

## HISTORY OF PHP

- PHP was founded in 1994.
- It is one of the most common server-side scripting languages and works well with HTML and Databases.
- PHP is commonly used in the creation of dynamic websites.
- Since we cannot always reinvent the wheel, once the basics of PHP are fully grasped one can then proceed to use PHP frameworks such as Laravel, Symphony and cakePHP. These frameworks provide scaffolding and solutions to common problems in back-end web development.
- PHP also provides underlying code for most content management systems (CMS).
- A CMS allows one to create and update their websites without having to write complex code themselves.
- Examples of CMS are such as wordpress, drupal and joomla

## HOW IS PHP USED IN HTML?

- PHP is often used to create dynamic web pages.
- A dynamic webpage is one whereby each visitor gets a customised page that can look different from how the site looks to a different visitor.
- Static web pages provide the same content to each visitor.
- In order to create this dynamic behavior, PHP was designed to work closely with HTML.
- PHP can be used directly in-line with an HTML document. When the web site is delivered from the back-end to the front-end, the PHP content is executed and added to the HTML to form one HTML document.
- The start of in-line PHP is denoted with `<?php` and the end is denoted with `?>`.
- In PHP, the `echo` keyword is used to output text.
- The text in this case is everything between the double quotes (`"`). An instruction written in PHP is called a *statement*.
- A semicolon (`;`) is required at the end of each statement in PHP.

**NB:** The code to start the web server is: `php -S localhost:8888` (You can use any port number that you like)

## HOW IS PHP EXECUTED?

PHP is flexible and can be executed from the terminal.

We can use PHP as a general purpose programming language to write programs that give simple instructions to the computer without involving HTML or the web.

When this is done, the output of the program is logged to the terminal. This is useful when testing functionality or for writing simple local programs.

When writing a PHP script file, we still need to denote that we are beginning our PHP code using `<?php`, but the closing tag is no longer required. It is typically left out by convention.

Generally PHP ignores whitespaces (Tabs, Spaces and new lines)

## PHP COMMENTS

Sometimes, we want to include text in our files that we don't want the computer to execute or display to the end user. We can do this with *comments*. Comments can be used to annotate our code to make it clearer to ourselves or others. They are also useful to prevent lines of code from being executed without deleting them.

In PHP, there are two main ways to add comments to our code. The first is single line comments. These are typically used for short explanations or points of clarification. Either `#` or `//` can be used to create a single line comment. Anything on the same line after these symbols is not executed by PHP.

The second type of comment is a multi-line comment. This is used for longer descriptions, a more detailed guide on how to properly use the section of code, or to

prevent several lines of code from being executed. These comments are started with `/*` and ended with `*/`.

## REVIEW

- Despite its age, PHP is still a commonly used technology in web development.
- PHP is designed to interact with HTML to generate dynamic websites.
- Embedding PHP in HTML is done by placing PHP code between `<?php` and `?>` tags.
- Every statement in PHP must be terminated with a semicolon `;`.
- PHP files have a `.php` extension and the file always starts with the opening PHP tag `<?php`. The closing tag is implied and left out by convention.
- Whitespace is generally ignored when executing PHP code.
- Keywords are not case sensitive in PHP. As a convention, use the standard casing.
- Single line comments are made in PHP using `#` or `//`. Multi-line comments are placed between `/*` and `*/`.