣ ΗΜΕΡΩΝ	ΥΠΟΛΟΙΠΟ ΗΜΕΡΩΝ
ΚΑΝΟΝΙΚΗ	25	2	23
ΠΡΟΣΩΠΙΚΗ	180 (σε λεπτά)	0 (σε λεπτά)	180 (σε λεπτά)
ΕΡΓΑΣΙΑΣ	X	0	X
ΕΡΓΑΣΙΑΣ ΩΡΩΝ	X	0	X
ΡΕΠΟ	X	0	X
ΓΟΝΙΚΗ	5	0	5
ΕΚΛΟΓΙΚΗ	3	0	3
ΠΕΝΘΟΥΣ	3	0	3
ΓΑΜΟΥ	5	0	5
ΕΚΠΑΙΔΕΥΤΙΚΗ	X	0	X
ΑΙΜΟΔΟΤΙΚΗ	X	0	X
ΜΕΤΑΚΟΜΙΣΗΣ	2	0	2
ΥΠ. ΔΗΛΩΣΗ ΑΣΘΕΝΕΙΑΣ	X	0	X
ΠΑΡΑΣΤΑΣΗ ΣΕ ΔΙΚΗ ΚΑΝΟΝΙΚΗ ΠΑΛΙΟΥ ΕΤΟΥΣ	0	0	0

Εικόνα 2. 11 Φόρμα προσωπικού ιστορικού αδειών



- Η φόρμα **ιστορικού υπαλλήλων**(employee\_history. php):  
Οι χρήστες που έχουν δικαίωμα πρόσβασης σε αυτή τη φόρμα έχουν τη δυνατότητα να ελέγχουν πόσες και ποιές άδειες έχουν χορηγηθεί και σε ποιούς υπαλλήλους.  
Αφού έχει επιλεγεί ένας συγκεκριμένος υπάλληλος (ή όλοι) και μια συγκεκριμένη άδεια(ή όλες) και εισαχθεί το διάστημα για το οποίο γίνεται ο έλεγχος φορτώνεται εκ νέου το αρχείο employee\_history. php πατώντας το κουμπί «Αναζήτηση».

08/12/2009

Ιστορικό Υπαλλήλων

Επιλέξτε υπάλληλο : ΟΛΟΙ

Τύπος άδειας : ΟΛΕΣ

Έναρξη άδειας : Ημέρα : Μήνας : Χρόνος :

Τέλος άδειας : Ημέρα : Μήνας : Χρόνος :

Αναζήτηση

Μενού Επιλογών

- Αρχική
- Επεξεργασία λογαριασμού
- Ιστορικό Υπαλλήλων
- Έγκριση άδειας
- Επεξεργασία χρηστών
- Αποσύνδεση

Εικόνα 2. 12 Φόρμα ιστορικού υπαλλήλων

Ανάλογα με το σκοπό της αναζήτησης υπάρχουν τέσσερις δυνατές επιλογές:

- Όλοι οι υπάλληλοι και όλοι οι τύποι αδειών
- Όλοι οι υπάλληλοι και ένας συγκεκριμένος τύπος άδειας.
- Ένας συγκεκριμένος υπάλληλος και όλοι οι τύποι αδειών.
- Ένας συγκεκριμένος υπάλληλος και ένας συγκεκριμένος τύπος άδειας.

(Κώδικας στο παράρτημα Γ)

- Η φόρμα **έγκρισης άδειας**(approve. php):

Στη φόρμα αυτή γίνεται η έγκριση των αδειών που ζητήθηκαν από τους υπαλλήλους. Ο προϊστάμενος του κάθε υπαλλήλου έχει τη δυνατότητα είτε να εγκρίνει είτε να απορρίψει την αίτηση. Προαιρετικά υπάρχει η δυνατότητα εκτύπωσης της αίτησης καθώς και η αιτιολογία στην περίπτωση απόρριψής της. Σε κάθε μια από τις παραπάνω περιπτώσεις αποστέλλεται email ενημέρωσης του χρήστη.

(Κώδικας στο παράρτημα Δ)

Εικόνα 2. 13 Φόρμα έγκρισης άδειας

- Η φόρμα **επεξεργασίας χρηστών**(admpofile. php):

Στη φόρμα αυτή γίνεται επεξεργασία των προσωπικών στοιχείων όλων των υπαλλήλων από τους χρήστες με τα κατάλληλα δικαιώματα. Η επιλογή του υπαλλήλου του οποίου τα στοιχεία θέλουμε να επεξεργαστούμε γίνεται με τους δύο παρακάτω τρόπους:

- Πληκτρολογώντας ακριβώς το επώνυμο και το όνομά του.
- Επιλέγοντας τον από την λίστα υπαλλήλων.

Πατώντας το πλήκτρο «επεξεργασία» φορτώνεται εκ νέου το αρχείο admpofile. php με τα στοιχεία του υπαλλήλου που επιλέχθηκε.

Εικόνα 2. 14 Φόρμα επεξεργασίας χρηστών

Ο παρακάτω κώδικας ελέγχει τον τρόπο που επιλέχθηκε ο χρήστης.

---

```
if($_POST['scrollbar']!=""){
    $fullname = $_POST['scrollbar'];
    list($last, $first) = explode(' ', $fullname);
    $query=mysql_query("SELECT UserID FROM employee WHERE
surname='$last' AND name='$first' ");
}//if scrollbar
else if($_POST['surname']!="" && $_POST['name']!=""){
    $surnames=mysql_real_escape_string($_POST['surname']);
    $names=mysql_real_escape_string($_POST['name']);
    $query= mysql_query("SELECT UserID FROM employee WHERE
surname='$surnames' AND name='$names'");
    if(!$row=mysql_fetch_array($query)){
        ?><script language="JavaScript">
            alert ("Δεν υπάρχει χρήστης με τα στοιχεία που
δώσατε");
            window. open("admprofile. php" , '_self');
        </script><?php }//if//else if
```

---

Αφού γίνει ο παραπάνω έλεγχος εμφανίζονται τα προσωπικά στοιχεία του χρήστη. Μπορεί να γίνει επεξεργασία οποιουδήποτε στοιχείου είτε μεμονωμένα είτε σε συνδυασμό με κάποιο άλλο είτε όλα τα στοιχεία συγχρόνως.

08/12/2009

Φόρμα επεξεργασίας προφίλ

Username :  rap

Κωδικός :

Όνομα :  Δημήτρης

Επώνυμο :  Παπαδόπουλος

Ημέρες κανονικής άδειας :  25

Email :  drap@elkede.gr

Ειδικότητα :  Σχεδιαστής

Επίπεδο απασχόλησης :  Υπάλληλος

Διεύθυνση :  Μόδας

Διαγραφή :

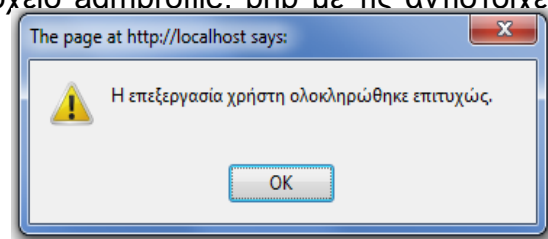
Αποθήκευση

Μενού Επιλογών

- Αργωσι
- Επεξεργασία λογαριασμού
- Ιστορικό Υπαλλήλων
- Εγκριση άδειας
- Επεξεργασία χρηστών
- Αποσύνδεση

Εικόνα 2. 15 Φόρμα επεξεργασίας χρηστών 2

Όταν πληκτρολογηθούν τα νέα στοιχεία του χρήστη και πατώντας το πλήκτρο «αποθήκευση» φορτώνεται και πάλι το αρχείο admnprofile. php με τις αντίστοιχες μεταβλητές και το σύστημα ενημερώνει τον χρήστη με κατάλληλο μήνυμα ότι η επεξεργασία ολοκληρώθηκε επιτυχώς.



Εικόνα 2. 16 Μήνυμα λάθους

Πέρα από την οποιαδήποτε επεξεργασία στοιχείων υπάρχει και η δυνατότητα οριστικής διαγραφής κάποιου υπαλλήλου επιλέγοντας το “check box” της διαγραφής και πατώντας αποθήκευση.

---

Κώδικας διαγραφής χρήστη

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

```
$del=$row['UserID'];
```

```
mysql_query("DELETE FROM members WHERE ID=$del");
```

```
mysql_query("DELETE FROM employee WHERE UserID=$del");
```

```
mysql_query("DELETE FROM adeies WHERE UserID=$del");
```

---

## 2.6 Φόρμες Διαχειριστή

- **Η φόρμα επεξεργασίας διευθύνσεων**(departments. php):

Στη φόρμα αυτή έχει πρόσβαση μόνο ο διαχειριστής της εφαρμογής έχοντας ως δυνατότητες τη δημιουργία μιας καινούργιας διεύθυνσης και την επεξεργασία μιας υπάρχουσας.

Εικόνα 2. 17 Φόρμα επεξεργασίας διευθύνσεων

**Δημιουργία διεύθυνσης:** Ο διαχειριστής πρέπει να εισάγει το όνομα της , να επιλέξει προϊστάμενο από την λίστα των υπαλλήλων και πατήσει το κουμπί «Δημιουργία». Στη συνέχεια φορτώνεται το αρχείο departments. php και εκτελείται ο παρακάτω κώδικας.

```
if(isset($_POST['submit'])) {
if ( $_POST['department']!= "" && $_POST['scrollbar']!= ""){
$fullname = mysql_real_escape_string($_POST['scrollbar']);
list($last, $first) = explode(' ', $fullname);
$department=$_POST['department'];
$sql= "INSERT INTO departments (name) VALUES ('$department')";
$sql2 = "UPDATE employee SET department= '$department' WHERE
surname = '$last' AND name = '$first' ";
$sql3 = "UPDATE employee SET groups = 3 WHERE surname = '$last'
AND name = '$first' ";
mysql_query($sql);
mysql_query($sql2);
mysql_query($sql3);
```

```
//if
else if ( $_POST['department']=="" && $_POST['scrollbar']=="" ) {
?><script language="JavaScript">
alert ("Παρακαλώ εισάγεται όλα τα στοιχεία");</script><?php
}//else
else if ( $_POST['department']!="" && $_POST['scrollbar']=="" ){
$department=$_POST['department'];
$sql= "INSERT INTO departments (name) VALUES ('$department')";
mysql_query($sql);
}//else if
}//ifsset
```

---

**Επεξεργασία διεύθυνσης:** Ο διαχειριστής πρέπει να επιλέξει από την λίστα μια από τις υπάρχουσες διευθύνσεις και να πατήσει το κουμπί «Αλλαγή», εμφανίζοντας μια νέα φόρμα. Στη φόρμα αυτή έχει την δυνατότητα να αλλάξει το όνομα της διεύθυνσης , να αλλάξει τον προϊστάμενο ή να τροποποιήσει και τα δύο.

Ο κώδικας που εκτελείται για τις παραπάνω περιπτώσεις δίνεται παρακάτω.

---

```
if(isset($_POST['submit4'])) {
$fullname = $_POST['scrollbar'];
list($last, $first) = explode(' ', $fullname);
$department=$_POST['department'];
$query=mysql_query("SELECT UserID FROM employee WHERE groups=3
AND department='$olddepartment'");
$row=mysql_fetch_assoc($query);
$oldid=$row['UserID'];

$query2=mysql_query("UPDATE employee SET groups=4 WHERE
UserID='$oldid'");
```

```
$sql= "UPDATE departments SET name='$department' WHERE
name='$olddepartment' ";
mysql_query($sql);
$sql2 = "UPDATE employee SET department='$department' WHERE
surname='$last' AND name='$first' ";
$sql3 = "UPDATE employee SET groups=3 WHERE surname='$last' AND
name='$first' ";
mysql_query($sql2);
mysql_query($sql3);
} //ifset?>
```

- **Η φόρμα Administration Panel(admin. php):**

Εικόνα 2. 18 Φόρμα διαχείρισης της εφαρμογής

Μία εξειδικευμένη φόρμα διαχείρισης της εφαρμογής με τις εξής λειτουργίες:

- **Διαγραφή μιας υπάρχουσας διεύθυνσης:** Επιλέγοντας μια διεύθυνση και πατώντας το πλήκτρο «Διαγραφή», τη διαγράφει οριστικά από τη βάση.

Κώδικας διαγραφής διεύθυνσης

```
if(isset($_POST['submit'])){
    $name=mysql_real_escape_string($_POST['delete']);
    mysql_query("DELETE FROM departments WHERE
name='$name'");
```

```
}//if submit
```

---

- ii. **Διαγραφή αδειών:** Ο διαχειριστής επιλέγει έναν υπάλληλο(ή όλους) από τη λίστα , πληκτρολογεί το διάστημα αναζήτησης και πατώντας το κουμπί «Διαγραφή» διαγράφονται όλες οι άδειες με βάση τα επιλεγμένα στοιχεία.
- 

Κώδικας διαγραφής αδειών

```
if(isset($_POST['submit2'])){
$name=$_POST['scrollbar'];
$date=mysql_real_escape_string($_POST['startdate']);
$edate=mysql_real_escape_string($_POST['enddate']);
if($name=="ΟΛΟΙ" && $date!="" && $edate!=""){
$result=mysql_query("INSERT INTO backup SELECT * FROM vacation
WHERE '$date'<=Start_day AND Start_day<='$edate' ");
$result2=mysql_query("DELETE FROM vacation WHERE
'$date'<=Start_day AND Start_day<='$edate' ");
}//if
else if($name!="ΟΛΟΙ" && $date=="" && $edate==""){
$fullname = $_POST['scrollbar'];
list($last, $first) = explode(' ', $fullname);
$sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$last' AND name='$first' ");
$row=mysql_fetch_assoc($sql);
$id=$row['UserID'];
$result=mysql_query("INSERT INTO backup SELECT * FROM vacation
WHERE Req_by='$id' ");
$result2=mysql_query("DELETE FROM vacation WHERE Req_by='$id' ");
}//else if no date
else if($name!="ΟΛΟΙ" && $date!="" && $edate!=""){
$fullname = $_POST['scrollbar'];
list($last, $first) = explode(' ', $fullname);
```



```
$sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$last' AND name='$first' ");
$row=mysql_fetch_assoc($sql);
$id=$row['UserID'];
$result=mysql_query("INSERT INTO backup SELECT * FROM vacation
WHERE Req_by='$id' AND '$sdate'<=Start_day AND Start_day<='$edate'
");
$result2=mysql_query("DELETE FROM vacation WHERE Req_by='$id'
AND '$sdate'<=Start_day AND Start_day<='$edate'");
} //else if
?>
```

---

- iii. **Μηδενισμός ημερών αδειών:** Μηδενίζει το σύνολο των ημερών που έχουν εγκριθεί για κάθε υπάλληλο και κάθε τύπο άδειας. Σκοπός της λειτουργίας αυτής είναι στην αρχή κάθε νέου έτους ο υπάλληλος να μπορεί να πάρει όλες τις ημέρες άδειας που δικαιούται χωρίς να υπερβαίνει το όριο.
- 

Κώδικας μηδενισμός ημερών

```
if(isset($_POST['submit3'])){
//$sql=mysql_query("INSERT INTO adeies_backup SELECT * FROM
adeies");
$sql=mysql_query("SELECT UserID FROM employee GROUP BY
UserID");
while($row=mysql_fetch_array($sql)){
$free=0;
$max=0;
$used=0;
$sql2=mysql_query("SELECT normaldays FROM employee WHERE
UserID='$row[0]'");
$array=mysql_fetch_assoc($sql2);
$max=$array['normaldays'];
```

```
$sql3=mysql_query("SELECT days FROM adeies WHERE  
UserID='$row[0]' AND type='KANONIKH");  
while($array2=mysql_fetch_assoc($sql3)){ $used=$array2['days'];} //while  
$free=$max-$used;  
  
$query=mysql_query("UPDATE employee SET previous_year='$free'  
WHERE UserID='$row[0]'");  
  
} //while;  
  
$query=mysql_query("UPDATE adeies SET days=0 ");  
}
```

- **Η φόρμα καταχώρησης χρηστών(Registration. php):**

Σε αυτή την φόρμα ο διαχειριστής μπορεί να καταχωρήσει τους υπαλλήλους στην εφαρμογή , πληκτρολογώντας τα προσωπικά τους στοιχεία και επιλέγοντας το τμήμα στο οποίο εργάζονται και το επίπεδο απασχόλησής τους σε αυτό.

The screenshot shows a web interface for user registration. At the top left, the date '09/12/2009' is displayed. The main heading is 'Φόρμα καταχώρησης χρηστών' (User Registration Form). The form contains the following fields and controls:

- Όνομα (Name):
- Επώνυμο (Surname):
- Username:
- Κωδικός (Code):
- Email:
- Ημέρες Κανονικής άδειας (Days of Normal Leave):
- Ειδικότητα (Specialty):
- Διεύθυνση (Department):  (Dropdown menu)
- Επίπεδο απασχόλησης (Level of Assignment):  (Dropdown menu)
- Εγγραφή (Registration):

On the right side, there is a 'Μενού Επιλογών' (Options Menu) with the following items:

- Αρχική (Home)
- Επεξεργασία λογαριασμού (Account Management)
- Administration panel
- Καταχώρηση χρηστών (User Registration)
- Επεξεργασία χρηστών (User Management)
- Επεξεργασία τμημάτων (Department Management)
- Αποσύνδεση (Logout)

Εικόνα 2. 19 – Φόρμα καταχώρησης χρηστών

Το username κάθε υπαλλήλου στην εφαρμογή πρέπει να είναι απαραίτητα μοναδικό. Επίσης για να γίνει επιτυχώς η εγγραφή του υπαλλήλου θα πρέπει να έχουν συμπληρωθεί όλα τα πεδία της φόρμας πριν πατηθεί το πλήκτρο «Εγγραφή».

Ο έλεγχος για τις δύο αυτούς περιορισμούς πραγματοποιείται από τα παρακάτω κομμάτια κώδικα.

---

#### Κώδικας ελέγχου στοιχείων

```
if ( $names=="" || $surnames=="" || $mails=="" || $group=="" ||  
$username=="" || $password=="" || $spec==""){ ?>  
    <script language="JavaScript">  
        alert ("Πρέπει να εισάγεται όλα τα στοιχεία σας ,  
Προσπαθήστε ξανά")  
    </script>  
<?php }//if
```

#### Κώδικας ελέγχου μοναδικότητας username

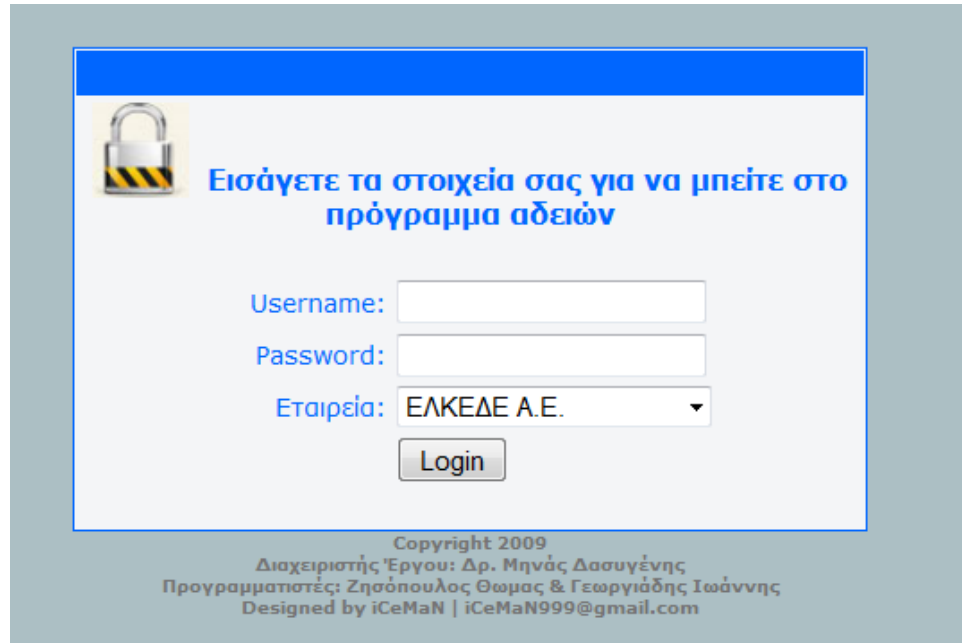
```
else if(mysql_num_rows($check)){  
    ?><script language="JavaScript">  
alert ("Το username υπάρχει ήδη! Παρακαλώ δώστε κάποιο άλλο  
Username");  
</script><?php }//else if
```

---

## 2.7 Λοιπές φόρμες

- Η φόρμα **εισόδου**(index. php):

Η αρχική φόρμα που βλέπει ο χρήστης. Σε αυτήν εισάγει τα στοιχεία που του έχουν δοθεί από τον διαχειριστή προκειμένου να κάνει login στην εφαρμογή.



Εισάγετε τα στοιχεία σας για να μπείτε στο πρόγραμμα αδειών

Username:

Password:

Εταιρεία: ΕΛΚΕΔΕ Α.Ε. ▼

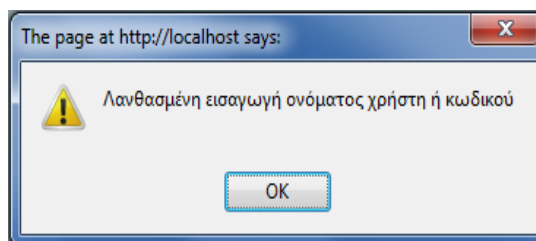
Login

Copyright 2009  
Διαχειριστής Έργου: Δρ. Μηνάς Δασυγένης  
Προγραμματιστές: Ζησόπουλος Θωμάς & Γεωργιάδης Ιωάννης  
Designed by iCeMaN | iCeMaN999@gmail.com

Εικόνα 2. 20 Φόρμα εισόδου

Αφού ο υπάλληλος εισάγει τα στοιχεία του , πρέπει να επιλέξει την εταιρία στην οποία ανήκει και στη συνέχεια να πατήσει το κουμπί «Login». Εφόσον έχουν εισήχθη τα στοιχεία του , φορτώνεται το αρχείο check. php , που συνδέεται στην κατάλληλη βάση δεδομένων ανάλογα με την εταιρία που έχει διαλέξει ο υπάλληλος. Στη συνέχεια ελέγχεται αν τα στοιχεία που δόθηκαν είναι σωστά και υπάρχουν στην βάση δεδομένων.

Στην περίπτωση που έχουν δοθεί λανθασμένα στοιχεία το σύστημα τον επαναφέρει στην φόρμα index. php και εμφανίζει μήνυμα λάθους.



Εικόνα 2. 21 Μήνυμα λάθους κατα την είσοδο

Στη περίπτωση που τα στοιχεία είναι σωστά τότε το σύστημα μεταφέρει τον υπάλληλο στην φόρμα home. php.

```
<?php
if(isset($_POST['submit'])) :
if($_POST['database']=="ΚΕΚ ΕΛΚΕΔΕ ΕΠΕ"){
    include ("ELKEDE/Config/kek-config. php");
}
else if($_POST['database']=="ΕΞΥΠΠ ΕΛΚΕΔΕ ΕΠΕ"){
    include ("ELKEDE/Config/exypp-config. php");
}
else if($_POST['database']=="ΕΛΚΕΔΕ Α. Ε. "){
    include("ELKEDE/Config/config. php");
}
session_start();
$username = strip_tags($_POST['username']);
$password = sha1(strip_tags($_POST['password']));
// Make the query a wee-bit safer
$query = sprintf("SELECT ID FROM members WHERE username = '%s'
AND      user_password      =      '%s'      LIMIT      1;",
mysql_real_escape_string($username),
mysql_real_escape_string($password));
$result = mysql_query($query);
if(1 != mysql_num_rows($result)) :
// MySQL returned zero rows (or there's something wrong with the query)
header('Location: index. php?msg=login_failed');

else :
// We found the row that we were looking for
$row = mysql_fetch_assoc($result);
// Register the user ID for further use
$_SESSION['member_ID'] = $row['ID'];
$sql = sprintf("SELECT name FROM employee WHERE UserID =
'$member_ID' ");
$_SESSION['UserID'] = $row['UserID'];
```

```
$_SESSION['database']= $_POST['database'];
header('Location: ELKEDE/home. php');
endif;
endif;
?>
```

---

- Η φόρμα **logout. php**:

Είναι η φόρμα αποσύνδεσης των χρηστών από την εφαρμογή , εκτελείται αυτομάτως μόλις κάποιος χρήστης πατήσει από το μενού την επιλογή «Αποσύνδεση» και τον μεταφέρει στην αρχική φόρμα(index. php).

Ο κώδικας που την υλοποιεί είναι ο ακόλουθος:

---

```
<?php
session_start();
if(true === session_unregister('member_ID')) :
header('Location: . . /index. php?msg=logout_complete');
else :
unset($_SESSION['member_ID']);
sleep(3);
header('Location: index. php?msg=logout_complete');
endif;?>
```

---

- Η φόρμα **μεθόδων**(functions. php):

Η φόρμα αυτή αποτελείται από όλες τις μεθόδους που χρησιμοποιούνται περισσότερες από μια φορά , ώστε να αποφεύγεται η επανάληψη τους στον κώδικα.

- Οι φόρμες **σύνδεσης**(config. php , exypp-config. php , kek-config. php):

Στις φόρμες αυτές υπάρχει ο κώδικας για να πραγματοποιηθεί επιτυχώς η σύνδεση με την κατάλληλη βάση ανάλογα με την εταιρία που έχει επιλεγεί στην φόρμα index. php.

Κώδικας σύνδεσης σε βάση δεδομένων

```
<?php
$connection = mysql_connect("server", "username", "password");
mysql_select_db("database_name", $connection);
if (!$connection) { die('Could not connect: ' . mysql_error()); }
?>
```

---

- Η φόρμα **εκτύπωσης**(adeia. pdf):  
Σκοπός της φόρμας αυτής είναι η εκτύπωση αδειών που έχουν ζητηθεί ή έχουν εγκριθεί με χρήση της ανάλογης φόρμας PDF. Για την μορφοποίηση και την εμφάνιση της φόρμας PDF χρησιμοποιήθηκαν οι βιβλιοθήκες και οι φόρμες του πακέτου TCPDF.  
[http://www.tecnick.com/public/code/cp\\_dpage.php?aiocp\\_dp=tcpdf](http://www.tecnick.com/public/code/cp_dpage.php?aiocp_dp=tcpdf)

## 2.8 Επίλογος

Στο δεύτερο κεφάλαιο αναφέρθηκαν εκτενώς οι πίνακες που δημιουργήθηκαν κατά την υλοποίηση της εφαρμογής και ο τρόπος χρήσης τους. Οι πίνακες αυτοί χωρίζονται σε δύο επιμέρους κατηγορίες. Σε αυτούς που καταγράφονται οι χρήστες και όλες οι άδειες προκειμένου να δημιουργηθεί η βάση δεδομένων της εταιρίας και σε αυτούς που τηρούν τα στατιστικά στοιχεία όλων των υπαλλήλων.

Αναλύθηκαν οι φόρμες εισαγωγής δεδομένων, δηλαδή αυτές που οι υπάλληλοι χρησιμοποιούν προκειμένου να ζητήσουν κάποια άδεια. Επίσης αναλύθηκαν οι φόρμες διαχείρισης της εφαρμογής από τον διαχειριστή, δηλαδή αυτές που ο διαχειριστής χρησιμοποιεί προκειμένου να κάνει όλες τις αναγκαίες μετατροπές που πιθανώς παρουσιαστούν, όπως για παράδειγμα η διαγραφή ενός υπαλλήλου από τη βάση δεδομένων της εταιρίας. Τέλος δόθηκαν οι φόρμες εισαγωγής και εξαγωγής (login-logout) από το σύστημα και ο κώδικας υλοποίησής τους.

Στο επόμενο κεφάλαιο θα αναλυθεί η γενική δομή της εταιρίας, όλα τα επίπεδα απασχόλησης των εργαζομένων της και θα επεξηγηθεί συνολικά η ροή της εφαρμογής. Συγκεκριμένα θα αναλυθεί ο τρόπος αίτησης και χορήγησης άδειας καθώς και διαδρομή που ακολουθεί η κάθε άδεια, μεταξύ των υπάρχοντων τμημάτων, προκειμένου τελικά να εγκριθεί.



### **3ο ΚΕΦΑΛΑΙΟ**

Δομή της εταιρίας – Ανάλυση ροής της εφαρμογής

### 3.1 Εισαγωγή

Στο κεφάλαιο αυτό αναφέρεται αναλυτικά η δομή της εταιρίας καθώς και η ιεραρχία των μελών της. Δίνεται ένα συνολικό οργανόγραμμα των τμημάτων και των βαθμίδων απασχόλησης των υπαλλήλων. Τέλος, αναλύεται όλη η διαδικασία παραχώρησης άδειας στους υπαλλήλους καθώς και τα στάδια που απαιτούνται για την έγκρισή της.

### 3.2 Η δομή των εταιρειών

Η εφαρμογή που αναπτύχθηκε ουσιαστικά δεν αναφέρεται σε μία εταιρία αλλά σε ένα σύνολο τριών εταιριών, μιας μητρικής και δύο θυγατρικών της. Κάθε εταιρία διαιρείται σε τέσσερα επίπεδα απασχόλησης. Οι υπάλληλοι των εταιριών πρέπει να επιλέξουν σε ποία εταιρία ανήκουν από το αρχικό μενού της εφαρμογής, πριν συνδεθούν στο πρόγραμμα, προκειμένου να τους χορηγηθεί κάποια άδεια.

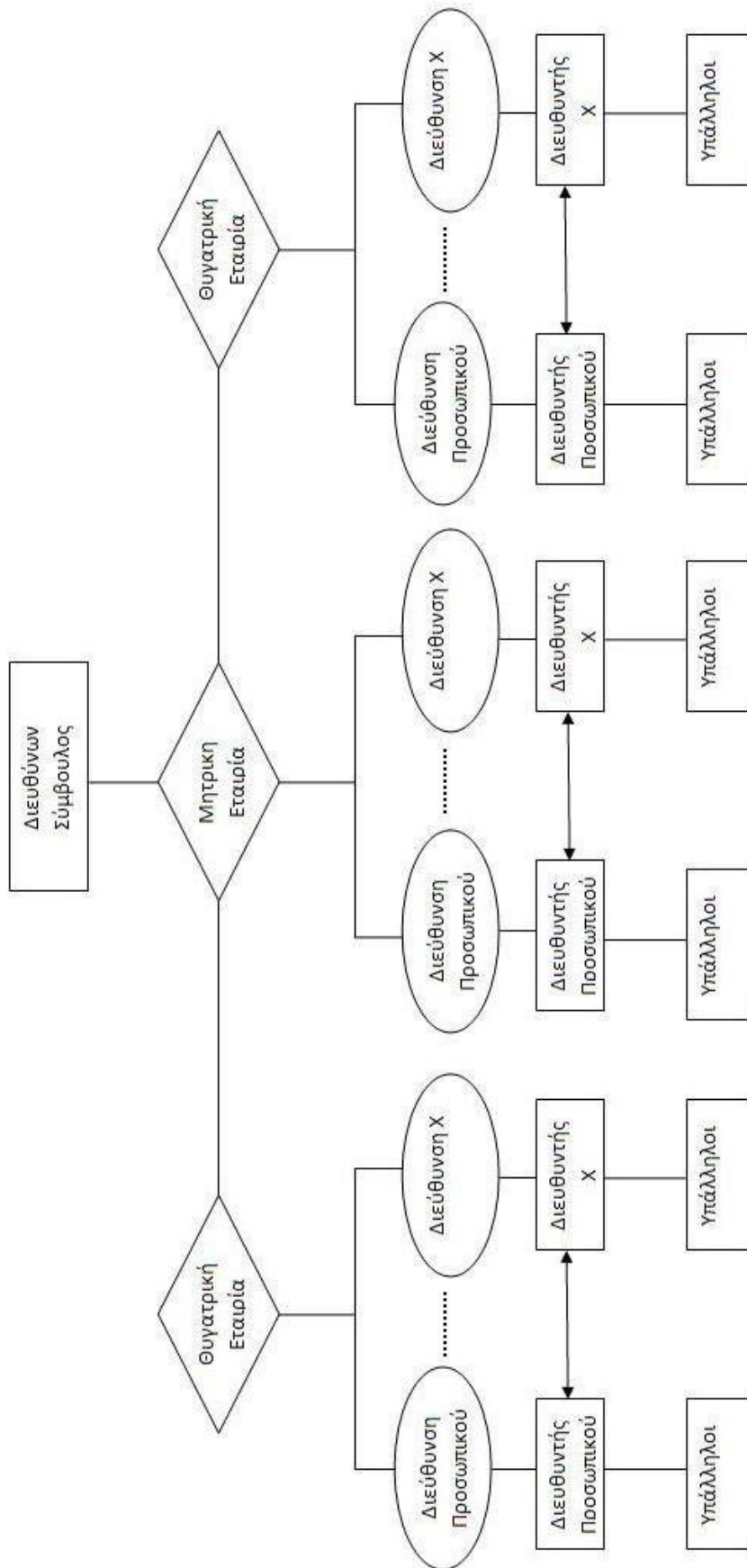
Όλες οι εταιρίες έχουν τον ίδιο διευθύνων σύμβουλο, ο οποίος φυσικά ανήκει στο ανώτερο επίπεδο της ιεραρχίας και δεν ζητάει να του χορηγηθούν άδειες. Ωστόσο συμμετέχει στην όλη διαδικασία της εφαρμογής καθώς είναι αυτός που τελικά εγκρίνει τις κανονικές άδειες όλων των υπαλλήλων, προκειμένου να τους χορηγηθούν. Τέλος, ο διευθύνων σύμβουλος έχει πρόσβαση στα στατιστικά των αδειών που χορηγούνται σε όλους τους υπαλλήλους, όλων των εταιριών.

Κάτω από το επίπεδο του διευθύνων συμβούλου βρίσκεται αυτό του διευθυντή προσωπικού. Η κάθε μια εταιρία έχει τον δικό της διευθυντή προσωπικού, ο οποίος είναι ο βασικός αρμόδιος χορήγησης αδειών για όλους τους υπαλλήλους της εταιρίας που ανήκει. Στο στάδιο αυτό εγκρίνονται και τελικά χορηγούνται όλες οι άδειες που έχουν ζητηθεί, με εξαίρεση την κανονική άδεια. Ειδικά για την άδεια αυτή απαιτείται επιπλέον έγκριση από τον διευθύνων σύμβουλο των εταιριών, όπως έχει ήδη αναφερθεί, και στη διαδικασία έγκρισής της συμμετέχει κανονικά και ο διευθυντής προσωπικού.

Ένα επίπεδο πιο κάτω βρίσκονται οι προϊστάμενοι των τμημάτων των εταιριών. Σε κάθε εταιρία υπάρχουν τρία τμήματα που το καθένα έχει τον δικό του προϊστάμενο. Οι προϊστάμενοι εγκρίνουν αρχικά τις άδειες των υπαλλήλων των τμημάτων τους. Καμία άδεια δεν θα προωθηθεί σε ανώτερο επίπεδο εάν προηγουμένως δεν έχει εγκριθεί από τον αρμόδιο προϊστάμενο. Οι προϊστάμενοι με τη σειρά τους είναι υποχρεωμένοι να ζητούν έγκριση όλων των αδειών τους από τα ανώτερα επίπεδα της ιεραρχίας.

Στο κατώτερο επίπεδο της ιεραρχίας βρίσκονται οι απλοί υπάλληλοι, οι οποίοι αποτελούν και τον κύριο όγκο των μελών των εταιριών. Οι άδειες των απλών υπαλλήλων πρέπει να περάσουν από όλα τα στάδια της ιεραρχίας προκειμένου να εγκριθούν. Όπως είναι φυσικό, οι απλοί υπάλληλοι δεν έχουν καμία αρμοδιότητα έγκρισης άδειας.

### 3.3 Οργανόγραμμα



### 3.4 Ο χρήστης administrator-διαχειριστής

Πέρα από τα τέσσερα επίπεδα απασχόλησης, για τις ανάγκες της εφαρμογής, έχει δημιουργηθεί ένα επιπλέον επίπεδο, αυτό του administrator. Στην πραγματικότητα δεν πρόκειται για υπαρκτό επίπεδο στη δομή της εταιρίας, αλλά για έναν υπάλληλο που σκοπό έχει την διαχείριση της εφαρμογής χορήγησης αδειών. Τα καθήκοντα του administrator μπορεί να τα αναλάβει οποιοσδήποτε υπάλληλος του ανώτερου επιπέδου απασχόλησης που έχει επιλεγεί από τη διοίκηση της εταιρίας. Ο administrator είναι επιπλέον καταχωρημένος στη βάση της εφαρμογής με την τυπική ιδιότητα του υπαλλήλου της εταιρίας και με βάση αυτή συμμετέχει κανονικά στη διαδικασία έγκρισης-χορήγησης άδειας.

Ο χρήστης administrator έχει δύο βασικές αρμοδιότητες όσον αφορά τους υπαλλήλους της εταιρίας:

#### 1. Την καταχώρηση και επεξεργασία των υπαλλήλων στη βάση.

Όταν ένας νέος υπάλληλος προληφθεί στην εταιρία ο administrator καταχωρεί τα στοιχεία του στη βάση και του παραχωρεί ένα username και έναν κωδικό για την πρόσβαση του στην εφαρμογή. Επίσης καταχωρεί στη σύστημα των αριθμό των ημερών κανονικής άδειας που του αναλογούν, καθώς οι διαθέσιμες μέρες κανονικής άδειας των υπαλλήλων ποικίλλουν ανάλογα με τη θέση και τα χρόνια απασχόλησης στην εταιρία. Ακόμη ο administrator έχει τη δυνατότητα να επεξεργάζεται τους λογαριασμούς των ήδη καταχωρημένων υπαλλήλων ώστε να μπορεί να διαχειρίζεται διάφορες πιθανές αλλαγές όπως κάποια προαγωγή ή αλλαγή των διαθέσιμων ημερών κανονικής άδειας. Τέλος έχει τη δυνατότητα να διαγράφει οριστικά κάποιον υπάλληλο όταν αυτός απομακρυνθεί από την εταιρία.

#### 2. Την επεξεργασία των τμημάτων των εταιριών.

Ο administrator έχει τη δυνατότητα να δημιουργήσει ένα νέο τμήμα σε μια ή σε όλες τις εταιρίες και να καταχωρήσει σε αυτό τον νέο προϊστάμενό του. Επίσης μπορεί να αλλάξει τον προϊστάμενο ή και το όνομα ενός υπάρχοντος τμήματος. Η επιλογή των προϊσταμένων γίνεται από τους ήδη καταχωρημένους υπαλλήλους των εταιριών. Επομένως, σε περίπτωση που ο προϊστάμενος που έχει επιλεγεί είναι κάποιος νέος υπάλληλος θα πρέπει πρώτα να καταχωρηθεί στο σύστημα και στη συνέχεια να καταχωρηθεί στη στο τμήμα.

Εκτός από τις αρμοδιότητες που αφορούν τους υπαλλήλους, ο administrator έχει άλλες δύο, πολύ βασικές, που αφορούν το σύνολο της εταιρίας. Πρώτη από αυτές είναι η διαγραφή ενός ολόκληρου τμήματος. Με αυτόν τον τρόπο η διεύθυνση του τμήματος σβήνεται οριστικά από τη βάση δεδομένων, όχι όμως και οι υπάλληλοι που την απαρτίζουν. Για να γίνει αυτό όπως και κάθε αλλαγή πρέπει να γίνεται μεμονωμένα από τον administrator.

### 3.5 Ροή των αδειών

Για τη χορήγηση μιας άδειας έχει δημιουργηθεί στη βάση της εφαρμογής μια μεταβλητή stage που αντιστοιχεί στην κατάσταση που βρίσκεται η αίτηση κάποιας άδειας. Δηλαδή εάν έχει εγκριθεί, είναι προς έγκριση ή έχει

απορριφθεί. Επίσης η μεταβλητή αυτή δείχνει σε ποιο επίπεδο απασχόληση βρίσκεται η προς έγκριση άδεια.

Όταν ένας απλός υπάλληλος ζητήσει κανονική άδεια η μεταβλητή stage παίρνει την τιμή μηδέν(0). Εάν η άδεια εγκριθεί από τον προϊστάμενο η μεταβλητή αλλάζει στην τιμή ένα(1) και ανεβαίνει στο επόμενο επίπεδο απασχόλησης. Στη συνέχεια αλλάζει στην τιμή δύο(2) και φτάνει στο ανώτερο επίπεδο(του διευθύνον συμβούλου) όπου εάν εγκριθεί παίρνει την τιμή τρία(3) ενώ εάν απορριφθεί την τιμή τέσσερα(4). Ταυτόχρονα στέλνεται ένα e-mail στον υπάλληλο για να ενημερωθεί εάν η άδειά του έχει εγκριθεί η απορριφθεί. Η ίδια διαδικασία ακολουθείται και για όλους τους υπόλοιπους τύπους άδειας με μόνη διαφορά ότι δεν υπάρχει η κατάσταση (stage) δύο. Ο λόγος είναι ότι για όλους τους άλλους τύπους άδειας δεν χρειάζεται έγκριση από τον διευθύνον σύμβουλο οπότε κάποια ενδιάμεση κατάσταση θα πρέπει να παρακάμπτεται.

Τέλος, όταν κάποιος προϊστάμενος ζητήσει άδεια η μεταβλητή stage παίρνει αρχικά την τιμή ένα(1) και όχι μηδέν(0). Η υπόλοιπη διαδικασία για όλες τις άδειες είναι αντίστοιχη αυτής του απλού υπαλλήλου.

### 3.6 Λοιπές δυνατότητες του συστήματος

Εκτός από τη χορήγηση άδειας που αποτελεί και το βασικό σκοπό της εφαρμογής, παρέχονται στους υπαλλήλους ορισμένες ακόμα δυνατότητες ανάλογα με το επίπεδο απασχόλησης τους. Βασικότερη όλων, που αναφέρεται και στο σύνολο της εφαρμογής, είναι η τήρηση στατιστικών στοιχείων για όλες τις άδειες που έχουν ζητηθεί από όλους τους υπαλλήλους, για μεγάλα χρονικά διαστήματα. Ο διευθύνον σύμβουλος είναι ο μόνος που έχει πρόσβαση στα στατιστικά αυτά μέσω του κυρίως μενού του. Με αυτόν τον τρόπο έχει τη δυνατότητα να ελέγχει ατομικά τις άδειες που ζητούνται καθώς επίσης και τη χρονική περίοδο που ζητούνται. Τα στατιστικά αυτά διαγράφονται μόνο από τον administrator και μόνο μετά από αίτηση του διευθύνον συμβούλου.

Επιπλέον ο κάθε υπάλληλος έχει τη δυνατότητα να ελέγχει πόσες και ποιές άδειες ζήτησε. Όλες οι άδειες που ζητούνται, καταγράφονται σε ετήσια βάση και προβάλλονται στο μενού του προσωπικού ιστορικού αδειών. Εκεί οι υπάλληλοι μπορούν να δουν πόσες και ποιές άδειές τους έχουν εγκριθεί. Επίσης, μπορούν να μάθουν τα όρια ημερών καθώς και το υπόλοιπο των ημερών που δικαιούνται για κάθε τύπο άδειας. Τέλος, έχουν τη δυνατότητα να ενημερώνονται για τις άδειες που βρίσκονται κάθε στιγμή σε αναμονή.

Μια τελευταία δυνατότητα που παρέχει η εφαρμογή σε όλους τους υπαλλήλους είναι η επεξεργασία του λογαριασμού τους, δηλαδή η αλλαγή κάποιων προσωπικών στοιχείων που έχουν καταχωρηθεί από τον administrator. Αυτά είναι η κωδικός πρόσβασης στην εφαρμογή και το e-mail τους. Για τα υπόλοιπα στοιχεία, όπως το username και η διεύθυνση στην οποία ανήκει, η τυχών αλλαγή μπορεί να γίνει μόνο από τον administrator.

### 3.7 Επίλογος

Στο τρίτο κεφάλαιο αναλύθηκε η δομή της εταιρίας και δόθηκε το πλήρες οργανόγραμμα λειτουργίας της. Επίσης εξηγήθηκε εκτενώς ο τρόπος αίτησης και παραχώρησης οποιασδήποτε άδειας καθώς και η διαδικασία της ροής μιας αίτησης στα διάφορα επίπεδα απασχόλησης μέχρις ότου εγκριθεί οριστικά. Τέλος καταγράφηκαν όλες οι επιμέρους δυνατότητες που παρέχει η εφαρμογή στους χρήστες όπως το προσωπικό ιστορικό αδειών των υπαλλήλων και τα συνολικά στατιστικά αδειών που διαχειρίζεται ο διευθύνων σύμβουλος των εταιριών.

Στο επόμενο κεφάλαιο αναφέρονται όλα τα προβλήματα που παρουσιάστηκαν κατά την εκπόνηση της εργασίας και αναλύεται ο τρόπος επίλυσής τους. Επίσης αναφέρονται όλα τα ζητήματα που αφορούν την ασφάλεια της εφαρμογής καθώς και ο τρόπος διασφάλισης της σωστής λειτουργίας της.

## 4ο ΚΕΦΑΛΑΙΟ

Προβλήματα που αντιμετωπίστηκαν και Ασφάλεια εφαρμογής

## 4.1 Προβλήματα που αντιμετωπιστήκαν

### 4.1.1 Εισαγωγή

Κατά την εκπόνηση της πτυχιακής εργασίας παρουσιάστηκαν και αντιμετωπίστηκαν ορισμένα βασικά προβλήματα. Τα προβλήματα αυτά κατατάσσονται σε δύο κατηγορίες. Αυτά που αφορούν το σύνολο της εργασίας και τον τρόπο εκπόνησής της και αυτά που αφορούν κομμάτια του κώδικα και γενικά της εφαρμογής.

### 4.1.2 Προβλήματα που αφορούν στο σύνολο της εργασίας

Πρωταρχικό πρόβλημα που παρουσιάστηκε και αναφέρεται στο σύνολο της εφαρμογής ήταν η ανάγκη χρήσης της γλώσσας PHP. Η συγκεκριμένη γλώσσα δεν αποτελεί αντικείμενο σπουδών του τμήματος και επομένως δεν υπήρχε η απαραίτητη γνώση για τη χρήση της. Για την κατανόηση και την εξοικείωση με τη γλώσσα συγκεντρώθηκε και χρησιμοποιήθηκε υλικό από διάφορα βιβλία καθώς και από το διαδίκτυο. Αναζητήθηκαν και χρησιμοποιήθηκαν έτοιμα κομμάτια κώδικα, προσαρμοσμένα στις ανάγκες της εφαρμογής.

Ένα ακόμα πρόβλημα ήταν το γεγονός ότι η εργασία αποτελεί πραγματική εφαρμογή για κάποια υπαρκτή εταιρία. Η εταιρία αυτή διαθέτει ήδη το δικό της σύστημα αίτησης και παραχώρησης άδειας και η εφαρμογή θα έπρεπε να προσαρμοστεί στο πρότυπο αυτό. Για το λόγο αυτό όταν η εφαρμογή υλοποιήθηκε εξ ολοκλήρου σύμφωνα με τις αρχικές κατευθύνσεις που δόθηκαν, το πρόγραμμα δόθηκε στην εταιρία ώστε να αξιολογηθεί στο σύνολό του.

Κατά την αξιολόγηση αυτή οι υπεύθυνοι της εταιρίας αποφάσισαν να ζητήσουν αλλαγές καθώς και επιπλέον επιλογές στην εφαρμογή. Ενδεικτικά ζητήθηκαν επιπλέον τύποι άδειας που δεν ήταν διαθέσιμοι στο υπάρχον σύστημα της εταιρίας και δεν είχαν ζητηθεί στο αρχικό πλάνο της εργασίας. Οι αλλαγές αυτές είχαν επίπτωση σε ολόκληρη την υλοποίηση της εφαρμογής καθώς θα έπρεπε να προστεθούν επιπλέον πεδία τους υπάρχοντες πίνακες και να αλλαχθούν οι συσχετίσεις μεταξύ των πινάκων.

Το πρόβλημα αυτό παρουσιάστηκε ορισμένες επιπλέον φορές σε όλη τη διάρκεια υλοποίησης της εφαρμογής και γενικότερα η έλλειψη συγκεκριμένου πλάνου απαιτήσεων από την εταιρία καθυστέρησε αισθητά όλη τη διαδικασία εκπόνησης και ολοκλήρωσης της εφαρμογής και κατ' επέκταση και της πτυχιακής εργασίας



### 4.1.3 Προβλήματα που αφορούν στο κώδικα

Όσον αφορά τον κώδικα και γενικότερα την εφαρμογή, βασικότερο πρόβλημα που αντιμετωπίστηκε ήταν η διατήρηση της ποιότητας της βάσης δεδομένων της εφαρμογής. Επειδή οι εταιρίες που συμμετέχουν στο πρόγραμμα είναι τρείς και υπάρχει μεγάλος όγκος υπαλλήλων , η βάση φορτώνεται με πολλά στοιχεία που θα την καθιστούσαν αργή. Για τον λόγο αυτό εφαρμογή δίνει τη δυνατότητα διαγραφής παλιών αδειών από τη βάση. Ο τρόπος διαγραφής σχετίζεται και με την ασφάλεια της εφαρμογής και αναλύεται παρακάτω.

Ένα επιπλέον πρόβλημα ήταν η διαγραφή των διαθέσιμων ημερών κανονικής άδειας κάθε υπαλλήλου. Κάθε υπάλληλος έχει το δικαίωμα να χρησιμοποιήσει μέρος των ημερών κανονικής άδειας προηγούμενου έτους στο επόμενο εφόσον δεν έχει υπερβεί το διαθέσιμο όριό τους. Για το λόγο αυτό, σε αντίθεση με όλες τις υπόλοιπες άδειες που μηδενίζονται αυτόματα κάθε έτος, οι διαθέσιμες μέρες κανονικής άδειας επιλέχθηκε να μηδενίζονται από τον διαχειριστή. Με τον τρόπο αυτό πετυχαίνεται η διατήρηση των δικαιωμάτων των υπαλλήλων στην κανονική τους άδεια.

## 4.2 Ασφάλεια εφαρμογής

### 4.2.1 Εισαγωγή

Κατά την υλοποίηση της εφαρμογής παρουσιάστηκαν ορισμένα ζητήματα που είχαν να κάνουν με την ασφάλειά της. Αυτά τα ζητήματα έπρεπε να επιλυθούν προκειμένου να διασφαλιστεί η σωστή λειτουργία της εφαρμογής. Επίσης, έπρεπε κατά τη διάρκεια της υλοποίησης να προβλεφτούν όλες οι πιθανότητες λανθασμένης χρήσης του προγράμματος από τους χρήστες. Με αυτόν τον τρόπο δημιουργήθηκε ένα πλαίσιο καθοδήγησής τους από την ίδια την εφαρμογή, ώστε να πετυχαίνεται η ορθή χρήση της.

### 4.2.2 Εισαγωγή των χρηστών στην εφαρμογή.

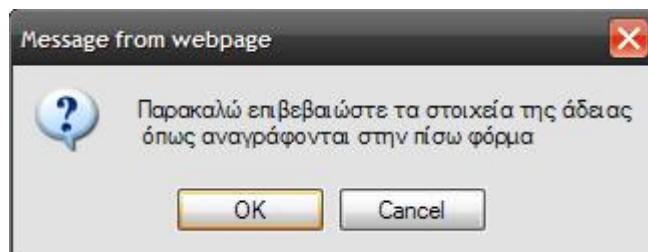
Κάθε υπάλληλος των τριών εταιριών διαθέτει προσωπικό username και κωδικό. Τα δύο αυτά πεδία είναι , ασφαλώς , μοναδικά στη βάση για κάθε χρήστη. Ο κωδικός του κάθε υπαλλήλου κωδικοποιείται επαρκώς (με χρήση της κωδικοποίησης md5 ) κατά την καταχώρησή του στη βάση προκειμένου να μη μπορεί να υποκλαπεί.

ID	username	user_password
1	Administrator	d033e22ae348aeb5660fc2140aec35850c4da997
2	dzapr	d354db7a7a19f96cdb30b54d5e4349d3d47598c5
3	tzisop	3d02c9cf254d0a907f93a7f36bc35a048d8e2d31
4	gpag	0e2defc5afa4e2d1c88dcc7b8921c2c11c7f53e9
5	tgjak	3573c3a96e06af39dfc671bacb2e5ced72ae4f0b
6	ageo	9341b8893b379aa5b5fddfd5fa7ca372ce01c8ce
7	mpet	1b2144acd7a7678aad1e14382a93eb52a3a25bae
8	tpar	80258530ca82372d2504c80aba4c4ab4b79612b2
9	tkar	c010ec6406c07cf7bb42ad62036b55993fbc0511
10	gkur	c66807257970070cf92ba035ed45e96be7e48c24
11	igeo	8d0c1e2454136e536605d5aeb81997044524a560
12	etou	21d329172f54543449e9d7fc322c4c6d39236f15
13	aman	8ad73f92d9f9d8ffbb8fcc019cab870008ac555c
14	smaur	f480581728888084ee56e10ea73c0b76cd0d5f13
15	mkout	b09166ee231346f82b5cea5eedfc1ea909dc6cf6

Εικόνα 4-1 Στοιχεία χρηστών

#### 4.2.3 Ορθή συμπλήρωση των ημερομηνιών αίτησης αδειών.

Κατά την συμπλήρωση των πεδίων των ημερομηνιών χρησιμοποιήθηκε η επιλογή drop down menu ώστε να αποφεύγεται η καταχώρηση ημερομηνιών που δεν υπάρχουν(παράδειγμα, 32/01/2011). Η ίδια πρακτική υιοθετήθηκε και στην επιλογή των αδειών ώστε να αποφεύγονται τα ορθογραφικά λάθη. Τέλος, ο χρήστης ενημερώνεται με κατάλληλο μήνυμα και δεν μπορεί να προχωρήσει παρακάτω εάν δεν έχει συμπληρώσει όλα τα απαιτούμενα πεδία.



Εικόνα 4-2 Μήνυμα λαθους

#### 4.2.4 Ασφάλεια των επιλογών στο μενού του διαχειριστή.

Είναι το κύριο σημείο αντιμετώπισης προβλημάτων ασφάλειας καθώς ο διαχειριστής είναι υπεύθυνος για όλες τις καταχωρήσεις χρηστών καθώς και όλες τις αλλαγές που σχετίζονται με το σύνολο της βάσης .

Κατά την καταχώρηση χρηστών η εφαρμογή υποχρεώνει τον διαχειριστή να συμπληρώνει όλα τα πεδία που αφορούν τον χρήστη ώστε να αποφεύγεται η τήρηση ελλείπων δεδομένων. Σε αντίθετη περίπτωση σταματά η διαδικασία εγγραφής με ταυτόχρονη εμφάνιση κατάλληλου μηνύματος ενημέρωσης του διαχειριστή.

Κατά την επεξεργασία τόσο των λογαριασμών των υπαλλήλων όσο και των τμημάτων της εταιρίας η εφαρμογή, με κατάλληλα μηνύματα ζητάει από τον χρήστη την επιβεβαίωση των επιλογών του.

Επίσης, ζητούνται όλα τα απαραίτητα στοιχεία ώστε η εφαρμογή να προχωρήσει στο επόμενο στάδιο.



Εικόνα 4-3 Μήνυμα Επιβεβαίωσης

Στο μενού του διαχειριστή υπάρχει η καρτέλα διαχείρισης της εφαρμογής(administrator panel) στην οποία είναι συγκεντρωμένες όλες οι λειτουργίες διαχείρισης της βάσης. Για τη γενική ασφάλεια της εφαρμογής έχει επιλεγεί ο διαχειριστής να είναι ο μόνος υπάλληλος που έχει πρόσβαση στην καρτέλα αυτή.

Οι δύο λειτουργίες της καρτέλας διαχείρισης είναι οι εξής:

- Διαγραφή διευθύνσεων

Εάν κάποια στιγμή κριθεί απαραίτητη η διαγραφή κάποιας διεύθυνσης, ο διαχειριστής(και μόνο αυτός) έχει τη δυνατότητα να τη διαγράψει, επιλέγοντάς την και πατώντας το πλήκτρο διαγραφή. Για τη αποφυγή λάθους η εφαρμογή ρωτάει τον διαχειριστή εάν είναι σίγουρος για την επιλογή. Μέτα τη διαγραφή της διεύθυνσης το τμήμα δεν υπάρχει αλλά οι υπάλληλοι που το απάρτιζαν δε διαγράφονται αυτόματα. Η διαγραφή ή η μεταφορά τους σε άλλο τμήμα γίνεται από την επεξεργασία χρηστών στο μενού του διαχειριστή. Ο λόγος οι υιοθετήθηκε αυτή η πρακτική είναι ότι δε μπορούμε να ξέρουμε πόσοι και ποιοί υπάλληλοι του τμήματος θα εξακολουθήσουν να εργάζονται στην εταιρία ούτε και σε ποιοό πόστο.

- Διαγραφή αδειών

Όταν κριθεί απαραίτητο λόγο υπερφόρτωσης της βάσης και προκειμένου να γίνεται ταχύτερα η προσπέλασή της, ο διαχειριστής μπορεί να διαγράψει όλες ή μέρος των εγγεγραμμένων αδειών. Για στατιστικούς λόγους έχει ζητηθεί από την εταιρία να τηρούνται αρχεία με όλες τις παλιές άδειες. Για τον λόγο αυτό έχει επιλεγεί η διαγραφή των αδειών από τον διαχειριστή(αντί για τη χρήση κάποιου trigger) ώστε αυτές ταυτόχρονα να μεταφέρονται στη βάση τήρησης στατιστικών στοιχείων(backup).

### 4.3 Επίλογος

Στο τέταρτο κεφάλαιο αναφέρθηκαν όλα τα προβλήματα που παρουσιάστηκαν κατά την υλοποίηση της εφαρμογής καθώς και τρόπος επίλυσής τους. Τέλος αναλύθηκαν όλα τα ζητήματα που αποτελούσαν πιθανότητα λανθασμένης χρήσης του προγράμματος από τους χρήστες και ο τρόπος αντιμετώπισής τους, προκειμένου να διασφαλιστεί η ορθή λειτουργία της.

# ΠΑΡΑΡΤΗΜΑΤΑ

## ΠΑΡΑΡΤΗΜΑ Α

Παρατίθεται ο κώδικας του αρχείου “adeia.php” με τον οποίο καταχωρούνται όλοι οι τύποι αδειών στην βάση δεδομένων. Επίσης ο κώδικας που εμφανίζει τα πεδία που πρέπει να συμπληρωθούν ανάλογα με τον τύπο της άδειας που έχει επιλεγεί. Τέλος, δίνεται ο κώδικας που ελέγχει την ορθή συμπλήρωση αδειών από τον χρήστη.

```
<?php
session_start();
if(!isset($_SESSION['member_ID'])):
header('Location: . . /index. php');
endif;

$member_ID=$_SESSION['member_ID'];
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1. 0 Transitional//EN" "http://www. w3.
org/TR/xhtml1/DTD/xhtml1-transitional. dtd">
<html xmlns="http://www. w3. org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
if($_SESSION['database']=="ΕΛΚΕΔΕ Α. Ε. ") {?>
<link rel="stylesheet" href="Styles/style. css" type="text/css" media="screen" />
<?php
include ("Config/config. php");
}
else if ($_SESSION['database']=="ΕΞΥΠΠ ΕΛΚΕΔΕ ΕΠΕ") {?>
<link rel="stylesheet" href="Styles/style2. css" type="text/css" media="screen" />
<?php
include ("Config/exypp-config. php");
}
else if ($_SESSION['database']=="ΚΕΚ ΕΛΚΕΔΕ ΕΠΕ") {?>
<link rel="stylesheet" href="Styles/style3. css" type="text/css" media="screen" />
<?php
include ("Config/kek-config. php");
}

include ("functions. php");
$id=$_SESSION['member_ID'];
$query=mysql_query("SELECT groups FROM employee WHERE UserID='$id'");
$row=mysql_fetch_assoc($query);
if( $row['groups'] == 2 || $row['groups'] == 0 || $row['groups']==1 ) {
// Έλεγχος για το αν ο χρήστης που εισέρχεται είναι ο Administrator
header('Location: . . /index. php');
} //if
?>
<link href="images/tab. gif" rel="shortcut icon" type="image/vnd. microsoft. icon" />
<title>Πρόγραμμα αδειών --- Καταχώρηση άδειας</title>
</head>

<?php
if(isset($_POST['submit'])) {

$yday=$_POST['startday'];
$smmonth=$_POST['startmonth'];
$year=$_POST['startyear'];
```

```

$eday=$_POST['endday'];
$emonth=$_POST['endmonth'];
$eyear=$_POST['endyear'];
$shour=$_POST['starthour'];
$ehour=$_POST['endhour'];
$query=mysql_query("SELECT groups FROM employee WHERE UserID='$member_ID'");
$row=mysql_fetch_assoc($query);
$group=$row['groups'];
$diff=@get_time_difference($shour, $ehour);
$dif=$diff['hours']*60+$diff['minutes'];
$type=$_POST['type'];
$exc=$_POST['excuse'];
$pday=$_POST['privateday'];
$pmmonth=$_POST['privatemonth'];
$pyear=$_POST['privateyear'];
$sumdays=$_POST['sumdays'];
$bday=$_POST['dayback'];
$bmonth=$_POST['monthback'];
$byear=$_POST['yearback'];

$check=true;

$sdate=$year . '/' . $smmonth . '/' . $sday;
$edate=$year . '/' . $emmonth . '/' . $eday;
$pdate=$pyear . '/' . $pmmonth . '/' . $pday;
$bdate=$byear . '/' . $bmonth . '/' . $bday;

$dayarray=date_diff2($sdate, $edate);
$datedays=$dayarray[0]+1;
$now=date('Y/m/d');
$x=date_diff2($sdate, $now);
$difference=$x[0]+1;

$x2=date_diff2($pdate, $now);
$difference2=$x2[0]+1;

if($type=="KANONIKH"){
$sql= mysql_query("SELECT days FROM adeies WHERE UserID='$member_ID' AND type='$type'");
$row = mysql_fetch_array($sql);
$days=$row[0];

$sql2= mysql_query("SELECT normaldays FROM employee WHERE UserID='$member_ID'");
$row = mysql_fetch_array($sql2);
$maxdays=$row[0];
$sum=$maxdays-$days;

if ( $type==" " || $sday==" " || $smmonth==" " || $syear==" " || $eday==" " || $emmonth==" " || $eyear==" " || $bday==" "
|| $bmonth==" " || $byear==" "){
$check=false;?>
<script language="JavaScript">
alert ("Πρέπει να εισάγετε όλα τα στοιχεία, Προσπαθήστε ξανά")
</script>
<?php }//if null

```

```
else if($difference>10){
$check=false;?>
<script language="JavaScript">
alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα.
Προσπαθείστε ξανά")
</script>
<?php }//if
else if($datedays<0 ){
$check=false;?>
<script language="JavaScript">
alert ("Λάθος Ημερομηνία | Εισάγετε νεα ημερομηνία")
</script>
<?php }//if

else if($datedays>$sum){
$check=false;?>
<script language="JavaScript">
alert ("Έχετε υπερβεί το όριο ημερών σας | Εισάγετε νεα ημερομηνία")
</script>
<?php }// if

}//if

else if($type=="ΠΡΟΣΩΠΙΚΗ"){
$sql= mysql_query("SELECT days FROM adeies WHERE UserID='$member_ID' AND type='$type'");
$row = mysql_fetch_array($sql);
$days=$row[0];

$sql2= mysql_query("SELECT private_min FROM employee WHERE UserID='$member_ID'");
$row = mysql_fetch_array($sql2);
$maxmin=$row[0];
$sum=$maxmin-$days;
$mins=$dif;

if ( $type==" " || $pday==" " || $pmonth==" " || $pyear==" "){
$check=false;?>
<script language="JavaScript">
alert ("Πρέπει να εισάγετε όλα τα στοιχεία, Προσπαθήστε ξανά")
</script>
<?php }//if null
else if($difference2>10){
$check=false;?>
<script language="JavaScript">
alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα.
Προσπαθείστε ξανά")
</script>
<?php }//if

else if($mins>$sum){
$check=false;?>
<script language="JavaScript">
alert ("Έχετε υπερβεί το όριο ωρών σας | Εισάγετε νεα ημερομηνία")
</script>
<?php }//if
```



```

} //if

else if($type=="ΕΡΓΑΣΙΑΣ" || $type=="ΡΕΠΟ"){
    $sql= mysql_query("SELECT days FROM adeies WHERE UserID='$member_ID' AND type='$type'");
    $row = mysql_fetch_array($sql , MYSQL_NUM);
    $mins=$dif;

    if ( $type==" " || $pday==" " || $pmonth==" " || $pyear==" " || $exc==" " || $shour==" " || $ehour==" "){
        $check=false;?>
        <script language="JavaScript">
        alert ("Πρέπει να εισάγετε όλα τα στοιχεία. Προσπαθήστε ξανά")
        </script>
        <?php } //if null
    else if($difference>10){
        $check=false;?>
        <script language="JavaScript">
        alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα.
        Προσπαθείστε ξανά")
        </script>
        <?php } //if
    } //else if

    else if($type=="ΕΡΓΑΣΙΑΣ ΩΡΩΝ"){
        $sql= mysql_query("SELECT days FROM adeies WHERE UserID='$member_ID' AND type='$type'");
        $row = mysql_fetch_array($sql , MYSQL_NUM);
        $mins=$dif;

        if ( $type==" " || $pday==" " || $pmonth==" " || $pyear==" " || $exc==" " || $shour==" " || $ehour==" "){
            $check=false;?>
            <script language="JavaScript">
            alert ("Πρέπει να εισάγετε όλα τα στοιχεία. Προσπαθήστε ξανά")
            </script>
            <?php } //if null
        else if($difference2>10){
            $check=false;?>
            <script language="JavaScript">
            alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα.
            Προσπαθείστε ξανά")
            </script>
            <?php } //if
        } //else if

        else {
            $sql= mysql_query("SELECT days FROM adeies WHERE UserID='$member_ID' AND type='$type'");
            $row = mysql_fetch_array($sql , MYSQL_NUM);
            $days=$row[0];

            $sql2= mysql_query("SELECT maxdays FROM vacation_type WHERE type='$type'");
            $row = mysql_fetch_array($sql2 , MYSQL_NUM);
            $maxdays=$row[0];
            $sum=$maxdays-$days;

            if ( $type==" " || $sday==" " || $smmonth==" " || $syear==" " || $eday==" " || $emonth==" " || $eyear==" " ){
                $check=false;?>
                <script language="JavaScript">

```

```

alert ("Πρέπει να εισάγετε όλα τα στοιχεία, Προσπαθήστε ξανά")
</script>
<?php }//if null

else if($difference>10){
$check=false;?>
<script language="JavaScript">
alert ("Η ημερομηνία έναρξης πρέπει να είναι το περισσότερο 10 μέρες πριν την σημερινή ημέρα.
Προσπαθείστε ξανά")
</script>
<?php }//if

else if($datedays<0 ){
$check=false;?>
<script language="JavaScript">
alert ("Λάθος Ημερομηνία | Εισάγετε νέα ημερομηνία")
</script>
<?php }//if

else if($datedays>$sum){
$check=false;?>
<script language="JavaScript">
alert ("Έχετε υπερβεί το όριο ημερών σας | Εισάγετε νέα ημερομηνία")
</script>
<?php }// if

}//else

//-----//
-----//
if($check==true){
if($group=='3'){
if ( $type=="ΕΡΓΑΣΙΑΣ ΩΡΩΝ" || $type=="ΠΡΟΣΩΠΙΚΗ" || $type=="ΡΕΠΟ" ){
$query="INSERT INTO vacation ( Req_by , type , Start_day , End_day , dayssum , hourssum , Reason ,
Stage , Start_hour , End_hour ) VALUES ( '$member_ID' , '$type' , '$pdate' , '$pdate' , '$datedays' , '$dif' ,
'$exc' , 1 , '$shour' , '$ehour' );
mysql_query($query);
if($days!=""){
$newmins=$mins+$days;
mysql_query("UPDATE adeies SET days='$newmins' WHERE UserID='$member_ID' AND type='$type' ") or
die("Error, UDPATE Into database FAILED");
}//if

else{
mysql_query("INSERT INTO adeies ( UserID , type , days ) VALUES ( '$member_ID' , '$type' , '$mins' )") or
die("Error, Insert Into database FAILED");
}//else

}//if

else{
$query="INSERT INTO vacation ( Req_by , type , Start_day , End_day , dayssum , Reason , Stage ,
Start_hour , End_hour , days , Back_day) VALUES ( '$member_ID' , '$type' , '$sdate' , '$edate' ,
'$datedays' , '$exc' , 1 , '$shour' , '$ehour' , '$sumdays' , '$bdate' )";
mysql_query($query);
if(isset($days)){
$newday=$datedays+$days;

```

```

mysql_query("UPDATE adeies SET days='$newday' WHERE UserID='$member_ID' AND type='$type' ") or
die('Error, UDPATE Into database FAILED');
} //if

else{
mysql_query("INSERT INTO adeies ( UserID , type , days ) VALUES ( '$member_ID' , '$type' ,
'$datedays' )") or die('Error, Insert Into database FAILED');
} //else
} //else

?>
<script language="JavaScript">
alert ("Η αιτήσή σας καταχωρήθηκε με επιτυχία. ")
</script>
<?php

} //if

else{
if( $type=="ΕΡΓΑΣΙΑΣ ΩΡΩΝ" || $type=="ΠΡΟΣΩΠΙΚΗ" || $type=="ΠΕΠΟ" ){
$query="INSERT INTO vacation ( Req_by , type , Start_day , End_day , dayssum , hourssum , Reason ,
Stage , Start_hour , End_hour ) VALUES ( '$member_ID' , '$type' , '$pdate' , '$pdate' , '$datedays' , '$dif' ,
'$exc' , 0 , '$shour' , '$ehour' )";
mysql_query($query);
if(isset($days)){
$newmins=$mins+$days;
mysql_query("UPDATE adeies SET days='$newmins' WHERE UserID='$member_ID' AND type='$type' ") or
die('Error, UDPATE Into database FAILED');
} //if

else{
mysql_query("INSERT INTO adeies ( UserID , type , days ) VALUES ( '$member_ID' , '$type' , '$mins' )") or
die('Error, Insert Into database FAILED');
} //else
} //if
else{
$query="INSERT INTO vacation ( Req_by , type , Start_day , End_day , dayssum , Reason , Stage ,
Start_hour , End_hour , days , Back_day) VALUES ( '$member_ID' , '$type' , '$sdate' , '$edate' ,
'$datedays' , '$exc' , 0 , '$shour' , '$ehour' , '$sumdays' , '$bdate' )";
mysql_query($query);
if(isset($days)){
$newday=$datedays+$days;
mysql_query("UPDATE adeies SET days='$newday' WHERE UserID='$member_ID' AND type='$type' ") or
die('Error, UDPATE Into database FAILED');
} //if

else{
mysql_query("INSERT INTO adeies ( UserID , type , days ) VALUES ( '$member_ID' , '$type' ,
'$datedays' )") or die('Error, Insert Into database FAILED');
} //else
} //else
?>

<script language="JavaScript">
alert ("Η αιτήσή σας καταχωρήθηκε με επιτυχία. ")
</script>
<?php

```

```
    }//else

    }//if

    }//ifset submit
    ?>

<body>

<div id="wrapper">
<div id="header">

</div>

<div id="menu">
<?php
$result=mysql_query("Select Name , Surname FROM employee WHERE UserID = '$member_ID'");
while( $row = mysql_fetch_array($result , MYSQL_NUM)){
echo $row[1]; ?>
<?php echo $row[0];
}//while?>
</div>

<div id="sidebar">
<div id="feed">
<font color="#efefef">Μενού Επιλογών</font>
</div>
<ul>
<?php menu($member_ID); ?>
</ul>
<div id="sidebar-bottom">
</div>
</div>

<div id="content">
<div id="border_left"></div>
<font color="#333333" style="font-size:15px" ><font style="font-weight:bold"><?php
echo date('d/m/Y');?></font></font>
<br />

<div class="entry">
<div class="entry-title">Καταχώρηση Άδειας</div>
<div class="date"></div>
<p><br/></p>
<div class="comments"></div>
</div>

<form id="myform" action="adeia. php?action=submit" method="post" onsubmit="return confirm('Παρακαλώ
επιβεβαιώστε τα στοιχεία της άδειας η όπως αναγράφονται στην πίσω φόρμα');">

<table id="targetTable" width="100%" style="font-size:14px">
<tr>
```

```

<td align="right" width="114" style="vertical-align:middle">
Τύπος άδειας :
</td>
<td style="color:#F00" width="195">&nbsp;&nbsp;&nbsp;<select name="type" id="type" ><option> </option>
<?php
$result=mysql_query("Select Type FROM vacation_type ORDER BY ID");
$i=0;
while( $row = mysql_fetch_array($result , MYSQL_NUM)){
if($row[0]!="ΚΑΝΟΝΙΚΗ ΠΑΛΙΟΥ ΕΤΟΥΣ"){
?>
<option> <?php $i++;
echo $row[0];?> </option>
<?php } ?>
</select>
</td>
</tr>

<tr height="30%"></tr>

<tr id="start">
<td align="right" width="114">
Έναρξη άδειας :
</td>
<td style="color:#06F" align="center" width="195">Ημέρα :&nbsp;&nbsp;<select name="startday" id="startday">
<?php
$i=0;
while( $i<31 ){ $i++; ?> <option><?php echo $i;?></option><?php }?></select>

</td>
<td width="111" style="color:#06F" >Μήνας :&nbsp;&nbsp;<select name="startmonth" id="startmonth" >
<?php $i=0;
while ( $i<12){ $i++; ?> <option><?php echo $i;?></option><?php }?></select>
</td>
<td width="127" align="center" style="color:#06F">Χρόνος :&nbsp;&nbsp;<select name="startyear" id="startyear" >
<option>2009</option><option>2010</option></select>

</td>
</tr>
<tr height="30%"></tr>
<tr id="end">
<td align="right">
Τέλος άδειας :
</td>
<td style="color:#06F" align="center" width="195">Ημέρα :&nbsp;&nbsp;<select name="endday" id="endday" >
<?php $i=0;
while ( $i<31){ $i++; ?> <option><?php echo $i;?></option><?php }?></select>

</td>
<td width="111" style="color:#06F" >Μήνας :&nbsp;&nbsp;<select name="endmonth" id="endmonth" >
<?php $i=0;
while ( $i<12){ $i++; ?> <option><?php echo $i;?></option><?php }?></select>
</td>
<td width="127" align="center" style="color:#06F">Χρόνος :&nbsp;&nbsp;<select name="endyear" id="endyear" >
<option>2009</option><option>2010</option></select>

</td>
</tr>
<tr id="return" style="display:none">

```





```
end. style. display = 'none';
var row = document. getElementById("dilosi");
row. style. display = 'none';
}

else if (chosenoption. value=="ΕΡΓΑΣΙΑΣ ΩΡΩΝ"){
var row = document. getElementById("captionRow");
row. style. display = "";
var row = document. getElementById("captionRow1");
row. style. display = "";
var start = document. getElementById("start");
start. style. display = 'none';
var end = document. getElementById("end");
end. style. display = 'none';
var row = document. getElementById("back");
row. style. display = 'none';
var row = document. getElementById("return");
row. style. display = 'none';
var row = document. getElementById("message");
row. style. display = 'none';
var end = document. getElementById("private");
end. style. display = "";
var row = document. getElementById("dilosi");
row. style. display = 'none';
}

else if (chosenoption. value=="ΠΡΟΣΩΠΙΚΗ"){
var row = document. getElementById("captionRow");
row. style. display = 'none';
var row = document. getElementById("captionRow1");
row. style. display = "";
var row = document. getElementById("start");
row. style. display = 'none';
var row = document. getElementById("end");
row. style. display = 'none';
var row = document. getElementById("back");
row. style. display = 'none';
var row = document. getElementById("return");
row. style. display = 'none';
var row = document. getElementById("message");
row. style. display = 'none';
var end = document. getElementById("private");
end. style. display = "";
var row = document. getElementById("dilosi");
row. style. display = 'none';
}

else if(chosenoption. value=="KANONIKH" ){
var row = document. getElementById("captionRow");
row. style. display = 'none';
var row = document. getElementById("captionRow1");
row. style. display = 'none';
var row = document. getElementById("start");
row. style. display = "";
var row = document. getElementById("end");
row. style. display = "";
var row = document. getElementById("back");
row. style. display = "";
```



```
var row = document.getElementById("return");
row.style.display = "";
var row = document.getElementById("message");
row.style.display = 'none';
var end = document.getElementById("private");
end.style.display = 'none';
var row = document.getElementById("dilos");
row.style.display = 'none';
}

else if (chosenoption.value=="ΕΚΠΑΙΔΕΥΤΙΚΗ" || chosenoption.value=="ΑΙΜΟΔΟΤΙΚΗ"){
var row = document.getElementById("captionRow");
row.style.display = 'none';
var row = document.getElementById("captionRow1");
row.style.display = 'none';
var row = document.getElementById("start");
row.style.display = "";
var row = document.getElementById("end");
row.style.display = "";
var row = document.getElementById("back");
row.style.display = "";
var row = document.getElementById("return");
row.style.display = 'none';
var row = document.getElementById("message");
row.style.display = "";
var end = document.getElementById("private");
end.style.display = 'none';
var row = document.getElementById("dilos");
row.style.display = 'none';
}

else if (chosenoption.value=="ΥΠ. ΔΗΛΩΣΗ ΑΣΘΕΝΕΙΑΣ"){
var row = document.getElementById("captionRow");
row.style.display = 'none';
var row = document.getElementById("captionRow1");
row.style.display = 'none';
var row = document.getElementById("start");
row.style.display = "";
var row = document.getElementById("end");
row.style.display = "";
var row = document.getElementById("back");
row.style.display = "";
var row = document.getElementById("return");
row.style.display = 'none';
var end = document.getElementById("private");
end.style.display = 'none';
var row = document.getElementById("message");
row.style.display = 'none';
var row = document.getElementById("dilos");
row.style.display = "";
}

else {
var row = document.getElementById("captionRow");
row.style.display = 'none';
var row = document.getElementById("captionRow1");
row.style.display = 'none';
var row = document.getElementById("start");
```

```
row.style.display = "";
var row = document.getElementById("end");
row.style.display = "";
var row = document.getElementById("back");
row.style.display = "";
var row = document.getElementById("return");
row.style.display = 'none';
var end = document.getElementById("private");
end.style.display = 'none';
var row = document.getElementById("message");
row.style.display = 'none';
var row = document.getElementById("dilos");
row.style.display = 'none';
}
```

```
}//
//run some code when "onchange" event fires
```

```
</script>
```

```
<div id="footer">
```

```
<div id="footer-valid">
```

```
Copyright 2009<br />Διαχειριστής Έργου: Δρ. Μηνάς Δασυγένης<br />Designed by iCeMaN |  
iCeMaN999@gmail.com
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

## ΠΑΡΑΡΤΗΜΑ Β

Παρατίθεται ο κώδικας του αρχείου "profile.php". Ο κώδικας αυτός δίνει την δυνατότητα στον χρήστη να αλλάξει τον κωδικό πρόσβασης του και το email του. Τέλος, εμφανίζει και τα υπόλοιπα στοιχεία που ο χρήστης δεν έχει τη δυνατότητα να αλλάξει.

```
<?php
session_start();
if(!isset($_SESSION['member_ID'])):
header('Location: ../index.php');
endif;
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.
org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
if($_SESSION['database']=="ΕΛΚΕΔΕ Α. Ε."){?>
<link rel="stylesheet" href="Styles/style.css" type="text/css" media="screen" />
<?php
include ("Config/config.php");
}
else if ($_SESSION['database']=="ΕΞΥΠΠ ΕΛΚΕΔΕ ΕΠΕ"){?>
<link rel="stylesheet" href="Styles/style2.css" type="text/css" media="screen" />
<?php
include ("Config/exypp-config.php");
}
else if ($_SESSION['database']=="ΚΕΚ ΕΛΚΕΔΕ ΕΠΕ"){?>
<link rel="stylesheet" href="Styles/style3.css" type="text/css" media="screen" />
<?php
include ("Config/kek-config.php");
}

include ("functions.php");
?>
<link href="images/tab.gif" rel="shortcut icon" type="image/vnd.microsoft.icon" />
<title>Πρόγραμμα αδειών --- Προσωπικό προφίλ</title>
</head>
<?php
$member_ID=$_SESSION['member_ID'];

if(isset($_POST['submit'])) {

if( $_POST['pass']!="" && $_POST['mail']=="" && $_POST['pass2']!="") {
$password = mysql_real_escape_string($_POST['pass']);
$password2= mysql_real_escape_string($_POST['pass2']);
if($password==$password2){
$encrypt = sha1($_POST['pass']);
$query = "UPDATE members SET user_password= '$encrypt' WHERE ID = '$member_ID' ";
$results = mysql_query($query);

?><script language="JavaScript">
alert ("Ο κωδικός άλλαξε επιτυχώς")
</script>
<?php
}//if
else if ($password!=$password2){
```

```
?>
<script language="JavaScript">
alert ("Λανθασμένη επιβεβαίωση κωδικού");
</script>
<?php
}//else

}//if

else if ( $_POST['pass']=="" && $_POST['mail']!="" && $_POST['pass2']=="" ) {
$mails = mysql_real_escape_string($_POST['mail']);
$query = "UPDATE employee SET email= '$mail' WHERE UserID = '$member_ID' ";
$results = mysql_query($query);
?>
<script language="JavaScript">
alert ("To e-mail άλλαξε επιτυχώς!!")
</script>
<?php

}//elseif

else if ( $_POST['pass']!="" && $_POST['mail']!="" && $_POST['pass2']!="" ) {
$password = mysql_real_escape_string($_POST['pass']);
$password2= mysql_real_escape_string($_POST['pass2']);
if($password==$password2){
$encrypt = sha1($_POST['pass']);
$query = "UPDATE members SET user_password= '$encrypt' WHERE ID = '$member_ID' ";
$results = mysql_query($query);
$mails = mysql_real_escape_string($_POST['mail']);
$sql = "UPDATE employee SET email= '$mail' WHERE UserID = '$member_ID' ";
$results = mysql_query($sql);
?>
<script language="JavaScript">
alert ("To e-mail και ο κωδικός αλλάξαν επιτυχώς!!")
</script>
<?php
}//if
else{
?>
<script language="JavaScript">
alert ("Λανθασμένη επιβεβαίωση κωδικού");
</script>
<?php
}//else
}//elseif

else if ( $_POST['pass']=="" && $_POST['mail'] =="" && $_POST['pass2']!="" ) {
?>
<script language="JavaScript">
alert ("Κανένα στοιχείο δεν τροποποιήθηκε!!")
</script>
<?php }//else

else if ( $_POST['pass']=="" && $_POST['mail'] =="" && $_POST['pass2']!="" ) { ?>
<script language="JavaScript">
alert ("Πρέπει να εισάγετε 2 φορές τον κωδικό");
</script>
<?php
```



```

<td><input name="pass" type="password" size="20" /></td>
</tr>
<tr height="30%"></tr>
<tr>
<td align="right">
Επαλήθευση κωδικού :
</td>
<td><input name="pass2" type="password" size="20" /></td>
</tr>
<tr height="30%"></tr>
<tr>
<td align="right" width="144">
Όνομα :
</td>
<td style="color:#06F">&nbsp;&nbsp;&nbsp;<?php
$check = sprintf( "SELECT name FROM employee WHERE UserID='$member_ID' ");
$result= mysql_query($check);
$row = mysql_fetch_assoc($result);
echo "$row[name]";?>
</td>
</tr>

<tr height="30%"></tr>
<tr>
<td align="right">
Επώνυμο :
</td>
<td style="color:#06F">&nbsp;&nbsp;&nbsp;<?php
$check = sprintf( "SELECT surname FROM employee WHERE UserID='$member_ID' ");
$result= mysql_query($check);
$row = mysql_fetch_assoc($result);
echo "$row[surname]"; ?>
</td>
</tr>
<tr height="30%"></tr>
<tr>
<td align="right">
Email :
</td>
<td style="color:#06F"><input name="mail" type="text" size="20" />&nbsp;&nbsp;&nbsp;<?php
$check = sprintf( "SELECT email FROM employee WHERE UserID='$member_ID' ");
$result= mysql_query($check);
$row = mysql_fetch_assoc($result);
echo "$row[email]"; ?></font>
</td>
</tr>
<tr height="30%"></tr>
<tr>
<td align="right">
Διεύθυνση :
</td>
<td style="color:#06F">&nbsp;&nbsp;&nbsp;<?php
$check = sprintf( "SELECT department FROM employee WHERE UserID='$member_ID' ");
$result= mysql_query($check);
$row = mysql_fetch_assoc($result);
if(isset($row['department'])){
echo "$row[department]"; ?>
</td>

```

```
</tr>
<tr height="30%"></tr>
<tr>
<td colspan="2">&nbsp;</td>
</tr>
<tr>
<td>&nbsp;</td>
<td>
<input align="right" name="submit" type="submit" value="Αποθήκευση" class="submit" />
</td>
</tr>
<tr>
<td colspan="2">

</td>
</tr>
</table>

</form>
</div>
<div id="footer">
<div id="footer-valid">
Copyright 2009<br />Διαχειριστής Έργου: Δρ. Μηνάς Δασυγένης<br />Designed by iCeMaN |
iCeMaN999@gmail.com
</div>
</div>

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

## ΠΑΡΑΡΤΗΜΑ Γ

Παρατίθεται ο κώδικας του αρχείου `employee_history.php`. Ο κώδικας αυτός δίνει την δυνατότητα στους χρήστες με τα κατάλληλα δικαιώματα (Διευθυντής Προσωπικού και Διευθύνων Σύμβουλος) να ελέγχουν το ιστορικό όλων των υπαλλήλων της εταιρίας.

```
<?php
session_start();
if(!isset($_SESSION['member_ID'])):
header('Location: ../index.php');
endif;

?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
if($_SESSION['database']=="ΕΛΚΕΔΕ Α. Ε."){?>
<link rel="stylesheet" href="Styles/style.css" type="text/css" media="screen" />
<?php
include ("Config/config.php");
}
else if ($_SESSION['database']=="ΕΞΥΠΠ ΕΛΚΕΔΕ ΕΠΕ"){?>
<link rel="stylesheet" href="Styles/style2.css" type="text/css" media="screen" />
<?php
include ("Config/exypp-config.php");
}
else if ($_SESSION['database']=="ΚΕΚ ΕΛΚΕΔΕ ΕΠΕ"){?>
<link rel="stylesheet" href="Styles/style3.css" type="text/css" media="screen" />
<?php
include ("Config/kek-config.php");
}

include ("functions.php");
$id=$_SESSION['member_ID'];
$query=mysql_query("SELECT groups FROM employee WHERE UserID='$id'");
$row=mysql_fetch_assoc($query);
if( $row['groups'] == 4 || $row['groups'] == 0 || $row['groups'] == 3) {
// Έλεγχος για το αν ο χρήστης που εισέρχεται είναι ο Administrator κ.λ.π
header('Location: ../index.php');
}//if
?>
<link href="images/tab.gif" rel="shortcut icon" type="image/vnd.microsoft.icon" />
<title>Πρόγραμμα αδειών --- Ιστορικό Υπαλλήλων</title>
</head>
<body>
<div id="wrapper">

<div id="header">

</div>

<div id="menu">
<?php
$member_ID=$_SESSION['member_ID'];
```







```

$人=$row['UserID'];
$人day=mysql_real_escape_string($_POST['startday']);
$人month=mysql_real_escape_string($_POST['startmonth']);
$人year=mysql_real_escape_string($_POST['startyear']);
$人day=mysql_real_escape_string($_POST['endday']);
$人month=mysql_real_escape_string($_POST['endmonth']);
$人year=mysql_real_escape_string($_POST['endyear']);
$type=mysql_real_escape_string($_POST['type']);

$人date='20' . $人year . '/' . $人month . '/' . $人day;
$人date='20' . $人year . '/' . $人month . '/' . $人day;

list($人year, $人month, $人day) = explode('/', $人date);
list($人year, $人month, $人day) = explode('/', $人date);

$人day=$人day . '/' . $人month . '/' . $人year;
$人day=$人day . '/' . $人month . '/' . $人year;

if($人fullname=="ΟΛΟΙ" && $人type!="ΟΛΕΣ" && $人type!="ΟΛΕΣ ΕΚΤΟΣ ΚΑΝΟΝΙΚΗΣ"){
if($人date=="20// " || $人date=="20// " ){
$人query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE type='$人type' AND stage=3");
$人row=mysql_fetch_assoc($人query);
?>Έχουν δωθει <font color="#C10000"><?php echo $人row['counta'];?></font><?php echo ' άδειες τύπου
';?><font color="#C10000"><?php echo $人type ; ?></font><br />
Αναλυτικά :<br />

<table width="100%">
<?php $人query=mysql_query("SELECT surname , name FROM employee WHERE surname!='Administrator'
ORDER BY surname");
while($人row=mysql_fetch_array($人query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo $人row[0] , ' ' ,
$人row[1];?></font></td></tr><?php $人sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$人row[0]' AND name='$人row[1]'");
$人sql2=mysql_fetch_assoc($人sql);
$人result=mysql_query("SELECT type , Start_day , End_day , dayssum , Req_at , ID FROM vacation
WHERE Req_by='$人sql2[UserID]' AND type='$人type' AND stage=3");
while($人row2=mysql_fetch_array($人result)){
list($人year, $人month, $人day) = explode('-', $人row2[1]);
list($人year, $人month, $人day) = explode('-', $人row2[2]);
$人date=$人day . '/' . $人month . '/' . $人year;
$人date=$人day . '/' . $人month . '/' . $人year;
?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $人date?></font> εώς τις <font
color="#C10000"><?php echo $人date?></font>. Σύνολο <font color="#C10000"><?php echo
$人row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $人row2[5];?>," style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php }//while

}//while
?>
</table><?php }//if set date

else{
$人query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE '$人date'<Start_day AND
Start_day<'$人date' AND type='$人type' AND stage=3");
$人row=mysql_fetch_assoc($人query);

```

```
?>Εχουν δωθεί <font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες τύπου
';?><font color="#C10000"><?php echo $type; ?></font><?php echo ' απο ' ;?><font color="#C10000"><?php
echo $sday ; ?></font><?php echo ' έως ' ;?><font color="#C10000"><?php echo $eday ;?> </font><br />
Αναλυτικά :<br />
```

```
<table width="100%">
<?php $query=mysql_query("SELECT surname , name FROM employee WHERE surname!='Administrator'
ORDER BY surname");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo $row[0] , ' ' ,
$row[1];?></font></td></tr><?php
$sql=mysql_query("SELECT UserID FROM employee WHERE surname='$row[0]' AND name='$row[1]'");
$sql2=mysql_fetch_assoc($sql);
$result=mysql_query("SELECT type , Start_day , End_day , dayssum , Req_at , ID FROM vacation
WHERE '$sdate'<Start_day AND '$edate'>Start_day AND Req_by='$sql2[UserID]' AND type='$type' AND
stage=3");
while($row2=mysql_fetch_array($result)){
list($syear, $smmonth, $sday) = explode('-', $row2[1]);
list($eyear, $emmonth, $eday) = explode('-', $row2[2]);
$sdates=$sday. '/'. $smmonth. '/'. $syear;
$edates=$eday. '/'. $emmonth. '/'. $eyear;
?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdates?></font> έως τις
<font color="#C10000"><?php echo $edates?></font>. Σύνολο <font color="#C10000"><?php echo
$row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $row2[5];?>);" style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php }//while
```

```
}//while
?>
</table><?php }//else date
}//if
```

```
else if($type=="ΟΛΕΣ" && $fullname!="ΟΛΟΙ"){
if($sdate=="20/" || $edate=="20/" ){
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Req_by='$id' AND stage=3");
$row=mysql_fetch_assoc($query);
echo ' Ο Χρήστης ' ;?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' πήρε ' ;
?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες ' ;?> </font>
```

```
<table width="100%">
<?php $query=mysql_query("SELECT type FROM vacation_type GROUP BY type");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo
$row[0];?></font></td></tr><?php $result=mysql_query("SELECT type , Start_day , End_day , dayssum ,
Req_at , ID FROM vacation WHERE Req_by='$id' AND type='$row[0]' AND stage=3");
while($row2=mysql_fetch_array($result)){
list($syear, $smmonth, $sday) = explode('-', $row2[1]);
list($eyear, $emmonth, $eday) = explode('-', $row2[2]);
$sdate=$sday. '/'. $smmonth. '/'. $syear;
$edate=$eday. '/'. $emmonth. '/'. $eyear;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdate?></font> έως τις <font
color="#C10000"><?php echo $edate?></font>. Σύνολο <font color="#C10000"><?php echo
$row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $row2[5];?>);" style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php }//while
```

```

} //while
?>
</table><?php //if not set
else {
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Req_by='$id' AND '$sdate'<Start_day AND Start_day<'$edate' AND stage=3");
$row=mysql_fetch_assoc($query);
echo ' Ο Χρήστης '; ?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' πήρε ';
?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες απο ' ;?><font
color="#C10000"><?php echo $sdate ; ?></font><?php echo ' έως ' ; ?><font color="#C10000"><?php echo
$edate ;?> </font>

<table width="100%">
<?php $query=mysql_query("SELECT type FROM vacation_type GROUP BY type");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo
$row[0];?></font></td></tr><?php $result=mysql_query("SELECT type , Start_day , End_day , dayssum ,
Req_at , ID FROM vacation WHERE '$sdate'<Start_day AND Start_day<'$edate' AND Req_by='$id' AND
type='$row[0]' AND stage=3");
while($row2=mysql_fetch_array($result)){
list($year, $month, $sday) = explode('-', $row2[1]);
list($year, $month, $eday) = explode('-', $row2[2]);
$sdates=$sday. '/'. $month. '/'. $year;
$edates=$eday. '/'. $month. '/'. $year;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdates?></font> έως τις
<font color="#C10000"><?php echo $edates?></font>. Σύνολο <font color="#C10000"><?php echo
$row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $row2[5];?>)" style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php //while

} //while
?>
</table><?php //else not set

} //else

else if($type=="ΟΛΕΣ" && $fullname=="ΟΛΟΙ"){
if($sdate=="20/" || $edate=="20/" ){
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE stage=3");
$row=mysql_fetch_assoc($query);
?>Έχουν δωθεί <font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες ';?>
<table width="100%">
<?php $query=mysql_query("SELECT surname , name FROM employee WHERE surname!='Administrator'
ORDER BY surname");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo $row[0] , ' ' ,
$row[1];?></font></td></tr><?php $sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$row[0]' AND name='$row[1]'");
$sql2=mysql_fetch_assoc($sql);
$result=mysql_query("SELECT type , Start_day , End_day , dayssum , Req_at , ID FROM vacation
WHERE Req_by='$sql2[UserID]' AND stage=3");
while($row2=mysql_fetch_array($result)){
list($year, $month, $sday) = explode('-', $row2[1]);
list($year, $month, $eday) = explode('-', $row2[2]);
$sdate=$sday. '/'. $month. '/'. $year;
$edate=$eday. '/'. $month. '/'. $year;?>
<tr><td align="center" >Πήρε άδεια <font color="#C10000"><?php echo $row2[0];?></font> από τις <font
color="#C10000"><?php echo $sdate?></font> έως τις <font color="#C10000"><?php echo $edate?></font>.

```

```

Σύνολο <font color="#C10000"><?php echo $row2[3];?></font> ημερών. </td><td><button
onclick="site(<?php echo $row2[5];?>);" style="font-size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php }//while

```

```

}//while
?>
</table><?php }//if not set

```

```

else{
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Start_day<'$sdate' AND
Start_day>'$sdate' AND stage=3");
$row=mysql_fetch_assoc($query);
?>Έχουν δωθεί <font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες απο '
;?><font color="#C10000"><?php echo $sdate ; ?></font><?php echo ' έως ' ; ?><font
color="#C10000"><?php echo $sdate ;?> </font>
<table width="100%">
<?php

```

```

$query=mysql_query("SELECT surname , name FROM employee WHERE surname!='Administrator' ORDER
BY surname");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo $row[0] , ' ' ,
$row[1];?></font></td></tr><?php $sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$row[0]' AND name='$row[1]'");
$sql2=mysql_fetch_assoc($sql);
$result=mysql_query("SELECT type , Start_day , End_day , dayssum , Req_at , ID FROM vacation
WHERE Start_day>'$sdate' AND Start_day<'$sdate' AND Req_by='$sql2[UserID]' AND stage=3");
while($row2=mysql_fetch_array($result)){
list($year, $month, $day) = explode('-', $row2[1]);
list($year, $month, $day) = explode('-', $row2[2]);
$sdates=$day. '.'. $month. '.'. $year;
$edates=$day. '.'. $month. '.'. $year;?>
<tr><td align="center" >Πήρε άδεια <font color="#C10000"><?php echo $row2[0]?></font> από τις <font
color="#C10000"><?php echo $sdates?></font> έως τις <font color="#C10000"><?php echo
$edates?></font>. Σύνολο <font color="#C10000"><?php echo $row2[3];?></font> ημερών. </td><td><button
onclick="site(<?php echo $row2[5];?>);" style="font-size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php }//while

```

```

}//while
?>
</table><?php }//else

```

```

}//else

```

```

else if($type=="ΟΛΕΣ ΕΚΤΟΣ ΚΑΝΟΝΙΚΗΣ" && $fullname=="ΟΛΟΙ"){
if($sdate=="20/" || $sdate=="20/" ){
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE stage=3 AND
type!='ΚΑΝΟΝΙΚΗ'");
$row=mysql_fetch_assoc($query);
?>Έχουν δωθεί <font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες ';?>
<table width="100%">
<?php $query=mysql_query("SELECT surname , name FROM employee WHERE surname!='Administrator'
ORDER BY surname");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo $row[0] , ' ' ,
$row[1];?></font></td></tr><?php $sql=mysql_query("SELECT UserID FROM employee WHERE
surname='$row[0]' AND name='$row[1]'");
$sql2=mysql_fetch_assoc($sql);

```





```
echo ' Ο Χρήστης '; ?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' πήρε ';
?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες ';?> </font>
```

```
<table width="100%">
<?php $query=mysql_query("SELECT type FROM vacation_type GROUP BY type");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo
$row[0];?></font></td></tr><?php $result=mysql_query("SELECT type , Start_day , End_day , dayssum ,
Req_at , ID FROM vacation WHERE Req_by='$id' AND type='$row[0]' AND stage=3 AND
type!='KANONIKH'");
while($row2=mysql_fetch_array($result)){
list($year, $smmonth, $sday) = explode('-', $row2[1]);
list($eyear, $emmonth, $eday) = explode('-', $row2[2]);
$sdate=$sday. '/'. $smmonth. '/'. $year;
$edate=$eday. '/'. $emmonth. '/'. $year;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdate?></font> έως τις <font
color="#C10000"><?php echo $edate?></font>. Σύνολο <font color="#C10000"><?php echo
$row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $row2[5];?>);" style="font-
size:12px">EKTYΠΩΣΗ</button></td></tr>
<?php }/while
```

```
}//while
?>
</table><?php //if not set
else {
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Req_by='$id' AND
'$sdate'<Start_day AND Start_day<'$edate' AND stage=3 AND type!='KANONIKH'");
$row=mysql_fetch_assoc($query);
echo ' Ο Χρήστης '; ?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' πήρε ';
?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες απο ' ;?><font
color="#C10000"><?php echo $sdate ; ?></font><?php echo ' έως ' ; ?><font color="#C10000"><?php echo
$edate ;?> </font>
```

```
<table width="100%">
<?php $query=mysql_query("SELECT type FROM vacation_type GROUP BY type");
while($row=mysql_fetch_array($query)){?>
<tr width="100%"><td align="center"><font style="font-weight:bold"><?php echo
$row[0];?></font></td></tr><?php $result=mysql_query("SELECT type , Start_day , End_day , dayssum ,
Req_at , ID FROM vacation WHERE '$sdate'<Start_day AND Start_day<'$edate' AND Req_by='$id' AND
type='$row[0]' AND stage=3 AND type!='KANONIKH'");
while($row2=mysql_fetch_array($result)){
list($year, $smmonth, $sday) = explode('-', $row2[1]);
list($eyear, $emmonth, $eday) = explode('-', $row2[2]);
$sdates=$sday. '/'. $smmonth. '/'. $year;
$edates=$eday. '/'. $emmonth. '/'. $year;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdates?></font> έως τις
<font color="#C10000"><?php echo $edates?></font>. Σύνολο <font color="#C10000"><?php echo
$row2[3];?></font> ημερών. </td><td><button onclick="site(<?php echo $row2[5];?>);" style="font-
size:12px">EKTYΠΩΣΗ</button></td></tr>
<?php }/while
```

```
}//while
?>
</table><?php //else not set
```

```
}//else
```

```
else{
```



```

if($sdate=="20/" || $edate=="20/" ){
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Req_by='$id' AND
type='$type' AND stage=3");
$row=mysql_fetch_assoc($query);
echo ' Ο Χρήστης '; ?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' έχει
πάρει '; ?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες τύπου ';?><font
color="#C10000"><?php echo $type ; ?></font><br />
Αναλυτικά :<br />
<table width=100%><?php
$result=mysql_query("SELECT Start_day , End_day , dayssum , ID FROM vacation WHERE Req_by='$id'
AND type='$type' AND stage=3");
while($row=mysql_fetch_array($result)){
list($syear, $smmonth, $sday) = explode('-', $row[0]);
list($eyear, $emmonth, $eday) = explode('-', $row[1]);
$sdate=$sday. '/'. $smmonth. '/'. $syear;
$edate=$eday. '/'. $emmonth. '/'. $eyear;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdate?></font> έως τις <font
color="#C10000"><?php echo $edate?></font>. Σύνολο <font color="#C10000"><?php echo
$row[2];?></font> ημερών. </td><td><button onclick="site(<?php echo $row[3];?>);" style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php

} //while
?></table><?php } //if not set
else{
$query=mysql_query("SELECT count(type) AS counta FROM vacation WHERE Req_by='$id' AND
type='$type' AND stage=3");
$row=mysql_fetch_assoc($query);
echo ' Ο Χρήστης '; ?><font color="#C10000"><?php echo $last , ' ' , $first;?> </font> <?php echo ' ζήτησε ';
?><font color="#C10000"><?php echo $row['counta'] ;?></font><?php echo ' άδειες τύπου ';?><font
color="#C10000"><?php echo $type ; ?></font><?php echo ' απο ';?><font color="#C10000"><?php echo
$sdate ; ?></font><?php echo ' έως '; ?><font color="#C10000"><?php echo $edate ;?> </font></font><br />
Αναλυτικά :<br />
<table width=100%><?php
$result=mysql_query("SELECT Start_day , End_day , dayssum , ID FROM vacation WHERE
'$sdate'<Start_day AND Start_day<'$edate' AND Req_by='$id' AND type='$type' AND stage=3");
while($row=mysql_fetch_array($result)){
list($syear, $smmonth, $sday) = explode('-', $row[0]);
list($eyear, $emmonth, $eday) = explode('-', $row[1]);
$sdates=$sday. '/'. $smmonth. '/'. $syear;
$edates=$eday. '/'. $emmonth. '/'. $eyear;?>
<tr><td align="center" >Πήρε άδεια από τις <font color="#C10000"><?php echo $sdates?></font> έως τις
<font color="#C10000"><?php echo $edates?></font>. Σύνολο <font color="#C10000"><?php echo
$row[2];?></font> ημερών. </td><td><button onclick="site(<?php echo $row[3];?>);" style="font-
size:12px">ΕΚΤΥΠΩΣΗ</button></td></tr>
<?php

} //while
?></table><?php } //else is set
} //else
} //if submit
?>
<script>
function site(x){
var id=x;
window. open('adeiapdf. php?id='+id);
}
</script>

```

```
</form>
<div class="comments"> </div>
</div>
<div id="footer">
<div id="footer-valid">
Copyright 2009<br />Διαχειριστής Έργου: Δρ. Μηνάς Δασυγένης<br />Designed by iCeMaN |
iCeMaN999@gmail.com
</div>
</div>
</div>
</body>
</html>
```

## ΠΑΡΑΡΤΗΜΑ Δ

Παρατίθεται ο κώδικας του αρχείου approve.php. Ο κώδικας αυτός δίνει την δυνατότητα στους χρήστες με τα κατάλληλα δικαιώματα (Προϊστάμενος τμήματος, Διευθυντής Προσωπικού και Διευθύνων Σύμβουλος) να εγκρίνουν ή να απορρίπτουν τις άδειες που έχουν ζητήσει οι υφιστάμενοί τους.

```
<?php
session_start();
if(!isset($_SESSION['member_ID'])):
header('Location: ../index.php');
endif;

$member_ID=$_SESSION['member_ID'];
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<?php
if($_SESSION['database']=="ΕΛΚΕΔΕ Α. Ε. ")?>
<link rel="stylesheet" href="Styles/style.css" type="text/css" media="screen" />
<?php
include ("Config/config.php");
}
else if ($_SESSION['database']=="ΕΞΥΠΠ ΕΛΚΕΔΕ ΕΠΕ")?>
<link rel="stylesheet" href="Styles/style2.css" type="text/css" media="screen" />
<?php
include ("Config/exypp-config.php");
}
else if ($_SESSION['database']=="ΚΕΚ ΕΛΚΕΔΕ ΕΠΕ")?>
<link rel="stylesheet" href="Styles/style3.css" type="text/css" media="screen" />
<?php
include ("Config/kek-config.php");
}

include ("functions.php");
$id=$_SESSION['member_ID'];
$query=mysql_query("SELECT groups FROM employee WHERE UserID='$id'");
$row=mysql_fetch_assoc($query);
if( $row['groups'] == 4 || $row['groups'] == 0) { //
Έλεγχος για το αν ο χρήστης που εισέρχεται είναι ο Administrator
header('Location: ../index.php');
} //if
?>
<link href="images/tab.gif" rel="shortcut icon" type="image/vnd.microsoft.icon" />
<title>Πρόγραμμα αδειών --- Έγκριση αδειών</title>
</head>
<body>
<?php

//Έλεγχος για το αν έχει πάρει τιμή το πεδίο submit
if(isset($_POST['submit'])) {
if ($vid!="") {
if($group=='3'){
```

```

$query = mysql_query ( "UPDATE vacation SET Stage = '1' WHERE vacation. ID='$vid' " );
//Αλλαγή του πεδίου stage στην βάση για να προωθηθεί στο επόμενο στάδιο.
} //if

else if ($group=='2'){
if($tipos=="KANONIKH"){
$query = mysql_query ( "UPDATE vacation SET Stage = '2' WHERE vacation. ID='$vid' " ); //Αλλαγή του
πεδίου stage στην βάση για να προωθηθεί στο επόμενο στάδιο.
} //if
else{
$query = mysql_query ( "UPDATE vacation SET Stage = '3' WHERE vacation. ID='$vid' " ); //Αλλαγή του
πεδίου stage στην βάση για να εγριθεί η άδεια.
//Αποστολή email στον χρήστη που ζήτησε την άδεια.
$result = mysql_query ( "SELECT email FROM employee WHERE UserID = '$who' " );
while( $row = mysql_fetch_array($result) )
approved($row[0]); //Καλούμε την function approved με
παράμετρο το email του υπαλλήλου που ζήτησε άδεια.
} //else
} //else if

else if ($group=='1'){
$query = mysql_query ( "UPDATE vacation SET Stage = '3' WHERE vacation. ID='$vid' " );
//Αλλαγή του πεδίου stage στην βάση για να εγριθεί η άδεια.
//Αποστολή email στον χρήστη που ζήτησε την άδεια.
$result = mysql_query ( "SELECT email FROM employee WHERE UserID = '$who' " );
while( $row = mysql_fetch_array($result) )
approved($row[0]); //Καλούμε την function notapproved με
παράμετρο το email του υπαλλήλου που ζήτησε άδεια.
} //else if

} //if
} //if isset
//Έλεγχος για το αν έχει πάρει τιμή το πεδίο submit1
if(isset($_POST['submit1'])) {
$exc=mysql_real_escape_string($_POST['excuse']);
if ($vid<>"") {
//Αλλαγή του πεδίου stage στην βάση για να καταχωρηθεί ως απορριφθείσα άδεια.
$query = mysql_query ( "UPDATE vacation SET Stage = '4' WHERE vacation. ID='$vid' " );

$query = mysql_query ( "UPDATE vacation SET Rej_Reason = '$exc' WHERE vacation. ID='$vid' " );

$result = mysql_query ( " SELECT days FROM adeies WHERE UserID = '$who' AND type='$tipos'");
while( $row = mysql_fetch_array($result) ){
$dif=$row[0]-$sum; //Υπολογισμός των ημερών που πρέπει να αφαιρεθούν
απο την βάση.
} //while

// Αφαίρεση των ημερών που ζητήθηκαν απο την βάση.
$result= mysql_query ( "UPDATE adeies SET days = '$dif' WHERE UserID='$who' AND type='$tipos' " );

if($group==3){ //Αν απορίφθηκε απο τον προϊστάμενο

$result = mysql_query ( "SELECT email FROM employee WHERE UserID = '$who' " );
$excuse=mysql_real_escape_string($_POST['excuse']);
while( $row = mysql_fetch_array($result) )
//Αποστολή email στον χρήστη που ζήτησε την άδεια.
$subject = "Αίτηση άδειας";

```

```
$body = "Η αίτηση σας για άδεια δεν εγκρίθηκε! ". $y;
mail( $row[0] , $subject, $body);
} //if group

else if($group==2){
    //Αν απορίφθηκε απο τον Διευθ. προσωπικού

    $result = mysql_query ( "SELECT email , department FROM employee WHERE UserID = '$who' " );
    $excuse=mysql_real_escape_string($_POST['excuse']);
    while( $row = mysql_fetch_array($result) )
    $sql=mysql_query("SELECT email FROM employee WHERE department='$row[1]' AND group = 3");
    $row=mysql_fetch_assoc($sql);
    $super_email=$row['email'];

    //Αποστολή email στον χρήστη που ζήτησε την άδεια.
    $subject = "Αίτηση άδειας";
    $body = "Η αίτηση σας για άδεια δεν εγκρίθηκε! ". $y;
    mail( $row[0] , $subject, $body);

    //Αποστολή email στον προϊστάμενο του χρήστη που ζήτησε την άδεια.
    $subject = "Αίτηση άδειας του χρήστη με email". $row[0];
    $body = "Η αίτηση για άδεια δεν εγκρίθηκε! ". $y;
    mail( $super_email, $subject, $body);

} //if group =2

else if($group==1){
    //Αν απορίφθηκε απο τον Διευθ. σύμβουλο

    $result = mysql_query ( "SELECT email , department FROM employee WHERE UserID = '$who' " );
    $excuse=mysql_real_escape_string($_POST['excuse']);
    while( $row = mysql_fetch_array($result) )
    $sql=mysql_query("SELECT email FROM employee WHERE department='$row[1]' AND group = 3");
    $row=mysql_fetch_assoc($sql);
    $super_email=$row['email'];
    $sql=mysql_query("SELECT email FROM employee WHERE group = 2");
    $row=mysql_fetch_assoc($sql);
    $boss_email=$row['email'];

    //Αποστολή email στον χρήστη που ζήτησε την άδεια.
    $subject = "Αίτηση άδειας";
    $body = "Η αίτηση σας για άδεια δεν εγκρίθηκε! ". $y;
    mail( $row[0] , $subject, $body);

    //Αποστολή email στον προϊστάμενο του χρήστη που ζήτησε την άδεια.
    $subject = "Αίτηση άδειας του χρήστη με email". $row[0];
    $body = "Η αίτηση για άδεια δεν εγκρίθηκε! ". $y;
    mail( $super_email, $subject, $body);

    //Αποστολή email στον Διευθ. Προσωπικού
    $subject = "Αίτηση άδειας του χρήστη με email". $row[0];
    $body = "Η αίτηση για άδεια δεν εγκρίθηκε! ". $y;
    mail( $boss_mail, $subject, $body);

} //if group =1

} //if vid
```

```

} //if isset
?>
<div id="wrapper">
<div id="header">

</div>

<div id="menu">
<?php
$result=mysql_query("Select Name , Surname FROM employee WHERE UserID = '$member_ID'");
while( $row = mysql_fetch_array($result , MYSQL_NUM)){
echo $row[1]; ?>
<?php echo $row[0];
} //while?>

</div>

<div id="sidebar">
<div id="feed">
<font color="#efefef">Μενού Επιλογών</font>
</div>
<ul>
<?php menu($member_ID); ?>
</ul>
<div id="sidebar-bottom">
</div>
</div>

<div id="content">
<div id="border_left">
<font color="#333333" style="font-size:15px" ><font style="font-weight:bold"><?php
echo date('d/m/Y');?></font></font>
<br />

<div class="entry">
<div class="entry-title">Έγκριση άδειας</div>
<div class="date"></div>
<p><br/></p>

<?php
//Καταχώρηση σε μεταβλητές των στοιχείων που θα μας χρειαστούν απο την βάση.
$depar= mysql_query (" SELECT department , groups FROM employee WHERE UserID='$member_ID' ");
$row = mysql_fetch_assoc($depar);
$test=$row['department'];
$group=$row['groups'];

//Έλεγχος σε πιο group ανήκει ο χρήσης , ώστε να εμφανιστούν ΜΟΝΟ οι άδειες που πρέπει ( Του
κατώτερου ιεραρχικά υπαλλήλου του τμηματός του).
if( $group=='3'){
$results= mysql_query( "SELECT vacation. ID , vacation. type , vacation. Start_day , vacation. End_day ,
vacation. Req_at , vacation. dayssum , vacation. hourssum , employee. name , employee. surname ,
vacation. Req_by , employee. department , vacation. Start_hour , vacation. End_hour , vacation. Reason
FROM vacation , employee
WHERE Stage=0 AND vacation. Req_by=employee. UserID AND employee. department='$test' AND
employee. groups>3" );
} //if

else if ( $group=='2'){
```

```
$results= mysql_query( "SELECT vacation. ID , vacation. type , vacation. Start_day , vacation. End_day ,
vacation. Req_at , vacation. dayssum , vacation. hourssum , employee. name , employee. surname ,
vacation. Req_by , employee. department , vacation. Start_hour , vacation. End_hour , vacation. Reason
FROM vacation , employee
WHERE Stage=1 AND vacation. Req_by=employee. UserID AND employee. groups>2" );
```

```
//else if
else if ( $group=='1'){
$results= mysql_query( "SELECT vacation. ID , vacation. type , vacation. Start_day , vacation. End_day ,
vacation. Req_at , vacation. dayssum , vacation. hourssum , employee. name , employee. surname ,
vacation. Req_by , employee. department , vacation. Start_hour , vacation. End_hour , vacation. Reason
FROM vacation , employee
WHERE Stage=2 AND vacation. Req_by=employee. UserID AND employee. groups>1 AND vacation.
type='KANONIKH" );
```

```
//else if
```

```
$i=0;
while($row = mysql_fetch_array($results)){
$i++;
```

```
list($year, $smmonth, $sday) = explode('-', $row[2]);
list($year, $emonth, $eday) = explode('-', $row[3]);
// list($byear, $bmonth, $bday) = explode('-', $backday);
$sdate=$sday. '/'. $smmonth. '/'. $year;
$edate=$eday. '/'. $emonth. '/'. $year;
```

?><!--//Αποστολή των στοιχείων που χρειαζόμαστε μόλις πατήσουμε ένα απο τα κουμπια ώστε να γίνει η κατάλληλη διεργασία-->

```
<form id="myform" action="approve. php?action=submit&vid=<?php echo $row[0]?>&group=<?php echo
$group?>&tipos=<?php echo $row[1]?>&who=<?php echo $row[9]?>&sum=<?php echo $row[5]?>"
method="post" onSubmit="return confirm('Είστε σίγουρος;')"> <?php
if($row[1]=="ΠΡΟΣΩΠΙΚΗ"){?>
```

```
<table style="font-size:14px">
```

```
<tr>
```

```
<?php if($group==2){?>
```

```
<td width="500" align="center"><font size="-1">Ο/Η υπάλληλος <font color="#FF0000"><?php echo
$row[7];?>&nbsp;<?php echo $row[8];?></font> απο το τμήμα <font color="#FF0000"><?php echo
$row[10];?></font> ζήτησε άδεια <font color="#FF0000"><?php echo $row[1]?></font> σύνολο <font
color="#FF0000"><?php echo $row[6]/60?></font> ωρών. <br /> Από τις <font color="#FF0000"><?php echo
$row[11]?></font> μέχρι τις <font color="#FF0000"><?php echo $row[12]?>. </font></font>Στις <font
color="#FF0000"><?php echo $sdate?></font>. </td>
</tr><?php }/if?>
```

```
<?php if($group!=2){?>
```

```
<td width="500" align="center">Ο/Η υπάλληλος <font color="#FF0000"><?php echo $row[7];?>&nbsp;<?php
echo $row[8];?></font> ζήτησε άδεια <font color="#FF0000"><?php echo $row[1]?></font> σύνολο <font
color="#FF0000"><?php echo $row[6]/60?></font> ωρών. . <br /> Από τις <font color="#FF0000"><?php
echo $row[11]?></font> μέχρι τις <font color="#FF0000"><?php echo $row[12]?>. </font></font>Στις <font
color="#FF0000"><?php echo $sdate?></font></td>
</tr><?php }/if?>
```

```
<tr>
```

```
<td height="10px"></td>
```

```
</tr>
```

```
<tr>
```









## ΒΙΒΛΙΟΓΡΑΦΙΑ

Ιστοσελίδες

<http://www.php.net/>

<http://el.wikipedia.org/wiki/HTML>

<http://el.wikipedia.org/wiki/CSS>

<http://en.wikipedia.org/wiki/PHP>

<http://www.w3schools.com>