

## UNIT-1

### Topics:

**Introduction to PHP:** Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

**File Handling in PHP:** File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.

---

### Introduction

**What is PHP? What are the common uses of PHP?** (2 marks May 2017)

- PHP stands for “PHP: Hypertext Preprocessor”. Earlier it was called Personal Home Page.
- PHP is a server side scripting language that is embedded in HTML.
- The default file extension for PHP files is ".php".
- PHP supports all the databases that are present in the market.
- PHP applications are platform independent. PHP application developed in one OS can be easily executed in other OS also.

### Common uses of PHP:

- PHP performs system functions, i.e. Using System function we can create, open, read, write & close Files.
- Using PHP we can create Dynamic Page Content i.e. anew Feature is added without disturbing the Old feature.
- Collect FORM data- User data is collected to Database i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete and modify elements within your database through PHP.
- Control Access- Using PHP, you can restrict users to access some pages of your website.
- Cookies- We can create, access cookies variables and set cookies.
- PHP can encrypt data.

### Characteristics/Features of PHP:

- Cross Platform-PHP code will be run on every platform, Linux, Unix, Mac OS X, Windows.
- Cross Server- We can access data from different servers on our php application.
- Cross database- PHP supports all the database(Oracle, MySQL, SQLite etc.) that are present in the market.
- Interpreted- It is an interpreted language, i.e. there is no need for compilation.
- Open Source-Open source means you no need to pay for use php, you can free download and use.

## **PHP Basic Syntax-**

<b><u>Universal Style Tag</u></b> A PHP script starts with <b>&lt;?php</b> and ends with <b>?&gt;</b>	<code>&lt;?php</code> <code>// PHP code goes here</code> <code>?&gt;</code>
<b><u>Comments in PHP</u></b>	<code>// This is single-line comment</code> <code># This is also a single-line comment</code> <code>/*</code> This is a multiple-lines comment block that spans over multiple lines <code>*/</code>

## **Output Functions in PHP**

In PHP there are two basic ways to get output: echo and print.

### **1. The PHP echo Statement**

The echo() function outputs one or more strings.

The echo statement can be used with or without parentheses: echo or echo().

**Syntax:** echo(*strings*)

#### **Example**

```
<?php
echo "Hello world!";
echo ("Hello world!");
?>
```

### **2. PHP print() Function :**

**Syntax:** print(*strings*)

#### **Example:**

```
<?php
print "Hiee!";
?>
```

echo and print are used to output data to the screen. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions. echo can take multiple parameters (although such usage is rare) while print can take one argument.

echo is marginally faster than print.

## **Printing Some Text**

```
<html>
<head><title>Displaying TEXT from PHP</title></head>
<body>
<?php
echo 'Welcome to PHP.'; //echo statment using single quote
echo "Welcome to PHP."; //echo statment using double quote
echo ("Welcome to PHP."); //We can pass text inside parenthesis
?>
</body>
</html>
```

## Mixing HTML and PHP

```
<html>           // Page starts with standard <html> <head> <title> section.
<head>
<title>
Using PHP and HTML together
</title>
</head>

<body>           // <body> section
<h1> Using PHP and HTML together</h1> //Contain <h1> header and some text
Here is PHP info: </br></br>
<?php           // <?php starts a PHP section
phpinfo(); //PHP function phpinfo(); displays information about the PHP installation, Every PHP
statement ends with semicolon
?>
</body>
</html>
```

When this page is run by PHP engine on the server HTML will be passed through browser and PHP part will be executed

---

## Variables :

Variables are "containers" that are used for storing information.

In PHP, a variable starts with the \$ sign, followed by the name of the variable.

### What are the rules for naming a variable in PHP?

- Every variable starts with the \$ sign, followed by the name of the variable.
- A variable name must start with a letter or the underscore character(can't 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)

### Storing Data in Variables

PHP variables can hold numbers or strings of characters. We can store information in variables with an assignment operator/ single equal sign (=).

```
<?php
$name="ACE";
$age = 21;
?>
```

After the execution of the statements above, the variable \$name will hold the value ACE, the variable \$age will hold the value **21**.

## **Destroying Variable**

Want to uncreate a variable? We can indeed uncreate a variable in PHP using unset() function. The unset() function destroys a given variable.

```
<?php
$car="TATA";
echo "Before unset(), My car is $car"; //Interpolating String: PHP will place the value held in
$car, inside a double-quoted text String
unset($car);
echo "After unset(), My car is ", $car;
?>
```

## **Creating Variable Variables (\$ and \$\$)**

Lets create a variable named \$name and set the value='Siri'. \$name="Siri"

\$\$name uses the value of the variable \$name

\$\$var is known as reference variable where as \$var is normal variable.

```
<?php
$name="Siri";
$$name="Siri CSEA";
echo $name."<br/>";
echo $$name."<br/>";
echo $$Siri;
?>
```

---

## **Data Types**

**Explain about various data types in PHP (3 marks)**

In other programming language C, C++, JAVA, we need to specify the data type for each variable, but In PHP we don't have to specify data type the variable is. PHP automatically converts the variable to the correct data type, depending on its value.

**PHP supports the following data types:**

- 1) Boolean      Holds true/false values
- 2) Integer Holds non decimal numbers like -1, 0 5 and so on (Positive or negative)
- 3) String    Holds text like "Welcome to PHP" (text inside quotes)
- 4) Float    Holds floating point numbers (also called double) like 3.14159 or 2.7128 with decimal point
- 5) Array    Holds array of data items
- 6) Object    Holds Programming Objects
- 7) Resource      Holds a data resource
- 8) NULL    Holds a value of NULL