

An introduction into PHP programming

Before we start, you should have atleast a basic understanding of *HTML*, *CSS* and *JavaScript*.

What is PHP?

PHP is a widely used open-source programming language, in my opinion, PHP is the most easiest way to start building interactive websites. It's easy enough to learn web development, and it's also powerful enough to run the largest social network, Facebook! PHP codes are executed on the server, not exposed to the client so it's safe also. It's free to download and use, also there's a ton of tutorials available for it on the internet... like this document.

What is a PHP file?

PHP files can contain PHP (obviously), HTML, CSS and JavaScript code. PHP Is executed on the server and returns HTML to the client, and PHP files has the **.php** file extension.

What can PHP do?

- generate dynamic page content
- create, open, read, write, delete, and close files on the server
- collect form data
- send and receive cookies
- add, delete, modify data in your database
- be used to control user-access
- encrypt/decrypt data
- outputs not only HTML but it can generate PDF, images or other files

Basic PHP things

A PHP code starts with **<?php** and ends with **?>**.

Example:

```
<?php
    echo "I am a php code";
?>
```

A PHP file usually contains HTML and PHP code only. CSS and JavaScript should be stored in separate files.

Comments in PHP

You can do single line comment with:

```
//this is a comment
```

...and multi-line comment with:

```
/*  
this is  
a multi-line  
comment  
*/
```

Creating variables in PHP

Variables in PHP starts with a \$ sign and after the name.

```
<?php  
    $text = "something";  
    echo $text;  
?>
```

The code above outputs "something" in HTML.

Note: Unlike other programming languages, PHP has no command for declaring a variable. It is created the moment you first assign a value to it.

Rules for PHP variable names:

- 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)

Variables scope

If you put a variable outside a function it will be available to all other functions, like a "global" variable. Example:

```
<?php  
    $globalvar = "asd";  
    function lorem() {  
        echo $globalvar;  
    }  
    lorem();  
?>
```

Here's another example where a variable is not "global":

```
<?php
    function lorem() {
        $var = "asd";
        echo $var;
    }
    lorem();
?>
```

Also there's a "global" keyword you can use:

```
<?php
    function lorem() {
        $var = "asd";
        global $test = "I am global";
        echo $globalvar;
    }
    lorem();
    echo $test;
?>
```

PHP Classes

And now let's talk about something that makes things work. We have seen Classes used in the workshop's so this is why I bring that up:

```
<?php
class Fruit {
    // Properties
    public $name;
    public $color;

    // Methods
    function set_name($name) {
        $this->name = $name;
    }
    function get_name() {
        return $this->name;
    }
    function set_color($color) {
        $this->color = $color;
    }
    function get_color() {
        return $this->color;
    }
}
```

```
$apple = new Fruit();  
$apple->set_name('Apple');  
$apple->set_color('Red');  
echo "Name: " . $apple->get_name();  
echo "<br>";  
echo "Color: " . $apple->get_color();  
?>
```

This code will set you up a Fruit class, with a name and color variable in it.

What's \$this?

The \$this keyword refers to the current object, and is only available inside methods.

Sample Project

There's a sample project available at this link:

<https://github.com/stephenmiller04/TEAMAPHPTUT>

It's a simple PHP CMS with comments so you can understand what's happening here and master the basics of PHP!