

Server side scripting languages

6 Συναρτήσεις στη PHP



php



Β' Τεχνική Σχολή Λεμεσού
Σχολική χρονιά 2019 – 2020

Άριστος Πασιάς

Συναρτήσεις στη PHP

- Συναρτήσεις είναι τμήματα κώδικα που εκτελούν μια συγκεκριμένη εργασία και μπορούν να επαναλαμβάνονται πολλές φορές μέσα σε ένα πρόγραμμα.
- Η PHP έχει μια τεράστια συλλογή από εσωτερικές ή ενσωματωμένες συναρτήσεις που μπορείτε να καλέσετε απευθείας μέσα από το σενάριο PHP για να εκτελέσετε μια συγκεκριμένη εργασία

Ενσωματωμένες Συναρτήσεις παραδείγματα (Μαθηματικές)

- Μέγιστος και Ελάχιστος αριθμός
`min(arg1, arg2, ...), max(arg1, arg2, ...)`
- Τυχαίος ακέραιος αριθμός
– Π.χ. για τυχαία επιλογή διαφημιστικού banner
`rand([min, max])`
- Μορφοποίηση
`number_format (αριθμός, δεκαδικά, υποδιαστολή, χιλιάδες)`
– μόνο η πρώτη παράμετρος υποχρεωτική.

Παράδειγμα

```
$p = 12345.67;  
number_format($p, 2, ".", ","); //returns 12,345.67
```

Ενσωματωμένες Συναρτήσεις παραδείγματα (Μαθηματικές)

- **abs(value)** – απόλυτη τιμή αριθμού

```
$x=abs(-5);
```

```
$y=abs(42);
```

```
echo "x=$x y=$y"; // x=5 y=42
```

- **sqrt(value)** – τετραγωνική ρίζα αριθμού

```
$x=sqrt(25);
```

```
echo "x=$x"; // x=5
```

- **is_numeric(value)** – έλεγχος για αριθμό

```
if (is_numeric($input))
```

```
    echo "Got Valid Number=$input";
```

```
else
```

```
    echo "Not Valid Number=$input";
```

Ενσωματωμένες Συναρτήσεις παραδείγματα (Μαθηματικές)

- **round(value,[dec])** – στρογγυλοποίηση

```
$x=round (-5.456);  
$y=round (3.7342, 2);  
print "x=$x y=$y"; // x=-5 y=3.73
```
- **ceil(value)** – στρογγυλοποίηση προς τα πάνω

```
$x=ceil (3.7342);  
print "x=$x"; // x=4
```
- **floor(value)** – στρογγυλοποίηση προς τα κάτω

```
$x=floor (-5.456);  
print "x=$x"; // x=-6
```

Ενσωματωμένες Συναρτήσεις παραδείγματα (Αλφαριθμητικών)

- `strlen(string)`, μήκος αλφαριθμητικού

```
$comments = "Good Job";  
$len = strlen($comments);  
echo "Length: $len); // Length: 8
```
- `trim(string)`, αφαίρεση κενών δεξιά και αριστερά

```
$name_withspace = " Joe Jackson ";  
$name = trim($name_withspace);  
echo "Name: $name"; // Name: Joe Jackson
```

Ενσωματωμένες Συναρτήσεις παραδείγματα (Αλφαριθμητικών)

- `strtolower(string)`, `strtoupper(string)`, πεζά, κεφαλαία

```
$inquote = "Now Is The Time";  
$lower = strtolower($inquote);  
$upper = strtoupper($inquote);  
echo "lower:$lower"; // now is the time  
echo "upper:$upper"; // NOW IS THE TIME
```

- `substr(string, start, [end])`, τμήμα αλφαριθμητικού

```
$date = "31 October 2007";  
$day = substr($date, 0, 2); // "31"  
$month = substr($date, 3, 7); // "October"  
$year = substr($date, 11); // "year"  
echo "Day:$day, Month:$month, Year:$year";
```

Ορισμός Συναρτήσεων

PHP User Defined Functions



Μια συνάρτηση είναι ένα μπλοκ εντολών που μπορεί να χρησιμοποιηθεί επανειλημμένα σε ένα πρόγραμμα. Δεν εκτελείται αυτόματα όταν φορτώνεται μια σελίδα αλλά όταν γίνει μια κλήση προς τη συνάρτηση. Τα ονόματα των συναρτήσεων ξεκινούν πάντα με γράμμα και δεν είναι case sensitive.

```
function functionName() {  
    code to be executed;  
}
```

```
<?php  
function writeMsg() {  
    echo "You are Wellcome!";  
}  
writeMsg(); // call the function  
?>
```

Ορίσματα Συναρτήσεων



Σε μια συνάρτηση είναι δυνατόν να μεταβιβαστούν δεδομένα μέσω των ορισμάτων. Ένα όρισμα είναι ακριβώς όπως μια μεταβλητή. Καθορίζονται μέσα σε παρενθέσεις, μετά το όνομα της συνάρτησης. Μπορείτε να προσθέσετε όσα ορίσματα θέλετε, απλά διαχωρίζοντας τα με κόμμα.

```
<?php
function classgrade($name, $grade) {
    echo "Μπράβο. $name. πέτυχες με. $grade <br>";
}

classgrate(«Ανδρέα», "19");
classgrate(«Σπύρο», "18");
classgrate(«Πάνο», «20»);
?>
```

Προεπιλεγμένη τιμή Ορίσματος



```
<?php
function setHeight($minheight = 50) {
    echo "The height is : $minheight <br>"; }
setHeight(350);
setHeight(); // Θα πάρει την προκαθορισμένη τιμή 50
setHeight(135);
?>
```

Επιστροφή τιμής - εντολή Return

```
<?php
function sum($x, $y) {
    $z = $x + $y;
    return $z; }
echo "5 + 10 = " . sum(5, 10) . "<br>";
echo "7 + 13 = " . sum(7, 13) . "<br>";
echo "2 + 4 = " . sum(2, 4);
?>
//Εναλλακτικά $r=sum(2,4)
```

Προεπιλεγμένη τιμή Ορίσματος και Style Παράδειγμα



```
<?php // Defining function
function customFont($font, $size=1.5){
echo "<p style=\"font-family: $font; font-size:
{$size}em;\">Hello, world!</p>";
}
// Calling function
customFont("Arial", 2);
customFont("Times", 3);
customFont("Courier");
?>
```

▶ Hello, world!

Hello, world!

Hello, world!

Passing Arguments to a Function by Reference &



```
<?php
/* Defining a function that multiply a number by
itself and return the new value */
function selfMultiply(&$number){
    $number *= $number;
    return $number;
}
$mynum = 5;
echo $mynum; // Outputs: 5
selfMultiply($mynum);
echo $mynum; // Outputs: 25
?>
```

Τύπος συνάρτησης στη PHP



Δεν χρειάζεται να δηλώσουμε στη PHP τον τύπο των δεδομένων μιας μεταβλητής. Η PHP συσχετίζει αυτόματα έναν τύπο δεδομένων με τη μεταβλητή, ανάλογα με την τιμή τους. Οπότε, μπορείτε να εκχωρήσετε μια συμβολοσειρά σε μια μεταβλητή που η τιμή της ήταν ένας ακέραιος αριθμός χωρίς να προκαλέσετε σφάλματα.

Στην **PHP 7**, προστέθηκε η **δυνατότητα** δήλωσης τύπου. Μπορούμε να καθορίσουμε τον τύπο όταν δηλώνουμε μια συνάρτηση και να επιτρέψουμε την αυστηρή απαίτηση, να πετάξει ένα "σφάλμα" σε μια αναντιστοιχία τύπου.

```
<?php // or ..<?php declare(strict_types=1);  
    //strict requirement  
function addNums(int $a, int $b) {  
    return $a + $b;  
}  
echo addNumbers(5, "5 days");  
    // since strict is NOT enabled "5 days" is changed to  
    int(5), and it will return 10  
    // Error  
?>
```

Τύπος συνάρτησης στη PHP



Στην PHP 7 μπορούμε επίσης να δηλώσουμε τον τύπο της μεταβλητής που επιστρέφει η συνάρτηση όπως φαίνεται στο πιο κάτω παράδειγμα.

```
<?php declare(strict_types=1);  
// strict requirement  
function addNumbers(float $a, float $b) :  
    float {  
    return $a + $b;  
}  
echo addNumbers(1.2, 5.2);  
?>
```

Επαναληπτική συνάρτηση (Recursive Function)



Είναι μια συνάρτηση που καλείται ξανά και ξανά μέχρι να ικανοποιηθεί μια συνθήκη. Χρησιμοποιούνται συχνά για την επίλυση πολύπλοκων μαθηματικών υπολογισμών ή για την επεξεργασία πολυδιάστατων πινάκων.

```
<?php // Defining recursive function function printValues($arr) {
global $count; global $items; // Check input is an array
if(!is_array($arr)){ die("ERROR: Input is not an array"); } /*
Loop through array, if value is itself an array recursively call
the function else add the value found to the output items array,
and increment counter by 1 for each value found */ foreach($arr
as $a){ if(is_array($a)){ printValues($a); } else{ $items[] = $a;
$count++; } } // Return total count and values found in array
return array('total' => $count, 'values' => $items); } // Define
nested array $species = array( "birds" => array( "Eagle",
"Parrot", "Swan" ), "mammals" => array( "Human", "cat" => array(
"Lion", "Tiger", "Jaguar" ), "Elephant", "Monkey" ), "reptiles"
=> array( "snake" => array( "Cobra" => array( "King Cobra",
"Egyptian cobra" ), "Viper", "Anaconda" ), "Crocodile",
"Dinosaur" => array( "T-rex", "Alamosaurus" ) ) ); // Count and
print values in nested array $result = printValues($species);
echo $result['total'] . ' value(s) found: '; echo implode(', ',
$result['values']); ?>
```

Ημερομηνίες – η συνάρτηση date()



Η συνάρτηση date() μετατρέπει το χρόνο σε μια μορφή που είναι ευανάγνωστη στους ανθρώπους. Η σύνταξη της συνάρτησης PHP date() μπορεί να δοθεί με. Η παράμετρος format καθορίζει τη μορφή της επιστρεφόμενης ημερομηνίας και ώρας. Ωστόσο, η timestamp() είναι προαιρετική, αν δεν συμπεριληφθεί θα χρησιμοποιηθεί η τρέχουσα ημερομηνία και ώρα.

```
date(format, timestamp)
```

```
//Current server time
```

```
<?php
```

```
$today = date("d/m/Y"); // 26/10/2019
```

```
echo $today;
```

```
echo date("h:i:s") . "<br>"; //03:57:55
```

```
?>
```

Παράμετροι συνάρτησης date()



d – Ημέρα του μήνα; Δύο ψηφία (01 ως 31)
D – Ημέρα της εβδομάδας; Τρία γράμματα (Mon to Sun)
m – Μήνας σε αριθμούς; Δύο ψηφία (01 ως 12)
M – Μήνας με γράμματα; Τρία γράμματα (Jan ως Dec)
y – Έτος σε δύο ψηφία (08 ή 19)
Y – Έτος σε τέσσερα ψηφία (2008 or 2019)

h – Ώρες σε 12-ωρη μορφή (01 ως 12)
H – Ώρες σε 24-ωρη μορφή (00 to 23)
i – Λεπτά (00 to 59)
s – Δευτερόλεπτα (00 to 59)
a – Πεζοί χαρακτήρες ante meridiem and post meridiem (am or pm)
A – Κεφαλαία ante meridiem and post meridiem (AM or PM)

Παράμετροι συνάρτησης date()



```
<?php
$timestamp = time();
echo($timestamp);
echo(date("F d, Y h:i:s", $timestamp));
?>
```

1564150138

July 26, 2019 04:08:58

```
//Μετατροπή ημερομηνίας σε timestamp
mktime(hour, minute, second, month, day, year)
```