

# Swami Sahajanand College of Computer Science

**B.C.A. SEM - V**

**Subject: Web Application Development Using PHP**

## UNIT 1

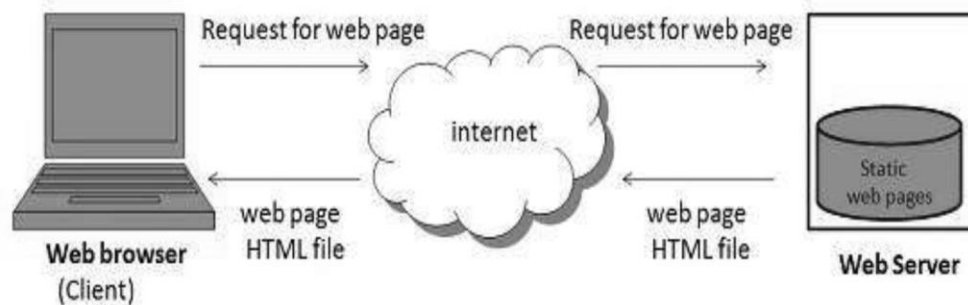
### Introduction

- ◆ Fundamental of webpage, website and apache server
- ◆ Static and Dynamic Website
- ◆ Introduction of PHP-Features, Advantages and Limitations
- ◆ Data Type, Variable, Constant
- ◆ Operator in PHP.

## ♦ Fundamental of webpage, website and apache server.

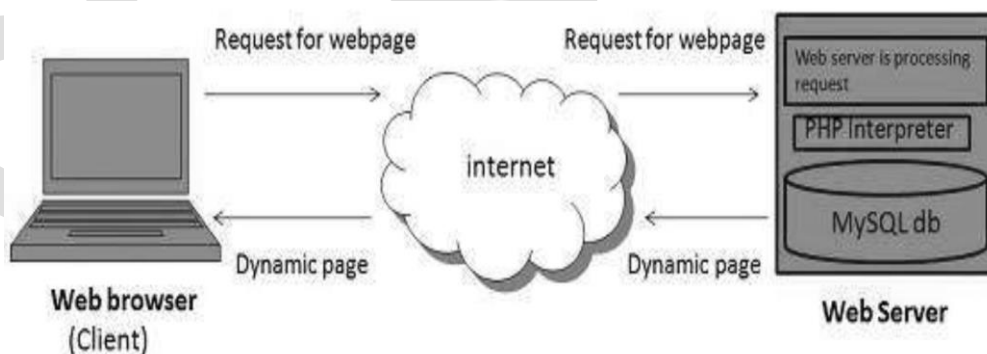
### ♦ 1) Web Page

- ♦ **Web page** is a document available on World Wide Web. Web Pages are stored on web server and can be viewed using a web browser.
- ♦ A web page can contains huge information including text, graphics, audio, video and hyperlinks. These hyperlinks are the link to other web pages
- ♦ Collection of linked web pages on a web server is known as **website**.
- ♦ There is unique **Uniform Resource Locator (URL)** is associated with each web page.
- ♦ **Static Web page**
- ♦ Static web pages are also known as flat or stationary web page. They are loaded on the client's browser as exactly they are stored on the web server. Such web pages contain only static information.
- ♦ User can only read the information but can't do any modification or interact with the information.
- ♦ Static web pages are created using only HTML.
- ♦ Static web pages are only used when the information is no more required to be modified.



### ♦ Dynamic Web page

- ♦ Dynamic web page shows different information at different point of time. It is possible to change a portion of a web page without loading the entire web page.



### ♦ 2) Website:

- ♦ A website is a collection of publicly accessible, interlinked Web pages that share a single domain name. Websites can be created and maintained by an individual, group, business or organization to serve a variety of purposes.
- ♦ A website is also known as a “web presence” or simply “site”.
- ♦ Websites come in a nearly endless variety, including educational sites, news sites, forums, social media sites, e-commerce sites, and so on.

- ◆ The pages within a website are usually a mix of text and other media. That said, there are no rules dictating the form of a website.
- ◆ The homepage (or simply “home”) represents the main page of the site itself. Frequently, the homepage is a sort of “hub” from which all other pages can be accessed.
- ◆ Websites are hosted on servers, and require a web browser such as Chrome, Firefox, or Internet Explorer to be visited (either on a computer or mobile device).
- ◆ A website can be accessed directly by entering its URL address or by searching it on a search engine such as Google or Bing.
- ◆ Originally, websites were categorized by their top-level domains. Some examples include:
  - Government agency websites = .gov
  - Educational institutions' websites = .edu
  - Nonprofit organizations' websites = .org
  - Commercial websites = .com
  - Information sites = .info
- ◆ The first website was created in 1990 by Tim Berners-Lee, a British physicist at CERN.
- ◆ 3 years later, in 1993, CERN announced that everyone could access and use the World Wide Web for free.
- ◆ **3) Apache:**
- ◆ Apache is one of the most popular web servers.
- ◆ It is more popular as simply Apache.
- ◆ It represents an open-source web server platform.
- ◆ It is base of most of the websites we see today on the World Wide Web.
- ◆ Apache was introduced in mid 90's.
- ◆ It then gradually adopted as a preferred server platform on the web.
- ◆ Apache is the main driving force behind today's web expansion.
- ◆ As a web server 'pioneer', Apache is standard for the development of other successful web server platforms.
- ◆ It was developed by Apache Software Foundation. It is actually a type of open source community.
- ◆ Its development is backed up by the efforts of many supporters worldwide.
- ◆ They make it well maintained and regularly updated with new useful features and functionalities up to the latest quality and security requirements in HTTP service delivery.
- ◆ Apache – a Cross Platform Web Server.
- ◆ The wide use of the Apache server on the web is visible through the multitude of web platforms and operating systems it is working with.
- ◆ The application is available for a wide variety of operating systems, including Unix, FreeBSD, Linux, Solaris, Novell NetWare, OS X, Microsoft Windows, OS/2, TPF, and eComStation.
- ◆ Most of them are well established and extremely popular among users, including Windows, Linux, Unix, Solaris, Mac OS X, Microsoft Windows, NetWare, FreeBSD, OS/2.
- ◆ The Apache server is distributed with a rich set of modules, allowing for users to run miscellaneous scripts and applications on it.
- ◆ The reasons behind its popularity, to name a few, are:
  - It is free to download and install.
  - It is open source: the source code is visible to anyone and everyone, which basically enables anyone (who can rise up to the challenge) to adjust the code, optimize it, and fix errors and security holes. People can add new features and write new modules.

- It suits all needs: Apache can be used for small websites of one or two pages, or huge websites of hundreds and thousands of pages, serving millions of regular visitors each month. It can serve both static and dynamic content.

### ◆ **Features of Apache Server**

- ◆ support for CGI (Common Gateway Interface) and SSI (Server Side Includes)
- ◆ URL redirection
- ◆ Automatic directory listings.
- ◆ user authentication
- ◆ anonymous user access
- ◆ loading modules support
- ◆ HTTP header metafiles,
- ◆ proxy caching abilities
- ◆ Virtual hosting.
- ◆ Apache allows one Apache installation to serve many different actual websites.
- ◆ For example, one machine with one Apache installation could simultaneously serve www.example.com, www.example.org, etc.
- ◆ Apache features configurable error messages.
- ◆ DBMS-based authentication databases and content negotiation.
- ◆ It is also supported by several graphical user interfaces (GUIs).
- ◆ It supports password authentication and digital certificate authentication.
- ◆ Apache has a built in search engine and an HTML authorizing tool and supports FTP

### ◆ **How Apache Works**

- ◆ The Apache server is set up configuration files to run. In which contents are added to control its behavior. In its idle state, Apache listens to the IP addresses identified in its config file (HTTPd.conf). Whenever it receives a request, it analyzes the headers, applies the rules specified for it in the Config file, and takes action.
- ◆ But one server can host many websites. To achieve this, every one of those websites has to be assigned a different name.
- ◆ Since IP addresses are difficult to remember, we, as visitors to specific sites, usually type in their respective domain names into the URL address box on our browsers. The browser then connects to a DNS server, which translates the domain names to their IP addresses. The browser then takes the returned IP address and connects to it. The browser also sends a Host header with the request so that, if the server is hosting multiple sites, it will know which one to serve back.
- ◆ For example, typing in www.google.com into your browser's address field might send the following request to the server at that IP address:
- ◆ GET / HTTP/1.1
- ◆ Host: www.google.com
- ◆ The first line contains several pieces of information. First, there is the method (in this case it's a GET), the URI, which specifies which page to be retrieved or which program to be run, and finally there is the HTTP version.
- ◆ HTTP is a request / response stateless protocol. It's a set of rules that govern communication between a client and the server. The client makes a request, the server sends back a response, and communication stops.
- ◆ If the request is successful, the server returns a 200 status code (which means that the page is

found), response headers, along with the requested data. The response header of an Apache server might look something like the following:

- ◆ HTTP/1.1 200 OK
- ◆ Date: Sun, 10 Jun 2012 19:19:21 GMT
- ◆ Server: Apache
- ◆ Cache-Control: no-cache
- ◆ Last-Modified: Sun, 10 Jun 2012 19:19:21 GMT
- ◆ Content-Type: text/html
- ◆ Content-Length: 7560
- ◆ If the request didn't go throw, the client would get an error code and message, such as the following response header in case of a page not found error:
- ◆ HTTP/1.1 404 Not Found

### ◆ Static Website and Dynamic Website

- ◆ **Static Website:**
- ◆ A static website contains Web pages with fixed content.
- ◆ Each page is coded in HTML and displays the same information to every visitor.
- ◆ Static sites are the most basic type of website and are the easiest to create.
- ◆ They do not require any Web programming or database design.
- ◆ A static site can be built by simply creating a few HTML pages and publishing them to a Web server.
- ◆ Since static Web pages contain fixed code, the content of each page does not change unless it is manually updated by the webmaster.
- ◆ This works well for small websites, but it can make large sites with hundreds or thousands of pages difficult to maintain.
- ◆ Therefore, larger websites typically use dynamic pages, which can be updated by simply modifying a database record.
- ◆ Static sites that contain a lot of pages are often designed using templates.
- ◆ This makes it possible to update several pages at once, and also helps provide a consistent layout throughout the site.

### Static Website



Server



Client/Browser

### ◆ Advantages of Static Website

#### ◆ Security

- ◆ As there's no intermediary, i.e., the database involved, the threat of code injection is minimal for a static website. No plugins and dynamic software to host the website.

#### ◆ Reliability

- ◆ You must have occasionally come across an error message saying that, "The connection could not be established." This primarily occurs because of the database errors. Serving just basic HTML files makes it easier to host them anywhere with ease like on a CDN.

#### ◆ Speed

- ◆ The absence of middleman/database makes the static site much more speedy and easy to load.
- ◆ A static site is ten times faster than a dynamic site that is built with a CMS.
- ◆ Another reason for a static website to run faster is that it can be served from the node closest to the browser.

#### ◆ Hosting and Price

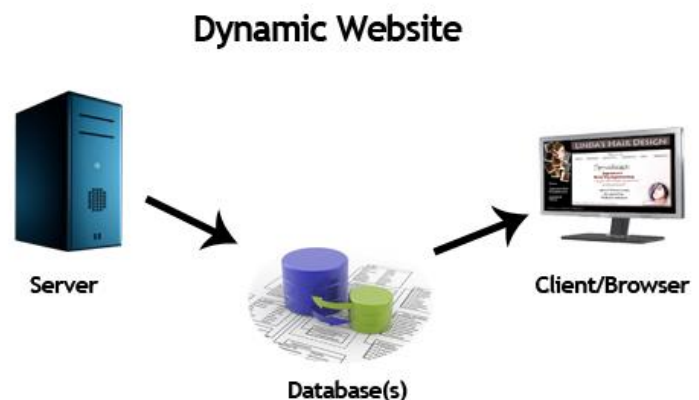
- ◆ Static websites have basic HTML files which require less space making the hosting of these websites cheaper to that of dynamic websites. Organizations with static website save up on the costs and channel those resources to integrate Git or automated builds to incorporate the latest changes in the system.

#### ◆ Scalability

- ◆ What to do when your website is finally up and running? Handling massive traffic on a dynamic website might be a cumbersome process as it requires complex code playing on the server. Basic static websites with HTML files can be easily scaled up by just increasing the bandwidth.

### ◆ Dynamic website:

- ◆ Dynamic website is a collection of dynamic web pages whose content changes dynamically.
- ◆ It accesses content from a database or Content Management System (CMS).
- ◆ Therefore, when you alter or update the content of the database, the content of the website is also altered or updated.
- ◆ Dynamic website uses client-side scripting or server-side scripting, or both to generate dynamic content.
- ◆ Client side scripting generates content at the client computer on the basis of user input.
- ◆ Web browser downloads the web page from the server and processes the code within the page to render information to the user.
- ◆ In server side scripting, the software runs on the server and processing is completed in the server then plain pages are sent to the user.





### ◆ Advantages of Dynamic Websites

#### ◆ Easier design updates

- ◆ Since each part of a web page is separate, it is much simpler to update something across many pages, all at once.
- ◆ For example, if you have a change to your website's navigation you only have to change it in one place (the file that holds navigation information) and it is updated on all pages that contain that file.
- ◆ If you had a static website, you would have to edit every single page in order to make a change to the site's navigation. This is tedious, repetitive, and prone to errors.

#### ◆ More flexible data

- ◆ Because a dynamic site pulls together a bunch of page to make a whole page, you can store your content (and other parts of your site) in a database.
- ◆ The advantage of this is that you can easily access and edit your content in a variety of ways or even have it load across multiple websites (for example, if you had two distinct brands with some overlapping content.)
- ◆ Databases also make it easy to search your content, categorize it, load it in dynamic ways (such as displaying the 5 most recent entries about sock puppets), and back it up.
- ◆ All of this is nearly impossible with a static site.

#### ◆ Easier content updates

- ◆ Dynamic websites with a content management system (CMS) make it simple for a non-technical person to create and update the content of the site. Because the various parts of the page are all separate, a content creator won't need to know any HTML in order to create a new page or article for the site.
- ◆ You can simply concentrate on writing the content and the dynamic site takes care of putting that content into the right spot on the website. With a static site the content creator would have to know HTML or employ the help of someone who does in order to create a new web page for a site.

#### ◆ Difference Between Static & Dynamic Website

Static Website	Dynamic Website
Prebuilt content is same every time the page is loaded.	Content is generated quickly and changes regularly.
It uses the <b>HTML</b> code for developing a website.	It uses the server side languages such as <b>PHP, SERVLET, JSP, and ASP.NET</b> etc. for developing a website.
It sends exactly the same response for every request.	It may generate different HTML for each of the request.
The content is only changed when someone publishes and updates the file (sends it to the web server).	The page contains "server-side" code which allows the server to generate the unique content when the page is loaded.
Flexibility is the main advantage of static website.	Content Management System (CMS) is the main advantage of dynamic website.

## ♦ Introduction of PHP-Features, Advantages and Limitations

### ♦ **Introduction**

- ♦ Full form of PHP is Hypertext Preprocessor. It was previously known as personal home page.
- ♦ PHP was originally created by Rasmus Lerdorf in 1995.
- ♦ PHP is one type of server side scripting language.
- ♦ So PHP code execute on the server rather than on client.
- ♦ PHP is basically used to create dynamic & interactive web pages.
- ♦ Dynamic web page's content varies according to place, time, user and behavior of user.
- ♦ To generate such type of web site or web page one need to use server side scripting language and many more tools & technology like HTML, JAVASCRIPT, CSS, AJAX etc.
- ♦ PHP is one of the most popular and widely used accepted server sides scripting language.
- ♦ PHP is an Interpreted language, hence it doesn't need a compiler.
- ♦ To run and execute PHP code, we need a Web server on which PHP must be installed.
- ♦ PHP is a server side scripting language, which means that PHP is executed on the server and the result is sent to the browser in plain HTML.
- ♦ PHP is open source and free.
- ♦ PHP has in-built support for MySQL, which is one of the most widely used Database management system.
- ♦ It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server
- ♦ PHP supports a large number of major protocols such as POP3, IMAP, and LDAP.

### ♦ **Features of PHP:**

- ♦ It is most popular and frequently used worldwide scripting language.

#### **Simple**

- ♦ It is very simple and easy to use, compare to other scripting language it is very simple and easy, this is widely used all over the world.

#### **Interpreted**

- ♦ It is an interpreted language, i.e. there is no need for compilation.

#### ♦ **Faster**

- ♦ It is faster than other scripting language e.g. asp and jsp.

#### **Open Source**

- ♦ Open source means you no need to pay for use php, you can free download and use.

#### **Platform Independent**

- ♦ PHP code will be run on every platform, Linux, Unix, Mac OS X, Windows.

#### **Case Sensitive**

- ♦ PHP is case sensitive scripting language at time of variable declaration. In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive.

#### **Error Reporting**

- ♦ PHP have some predefined error reporting constants to generate a warning or error notice.

#### **Loosely Typed Language**

- ♦ PHP supports variable usage without declaring its data type. It will be taken at the time of the execution based on the type of data it has on its value.

#### **Familiarity**

- ♦ If you are in programming background then you can easily understand the PHP syntax. And



you can write PHP script because of most of PHP syntax inherited from other languages like C or Pascal.

**Efficiency**

- ◆ PHP 4.0 introduced resource allocation mechanisms and Eliminating unnecessary memory allocation, in addition to session management features.

**Security**

- ◆ Several trusted data encryption options are supported in PHP's predefined function set. You can use a lot of third-party applications to secure our data, allowing for securing our application.

**Flexibility:**

- ◆ You can say that PHP is a very flexible language because of PHP is an embedded language you can embed PHP scripts with HTML, JAVA SCRIPT, WML, XML, and many others. You can run your PHP script any device like mobile Phone, tabs, laptops, PC and other because of PHP script execute on the server then after sending to the browser of your device.

**Free**

- ◆ PHP is an open source programming language so you can download freely there is no need to buy a license or anything.

**Object Oriented**

- ◆ PHP has added some object-oriented programming features, and Object Oriented programming became possible with PHP 4. With the introduction of PHP 5, the PHP developers have really beefed up the object-oriented features of PHP, resulting in both more speed and added features.

- ◆ **Advantages PHP:**

**Open Source**

- ◆ PHP is open-source and free of cost, which helps developers to install it quickly and readily available for use.
- ◆ There are a lot of PHP frameworks and developer can choose any of the frameworks to work.
- ◆ As it is open-source, it makes the system ready with PHP in quick time and makes the web development faster with the help of providing the tools and other features easily.

**Platform Independent**

- ◆ PHP is mainly supported by all the operating systems like Windows, Unix, Linux etc.
- ◆ The PHP based developed web applications can be easily run on any platform. It can be integrated with other programming language and database easily.

**Simple and Easy**

- ◆ This advantage of PHP is simple and easy to learn and code. It is mainly organized code and clean, which helps the new developers also.
- ◆ The command functions of PHP can easily learn and understood. The one who knows any programming language can easily work on PHP.

**Database**

- ◆ PHP is easily connected with the database and make the connection securely with databases. It has a built-in module that is used to connect to the database easily. There are many web applications, which require strong programming language with a good database management system. PHP and its database connection solve the purpose for development of web applications. It reduces the time to connect to a database management system as well. Multiple databases can be integrated with PHP.

**Fast**

- ◆ PHP is known as the fastest Programming language as compared to another. PHP applications can be easily loaded over the slow Internet and data speed. Other applications take a lot of time to connect the database and fetch the data after executing certain queries to the database. PHP does not face this problem and it loads the website very easily and fast. The fast speed of PHP provides the developer with an edge to develop the web applications in PHP programming language.

**Maintenance**

- ◆ PHP framework is mainly used to make the web application development easier and maintain the code automatically. The model view controller architecture in PHP framework helps the code to be easily maintained and used. The MVC architecture helps the separation of a file for different module separately.

**Support**

- ◆ This advantage of PHP has great online support and community, which helps the new developers to help in writing the code and developing the web applications. The documentation provided at the official site helps in using the different features of PHP and its framework. The latest updates are released timely by the PHP to make it better for the developer to develop the web-based applications.

**Testing**

- ◆ PHP based web applications can be easily tested. PHP unit uses to perform the unit testing quickly and easily. It also helps the programmers to write test cases and perform the testing smoothly. For PHP based web applications, the developers do not need to write the additional code. PHP frameworks help in automating the different tasks.

**Security**

- ◆ PHP frameworks built-in feature and tools make it easier to protect the web applications from the outer attacks and security threats. The security threats can be like SQL injection, data tampering, and forgery etc. To protect from these security threats, developers used PHP frameworks for developing web applications.

**Stable**

- ◆ PHP is also stable as compared to other programming languages. It has been in existence for a long time. The developers have worked on PHP to make it easy for the programmers to work on developing the PHP web-based applications. They have fixed the issues and bugs over the period of time for the different version of PHP and make it very stable.

- ◆ **Limitations of PHP:**

**Security**

- ◆ Since it is open sourced, all people can see the source code. If there are bugs in the source code, it can be used by people to explore the weakness of it.

**Not suitable of large applications**

- ◆ It will be difficult to use it for programming huge applications. Since the programming language is not highly modular, huge applications created out of the programming language will be difficult to maintain.

**Weak type**

- ◆ Implicit conversion may surprise unwary programmers and lead to unexpected bugs. Confusion between arrays and hash tables. This is slow and could be faster. There are often a few ways to accomplish a task. It is not strongly typed. It is interpreted and uses curly braces.

**Poor Error Handling Method**

- ◆ The framework has a bad error handling method. It is not a proper solution for the developers. Therefore, as a qualified PHP developer, you will have to overcome it.

**PHP is unable to handle large number of apps**

- ◆ The technology is helpless to support a bunch of apps. It is highly tough to manage because, it is not competent modular. It already imitates the features of Java language.

**◆ Data Type, Variable, Constant****◆ Basic Syntax**

- ◆ PHP script always starts with <?php and ends with ?>.
- ◆ A PHP script can be placed anywhere in the document.
- ◆ It can be used any number of time in web page.
- ◆ It has following delimiter to start and complete php script
- ◆ <?php ?>
- ◆ One can embed PHP code with html code or can create pure PHP web page that contains only php code.
- ◆ However second option is required very few times.
- ◆ A PHP file must have a .php extension.
- ◆ A PHP file normally contains HTML tags, and some PHP scripting code.
- ◆ Below, we have an example of a simple PHP script which sends the text "Hello World" to the Browser:

```
<Html>
<Body>
<?php
    echo "Hello World";
?>
</body>
</html>
```

**◆ Output: Hello World**

- ◆ Each code line in PHP must end with a semicolon. The semicolon is a separator and is used to distinguish one set of instructions from another.
- ◆ There are two basic statements to output text with PHP: echo and print. In the example above we have used the echo statement to output the text "Hello World".

**◆ Comments in PHP**

- ◆ In PHP, we use // to make a single-line comment or /\* and \*/ to make a large comment block.

```
<Html>
<Body>
<?php
//This is a comment
/* This is a comment block */
?>
</body>
</html>
```

### ◆ 1) Data types

- ◆ PHP is loosely typed server side scripting language.
- ◆ It means that it automatically determines the data type at the time data is assigned to each variable.
- ◆ A variable can hold a string and then later in the script it can hold an integer or some other data type.
- ◆ However there are eight standard data type in php.
- ◆ **Standard Data Types**

Type	Example	Description
Boolean	true	One of the special values true or false
Integer	5	A whole number
Float or double	3.234	A floating-point number
String	"hello"	A collection of characters
Object		An instance of a class
Array		An ordered set of keys and values
Resource		Reference to a third-party resource (database)
NULL		An uninitialized variable

### ◆ 2) Variables in PHP

- ◆ Variables are used for storing values, such as numbers, strings or functions results, so that they can be used many times in a script.
- ◆ Variables are used for storing values, like text strings, numbers or arrays.
- ◆ When a variable is set it can be used over and over again in your script.
- ◆ All variables in PHP start with a \$ sign symbol.
- ◆ The correct way of setting a variable in PHP:
- ◆ **\$var\_name = value;**
- ◆ New PHP programmers often forget the \$ sign at the beginning of the variable. In that case it will not work.
- ◆ Let's try creating a variable with a string, and a variable with a number:
 

```
<?php
$txt = "Hello World!";
$number = 16;
?>
```

### ◆ **Variable Naming Rules**

- ◆ A variable name must start with a letter or an underscore "\_"
- ◆ A variable name can only contain alpha-numeric characters and underscores (a-Z, 0-9, and \_)
- ◆ A variable name should not contain spaces. If a variable name is more than one word, it should be separated with underscore (\$my\_string), or with capitalization (\$myString) PHP String.
- ◆ Variable names are case sensitive in PHP. It means \$rupee and \$Rupee are different variables.
- ◆ A string variable is used to store and manipulate a piece of text.

### ◆ **PHP echo and print Statements:**

- ◆ echo and print are more or less the same.

- ◆ They are both 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.
- ◆ **PHP echo Statement**
- ◆ The echo statement can be used with or without parentheses: echo or echo().
- ◆ The following example shows how to output text with the echo command (notice that the text can contain HTML markup):
- ◆ **Example:**  

```
<?php  
echo "Hello world!<br>";  
echo "This ", "string ", "was ", "made ", "with multiple parameters.";
```
- ◆ **Output:**
- ◆ Hello world!
- ◆ This string was made with multiple parameters.
- ◆ **PHP print Statement**
- ◆ The print statement can be used with or without parentheses: print or print().
- ◆ The following example shows how to output text with the print command (notice that the text can contain HTML markup):
- ◆ **Example:**  

```
<?php  
print "Hello world!<br>";
```
- ◆ **Output:**
- ◆ Hello world!
- ◆ **3) PHP Constants:**
- ◆ Constants are like variables except that once they are defined they cannot be changed or undefined.
- ◆ A constant is an identifier (name) for a simple value.
- ◆ The value cannot be changed during the script.
- ◆ A valid constant name starts with a letter or underscore (no \$ sign before the constant name).
- ◆ Note: Unlike variables, constants are automatically global across the entire script.
- ◆ PHP constants can be defined by 2 ways:
- ◆ Using define() function
- ◆ Using const keyword
- ◆ **PHP constant: define()**
- ◆ To create a constant, use the define() function.
- ◆ Let's see the syntax of define() function in PHP.
- ◆ **define(name, value, case-insensitive)**

**◆ Parameters:**

- **name:** Specifies the name of the constant
- **value:** Specifies the value of the constant
- **case-insensitive:** Default value is false. It means it is case sensitive by default.

◆ Let's see the example to define PHP constant using define().

◆ The example below creates a constant with a case-sensitive name:

**◆ Example**

```
<?php
define("GREETING", "Hi This is Manish");
echo GREETING;
?>
```

**◆ Output**

◆ Hi! This is Manish

◆ The example below creates a constant with a case-insensitive name:

**◆ Example**

```
<?php
define("GREETING", "Hi! This is Manish", true);
echo greeting;
echo GREETING;
?>
```

**◆ Output**

Hi! This is Manish  
Hi! This is Manish

**◆ PHP constant: const keyword**

◆ The **const** keyword defines constants at compile time. It is a language construct not a function.

◆ It is bit faster than define().

◆ It is always case sensitive.

**◆ Example**

```
<?php
Const GREETING=" Hi! This is Manish";
echo GREETING;
?>
```

**◆ Output**

Hi! This is Manish

**◆ Constants are Global**

◆ Constants are automatically global and can be used across the entire script.

◆ The example below uses a constant inside a function, even if it is defined outside the function:

**◆ Example**

```
<?php
define("GREETING", "Hi! This is Manish");
function myTest() {
echo GREETING;
}
```



```
myTest();
```

```
?>
```

#### ◆ **Output**

```
Hi! This is Manish
```

### ◆ **PHP Operators:**

- ◆ Operators are used to perform operations on variables and values.

- ◆ PHP divides the operators in the following groups:

- Arithmetic operators
- Assignment operators
- Comparison operators
- Increment/Decrement operators
- Logical operators
- String operators
- Array operators

#### ◆ **Arithmetic Operators**

- ◆ The PHP arithmetic operators are used with numeric values to perform common arithmetical operations, such as addition, subtraction, multiplication etc.

Operator	Name	Example	Result
+	Addition	$\$x + \$y$	Sum of $\$x$ and $\$y$
-	Subtraction	$\$x - \$y$	Difference of $\$x$ and $\$y$
*	Multiplication	$\$x * \$y$	Product of $\$x$ and $\$y$
/	Division	$\$x / \$y$	Quotient of $\$x$ and $\$y$
%	Modulus	$\$x \% \$y$	Remainder of $\$x$ divided by $\$y$
**	Exponentiation	$\$x ** \$y$	Result of raising $\$x$ to the $\$y$ 'th power (Introduced in PHP 5.6)

#### ◆ **Assignment Operators**

- ◆ The PHP assignment operators are used with numeric values to write a value to a variable.
- ◆ The basic assignment operator in PHP is "=". It means that the left operand gets set to the value of the assignment expression on the right.

Assignment	Same as...	Description
$x = y$	$x = y$	The left operand gets set to the value of the expression on the right
$x += y$	$x = x + y$	Addition
$x -= y$	$x = x - y$	Subtraction
$x *= y$	$x = x * y$	Multiplication
$x /= y$	$x = x / y$	Division
$x \% = y$	$x = x \% y$	Modulus

#### ◆ **Comparison Operators**

- ◆ The PHP comparison operators are used to compare two values (number or string):

Operator	Name	Example	Result
==	Equal	$\$x == \$y$	Returns true if $\$x$ is equal to $\$y$
===	Identical	$\$x === \$y$	Returns true if $\$x$ is equal to $\$y$ , and they are of the same type

!=	Not equal	\$x != \$y	Returns true if \$x is not equal to \$y
<>	Not equal	\$x <> \$y	Returns true if \$x is not equal to \$y
!==	Not identical	\$x !== \$y	Returns true if \$x is not equal to \$y, or they are not of the same type
>	Greater than	\$x > \$y	Returns true if \$x is greater than \$y
<	Less than	\$x < \$y	Returns true if \$x is less than \$y
>=	Greater than or equal to	\$x >= \$y	Returns true if \$x is greater than or equal to \$y
<=	Less than or equal to	\$x <= \$y	Returns true if \$x is less than or equal to \$y

#### ◆ Increment / Decrement Operators

- ◆ The PHP increment operators are used to increment a variable's value. The PHP decrement operators are used to decrement a variable's value.

Operator	Name	Description
++\$x	Pre-increment	Increments \$x by one, then returns \$x
\$x++	Post-increment	Returns \$x, then increments \$x by one
--\$x	Pre-decrement	Decrements \$x by one, then returns \$x
\$x--	Post-decrement	Returns \$x, then decrements \$x by one

#### ◆ Logical Operators

- ◆ The PHP logical operators are used to combine conditional statements.

Operator	Name	Example	Result
And	And	\$x and \$y	True if both \$x and \$y are true
or	Or	\$x or \$y	True if either \$x or \$y is true
xor	Xor	\$x xor \$y	True if either \$x or \$y is true, but not both
&&	And	\$x && \$y	True if both \$x and \$y are true
	Or	\$x    \$y	True if either \$x or \$y is true
!	Not	!\$x	True if \$x is not true

#### ◆ String Operators

- ◆ PHP has two operators that are specially designed for strings.

Operator	Name	Example	Result
.	Concatenation	\$txt1. \$txt2	Concatenation of \$txt1 and \$txt2
.=	Concatenation assignment	\$txt1 .= \$txt2	Appends \$txt2 to \$txt1

#### ◆ Array Operators

- ◆ The PHP array operators are used to compare arrays.

Operator	Name	Example	Result
+	Union	\$x + \$y	Union of \$x and \$y
==	Equality	\$x == \$y	Returns true if \$x and \$y have the same key/value pairs
===	Identity	\$x === \$y	Returns true if \$x and \$y have the same key/value pairs in the same order and of the same types
!=	Inequality	\$x != \$y	Returns true if \$x is not equal to \$y
<>	Inequality	\$x <> \$y	Returns true if \$x is not equal to \$y
!==	Nonidentity	\$x !== \$y	Returns true if \$x is not identical to \$y