

# Tutorial 07 – PHP Basics

## Table of Contents

Learning Outcomes .....	1
1. Introduction of PHP .....	2
What is a PHP File? .....	2
Why PHP? .....	2
2. Basic PHP Syntax.....	3
First PHP Program.....	3
Comments in PHP .....	3
Variables in PHP .....	4
PHP is a Loosely Typed Language .....	4
String Variables in PHP - Concatenation Operator .....	4
String Variables in PHP - strlen() function.....	5
String Variables in PHP – strpos() function .....	5
PHP Operators – Arithmetic Operators .....	5
PHP Operators - Comparison Operators .....	6
PHP Operators – Assignment Operators .....	6
PHP Exercise .....	7

## Learning Outcomes

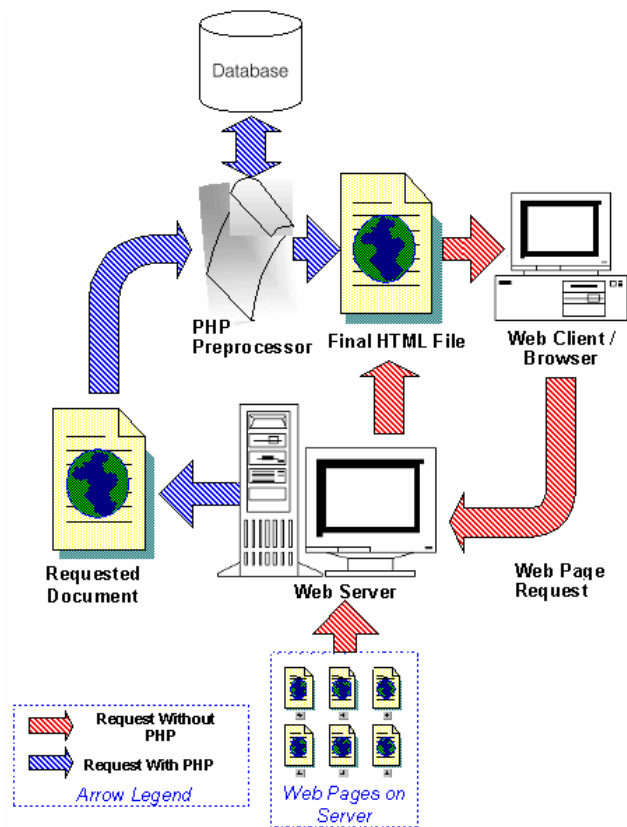
1. Basic PHP Syntax
2. Declare variables
3. Operators

## 1. Introduction of PHP

- PHP stands for PHP: Hypertext Preprocessor
- PHP is a server-side scripting language, like ASP
- PHP scripts are executed on the server
- PHP is an open source software
- PHP is free to download and use

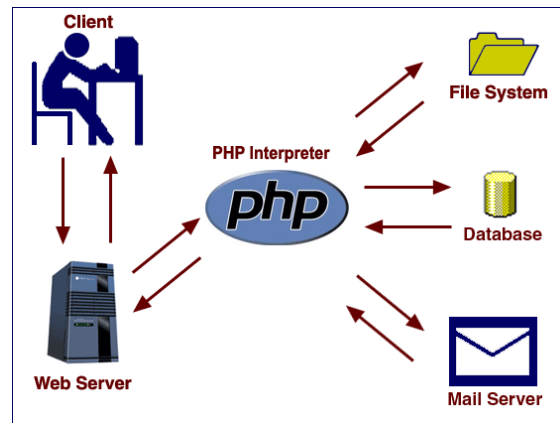
### What is a PHP File?

- PHP files can contain text, HTML tags and scripts
- PHP code is executed on the server, and the plain HTML result is sent to the browser.
- PHP files are returned to the browser as plain HTML
- PHP files have a file extension of ".php"



### Why PHP?

- PHP runs on different platforms (e.g. Windows, Linux, Unix)
- PHP is compatible with almost all servers used today (e.g. Apache)
- PHP is FREE to download from the official PHP resource: [www.php.net](http://www.php.net)
- PHP is easy to learn and runs efficiently on the server side



## 2. Basic PHP Syntax

<pre>&lt;?php     // PHP Comment ?&gt;</pre>	<ul style="list-style-type: none"> <li>• A PHP scripting block always starts with <code>&lt;?php</code> and ends with <code>?&gt;</code>.</li> <li>• A PHP scripting block can be placed anywhere in the document.</li> <li>• On servers with shorthand support enabled you can start a scripting block with <code>&lt;?</code> and end with <code>?&gt;</code>.</li> <li>• For maximum compatibility, we recommend that you use the standard form (<code>&lt;?php</code>) rather than the shorthand form.</li> </ul>
--	---

### First PHP Program

[php01_01_helloworld.php]	
<pre>&lt;html&gt;   &lt;body&gt;     &lt;?php       echo "Hello World";     ?&gt;   &lt;/body&gt; &lt;/html&gt;</pre>	<ul style="list-style-type: none"> <li>• A PHP file normally contains HTML tags, just like an HTML file, and some PHP scripting code.</li> <li>• Below, we have an example of a simple PHP script which sends the text "Hello World" to the browser:</li> <li>• Each code line in PHP must end with a semicolon. The semicolon is a separator and is used to distinguish one set of instructions from another.</li> <li>• We have used the echo statement to output the text "Hello World".</li> <li>• Note: The file must have a .php extension. If the file has a .html extension, the PHP code will not be executed.</li> </ul>

### Comments in PHP

[php01_02_comment.php]	In PHP, we use
<pre>&lt;?php // This is a comment /* This is Comment Block */ ?&gt;</pre>	<ul style="list-style-type: none"> <li>• <code>//</code> to make a single-line comment or</li> <li>• <code>/*</code> and <code>*/</code> to make a large comment block.</li> </ul>

## Variables in PHP

<pre>&lt;?php     \$var_name = value;     \$valid1 = true;     \$1d = false; // invalid     variable = "no"; ?&gt;</pre>	<ul style="list-style-type: none"> <li>Variables are used for storing a values, like text strings, numbers or arrays.</li> <li>When a variable is declared, it can be used over and over again in your script.</li> <li>All variables in PHP start with a \$ sign symbol.</li> <li>A variable name must start with a letter or an underscore <code>_</code>, can only contain alpha-numeric characters and underscores (i.e. a-z, A-Z, 0-9, and <code>_</code>), and should not contain spaces.</li> </ul>
--	--

## PHP is a Loosely Typed Language

<pre>&lt;?php     \$txt="Hello World!";     \$x=16;     \$txt=23; ?&gt;</pre>	<ul style="list-style-type: none"> <li>a variable does not need to be declared before being set.</li> <li>In the example, you see that you do not have to tell PHP which data type the variable is.</li> <li>PHP automatically converts the variable to the correct data type, depending on how they are set.</li> <li>In a strongly typed programming language, you have to declare (define) the type and name of the variable before using (setting) it.</li> <li>In PHP, the variable is declared automatically when you use it.</li> </ul>
---	--

## String Variables in PHP - Concatenation Operator

php01_03_concatenation.php	
<pre>&lt;?php \$txt1="Hello World!"; \$txt2="What a nice day!";  echo \$txt1 . " " . \$txt2;  ?&gt;</pre>	<ul style="list-style-type: none"> <li>The concatenation operator (<code>.</code>) is used to put two string values together.</li> <li>e.g.</li> </ul>

## String Variables in PHP - strlen() function

php01_04_strlen.php	
<pre>&lt;?php echo strlen("Hello world!"); ?&gt;</pre>	<p>strlen() function</p> <ul style="list-style-type: none"> <li>The strlen() function is used to return the length of a string.</li> </ul>

## String Variables in PHP – strpos() function

php01_05_strpos.php	
<pre>&lt;?php echo strpos("Hello world!", "world"); ?&gt;</pre>	<p>strpos() function</p> <ul style="list-style-type: none"> <li>The strpos() function is used to search for character within a string.</li> <li>If a match is found, this function will return the position of the first match.</li> <li>If no match is found, it will return FALSE.</li> <li>Note: string positions start at 0.</li> </ul>

## PHP Operators – Arithmetic Operators

Operator	Description	Example	Result
+	Addition	<pre>\$x = 2; \$x = \$x + 2;</pre>	4
-	Subtraction	<pre>\$x = 2; \$x = 5 - \$x;</pre>	3
*	Multiplication	<pre>\$x = 4; \$x = \$x * 5;</pre>	20
/	Division	<pre>15/5 5/2</pre>	3 2.5
%	Modulus (division remainder)	<pre>5%2 10%8 10%2</pre>	1 2 0
++	Increment	<pre>\$x = 5; \$x++;</pre>	x=6
--	Decrement	<pre>\$x = 5; \$x--;</pre>	x=4

## PHP Operators - Comparison Operators

<code>\$a == \$b</code>	Equal	TRUE if \$a is equal to \$b after type juggling.
<code>\$a === \$b</code>	Identical	TRUE if \$a is equal to \$b, and they are of the same type.
<code>\$a != \$b</code>	Not equal	TRUE if \$a is not equal to \$b after type juggling.
<code>\$a &lt;&gt; \$b</code>	Not equal	TRUE if \$a is not equal to \$b after type juggling.
<code>\$a !== \$b</code>	Not identical	TRUE if \$a is not equal to \$b, or they are not of the same type.
<code>\$a &lt; \$b</code>	Less than	TRUE if \$a is strictly less than \$b.
<code>\$a &gt; \$b</code>	Greater than	TRUE if \$a is strictly greater than \$b.
<code>\$a &lt;= \$b</code>	Less than or equal to	TRUE if \$a is less than or equal to \$b.
<code>\$a &gt;= \$b</code>	Greater than or equal to	TRUE if \$a is greater than or equal to \$b.

## PHP Operators – Assignment Operators

Operator	Example	Is The Same As
<code>=</code>	<code>\$x=\$y</code>	<code>\$x=\$y</code>
<code>+=</code>	<code>\$x+= \$y</code>	<code>\$x=\$x+\$y</code>
<code>-=</code>	<code>\$x-= \$y</code>	<code>\$x=\$x-\$y</code>
<code>*=</code>	<code>\$x*= \$y</code>	<code>\$x=\$x*\$y</code>
<code>/=</code>	<code>\$x/= \$y</code>	<code>\$x=\$x/\$y</code>
<code>.=</code>	<code>\$x.= \$y</code>	<code>\$x=\$x.\$y</code>
<code>%=</code>	<code>\$x%= \$y</code>	<code>\$x=\$x%\$y</code>

## PHP Exercise

1. Use a text editor to type the following. Save the file with name SEHH1016\_T7.php, and put it in the **root directory** of your Apache Web server. It should be the “htdocs” folder by default.

```
<HTML>
  <BODY>
    <P> First exercise: </P>

    <?php
      $msg1 = "My first PHP file.<BR>";
      $msg2 = "PHP is interesting.";

      echo $msg1;
      echo $msg2;

    ?>

  </BODY>
</HTML>
```

2. Start a Web browser. Type in the address bar the URL of your PHP file: [http://localhost/SEHH1016\\_T7.php](http://localhost/SEHH1016_T7.php). You should now be able to see the two messages (msg1 and msg2) in the PHP file.
3. Now, in the PHP file, add another variable called “msg3”. Set it with a value “This is a testing message!”.
4. Modify the PHP file such that the third message is placed **below** the second message. View the result using the Web browser.
5. By using another variable called “line”, insert a horizontal line below the three text messages.
6. Modify the value in “msg2”, so that the font colour of the second message is changed to blue. (Hint: use the escape character)
7. Save your work and view the result through the Web server with a Web browser.