Basic of PHP

version 1.0

Last generated: December 02, 2019

Table of Contents

CRUD in php

Getting started with Php

Summary: This is brief instruction about the requirments that you need to have before you start writing codes on Php.

Installations

You need to have following requirments to run php on your local machine

1. Download and Install Xampp

XAMPP stands for Cross-Platform (X), Apache (A), MariaDB (M), PHP (P) and Perl (P). It is a simple, lightweight Apache distribution that makes it extremely easy for developers to create a local web server for testing and deployment purposes

You can download xampp here

2. Download and Install a code editor

You can have one of these editors to write/edit your code:-

- Visual code studio
- Atom
- Sublime-Text

You can also use other tools as per your prefrence.

Git & Github

1. Github account

GitHub is a web-based hosting service for software development projects that use the Git revision control system.

For beginners it can be a best place to store your code. Learning about version control system in the earlier phase of programming can give you a head-start later phase of coding and developement.

2. Gitbash for windows

Git-Bash is an application for Microsoft Windows environments which provides an emulation layer for a Git command line experience. Bash is a popular default shell on Linux and macOS.

> ℹ **Note:** You don’t have to download git bash if you use Mac or Linux as your main operating system.

[]:

Introduction

Overview

This site provides documentation, training, and other notes for the php programming language. The documentation is designed for people who have no prior knowledge about php.

What is PHP?

- php stands for Hypertext Preprocessor.
- PHP is a server scripting language, and a powerful tool for making dynamic and interactive Web pages.
- PHP is a widely-used, free, and efficient alternative to

What you should know?

Before you get started you should have basic knowledge about following topic:-

- HTML
- CSS
- JS

What is a PHP file ?

PHP files have extension ".php". The code runs on the server and the result is returned to the browser as plain HTML.

A PHP file can contain HTML, CSS, JavaScript and PHP code.

> **Important information**: PHP is the core of wordpress, the biggest blogging platform. It is also used by facebook, the largest social media site. More than anything it is very easy to learn.

Getting started

To get started, see Getting Started (page 3).

[]:

About the author

Summary: This document can be used by instructors as a medium to teach php and students can use this a means to study php.

Introdution to HTML

Summary: HTML one of the major prerequisite you need to have in order to understand php.

What is HTML?

- HTML stands for Hypertext Markup Language.
- HTML is a the markup language designed for creating the webpage.
- HTML describes the structure of the web page.
- HTML elements are represented by tags.
- Browser uses HTML tags to render HTML content.

Note: The file extension for a HTML file is .html

A Basic HTML Document?

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>Heading</h1>
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, se
d do eiusmod tempor incididunt ut labore et dolore magna aliqu
a. Ut enim ad minim veniam, quis nostrud exercitation ullamco l
aboris nisi ut aliquip ex ea commodo consequat. </p>

</body>
</html>
```

DEMO

Understanding the tags

- The `<!DOCTYPE html>` declaration defines this document to be HTML5
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the document
- The `<title>` element specifies a title for the document
- The `<body>` element contains the visible page content
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

⚠ Important: Some of the important in html are `<table>`, `<form>`, `<img>` etc.

If you want to know more about HTML you can check the HTML_tutorial by w3 school.

Introdution to CSS

Summary: CSS is another major prerequisite you need to have in order to understand php.

What is a CSS?

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- External stylesheets are stored in CSS files

> ℹ **Note:** The file extension for an external CSS file is .CSS

Three basic ways of implementing CSS

1. Inline CSS

The inline CSS is written to uniquly style a single element.

Inline styles are defined within the “style” attribute of the relevant element:

Example of inline CSS

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;text-align:center;">This is a heading</h
1>
<p style="color:red;">This is a paragraph.</p>

</body>
</html>
```

DEMO

2. Internal CSS

The internal style is defined inside the `<style>` element, inside the head section.

Exmaple of Internal CSS

```
<!DOCTYPE html>
<html>
<head>
<style>
body {
  background-color: linen;
}

h1 {
  color: maroon;
  margin-left: 40px;
}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

DEMO

3. External CSS

External styles are defined within the `<link>` element, inside the `<head>` section of an HTML page:

Example of External CSS

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Here is how the “mystyle.css” file looks like:

```
body {
  background-color: lightblue;
}

h1 {
  color: navy;
  margin-left: 20px;
}
```

DEMO

⚠ **Important:** A CSS file should not contain html tags

> If you want to know more about CSS you can check the CSS_tutorial by w3 school.

[]:

HTML tables

Summary: This post will give you the brief introduction about HTML tables. However, It won't cover everything about HTML tables but just enough to use it for you PHP lesson.

- A HTML table is defined by `<table>` tag.
- The head is defined by `<th>` tag.
- The row is defined by `<tr>` tag.
- The data is defined with `<td>` tag.

Here is what a simple HTML table looks like:-

Example of simple table

```
<table>
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Charls</td>
    <td>Doe</td>
    <td>22</td>
  </tr>
  <tr>
    <td>Dudly</td>
    <td>Jons</td>
    <td>44</td>
  </tr>
</table>
```

DEMO

However, we can add css properties to enhance the design of the table

Example of a styled table.

```
<head>
 <style>
table, th, td {
  border: 1px solid black;
}
</style>
</head>
<body>
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>Charls</td>
    <td>Doe</td>
    <td>22</td>
  </tr>
  <tr>
    <td>Dudly</td>
    <td>Jons</td>
    <td>44</td>
  </tr>
</table>
```

DEMO

HTML Forms

Summary: This post will give you the brief introduction about HTML forms. The resources available here is just enough to understand this course.

The `<form>` Tag

The html form is defined by `<form>` tag.

Syntax of Form Element

```
<form>
...
Various input tags
...
</form>
```

The `<input>` Tag

The `<input>` element is most important element to make forms in html.

The input element can have various forms that depens on text attribute:-

Some common types of input elements.

1. **text** :- Used to define text field.
2. **radio** :- Used to select one of many choices.
3. **checkbox** :- Used to select Multiple choice.
4. **submit** :- Used to define submitt button for the forms data.
5. **password** :- Used to define password field.
6. **date**:- used to define date field.

Example of Simple form

```
<form action="">
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname"><br>
  email:<br>
  <input type="email" name="email"><br>
  password:<br>
  <input type="password" name="password"><br>
  <input type="submit" value="Submit">
</form>
```

DEMO

The <fieldset> and <legend> tags.

The `<fieldset>` is used to group related date in a form and the `<legend>` tag is used to define the csption fo the fieldset.

Examples of forms using `<fieldset>` and `<legend>`

```
<form action="">
  <fieldset>
    <legend>Fill the Form:</legend>
<form action="/action_page.php">
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname"><br>
  email:<br>
  <input type="email" name="email"><br>
  password:<br>
  <input type="password" name="password"><br>
  <input type="submit" value="Submit">
</form>
  </fieldset>
</form>
```

DEMO

Designed Forms in CSS

You can use various CSS properties to design a form.

Examples of CSS Designed Form

```
<style>
input{
  width: 100%;
  padding: 12px 20px;
  margin: 8px 0;
  display: inline-block;
  border: 1px solid #ccc;
  border-radius: 4px;
  box-sizing: border-box;
}

input[type=submit] {
  width: 100%;
  background-color: red;
  color: white;
  padding: 14px 20px;
  margin: 8px 0;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

</style>


<h3>Using CSS to style an HTML Form</h3>

<form action="">
  <fieldset>
    <legend>Fill the Form:</legend>
<form action="/action_page.php">
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname"><br>
  email:<br>
  <input type="email" name="email"><br>
  password:<br>
  <input type="password" name="password"><br>
  <input type="submit" value="Submit">
</form>
  </fieldset>
</form>
```

DEMO

HTML and CSS practice questions

Summary: Here you can find questions regarding HTML and CSS.

Question 1

Make a simple HTML page. The page must contain your picture information about your hobbies and education. Use External CSS to style your page.Example (page 20)

Question 2

Make a HTML table with 5 entities(columns) and 6 records(rows). Use external CSS to style the table.Example (page 20)

Question 3

Make a HTML form to that takes input as following entities Name(text), Email(email), Phone-number(tel), Gender(radio), Message(textarea), Submit Button(submit). Use appropriate input types. Also, use External CSS to style your page Example (page 20)

echo

> **Summary:** This post will help you understand how to use echo() function on php.

echo is a php function that can be used to print one or many string.

Syntax of echo

```
echo (string);
```

Example-1

A simple echo statement

```
<?php
echo "Welcome to php class.";
?>
```

DEMO

You can also print variables

Example-2 (Variables)

You can also print variables.

> ⚠ **Important:** PHP variables will be discussed later in the documentation. For now just know that a php variable starts with a $ sign.

```
<?php
$str = "Welcome to PHP Class";
echo $str;
?>
```

DEMO

Example-3 (HTML tags)

You can also embed HTML tags in the echo statement.

```
<?php
echo "<h1>This is h1 tag</h1>";
echo "<p>This is p tag</p>";
?>
```

DEMO

Example-4 (Joining)

You can join the two variables together by using `.` as a joining operator:-

```
<?php
$str1 = "Welcome to";
$str2 = "PHP class";
echo $str1 . " " . $str2;
?>
```

DEMO

[]:

Comments

Summary: This post will help you understand how to comment your codes in php.

A comment is the part of the program that is not executed. The comments are only visible to the people who read source code.

Why are comments used ?

- Comments can help readers to understand code.
- Comments can be used to Describe the codes.
- Comments can be used to make codes dormant i.e. making codes unexecutable.

Example-1 (Single line comments)

This example shhows how single line comment is written in php.

```
<?php
// single-line comment by double back slash

# single-line comment by using a hash
?>
```

DEMO

Example-2 (Multiple line comments)

This example shows how multiple line comments are written in php

```
<?php
/* This is a multiple like comment
and can be written in
more than one line.
*/
?>
```

DEMO

Variables

Summary: This post will help you understand what are variables and how to make variables in php.

Variable is a name that has a value associated with it. Variables are used to store values such as strings, numeric values, characters or memory address.

Declaring variables in PHP

Variables in PHP start with a `$` sign followed by the name of the variable.

For a variable to be valid the variable name must start with alphabet (a-z or A-Z) or underscore (_), followed by numbers, letters or underscore.

There cannot be white space in variable so if the variable has more than oneo word it is seperated by commas (for example `$two_words` , `deer_walk`)

Some Valid PHP Variables

Here are some valid php variables:

Example-1

```
<?php
$xyz = 'welcome';
$XYZ = 'welcome';
$Xyz = 'welcome';
$xyz1231 = 'welcome';
$xyZ = 'welcome';
$XYZ = 'welcome';
$_XYZ = 'welcome';
$XYZ_xyz = 'welcome';
?>
```

DEMO

> ⚠ **Important:** PHP variables are case sensitive.

Example-2(Operation)

```
<?php
$x = 20;
$y = 30;
echo $x + $y;
?>
```

DEMO

PHP: A loosly typed language

PHP automatically detects the datatype of the variable by the type of value we give the variable.

The data type is not set in strict manner, that means you can do things like adding strings to intiger without causing an error.

However, PHP 7 allows user to assigne datatype by enabling strict. This gives an option to declare datatype while assigning a variable.

```
<?php
$a = 2;
$b = 2.33;

echo $a + $b;
?>
```

DEMO

> `strict` and `non-strict` requirments will be discussed later in the documentation.

> Global and Local variables will also be discussed later in the

> documentation.

Data Types

> **Summary:** This post will help you understand datatypes in php.

Variables store different datatypes like string, intiger, float, boolean, array.

> ℹ **Note:** var_dump() is a function that returns datatype and value.

String

A string is a group of characters put together, like “deer walk”:

A string is always assigned inside a duble or a single quote:-

```
<?php
$name = "Class of 2022";
var_dump($name)
?>
```

DEMO

Integer

Integer is a datatype with non-decimal numbers. A integer can be either Positive or negetive. Arthmatic opreations can be performed on intigers.

```
<?php
$x = 5985;
$y = 2722;
$z = $x + $y;
var_dump($z);
?>
```

DEMO

Float

A float is a number with decimal or exponential value.

```php
<?php
$x = 12.124;
var_dump($x);
?>
```

DEMO

Boolean

Boolean is used in conditional testing to test if the statement is True or False.

```php
<?php
$true = true;
$false = false;

var_dump($true);
var_dump($false);
?>
```

Demo

Array

Array is used to store multiple valued in a single variable.

```php
<?php
$company = array("google","facebook","twitter");
var_dump($company);
var_dump($company[0]);
var_dump($company[1]);
var_dump($company[2]);
?>
```

DEMO

There are more datatypes. For more information on datatypes check out w3school .

Operators in PHP

Summary: This post can help you undersand about operators in PHP.

Operators are used to perform operatons. The types of operators in php are:

- Arithmetic Operators
- Assignment Operators
- Comparison Operators
- Increament/Decrement Operators
- Logical Operators
- String Operators
- Array Operators
- Conditional Assignment Operators

Arthmetic Operators

Arthmatic operators are used to perform common arithmetical operations such as addition, subtraction, division.

Name (Operators)	Discription	Demo
Addition (+)	Performs Addition	DEMO
Subtraction (-)	Performs Subtraction	DEMO
Multiplication (*)	Performs Multiplication	DEMO
Division(/)	Performs Division	DEMO
Modulus (%)	Finds Reminder	DEMO
Exponentiation (**)	Raise to the power	DEMO

Assignment Operator

Php assignment operators are used with numerical value to write a value to a variable.

Expression	Equivalent	Discription	Demo
a = b	a = b	The value on the left hand side get assigned to the value on the right hand side	DEMO
a += b	a = a + b	Addition	DEMO
a-=b	a = a - b	Subtraction	DEMO

The multiplication, division and modulus will go on in the same way.

Comparision Operators

Operator	Discription	Demo
==	True if two variables are equal	DEMO (page 31)
===	True if two variables are identical i.e. both value and datatype	DEMO (page 31)
!=	True if two variables are not equal	DEMO (page 31)
<>	True if are not equal	DEMO (page 31)
!==	True if two variables are not identical	DEMO (page 31)
>	Greater than	DEMO (page 31)

Operator	Discription	Demo
<	Less than	DEMO (page 31)
>=	Greater or equal to	DEMO (page 31)
<=	Less than or equal to	DEMO (page 31)

Please refere W3school for further information on PHP operators

DUMMY

Summary: dummy

Conditional Statement

Summary: This post will help you understand conditional logic in PHP

Conditional statements are used to perform actions based on conditions.

Types of conditional statements in PHP:-

1. if statement
2. if...else statement
3. if...elseif...else statement
4. switch statement

if Statement

The `if` statement executes codes if one condition is true.

Syntax

```
if (condition) {
    //code;
}
```

Example

```
<?php
$t = 20;

if ($t == "20") {
    echo "I have Rs. 20";
}
?>
```

DEMO

if...else Statement

The `if...else` statement executes some code if the condition is true and another code if the condition is false.

Syntax

```
if (condition) {
   // code1
} else {
   // code2;
}
```

Example

```
<?php
$t = 10;

if ($t == "20") {
    echo "I have Rs. 20";
} else {
    echo "I don't have Rs. 20";
}
?>
```

DEMO

if...elseif...else Statement

If there are more than one condition then we can use `if...elseif...else` condition.

Syntax

```
if (condition) {
    //code1
} elseif (condition) {
    //code2
} else {
    //code3
}
```

Example

```php
<?php
$t = 10;

if ($t == "20") {
    echo "I have Rs. 20";
} elseif($t < 20) {
    echo "I have less than Rs. 20";
} else{
    echo "I have more than Rs. 20";
}
?>
```

DEMO

Switch

Summary: This post will describe about Switch case in php.

The `switch` statement in PHP can be used to perform differnet action based on different cases. If can be used as a substitute for if…elseif

Syntax

```
switch (a) {
    case value1:
        code to be executed if a = value1;
        break;
    case var2:
        code to be executed if a=var2;
        break;
    ...
    default:
        code to be executed if a is different from all var;
}
```

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy

DUMMY

Summary: dummy