|  |  |
| --- | --- |
|  | <html>  <body>  <?php  echo "My first PHP script!<br>";  ?>  </body>  </html> |
|  | $x = 5 + 5;  echo $x;  echo "<br>"; |
|  | $color = "red";  echo "My car is ".$color."<br>"; |
|  | $x = 5;  $y = 4;  echo $x + $y; |
|  | echo "<h2>PHP is Fun!</h2>";  echo "Hello world!<br>"; |
|  | $txt1 = "Learn PHP";  $txt2 = "coding";  $x = 5;  $y = 4;  echo "<h2>" . $txt1 . "</h2>";  echo "Study PHP at " . $txt2 . "<br>";  echo $x + $y; |
|  | print "<h2>PHP is Fun!</h2>";  print "Hello world!<br>";  print "I'm about to learn PHP!"; |
|  | $txt1 = "Learn PHP";  $txt2 = "coding";  $x = 5;  $y = 4;  print "<h2>" . $txt1 . "</h2>";  print "Study PHP at " . $txt2 . "<br>";  print $x + $y; |
|  | <?php  $a=15;  $b=30;  $total=$a+$b;  print $total;  ?> |
|  |  |

**PHP:** Introduction to PHP, Features, sample code.

PHP script working, PHP syntax, conditions & Loops, Functions, String manipulation, Arrays & Functions.

Advance PHP Databases: Basic command with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables.

Comment

// This is a single-line comment  
  
# This is also a single-line comment

/\*  
This is a multiple-lines comment block  
that spans over multiple  
lines  
\*/

Creating (Declaring) PHP Variables

$txt = "Hello world!";  
$x = 5;  
$y = 10.5;

echo "x " . $x . "y" . $y;

Output

x 5 y 10.5

$x = 5;  
$y = 4;

echo $x + $y;

## Output

## 9

## PHP Data Types

PHP supports the following data types:

* String
* Integer
* Float (floating point numbers - also called double)
* Boolean
* Array
* Object
* NULL
* Resource

## PHP String

$x = "Hello world!";

## PHP Integer

An integer data type is a non-decimal number between -2,147,483,648 and 2,147,483,647.

$x = 5985;

## PHP Boolean

$x = true;  
$y = false;

## PHP NULL Value

Null is a special data type which can have only one value: NULL.

$x=null;

String functions in PHP

echo strlen("Hello world!");

echo str\_word\_count("Hello world!");

echo strrev("Hello world!");

echo strpos("Hello world!", "world");

echo str\_replace("world", "Dolly", "Hello world!");

## PHP Numbers

PHP has the following functions to check if the type of a variable is integer:

* is\_int()
* is\_integer() - alias of is\_int()
* is\_long() - alias of is\_int()

$x = 5985;  
var\_dump(is\_int($x));

$x = 59.85;  
var\_dump(is\_int($x));

Output:

bool(true)  
bool(false)

* is\_float()
* is\_double() - alias of is\_float()

$x = 10.3434;  
var\_dump(is\_float($x));

## PHP Infinity

* [is\_finite()](https://www.w3schools.com/php/func_math_is_finite.asp)
* [is\_infinite()](https://www.w3schools.com/php/func_math_is_infinite.asp)

$x = 1.9e41

var\_dump($x);

Output

float(INF)

## PHP Numerical Strings

$x = 5985;  
var\_dump(is\_numeric($x)); //TRUE  
  
$x = "5985";  
var\_dump(is\_numeric($x)); //TRUE   
  
$x = "59.85" + 100;  
var\_dump(is\_numeric($x));//TRUE   
  
$x = "Hello";  
var\_dump(is\_numeric($x));//False

True

True

True

False

## PHP Casting Strings and Floats to Integers

$x = 23465.768;  
$int\_cast = (int)$x;  
echo $int\_cast; //23465

# PHP Math

## PHP pi() Function

echo(pi());

## PHP min() and max() Functions

echo(min(0, 150, 30, 20, -8, -200));  // returns -200

echo(max(0, 150, 30, 20, -8, -200));  // returns 150

## PHP sqrt() Function

echo(sqrt(64));

## PHP round() Function

echo(round(0.80));  // returns 1

echo(round(0.49));  // returns 0

## Random Numbers

The rand() function generates a random number:

echo(rand());

echo(rand(10, 100));

## PHP Conditional Statements

In PHP we have the following conditional statements:

* if statement - executes some code if one condition is true
* if...else statement - executes some code if a condition is true and another code if that condition is false
* if...elseif...else statement - executes different codes for more than two conditions
* switch statement - selects one of many blocks of code to be executed

$t = date("H");  
  
if ($t < "20") {  
  echo "Have a good day!";  
}

$ch=”red”;  
switch ($ch) {  
  case "red":  
    echo "Your favorite color is red!";  
    break;  
  case "blue":  
    echo "Your favorite color is blue!";  
    break;  
  case "green":  
    echo "Your favorite color is green!";  
    break;  
  default:  
    echo "Your favorite color is neither red, blue, nor green!";  
}

In PHP, we have the following loop types:

* while - loops through a block of code as long as the specified condition is true
* do...while - loops through a block of code once, and then repeats the loop as long as the specified condition is true
* for - loops through a block of code a specified number of times
* foreach - loops through a block of code for each element in an array

**for ($i=1;$i<=5;$i++)**

**{**

**}**

$colors = array("red", "green", "blue", "yellow");  
  
foreach ($colors as $value) {  
  echo "$value <br>";

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");  
  
foreach($age as $x => $val) {  
  echo "$x = $val<br>";  
}

# PHP Functions

**1)**

function writeMsg() {  
  echo "Hello world!";  
}  
  
writeMsg(); // call the function

2)

function writeMsg($s) {  
  echo $s;  
}  
  
writeMsg(“TY”); // call the function

3)

function addNumbers(float $a, float $b) {

return $a + $b;

}

echo addNumbers(1.2, 5.2);

# PHP Arrays

1)

$a = array(1,2,3);

echo "Class " . $a[0] . ", " . $a[1] . " and " . $a[2] . ".";

Output:

Class FY, SY and TY.

echo count($a);

Output:

3

## PHP Indexed Arrays

$cars = array("Volvo", "BMW", "Toyota");

$cars[0] = "Volvo";  
$cars[1] = "BMW";  
$cars[2] = "Toyota";

# PHP Associative Arrays

Associative arrays are arrays that use named keys that you assign to them.

There are two ways to create an associative array:

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

or:

$age['Peter'] = "35";  
$age['Ben'] = "37";  
$age['Joe'] = "43";

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

echo "Peter is " . $age['Peter'] . " years old.";

Output:

Peter is 35 years old.

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach($age as $x => $x\_value) {

echo "Key=" . $x . ", Value=" . $x\_value;

echo "<br>";

}

Output:

Key=Peter, Value=35  
Key=Ben, Value=37  
Key=Joe, Value=43

Multidimensional Arrays

$a = array (

array("TY","WT",18),

array("TY","WT",13),

array("TY","WT",2),

array("TY","WT",15)

);

for ($row = 0; $row < 4; $row++) {

echo "<p><b>Row number $row</b></p>";

echo "<ul>";

for ($col = 0; $col < 3; $col++) {

echo "<li>".$a[$row][$col]."</li>";

}

echo "</ul>";

}

Output:

**Row number 0**

* TY
* WT
* 18

**Row number 1**

* TY
* WT
* 13

**Row number 2**

* TY
* WT
* 2

**Row number 3**

* TY
* WT
* 15

## PHP - Sort Functions For Arrays

* sort() - sort arrays in ascending order
* rsort() - sort arrays in descending order
* asort() - sort associative arrays in ascending order, according to the value
* ksort() - sort associative arrays in ascending order, according to the key
* arsort() - sort associative arrays in descending order, according to the value
* krsort() - sort associative arrays in descending order, according to the key

$a = array("SY", "TY", "FY");

sort($a);

$len = count($a);

for($x = 0; $x < $len; $x++) {

echo $a[$x];

echo "<br>";

}

Output:

FY  
SY  
TY

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

asort($age);

foreach($age as $x => $x\_value) {

echo "Key=" . $x . ", Value=" . $x\_value;

echo "<br>";

}

Output:

Key=Peter, Value=35  
Key=Ben, Value=37  
Key=Joe, Value=43

PHP Databases: Basic command with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables.