

Τ.Ε.Ι. ΗΠΕΙΡΟΥ

Τ.Ε.Ι. OF EPIRUS



ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ & ΟΙΚΟΝΟΜΙΑΣ (Σ.Δ.Ο)
ΤΜΗΜΑ ΤΗΛΕΠΛΗΡΟΦΟΡΙΚΗΣ & ΔΙΟΙΚΗΣΗΣ

SCHOOL OF MANAGEMENT AND ECONOMICS
DEPARTMENT OF COMMUNICATIONS,
INFORMATICS AND MANAGEMENT



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ
ΘΕΜΑ: ΚΑΤΑΣΚΕΥΗ ΣΕΛΙΔΑΣ
ΗΛΕΚΤΡΟΝΙΚΗΣ ΒΑΘΜΟΛΟΓΙΑΣ

ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ

Σύντομη γενική περιγραφή του θέματος

Το θέμα που θα αναπτυχθεί είναι η κατασκευή ενός web site που θα παρουσιάζει μια αναλυτική κατάσταση βαθμολογίας για τον κάθε φοιτητή. Η αρχική σελίδα θα ζητά από το χρήστη να εισάγει το πάσο του και μετά θα πατάει το κουμπί submit και θα στέλνεται το πάσο στον Server του τμήματος και θα ερευνάτε αν πληρεί κάποια κριτήρια όπως αν έχει όντως τα απαραίτητα ψηφία .

Εφόσον δεν βρεθεί κάποια εγγραφή με το συγκεκριμένο πάσο τότε εμφανίζεται το σχετικό μήνυμα που ενημερώνει τον χρήστη. Υπάρχει μια βάση δεδομένων που είναι αποθηκευμένα όλα τα αποτελέσματα για κάθε εξεταστική για κάθε μάθημα καθώς και τα ονόματα των μαθημάτων που υπάρχουν και ο αντίστοιχος κωδικός τους.

Στην βάση δεδομένων για κάθε μάθημα υπάρχει και ένας πίνακας που αποθηκεύονται τα αποτελέσματα από κάθε εξεταστική ανεξάρτητα αν είναι εργαστήριο ή διάλεξη. Η δομή του πίνακα κάθε μαθήματος είναι η εξής :

ID, LAST, PASO, YEAR, EKSAMINO, BATHMOS, PERIODOS και TYPE.

Η Γραμματεία του τμήματος βγάζει ένα αρχείο με τους βαθμούς των φοιτητών για κάθε μάθημα, αν είναι πρώτη εξεταστική η δεύτερη αν είναι εαρινό η χειμερινό το εξάμηνο. Τα αρχεία έχουν την παρακάτω μορφή :

1	XXXXX ΑΓΓΕΛΙΚΗ	111	1998	1	5.00
2	XXXXX ΙΩΑΝΝΗΣ	222	1998	1	0.00
3	XXXXX ΠΑΡΑΣΚΕΥΗ	333	1998	2	5.00
4	XXXXX ΠΑΝΑΓΙΩΤΗΣ	444	1998	2	7.00



Οπότε οι αριθμοί στην αρχή θα αποθηκευτούν στο πεδίο ID, αυτός ο αριθμός δεν χρειάζεται σε αυτό που θέλω να κάνω. Τον αποθηκεύω όμως για να μην αναγκάζομαι να τον σβήνω από κάθε αρχείο. Μετά ακολουθεί το επίθετο με το όνομα του σπουδαστή που αποθηκεύονται στο πεδίο LAST, το πάσο αποθηκεύεται στο PASO, στο πεδίο YEAR αποθηκεύεται το έτος στο EKSAMINO το εξάμηνο και ακολούθως βαθμός στο πεδίο BATHMOS. Οπότε μένουν κενά τα πεδία PERIODOS ,TYPE που αποθηκεύονται η περίοδος και ο τύπος του μαθήματος αν είναι δηλαδή εργαστήριο ή διάλεξη. Τα αρχεία που παίρνω από την γραμματεία έχουν για κατάληξη έναν τριψήφιο αριθμό όπως 991 .

Τα δύο πρώτα ψηφία δηλώνουν το έτος και το τελευταίο την περίοδο για 1 είναι πρώτη χειμερινή για 2 ,δεύτερη χειμερινή ,για 3 είναι πρώτη εαρινή και τέλος για 4 είναι δεύτερη εαρινή. Έτσι ξέρω τι τιμή πρέπει να βάλω στο πεδίο PERIODOS. Όλα τα αρχεία που αφορούν εργαστήρια τα έχω βάλει σε κατάλογο και σε έναν άλλο όλες τις διαλέξεις έτσι όταν εισάγω το κάθε μάθημα εισάγω με ένα πρόγραμμα και αν είναι διάλεξη ή εργαστήριο.

Όπως γίνεται αντιληπτό όλο το site βασίζεται σε αντίγραφα της γραμματείας οπότε και κάποιος να καταφέρει να τροποποιήσει τους βαθμούς του δεν θα καταφέρει να περάσει το μάθημα. Εκτός αυτού έχω κατασκευάσει και μια σελίδα που θα μπορεί η γραμματεία εισάγοντας έναν κωδικό να διαλέξει ένα αρχείο από τον σκληρό της δίσκο και να το στείλει στον Server του Τ.Ε.Ι μαζί με επεξήγηση αν είναι εργαστήριο ή διάλεξη. και αυτός να το επεξεργαστεί και να το φορτώσει. Έτσι η ανακοίνωση των αποτελεσμάτων της κάθε εξεταστικής είναι άμεση.

ΒΑΣΙΚΟ ΚΕΙΜΕΝΟ

Για την υλοποίηση του έργου χρησιμοποιήθηκαν τα ακόλουθα:
Λειτουργικό σύστημα Windows XP Professional Ελληνική έκδοση (της εταιρίας Microsoft www.microsoft.com), για βάση δεδομένων χρησιμοποιήθηκε η MySQL έκδοση 4.0.13 της εταιρίας MySQL AB (www.mysql.com) για το λόγο ότι είναι δωρεάν η χρήση της για μη κερδοσκοπικούς σκοπούς.

Για Web Server χρησιμοποιήθηκε ο Apache (www.apache.org/) έκδοση 2, που είναι ένα Project από διάφορους προγραμματιστές από όλο τον κόσμο. Ο Apache είναι δωρεάν, και ταυτόχρονα ο πιο ασφαλής και ο πιο γρήγορος Web Server.

Για τα Script χρησιμοποιήθηκε η PHP (www.php.net/) έκδοση 4.3.5 που είναι ελεύθερη για χρήση και είναι η αμέσως μετά το CGI η ταχύτερη γλώσσα. Η php, ο apache, η mysql μπορούν να δουλέψουν και κάτω από Unix like συστήματα οπότε ένα άμεσο πλεονέκτημα που έχουμε είναι ότι, αν για κάποιο λόγο χρειαστεί να φύγουμε από την πλατφόρμα των Windows μπορούμε με λίγες αλλαγές στον κώδικα να μεταβούμε σε Unix συστήματα. Για την εγγραφή του κώδικα χρησιμοποιήθηκε μόνο το σημειωματάριο (ένας απλός κειμενογράφος). Ακόμα χρησιμοποιήθηκε JavaScript (www.sun.com) που είναι άλλη μια γλώσσα για Script. Η JavaScript είναι συμβατή με κάθε Browser οπότε ο χρήστης δεν χρειάζεται να εγκαταστήσει τίποτα στον H/Y του.

Χρησιμοποιήθηκε η γλώσσα περιγραφής HTML 4 για την οποιαδήποτε απεικόνιση στον Browser. Για περισσότερες πληροφορίες για την HTML www.w3c.org.

Η HTML είναι στατική γλώσσα και τα περιεχόμενα μιας σελίδας που κατασκευάστηκε αποκλειστικά με αυτήν δεν μπορούν να αλλάξουν.

Εμείς θέλουμε για κάθε φοιτητή να δείχνουμε μόνο τους βαθμούς του οπότε τα περιεχόμενα της σελίδας θα μεταβάλλονται. Εδώ έρχεται η PHP και παίρνει κάποια δεδομένα που στέλνουμε εμείς και τα επεξεργάζεται, δημιουργώντας έτσι σελίδες διαφορετικές για κάθε χρήστη.

Η όλη λογική είναι ότι ο χρήστης στέλνει μια σελίδα με κάποια δεδομένα στον Server και αυτός επεξεργάζεται τα δεδομένα και βάση αυτών παράγει εκείνη την στιγμή μια σελίδα HTML και τη στέλνει πίσω στο χρήστη. Με αυτόν τον τρόπο ο κάθε χρήστης θα βλέπει διαφορετικές σελίδες από τους άλλους, αλλά και από τον εαυτό του, γιατί όταν ανανεώνεται η βάση δεδομένων τότε κατασκευάζεται μια σελίδα με τα τελευταία δεδομένα. Οπότε απαιτείται ελάχιστη συντήρηση.

Ο κώδικας που επεξεργάζεται από τον Server λέγεται Server Side Code. Υπάρχει και κώδικας που επεξεργάζεται από τον Browser λέγεται Client Side. Όταν είναι εφικτό και γίνουν κάποιοι έλεγχοι στον Client καλό είναι να υλοποιούνται εκεί γιατί σε αντίθεση με τον Server Side Code δε χρειάζεται να ταξιδέψουν μέσω του Internet και στο τέλος να διαπιστωθεί ότι δεν είναι σωστά τα δεδομένα (π.χ. άμα βάλει κάποιος αρνητική τιμή σε κάποιο πεδίο ηλικίας). Ένα άλλο πλεονέκτημα που κερδίζουμε είναι ότι ο Server επιφορτίζεται με λιγότερο έργο μιας και το έχει αναλάβει ο Client.

Ακόμα χρησιμοποιήθηκε Visual Basic Script, που είναι ήδη εγκατεστημένο μαζί με τα Windows.

Γενικά με τον όρο Script εννοούμε ένα υποσύνολο μιας γλώσσας προγραμματισμού. Συνήθως τα Script δεν γίνονται compile αλλά μεταγλωττίζονται

την ώρα της εκτέλεσής τους. Σε αυτήν την κατηγορία ανήκουν τα Visual Basic , Java ,C # scripts.

ΕΓΚΑΤΑΣΤΑΣΕΙΣ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ

Υπάρχουν δύο τρόποι για να εγκαταστήσεις PHP και Apache. Ο πρώτος είναι να εγκαταστήσεις μόνος σου την PHP και τον APACHE και μετά αν τροποποιήσεις το αρχείο httpd.conf που είναι το αρχείο ρυθμίσεων του APACHE ώστε να εργάζεται αρμονικά με την PHP. Ο δεύτερος τρόπος είναι να κατεβάσεις το πακέτο AppServ από την σελίδα <http://www.appservnetwork.com/> . Το AppServ είναι ένα ανοιχτό Project που περιέχει την PHP και τον Apache σε ένα πακέτο. Επειδή δεν χρειάζονται οι ίδιες ρυθμίσεις για κάθε έκδοση της PHP με τον Apache το AppServ κάνει αυτόματα της ρυθμίσεις που είναι απαραίτητες ώστε να επικοινωνούν χωρίς προβλήματα.

Αν τρέχει ένας άλλος Server στον Η/Υ όπως ο IIS θα πρέπει να τον απενεργοποιήσουμε γιατί και οι δύο Server θα ακροάζονται το port 80 (αυτό το port είναι το default port για κάθε WEB Server) και θα υπάρχει conflict.

Οπότε ο Apache μπορεί να λειτουργεί κανονικά αλλά να μην επεξεργάζεται τα αιτήματα γιατί θα εργάζεται ο IIS σε αυτό το port.

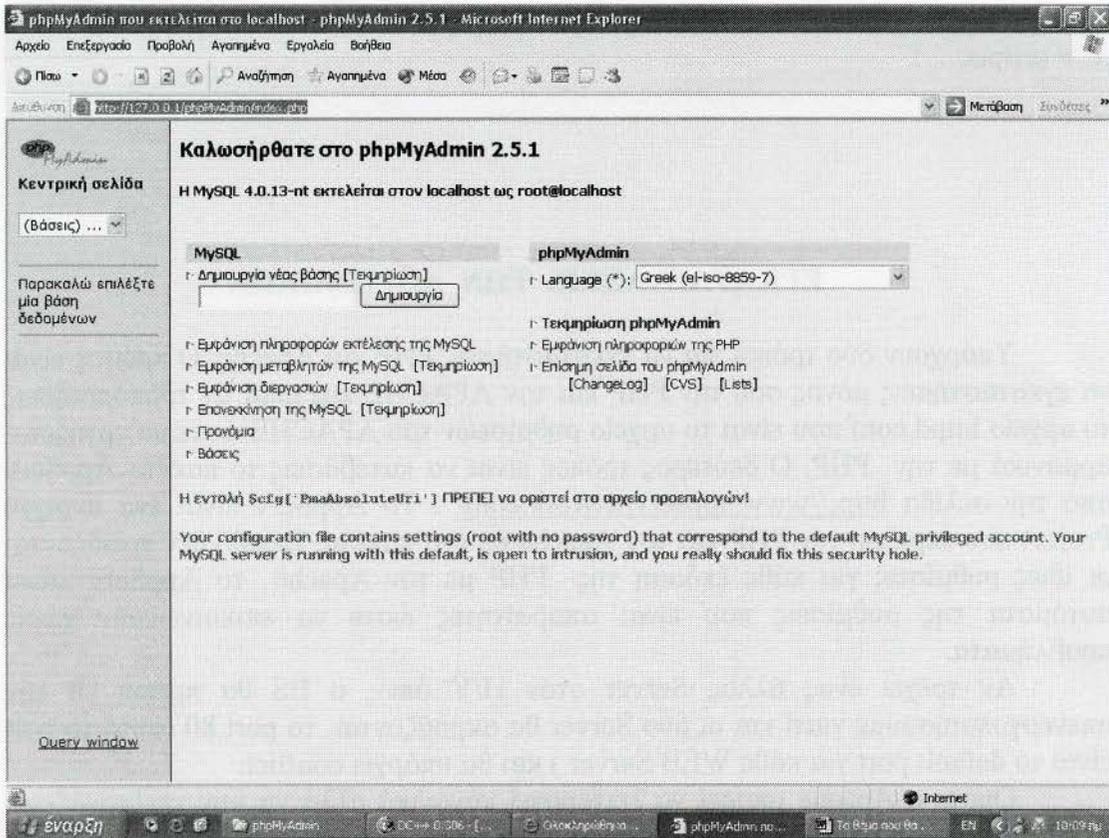
Το AppServ δημιουργεί στο root έναν κατάλογο με όνομα AppServ και μέσα σε αυτόν υπάρχουν οι κατάλογοι για την PHP, MySQL, Apache.

Στον κατάλογο C:\AppServ\www αποθηκεύονται οι σελίδες που θα βλέπει ο χρήστης.

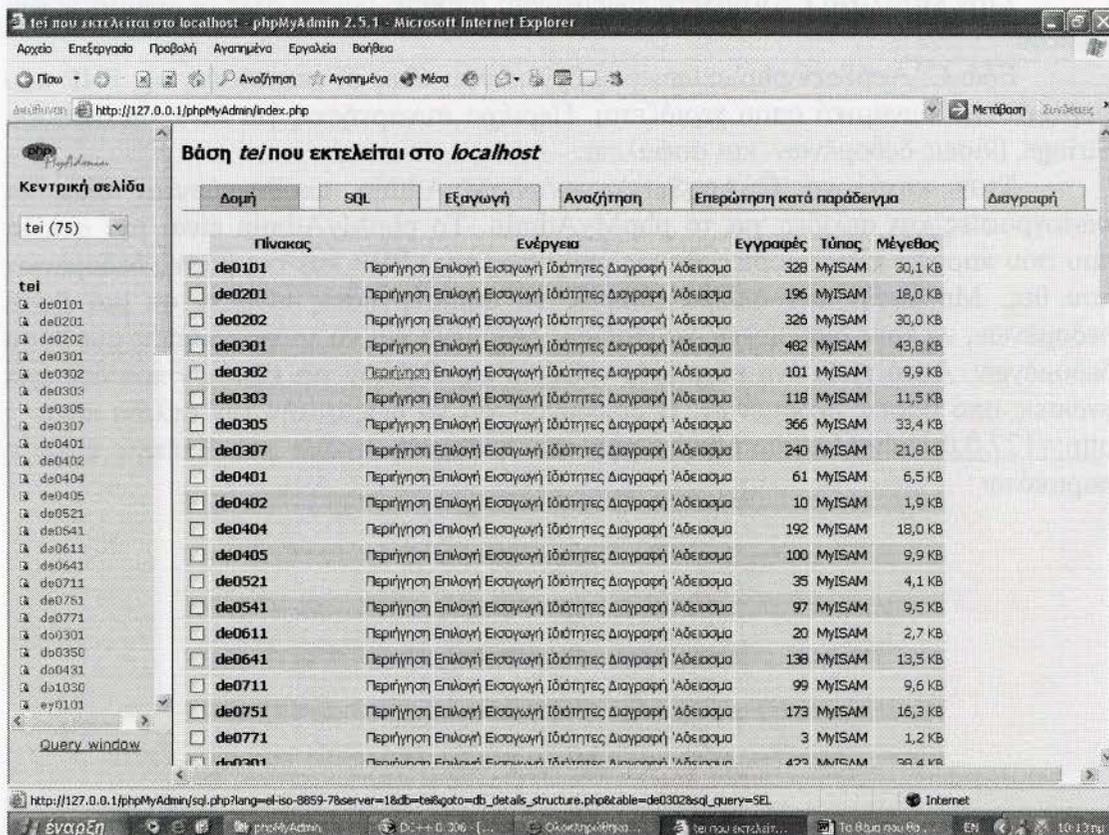
Στον κατάλογο C:\AppServ\apache\conf αποθηκεύονται όλες οι ρυθμίσεις του Apache.

Εδώ C:\AppServ\php\extensions βρίσκονται όλες οι επεκτάσεις της PHP που φορτώνονται δυναμικά όπου χρειάζεται. Περιέχει συναρτήσεις για Java για διάφορα Strings, βάσεις δεδομένων και ασφάλεια.

Στον κατάλογο C:\AppServ\www\phpMyAdmin αποθηκεύονται όλες οι φωτογραφίες και σελίδες για το phpMyAdmin. Το phpMyAdmin είναι μια σελίδα που σου παρέχει πληροφορίες για τις ρυθμίσεις της PHP και της βάσης δεδομένων που θες. Με αυτήν την σελίδα μπορείς να δημιουργήσεις πίνακες σε μια βάση δεδομένων, να τους διαγράψεις να εισάγεις δεδομένα και να τροποποιήσεις μια βάση δεδομένων. Αυτό είναι ένα εύχρηστο γραφικό περιβάλλον για κάποιον που δεν έχει γνώσεις από βάσεις δεδομένων. Η διεύθυνση για να δεις αυτήν την σελίδα είναι : <http://127.0.0.1/phpMyAdmin/index.php> και η αρχική εικόνα που βλέπεις είναι η παρακάτω :



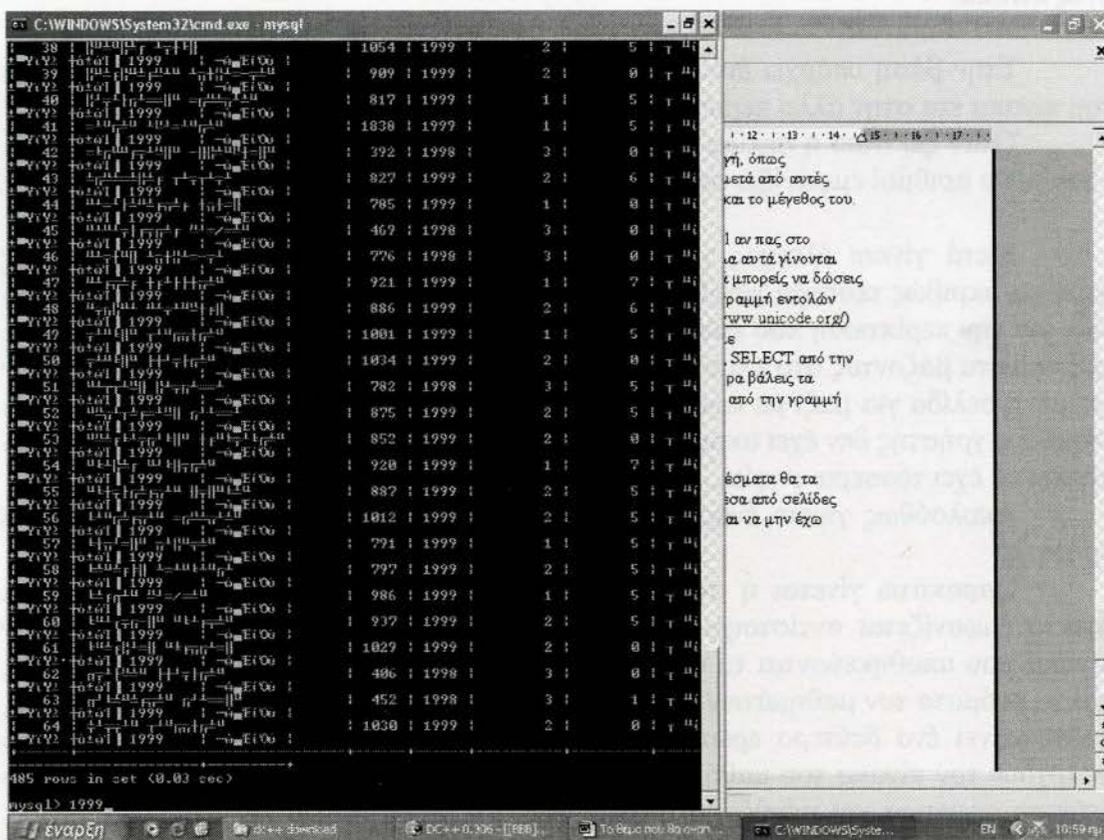
Όπως φαίνεται και στην εικόνα στην πάνω αριστερή μεριά γράφει Βάσεις και ανοίγοντας το μενού επιλέγεις όποια βάση θες εσύ να διαχειριστείς. Αφού επιλέξεις μια βάση βλέπεις μια παρόμοια εικόνα σαν την παρακάτω.



Τώρα κάτω από το μενού, που έγραφε βάσεις, γράφει το όνομα της βάσης δεδομένων που επέλεξες και από κάτω σου εμφανίζει τους πίνακες που έχει η επιλεγμένη βάση. Στο κέντρο της οθόνης και πάνω σου εμφανίζεται ένα μενού με ενέργειες που μπορείς να κάνεις πάνω στους πίνακες. Κάτω από αυτό το μενού εμφανίζεται το όνομα του κάθε πίνακα και δίπλα σε αυτό μια επιλογή, όπως περιήγηση, επιλογή, εισαγωγή, ιδιότητες διαγραφή, άδειασμα και μετά από αυτές εμφανίζετε ο αριθμός των εγγραφών που έχει, ο τύπος του καθώς και το μέγεθος του.

Τις αντίστοιχες εντολές μπορείς να της δώσεις στην MySQL αν πας στο κατάλογο C:\AppServ\mysql\bin και γράψεις mysql.exe .Φυσικά όλα αυτά γίνονται από την γραμμή εντολών DOS-Prompt (MS DOS PROMPT). Μετά μπορείς να δώσεις κατευθείαν εντολές στην Mysql για να διαχειριστείς την βάση. Η γραμμή εντολών λειτουργεί με ASCII χαρακτήρες και η PHP με UNICODE (<http://www.unicode.org/>)

Έτσι υπάρχει ασυμβατότητα και τα αποτελέσματα εμφανίζονται με ακαταλαβίστικους χαρακτήρες στην σελίδα PHP αλλά όταν κάνεις SELECT από την γραμμή εντολών σου εμφανίζει κανονικά τα αποτελέσματα. Αν τώρα βάλεις τα δεδομένα μέσα στην βάση από την PHP και κάνεις SELECT από την γραμμή εντολών θα δεις από εκεί τώρα τα ακαταλαβίστικα.



Έτσι η μοναδική λύση που ήταν εφικτή μιας και τα αποτελέσματα θα τα έβλεπες μέσα από μια σελίδα PHP ήταν να γίνει και η εισαγωγή μέσα από σελίδες PHP ώστε να μην χρησιμοποιήσω καθόλου την γραμμή εντολών και να μην έχω προβλήματα ασυμβατότητας.

ΕΠΕΞΗΓΗΣΗ ΤΩΝ ΑΡΧΕΙΩΝ

Η κατασκευή του site αποτελείται από τα παρακάτω αρχεία : auto_import.php, index.html, javascript.html, loadData.php, op.php , results.php, script.vbs , startupload.html, upload.php. Το αρχείο index.html είναι η αρχική σελίδα, που περιέχει ένα πεδίο κειμένου που ο χρήστης μπορεί να εισάγει το πάσο του και ένα κουμπί για να στο στείλει στην επόμενη σελίδα που είναι η results.php. Η results.php είναι η σελίδα που συνδέεται με την βάση και εμφανίζει τα αποτελέσματα στη οθόνη.

Για κάθε μάθημα σχεδιάστηκε και ένας πίνακας στην βάση. Αυτό έγινε για τους εξής λόγους :

α) Ασφάλεια. Αν για κάποιο λόγο πάθει κάτι ένας πίνακας για ένα μάθημα, ο φοιτητής θα μπορεί έτσι να δει τους βαθμούς από τα υπόλοιπα μαθήματα. Λόγο του ότι ο κάθε πίνακας περιέχει μικρό όγκο δεδομένων σε σχέση αν όλα τα αποτελέσματα αποθηκευόταν σε ένα πίνακα, έτσι θα μπορεί να παρθούν ευκολότερα αντίγραφα δεδομένων. Με αυτόν τον τρόπο όταν θα γίνεται κάποια αλλαγή και είναι λάθος θα επηρεάζει μόνο τον συγκεκριμένο πίνακα.

β) Ευελιξία στην συντήρηση. Αν καταργηθεί ένα μάθημα ή προστεθεί ένα καινούριο δεν χρειάζεται να γίνει καμιά αλλαγή εκτός από τον προσθήκη η αφαίρεση ενός πίνακα.

Στην βάση υπάρχει ένας πίνακας με δυο στήλες, στην μια περιέχει το όνομα του πίνακα και στην άλλη περιέχει το όνομα του μαθήματος.

Όταν ξεκινάει η σελίδα κάνει έλεγχο για αυτό που έδωσε ο χρήστης. Αν δεν είναι μόνο αριθμοί εμφανίζει το μήνυμα «Το πάσο που εισάγατε δεν είναι σωστό...» .

Μετά γίνεται έλεγχος για να διαπιστωθεί αν αυτό που έδωσε ο χρήστης περιέχει ακριβώς τέσσερα ψηφία. Αυτός ο έλεγχος γίνεται για λόγους ασφαλείας του site, για την περίπτωση που κάποιος κακόβουλος κλείσει το <input> της HTML χειροκίνητα βάζοντας στο πλαίσιο κειμένου ">" και προσπαθήσει να το στείλει στην επόμενη σελίδα για μαζί με κώδικα για να εκτελεστεί. Στην περίπτωση που αυτό που έστειλε ο χρήστης δεν έχει ακριβώς τέσσερα ψηφία εμφανίζεται το μήνυμα «Το πάσο πρέπει να έχει τέσσερα ψηφία...» και τερματίζει αμέσως.

Ακολούθως γίνεται ένας έλεγχος για αν δεν έχει τιμή αυτό που εισήγαγε ο χρήστης.

Παρακάτω γίνεται η σύνδεση με την βάση δεδομένων. Αν αυτό δεν είναι επιτυχές εμφανίζεται αντίστοιχο μήνυμα. Μετά διαβάζει όλα τα δεδομένα από τον πίνακα που αποθηκεύονται τα ονόματα των άλλων πινάκων μέσα στην βάση καθώς και τα ονόματα των μαθημάτων. Κάθε εγγραφή θα περιέχει το όνομα ενός μαθήματος, οπότε κάνει ένα δεύτερο ερώτημα στην βάση και διαβάζει όλα τα δεδομένα που υπάρχουν τον πίνακα του μαθήματος και τα εμφανίζει στην οθόνη, έτσι τελειώνει το δεύτερο ερώτημα και επανέρχεται στην δεύτερη εγγραφή του πρώτου ερωτήματος, που τώρα περιέχει το όνομα του επόμενου μαθήματος και το όνομα του πίνακα που είναι αποθηκευμένα τα αποτελέσματα για αυτό το μάθημα, έτσι τροφοδοτείται πάλι το δεύτερο ερώτημα στην βάση αυτήν την φορά με νέο όνομα μαθήματος. Αφού εμφανίσει το δεύτερο ερώτημα τα αποτελέσματα στην οθόνη τότε τελειώνει και επανερχόμαστε στο πρώτο ερώτημα που περιέχει το όνομα από το τρίτο μάθημα έτσι αυτό επαναλαμβάνεται συνέχεια μέχρι να εξαντληθούν όλοι οι πίνακες των μαθημάτων από την βάση.

Τα αποτελέσματα εμφανίζονται σε μορφή πίνακα. Αν κάποιος φοιτητής έχει δώσει το μάθημα πάνω από μία φορές την μια φορά που έδωσε το μάθημα θα του εμφανίζεται με κίτρινο χρώμα και την άλλη με πορτοκαλί.

Πρώτα εμφανίζεται το όνομα του μαθήματος μετά το όνομα του φοιτητή μετά το αριθμός του πάσου του μετά η χρονιά εγγραφής, το εξάμηνο που φοιτούσε όταν έδωσε το συγκεκριμένο μάθημα, ακολουθεί ο βαθμός του, η περίοδος που το έδωσε και ο τύπος, δηλαδή αν ήταν εργαστήριο ή διάλεξη.

The screenshot shows a web browser window with the URL <http://127.0.0.1/results.php>. The browser title is "Ηλεκτρονική Βαθμολογία - Microsoft Internet Explorer". The table displayed contains the following data:

Όνομα μαθήματος	Όνομα	Πάσο	Χρονιά εγγραφής	Εξάμηνο	Βαθμός	Περίοδος	Εργαστήριο ή Διάλεξη
ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Η/Υ I (PASCAL) - (ΠΑΛΙΟΙ)	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	5	A Περίοδος Εαρινή 2004	Διάλεξη
ΔΙΚΤΥΑ Η/Υ III	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	6	5.5	A Περίοδος Εαρινή 2002	Διάλεξη
ΕΙΣΑΓΩΓΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΔΙΟΙΚΗΣΗΣ	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	9	A Περίοδος Εαρινή 2004	Διάλεξη
ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ (SOFTWARE) (ΠΑΛΙΟΙ)	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	1	8	A Περίοδος Χειμερινή 2004	Διάλεξη
ΕΙΣΑΓΩΓΗ ΣΤΟ MANAGEMENT	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	1	7	A Περίοδος Χειμερινή 2004	Διάλεξη
ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ Η/Υ	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	9	A Περίοδος Εαρινή 2004	Διάλεξη
ΜΑΘΗΜΑΤΙΚΑ I	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	1	0	A Περίοδος Χειμερινή 2004	Διάλεξη
ΜΑΘΗΜΑΤΙΚΑ III	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	0	A Περίοδος Εαρινή 2004	Διάλεξη
ΜΑΘΗΜΑΤΙΚΑ III	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	5.5	B Περίοδος Εαρινή 2004	Διάλεξη
ΜΑΚΡΟΟΙΚΟΝΟΜΙΑ	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	2	5	A Περίοδος Εαρινή 2004	Διάλεξη
ΣΥΓΧΡΟΝΑ ΤΗΛ/ΚΑ ΣΥΣΤΗΜΑΤΑ & ΨΗΦΙΑΚΗ	ΣΙΛΙΩΝΗΣ ΠΕΤΡΟΣ	1962	2000	6	3	A Περίοδος Εαρινή 2002	Διάλεξη

Το αρχείο script.vbs έχει σαν σκοπό να καταγράψει όλα τα αρχεία που υπάρχουν σε ένα κατάλογο και να τα αποθηκεύσει σε ένα αρχείο κειμένου. Τα αρχεία αυτά έχουν ονόματα με Ελληνικούς χαρακτήρες που δεν αναγνωρίζει η βάση και πρέπει να αλλάξουν σε Αγγλικούς.

Μετά τρέχουμε το αρχείο or.php που έχει σαν σκοπό να αλλάξει το ονόματα των αρχείων σε Αγγλικούς χαρακτήρες. Έτσι όταν ξεκινά το αρχείο δημιουργείτε ένας δισδιάστατος πίνακας, όπου στην μια σειρά υπάρχουν οι Ελληνικοί χαρακτήρες και στην άλλη σειρά οι Αγγλικοί. Έτσι στην πρώτη θέση της πρώτης γραμμής υπάρχει το Αγγλικό γράμμα 'A' και στην πρώτη θέση της δεύτερης γραμμής υπάρχει το Ελληνικό γράμμα 'Α' και αυτό συνεχίζεται για όλα τα γράμματα του Ελληνικού αλφαβήτου.

Μετά ανοίγει το αρχείο c:\testfile.txt που δημιούργησε το script.vbs και για κάθε σειρά διαβάζει το όνομα του αρχείου που είναι στα Ελληνικά.

Με τις κατάλληλες συναρτήσεις αφαιρεί την κατάληξη και κρατάει μόνο το όνομα.

Αφού απομονώνει ένα ένα τα γράμματα από το όνομα του αρχείου τα αντικαταστέ με το αντίστοιχο γράμμα του Αγγλικού αλφαβήτου και κατασκευάζει το αντίστοιχο όνομα με Αγγλικούς χαρακτήρες.

Μετά εκτελείται μια εντολή του συστήματος για μετονομασία αρχείου μέσω της PHP που παίρνει δυο παραμέτρους μια το αρχικό όνομα και μια το νέο όνομα που της να έχει. Και με αυτόν τον τρόπο γίνεται μετονομασία και τα αρχεία έχουν Αγγλικούς χαρακτήρες πλέον.

Τώρα σβήνεις το αρχείο `c:\testfile.txt` μια και δεν σου χρειάζεται και τρέχεις ξανά το `script.vbs` έτσι τώρα δημιουργείς πάλι το αρχείο `c:\testfile.txt` που περιέχει τα ονόματα των αρχείων αυτή την φορά με Αγγλικούς χαρακτήρες.

Κατασκευάζεις στην βάση έναν πίνακα με δύο στήλες αλφαριθμητικών που στην μια θα αποθηκεύεται το όνομα του πίνακα (δηλαδή ο κωδικός του μαθήματος π.χ. `gy0101`) και στην άλλη θα αποθηκεύεται η περιγραφή π.χ. Μαθηματικά .

Για να φορτώσεις τους κωδικούς των μαθημάτων, τρέχεις το αρχείο `loadData.php`. Αυτό ανοίγει το αρχείο των κωδικών των μαθημάτων `c:\testfile.txt` και για κάθε σειρά παίρνει το όνομα του μαθήματος χωρίς την κατάληξη και εισάγει τα ονόματα των μαθημάτων (`gy0101`) στον πίνακα `tbl` της βάσης ,τόρα εισάγουμε από ένα αρχείο που έχουμε δημιουργήσει τις περιγραφές των ονομάτων (Μαθηματικά, Φυσική).

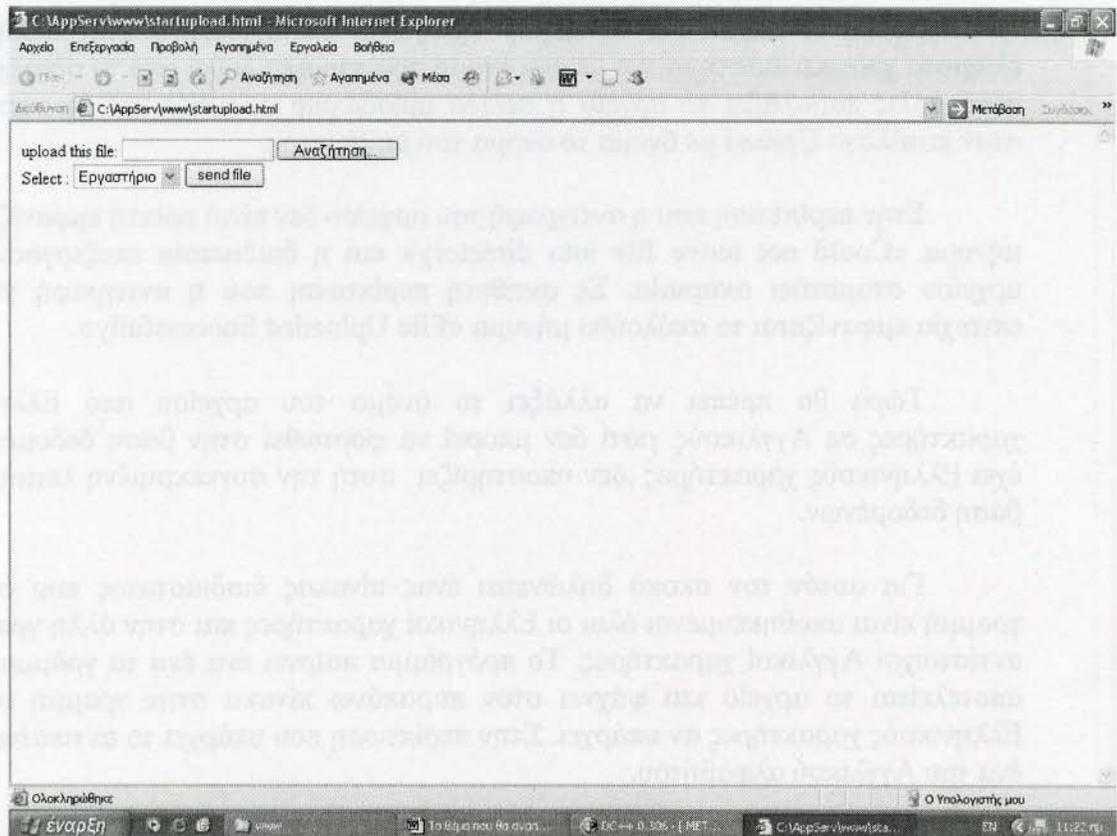
Παρακάτω απομονώνει την κατάληξη και ανάλογα με την κατάληξη κατασκευάζει το έτος και την περίοδο που αναφέρεται η βαθμολογία. Αφού το γέμισμα του πίνακα με τους κωδικούς των μαθημάτων έχει ολοκληρωθεί μετά βρίσκει στο σκληρό δίσκο ένα ένα τα αρχεία και μετά τα βάζει στον αντίστοιχο πίνακα στη βάση. Ακριβός αμέσως κάνει ένα `update` στον πίνακα που έγινε η εισαγωγή για να βάλει το έτος την περίοδο όπου είναι `NULL` (με την εισαγωγή του αρχείου στην βάση δεν εισάγονται το έτος και η περίοδος έτσι είναι κενά τα αντίστοιχα πεδία).

ΑΥΤΟΜΑΤΗ ΕΙΣΑΓΩΓΗ ΑΠΟ ΤΗΝ ΓΡΑΜΜΑΤΕΙΑ.

Η παραπάνω λειτουργία αφορούσε την εισαγωγή στην βάση όσων βαθμών υπήρχαν μέχρι σήμερα. Έτσι έπρεπε να κατασκευαστεί και μια σελίδα που θα βάζει ένα ένα τα αρχεία μόνη της η γραμματεία, με αποτέλεσμα την αυτόματη ανανέωση αλλά και την άμεση ανανέωση.

Με αυτή την λογική κατασκευάστηκε μια σελίδα που δεν θα υπάρχει στο site του T.E.I. link σε αυτή την σελίδα, για να μειώσουμε της πιθανότητες κάποιος εξωτερικός παράγοντας να εισάγει στην βάση ψευδή δεδομένα, αν και αυτό είναι πολύ δύσκολο μιας και δεν θα γνωρίζει τον αριθμό των πεδίων των πινάκων καθώς και τον τύπο των δεδομένων που απαιτεί ο πίνακας. Ακόμα το site θα δίνει την δυνατότητα στην γραμματεία να εισάγει ένα κωδικό και αν ο κωδικός είναι έγκυρος τότε να γίνετε η εισαγωγή του αρχείου στην βάση.

Η εικόνα που θα έχει περίπου η σελίδα θα είναι η παρακάτω :



Όπως φαίνεται στην εικόνα θα υπάρχει ένα κουμπί «Αναζήτηση...» που θα το χρησιμοποιεί ο χρήστης για να βρει το αρχείο με την βαθμολογία που θέλει να εισάγει στην βάση δεδομένων.

Από κάτω ακριβώς θα μπορεί να επιλέξει αν το αρχείο που θα εισαχθεί αφορά εργαστήριο ή διάλεξη και μετά ο χρήστης θα πατά το κουμπί «send file» και το αρχείο θα στέλνεται στον server του T.E.I. και αλλάζει το όνομά του σε αγγλικούς χαρακτήρες για να μπορέσει να εισαχθεί στην βάση δεδομένων και μετά γίνεται η εισαγωγή του.

Στον Server του θα δημιουργηθεί ένας κατάλογος με όνομα upload όπου θα μεταφέρονται όλα τα αρχεία που θα στέλνει η γραμματεία.

Η όλη διαδικασία γίνεται από τα παρακάτω δύο αρχεία.:
startupload.html , upload.php.

Όταν ξεκινάει το αρχείο upload.php παίρνει την παράμετρο που έστειλε η σελίδα startupload.html δηλαδή αν είναι εργαστήριο η διάλεξη. Μετά γίνονται κάποιοι έλεγχοι όπως αν έχει όντως στείλει κάποιο αρχείο η γραμματεία, σε περίπτωση που δεν έχει σταλεί τίποτα, τότε η όλη διαδικασία τερματίζεται και εμφανίζεται το μήνυμα «Problem : No File Uploade».

Ο δεύτερος έλεγχος που γίνεται αφορά το μέγεθος. Ο χρήστης μπορεί όντως να έχει στείλει κάποιο αρχείο για εισαγωγή αλλά το αρχείο αυτό μπορεί να έχει μηδενικό μέγεθος, σε αυτή την περίπτωση πάλι τερματίζεται η διαδικασία και εμφανίζεται το ακόλουθο μήνυμα «Problem : Uploaded file is zero length».

Μόλις το παραλάβει το αρχείο παραμένει αποθηκευμένο στον σκληρό για ελάχιστο χρονικό διάστημα και με ένα όνομα που του έχει δοθεί από το σύστημα. Για αυτό μόλις παραλάβει το αρχείο η σελίδα upload.php φροντίζει να το αντιγράψει στον κατάλογο Upload με όνομα το όνομα του μαθήματος.

Στην περίπτωση που η αντιγραφή του αρχείου δεν είναι εφικτή εμφανίζεται το μήνυμα «Could not move file into directory» και η διαδικασία επεξεργασίας του αρχείου σταματάει ακαριαία. Σε αντίθετη περίπτωση που η αντιγραφή γίνει με επιτυχία εμφανίζεται το ακόλουθο μήνυμα «File Uploaded Successfully»

Τώρα θα πρέπει να αλλάξει το όνομα του αρχείου από Ελληνικούς χαρακτήρες σε Αγγλικούς γιατί δεν μπορεί να φορτωθεί στην βάση δεδομένων αν έχει Ελληνικούς χαρακτήρες ,δεν υποστηρίζει αυτή την συγκεκριμένη λειτουργία η βάση δεδομένων.

Για αυτόν τον σκοπό δηλώνεται ένας πίνακας δισδιάστατος που στη μια γραμμή είναι αποθηκευμένοι όλοι οι Ελληνικοί χαρακτήρες και στην άλλη γραμμή οι αντίστοιχοι Αγγλικοί χαρακτήρες. Το πρόγραμμα παίρνει ένα ένα τα γράμματα που αποτελείται το αρχείο και ψάχνει στον παραπάνω πίνακα στην γραμμή με τους Ελληνικούς χαρακτήρες αν υπάρχει. Στην περίπτωση που υπάρχει το αντικαταστεί με ένα του Αγγλικού αλφαβήτου.

Όταν ολοκληρωθεί αυτή η διαδικασία το πρόγραμμα μετονομάζει το αρχείο που υπάρχει στον σκληρό δίσκο.

Μετά φορτώνει το αρχείο την βάση. Ανάλογα με την κατάληξη που έχει το αρχείο βάζει σε μια μεταβλητή την περίοδο που αντιστοιχεί στην κατάληξη π.χ. αν έχει κατάληξη .991 σημαίνει πως η βαθμολογία αφορά το έτος 1999 και είναι για την πρώτη εξεταστική του χειμερινού εξαμήνου, αν έχει κατάληξη 992 σημαίνει πως αφορά το έτος 1999 και την δεύτερη εξεταστική ,η κατάληξη .993 σημαίνει πως είναι για το έτος 1999 για την πρώτη εαρινή εξεταστική και αν έχει κατάληξη .994 σημαίνει πως είναι για το έτος 1999 και την δεύτερη εξεταστική του εαρινού εξαμήνου.

Οπότε μετά την εισαγωγή των βαθμών στον πίνακα γίνεται μια ανανέωση αυτών των εγγραφών που δεν έχουν έτος και περίοδο εξεταστικής.

Ο Συνολικός κώδικας.

01. Κώδικας του αρχείου script.vbs.

```
Dim fso, fl,f,fc
Set fso = CreateObject("Scripting.FileSystemObject")
Set fl = fso.CreateTextFile("c:\testfile.txt", True)
Set f = fso.GetFolder("c:\ergastiria")
Set fc = f.Files
  For Each fil in fc
    fl.writeLine(fil.name)
  Next
```

02. Κώδικας του αρχείου index.html

```
<html>
  <body>
    <form method="post" action="results.php">
      <center>paso:<input type="text"
name="paso"></center><br>
      <center>submit:<input type="submit"
name="send"></center>
    </form>
  </body>
</html>
```

03.Κώδικας του αρχείου loadData.php .

```
<html>
  <head><title>auto batch import</title>
  </head>
<body>
<?

$link_main=mysql_connect("localhost","noc1","");//alazis to onoma tou
ipologisti ama xriazete
```

```

if(!$link_main)
    echo "lathos sti sindesi main";

    mysql_select_db("test2",$link_main);//alazis tin bazi ama
xriazete

$fp=fopen("c:\\testfile.txt","r");

//trabas apo ton pinana tbl ta onomata ton pinakon pou prepi na
dimiourgithoun kai kataskebazis tous pinakes

/*
$in="select name from tbl group by name";
$tbls=mysql_query("$in",$link_main);

while($row=mysql_fetch_row($tbls))
{
    $txt="CREATE TABLE $row[0] (id int(11),last varchar(60),
    paso int(11), year int(11) , eksamino int(11) ,
    bathmos double ,periodos varchar(40) ,type varchar(40) )";

    mysql_query("$txt",$link_main);
}
*/

//OSA INE TO ONOMATA TON ARXION TOSA INE KAI TA ARXIA

while(!feof($fp))
{
    $sword=trim(fgets($fp,100));
    $megethos=strlen($sword);
    $megethos=$megethos-4;
    $str1=substr($sword,-3);//ton tripsifio aritho apo to telos
    $str2=substr($str1,0,2);//to etos
    $str3=substr($str1,2,2);//tin periodo
    $str4=substr($sword,0,$megethos);//mono to onoma apo to
arxio xoris tin kataliksi
    echo"<br>";
}

```

```
//auto bazi mesa stin bazi tei ston pikana tbl ta onomata apo ta arxia  
gia na ta paris meta me group by
```

```
//tha trelksi mia fora auto, gia auton ton logo ine se sxolia
```

```
//$in="insert into tbl values('$str4')";
```

```
//mysql_query("$in",$link_main);
```

```
if($str2==90)
```

```
    $etos=1990;
```

```
if($str2==91)
```

```
    $etos=1991;
```

```
if($str2==92)
```

```
    $etos=1992;
```

```
if($str2==93)
```

```
    $etos=1993;
```

```
if($str2==94)
```

```
    $etos=1994;
```

```
if($str2==95)
```

```
    $etos=1995;
```

```
if($str2==96)
```

```
    $etos=1996;
```

```
if($str2==97)
```

```
    $etos=1997;
```

```
if($str2==98)
```

```
    $etos=1998;
```

```
if($str2==99)
```

```
    $etos=1999;
```

```
if($str2==00)
```

```
    $etos=2000;
```

```
if($str2==01)
```

```
    $etos=2001;
```

```
if($str2==02)
```

```
    $etos=2002;
```

```
if($str2==03)
```

```
    $etos=2003;
```

```
if($str2==04)
```

```
    $etos=2004;
```

```
if($str2==05)
```

```
    $etos=2005;
```

```

if($str2==06)
    $setos=2006;
if($str2==07)
    $setos=2007;
if($str2==08)
    $setos=2008;
if($str2==09)
    $setos=2009;

//edo gemizoume tous pinakes me ta stixia apo ta arxia

    // edo theli tria "/" gia to path kai tessera giat o arxio
    $load="load data infile 'C:\\\\ergastiria\\\\$word' into table $str4";
    echo "$load";
    echo "<br>";
    mysql_query("$load",$link_main);
    echo mysql_errno() . ": " . mysql_error(). "\n";

    if($str3==1)
        mysql_query("update $str4 set periodos='A Περίοδος Χειμερινή
$setos',type='Διάλεξη' where periodos is null",$link_main);
    if($str3==2)
        mysql_query("update $str4 set periodos='B Περίοδος Χειμερινή
$setos',type='Διάλεξη' where periodos is null",$link_main);
    if($str3==3)
        mysql_query("update $str4 set periodos='A Περίοδος Εαρινή
$setos',type='Διάλεξη' where periodos is null",$link_main);
    if($str3==4)
        mysql_query("update $str4 set periodos='B Περίοδος Εαρινή
$setos',type='Διάλεξη' where periodos is null",$link_main);

}

fclose($fp);

```

```
?>
</body>
</html>
```

03. Κώδικας του αρχείου or.php.

```
<html>
<META http-equiv=Content-Type content="text/html; charset=iso-
8859-7">
  <head>
    <title>ade gia</title>
  </head>
  <body>

<?

//diabazi ena erxio kimenou kai metatrepi tis leksis apo ellinikous
xaraktires se aglikous
//kai meta metonomazi ola ta arxia pou exoun ellinikous xaraktires se
adistixous agglikous
//alazis to path pou ine o katalogos me ta arxia kimenou to telos autou
to script

$alph=array(array("A","B","G","D","E","Z","I","TH","I","K","L","M
",
    "N","KS","O","P","R","S","T","Y","F","X","PS","W"),
    array("A","B","Γ","Δ","E","Z","H","Θ","I","K","Λ","M",
    "N","Ξ","O","Π","P","Σ","T","Y","Φ","X","Ψ","Ω"));
$new;
$flag1=0;
$flag2=0;

$fp=fopen("c:\\testfile.txt","r");

while(!feof($fp))
{
$new="";
$word=substr(fgets($fp,100),0,10);
```

```

$str1=substr($word,0,1);
$str2=substr($word,1,1);
$str3=substr($word,2,8);
for($j=0;$j<2;$j++)
    for($i=0;$i<24;$i++)
    {
        if(($alph[1][$i]==$str1)and($flag1==0))
        {
            $new=$new.$alph[0][$i];
            $flag1=1;
        }
        if(($alph[1][$i]==$str2)and($flag2==0))
        {
            $new=$new.$alph[0][$i];
            $flag2=1;
        }
    }
$flag1=0;
$flag2=0;
$new=$new.$str3;
exec("RENAME C:\\ergastiria\\$word $new");
echo"<br>";
}
fclose($fp);
echo "OK";
?>
</body>
</html>

```

04. Κώδικας του αρχείου upload.php .

```

<html>
    <body>
<?
$tipos=$_REQUEST['tipos']; //perni apo tin forma starupload an ine
ergastirio i dialeksi
//=====

```

```

//na fiakso na perni apo tin proigoumeni forma me
combo button an tha ine dialeksi
//h argastirio kai na to alazi mono tou

//=====
=====

//arxi anebasmatos tou arxiou kai apothikeusis tou ston katalogo upload
$userfile_name=$HTTP_POST_FILES['userfile']['name'];

echo $userfile_name;

if ($userfile_name==null)
{
    echo "Problem : No File Uploaded";
    exit;
}

if ($userfile_size==0)
{
    echo "Problem : Uploaded file is zero length";
    exit;
}

//if ($userfile_type!="text/plain")
//{
//    echo "Problem :File is not text";
//    exit;
//}

$upfile="c:\\AppServ\\www\\upload\\".$userfile_name;
if(!copy($userfile,$upfile))
{
    echo "Problem : Could not move file into directory";
    exit;
}

echo "File Uploaded Successfully<br><br>";
//telos anebasmatos arxiou kai apothikeusis tou ston katalogo upload
//=====
=====

```

```
$alph=array(array("A","B","G","D","E","Z","I","TH","I","K","L","M",
    "N","KS","O","P","R","S","T","Y","F","X","PS","W"),
    array("A","B","Γ","Δ","E","Z","H","Θ","I","K","Λ","M",
    "N","Ξ","O","Π","P","Σ","T","Y","Φ","X","Ψ","Ω"));
```

```
$new="";
```

```
$word=trim($userfile_name);
```

```
    $megethos=strlen($word);
```

```
    $megethos=$megethos-4;
```

```
    $str1=substr($word,-3);//ton tripsifio aritho apo to telos
```

```
    $str2=substr($str1,0,2);//to etos
```

```
    $str3=substr($str1,2,2);//tin periodo
```

```
    $str4=substr($word,0,$megethos);//mono to onoma apo to
```

```
arxio xoris tin kataliksi
```

```
    $str6=substr($word,-8);//to onoma xoris ta arxika gramata
```

```
for ($j=0;$j<4;$j++)
```

```
{
```

```
    $str5=$word{$j};//apomonini ena ena ta gramata
```

```
        for($i=0;$i<24;$i++)
```

```
        {
```

```
            $a=$alph[1][$i];
```

```
            if ($a==$str5)
```

```
            {
```

```
                $a=$alph[0][$i];
```

```
                $new=$new.$a;
```

```
            }
```

```
        }
```

```
    }
```

```
$new=$new.$str6;
```

```
exec("RENAME C:\\AppServ\\www\\upload\\$word $new");
```

```
=====
```

```

//arxi

//sindesi me tin basi
$link_main=mysql_connect("localhost","noc1","");//alazis to onoma tou
ipologisti ama xriazete

if(!$link_main)
    echo "lathos sti sindesi main";

mysql_select_db("test2",$link_main);//alazis tin bazi ama
xriazete

if($str2==90)
    $setos=1990;
if($str2==91)
    $setos=1991;
if($str2==92)
    $setos=1992;
if($str2==93)
    $setos=1993;
if($str2==94)
    $setos=1994;
if($str2==95)
    $setos=1995;
if($str2==96)
    $setos=1996;
if($str2==97)
    $setos=1997;
if($str2==98)
    $setos=1998;
if($str2==99)
    $setos=1999;
if($str2==00)
    $setos=2000;
if($str2==01)
    $setos=2001;
if($str2==02)
    $setos=2002;
if($str2==03)
    $setos=2003;
if($str2==04)

```

```

        $setos=2004;
if($str2==05)
        $setos=2005;
if($str2==06)
        $setos=2006;
if($str2==07)
        $setos=2007;
if($str2==08)
        $setos=2008;
if($str2==09)
        $setos=2009;

```

```

$str7=substr($new,0,-4);

```

```

$load="load data infile 'C:\\AppServ\\www\\upload\\$new' into table
$str7";

```

```

        echo "$load";
        echo "<br>";
        mysql_query("$load",$link_main);
echo mysql_errno() . ": " . mysql_error(). "\n";
        $p="update $str4 set periodos=A Περίοδος Χειμερινή
$setos,type=$tipos; where periodos is null";
        if($str3==1)
            mysql_query("update $str7 set periodos='A Περίοδος Χειμερινή
$setos',type='$tipos;' where periodos is null",$link_main);
        if($str3==2)
            mysql_query("update $str7 set periodos='B Περίοδος Χειμερινή
$setos',type='$tipos;' where periodos is null",$link_main);
        if($str3==3)
            mysql_query("update $str7 set periodos='A Περίοδος Εαρινή
$setos',type='$tipos;' where periodos is null",$link_main);
        if($str3==4)
            mysql_query("update $str7 set periodos='B Περίοδος Εαρινή
$setos',type='$tipos;' where periodos is null",$link_main);

```

```

echo $p;

```

```

//telos

```

```

//=====
=====

```

```
?>
    </body>
</html>
```

05.Κώδικας του αρχείου results.php .

```
<html>
<meta http-equiv="Content-Language" content="el">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-7">
<head><title>Ηλεκτρονική Βαθμολογία</title>
</head>
<body align=center>
<?
$passo=$_REQUEST['passo'];

if(!is_numeric($passo))
echo "<center>Το πάσο που εισάγατε δεν είναι σωστό..</center>";

if (mb_strlen($passo)!=4): //prepi na piraksis to arxio php ine gia na treksi
to mb_strlen()
    echo "<center>Το πάσο πρέπει να έχει τέσσερα ψηφία...</center>";
    exit;
endif;

if(is_numeric($passo))//ama to passo ine keno i null na min kani tipota giati
xtipane to where
{
```

```

$link_main=mysql_connect("localhost","rc","");
if(!$link_main)
    echo "lathos sti sindesi main";

mysql_select_db("tei",$link_main);
$result_main=mysql_query("select * from
lessons",$link_main);

if(!$result_main)
    echo "lathos erotima main";

echo "<table border=3 ALIGN=center width='80%'>";
echo "<tr bgcolor=Green>";
echo "<td>Όνομα
μαθήματος</td><td>Όνομα</td><td>Πάσο</td><td>Χρονιά
Εγγραφής</td><td>Εξάμηνο</td><td>Βαθμός</td><td>Περίοδος</td><
td>Εργαστήριο ή Διάλεξη</td>";

while($row=mysql_fetch_row($result_main))//gia kathe
mathima ekeli ta porakato
{
    $pinakas=$row[0];//bazi stin metabliti to onoma tou
kathe pinaka mathimatos pou iparxi stin basi
    //echo $row[1];//emfanizi ton titlo tou mathimatos prin
apo kathe pinaka

//=====
$link_tables=mysql_connect("localhost","rc","");
if(!$link_tables)
    echo "lathos sti sindesi tables";

mysql_select_db("$pinakas",$link_tables);

$result_tables=mysql_query("select * from $pinakas
where paso=$paso",$link_tables);
if($result_tables==true)
{

//echo mysql_errno() . " : " . mysql_error(). "\n";

```

```

        if(!$result_tables)
//          echo "to paso pou isagate den ine sosto";//na
fitakso na elegxo an ine null to erotima pou erxete
echo "<br>";

        $n=mysql_num_rows($result_tables);//bazi sto n ton
arithmo ton egrafon pou iparxoun sto kathe mathima
        if ($n!=0)//an o arithmos ine 0 tote den emfanizi
kenous pinakes
        {
            //echo $row[1];//emfanizi ton titlo tou
mathimatos prin apo kathe pinaka
            $math=$row[1];
            echo "</tr>";
            $i=0;
            $hide=0;

            while($row_tables=mysql_fetch_row($result_tables))
                {$i++;

                    if($i%2==0)
                        echo "<tr bgcolor=orange>";//gia na
bgeni i kathe grami me diaforetiko xroma
                    if($i%2!=0)
                        echo "<tr bgcolor=yellow>";
                        echo "<td>$math</td>";
                        foreach($row_tables as $v)
                            {
                                $hide++;//gia na arxisi apo to triplo
column tou pinaka giati kseroume to epitheto kai to paso afou to edose
                                if($hide>1)
                                    {
                                        echo "<td>$v</td>";
                                    }
                                }
                                $hide=0;
                                echo "</tr>";
                            }
                }
        }

```

```

//=====
=

```

```

    }
    }
    echo "</table>";

//if ($$result_main!=null)
mysql_close($link_tables);
mysql_close($link_main);
}

?>
</body>
</html>

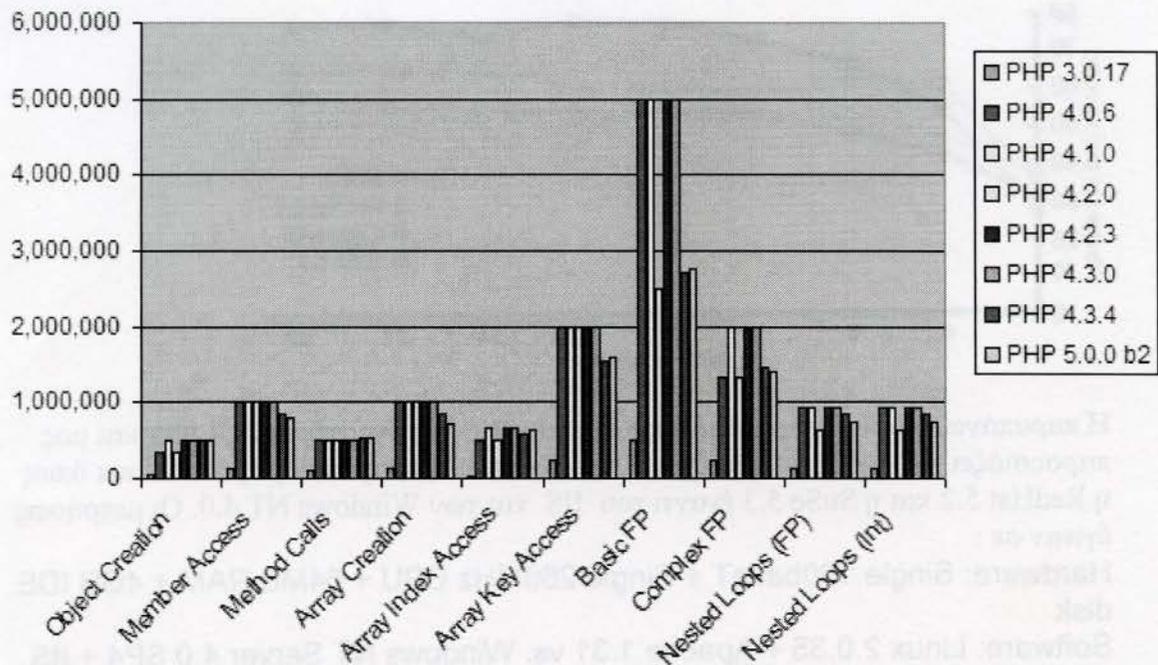
```

06.Κώδικας του αρχείου startupload.html .

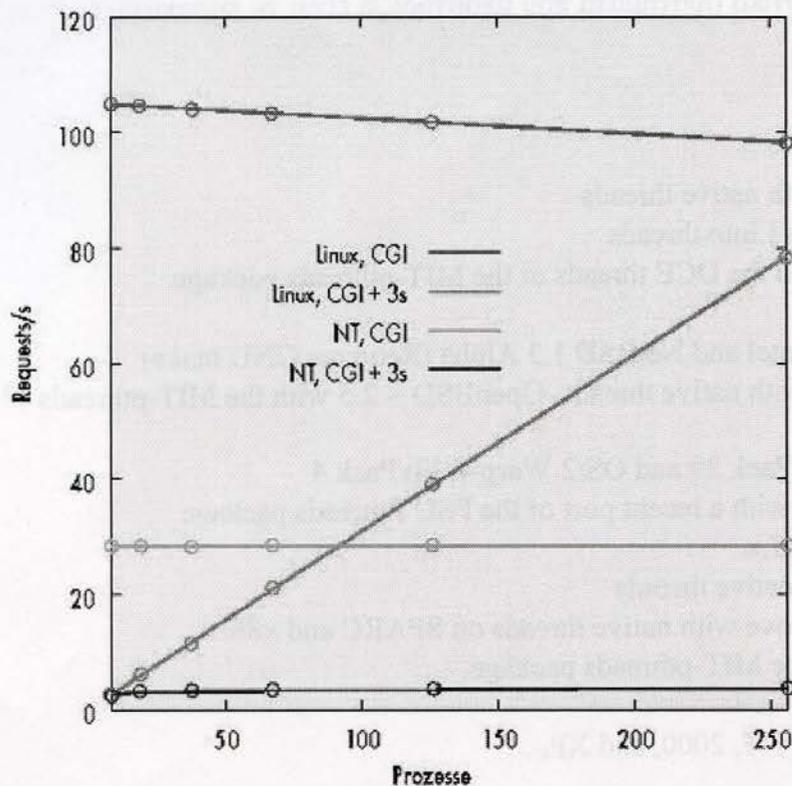
```

<html>
  <body>
    <form enctype="multipart/form-data" action="upload.php"
method=post>
      <input type="hidden" name="MAX_FILE_SIZE"
value="32000">
      upload this file: <input name="userfile" type="file">
      <br>
      Select : <SELECT name="tipos">
        <OPTION selected
value="Διάλεξη">Διάλεξη</OPTION>
        <OPTION selected
value="Εργαστήριο">Εργαστήριο</OPTION>
      </SELECT>
      <input type="submit" value="send file">
    </form>
  </body>
</html>

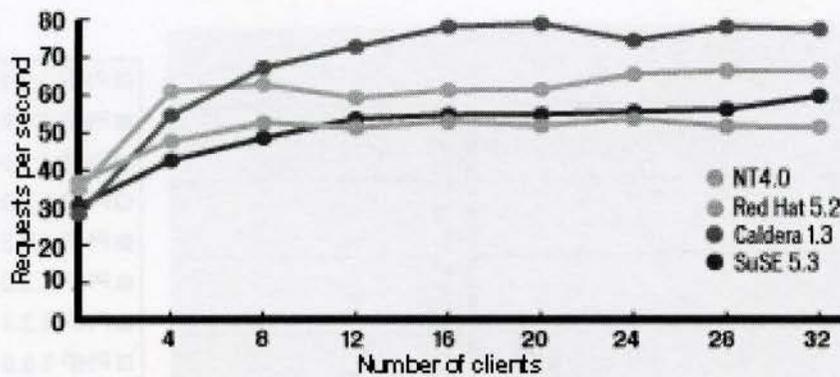
```



Η παραπάνω εικόνα κάνει συγκριτική δοκιμή σε διάφορες εκδόσεις τις PHP. Όσο μικρότερα είναι τα ραβδογράμματα τόσο πιο γρήγορα εκτελούνται οι διεργασίες.



Η παραπάνω γραφική παράσταση κάνει μια συγκριτική δοκιμή ανάμεσα σε τέσσερα συστήματα τα δύο τρέχουν Linux και τα άλλα δυο NT γίνεται φανερό ότι το Linux είναι πιο γρήγορο από τα windows NT στην επεξεργασία CGI που είναι ο πρόγονος της PHP.



Η παραπάνω γραφική παράσταση προέρχεται από τα εργαστήρια ZDLabs και μας παρουσιάζει τις επιδόσεις που έχει ο Apache σε διάφορες διανομές του Linux όπως η RedHat 5.2 και η SuSe 5.3 έναντι του IIS και των Windows NT 4.0. Οι μετρήσεις έγιναν σε :

Hardware: Single 100baseT + Single 266MHz CPU + 64MB RAM + 4GB IDE disk

Software: Linux 2.0.35 + Apache 1.31 vs. Windows NT Server 4.0 SP4 + IIS 4.0

Η Mysql είναι γραμμένη σε γλώσσα C και C++ και είναι δοκιμασμένη σε πολλά λειτουργικά και σε πολλούς compilers για αυτόν τον λόγο παρέχει μια ιδιαίτερη σταθερότητα. Τα λειτουργικά συστήματα που υποστηρίζει είναι τα παρακάτω :

- IBM-AIX
- BSD/OS
- DEC Unix 4.x with native threads
- FreeBSD 4.x with Linuxthreads
- HP-UX 10.20 with the DCE threads or the MIT-pthreads package
- Mac OS X
- NetBSD 1.3/1.4 Intel and NetBSD 1.3 Alpha (Requires GNU make)
- OpenBSD > 2.5 with native threads. OpenBSD < 2.5 with the MIT-pthreads package
- OS/2 Warp 3, FixPack 29 and OS/2 Warp 4, FixPack 4
- SCO OpenServer with a recent port of the FSU Pthreads package
- SCO UnixWare 7.1.x
- SGI Irix 6.x with native threads
- Solaris 2.5 and above with native threads on SPARC and x86
- SunOS 4.x with the MIT-pthreads package.
- Tru64 Unix
- Windows 9x, Me, NT, 2000, and XP.

Υποστηρίζει και APIs (Applications Prorammmers Interface) για τις γλώσσες C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, and Tcl.

Παρέχει την δυνατότητα για πολλαπλά νήματα που υποστηρίζονται από τον πυρήνα με αποτέλεσμα να μπορεί εύκολα να χρησιμοποιηθεί σε υπολογιστή που έχει πολλούς επεξεργαστές πάνω του.

Η MySql μπορεί να υποστηρίζει transactions ή όχι.

Χρησιμοποιεί B-tree πίνακες (MyISAM) με συμπίεση index.

Η MySQL έχει δοκιμαστεί με το πρόγραμμα Purify (ένα εμπορικό πρόγραμμα ελέγχου διαρροής μνήμης) όπως και με το πρόγραμμα Valgrind, που είναι ένα πρόγραμμα ανοιχτού κώδικα (<http://developer.kde.org/~sewardj/>).

Η MySQL είναι της μορφής client/server σαν ξεχωριστό πρόγραμμα για δικτυακό περιβάλλον. Είναι επίσης διαθέσιμη μια βιβλιοθήκη όπου μπορεί να ενσωματωθεί σε μια εφαρμογή που δεν αλληλεπιδρά με άλλες. Μια τέτοια εφαρμογή μπορεί να χρησιμοποιηθεί σε απομονωμένο περιβάλλον όπου δεν έχετε σε επαφή με δίκτυο.

Παρέχει την δυνατότητα στο χρήστη να ορίσει τον τύπο μιας στήλης σε έναν από τους παρακάτω τύπους : FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET, ENUM.

Όσον αφορά το θέμα της ασφάλειας υπάρχει ένα σύστημα με δικαιώματα και κωδικούς που είναι πολύ παρεμετροποιήσιμο και ασφαλές παρέχοντας την δυνατότητα για επαλήθευση στο χρήστη που συνδέεται με την βάση. Οι κωδικοί είναι ασφαλείς επειδή όλα τα δεδομένα που στέλνονται ή ανακτώνται κρυπτογραφούνται και έτσι δεν υπάρχει η δυνατότητα υποκλοπής.

Η MySql έχει δοκιμαστεί με ένα πολύ μεγάλο αριθμό από εγγραφές (50,000,000 σε ένα πίνακα) καθώς και με ένα μεγάλο αριθμό από πίνακες (60,000) και με 5,000,000,000 εγγραφές συνολικά σε όλους τους πίνακες. Επιτρέπεται πάνω από 32 indexes για κάθε πίνακα.

Οι Clients μπορούν να συνδέονται με τον server χρησιμοποιώντας το πρωτόκολλο TCP/IP μαζί με sockets σε οποιαδήποτε πλατφόρμα.

Χρησιμοποιώντας το Connector/ODBC (Open-DataBase-Connectivity) μπορεί να συνδεθείς με τη βάση. Έτσι για παράδειγμα μπορείς να συνδεθείς στην MySql με την MS Access. Οι Clients μπορεί να τρέχουν σε περιβάλλον Windows ή Unix. Connector/ODBC όπου είναι ενσωματωμένοι στο λειτουργικό. Με το πρωτόκολλο Connector/JDBC μπορείς να συνδεθείς με την Java στην βάση.

Ο server έχει την δυνατότητα να εμφανίσει τα μηνύματα σε διάφορες γλώσσες μέσα στις οποίες είναι και η Ελληνική. Όλες οι αναζητήσεις είναι ευαίσθητες σε κεφαλαία ή μικρά. Η ταξινόμηση γίνεται με το επιλεγμένο character set το οποίο είναι το Swedish εξορισμού και όχι το Αμερικανικό.

Η έκδοση 3.22 έχει όριο στο μέγεθος ενός πίνακα που είναι στα 4GB (4 gigabyte). Με τη MyISAM μηχανή που υποστηρίζει έχει την δυνατότητα αύξησης σε 8 εκατομμύρια terabytes (2^{63} bytes). Αλλά το όριο κατεβαίνει όχι λόγω της MySQL αλλά του λειτουργικού συστήματος ο παρακάτω πίνακας μας πληροφορεί για το μέγιστο μέγεθος που επιτρέπει το κάθε λειτουργικό σύστημα.

Operating System	File-size Limit
Linux-Intel 32-bit	2GB, much more when using LFS
Linux-Alpha	8TB (?)
Solaris 2.5.1	2GB (4GB possible with patch)
Solaris 2.6	4GB (can be changed with flag)
Solaris 2.7 Intel	4GB
Solaris 2.7 UltraSPARC	512GB

Ο Apache 2.0 έχει νέα χαρακτηριστικά όπως πολλαπλά νήματα και πολλαπλούς επεξεργαστές με αποτέλεσμα να έχουμε αυξημένες δυνατότητες. Έχει ενσωματωθεί η δυνατότητα να υποστηρίζει πολλά πρωτόκολλα. Παρέχεται καλύτερη υποστήριξη σε διάφορα λειτουργικά όπως BeOS, OS/2, and Windows.

Υποστηρίζει το IPv6 πρωτόκολλο και τα modules μπορούν να γραφούν σαν φίλτρα και να ενεργούν πάνω σε ροή των δεδομένων.

Όπως και η MySQL υποστηρίζει μηνύματα λάθους σε πολλές γλώσσες.

Μπορούμε να τον βρούμε είτε σε μορφή κώδικα είτε ήδη έτοιμο για εγκατάσταση. Για την εγκατάσταση του Apache σε μορφή κώδικα χρειαζόμαστε εκτός από ένα ANSI-C μεταγλωττιστή και 50 MB χώρο στον σκληρό δίσκο για να γίνει η μεταγλώττιση και 10 για την εγκατάσταση. Μερικά από τα script είναι γραμμένα σε perl οπότε πρέπει να την έχουμε εγκαταστήσει και αυτήν στο σύστημά μας.

Authentication :

Η αγανώριση ενός δικτύου, ενός server ή client ή user. Δηλαδή η πιστοποίηση ότι είσαι αυτός που υποστηρίζεις.

Access Control :

Η ελεγχόμενη πρόσβαση με διάφορες σελίδες ή και η απόρριψη ενός χρήστη που έχει μια συγκεκριμένη διεύθυνση ώστε να μην μπορεί να δει καθόλου τα περιεχόμενα της σελίδας.

Authorization :

Με αυτή την λειτουργία επιτρέπεται η όχι η πρόσβαση στο site, ο έλεγχος συνήθως γίνεται με ένα όνομα χρήστη και με έναν κωδικό.

OpenSSL :

Είναι ότι ένα πρωτόκολλο που επιτρέπει την ασφαλή ανταλλαγή δεδομένων. Όπως δηλώνει και το όνομά του είναι ανοιχτού κώδικα. Χρησιμοποιείται στο ηλεκτρονικό εμπόριο για την ασφαλή αποστολή προσωπικών δεδομένων όπως οι πιστωτικές κάρτες γιατί τα δεδομένα κρυπτογραφούνται.

Plaintext :

Η αποστολή μη κρυπτογραφημένων πληροφοριών μέσω του δικτύου.

Private Key :

Ένα μυστικό κλειδί που χρησιμοποιείται για να κρυπτογραφηθούν τα εισερχόμενα μηνύματα και να υπογραφούν τα εξερχόμενα μηνύματα (δηλαδή να κρυπτογραφηθούν).

Proxy :

Ένα μηχανήμα που είναι ενδιάμεσα του server και του client. Δέχεται τα αιτήματα από τους πελάτες και τα μεταδίδει στον server και μετά επιστρέφει την απάντηση που πήρε από τον server στον client. Αν πολλοί client θέλουν τα ίδια δεδομένα ο αντί να τα ζητήσει ο proxy ξανά από τον server τους τα δίνει από την cache μνήμη του που είναι είδη αποθηκευμένα και έτσι έχει λιγότερο φόρτο ο Server.

Public Key :

Το δημόσιο κλειδί χρησιμοποιείται για να κρυπτογραφούνται μηνύματα του χρήστη και να αποκρυπτογραφούνται μηνύματα που δημιουργήθηκαν από τον χρήστη.

Secure Sockets Layer (SSL) :

Είναι πρωτόκολλο που δημιουργήθηκε από την Netscape Communications για γενική πιστοποίηση και κρυπτογράφηση δεδομένων πάνω από το TCP/IP. Η πιο διαδεδομένη του χρήση είναι η χρησιμοποίηση του SSL πάνω από το HTTP (HyperText Transfer Protocol) και ο συνδυασμός του έχει αποτέλεσμα το HTTPS.

Session :

Η δέσμευση μιας ποσότητας μνήμης στον server για την αποθήκευση πληροφοριών του πελάτη. Τέτοιες πληροφορίες μπορεί να είναι το όνομά του και στην περίπτωση ενός ηλεκτρονικού καταστήματος το καλάθι αγορών του, όπως τα προϊόντα ,οι τιμές κ.τ.λ..

Η PHP, της οποίας τα αρχικά αντιπροσωπεύουν το "PHP: Hypertext Preprocessor" είναι μια ευρέως χρησιμοποιούμενη, ανοιχτού κώδικα, γενικού σκοπού scripting γλώσσα προγραμματισμού, η οποία είναι ειδικά κατάλληλη για ανάπτυξη εφαρμογών για το Web και μπορεί να ενσωματωθεί στην HTML. Η σύνταξη της παίρνει στοιχεία των C, Java, και Perl και είναι εύκολη στην μάθηση. Ο κύριος στόχος της γλώσσας είναι να επιτρέπει σε web developers να γράφουν δυναμικά παραγόμενες σελίδες (webpages) γρήγορα, αλλά κάποιος μπορεί να κάνει πολύ περισσότερα με την PHP.

Η HTTP αναγνώριση με την PHP είναι διαθέσιμη μόνο όταν εκτελείται σαν Apache module και έτσι δεν είναι διαθέσιμη στην CGI έκδοση. Σε ένα Apache module PHP script, είναι δυνατόν να χρησιμοποιηθεί η **header()** συνάρτηση για να σταλεί ένα "Authentication Required" μήνυμα στον browser του client αναγκάζοντας τον να πετάξει ένα Username/Password παράθυρο εισόδου. Όταν ο χρήστης εισάγει ένα username και ένα password, το URL που περιέχει το PHP script θα καλεστεί ξανά με τις προκαθορισμένες μεταβλητές PHP_AUTH_USER, PHP_AUTH_PW, και AUTH_TYPE ορισμένες στο username, password και τον τύπο του authentication αντίστοιχα. Αυτές οι προκαθορισμένες μεταβλητές βρίσκονται στους \$_SERVER και \$HTTP_SERVER_VARS πίνακες. Μόνο η "Basic" αναγνώριση υποστηρίζεται.

Η PHP έχει διαφανή υποστήριξη για HTTP cookies. Τα cookies είναι ένας μηχανισμός για αποθήκευση δεδομένων στον απομακρυσμένο browser και έτσι εντοπίζονται ή αναγνωρίζονται χρήστες που επιστρέφουν. Μπορείτε να ορίσετε cookies χρησιμοποιώντας την συνάρτηση **setcookie()**. Τα cookies είναι μέρος του HTTP header, έτσι η **setcookie()** πρέπει να καλεστεί πριν οποιαδήποτε έξοδος σταλεί στον browser. Αυτός είναι ο ίδιος περιορισμός που έχει η συνάρτηση **header()**. Μπορείτε να χρησιμοποιήσετε τις output buffering συναρτήσεις για να καθυστερήσετε την έξοδο του script σας μέχρι να αποφασίσετε αν θα θέσετε κάποιο cookie ή στείλετε κάποιους headers.

Οποιαδήποτε cookies στέλνται σε σας από τον client θα μετατρέπονται αυτόματα σε PHP μεταβλητές όπως τα δεδομένα των GET και POST μεθόδων, ανάλογα με τις register_globals και variables_order μεταβλητές ρυθμίσεων. Αν επιθυμείτε να καθορίσετε πολλές τιμές σε ένα μοναδικό cookie, απλά προσθέστε [] στο όνομα του cookie.

Από την PHP 4.1.0, το \$_COOKIE auto-global array θα περιέχει πάντα οποιαδήποτε cookies στέλνται από τον client. Το \$HTTP_COOKIE_VARS ορίζεται επίσης σε νεότερες εκδόσεις της PHP όταν η επιλογή ρύθμισης track_vars ορίζεται.

Η PHP είναι ικανή για να λαμβάνει upload αρχείων από οποιοδήποτε RFC-1867 συμβατό browser (αυτό συμπεριλαμβάνει τους Netscape Navigator 3 ή μεγαλύτερο, Microsoft Internet Explorer 3 με ένα patch από τη Microsoft, ή μεγαλύτερο χωρίς κάποιο patch). Αυτό το χαρακτηριστικό επιτρέπει στους ανθρώπους να κάνουν upload τόσο κειμένου, όσο και binary αρχεία. Με τις συναρτήσεις αναγνώρισης και χειρισμού αρχείων της PHP, έχετε πλήρη έλεγχο στο ποιός επιτρέπεται να κάνει upload και τι θα γίνει με το αρχείο από τη στιγμή που έχει γίνει upload και μετά.

Όσο το `allow_url_fopen` είναι ενεργοποιημένο στο `php.ini`, μπορείτε να χρησιμοποιείτε τα HTTP και FTP URL με τις περισσότερες συναρτήσεις που παίρνουν ένα όνομα αρχείου σαν παράμετρο. Επιπλέον, τα URL μπορούν να χρησιμοποιηθούν με τις **include()**, **include once()**, **require()** και **require once()** δηλώσεις.

Σημείωση: Τα ακόλουθα ισχύουν από την 3.0.7 και μετά.

Εσωτερικά στην PHP ένα status σύνδεσης (connection status) διατηρείται. Υπάρχουν 3 δυνατές καταστάσεις:

- 0 - NORMAL
- 1 - ABORTED
- 2 - TIMEOUT

Όταν ένα PHP script εκτελείται, κανονικά η κατάσταση NORMAL είναι ενεργή. Αν ένας remote (απομακρυσμένος) client αποσυνδεθεί, η ABORTED σημαία κατάστασης ενεργοποιείται. Μια αποσύνδεση ενός remote client συνήθως προκαλείται όταν ο χρήστης πατήσει το κουμπί STOP. Αν το όριο χρόνου της PHP (δείτε την **set time limit()**) περαστεί, η TIMEOUT σημαία κατάστασης ενεργοποιείται.

Μπορείτε να αποφασίσετε αν θέλετε μια αποσύνδεση ενός client να προκαλεί το script σας να σταματά. Κάποτε είναι βολικό να έχετε πάντα τα scripts σας να τρέχουν μέχρι τέλους ακόμη και αν δεν υπάρχει κάποιος remote browser να λαμβάνει την έξοδο. Η προκαθορισμένη συμπεριφορά όμως είναι το script σας να τερματίζει όταν ο remote client αποσυνδέεται. Αυτή η συμπεριφορά μπορεί να οριστεί μέσω του `ignore_user_abort` directive στο `php.ini` όπως και μέσω του αντίστοιχου "`php_value ignore_user_abort`" Apache `.conf` directive ή με την **ignore user abort()** συνάρτηση.

Αν δεν πείτε στην PHP να αγνοήσει μια αποσύνδεση ενός χρήστη και ο χρήστης αποσυνδεθεί, το script σας θα τερματίσει. Η μόνη εξαίρεση είναι αν έχετε κάνει `register_shutdown_function` χρησιμοποιώντας την **register shutdown function()**. Με μια `shutdown` συνάρτηση, όταν ο remote χρήστης πατήσει το STOP κουμπί, την επόμενη φορά που το script θα προσπαθήσει να έχει κάτι στην έξοδο, η PHP θα ανιχνεύσει πως η σύνδεση έχει τερματιστεί και η `shutdown` συνάρτηση θα καλεστεί. Αυτή η `shutdown` συνάρτηση θα καλεστεί στο τέλος της κανονικής εκτέλεσης του script σας, έτσι για να κάνετε κάτι διαφορετικό στην περίπτωση που ένας client αποσυνδέεται μπορείτε να χρησιμοποιήσετε την **connection aborted()** συνάρτηση. Αυτή η συνάρτηση θα επιστρέψει **TRUE** αν η σύνδεση έχει τερματιστεί (`aborted`).

Το script σας μπορεί επίσης να τερματιστεί από τον ενσωματωμένο timer (χρονομετρητή) των script. Ο προκαθορισμένος timeout χρόνος είναι 30 δευτερόλεπτα. Μπορεί να αλλαχτεί χρησιμοποιώντας το `max_execution_time` directive στο `php.ini` ή το αντίστοιχο "`php_value max_execution_time`" directive στα Apache `.conf` όπως επίσης και με την συνάρτηση **set time limit()**.

Όταν ο timer λήξει, το script θα τερματιστεί και, όπως και στην παραπάνω περίπτωση αποσύνδεσης του client, αν μια shutdown συνάρτηση έχει καταχωρηθεί (registered) θα καλεστεί. Μέσα σε αυτή την shutdown συνάρτηση μπορείτε να ελέγξετε αν ένα timeout ήταν το αίτιο της κλήσης της shutdown συνάρτησης, καλώντας την συνάρτηση **connection timeout()**. Αυτή η συνάρτηση επιστρέφει **TRUE** αν ένα timeout δημιούργησε την κλήση της shutdown συνάρτησης.

Κάτι που αξίζει να σημειωθεί είναι ότι τόσο η ABORTED όσο και η TIMEOUT κατάσταση μπορούν να είναι ενεργές την ίδια στιγμή. Αυτό είναι πιθανόν αν πείτε στην PHP να αγνοεί τα abort των χρηστών. Η PHP ακόμη θα σημειώνει το γεγονός ότι κάποιος χρήστης μπορεί να έχει διακόψει τη σύνδεση, αλλά το script θα συνεχίσει να τρέχει. Αν τότε φτάσει το όριο χρόνου θα τερματιστεί και η shutdown συνάρτησή σας, αν υπάρχει, θα καλεστεί.

Σε αυτό το σημείο θα βρείτε ότι η **connection timeout()** και η **connection aborted()** επιστρέφουν **TRUE**. Μπορείτε επίσης να ελέγξετε και τις δύο καταστάσεις σε μία κλήση χρησιμοποιώντας την **connection status()**. Αυτή η συνάρτηση επιστρέφει ένα πεδίο με bits (bitfield) των ενεργών καταστάσεων. Έτσι, αν και οι δύο καταστάσεις είναι ενεργές για παράδειγμα, θα επιστρέψει 3.

Από την έκδοση 4.3.0, η PHP υποστηρίζει ένα νέο τύπο SAPI (Server Application Programming Interface) με όνομα CLI το οποίο σημαίνει *Command Line Interface*. Όπως υποδηλώνει το όνομα, το επίκεντρο αυτού του τύπου SAPI είναι η ανάπτυξη εφαρμογών για το shell (ή επίσης για desktop) με την PHP. Υπάρχουν αρκετές διαφορές μεταξύ του CLI SAPI και άλλων SAPI τα οποία εξηγούνται σε αυτό το κεφάλαιο. Αξίζει να σημειωθεί πως το CLI και το CGI είναι διαφορετικά SAPI αν και μοιράζονται πολλές ίδιες συμπεριφορές.

Το CLI SAPI έχει κυκλοφορήσει για πρώτη φορά με την PHP 4.2.0, αλλά ήταν ακόμη πειραματικό και έπρεπε να ενεργοποιηθεί ρητά με το `--enable-cli` όταν έτρεχε το `./configure`. Από την PHP 4.3.0 το CLI SAPI δεν είναι πλέον πειραματικό και η επιλογή `--enable-cli` είναι προεπιλεγμένα ενεργοποιημένη. Μπορείτε να χρησιμοποιήσετε το `--disable-cli` για να την απενεργοποιήσετε.

Από την PHP 4.3.0, το όνομα, η τοποθεσία και η ύπαρξη των CLI/CGI binaries θα διαφέρει ανάλογα με τον τρόπο που εγκαταστάθηκε η PHP στο σύστημά σας. Ως προεπιλογή, όταν εκτελείται το `make`, τόσο το CGI όσο και το CLI γίνονται `build` και τοποθετούνται ως `sapi/cgi/php` και `sapi/cli/php` αντίστοιχα, στον κατάλογο του `source` της `php`. Θα προσέξετε πως και τα δύο έχουν όνομα `php`. Αυτό που συμβαίνει κατά τη διάρκεια του `make install` εξαρτάται από τη γραμμή του `configure` σας. Αν ένα module SAPI επιλεγεί στο `configure`, όπως το `apxs`, ή η επιλογή `--disable-cgi` χρησιμοποιηθεί, το CLI αντιγράφεται στο `{PREFIX}/bin/php` στη διάρκεια του `make install` αλλιώς το CGI τοποθετείται εκεί. Έτσι, για παράδειγμα, αν το `--with-apxs` βρίσκεται στην `configure` γραμμή σας, το CLI αντιγράφεται στο `{PREFIX}/bin/php` κατά τη διάρκεια του `make install`.

Αν θέλετε να παρακάμψετε την εγκατάσταση του CGI binary, χρησιμοποιήστε το `make install-cli` μετά το `make install`. Εναλλακτικά μπορείτε να ορίσετε το `--disable-cgi` στην `configure` γραμμή σας.

Σημείωση: Επειδή τόσο το `--enable-cli` όσο και το `--enable-cgi` είναι ενεργοποιημένα by default (ως προεπιλογή), απλά έχοντας το `--enable-cli` στην `configure` γραμμή σας δεν σημαίνει απαραίτητα πως το CLI θα αντιγραφεί ως `{PREFIX}/bin/php` κατά το `make install`.

Τα windows πακέτα μεταξύ της PHP 4.2.0 και της PHP 4.2.3 έδιναν το CLI σαν `php-cli.exe`, έχοντας το στον ίδιο κατάλογο με το CGI `php.exe`. Από την PHP 4.3.0 το windows πακέτο δίνει το CLI σαν `php.exe` σε ένα ξεχωριστό κατάλογο με όνομα `cli`, έτσι `cli/php.exe`.

Τι SAPI έχω;: Από ένα shell, η εντολή `php -v` θα σας πει αν η `php` είναι CGI ή CLI. Δείτε επίσης τη συνάρτηση `php_sapi_name()` και τη σταθερά `PHP_SAPI`.

Σημείωση: Μια manual σελίδα του unix έχει προστεθεί στην PHP 4.3.2. Μπορείτε να την δείτε γράφοντας `man php` στο shell περιβάλλον σας.

Οι κύριες διαφορές του CLI SAPI συγκριτικά με το SAPI είναι:

- Αντίθετα με το CGI SAPI, δεν εκτυπώνονται headers στην έξοδο.

Αν και το CGI SAPI προσφέρει ένα τρόπο για να μην φανούν HTTP headers, δεν υπάρχει παρόμοιος διακόπτης για να τους ενεργοποιήσει κανείς στο CLI SAPI.

Το CLI ξεκινά σε quiet mode ως προεπιλογή, αν και ο `-q` διακόπτης κρατιέται για συμβατότητα για να μπορείτε να το χρησιμοποιείτε σε παλαιότερα CGI scripts.

Δεν αλλάζει τον working κατάλογο σε αυτόν του script. (Ο διακόπτης `-C` υπάρχει για συμβατότητα)

Απλά μηνύματα λάθους σε μορφή κειμένου (δεν υπάρχει HTML μορφοποίηση).

- Υπάρχουν συγκεκριμένα `php.ini` directives τα οποία γίνονται override από το CLI SAPI επειδή δεν έχουν νόημα σε shell περιβάλλοντα:

Πίνακας 23-1. php.ini directives που γίνονται override

Directive	Προεπιλεγμένη τιμή του CLI SAPI	Σχόλια
<u>html_errors</u>	FALSE	Μπορεί να είναι αρκετά δύσκολο να διαβαστούν τα μηνύματα σφαλμάτων στο shell όταν είναι γεμάτα με όλα εκείνα τα άσκοπα HTML tags, έτσι αυτό το έχει ως προεπιλογή το FALSE .
<u>implicit_flush</u>	TRUE	Είναι θεμιτό οποιαδήποτε έξοδος προερχόμενη από τις print() , echo() και τους φίλους τους να γράφεται αμέσως στο output και να μην γίνονται cache σε κάποιο buffer. Μπορείτε ακόμη να χρησιμοποιήσετε <u>output buffering</u> αν θέλετε να αναβάλλετε ή να χειριστείτε το standard output.
<u>max execution time</u>	0 (unlimited)	Λόγω των άπειρων δυνατοτήτων στη χρήση της PHP σε περιβάλλοντα shell, ο μέγιστος χρόνος εκτέλεσης έχει οριστεί να είναι απεριόριστος. Εκεί που οι εφαρμογές γραμμένες για το web συχνά εκτελούνται πολύ γρήγορα, οι shell εφαρμογές τείνουν να έχουν πολύ μεγαλύτερο χρόνο εκτέλεσης.
<u>register_argc_argv</u>	TRUE	Επειδή αυτή η ρύθμιση είναι TRUE πάντα θα έχετε πρόσβαση στο <i>argc</i> (ο αριθμός των arguments που περνιούνται στην εφαρμογή) και το <i>argv</i> (array των ίδιων των argument) στο CLI SAPI. Από την PHP 4.3.0, οι PHP μεταβλητές \$argc και \$argv καταχωρούνται και τιμολογούνται με τις ανάλογες τιμές όταν χρησιμοποιείται το CLI SAPI. Πριν από αυτή την έκδοση, η δημιουργία

Directive	Προεπιλεγμένη τιμή του CLI SAPI	Σχόλια
		αυτών των μεταβλητών συμπεριφερόταν όπως γίνεται στις CGI και MODULE εκδόσεις που απαιτούν το PHP directive <code>register_globals</code> να είναι ενεργοποιημένο. Ανεξάρτητα της έκδοσης ή της <code>register_globals</code> ρύθμισης, μπορείτε πάντα να πάτε μέσω του <code>\$_SERVER</code> ή του <code>\$HTTP_SERVER_VARS</code> . Για παράδειγμα: <code>\$_SERVER['argv']</code>

Σημείωση: Αυτά τα directives δεν μπορούν να αρχικοποιηθούν με κάποια άλλη τιμή από το αρχείο ρυθμίσεων `php.ini` ή ένα προσωπικό (αν οριστεί). Αυτός είναι ένας περιορισμός επειδή αυτές οι τιμές εφαρμόζονται μετά που αναλύονται όλα τα αρχεία ρυθμίσεων. Ωστόσο, οι τιμές τους μπορούν να αλλάξουν κατά το runtime (κάτι το οποίο δεν έχει νόημα για όλα εκείνα τα directives, π.χ. `register_argc_argv`).

- Για να γίνει πιο εύκολη η εργασία στο περιβάλλον του shell, οι παρακάτω σταθερές ορίζονται:

Πίνακας 23-2. Σταθερές ειδικές για το CLI

Σταθερά	Περιγραφή
STDIN	Ένα ήδη ανοιχτό stream στο stdin. Αυτό αποθηκεύεται ανοίγοντας το με την <pre>\$stdin = fopen('php://stdin', 'r');</pre>
STDOUT	Ένα ήδη ανοιχτό stream στο stdout. Αυτό αποθηκεύεται ανοίγοντας το με την <pre>\$stdout = fopen('php://stdout', 'w');</pre>
STDERR	Ένα ήδη ανοιχτό stream στο stderr. Αυτό αποθηκεύεται ανοίγοντας το με την <pre>\$stderr = fopen('php://stderr', 'w');</pre>

Δεδομένων των παραπάνω, αν δεν χρειάζεστε να ανοίξετε π.χ. ένα stream για το stderr οι ίδιοι, απλά χρησιμοποιήστε τη σταθερά αντί του ίδιου του stream resource:

```
php -r 'fwrite(STDERR, "stderr\n");'
```

Δεν χρειάζεται να κλείσετε ρητά αυτά τα stream, μια και κλείνονται αυτόματα από την PHP όταν το script τερματίζει.

- Το CLI SAPI **δεν** αλλάζει τον τρέχων κατάλογο στον κατάλογο του script που εκτελείται!

Παράδειγμα που δείχνει τις διαφορές στο CGI SAPI:

```
<?php
/* Our simple test application named test.php*/
echo getcwd(), "\n";
?>
```

Όταν χρησιμοποιείτε την CGI έκδοση, η έξοδος είναι:

```
$ pwd
/tmp

$ php -q another_directory/test.php
/tmp/another_directory
```

Αυτό δείχνει καθαρά ότι η PHP αλλάζει τον τρέχοντα κατάλογο σε αυτό του script που εκτελείται.

Χρησιμοποιώντας το CLI SAPI γίνεται:

```
$ pwd
/tmp

$ php -f another_directory/test.php
/tmp
```

Αυτό επιτρέπει μεγαλύτερη ευελιξία όταν γράφετε shell εργαλεία με την PHP.

Σημείωση: Το CGI SAPI υποστηρίζει την CLI SAPI συμπεριφορά μέσω του -C switch όταν εκτελείται από την γραμμή εντολών.

Η λίστα των επιλογών του command line options που δίνεται με το PHP binary μπορούν να ερωτηθούν ανά πάσα στιγμή τρέχοντας την PHP με το -h switch:

```
Usage: php [options] [-f] <file> [args...]
php [options] -r <code> [args...]
php [options] [-- args...]
-s      Display colour syntax highlighted source.
-w      Display source with stripped comments and whitespace.
-f <file> Parse <file>.
-v      Version number
```

```

-c <path>|<file> Look for php.ini file in this directory
-a          Run interactively
-d foo[=bar] Define INI entry foo with value 'bar'
-e          Generate extended information for debugger/profiler
-z <file>   Load Zend extension <file>.
-l          Syntax check only (lint)
-m          Show compiled in modules
-i          PHP information
-r <code>   Run PHP <code> without using script tags <?..?>
-h          This help

args...    Arguments passed to script. Use -- args when first argument
           starts with - or script is read from stdin

```

Το CLI SAPI έχει τρεις διαφορετικούς τρόπους για να πάρει τον PHP κώδικα που θέλετε να εκτελέσετε:

1. Λέγοντας της PHP να εκτελέσει ένα συγκεκριμένο αρχείο.

```
php my_script.php
```

```
php -f my_script.php
```

2. Και οι δύο τρόποι (χρησιμοποιώντας το -f switch ή όχι) εκτελούν το αρχείο my_script.php. Μπορείτε να διαλέξετε οποιοδήποτε αρχείο για να εκτελεστεί - τα PHP script σας δεν χρειάζεται να τελειώνουν με την .php επέκταση αλλά μπορείτε να έχετε οποιοδήποτε όνομα ή επέκταση θέλετε.
3. Δίνοντας τον PHP κώδικα προς εκτέλεση κατ' ευθείαν στην γραμμή εντολών.

```
php -r 'print_r(get_defined_constants());'
```

4. Προσοχή πρέπει να δίνεται σχετικά με την αντικατάσταση των μεταβλητών του shell και τη χρήση εισαγωγικών (quoting).
5. **Σημείωση:** Διαβάστε το παράδειγμα προσεκτικά, δεν υπάρχουν tags αρχής ή τέλους! Το -r switch απλά δεν τις χρειάζεται. Χρησιμοποιώντας τις θα οδηγήσει σε ένα σφάλμα του μεταφραστή.
6. Δίνοντας τον PHP κώδικα προς εκτέλεση μέσω του standard input (stdin).

Αυτό δίνει την ισχυρή δυνατότητα δυναμικής δημιουργίας PHP κώδικα και τροφοδοσίας του στο binary, όπως φαίνεται σε αυτό το (φανταστικό) παράδειγμα:

```
$ some_application | some_filter | php | sort -u >final_output.txt
```

Δεν μπορείτε να συνδιάσετε οποιοσδήποτε από τους τρεις κώδικες για να εκτελέσετε κώδικα.

Όπως οποιαδήποτε shell εφαρμογή, το PHP binary δέχεται ένα αριθμό από ορίσματα (arguments) αλλά το PHP script μπορεί επίσης να δεχτεί ορίσματα. Ο

αριθμός των ορισμάτων που μπορούν να περαστούν στο script σας δεν περιορίζεται από την PHP (το shell έχει ένα ορισμένο όριο μεγέθους του αριθμού των χαρακτήρων που μπορούν να περαστούν, συνήθως δεν θα φτάσετε αυτό το όριο). Τα arguments που περνιούνται στο script σας είναι διαθέσιμα μέσω του global array \$argv. Ο μηδενικός δείκτης (index) πάντα περιέχει το όνομα του script (το οποίο είναι - σε περίπτωση που ο PHP κώδικας έρχεται από είτε το standard input ή από το switch της γραμμής εντολών -r). Η δεύτερη registered global μεταβλητή είναι η \$argc η οποία περιέχει τον αριθμό των στοιχείων στο \$argv array (**όχι** τον αριθμό των arguments που δίνονται στο script).

Όσο τα arguments που θέλετε να περάσετε στο script σας δεν αρχίζουν με τον χαρακτήρα -, δεν υπάρχει τίποτα ιδιαίτερο που πρέπει να προσέξετε. Περνώντας ένα argument στο script σας το οποίο αρχίζει με - θα δημιουργήσει προβλήματα επειδή η ίδια η PHP νομίζει πως πρέπει να το χειριστεί. Για να το αποφύγετε αυτό, χρησιμοποιήστε το argument list separator (διαχωριστής) --. Μετά που αναλύεται αυτός ο διαχωριστής στην PHP, κάθε argument που ακολουθεί περνιέται ανέπαφος στο script σας.

```
# This will not execute the given code but will show the PHP usage
$ php -r 'var_dump($argv);' -h
Usage: php [options] [-f] <file> [args...]
[...]
```

```
# This will pass the '-h' argument to your script and prevent PHP from showing it's
usage
$ php -r 'var_dump($argv);' -- -h
array(2) {
  [0]=>
  string(1) "-"
  [1]=>
  string(2) "-h"
}
```

Ωστόσο, υπάρχει ακόμη ένας τρόπος χρήσης της PHP για shell scripting. Μπορείτε να γράψετε ένα script όπου η πρώτη γραμμή αρχίζει με #!/usr/bin/php. Ακολουθώντας αυτό μπορείτε να τοποθετήσετε κανονικό PHP κώδικα μεταξύ των PHP tags αρχής και τέλους. Από τη στιγμή που έχετε ορίσει τα attribute εκτέλεσης του αρχείου κατάλληλα (π.χ. chmod +x test) το script σας μπορεί να εκτελεστεί όπως ένα κανονικό shell ή perl script:

```
#!/usr/bin/php
<?php
  var_dump($argv);
?>
```

Υποθέτοντας ότι αυτό το αρχείο έχει όνομα test στον παρόντα κατάλογο, μπορείτε τώρα να κάνετε το ακόλουθο:

```
$ chmod 755 test
$ ./test -h -- foo
array(4) {
  [0]=>
  string(6) "./test"
  [1]=>
```

```
string(2) "-h"
[2]=>
string(2) "--"
[3]=>
string(3) "foo"
}
```

Όπως βλέπετε, σε αυτή την περίπτωση δεν χρειάζεται να δοθεί σημασία όταν περνιούνται παράμετροι που αρχίζουν με το - στο script σας.

Πίνακας 23-3. Επιλογές γραμμής εντολών

Επιλογή	Περιγραφή
-s	<p>Εμφανίζει κώδικα με χρωματικό syntax highlighting.</p> <p>Αυτή η επιλογή χρησιμοποιεί τον εσωτερικό μηχανισμό ανάλυσης του αρχείου και παράγει μια HTML highlighted έκδοση του και το γράφει στο standard output. Σημειώστε πως το μόνο που κάνει είναι να παράγει ένα μπλοκ από <code><code> [...] </code></code> HTML tags, όχι HTML headers.</p> <p>Σημείωση: Αυτή η επιλογή δεν λειτουργεί μαζί με το -r option.</p>
-w	<p>Εμφανίζει κώδικα με αφαιρεμένα τα σχόλια και τους κενούς χαρακτήρες.</p> <p>Σημείωση: Αυτή η επιλογή δεν δουλεύει μαζί με την επιλογή-r.</p>
-f	<p>Μεταφράζει και εκτελεί το δοσμένο αρχείο στην -f επιλογή. Αυτό το switch είναι προαιρετικό και μπορεί να παραληφθεί. Απλά παρέχοντας το όνομα του αρχείου προς εκτέλεση είναι αρκετό.</p>
-v	<p>Γράφει τις εκδόσεις των PHP, PHP SAPI, και Zend στο standard output, π.χ.</p> <pre>\$ php -v PHP 4.3.0 (cli), Copyright (c) 1997-2002 The PHP Group Zend Engine v1.3.0, Copyright (c) 1998-2002 Zend Technologies</pre>
-c	<p>Με αυτή την επιλογή κάποιος μπορεί είτε να ορίσει ένα κατάλογο όπου θα αναζητηθεί το php.ini είτε να ορίσει ένα προσωπικό αρχείο INI απ' ευθείας (το οποίο δεν χρειάζεται να ονομάζεται php.ini), π.χ.:</p> <pre>\$ php -c /custom/directory/ my_script.php \$ php -c /custom/directory/custom-file.ini my_script.php</pre>
-a	<p>Τρέχει την PHP interactively.</p>
-d	<p>Αυτή η επιλογή σας επιτρέπει να ορίσετε μια προσωπική τιμή για οποιοδήποτε από τα directive ρυθμίσεων που επιτρέπονται στο php.ini. Η σύνταξη είναι:</p> <pre>-d configuration_directive[=value]</pre> <p>Παραδείγματα:</p> <pre># Omitting the value part will set the given configuration directive to "1"</pre>

Επιλογή	Περιγραφή
	<pre>\$ php -d max_execution_time -r '\$foo = ini_get("max_execution_time"); var_dump(\$foo);' string(1) "1"</pre> <pre># Passing an empty value part will set the configuration directive to "" php -d max_execution_time= -r '\$foo = ini_get("max_execution_time"); var_dump(\$foo);' string(0) ""</pre> <pre># The configuration directive will be set to anything passed after the '=' character \$ php -d max_execution_time=20 -r '\$foo = ini_get("max_execution_time"); var_dump(\$foo);' string(2) "20"</pre> <pre>\$ php -d max_execution_time=doesntmakesense -r '\$foo = ini_get("max_execution_time"); var_dump(\$foo);' string(15) "doesntmakesense"</pre>
-e	Παράγει εκτενείς πληροφορίες για τον debugger/profiler.
-z	Φορτώνει την Zend επέκταση. Αν μόνο ένα όνομα αρχείο δοθεί, η PHP προσπαθεί να φορτώσει την επέκταση από το παρών path βιβλιοθηκών στο σύστημα σας (συνήθως ορίζεται το /etc/ld.so.conf στα Linux συστήματα). Περνώντας ένα όνομα αρχείο με ένα absolute (απόλυτο) path δεν θα χρησιμοποιηθεί το path αναζήτησης βιβλιοθηκών του συστήματος. Ένα relative (σχετικό) όνομα αρχείου με κατάλογο θα κάνει την PHP να προσπαθήσει να φορτώσει την επέκταση στον κατάλογο σχετικά (relative) στον τρέχων κατάλογο.
-l	Αυτή η επιλογή προσφέρει ένα βολικό τρόπο να γίνεται απλά ένας έλεγχος σύνταξης στον δοσμένο PHP κώδικα. Σε επιτυχία, το κείμενο No syntax errors detected in <filename> γράφεται στο standard output και το shell return code είναι 0. Σε αποτυχία, το κείμενο Errors parsing <filename> επιπρόσθετα στο κείμενο σφαλμάτων του εσωτερικού μεταφραστή γράφεται στο standard output και το shell return code ορίζεται σε 255. Αυτή η επιλογή δεν θα βρει fatal σφάλματα (όπως undefined συναρτήσεις). Χρησιμοποιήστε το -f αν θέλετε να ελέγξετε και για fatal σφάλματα. Σημείωση: Αυτή η επιλογή δεν δουλεύει μαζί με την -r επιλογή.
-m	Χρησιμοποιώντας αυτή την επιλογή, η PHP εκτυπώνει τα built-in (και φορτωμένα) PHP και Zend modules: <pre>\$ php -m [PHP Modules] xml tokenizer standard session posix</pre>

Επιλογή	Περιγραφή
	<pre>pcre overload mysql mbstring ctype</pre> <p>[Zend Modules]</p>
-i	<p>Αυτή η command line επιλογή καλεί την phpinfo(), και εκτυπώνει τα αποτελέσματα. Αν η PHP δεν δουλεύει σωστά, συμβουλευέστε να χρησιμοποιήσετε το <code>php -i</code> και να δείτε αν οποιαδήποτε μηνύματα σφαλμάτων εκτυπώνονται πριν ή στη θέση των πινάκων πληροφοριών. Προσέξτε πως η έξοδος είναι σε HTML και έτσι αρκετά μεγάλη.</p>
-r	<p>Αυτή η επιλογή επιτρέπει την εκτέλεση της PHP ακριβώς μέσω της γραμμής εντολών. Τα PHP tags αρχής και τέλους (<code><?php</code> και <code>?></code>) δεν χρειάζονται και θα προκαλέσουν ένα σφάλμα μεταφραστή αν υπάρχουν.</p> <p>Σημείωση: Πρέπει να δίνεται προσοχή στη χρήση αυτής της μορφής της PHP ώστε να μην υπάρχει σύγκρουση με την αντικατάσταση μεταβλητών της γραμμής εντολών που γίνεται από το shell.</p> <p>Παράδειγμα που δείχνει ένα σφάλμα του μεταφραστή (parser error):</p> <pre>\$ php -r "\$foo = get_defined_constants();" Command line code(1) : Parse error - parse error, unexpected '='</pre> <p>Το πρόβλημα εδώ είναι ότι το sh/bash εκτελεί αντικατάσταση μεταβλητών ακόμη και όταν χρησιμοποιούνται εισαγωγικά (double quotes) ". Μια και η μεταβλητή \$foo είναι απίθανο να έχει οριστεί, δεν αντιστοιχεί σε τίποτα (expands to nothing) το οποίο έχει ως αποτέλεσμα ο κώδικας που δίνεται στην PHP για εκτέλεση να είναι στην πραγματικότητα:</p> <pre>\$ php -r "= get_defined_constants();"</pre> <p>Ο σωστός τρόπος θα ήταν να χρησιμοποιηθούν μονά εισαγωγικά (single quotes) '. Οι μεταβλητές σε string που είναι single-quoted δεν αντικαθιστώνται από το sh/bash.</p> <pre>\$ php -r '\$foo = get_defined_constants(); var_dump(\$foo);' array(370) { ["E_ERROR"]=> int(1) ["E_WARNING"]=></pre>

Επιλογή	Περιγραφή
	<pre data-bbox="450 241 780 510">int(2) ["E_PARSE"]=> int(4) ["E_NOTICE"]=> int(8) ["E_CORE_ERROR"]=> [...]</pre> <p data-bbox="420 562 1203 815">Αν χρησιμοποιείτε ένα shell διαφορετικό από το sh/bash, μπορεί να βιώσετε και άλλα προβλήματα. Νιώστε ελεύθεροι να ανοίξετε ένα bug report ή να στείλετε ένα mail στο phpdoc@lists.php.net. Υπάρχει περίπτωση κάποιος εύκολα να συναντήσει προβλήματα όταν προσπαθεί να βάλει τις shell μεταβλητές μέσα στον κώδικα ή να χρησιμοποιήσει backslash για escaping. Έχετε προειδοποιηθεί.</p> <p data-bbox="420 857 1203 927">Σημείωση: Το -r είναι διαθέσιμο στο <i>CLI SAPI</i> και όχι στο <i>CGI SAPI</i>.</p>
-h	<p data-bbox="326 943 1282 1048">Με αυτή την επιλογή, μπορείτε να πάρετε πληροφορίες σχετικά με την ακριβή λίστα των επιλογών της γραμμής εντολών και κάποιες περιγραφές της μιας γραμμής για το τι κάνουν.</p>

Το εκτελέσιμο της PHP μπορεί να χρησιμοποιηθεί για να τρέξουν script της PHP απόλυτα ανεξάρτητα από τον web server. Αν είστε σε ένα Unix σύστημα, θα πρέπει να προσθέσετε μια ειδική πρώτη γραμμή στο PHP script σας, και να το κάνετε εκτελέσιμο, ώστε το σύστημα να ξέρει, ποιό πρόγραμμα θα πρέπει να τρέξει το script. Σε ένα Windows σύστημα μπορείτε να συσχετίσετε το php.exe με την double click επιλογή των .php αρχείων, ή να κάνετε ένα batch αρχείο να τρέχει το script μέσω της PHP. Η πρώτη γραμμή που προστίθεται στο script για να δουλεύει στο Unix δεν θα βλάψει στα Windows, έτσι μπορείτε να γράψετε cross platform προγράμματα με αυτό τον τρόπο. Ένα απλό παράδειγμα γραφής ενός προγράμματος PHP της γραμμής εντολών μπορεί να βρεθεί παρακάτω.

Παράδειγμα 23-1. Ένα Script που προορίζεται να τρέξει από την γραμμή εντολών (script.php)

```
#!/usr/bin/php
<?php

if ($argc != 2 || in_array($argv[1], array('--help', '-help', '-h', '-?'))) {
?>

This is a command line PHP script with one option.

Usage:
<?php echo $argv[0]; ?> <option>
```

<option> can be some word you would like to print out. With the --help, -help, -h, or -? options, you can get this help.

```
<?php
} else {
    echo $argv[1];
}
?>
```

Στο παραπάνω script, χρησιμοποιήσαμε την ειδική πρώτη γραμμή για να δείξουμε πως αυτό το αρχείο θα πρέπει να εκτελεστεί από την PHP. Δουλεύουμε με μια CLI έκδοση εδώ, έτσι δεν θα υπάρχουν εκτυπώσεις από HTTP headers. Υπάρχουν δύο μεταβλητές που μπορείτε να χρησιμοποιήσετε όταν γράφετε εφαρμογές γραμμής εντολών με την PHP: Οι \$argc και \$argv. Η πρώτη αποτελεί τον αριθμό των argument συν ένα (το όνομα του script που εκτελείται). Η δεύτερη είναι ένα array το οποίο περιέχει τα arguments, αρχίζοντας με το όνομα του script σαν τον αριθμό μηδέν (\$argv[0]).

Στο παραπάνω πρόγραμμα ελέγξαμε αν υπάρχουν λιγότερα ή περισσότερα από ένα arguments. Επίσης αν το argument ήταν --help, -help, -h ή -?, εκτυπώσαμε ένα μήνυμα βοήθειας, εκτυπώνοντας το όνομα του script δυναμικά. Αν λαμβάναμε κάποια άλλα argument τα εκτυπώναμε και αυτά.

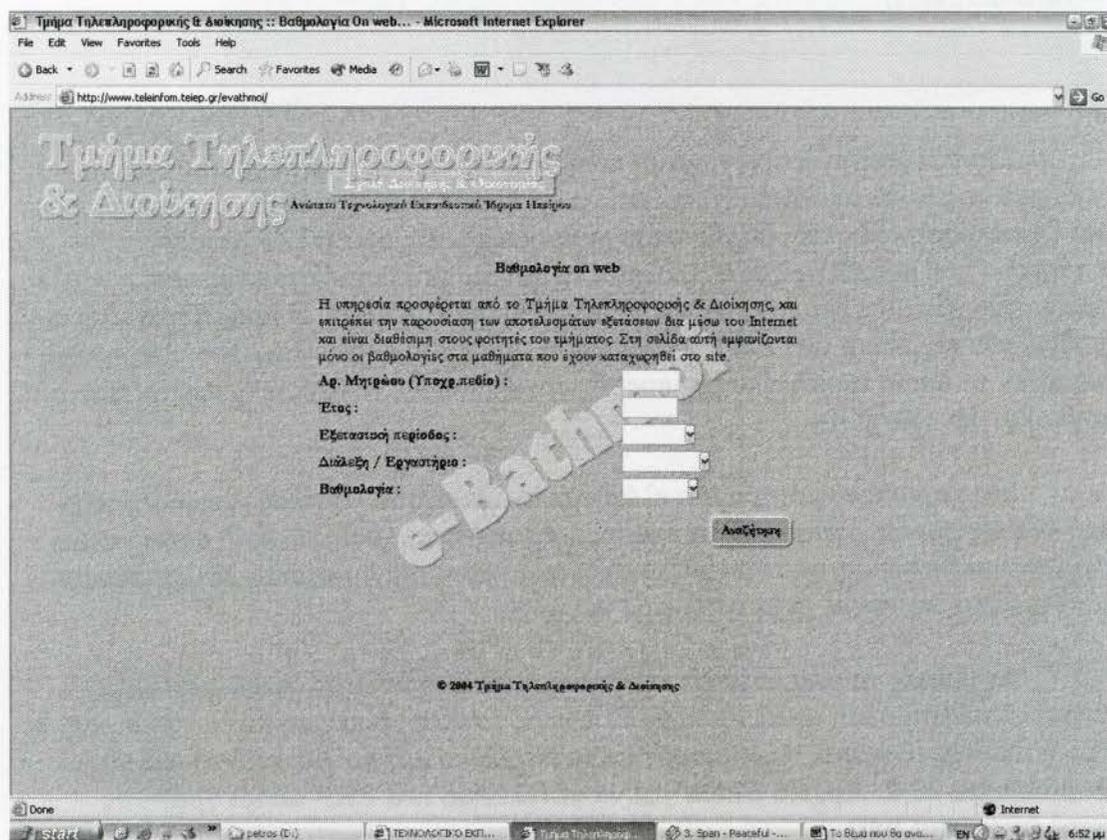
Αν θέλατε να εκτελέσετε το παραπάνω script στο Unix, χρειάζεται να το κάνετε εκτελέσιμο και απλά να το καλέσετε με το script.php echothis ή script.php -h. Στα Windows, μπορείτε να δημιουργήσετε ένα batch αρχείο για αυτό το σκοπό:
Παράδειγμα 23-2. Αρχείο Batch για να εκτελεστεί ένα PHP script γραμμής εντολών (script.bat)

```
@c:\php\cli\php.exe script.php %1 %2 %3 %4
```

Υποθέτοντας πως ονομάσατε το παραπάνω πρόγραμμα script.php, και έχετε το CLI php.exe σας στο c:\php\cli\php.exe αυτό το batch αρχείο θα εκτελεστεί για σας με τις επιπλέον επιλογές: script.bat echothis ή script.bat -h.

Η σελίδα που μπορεί κανείς να δει τις βαθμολογίες του στο διαδίκτυο είναι <http://www.teleinfom.teiep.gr/evathmoi/> όπου εμφανίζεται η παρακάτω

σελίδα και προτρέπει τον χρήστη να εισάγει τον Αρ. Μητρώου. Σε περίπτωση που ο χρήστης δεν εισάγει τον Αρ. Μητρώου τότε ειδοποιείται αυτόματα με ένα μήνυμα



που του εξηγεί πως πρέπει να εισάγει υποχρεωτικά έναν τετραψήφιο αριθμό και μετά να πατήσει το κουμπί αναζήτησης. Για το μήνυμα λάθους έχει γραφτεί JavaScript στην μεριά του client για να ελαττώσουμε την κίνηση στο Internet, γιατί στην περίπτωση που είναι λάθος ο Αρ. Μητρώου δεν θα σταλεί στον Server οπότε ο χρόνος που θα περάσει μέχρι να εμφανιστεί το μήνυμα είναι μηδαμινός αλλά και δε θα επιβαρύνουμε τον Server μιας και τον έλεγχο θα τον εκτελέσει η Η/Υ του χρήστη. Αν κάποιος χρήστης έχει έναν τριψήφιο αριθμό μητρώου θα πρέπει να εισάγει πριν το μηδέν, ώστε να γίνει τετραψήφιος και έτσι να μην εμφανιστεί το μήνυμα, δηλαδή κάποιος φοιτητής που έχει το πάσο 962 θα πρέπει να γράφει 0962.

Εκτός από το πεδίο που προτρέπει τον χρήστη να εισάγει τον Αρ. Μητρώου του υπάρχουν και πεδία που μπορεί ο χρήστης να εισάγει και άλλες πληροφορίες ώστε να μπορεί να εμφανίσει πιο αναλυτικά αποτελέσματα.

Τα υπόλοιπα πεδία είναι το έτος που εισάγει ο χρήστης, το έτος που έχει περάσει το μάθημα, η εξεταστική περίοδος, όπου μπορεί να επιλέξει ανάμεσα σε τρεις επιλογές, τη Χειμερινή, την Εαρινή περίοδο και κενή επιλογή, ανάλογα με την περίοδο που έδωσε το μάθημα.

Αν όμως αφήσει το πεδίο άθικτο τότε θα εμφανίσει και τα μαθήματα που έδωσε τη χειμερινή εξεταστική περίοδο καθώς και τα μαθήματα που έδωσε την εαρινή περίοδο . Στο πεδίο Διάλεξη /εργαστήριο μπορεί να επιλέξει ανάμεσα σε εργαστήριο ή διάλεξη ή τίποτα από τα δύο, όταν θέλει να εμφανίζει και τα εργαστήρια και τις διαλέξεις. Τέλος, στο πεδίο Βαθμολογία μπορεί να επιλέξει επιτυχόν για να δει μόνο τα μαθήματα που έχει περάσει ή μπορεί να επιλέξει να δει όλα τα μαθήματα.

Όταν ο χρήστης επιλέξει ένα έτος τότε γίνεται αυτόματα έλεγχος για το αν το έτος που εισήγαγε ο χρήστης έχει τέσσερα ψηφία.

Στην περίπτωση που δεν είναι έγκυρο το έτος θα εμφανιστεί μήνυμα που θα προτρέπει το χρήστη να εισάγει ένα τετραψήφιο έτος. Αν ο χρήστης εισάγει ένα έτος τετραψήφιο αλλά αυτό το έτος είναι μεγαλύτερο από το τρέχον έτος δεν θα εμφανιστεί κανένα μήνυμα λάθους και θα περάσει στην επόμενη σελίδα, που όπως είναι φυσικό δεν θα δει κανένα αποτέλεσμα μαθήματος. Αυτό γίνεται γιατί μπορεί τώρα να μην υπάρχει μέσα στην βάση ένα έτος μεγαλύτερο από το τρέχον αλλά κάποια στιγμή στο μέλλον θα εισαχθούν δεδομένα με αυτό το έτος, οπότε θα πρέπει να περάσει στην επόμενη σελίδα για να δει τα αποτελέσματα.

Δεν είναι υποχρεωτική η εισαγωγή στοιχείων πλην του αριθμού μητρώου, έτσι ανάλογα με το ποια πεδία συμπληρωθούν θα παραχθούν διαφορετικά αποτελέσματα. Αν συμπληρωθεί το πεδίο «Εξεταστική περίοδος» σε Χειμερινή, το πεδίο «Έτος» σε 2002 και το πεδίο «Βαθμολογία» σε Επιτυχόντα, τότε θα εμφανιστεί η παρακάτω σελίδα που στην πρώτη στήλη εμφανίζεται η περίοδος και το έτος που κάποιος φοιτητής πέρασε το μάθημα.

Στη δεύτερη στήλη εμφανίζεται το όνομα του μαθήματος, στην τρίτη στήλη εμφανίζεται ο βαθμός που πέρασε το μάθημα. Επειδή επιλέξαμε να εμφανίσει τα επιτυχόντα, εμφανίζεται πως ο φοιτητής έχει δώσει το μάθημα μια φορά και όλοι οι βαθμοί είναι μεγαλύτεροι του πέντε.

Στη τρίτη στήλη εμφανίζεται ο τύπος του μαθήματος, δηλαδή αν είναι εργαστήριο ή διάλεξη.

Κάθε εγγραφή εμφανίζεται με διαφορετικό χρώμα, ώστε να είναι εύκολη η εύρεση όταν το πλήθος των εγγραφών είναι μεγάλο.

Το όνομα του φοιτητή δεν εμφανίζεται πουθενά λόγω του προσωπικού των δεδομένων.

Τμήμα Τηλεπληροφορικής & Διοίκησης :: Βιβλιοθήκη On web... - Microsoft Internet Explorer

Address http://www.telefon.teiep.gr/evathnoi/results.php

Τμήμα Τηλεπληροφορικής & Διοίκησης

Ανώτατο Τεχνολογικό Εκπαιδευτικό Ίδρυμα Πελοποννήσου

Αποτέλεσμα αναζήτησης

Περίοδος Α / Β	Όνομα Μαθήματος	Βαθμός	Εργαστήρια / Διαλέξεις
Τυπικά Εξάμηνα Ε			
A Περίοδος Χειμερινή 2002	ΑΝΑΠΤΥΞΗ ΠΟΛΥΜΕΣΙΚΩΝ ΕΦΑΡΜΟΓΩΝ	7	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΗΛΕΚΤΡΟΝΙΚΟ ΕΜΠΟΡΙΟ	8	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΔΙΑΔΙΚΤΥΟΥ JAVA	9	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ Ι	6	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΙΙ	2	Διαλέξεις
B Περίοδος Χειμερινή 2002	ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΙΙ	5	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΤΗΛΕΜΑΤΙΚΗ	5	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΔΙΚΤΥΑ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ - ΔΙΑΣΥΝΔΕΣΗ ΔΙΚΤΥΩΝ	10	Διαλέξεις
A Περίοδος Χειμερινή 2002	ΑΝΑΠΤΥΞΗ ΠΟΛΥΜΕΣΙΚΩΝ ΕΦΑΡΜΟΓΩΝ	7	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΗΛΕΚΤΡΟΝΙΚΟ ΕΜΠΟΡΙΟ	8	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΔΙΑΔΙΚΤΥΟΥ JAVA	8	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ Ι	0	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΙΙ	7	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΤΗΛΕΜΑΤΙΚΗ	9	Εργαστήρια
A Περίοδος Χειμερινή 2002	ΔΙΚΤΥΑ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ - ΔΙΑΣΥΝΔΕΣΗ ΔΙΚΤΥΩΝ	10	Εργαστήρια

© 2004 Τμήμα Τηλεπληροφορικής & Διοίκησης - Τ.Ε.Ι. Ηπείρου

Αν τώρα εισάγεις τον αριθμό μητρώου και στο πεδίο βαθμολογία επιλέξεις επιτυχόντα θα σου εμφανιστεί η παρακάτω σελίδα.

Όπως μπορεί κανείς εύκολα να παρατηρήσει, εμφανίζονται όλες οι εγγραφές που υπάρχουν μέσα στη βάση και ο βαθμός είναι μεγαλύτερος από το πέντε.

Τα μαθήματα χωρίζονται ανά εξάμηνο και ταξινομούνται σε αύξουσα σειρά από το πρώτο εξάμηνο.

Ανάμεσα από κάθε εξάμηνο εισάγεται μια γραμμή με έντονα μαύρα γράμματα που πληροφορεί το χρήστη ότι η παρακάτω ομάδα μαθημάτων ανήκει στο εξάμηνο που αναγράφει.

Αν κάποιος φοιτητής περάσει το όγδοο εξάμηνο θα του εμφανιστεί ένα «Π» που θα του υποδηλώνει ότι βρίσκεται στο Πτυχίο.

Επίσης ομαδοποιεί τον τύπο του μαθήματος, δηλαδή εμφανίζει πρώτα όλες τις διαλέξεις του τρέχοντος εξαμήνου και μετά όλα τα εργαστήρια του εξαμήνου.

Τμήμα Τηλεπληροφορικής & Διοίκησης :: Βαθμολογία On web... - Microsoft Internet Explorer

Address: http://www.telerfon.tsiiep.gr/evathmol/results.php

Αποτελέσματα αναζητήσεως

Περίοδος Α / Β	Όνομα Μαθήματος	Βαθμός	Εργαστήρια / Διαλέξεις
Τυπικό Εξάμηνο Α			
A Περίοδος Χειμερινή 2000	ΑΓΓΛΙΚΑ Ι	8	Διαλέξεις
B Περίοδος Χειμερινή 2000	ΜΑΘΗΜΑΤΙΚΑ Ι	7	Διαλέξεις
A Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ (HARDWARE) - (καλοί)	8	Διαλέξεις
B Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ (SOFTWARE) - (καλοί)	6.3	Διαλέξεις
A Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΟ MANAGEMENT	7	Διαλέξεις
A Περίοδος Χειμερινή 2000	ΑΓΓΛΙΚΑ Ι	6.5	Εργαστήρια
A Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ (HARDWARE) - (καλοί)	9.5	Εργαστήρια
A Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ (SOFTWARE) - (καλοί)	8.5	Εργαστήρια
A Περίοδος Χειμερινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΟ MANAGEMENT	6.5	Εργαστήρια
Τυπικό Εξάμηνο Β			
B Περίοδος Εαρινή 2000	ΑΓΓΛΙΚΑ ΙΙ	6	Διαλέξεις
A Περίοδος Εαρινή 2000	ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Η/Υ Ι (PASCAL) ΙΙ (καλοί)	5	Διαλέξεις
A Περίοδος Εαρινή 2000	ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ Η/Υ	9	Διαλέξεις
B Περίοδος Εαρινή 2000	ΜΑΘΗΜΑΤΙΚΑ ΙΙ	5.5	Διαλέξεις
A Περίοδος Εαρινή 2000	ΜΑΚΡΟΟΙΚΟΝΟΜΙΑ	5	Διαλέξεις
A Περίοδος Εαρινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΔΙΟΙΚΗΣΗΣ	9	Διαλέξεις
A Περίοδος Εαρινή 2000	ΦΥΣΙΚΗ - ΗΛΕΚΤΡΟΝΙΚΗ	8	Διαλέξεις
A Περίοδος Εαρινή 2000	ΑΓΓΛΙΚΑ ΙΙ	7	Εργαστήρια
A Περίοδος Εαρινή 2000	ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ Η/Υ Ι (PASCAL) ΙΙ (καλοί)	8	Εργαστήρια
A Περίοδος Εαρινή 2000	ΛΕΙΤΟΥΡΓΙΚΑ ΣΥΣΤΗΜΑΤΑ Η/Υ	9.8	Εργαστήρια
A Περίοδος Εαρινή 2000	ΕΙΣΑΓΩΓΗ ΣΤΑ ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΔΙΟΙΚΗΣΗΣ	6	Εργαστήρια
A Περίοδος Εαρινή 2000	ΦΥΣΙΚΗ - ΗΛΕΚΤΡΟΝΙΚΗ	7	Εργαστήρια
Τυπικό Εξάμηνο Γ			
A Περίοδος Χειμερινή 2001	ΑΓΓΛΙΚΑ ΙΙΙ	7	Διαλέξεις
A Περίοδος Χειμερινή 2001	ΒΑΣΕΣ ΔΕΔΟΜΕΝΩΝ Ι	5	Διαλέξεις

Τι μπορεί να κάνει η PHP;

Οτιδήποτε. Η PHP επικεντρώνεται κυρίως στο server-side scripting, έτσι μπορείτε να κάνετε οτιδήποτε ένα άλλο CGI πρόγραμμα μπορεί να κάνει, όπως να μαζέψει δεδομένα, να παράγει δυναμικό περιεχόμενο σελίδων, ή να στείλει και να πάρει cookies. Αλλά η PHP μπορεί να κάνει πολύ περισσότερα.

Υπάρχουν τρεις κύριοι τομείς που χρησιμοποιείται ένα PHP script.

- **Server-side scripting.** Αυτό είναι το πιο παραδοσιακό και το κύριο πεδίο για την PHP. Χρειάζεστε τρία πράγματα για να δουλέψει αυτό. Τον PHP μεταγλωττιστή (parser) (CGI ή server module), ένα webserver (εξηγηρητή σελίδων) και ένα web browser ("φυλλομετρητή"). Πρέπει να τρέξετε τον webserver, με μια συνδεδεμένη εγκατάσταση της PHP. Μπορείτε να προσπελάσετε τα αποτελέσματα του PHP προγράμματος με ένα web browser, βλέποντας την σελίδα PHP μέσα από τον server.
- **Command line scripting.** Μπορείτε να φτιάξετε ένα PHP script για να το τρέχετε χωρίς server ή browser. Χρειάζεστε μόνο τον PHP μεταγλωττιστή για να την χρησιμοποιήσετε με αυτό τον τρόπο. Αυτός ο τύπος είναι ιδανικός για script που εκτελούνται συχνά με τη χρήση της cron (σε *nix ή Linux) ή με τον Task Scheduler (στα Windows). Αυτά τα script μπορούν επίσης να χρησιμοποιηθούν για απλές εργασίες επεξεργασίας κειμένου.

- Εγγραφή client-side GUI εφαρμογών (Γραφικά περιβάλλοντα χρηστών). Η PHP ίσως να μην είναι η πιο καλή γλώσσα για να γράψει κανείς παραθυριακές εφαρμογές, αλλά αν ξέρετε PHP πολύ καλά και θέλετε να χρησιμοποιήσετε κάποια προχωρημένα χαρακτηριστικά της PHP στις client-side εφαρμογές σας, μπορείτε επίσης να χρησιμοποιήσετε το PHP-GTK για αυτού του είδους τα προγράμματα. Έχετε επίσης τη δυνατότητα να γράφετε cross-platform εφαρμογές με αυτό τον τρόπο. Το PHP-GTK είναι μια επέκταση της PHP και δεν συμπεριλαμβάνεται στην κύρια διανομή.
- Η PHP μπορεί να χρησιμοποιηθεί σε όλα τα κύρια λειτουργικά συστήματα, συμπεριλαμβανομένου του Linux, πολλών εκδοχών του Unix (HP-UX, Solaris και OpenBSD), Microsoft Windows, Mac OS X, RISC OS και πιθανώς σε άλλα. Η PHP υποστηρίζει επίσης τους Apache, Microsoft Internet Information Server, Personal Web Server, Netscape και iPlanet servers, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd, και πολλούς άλλους webserver. Για την πλειοψηφία των server η PHP έχει ένα module, για τους υπόλοιπους η PHP μπορεί να λειτουργήσει ως ένας CGI επεξεργαστής.

Έτσι με την PHP έχετε την ελευθερία επιλογής ενός λειτουργικού συστήματος και ενός web server. Επιπλέον, έχετε επίσης την ελευθερία να χρησιμοποιήσετε συναρτησιακό (procedural) ή αντικειμενοστρεφή (object oriented) προγραμματισμό ή μια ανάμειξη τους. Αν και η παρούσα έκδοση δεν υποστηρίζει όλα τα πρότυπα χαρακτηριστικά, μεγάλες βιβλιοθήκες κώδικα και μεγάλες εφαρμογές (συμπεριλαμβανομένης και της βιβλιοθήκης PEAR) είναι γραμμένες μόνο με αντικειμενοστρεφή κώδικα.

Με την PHP δεν είστε περιορισμένοι να εξάγετε HTML. Οι δυνατότητες της PHP συμπεριλαμβάνουν την εξαγωγή εικόνων, αρχείων PDF, ακόμη και ταινίες Flash (χρησιμοποιώντας τα libswf και Ming) παράγονται αμέσως. Μπορείτε επίσης να εξάγετε εύκολα οποιοδήποτε κείμενο όπως XHTML και οποιοδήποτε άλλο XML αρχείο. Η PHP μπορεί να δημιουργεί αυτόματα αυτά τα αρχεία και να τα αποθηκεύει στο σύστημα αρχείων, αντί να τα εκτυπώνει, αποτελώντας έτσι μια server-side cache για το δυναμικό σας περιεχόμενο.

Ένα από τα πιο δυνατά και σημαντικά χαρακτηριστικά της PHP είναι η υποστήριξη που έχει για ένα μεγάλο σύνολο βάσεων δεδομένων. Η συγγραφή μιας σελίδας που υποστηρίζει βάσεις δεδομένων είναι εξαιρετικά απλή. Οι εξής βάσεις δεδομένων υποστηρίζονται μέχρι στιγμής:

Adabas D	Ingres	Oracle (OCI7 and OCI8)
dBase	InterBase	Ovrimos
Empress	FrontBase	PostgreSQL
FilePro (read-only)	mSQL	Solid
Hyperwave	Direct MS-SQL	Sybase
IBM DB2	MySQL	Velocis
Informix	ODBC	Unix dbm

Έχουμε επίσης μια αφαιρετική επέκταση DBX βάσεων δεδομένων (DBX database abstraction extension) που σας επιτρέπει διάφανα να χρησιμοποιείτε οποιαδήποτε βάση δεδομένων υποστηρίζεται από αυτή την επέκταση. Επιπλέον η PHP υποστηρίζει το ODBC, το Open Database Connection standard (Ανοιχτό πρότυπο Σύνδεσης Βάσεων δεδομένων) έτσι μπορείτε να συνδεθείτε σε οποιαδήποτε βάση δεδομένων που υποστηρίζει αυτό το παγκόσμιο πρότυπο.

Η PHP έχει επίσης υποστήριξη για επικοινωνία με άλλες υπηρεσίες χρησιμοποιώντας πρωτόκολλα όπως LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (στα Windows) και αμέτρητα άλλα. Μπορείτε επίσης να ανοίξετε raw network sockets και να αλληλεπιδράσετε με οποιοδήποτε άλλο πρωτόκολλο. Η PHP έχει ακόμη υποστήριξη για την περίπλοκη ανταλλαγή δεδομένων WDDX μεταξύ σχεδόν όλων των Web programming γλωσσών. Μιλώντας για δια-επικοινωνία, η PHP υποστηρίζει instantiation αντικειμένων Java και τα χρησιμοποιεί διάφανα σαν αντικείμενα PHP. Μπορείτε επίσης να χρησιμοποιήσετε την CORBA επέκταση μας για να προσπελάσετε remote (απομακρυσμένα) αντικείμενα.

Η PHP έχει εξαιρετικά χρήσιμα χαρακτηριστικά επεξεργασίας κειμένων, από την POSIX επέκταση ή τις Perl regular expressions μέχρι XML parsing αρχείων. Για τη μεταγλώττιση και την πρόσβαση αρχείων XML, υποστηρίζουμε τα πρότυπα SAX και DOM. Μπορείτε να χρησιμοποιήσετε την XSLT επέκταση μας για να μετατρέψετε τα XML αρχεία σε άλλες μορφές.

Καθώς χρησιμοποιείτε την PHP στον τομέα του ecommerce, θα βρείτε τις Cybercash payment, CyberMUT, VeriSign Payflow Pro και CCVS συναρτήσεις χρήσιμες για τα online προγράμματα πληρωμής σας.

Τελευταίο αλλά σημαντικό, έχουμε πολλές άλλες ενδιαφέρουσες επεκτάσεις, τις mmoGoSearch search engine συναρτήσεις, πολλά εργαλεία συμπίεσης (gzip, bz2), μετατροπές ημερολογίου, μεταφράσεις...

Όπως βλέπετε αυτή η σελίδα δεν είναι αρκετή για να απαριθμήσει όλα τα χαρακτηριστικά και πλεονεκτήματα της PHP.

Βγαίνοντας από την HTML

Όταν η PHP μεταγλωττίζει (parses) ένα αρχείο, απλά κάνει ένα πέρασμα στο κείμενο του αρχείου μέχρι να συναντήσει ένα από τα ειδικά tags που της λένε να αρχίσει να μεταφράζει το κείμενο ως κώδικα PHP. Ο parser (μεταγλωττιστής) τότε εκτελεί ολόκληρο τον κώδικα που βρίσκει, μέχρι να συναντήσει το επόμενο PHP tag κλεισίματος, το οποίο λέει στον parser να αρχίσει να κάνει ξανά, απλά ένα πέρασμα στο κείμενο. Αυτός είναι ο μηχανισμός που σας επιτρέπει να προσθέσετε PHP κώδικα μέσα σε HTML: ο,τιδήποτε βρίσκεται έξω από τα tags της PHP μένει τελείως μόνο, ενώ οτιδήποτε μέσα μεταγλωττίζεται ως κώδικας.

Υπάρχουν τέσσερα σύνολα από tags που μπορούν να χρησιμοποιηθούν για να δηλώσουμε τα κομμάτια που έχουν κώδικα σε PHP. Από αυτά, μόνο δύο (`<?php. . ?>` and `<script language="php">. . </script>`) είναι πάντα διαθέσιμα. Τα άλλα μπορούν να ενεργοποιηθούν και να απενεργοποιηθούν από το `php.ini` αρχείο ρυθμίσεων. Ενώ τα short-form tags και τα tags που μοιάζουν με αυτά της ASP μπορεί να είναι βολικά, δεν είναι τόσο portable όσο οι μακρύτερες εκδόσεις. Επίσης, αν σκοπεύετε να προσθέσετε PHP κώδικα σε XML ή XHTML, θα χρειαστεί να χρησιμοποιήσετε την `<?php. . ?>` φόρμα για να προσαρμοστεί στην XML.

Τα tags που υποστηρίζονται από την PHP είναι:

Παράδειγμα. Τρόποι για να βγείτε (escape) από την HTML

1. `<?php echo("if you want to serve XHTML or XML documents, do like this\n"); ?>`
2. `<? echo ("this is the simplest, an SGML processing instruction\n"); ?>`
`<?= expression ?> This is a shortcut for "<? echo expression ?>"`
3. `<script language="php">`
 `echo ("some editors (like FrontPage) don't`
 `like processing instructions");`
`</script>`
4. `<% echo ("You may optionally use ASP-style tags"); %>`

Ο πρώτος τρόπος, `<?php. . ?>`, είναι και ο προτιμότερος, καθώς επιτρέπει τη χρήση της PHP σε κώδικα συμβατό με την XML όπως η XHTML.

Ο δεύτερος τρόπος δεν είναι πάντα διαθέσιμος. Τα σύντομα tags είναι διαθέσιμα μόνο όταν έχουν ενεργοποιηθεί. Αυτό μπορεί να γίνει μέσω της συνάρτησης `short_tags()` (μόνο στην PHP 3), ενεργοποιώντας την επιλογή ρύθμισης `short_open_tag` στο αρχείο ρυθμίσεων της PHP, ή κάνοντας `compile` την PHP με την επιλογή `--enable-short-tags` στο `configure`. Ακόμη και αν είναι ενεργοποιημένο ως προεπιλογή στο `php.ini-dist`, η χρήση των short tags δεν προτιμάται.

Ο τέταρτος τρόπος είναι διαθέσιμος μόνο αν τα ASP-style tags έχουν ενεργοποιηθεί χρησιμοποιώντας την `asp_tags` επιλογή ρυθμίσεων.

Σημείωση: Υποστήριξη για τα ASP-style tags προστέθηκε στην έκδοση 3.0.4.

Σημείωση: Η χρήση των short tags θα πρέπει να αποφεύγεται κατά την ανάπτυξη εφαρμογών ή βιβλιοθηκών (libraries) που προορίζονται για διανομή (redistribution), ή εφαρμογή σε PHP servers που δεν τους χειρίζεστε οι ίδιοι, επειδή τα short tags μπορεί να μην υποστηρίζονται από τον τελικό server. Για μεταφέρσιμο (portable), κώδικα που θα προορίζεται για χρήση και από άλλους, βεβαιωθείτε ότι δεν κάνετε χρήση των short tags.

Το tag κλεισίματος για το block θα συμπεριλάβει το αμέσως επόμενο trailing newline αν υπάρχει. Επίσης, το tag κλεισίματος αυτόματα υποδηλώνει και ένα ερωτηματικό. Δεν χρειάζεται να έχετε ερωτηματικό για να τερματίσετε την τελευταία γραμμή ενός PHP block.

Η PHP σας επιτρέπει να χρησιμοποιήσετε δομές σαν αυτή:

Παράδειγμα Προχωρημένος τρόπος για να κάνετε escape

```
<?php
if ($expression) {
    ?>
    <strong>This is true.</strong>
    <?php
} else {
    ?>
    <strong>This is false.</strong>
    <?php
}
```

Αυτό λειτουργεί όπως περιμέναμε, επειδή όταν η PHP φτάνει στα ?> tags κλεισίματος, απλά αρχίζει να εμφανίζει ο,τιδήποτε βρει μέχρι να συναντήσει ένα άλλο tag ανοίγματος. Το παράδειγμα που δόθηκε εδώ έχει επινοηθεί, φυσικά, με σκοπό να εμφανίσουμε μεγάλα blocks κειμένου, αφού το να ξεφεύγουμε από τη μεταγλώττιση της PHP είναι γενικά πιο αποτελεσματικό από το να στέλνουμε ολόκληρο το κείμενο μέσω της συνάρτησης **echo()** ή της **print()** ή κάτι τέτοιο.

Παραπομπή Συναρτήσεων

Πίνακας Περιεχομένων

I. Apache-specific Functions
II. Array Functions
III. Aspell functions [deprecated]
IV. BCMath Arbitrary Precision Mathematics Functions
V. Bzip2 Compression Functions
VI. Calendar συναρτήσεις
VII. CCVS API Functions [deprecated]
VIII. COM support functions for Windows
IX. Class/Object συναρτήσεις
X. ClibPDF Functions
XI. Crack Functions

<u>XII. CURL, Client URL Library Functions</u>
<u>XIII. Cybercash Payment Functions</u>
<u>XIV. Cyrus IMAP administration Functions</u>
<u>XV. Character type συναρτήσεις</u>
<u>XVI. Database (dbm-style) Abstraction Layer Functions</u>
<u>XVII. Date/Time συναρτήσεις</u>
<u>XVIII. dBase Functions</u>
<u>XIX. DBM Functions [deprecated]</u>
<u>XX. dbx Functions</u>
<u>XXI. DB++ Functions</u>
<u>XXII. Direct IO Functions</u>
<u>XXIII. Directory συναρτήσεις</u>
<u>XXIV. DOM XML Functions</u>
<u>XXV. .NET Functions</u>
<u>XXVI. Error Handling and Logging Functions</u>
<u>XXVII. File Alteration Monitor Functions</u>
<u>XXVIII. FrontBase Functions</u>
<u>XXIX. filePro Functions</u>
<u>XXX. Filesystem Functions</u>
<u>XXXI. Forms Data Format Functions</u>
<u>XXXII. FriBiDi Functions</u>
<u>XXXIII. FTP συναρτήσεις</u>
<u>XXXIV. Function Handling Functions</u>
<u>XXXV. Gettext</u>
<u>XXXVI. GMP Functions</u>
<u>XXXVII. HTTP Συναρτήσεις</u>
<u>XXXVIII. Hyperwave Functions</u>
<u>XXXIX. Hyperwave API Functions</u>
<u>XL. iconv Functions</u>
<u>XLI. Image συναρτήσεις</u>
<u>XLII. IMAP, POP3 and NNTP Functions</u>
<u>XLIII. Informix Functions</u>
<u>XLIV. InterBase Functions</u>
<u>XLV. Ingres II Functions</u>
<u>XLVI. IRC Gateway Functions</u>
<u>XLVII. PHP / Java Integration</u>
<u>XLVIII. LDAP Functions</u>
<u>XLIX. LZF Functions</u>
<u>L. Mail συναρτήσεις</u>
<u>LI. mailparse Functions</u>
<u>LII. Mathematical συναρτήσεις</u>
<u>LIII. Multi-Byte String Functions</u>
<u>LIV. MCAL Functions</u>
<u>LV. Mcrypt Encryption Functions</u>
<u>LVI. MCVE Payment Functions</u>
<u>LVII. Mhash Functions</u>
<u>LVIII. Mimetype Functions</u>
<u>LIX. Microsoft SQL Server Functions</u>

LX. <u>Ming functions for Flash</u>
LXI. <u>Miscellaneous Functions</u>
LXII. <u>mnoGoSearch Functions</u>
LXIII. <u>mSQL Functions</u>
LXIV. <u>MySQL Functions</u>
LXV. <u>Improved MySQL Extension</u>
LXVI. <u>Mohawk Software Session Handler Functions</u>
LXVII. <u>muscat Functions</u>
LXVIII. <u>Network Functions</u>
LXIX. <u>Ncurses Terminal Screen Control Functions</u>
LXX. <u>Lotus Notes Functions</u>
LXXI. <u>NSAPI-specific Functions</u>
LXXII. <u>Unified ODBC Functions</u>
LXXIII. <u>Object Aggregation/Composition Functions</u>
LXXIV. <u>Oracle 8 functions</u>
LXXV. <u>OpenSSL Functions</u>
LXXVI. <u>Oracle Functions</u>
LXXVII. <u>Ovrimos SQL Functions</u>
LXXVIII. <u>Output Control Functions</u>
LXXIX. <u>Object property and method call overloading</u>
LXXX. <u>PDF functions</u>
LXXXI. <u>Verisign Payflow Pro Functions</u>
LXXXII. <u>PHP Options&Information</u>
LXXXIII. <u>POSIX Functions</u>
LXXXIV. <u>PostgreSQL Functions</u>
LXXXV. <u>Process Control Functions</u>
LXXXVI. <u>Program Execution Functions</u>
LXXXVII. <u>Printer Functions</u>
LXXXVIII. <u>Pspell Functions</u>
LXXXIX. <u>GNU Readline</u>
XC. <u>GNU Recode Functions</u>
XCI. <u>Regular Expression Functions (Perl-Compatible)</u>
XCII. <u>qtdom Functions</u>
XCIII. <u>Regular Expression συναρτήσεις (Εκτεταμένο POSIX)</u>
XCIV. <u>Semaphore, Shared Memory and IPC Functions</u>
XCV. <u>SESAM Database Functions</u>
XCVI. <u>Session Handling Functions</u>
XCVII. <u>Shared Memory Functions</u>
XCVIII. <u>SimpleXML functions</u>
XCIX. <u>SOAP Functions</u>
C. <u>SQLite</u>
CI. <u>Shockwave Flash Functions</u>
CII. <u>SNMP Functions</u>
CIII. <u>Socket Functions</u>
CIV. <u>Standard PHP Library (SPL) Functions</u>
CV. <u>Stream Functions</u>
CVI. <u>String συναρτήσεις</u>
CVII. <u>Sybase Functions</u>

CVIII. <u>TCP Wrappers Functions</u>
CIX. <u>Tidy Functions</u>
CX. <u>Tokenizer Functions</u>
CXI. <u>URL συναρτήσεις</u>
CXII. <u>Variable Functions</u>
CXIII. <u>ypopmail Functions</u>
CXIV. <u>W32api Functions</u>
CXV. <u>WDDX Functions</u>
CXVI. <u>XML parser συναρτήσεις</u>
CXVII. <u>XML-RPC Functions</u>
CXVIII. <u>xdiff Functions</u>
CXIX. <u>XSLT Functions</u>
CXX. <u>YAZ Functions</u>
CXXI. <u>YP/NIS Functions</u>
CXXII. <u>Zip File Functions (Read Only Access)</u>
CXXIII. <u>Zlib Compression Functions</u>

Προτεραιότητα τελεστών

Η προτεραιότητα των τελεστών καθορίζει το πόσο "σφιχτά" συνδέονται δυο εκφράσεις μαζί. Για παράδειγμα, στην έκφραση $1 + 5 * 3$, η απάντηση είναι 16 και όχι 18 επειδή ο τελεστής του πολλαπλασιασμού (" $*$ ") έχει προτεραιότητα σε σχέση με τον τελεστή της πρόσθεσης (" $+$ "). Οι παρενθέσεις μπορούν να χρησιμοποιηθούν για να αλλάξουν τις προτεραιότητες, αν χρειάζεται. Για παράδειγμα: Το $(1 + 5) * 3$ έχει ως αποτέλεσμα το 18.

Ο ακόλουθος πίνακας παραθέτει την προτεραιότητα των τελεστών με την χαμηλότερη προτεραιότητα να παρατίθεται πρώτη.

Operator Precedence

Σχετικότητα	Τελεστές
αριστερή	,
αριστερή	or
αριστερή	xor
αριστερή	and
δεξιά	print
αριστερή	= += -= *= /= .= %= &= = ^= <<= >>=
αριστερή	? :
αριστερή	
αριστερή	&&
αριστερή	

Σχετικότητα	Τελεστές
αριστερή	^
αριστερή	&
Χωρίς σύνδεση	== != === !==
Χωρίς σύνδεση	< <= > >=
Αριστερή	<< >>
Αριστερή	+ - .
Αριστερή	* / %
Δεξιά	! ~ ++ -- (int) (float) (string) (array) (object) @
Δεξιά	[
Χωρίς σύνδεση	new

Σημείωση: Παρόλο που το ! έχει προτεραιότητα έναντι του =, η PHP επιτρέπει εκφράσεις παρόμοιες με την ακόλουθη: if (!\$a = foo()), και σε κάθε περίπτωση το αποτέλεσμα από την foo() τοποθετείται στην \$a.

Αριθμητικοί Τελεστές

Αριθμητικοί Τελεστές

Παράδειγμα	Όνομα	Αποτέλεσμα
\$a + \$b	Πρόσθεση	Αποτέλεσμα του \$a και του \$b.
\$a - \$b	Αφαίρεση	Διαφορά του \$a και του \$b.
\$a * \$b	Πολλαπλασιασμός	Γινόμενο του \$a και του \$b.
\$a / \$b	Διαίρεση	Αποτέλεσμα του \$a και του \$b.
\$a % \$b	Modulus	Υπόλοιπο του \$a διαιρεμένου από το \$b.

Ο τελεστής διαίρεσης ("/") επιστρέφει μια τιμή κινητής υποδιαστολής κάθε φορά, ακόμη και αν οι δυο τελεστές είναι ακέραιοι (ή strings που μετατρέπονται σε ακέραιους).

Ανάθεση Τελεστών

Ο βασικός τελεστής ανάθεσης είναι ο "=". Η πρώτη σας σκέψη ίσως είναι ότι αυτό είναι το ίδιο με το "ισούται με". Αλλά δεν είναι. Αυτό που πραγματικά σημαίνει είναι ότι το αριστερό μέρος του τελεστή παίρνει την τιμή της έκφρασης στα δεξιά (δηλαδή, "ανατίθεται σε").

Η τιμή μιας έκφρασης ανάθεσης είναι η τιμή που της ανατίθεται. Δηλαδή, η τιμή του "\$a = 3" είναι 3. Αυτό σας επιτρέπει να κάνετε μερικά περίπλοκα πράγματα:

```
$a = ($b = 4) + 5; // $a is equal to 9 now, and $b has been set to 4.
```

Πέρα από το βασικό τελεστή ανάθεσης, υπάρχουν "σύνθετοι τελεστές" για όλους τους δυαδικούς αριθμητικούς και αλφαριθμητικούς τελεστές που σας επιτρέπουν να χρησιμοποιήσετε μια έκφραση και στη συνέχεια να θέσετε την τιμή στο αποτέλεσμα της έκφρασης. Για παράδειγμα:

```
$a = 3;  
$a += 5; // sets $a to 8, as if we had said: $a = $a + 5;  
$b = "Hello";  
$b .= "There!"; // sets $b to "Hello There!", just like $b = $b . "There!";
```

Σημειώστε ότι η ανάθεση αντιγράφει την αρχική μεταβλητή στην καινούρια (ανάθεση με τιμή), συνεπώς αλλαγές στο ένα δε θα επηρεάσουν το άλλο. Αυτό μπορεί επίσης να έχει σχέση αν χρειαστεί να αντιγράψετε κάτι όπως έναν μεγάλο πίνακα μέσα σε ένα στενό loop. Η PHP 4 υποστηρίζει την ανάθεση με αναφορά, χρησιμοποιώντας τη \$var = &\$othervar; σύνταξη, αλλά αυτό δεν είναι δυνατό στην PHP 3. 'Ανάθεση με αναφορά' σημαίνει πως και οι δυο μεταβλητές καταλήγουν στο να δείχνουν στα ίδια δεδομένα, και τίποτα δεν αντιγράφεται πουθενά.

Δυαδικοί Τελεστές

Οι δυαδικοί τελεστές σας επιτρέπουν να ενεργοποιήσετε ή να απενεργοποιήσετε συγκεκριμένα bits μέσα σε κάποιο ακέραιο. Αν και οι δύο παράμετροι (αριστερή και δεξιά) είναι strings, ο δυαδικός τελεστής θα ενεργήσει στους χαρακτήρες αυτού του string.

```
<?php  
echo 12 ^ 9; // Outputs '5'  
  
echo "12" ^ "9"; // Outputs the Backspace character (ascii 8)  
// ('1' (ascii 49) ^ ('9' (ascii 57))) = #8  
  
echo "hallo" ^ "hello"; // Outputs the ascii values #0 #4 #0 #0 #0  
// 'a' ^ 'e' = #4  
?>
```

Διαδικικοί Τελεστές

Παράδειγμα	Όνομα	Αποτέλεσμα
$\$a \& \b	And	Τα bits που είναι 1 τόσο στο $\$a$ όσο και στο $\$b$, ενεργοποιούνται.
$\$a \b	Or	Τα bits που είναι 1 είτε στο $\$a$ είτε στο $\$b$, ενεργοποιούνται.
$\$a \wedge \b	Xor	Τα bits που είναι 1 είτε στο $\$a$ είτε στο $\$b$, αλλά όχι και στα δύο, ενεργοποιούνται.
$\sim \$a$	Not	Τα bits που δεν είναι ενεργοποιημένα στο $\$a$, ενεργοποιούνται, και αντίστροφα.
$\$a \ll \b	Shift left	Μετακίνηση (σιφτάρισμα) των bits του $\$a$ κατά $\$b$ βήματα προς τα αριστερά (κάθε βήμα σημαίνει "πολλαπλασιασμός επί δύο")
$\$a \gg \b	Shift right	Μετακίνηση (σιφτάρισμα) των bits του $\$a$ κατά $\$b$ βήματα προς τα δεξιά (κάθε βήμα σημαίνει "διαίρεση επί δύο")

Τελεστές σύγκρισης

Οι τελεστές σύγκρισης, όπως λέει και το όνομα τους, σας επιτρέπουν να συγκρίνετε δυο τιμές. Ίσως ακόμη ενδιαφέρεστε να δείτε συνδέσμους σχετικά με τον πίνακα σύγκρισης τύπων, αφού δείχνουν παραδείγματα με συγκρίσεις που έχουν σχέση με τύπους.

Πίνακας. Τελεστές Σύγκρισης

Παράδειγμα	Όνομα	Αποτέλεσμα
$\$a == \b	Ισότητα	TRUE αν το $\$a$ είναι ίσο με το $\$b$.
$\$a === \b	Ομοιότητα	TRUE αν το $\$a$ είναι ίσο με το $\$b$, και είναι επιπλέον του ίδιου τύπου. (Στην PHP 4 μόνο)
$\$a != \b	΄νισα	TRUE αν το $\$a$ δεν είναι ίσο με το $\$b$.
$\$a <> \b	Όχι ίσα	TRUE αν το $\$a$ δεν είναι ίσο με το $\$b$.
$\$a !== \b	Ανόμοια	TRUE αν το $\$a$ δεν είναι ίσο με το $\$b$, ή αν δεν είναι του ίδιου τύπου. (στην PHP 4 μόνο)
$\$a < \b	Μικρότερο από	TRUE αν το $\$a$ είναι ακριβώς μικρότερο από το $\$b$.
$\$a > \b	Μεγαλύτερο από	TRUE αν το $\$a$ είναι αυστηρώς μεγαλύτερο από το $\$b$.
$\$a <= \b	Μικρότερο από ή ίσο με	TRUE αν το $\$a$ είναι μικρότερο από ή ίσο με το $\$b$.
$\$a >= \b	Μεγαλύτερο από ή ίσο με	TRUE αν το $\$a$ είναι μεγαλύτερο από ή ίσο με το $\$b$.

Ακόμη ένας τελεστής υπόθεσης είναι ο "?:" (ή τριαδικός) τελεστής, ο οποίος λειτουργεί όπως και στη C και σε πολλές άλλες γλώσσες.

```

<?php
// Example usage for: Ternary Operator
$action = (empty($_POST['action'])) ? 'default' : $_POST['action'];

// The above is identical to this if/else statement
if (empty($_POST['action'])) {
    $action = 'default';
} else {
    $action = $_POST['action'];
}
?>

```

Η έκφραση $(expr1) ? (expr2) : (expr3)$ παίρνει την τιμή $expr2$ αν $expr1$ γίνει **TRUE**, και την $expr3$ αν $expr1$ γίνει **FALSE**.

Τελεστές Ελέγχου Λαθών

Η PHP υποστηρίζει έναν τελεστή ελέγχου λαθών: το σύμβολο (**@**). Όταν προηγείται σε μια έκφραση στην PHP, οποιοδήποτε μήνυμα λάθους που ίσως δημιουργηθεί από αυτή την έκφραση θα αγνοηθεί.

Αν το χαρακτηριστικό των `track_errors` έχει ενεργοποιηθεί, οποιοδήποτε μήνυμα λάθους που θα δημιουργηθεί από την έκφραση θα αποθηκευτεί στη μεταβλητή `$php_errormsg`. Αυτή η μεταβλητή θα αλλάζει ανάλογα με κάθε λάθος, συνεπώς θα πρέπει να την ελέγχετε συχνά αν θέλετε να τη χρησιμοποιείτε.

```

<?php
/* Intentional file error */
$my_file = @file('non_existent_file') or
    die("Failed opening file: error was '$php_errormsg'");

// this works for any expression, not just functions:
$value = @$cache[$key];
// will not issue a notice if the index $key doesn't exist.

?>

```

Σημείωση: Ο τελεστής **@** δουλεύει μόνο στις εκφράσεις. Ένας απλός κανόνας είναι ο εξής: αν μπορείτε να πάρετε την τιμή από κάτι, μπορείτε να βάλετε ως πρόθεμα σ'αυτό, το **@**. Για παράδειγμα, μπορείτε να το προσθέσετε στις μεταβλητές, στις συναρτήσεις και στις κλήσεις του **include()**, στις σταθερές κ.ο.κ. Δεν μπορείτε να το προσθέσετε στους ορισμούς συναρτήσεων ή κλάσεων, ή υποθετικών δομών όπως τις **if** και **foreach** κ.ο.κ.

Σημείωση: Το πρόθεμα του τελεστή ελέγχου λαθών **"@"** δε θα αποτρέψει την εμφάνιση των μηνυμάτων που είναι αποτέλεσμα από λάθη μεταγλώττισης.

Τελεστές Εκτέλεσης

Η PHP υποστηρίζει έναν τελεστή εκτέλεσης: τα backticks (`). Σημειώστε ότι αυτά δεν είναι μονά εισαγωγικά (single-quotes)! Η PHP θα προσπαθήσει να εκτελέσει το περιεχόμενο των backticks σαν εντολή shell. Το αποτέλεσμα θα επιστραφεί (π.χ., δε θα πεταχτεί απλά στην έξοδο, αλλά μπορεί να ανατεθεί σε μια μεταβλητή). Η χρήση του backtick τελεστή είναι όμοια με τη `shell_exec()`.

```
$output = `ls -al`;
echo "<pre>$output</pre>";
```

Σημείωση: Ο backtick τελεστής είναι απενεργοποιημένος όταν το safe mode είναι ενεργοποιημένο ή η `shell_exec()` είναι απενεργοποιημένη.

Τελεστές Αύξησης/Μειώσης

Η PHP υποστηρίζει τελεστές πρό- και μετά- αύξησης όπως η C.

Πίνακας. Τελεστές Αύξησης/Μειώσης

Παράδειγμα	Όνομα	Αποτέλεσμα
<code>++\$a</code>	Προ-αύξηση	Αυξάνει το \$a κατά ένα και επιστρέφει το \$a.
<code>\$a++</code>	Μετά-αύξηση	Επιστρέφει το \$a, και μετά αυξάνει το \$a κατά ένα.
<code>--\$a</code>	Προ-μείωση	Μειώνει το \$a κατά ένα, και μετά επιστρέφει το \$a.
<code>\$a--</code>	Μετά-μείωση	Επιστρέφει το \$a, και μετά μειώνει το \$a κατά ένα.

Εδώ έχουμε ένα απλό παράδειγμα κώδικα:

```
<?php
echo "<h3>Postincrement</h3>";
$a = 5;
echo "Should be 5: " . $a++ . "<br />\n";
echo "Should be 6: " . $a . "<br />\n";

echo "<h3>Preincrement</h3>";
$a = 5;
echo "Should be 6: " . ++$a . "<br />\n";
echo "Should be 6: " . $a . "<br />\n";

echo "<h3>Postdecrement</h3>";
$a = 5;
echo "Should be 5: " . $a-- . "<br />\n";
echo "Should be 4: " . $a . "<br />\n";

echo "<h3>Predecrement</h3>";
$a = 5;
echo "Should be 4: " . --$a . "<br />\n";
echo "Should be 4: " . $a . "<br />\n";
?>
```

Η PHP ακολουθεί τις συνθήκες της Perl όταν διαχειρίζεται αριθμητικές πράξεις σε μεταβλητές χαρακτήρων και όχι αυτές τις C. Για παράδειγμα, στην Perl το 'Z'+1 επιστρέφει 'AA', ενώ στη C το 'Z'+1 επιστρέφει ']' (ord('Z') == 90, ord(']') == 91). Σημειώστε ότι οι μεταβλητές χαρακτήρων μπορούν να αυξηθούν αλλά όχι να μειωθούν.

Παράδειγμα 10-1. Αριθμητικές Πράξεις σε Μεταβλητές Χαρακτήρων

```
<?php
$i = 'W';
for($n=0; $n<6; $n++)
    echo ++$i . "\n";

/*
    Produces the output similar to the following:

X
Y
Z
AA
AB
AC

*/
```

Λογικοί Τελεστές

Πίνακας. Λογικοί Τελεστές

Παράδειγμα	Όνομα	Αποτέλεσμα
\$a and \$b	And	TRUE αν και το \$a και το \$b είναι TRUE.
\$a or \$b	Or	TRUE αν είτε το \$a είτε το \$b είναι TRUE.
\$a xor \$b	Xor	TRUE αν είτε το \$a είτε το \$b είναι TRUE, αλλά όχι και τα δυο.
! \$a	Not	TRUE αν το \$a δεν είναι TRUE.
\$a && \$b	And	TRUE αν και το \$a και το \$b είναι TRUE.
\$a \$b	Or	TRUE αν είτε το \$a είτε το \$b είναι TRUE.

Η εξήγηση για τις δυο διαφορετικές μορφές των "and" και των "or" τελεστών είναι ότι λειτουργούν με διαφορετικές προτεραιότητες.

Τελεστές για strings

Υπάρχουν δύο τελεστές για strings (αλφαριθμητικών). Ο πρώτος είναι ο τελεστής σύνδεσης ('.'), ο οποίος επιστρέφει τη σύνδεση των αριστερών και των δεξιών παραμέτρων. Ο δεύτερος είναι ο τελεστής ανάθεσης σύνδεσης ('.='), ο οποίος προσθέτει την παράμετρο της δεξιά πλευρά στην παράμετρο της αριστερής πλευράς. Παρακαλώ διαβάστε την Ανάθεση Τελεστών για περισσότερες πληροφορίες.

```
$a = "Hello ";  
$b = $a . "World!"; // now $b contains "Hello World!"  
  
$a = "Hello ";  
$a .= "World!"; // now $a contains "Hello World!"
```

Σταθερές

Μια σταθερά είναι ένας identifier (όνομα) για μια απλή τιμή. Όπως φαίνεται και από το όνομα, αυτή η τιμή δεν μπορεί να αλλάξει κατά την εκτέλεση του script. (Οι 'μαγικές σταθερές' `__FILE__` και `__LINE__` φαίνεται να είναι εξαίρεση σ' αυτόν τον κανόνα, αλλά δεν είναι πραγματικές σταθερές.) Είναι προκαθορισμένο πως μια σταθερά είναι case-sensitive. Κατά συνθήκη οι constant identifiers είναι πάντα στα κεφαλαία.

Το όνομα μιας σταθεράς ακολουθεί τους ίδιους κανόνες όπως και οποιαδήποτε ετικέτα (label) στην PHP. Ένα έγκυρο όνομα σταθεράς αρχίζει με ένα γράμμα ή underscore, ακολουθούμενο από οποιονδήποτε αριθμό γραμμάτων, αριθμών ή underscore. Ως regular expression, θα εκφραζόταν ως εξής: `[a-zA-Z_\x7f-\xff][a-zA-Z0-9_\x7f-\xff]*`

Σημείωση: Για τους σκοπούς μας εδώ, ένα γράμμα είναι από το a-z, A-Z, και οι ASCII χαρακτήρες από το 127 μέχρι το 255 (0x7f-0xff).

Όπως οι superglobals, η εμβέλεια (scope) μιας σταθεράς είναι global. Μπορείτε να έχετε πρόσβαση σ' αυτή από οποιοδήποτε μέρος στο script χωρίς να λαμβάνετε υπόψη την εμβέλεια.

Σύνταξη

Μπορείτε να ορίσετε μια σταθερά χρησιμοποιώντας την `define()`-συνάρτηση. Μόλις μια σταθερά οριστεί, δεν μπορεί ποτέ να αλλάξει ή να μην είναι ορισμένη.

Μόνο scalar (βαθμωτά) δεδομένα (**boolean**, **integer**, **float** και **string**) μπορούν να συμπεριλαμβάνονται στις σταθερές.

Μπορείτε να πάρετε την τιμή μιας σταθεράς απλά καλώντας την με το όνομα της. Σε αντίθεση με τις μεταβλητές, δεν θα πρέπει να βάζετε μπροστά από μια σταθερά το \$. Μπορείτε επίσης να χρησιμοποιήσετε τη συνάρτηση **constant()**, για να διαβάσετε την τιμή μιας σταθεράς, αν θέλετε να πάρετε το όνομα μιας σταθεράς δυναμικά. Χρησιμοποιείστε την **get_defined_constants()** για να πάρετε μια λίστα όλων των σταθερών που έχουν οριστεί.

Σημείωση: Οι σταθερές και οι (global) μεταβλητές είναι σε διαφορετικό namespace. Αυτό σημαίνει πως για παράδειγμα το **TRUE** και το **\$TRUE** είναι γενικώς διαφορετικά.

Αν χρησιμοποιείτε μία μη καθορισμένη σταθερά, η PHP υποθέτει ότι εννοείτε το όνομα της ίδιας της σταθεράς, σαν να είχατε καλέσει ως **string** (CONSTANT εναντίον "CONSTANT"). Ένα λάθος επιπέδου **E_NOTICE** θα εμφανιστεί όταν αυτό συμβεί.

Οι διαφορές μεταξύ σταθερών και μεταβλητών είναι οι εξής:

- Οι σταθερές δεν έχουν το σύμβολο του δολαρίου (\$) μπροστά τους.
- Οι σταθερές μπορούν να οριστούν μόνο χρησιμοποιώντας τη συνάρτηση **define()** και όχι με απλή ανάθεση.
- Οι σταθερές μπορούν να οριστούν και να προσπελαστούν από οπουδήποτε χωρίς να λάβουμε υπόψη τους κανόνες εμβέλειας μεταβλητών.
- Οι σταθερές δεν μπορούν να οριστούν ξανά ή να μην είναι καθορισμένες από τη στιγμή που έχουν οριστεί.
- Οι σταθερές μπορούν να υπολογίσουν μόνο βαθμωτές τιμές.

Παράδειγμα. Ορίζοντας σταθερές

```
<?php
define("CONSTANT", "Hello world.");
echo CONSTANT; // outputs "Hello world."
echo Constant; // outputs "Constant" and issues a notice.
```



ΠΕΡΙΕΧΟΜΕΝΑ

Σύντομη γενική περιγραφή του θέματος	1
ΕΓΚΑΤΑΣΤΑΣΕΙΣ ΤΩΝ ΛΟΓΙΣΜΙΚΩΝ.....	3
ΕΠΕΞΗΓΗΣΗ ΤΩΝ ΑΡΧΕΙΩΝ.....	6
Next.....	11
Τι μπορεί να κάνει η PHP;.....	47
Βγαίνοντας από την HTML.....	50
Παραπομπή Συναρτήσεων.....	51
Προτεραιότητα τελεστών	54
Αριθμητικοί Τελεστές.....	55
Ανάθεση Τελεστών.....	55
Διαδικοί Τελεστές.....	56
Τελεστές σύγκρισης.....	57
Τελεστές Ελέγχου Λαθών.....	58
Τελεστές Εκτέλεσης	59
Τελεστές Αύξησης/Μειώσης	59
Λογικοί Τελεστές.....	60
Τελεστές για strings.....	61
Σταθερές	61
Σύνταξη.....	61