

# Server side scripting languages

## 3. Μεταβλητές, Σταθερές και Τύποι δεδομένων στη PHP



**php**



Β' Τεχνική Σχολή Λεμεσού  
Σχολική χρονιά 2019 – 2020

Άριστος Πασιάς

# Μεταβλητές στη PHP



Οι μεταβλητές χρησιμοποιούνται για αποθήκευση τιμών, όπως αλφαριθμητικά, αριθμοί ή πίνακες. Όλες οι μεταβλητές στην PHP ξεκινούν με το σύμβολο **\$**. Ο σωστός τρόπος για να ορίσουμε μια μεταβλητή είναι:

```
$var_name = value ;
```

```
$myAge = 32
```

- Σε αντίθεση με τα ονόματα των εντολών, συναρτήσεων κλπ, τα ονόματα των μεταβλητών είναι **case-sensitive**
- Μια μεταβλητή δεν χρειάζεται να δηλωθεί πριν χρησιμοποιηθεί.
- Δεν χρειάζεται να δηλώσουμε τον τύπο της μεταβλητής.
- Οι τύποι δεδομένων που υποστηρίζονται από την PHP είναι οι εξής: String, Integer, Float (λέγεται και double), Boolean, Array, Object, NULL και Resource.
- Πολλές έτοιμες συναρτήσεις διαχείρισης μεταβλητών είναι ενσωματωμένες στη PHP.

# Η εντολές εξόδου echo και Print

Η εντολή echo χρησιμοποιείται για να εμφανίζουμε πληροφορίες στην οθόνη.

```
<?php
$txt1 = "Learn PHP";
$txt2 = "W3Schools.com";
$x = 5;
$y = 4;
echo "<h2>" . $txt1 . "</h2>";
echo "Study PHP at " . $txt2 .
    "<br>";
echo $x + $y;
?>
```

**Learn PHP**

Study PHP at [W3Schools.com](http://W3Schools.com)

# Η εντολές εξόδου echo και Print

Η εντολή print επίσης χρησιμοποιείται για να εμφανίζουμε πληροφορίες στην οθόνη.

```
<?php
$txt1 = "Learn PHP";
$txt2 = "W3Schools.com";
$x = 5;
$y = 4;

print "<h2>" . $txt1 . "</h2>";
print "Study PHP at " . $txt2 .
    "<br>";
print $x + $y;
?>
```

**Learn PHP**

Study PHP at [W3Schools.com](http://W3Schools.com)

# Ειδικοί χαρακτήρες

Ειδικός Χαρακτήρας	Νόημα (μόνο σε “ “ όχι σε ‘ ‘
<code>\n</code>	Νέα γραμμή (newline)
<code>\r</code>	carriage
<code>\t</code>	Οριζόντιο tab (στηλοθέτης)
<code>\\</code>	Χαρακτήρας backslash
<code>\\$</code>	Σύμβολο του δολαρίου
<code>\"</code>	Διπλά εισαγωγικά
<code>\[0-7]{1,3}</code>	Η σειρά των χαρακτήρων που ταιριάζει με την κανονική έκφραση είναι ένας χαρακτήρας του 8δικού συστήματος (octal notation)
<code>\x[0-9A-Fa-f]{1,2}</code>	Η σειρά των χαρακτήρων που ταιριάζει με την κανονική έκφραση είναι ένας χαρακτήρας του 16δικού συστήματος (hexadecimal notation)

# Άσκηση 1



Στον παρακάτω κώδικα δημιουργήστε δύο μεταβλητές με το όνομα \$myName και \$myAge και εκχωρήστε σε αυτές το όνομά σας και την ηλικία σας αντίστοιχα. Εμφανίστε την τιμή των παραπάνω μεταβλητών χρησιμοποιώντας την εντολή echo.

```
1. <html>
2.   <head>
3.     </head>
4.     <body>
5.       <p>
6.         <?php
7.         .....
8.         ?>
9.       </p>
10.    </body>
11. </html>
```

# Άσκηση 2



Εκτελέσετε τον παρακάτω κώδικα για να μάθετε το όνομα και το τηλέφωνο που έχει ο καθηγητής σας. Στη συνέχεια να αλλάξετε το όνομα και το τηλέφωνο με τα δικά σας. Τι καινούριο μάθατε από την εκτέλεση αυτού του κώδικα;

1. `<?php`
2. `$identity = 'Άριστος Πασιάς';`
3. `$telephone= 'Galaxy A7';`
4. `$sentence = "Ο $identity έχει ένα $telephone";`
5. `echo $sentence;`
6. `?>`

# Άσκηση 3



Εκτελέσετε τον παρακάτω κώδικα για να μάθετε δυο τρόπους να γράψετε την ίδια εντολή.

```
1. <?php
2. // αυτό...
3. echo "Hello, world!";
4. // ... είναι ίδιο με αυτό
5. echo "Hello," . " " . "world" .
   "!!";
6. ?>
```

# Εύρος μεταβλητής

- Στην PHP, οι μεταβλητές μπορούν να δηλωθούν οπουδήποτε μέσα στο Script.
- Το εύρος μιας μεταβλητής είναι το μέρος του Script όπου η μεταβλητή μπορεί να χρησιμοποιηθεί.
- Η PHP έχει τρία διαφορετικά όρια εύρους των μεταβλητών:
  - **local** (τοπικό)
  - **Global** (Καθολικό)
  - **Static** (στατικό)

# Μεταβλητή εύρους Global

Μια μεταβλητή που δηλώνεται έξω από μια συνάρτηση έχει εύρος GLOBAL και μπορεί να προσπελαστεί μόνο έξω από τη συνάρτηση.

```
<?php
$x = 5; // global scope

function myTest() {
    // Η χρήση της x μέσα στη συνάρτηση δεν θα δώσει
    // το σωστό αποτέλεσμα
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

echo "<p>Variable x outside function is: $x</p>";
?>
```

Variable x inside function is:  
Variable x outside function is: 5

# Μεταβλητή εύρους Local

Μια μεταβλητή που δηλώνεται μέσα σε μια συνάρτηση έχει εύρος LOCAL και μπορεί να προσπελαστεί μόνο μέσα σε αυτή τη συνάρτηση.

```
<?php
function myTest() {
    $x = 5; // local scope
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

// Η χρήση της x μέσα στη συνάρτηση δεν θα δώσει
το σωστό αποτέλεσμα
echo "<p>Variable x outside function is: $x</p>";
?>
```

Variable x inside function is: 5  
Variable x outside function is:

Μπορείτε να έχετε τοπικές μεταβλητές με το ίδιο όνομα σε διάφορες λειτουργίες, επειδή οι τοπικές μεταβλητές αναγνωρίζονται μόνο στη συνάρτηση στην οποία δηλώνονται.

# Global keyword

Μπορούμε να έχουμε πρόσβαση σε μια μεταβλητή Global μέσα από μια συνάρτηση χρησιμοποιώντας το keyword Global πριν από τη μεταβλητή:

```
<?php
$x = 5;
$y = 10;

function myTest() {
    global $x, $y;
    $y = $x + $y;
}

myTest();
echo $y; // εμφανίζει 15
?>
```

15

# Πίνακας \$GLOBALS[index]

Η PHP αποθηκεύει όλες τις μεταβλητές Global σε έναν πίνακα που ονομάζεται \$GLOBALS [index]. Ο δείκτης περιέχει το όνομα της μεταβλητής. Αυτός ο πίνακας είναι επίσης προσβάσιμος από τις συναρτήσεις και μπορεί να χρησιμοποιηθεί για την άμεση ενημέρωση των μεταβλητών.

```
<?php
    $x = 5;
    $y = 10;
    $z = 0;
function myTest() {
    $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}

myTest();
echo $y; // εμφανίζει 15
?>
```

# Static keyword

Κανονικά, όταν ολοκληρώνεται η εκτέλεση μιας συνάρτησης, όλες οι μεταβλητές της διαγράφονται. Αν θέλουμε μια Local μεταβλητή να μην διαγραφεί, χρησιμοποιούμε το keyword Static όταν δηλώνουμε για πρώτη φορά τη μεταβλητή:

```
<?php
function myTest() {
    static $x = 0;
    echo $x;
    $x++;
}
```

```
myTest();
myTest();
myTest();
?>
```

```
0
1
2
```

# Τύποι δεδομένων στη PHP

- String
- Integer
- Float ή (double)
- Boolean
- Array
- Object
- NULL
- Resource

- `$a = 'Βαρέθηκα πάλι';`
- `$b = 10 (var_dump($b);`
- `$c = 10.55`
- `$d = True`
- `$cars =  
array("Volvo"."BMW"."...");`
- Object
- NULL
- Resource

# PHP String Functions

Η PHP διαθέτει πολλές ενσωματωμένες συναρτήσεις με τις οποίες μπορούμε να διαχειριστούμε μεταβλητές τύπου String. Μερικές από αυτές:

- **strlen ()** επιστρέφει το μήκος μιας συμβολοσειράς.  
`echo strlen("Ο Χάρης είναι Ροκ!"); //επιστρέφει 18`
- **str\_word\_count ()** μετράει τον αριθμό των λέξεων σε μια συμβολοσειρά:  
`echo str_word_count("Είμαι άνω κάτω!"); // επιστρέφει 3`
- **strrev ()** αντιστρέφει μια συμβολοσειρά:  
`echo strrev("Καλή όρεξη!"); //επιστρέφει !ηξερό ήλακ`
- **strpos ()** αναζητά ένα συγκεκριμένο κείμενο μέσα σε μια συμβολοσειρά  
`echo strpos("Καλά Χριστούγεννα!", «γεννα»); // επιστρέφει 12`
- **str\_replace ()** αντικαθιστά μερικούς χαρακτήρες με άλλους σε μια συμβολοσειρά.  
`echo str_replace(«μέρα», «νύχτα», «Καλημέρα!»); // επιστρέφει Καληνύχτα`
- **strcmp()** – Συγκρίνει δύο αλφαριθμητικά

Για περισσότερες συναρτήσεις δείτε τον επίσημο οδηγό της PHP

<https://www.php.net/manual/en/ref.strings.php>

# Άσκηση



Να γράψετε τις απαραίτητες εντολές σε PHP με τις οποίες θα εκχωρείτε σε τρεις μεταβλητές το όνομα της ομάδας σας, τον αριθμό των νικών και των ισοπαλιών που πέτυχε. Τέλος να εμφανίζει ένα μήνυμα:

**«Άρης Λεμεσού 5N 6I 21B»**

```
1. <?php
2. $omada="Άρης Λεμεσού";
3. $nikes=5;
4. $isop=6;
5. $vathmi=($nikes*3)+$isop;
6. echo "$omada"." ".$nikes"."N"."
   "$isop"."I"." ".$vathmi"."B";
7. ?>
```

# PHP Constants (Σταθερές)

Οι σταθερές είναι σαν τις μεταβλητές εκτός από το ότι όταν ορισθούν δεν μπορούν να αλλάξουν τιμή κατά τη διάρκεια του Script.

Το όνομα μιας σταθεράς ξεκινά με ένα γράμμα (χωρίς τη χρήση του \$)

Σε αντίθεση με τις μεταβλητές, οι σταθερές ορίζονται αυτόματα ως Global σε ολόκληρο το script.

Για να ορίσετε μια σταθερά, χρησιμοποιούμε τη συνάρτηση **define ()**.

Παράδειγμα:

```
<?php
    define("Fellure", "Λυπάμαι χάσατε! προσπαθήστε ξανά!");
    echo Fellure;
?>
```

Γενική μορφή:

```
define(name,value,case-insensitive);

define("URL", "http://www.pashias.net", True);
```

Constants are Global

# Ασκήσεις

1. Ποια από τα ακόλουθα ονόματα μεταβλητών δεν είναι έγκυρα;
  - a. `$this_is_a_value`
  - b. `$_test_`
  - c. `$metritis`
  - d. `$the counter`
  - e. `$my-name`
  - f. `$count`
  - g. `$2test`
2. Ποιο είναι το αποτέλεσμα του ακόλουθου αποσπάσματος;
  1. `$test_val=6.5466;`
  2. `settype($test_val, "integer");`
  3. `echo $test_val;`
3. Ποιο θα είναι το αποτέλεσμα της ακόλουθης εντολής;
  1. `echo gettype("8");`