

PHP

1

Web Development

2

Front End (Client Side)

- ***HTML***
- ***CSS***
- ***JAVASCRIPT***

Back End (Server Side)

- **PHP**
- **MYSQL**

Content

3

- Introduction on PHP
- Install PHP
- Syntax
- Basic Structure
- Variables and data types
- Output (Echo, Print)
- Operators
- Control statement(If--else)
- Loops
- Functions
- Array

PHP Introduction

4

- PHP is an acronym for "**PHP: Hypertext Preprocessor**"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

PHP File

5

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

What Can PHP Do?

6

- PHP can generate ***dynamic page content***
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can add, delete, modify data in your ***database***
- PHP can be used to control user-access
- PHP can encrypt data

Why PHP?

7

- PHP runs on various platforms (***Windows***, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (***Apache, IIS***, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

Install PHP and Run on Server

8

- **Install**

- install a web server
- install PHP
- install a database, such as MySQL

- **Run**

- *Open Browser (go to **URL**)*
- *Server Must be on (**Xamp/Wamp** Server)*
- *Project file must put on **htdocs/www** folder*
- *URL Address: **127.0.0.1**/folder name/file name*
- *Or **localhost**/folder name/file name*

PHP Syntax

9

- A PHP script can be placed anywhere in the document.
- A PHP script starts with `<?php` and ends with `?>`
- Program lines end with ***semicolon (;)***
- PHP is not case sensitive but ***user defined*** keywords are case sensitive (declaring variables, functions, constant, class)

Example of PHP Scripts

10

- `<!DOCTYPE html>`
`<html>`
`<body>`
`<h1>My first PHP`
`page</h1>`

`<?php`
`echo "My First PHP`
`Script!";`
`?>`
`</body>`
`</html>`

- Output:
- *My First PHP Script!*

Output (echo and print)

11

- `<?php`
 echo "PHP is Fun!";
`?>`
- `<?php`
 print "I am about to
 learn PHP!";
`?>`

- Output for echo:
 PHP is Fun!
- Output for print:
 I am about to learn PHP!

PHP Variables

12

- A variable starts with the **\$** sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (**\$age** and **\$AGE** are two different variables)

Data Types

13

- **PHP supports the following data types:**
 - String
 - Integer
 - Float (floating point numbers - also called double)
 - Boolean
 - Array
 - Object
 - NULL
- ***var_dump()*** function is used to check the datatypes

Examples of Variable with data types

14

- *<?php*

```
$name="Hello";  
$age=30;  
$weight=65.5;  
echo $name;  
echo $age;  
echo $weight;  
var_dump($age);
```

- *?>*

- Output:

- *Hello*
- *30*
- *65.5*
- *int(30)*

PHP Operators

15

- Operators are used to perform operations on variables and values.
- PHP divides the operators in the following groups:
 - Arithmetic operators (+, -, *, /, %)
 - Assignment operators (=, with arithmetic)
 - Comparison operators (==, >, <, >=, <=, !=)
 - Increment/Decrement operators (++ , --)
 - Logical operators (&&, ||, !)
 - String operators (., .=)
 - Array operators
 - Conditional assignment operators

PHP Conditional Statements

16

- ***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

Examples

(17)

- `<?php`
 `$mark=40;`
 `if($mark>=40)`
 `echo "Passed";`
- `?>`
`<?php`
 `$mark=36;`
 `if($mark>=40)`
 `echo "Passed";`
 `else`
 `echo "Failed";`
`?>`

```
$mark=70;
if ($mark>=80)
    echo "It is A";
else if($mark<80 && $mark>=60)
    echo "It is B";
else if($mark<60 && $mark>=40)
    echo "It is C";
else
    echo "Failed";
```

Examples of Switch

18

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

Loops

19

- In PHP, we have the following loop types:
 - ***while*** - loops through a block of code 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

Examples

20

- ```
<?php
$x = 1;
while($x <= 5) {
 echo "The number is: $x

";
 $x++;
}
?>
```

- **Output:**  
The number is: 1  
The number is: 2  
The number is: 3  
The number is: 4  
The number is: 5

- ```
<?php
for ($x = 1; $x <= 5; $x++ ) {
    echo "The number is: $x
    <br>";
}
?>
```

- **Output:**
The number is: 1
The number is: 2
The number is: 3
The number is: 4
The number is: 5

Arrays

21

- An array is a special variable, which can hold more than one value at a time.
- ***array()*** function is used to create an array
- In PHP, there are three types of arrays:
 - **Indexed arrays** - Arrays with a numeric index
 - **Associative arrays** - Arrays with named keys
 - **Multidimensional arrays** - Arrays containing one or more arrays

Examples-Indexed Array

22

```
<?php
$cars = array("Volvo", "BMW", "Toyota");
$arlength = count($cars);

for($x = 0; $x < $arlength; $x++) {
    echo $cars[$x];
    echo "<br>";
}
?>
```

Associative array

23

- *<?php*
\$age
= array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
echo "Peter is " . \$age['Peter'] . " years old.";

- *?>*

- **Output:**

- *Peter is 35 years old.*

For each loop [Indexed Array]

24

- ```
<?php
$colors = array("red", "green", "blue", "yellow");

foreach ($colors as $value)
{
 echo "$value
";
}
?>
```



# For each loop [Associative Array]

25

- ```
<?php
$age
= array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach($age as $x => $x_value)
{
    echo "Key=" . $x . ", Value=" . $x_value;
    echo "<br>";
}
?>
```

Functions

26

- A function is a block of statements that can be used repeatedly in a program.
- A function will not execute automatically when a page loads.
- A function will be executed by a call to the function.

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

writeMsg(); // call the function
?>
```

Super Global Variables

27

- The PHP superglobal variables are:
- `$_SERVER`
- `$_REQUEST`
- `$_POST`
- `$_GET`
- `$_FILES`
- `$_COOKIE`
- `$_SESSION`

HTML and CSS in PHP

28

- Echo “<h1> Hello HTML</>”;

PHP in HTML

29

- File extension should be *.php*
- Inset PHP in HTML file with php tag.

References

30

- PHP: <https://www.w3schools.com/php/default.asp>
- ALL Web based Technology (HTML, CSS, JS, PHP, MYSQL)
- Video Tutorial:
<http://www.trainingwithliveproject.com/>

Home work

31

- HTML, CSS, MYSQL
- Assignment

Thank You

32