

# Getting Started with PHP

## Applications of PHP

PHP is a server scripting language used for making dynamic web pages. That means PHP allows you to use scripts on a web server to produce a response customized for each client's (user's) request.

PHP is the foundation of:

- Many CMS (WordPress, Drupal, Joomla)
- E-Commerce Platforms (WooCommerce, Magento)
- Web Development Frameworks (Laravel, CakePHP, Symfony)

## Terminating a PHP Statement

In PHP, a semicolon `;` must be used to terminate a statement.

Omitting a semicolon at the end of a statement will cause PHP to assume that the next statement is part of the current one since whitespace is usually ignored. This might throw an error.

## PHP Keyword Case-Insensitivity

PHP is NOT case sensitive for language keywords like `if`, `else`, `null`, `echo`, `foreach`, etc.

A specific example- `Echo` is interchangeable with `echo`. However, it is good practice to use the same casing from when the keyword or function was defined. On the other hand, constants, variables, array keys, class constants and class properties are case sensitive.

```
// Three statements, all with their own ;
echo "Bob";
echo "John";
$var1 = "Hi";
```

```
echo "Hi";
Echo "Hi";
//Both the above lines are correct in PHP/

$var1 = 5;
$Var1 = 10;
// These are two different variables in
PHP
```

## PHP Comments

Comments are text within code that won't be executed when the program is run.

Comments can be useful for explaining how code is working and leaving notes for other developers. PHP supports both *single line comments* and *multi-line comments*.

For *single line comments*, the syntax is `//` or `#`. Anything after these symbols on a line is ignored by the PHP interpreter.

For *multi-line comments*, anything between `/*` and `*/` is ignored by the interpreter.

```
echo "Hi";
```

```
// This is a single-line comment. This  
line is ignored by PHP.
```

```
# This is also a single-line comment. This  
line is also ingnored by PHP.
```

```
/* This is a multi-line comment.  
everything in between these two comment  
delimiters is ignored by PHP. */
```

## Dynamic Webpages using PHP

PHP is a *server scripting language*. This server side code can be used to fill out HTML templates in order to create a complete HTML document for a visitor. This finished document is called a *Dynamic Webpage*.

Dynamic PHP webpages can deliver a custom set of assets to each visitor, unlike static pages which deliver the same set of assets to everyone.

## PHP Script Execution

PHP can be served such that any browser making a connection to the server can execute the PHP file. This is called server-side scripting.

From the terminal, PHP scripts can be executed on demand and the output of the script is logged to the terminal. This is called command-line scripting.

## Embedding PHP Code

PHP can generate HTML when saved as a file with a `.php` extension. These files must always start with the tag `<?php` (closing tag is optional).

PHP can also be embedded into HTML. In this case, both opening tag `<?php` and closing tag `?>` are used. For example, in the given code, the PHP code has been embedded into the HTML by enclosing it within the `<?php` and `?>` tags.

```
<html>  
  <body>  
    <?php  
      echo "Hello PHP World!";  
    ?>  
  </body>  
</html>
```

## PHP echo Keyword

The `echo` keyword in PHP is used to write output to the browser, if being served.

If it is executed in the command-line, then the `echo` keyword writes the output to the terminal.



```
echo "Hi, how are you?"
```