

Εργαστήριο #7

Τι πρέπει να έχετε ολοκληρώσει από το προηγούμενο εργαστήριο.

- Θα πρέπει να ξέρετε να κατασκευάσετε μια ιστοσελίδα με φόρμα HTML και στοιχεία εισόδου τύπου text, radiobutton, checkbox, select και submit. Θα πρέπει επίσης να μπορείτε να κατασκευάσετε την αντίστοιχη σελίδα PHP για την “υποδοχή” των δεδομένων που συμπληρώνει ο χρήστης στα προαναφερθέντα στοιχεία εισόδου.

Οδηγίες

Στο σημερινό εργαστήριο θα δουλέψετε με μία και μοναδική σελίδα php, η οποία θα περιέχει ταυτόχρονα τη φόρμα HTML και τον κώδικα PHP για τον χειρισμό των δεδομένων του χρήστη.

Θα πρέπει να φτιάξετε μια φόρμα HTML για την εκτέλεση των τεσσάρων βασικών πράξεων (πρόσθεση, αφαίρεση, πολλαπλασιασμός και διαίρεση) μεταξύ δύο αριθμών float. Τα πεδία της φόρμας HTML θα πρέπει να εμφανίζονται όπως στο παρακάτω σχήμα:



⇒ 1. Διαβάστε το **Παράρτημα A**, για τις πρακτικές λεπτομέρειες στη χρήση μιας μοναδικής σελίδας που περιέχει και τη φόρμα HTML και τον κώδικα PHP.

⇒ 2. Διαβάστε το **Παράρτημα B** για μια σύντομη εισαγωγή στις συναρτήσεις στην PHP.

⇒ 3. Ως πρώτο βήμα, κατασκευάστε συνάρτηση `printfm($a,$b,$op,$result)`, η οποία θα “τυπώνει” τη φόρμα HTML στη σελίδα σας. Οι παράμετροι θα είναι:

- `$a`, `$b`: αριθμοί που θα εμφανιστούν προ-συμπληρωμένοι στα πεδία A και B της φόρμας. Αν κάποιο από τα `a`, `b` ισούται με `NULL`, δεν θα εμφανίζεται η αντίστοιχη προ-συμπληρωμένη τιμή. **Χρησιμοποιήστε τη συνάρτηση `is_null($var)` για να ελέγξετε αν μια μεταβλητή ισούται με `NULL`.**
- `$op`: η πράξη που θα φανεί προ-επιλεγμένη στη φόρμα. Αν ισούται με `NULL`, δεν θα υπάρχει προ-επιλεγμένη πράξη.
- `$result`: αν είναι διαφορετική από `NULL`, θα εμφανίζεται στα δεξιά του κουμπιού =.

Προσοχή! Η συνάρτηση `printfm()` δεν εκτελεί καμία αριθμητική πράξη· απλά τυπώνει τη φόρμα HTML!

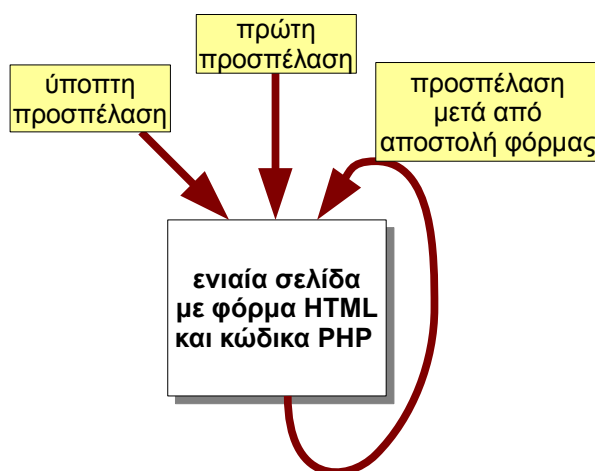
Πριν προχωρήσετε στο επόμενο βήμα, **βεβαιωθείτε ότι η συνάρτησή σας εκτελείται σωστά**, καλώντας την δοκιμαστικά με διάφορους συνδυασμούς παραμέτρων.

⇒ 4. Συνεχίστε στην κατασκευή του κυρίως προγράμματός σας:

1. Ελέγξτε αν υπάρχουν ως είσοδοι τα στοιχεία φόρμας “πράξη”, “α” και “β”.
 - Αν όχι, είτε μπαίνετε για πρώτη φορά στη σελίδα, είτε αντιμετωπίζετε “ύποπτη” προσπέλαση. Σε κάθε περίπτωση, τυπώστε απλά την κενή φόρμα.
 - Αν υπάρχει η πράξη, έχει μία από τις επιτρεπτές τιμές (από αυτές δηλαδή που στέλνει η φόρμα σας); Δείτε το **Παράρτημα Γ**, για το πώς μπορεί η PHP να σας βοηθήσει σε αυτόν τον έλεγχο.
 - Αν όχι, αντιμετωπίζετε “ύποπτη” προσπέλαση. Τυπώστε απλά την κενή φόρμα.
2. Εφόσον η “πράξη” υπάρχει και έχει αποδεκτή τιμή, ελέγξτε αν τα “α” και “β” έχουν μήκος > 0.
 - Αν όχι, ο χρήστης (μάλλον) ξέχασε να το/τα συμπληρώσει. Τυπώστε ένα μήνυμα λάθους και τη φόρμα με όσα στοιχεία υπάρχουν.
 - Αν ναι, εκτελέστε την πράξη (προσοχή στην περίπτωση διαίρεσης δια 0!) και τυπώστε τη φόρμα με τα στοιχεία συμπληρωμένα και με το αποτέλεσμα στα δεξιά του =.

Παράρτημα Α: Ενιαία σελίδα με φόρμα HTML και κώδικα PHP για χειρισμό δεδομένων χρήστη.

Η τιμή της ιδιότητας **action** της φόρμας HTML ισούται με τη διεύθυνση της ίδιας της σελίδας (η σελίδα “καλεί” τον εαυτό της). Στην ενιαία αυτή σελίδα, μπορείτε να “μπείτε” με 3 διαφορετικούς τρόπους (σχήμα 1):



Σχήμα 1

- **Είσοδος για πρώτη φορά:** ο χρήστης μπαίνει στην ιστοσελίδα ακολουθώντας ένα link ή πληκτρολογώντας απευθείας τη διεύθυνση URL. Στην περίπτωση αυτή δεν υπάρχουν καθόλου δεδομένα φόρμας.
- **Είσοδος μετά από αποστολή (submit) της φόρμας:** εδώ ο χρήστης βρισκόταν ήδη στη σελίδα και, αφού συμπλήρωσε τη φόρμα, πάτησε το

κουμπί αποστολής. Στην περίπτωση αυτή, η σελίδα ξαναφορτώνεται και **επιπλέον υπάρχουν τα δεδομένα της φόρμας που είχε συμπληρώσει ο χρήστης.**

- **“Υποπτη” είσοδος:** εδώ τα δεδομένα μπορεί να είναι “περίεργα”: **να λείπουν ή να έχουν τιμές που δεν προβλέπει η φόρμα.**

Θα πρέπει να καλύψετε όλες τις πιο πάνω πιθανότητες στον κώδικά σας!

Πρακτικά: τι βάζω ως τιμή στην ιδιότητα action;

Η PHP διαθέτει τη μεταβλητή `$_SERVER['SCRIPT_NAME']`, η οποία ισούται με τη διεύθυνση της σελίδας. Μπορείτε να τη χρησιμοποιήσετε στην ιδιότητα **action** της φόρμας:

```
<form action="<?php echo $_SERVER['SCRIPT_NAME']; ?>" ..κλπ..>
```

Έτσι, δεν χρειάζεται να κάνετε αλλαγές στη φόρμα σας αν μετονομάσετε ή αλλάξετε θέση στη σελίδα!



Προσοχή!! Σε πολλά παραδείγματα που θα βρείτε on-line ή σε βιβλία, χρησιμοποιείται η μεταβλητή `$_SERVER['PHP_SELF']` για τον ίδιο σκοπό. **Μην τη χρησιμοποιείτε, καθώς έχει προβλήματα ασφάλειας!!!**

Παράρτημα Β: Οι συναρτήσεις στην PHP (μικρή εισαγωγή).

Συναρτήσεις στην PHP δηλώνετε οπουδήποτε (πριν τις καλέσετε όμως!) ως εξής:

```
function funcname($param1, $param2, ...) {  
    // περιεχόμενο function  
}
```

- Εξ'ορισμού, οι παράμετροι της συνάρτησης περνούν ως τιμές (by value) (υπάρχει δυνατότητα για πέρασμα με αναφορά (by reference), αλλά δεν θα την καλύψουμε εδώ).
- Το ίδιο ισχύει για την επιστρεφόμενη τιμή (εξ'ορισμού by value).
- Εμβέλεια (scope) μεταβλητών: ως συνήθως, οι τοπικές (local) μεταβλητές μιας συνάρτησης δεν μπορούν να προσπελαστούν από κώδικα έξω από τη συνάρτηση.

Μια σημαντική διαφορά με την C: στην PHP **δεν μπορείτε μέσα στη συνάρτηση να δείτε global μεταβλητές** (μεταβλητές που έχουν οριστεί έξω από συναρτήσεις), παρά μόνο αν τις δηλώσετε (μέσα στη συνάρτηση) ως εξής:

```
function funcname () {  
    global $a, $b;
```

```
// περιεχόμενο function  
}
```

Παράρτημα Γ: Δημιουργία και έλεγχος πεδίων φόρμας μέσω PHP.

Συχνά τα πεδία επιλογής (radiobuttons, checkboxes, selects) σε μια φόρμα πρέπει να εμφανίζουν μια μακρά λίστα επιλογών. Μέσω των arrays της PHP και με τη χρήση loops μπορούμε να “αυτοματοποιήσουμε” τις διαδικασίες δημιουργίας και ελέγχου για τα πεδία αυτά.

Δημιουργία πεδίων επιλογής

Τοποθετήστε τις επιλογές σε ένα **array**, με κλειδί την τιμή που στέλνει κάθε πεδίο και τιμή το κείμενο που θα δει ο χρήστης, π.χ.:

```
$arr = array("opt1" => "Option 1", "opt2" => "Option 2",  
            "opt3" => "Option 3", "opt4" => "Option 4",  
            "opt5" => "Option 5");
```

Στη συνέχεια, μπορείτε να δημιουργήσετε π.χ. ένα στοιχείο **select** ως εξής:

```
<select name="selopt">  
<?php  
    foreach ($arr as $key => $val) {  
        echo '<option value="'. $key. '">' . $val. '</option>';  
    }  
>  
</select>
```

Έλεγχος τιμών εισόδου

Αν υπάρχει ο προηγούμενος πίνακας με τις πιθανές τιμές του στοιχείου εισόδου, τότε μπορείτε με τη βοήθεια της συνάρτησης `array_key_exists(key, arr)` να ελέγξετε αν αυτό που λαμβάνετε ανήκει στις επιτρεπτές τιμές:

```
if (isset($_GET['selopt']) &&  
    array_key_exists($_GET['selopt'], $arr)) {  
    echo '<p>valid input!</p>';  
}
```