PHP

PHP Basics

- Download PDF
- Download examples Codes

Visual Studio Code - PHP Extension Pack

https://marketplace.visualstudio.com/items?itemName=xdebug.php-pack

Online PHP Editors

- https://www.tutorialspoint.com/execute_php_online.php
- https://www.codingrooms.com/compiler/php/

What is PHP

PHP (recursive acronym for PHP: Hypertext Preprocessor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

Further reading: - Intro. to PHP

What is a PHP File?

- PHP files can contain text, HTML, CSS, JavaScript, and PHP code.
- PHP code are executed on the server, and the result is returned to the browser as plain HTML.
- PHP files have extension ".php".

Example #1 an introductory example

```
<!DOCTYPE html>
<html>
    <head>
        <title>Example</title>
    </head>
    <body>

        <?php
            echo "Hi, I'm a PHP script!";
        ?>

    </body>
</html>
```

phpinfo — Outputs information about PHP's configuration

Example #2 phpinfo example

```php
<?php
// Show all information, defaults to INFO_ALL
phpinfo();
// Show just the module information.
// phpinfo(8) yields identical results.
phpinfo(INFO_MODULES);
?>
```

In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive.

Example #3 echo example

```php
<?php

ECHO "Hello World!<br>";
echo "Hello World!<br>";
EcHo "Hello World!<br>";
?>
```

Download example code

Variables

- Variables are used to store data, like string of text, numbers, etc.
- Variable values can change over the course of a script.
- In PHP, a variable does not need to be declared before adding a value to it.
- PHP automatically converts the variable to the correct data type, depending on its value
- After declaring a variable it can be reused throughout the code.
- The assignment operator (=) used to assign value to a variable.
- all variable names are case-sensitive.

> In PHP variable can be declared as: **$var_name = value;**

Further reading: https://www.tutorialrepublic.com/php-tutorial/php-variables.php

Example #1 variable names

```php
<?php

$color = "red";
$COLOR="GREEN";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
```

```
echo "My boat is " . $coLOR . "<br>";

?>
```

Download example code

Example #2 disply of variable using echo statement

```
<?php

$txt = "W3Schools.com";
echo "I love $txt!";
// produce the same output
echo "<br>";
echo "I love " . $txt . "!";

?>
```

Download example code

Example #3 variable example

```
<?php

$x = 5;
$y = 4;
echo $x + $y;

?>
```

Download example code

Variable scope

- The scope of a variable is the context within which it is defined.
- For the most part all PHP variables only have a single scope.
- This single scope spans included and required files as well.

Further reading: https://www.php.net/manual/en/language.variables.scope.php

Local scope variable

Example #1 local scope variable example

```
<?php
function myTest() {
```

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

// using x outside the function will generate an error
echo "<p>Variable x outside function is: $x</p>";
?>
```

Download example code

Global scope variable

Example #2 global scope variable example

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

function myTest() {
    // using x inside this function will generate an error
    echo "<p>Variable x inside function is: $x</p>";
}
myTest();

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

Download example code

The global keyword

Example #3 using global

```
<?php
$x = 5; // global scope
$y = 10;

function myTest() {
 global $x, $y;  // use global scope variable in function
    $y = $x + $y;

}

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

Download example code

$GLOBALS - The $GLOBALS array is an associative array with the name of the global variable being the key and the contents of that variable being the value of the array element.

- $GLOBALS exists in any scope, this is because $GLOBALS is a superglobal.

Example #4 using $GLOBALS instead of global example

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

function myTest() {
    $GLOBALS['y'] = $GLOBALS['x'] + $GLOBALS['y'];
}

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

Download example code

Using static variables

- A static variable exists only in a local function scope, but it does not lose its value when program execution leaves this scope.

Example #5 Example demonstrating need for static variables

```
<?php
function test()
{
    $a = 0;
    echo $a;
    $a++;
}
?>
```

Example #6 use of static variables

```
<?php
function myTest() {
    static $x = 0;
    echo $x . "<br \>";
    $x++;
```

```
}

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

Download example code

String functions

- Download examples code

strlen

Example #1 Get The Lenght of a String

```
<?php
    echo "Length of a String: ";
    echo strlen("Hello world!"); // outputs 12
?>
```

str_word_count

Example #2 Count The Number of Words in a String

```
<?php
    echo "<br /> Count Words: ";
    echo str_word_count("Hello world!"); // outputs 2
?>
```

strrev

Example #3 Reverse a String

```
<?php
    echo "<br />Reverse a String: ";
    echo strrev("Hello world!"); // outputs !dlrow olleH
?>
```

strpos

Example #4 Search For a Specific Text Within a String

```php
 <?php
    echo "<br />Position: ";
    echo strpos("Hello world!", "world"); // outputs 6
?>
```

str_replace

Example #5 Replace Text Within a String

```php
<?php
    echo "<br />Replace: ";
    echo str_replace("world", "PHP", "Hello world!"); // outputs Hello
Dolly!
?>
```

Comments

Example #1 Comments example

```php
<!DOCTYPE html>
<html>
<body>

<?php
// 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
*/

// You can also use comments to leave out parts of a code line
$x = 5 /* + 15 */ + 5;
$name = "Muhammad Ahmad Nasir";
echo "<h2>$name</h2>";
echo '<h1>' .$x. '</h1>';
?>

</body>
</html>
```

Download example code

define function

define — Defines a named constant

Further reading: - https://www.php.net/manual/en/function.define.php

Example #1 Defines a named constant

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <title>PHP - Contants</title>
    </head>
    <body>
        <?php
        // define(name, value, case-insensitive)
        // case-insensitive: Specifies whether the constant name should be
case-insensitive. Default is false
define("GREETING", "Welcome to Department of CS & IT!");

function myTest() {
    echo GREETING;
}
myTest();
?>
    </body>
</html>
```

Download example code

functions

Example #1 Function examples

```
<!DOCTYPE html>
<html lang="en">
<head>
        <title>PHP - Functions</title>
    </head>
<body>
    <?php
// Syntax
//
/*function functionName() {
    code to be executed;
}*/

//
// Example1
//
```

```php
function writeMsg() {
    echo "Hello world!";
}

writeMsg(); // call the function

//
// Example2 - By Argument
//
 function familyName($fname) {
     echo "$fname Nasir" . "<br>";
 }

familyName("Muhammad"); // call
familyName("Ali");
familyName("Zeeshan");

// Example3 - By two arguments
//
//

function familyName($fname, $year) {
    echo "$fname. Born in $year <br>";
}

familyName("Muhammad Ali", "1975");
familyName("Muhammad Nasir", "1978");
familyName("Muhammad Hamza", "1983");*/

//
// Example 4 - Default Argument Value
//

function setHeight($minheight = 50) {
    echo "The height is : $minheight <br>";
}

setHeight(350);
setHeight(); // will use the default value of 50
setHeight(135);
setHeight(80);

//
// Example 5 - Returning Values
//

function sum($x, $y) {
    $z = $x + $y;
    return $z;
}
$i = sum(5, 10);
echo "5 + 10 = " . $i . "<br>";
echo "7 + 13 = " . sum(7, 13) . "<br>";
```

```
?>
    </body>
</html>
```

Download example code

Muhammad Yasir Bhutta

- Web
- Youtube
- Facebook
- Twitter