

Θέματα Προγραμματισμού Διαδικτύου ~ Sessions & Some PHP more ~

Στελιος Σφακιανάκης
Εαρινό 2019



Αυτή η εργασία χορηγείται με άδεια Creative Commons
Αναφορά Δημιουργού - Μη Εμπορική Χρήση - Παρόμοια Διανομή



Short echo tags

```
<p>Σήμερα είναι <?=date('d/m/y')?></p>
```

Το ίδιο:

```
<p>Σήμερα είναι <?php echo date('d/m/y')?></p>
```

- Ό,τι είναι μέσα σε `<?= ... ?>` υπολογίζεται και εκτυπώνεται η τιμή του
 - σαν να κάναμε `echo`
- Δεν χρειάζεται ερωτηματικό (;) στο τέλος



"Superglobals"

- Μεταβλητές που είναι διαθέσιμες οπουδήποτε:
 - **\$_SERVER** : πληροφορίες για τον Server και το περιβάλλον εκτέλεσης
 - Π.χ. το **\$_SERVER["REQUEST_METHOD"]** δίνει την HTTP μέθοδο (π.χ. **"GET"** ή **"POST"**)
 - **\$_GET** : μεταβλητές και τιμές του GET HTTP request (URL παράμετροι, δηλ. ότι είναι μετά το ? στο URL)
 - **\$_POST** : μεταβλητές και τιμές του POST HTTP request (συνήθως δεδομένα που στέλνονται με φόρμες που έχουν **method="POST"**)
 - **\$_SESSION** : μεταβλητές και τιμές που βάζουμε στο Session
 - **\$_REQUEST** : μεταβλητές και τιμές του HTTP request (είτε είναι GET ή POST)
 - **\$_FILES** : πληροφορίες για uploaded files



Σταθερές

- Χρησιμοποιούμε το **define** για να ορίσουμε μεταβλητές που δεν αλλάζουν τιμή (σταθερές)
- Δεν χρησιμοποιούν το **\$** στο όνομα τους
- Οι σταθερές αυτές έχουν global scope (δηλ. θα είναι διαθέσιμες οπουδήποτε στο κώδικα μας).
- Παράδειγμα:

```
// Connect to MySQL:  
  
define(DBHOST, "127.0.0.1");  
define(DBUSER, "root");  
define(DBPASSWORD, "my-secret-pw");  
define(DBNAME, "lab3");  
  
$mysqli = new mysqli(DBHOST, DBUSER, DBPASSWORD, DBNAME);
```



Έλεγχος για **null**, **false**, ή undefined τιμές

- **isset(\$var)**: τσεκάρει αν η μεταβλητή **\$var** έχει τιμή και η τιμή του δεν είναι **null**
- Π.χ. **isset(\$_GET["v"])**: ελέγχει αν το GET request περιέχει μια παράμετρο **"v"**
- **unset(\$var)** : "καταστρέφει" / "εξαφανίζει" μια μεταβλητή
- **is_null(\$var)** ή **\$var === null** : ελέγχει αν η μεταβλητή είναι **null**



mysql design

- Χρησιμοποιείτε **VARCHAR** (variable length characters) για strings με κυμαινόμενο μήκος
- **VARCHAR(50)**: string με το πολύ 50 χαρακτήρες, strings με πιο μεγάλο μήκος θα "κοπούν" στους 50 χαρακτήρες
- **CHAR(10)**: string ακριβώς 10 χαρακτήρες, αν κάνουμε INSERT λιγότερους τότε η MySQL προσθέτει κενά (' ') στο τέλος..



Validation & Sanitisation

- **Validation:** επιβεβαιώνει ότι το input είναι πράγματι αυτό που περιμένουμε. Π.χ. τσεκάρει αν ένα string είναι πράγματι έγκυρη διεύθυνση email:
 - `filter_var($v, FILTER_VALIDATE_EMAIL)`
 - Άλλα validation filters: <https://secure.php.net/filter.filters.validate>
- **Sanitization:** Αφαιρεί (ή κάνει "escape") χαρακτήρες που δεν επιτρέπονται στο input
 - `filter_var($v, FILTER_SANITIZE_NUMBER_INT)` : αφαιρεί όλους τους χαρακτήρες που δεν είναι ψηφία, 'συν', ή 'μείον'
-



Include και Require

- Πολλές φορές είναι προτιμότερο να έχουμε τον κώδικα μας μοιρασμένο σε περισσότερα του ενός αρχεία.
- Π.χ. μπορούμε να έχουμε τις συναρτήσεις μας σε ένα *functions.php* αρχείο το οποίο να (επανα)χρησιμοποιείται από άλλα αρχεία PHP
- Με την εντολή **include 'functions.php'** μπορούμε να ενσωματώσουμε τις εντολές του *functions.php* σε άλλο αρχείο
- Το **require** είναι όπως το **include** αλλά αν αποτύχει τότε τερματίζεται το script!



header.php

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
```

```
<style>
```

```
nav {
```

```
background-color: #c0c0c0;
```

```
overflow: hidden;
```

```
}
```

```
nav a {
```

```
float: left;
```

```
color: #f2f2f2;
```

```
text-align: center;
```

```
padding: 14px 16px;
```

```
text-decoration: none;
```

```
}
```

```
nav a.selected {
```

```
background-color: #4CAF50;
```

```
color: white;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<nav>
```

```
<a <?php if ($selected == 'home'){ ?>class="selected" <?php } ?> href="home.php">Αρχική</a>
```

```
<a <?php if ($selected == 'contact'):?>class="selected" <?php endif ?> href="contact.php">Επικοινωνία</a>
```

```
<a <?php if ($selected == 'login'):?>class="selected" <?php endif ?> href="login.php">Είσοδος</a>
```

```
</nav>
```

Αρχική

Επικοινωνία

Είσοδος



footer.php

```
<footer>&copy; 2019 – ΘΠΔ</footer>  
</body>  
</html>
```



home.php

```
<?php  
$selected = 'home';  
  
include('header.php');  
  
echo "<h3>Home</h3>";  
  
include('footer.php');  
  
?>
```



Sessions

- Το HTTP πρωτόκολλο είναι stateless ("δεν έχει μνήμη"). Αυτό σημαίνει ότι εάν έχω μια μεταβλητή που την υπολογίζω σε μια ιστοσελίδα αυτή δεν μπορεί να μεταφέρει την τιμή της σε άλλη ιστοσελίδα. Στην πραγματικότητα σε κάθε POST ή GET χάνονται όλοι οι υπολογισμοί και όλα τα δεδομένα εισόδου που είχαν γίνει στην προηγούμενη σελίδα.
- Υπάρχουν όμως εφαρμογές όπου αυτή η ιδιότητα της διαγραφής της κατάστασης μετά το κλείσιμο της ιστοσελίδας δεν είναι «βολική». Παράδειγμα είναι η διαδικασία authentication. Ο χρήστης θέλει να κάνει login σε μια εφαρμογή και το σύστημα να διατηρεί την κατάσταση αυτή (να τον «θυμάται») όσο αυτός επιθυμεί να παραμείνει στην εφαρμογή αυτή. Να τον ξεχάσει μόνο όταν ο χρήστης εγκαταλείψει την εφαρμογή ή κλείσει τον browser. Αυτή η επιθυμία των χρηστών δεν υποστηρίζεται από το HTTP. Σύμφωνα λοιπόν με το HTTP ο χρήστης θα πρέπει να κάνει login σε κάθε σελίδα της εφαρμογής που θέλει να δει!!!!
- Το πρόβλημα αυτό λύνεται με την χρήση του **Session** (συνεδρία) που δημιουργεί η γλώσσα PHP (και γενικά όλες οι γλώσσες προγραμματισμού)



Αρχιτεκτονική Session

- Τα session λειτουργούν με τη δημιουργία ενός μοναδικού αναγνωριστικού (Unique ID - UID) για κάθε επισκέπτη.
(Δημιουργείται μια συνεδρία μεταξύ server και browser)
- Οι μεταβλητές που θα πρέπει να θυμάται το session σώζονται στον server συσχετιζόμενες στο παρόν UID. Το UID αποθηκεύονται σε ένα **cookie** ή μεταφέρονται στη διεύθυνση URL.
- Γνωρίζοντας το UID μπορώ να έχω πρόσβαση στα περιεχόμενα του session.



Sessions στην PHP

- Έναρξη ενός session: `session_start()`
 - Πρέπει να είναι η πρώτη - πρώτη εντολή στο PHP αρχείο!
(Πριν από HTML και οτιδήποτε άλλο!)
- Κατάργηση ενός session: `session_destroy()`
- Αποθήκευση και ανάκτηση πληροφοριών από το session μέσω του super global `$_SESSION`
 - `$_SESSION['user'] = 'Stelios'`
 - `echo $_SESSION['user']`



Παράδειγμα Sessions

- Μελετήστε το παράδειγμα που θα βρείτε στο <https://eclass.teicrete.gr/modules/document/file.php/TP327/ΘΕΩΡΙΑ/php-sessions-example.zip>