

Learn PHP in 1 Day

By Krishna Rungta

Copyright 2019 - All Rights Reserved – Krishna Rungta

ALL RIGHTS RESERVED. No part of this publication may be reproduced or transmitted in any form whatsoever, electronic, or mechanical, including photocopying, recording, or by any informational storage or retrieval system without express written, dated and signed permission from the author.

Table Of Content

Chapter 1: What is PHP? Write your first PHP Program

1. [What is PHP?](#)
2. [What is a Scripting Language?](#)
3. [Programming Language Vs Scripting Language](#)
4. [What does PHP stand for?](#)
5. [Php Syntax](#)
6. [Why use PHP?](#)
7. [What is PHP used for & Market share](#)
8. [PHP vs Asp.Net VS JSP VS CFML](#)
9. [PHP File Extensions](#)
10. [PHP Hello world](#)

Chapter 2: How to Download & Install XAMPP & NetBeans: PHP Tutorial

1. [What is XAMPP?](#)
2. [Why use XAMPP?](#)
3. [How to Download and Install XAMPP](#)
4. [Basic Web server configuration](#)
5. [XAMPP Control Panel](#)
6. [Configure XAMPP](#)
7. [What is the best PHP IDE?](#)
8. [Introduction to Netbeans IDE](#)
9. [Creating a new PHP project using the Netbeans IDE](#)
10. [Running your first PHP Example](#)

Chapter 3: PHP Data Types, Variables, Constant, Operators Tutorial

1. [PHP Data Types](#)
2. [PHP Variable](#)
3. [Use of Variables](#)
4. [Variable Type Casting](#)
5. [PHP Constant](#)
6. [PHP Operators](#)

Chapter 4: PHP Comments, Include/Include once, Require/Require once

1. [Why use Comments?](#)
2. [PHP Comments](#)
3. [PHP Include & PHP Include once](#)
4. [Example : Include / Include once](#)
5. [PHP Require & PHP require once](#)
6. [Example : Require](#)
7. [Php include vs require](#)

Chapter 5: PHP Array: Associative, Multidimensional

1. [What is a PHP Array?](#)
2. [Numeric Arrays](#)
3. [PHP Associative Array](#)
4. [PHP Multi-dimensional arrays These are arrays that contain other nested](#)
5. [PHP Arrays: Operators](#)
6. [PHP Array Functions](#)

Chapter 6: PHP Control Structures: If else, Switch Case

1. [What is a control structure?](#)
2. [PHP IF Else](#)
3. [PHP Switch Case](#)

Chapter 7: PHP Loop: For, ForEach, While, Do While [Example]

Chapter 8: PHP String Functions: substr, strlen, strtolower, explode, strpos, str_replace

1. [What is a string?](#)
2. [PHP Create strings](#)
3. [PHP Create Strings Using Double quotes](#)
4. [PHP Heredoc](#)
5. [PHP Nowdoc](#)
6. [PHP string functions](#)

Chapter 9: PHP Function: Numeric, Built in, String, Date, User Defined

1. [What is a Function?](#)
2. [Why use Functions?](#)
3. [Built in Functions](#)
4. [String Functions](#)
5. [Numeric Functions](#)
6. [Date Function](#)
7. [Why use User Defined Functions?](#)

Chapter 10: PHP Registration Form using GET, POST Methods with Example

1. [What is Form?](#)
2. [When and why we are using forms?](#)
3. [Create a form](#)
4. [Submitting the form data to the server](#)
5. [GET vs POST Methods](#)
6. [Processing the registration form data](#)
7. [More examples](#)

Chapter 11: PHP Session & PHP Cookies with Example

1. [What is Cookie?](#)
2. [Why and when to use Cookies?](#)
3. [Creating Cookies](#)
4. [Retrieving the Cookie value](#)
5. [Delete Cookies](#)
6. [What is a Session?](#)
7. [Why and when to use Sessions?](#)
8. [Creating a Session](#)
9. [Destroying Session Variables](#)

Chapter 12: PHP File() Function: File_exists, Fopen, Fwrite, Fclose, Fgets, copy, unlink

1. [What is a File?](#)
2. [PHP File Formats Support](#)
3. [PHP files Functions](#)
4. [PHP File_exists Function](#)
5. [PHP Fopen Function](#)
6. [PHP Fwrite Function](#)

7. [PHP Fclose Function](#)
8. [Testing the code](#)
9. [PHP Fgets Function](#)
10. [PHP Copy Function](#)
11. [Deleting a file](#)
12. [PHP File_get_contents Function](#)

Chapter 13: PHP Try Catch Example: Exception & Error Handling Tutorial

1. [What is an Exception?](#)
2. [Why handle exception?](#)
3. [PHP Error handling](#)
4. [Error handling examples](#)
5. [Difference between Errors and Exception](#)
6. [Multiple Exceptions](#)
7. [Testing the code](#)

Chapter 14: PHP Regular Expressions Tutorial: Preg_match, Preg_split, Preg_replace

1. [What is a Regular Expressions?](#)
2. [Regular expressions in PHP](#)
3. [PHP Preg_match](#)
4. [PHP Preg_split](#)
5. [PHP Preg_replace](#)
6. [Meta characters](#)

Chapter 15: How to Send Email using PHP mail() Function

1. [What is PHP mail?](#)

2. [Why/When to use the mail PHP](#)
3. [Simple Mail Transmission Protocol \(SMTP\)](#)
4. [Sanitizing email user inputs](#)
5. [Secure Mail](#)

Chapter 16: PHP MySQLi Functions: mysqli_query, mysqli_connect, mysqli_fetch_array

1. [PHP mysqli_connect function](#)
2. [PHP mysqli_select_db function](#)
3. [PHP mysqli_query function](#)
4. [PHP mysqli_num_rows function](#)
5. [PHP mysqli_fetch_array function](#)
6. [PHP mysqli_close function](#)
7. [PHP Data Access Object PDO](#)

Chapter 17: PHP Object Oriented Programming (OOPs) concept Tutorial with Example

1. [What is OOPs?](#)
2. [Object Oriented Programming Principles](#)
3. [OOPs Concepts in PHP](#)
4. [What is UML?](#)
5. [How to Create a class in PHP](#)
6. [How implement Inheritance in PHP](#)
7. [How to Create object of the class](#)
8. [Testing our application](#)

Chapter 18: PHP Date & Time Function with Example

1. [What is PHP Date Function?](#)

2. [PHP Date Syntax & Example](#)
3. [What is a TimeStamp?](#)
4. [Getting a list of available time zone identifiers](#)
5. [PHP set Timezone Programmatically](#)
6. [PHP Mktime Function](#)
7. [PHP Date function reference](#)
8. [PHP Time parameters](#)
9. [Day parameters](#)
10. [Month Parameters](#)
11. [Year Parameters](#)

Chapter 19: PHP Security Function: strip_tags, filter_var, Md5 and sha1

1. [Potential security threats](#)
2. [PHP Application Security Best Practices](#)
3. [PHP strip_tags](#)
4. [PHP filter_var function](#)
5. [PHP Md5 and PHP sha1](#)

Chapter 20: PHP XML Tutorial: Create, Parse, Read with Example

1. [What is XML?](#)
2. [What is DOM?](#)
3. [XML Parsers](#)
4. [Why use XML?](#)
5. [XML Document example](#)
6. [How to Read XML using PHP](#)
7. [Testing our application](#)
8. [How to Create an XML document using PHP](#)

9. [Testing our application](#)

Chapter 21: PHP Projects: Create an Opinion Poll Application

1. [Step 1\) Database Connectivity](#)
2. [Step 2\) Coding our application](#)
3. [Step 3\) Testing our application](#)

Chapter 22: PHP Ajax Tutorial with Example

1. [What is Ajax?](#)
2. [Why use AJAX?](#)
3. [How to Create an PHP Ajax application](#)
4. [Step 1\) Creating the index page](#)
5. [Step 2\) Creating the frameworks page](#)
6. [Step 3\) Creating the JS script](#)
7. [Step 4\) Testing our PHP Ajax application](#)

Chapter 23: PHP MVC Framework Tutorial: CodeIgniter Example

1. [What is PHP MVC framework?](#)
2. [PHP MVC Design Pattern](#)
3. [Types of PHP MVC framework](#)
4. [Porting the opinion poll application to CodeIgniter](#)
5. [Database configuration settings](#)
6. [Creating Our Model](#)
7. [Creating Our Views](#)
8. [Testing our application](#)

Chapter 24: CakePHP Framework Tutorial for Beginners

1. [What is CakePHP?](#)
2. [Why use CakePHP Framework?](#)
3. [History of CakePHP](#)
4. [Features of CakePHP](#)
5. [CakePHP- Folder Structure](#)
6. [Configuration of CakePHP](#)
7. [Cake- PHP naming conventions](#)
8. [How does MVC work in CakePHP?](#)
9. [Disadvantages of using the CakePHP framework](#)

Chapter 25: PHP vs JavaScript: Must Know Differences

1. [PHP](#)
2. [JavaScript](#)
3. [PHP vs Java-Script](#)
4. [Repository](#)
5. [Trend of JavaScript vs PHP](#)

Chapter 1: What is PHP?

Write your first PHP Program

What is PHP?

PHP is a server side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages.

PHP scripts can only be interpreted on a server that has PHP installed.

The client computers accessing the PHP scripts require a web browser only.

A PHP file contains PHP tags and ends with the extension ".php".

What is a Scripting Language?

A script is a set of programming instructions that is interpreted at runtime.

A scripting language is a language that interprets scripts at runtime. Scripts are usually embedded into other software environments.

The purpose of the scripts is usually to enhance the performance or perform routine tasks for an application.

Server side scripts are interpreted on the server while client side scripts are interpreted by the client application.

PHP is a server side script that is interpreted on the server while JavaScript is an example of a client side script that is interpreted by the client browser. Both PHP and JavaScript can be embedded into HTML pages.

Programming Language Vs Scripting Language

Programming language	Scripting language
Has all the features needed to develop complete applications.	Mostly used for routine tasks
The code has to be compiled before it can be executed	The code is usually executed without compiling
Does not need to be embedded into other languages	Is usually embedded into other software environments.

What does PHP stand for?

PHP means - **Personal Home Page**, but it now stands for the recursive backronym PHP: Hypertext Preprocessor.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks.

Php Syntax

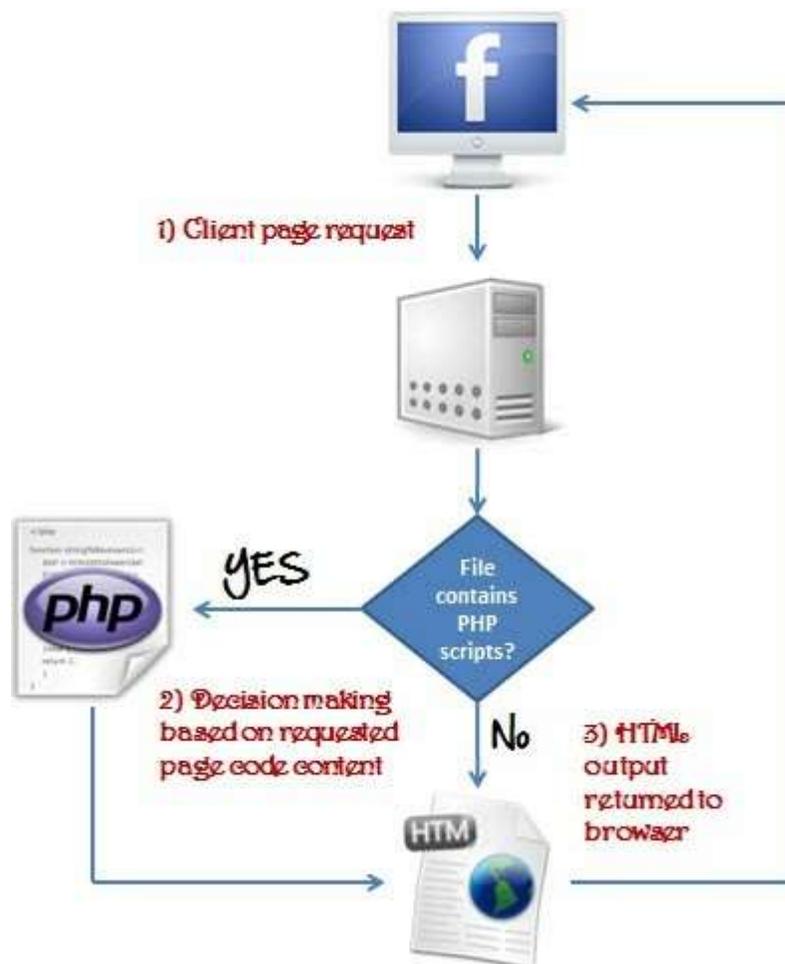
```
<?php  
    echo 'Hello World';  
?>
```

A PHP file can also contain tags such as HTML and client side scripts

such as JavaScript.

- **HTML is an added advantage** when learning PHP Language. You can even learn PHP without knowing HTML but it's recommended you at least know the basics of HTML.
- **Database management systems** DBMS for database powered applications.
- For more advanced topics such as interactive applications and web services, you will need **JavaScript and XML**.

The flowchart diagram shown below illustrates the basic architecture of a PHP web application and how the server handles the requests.



Why use PHP?

You have obviously heard of a number of programming languages out there; you may be wondering why we would want to use PHP as our poison for the web programming. Below are some of the compelling reasons.

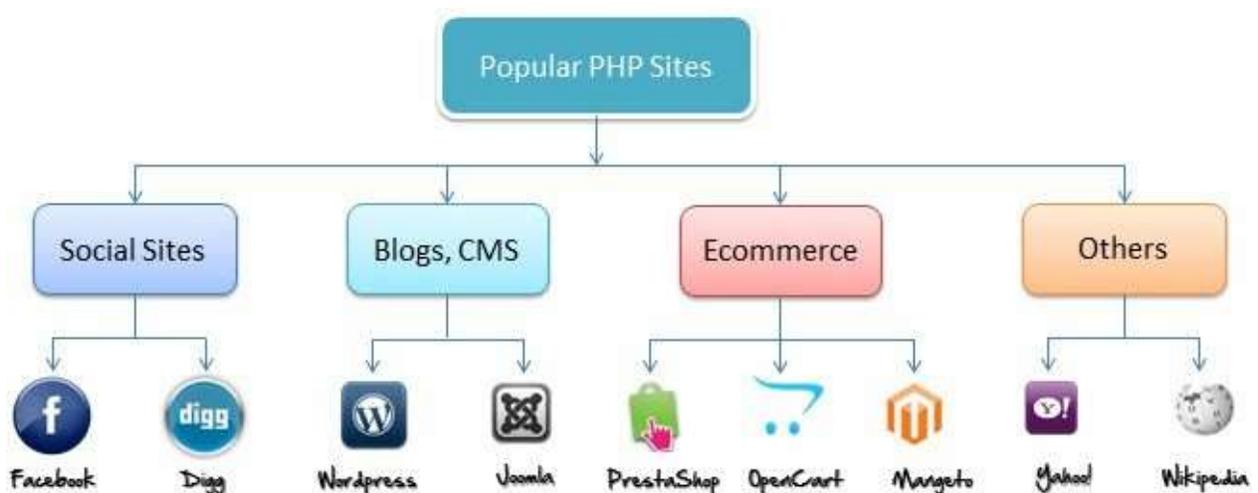
- PHP is **open source and free**.
- Short learning curve compared to other languages such as JSP, ASP etc.
- Large community document
- Most web hosting servers support PHP by default unlike other languages such as ASP that need IIS. This makes PHP a cost effective choice.
- PHP is regularly updated to keep abreast with the latest technology trends.
- Other benefit that you get with PHP is that it's a **server side scripting language**; this means you only need to install it on the server and client computers requesting for resources from the server do not need to have PHP installed; only a web browser would be enough.
- PHP has **in built support for working hand in hand with MySQL**; this doesn't mean you can't use PHP with other database management systems. You can still use PHP with
 - Postgres
 - Oracle
 - MS SQL Server
 - ODBC etc.
- PHP is **cross platform**; this means you can deploy your application on a number of different operating systems such as windows, Linux, Mac OS etc.

What is PHP used for & Market share

In terms of market share, there are over 20 million websites and application on the internet developed using PHP scripting language.

This may be attributed to the points raised above;

The diagram below shows some of the popular sites that use PHP



PHP vs Asp.Net VS JSP VS CFML

ASP – Active Server Pages, JSP – Java Server Pages, CFML – Cold Fusion

Markup language The table below compares the various server side scripting languages with PHP

FEATURE	PHP	ASP	JSP	CFML
Learning curve	short	Longer than PHP	Longer than PHP	Longer than PHP
Web hosting	Supported by almost all hosting servers	Needs dedicated server	Fairly supported	Needs dedicated server
Open source	Yes	No	Yes	Both commercial and open source
Web services support	Built in	Uses the .NET framework	Uses add on libraries	Built in
Integration with HTML	Easy	Fairly complex	Fairly complex	Easy
MySQL support	Native	Needs third party drivers	Needs third party drivers	Current version has native support. Older versions use ODBC
Easily extended by other languages	Yes	No	Extended using Java classes and libraries.	Yes

PHP File Extensions

File extension and Tags In order for the **server** to **identify** our **PHP files** and **scripts**, we must **save** the **file** with the **“.php”** extension. Older PHP file extensions include

- .phtml
- .php3
- .php4
- .php5
- .phps

PHP was designed to work with HTML, and as such, it can be embedded into the HTML code.

```
<HTML> <PHP CODE> </HTML>
```

You can create PHP files without any html tags and that is called Pure PHP file.

The server interprets the PHP code and outputs the results as HTML code to the web browsers.

In order for the server to identify the PHP code from the HTML code, we must always enclose the PHP code in PHP tags.

A PHP tag starts with the less than symbol followed by the question mark and then the words “php”.

PHP is a case sensitive language, “VAR” is not the same as “var”.

The PHP tags themselves are not case-sensitive, but it is strongly recommended that we use lower case letter. The code below illustrates the above point.

```
<?php ... ?>
```

We will be referring to the PHP lines of code as statements. PHP statements end with a semi colon (;). If you only have one statement, you can omit the semi colon. If you have more than one statement, then you must end each line with a semi colon. For the sake of consistency, it is recommended that you always end your statement(s) with a semi colon. PHP scripts are executed on the server. The output is returned in form of HTML.

PHP Hello world

The program shown below is a basic PHP application that outputs the words "Hello World!" When viewed in a web browser.

```
<?php  
echo "Hello world";  
?>
```

Output:

```
Hello world
```

Summary

- PHP stands for Hypertext pre-processor
- PHP is a server side scripting language. This means that it is executed on the server. The client applications do not need to have PHP installed.
- PHP files are saved with the ".php" file extension, and the PHP development code is enclosed in tags.
- PHP is open source and cross platform

[Buy Now \\$9.99](#)