Unit 9 PHP-Part – 1

Structure:

9.1 Introduction

In the previous units, we have discussed about DHTML, DHTML is a consultant for a collection of technologies used together to create interactiveand animated web sites by using a combination of static markup language,a client programming language, a presentation definition language and document object model. In this unit you will learn PHP. PHP is an HTML- embedded scripting language. PHP stands for Hypertext

Preprocessors. Yes, it is recursive abbreviation. PHP is a scripting language, and is very
similar to C, Java and Perl. PHP allows the programmer to dynamically generate content, instead of statically like regular HTML. The goal of PHP is to allow web developers to write dynamically generated page quickly.

If you are plotting to develop your own website, PHP is one of the best technologies which provide dynamism for your web site. PHP provides a secure, open source and well-formed web development frame work.

In this unit you will learn history of PHP, Installation steps, and also you will learn, how to write a PHP program, and then some PHP functions like how to pass parameter in a function Get and Post function of PHP.

Objectives:

After studying this unit, you should be able to:

- describe PHP
- use array with PHP
- define operators of PHP
- list out different control flow statements
- define various functions of PHP

9.1.1 History and framework of PHP

The first version of PHP was developed by Rasmus Lerdorf in 1995. Previously it was known as Perl scripts which he called Personal Home Page Tools. Later the tools were rewritten in C in order to give them more functionality. This set of tools was called PHP/FI or Personal Home Page/Form Interpreter. PHP/FI went through two revisions before it was completely rewritten by Andi Gutmans and Zeev Suraki in 1997. PHP version 3.0 was the first version to carry the name PHP Hypertext Preprocessor. It was also the first version to have support for different databases, protocols and APIs.

PHP has gone through two more revisions. In 2000, PHP 4 was released based on the Zend Engine. The Zend Engine is an open source scripting engine also developed by Andi Gutmans and Zeev Suraki. This version provided support for more web servers and more secure ways to handle user input.

9.1.2 Integration of PHP into Web Environment

Hypertext Pre-processor (PHPs) is a server-side scripting language, and server-side scripts are special commands we can place in Web pages. Those commands are processed before the pages are sent from our Server to the Web browser of our users. A typical PHP files will content commands to be executed in the server in addition to the usual mixture of text and HTML (Hypertext Markup Language) tags. PHP is a powerful tool for makingdynamic and interactive Web pages.

When we type a URL in the Address box or click a link on a Web page, we're asking a Web server on a computer somewhere to send a file to the Web browser on our computer. If that file is a normal HTML file, it looks exactly the same when our Web browser receives it as it did before the Webserver sent it. After receiving the file, our Web browser displays its contents as a combination of text, images, and sounds. In the case of PHP page, the process is similar, except there's an extra processing step that takes place just before the Web server sends the file. Before Web server sends the PHP file to the Web browser, it runs all server-side scripts contained in the page. Some of these scripts display the current date, time, and other information. Other process information the user has just typed into a form, such as a page in the Web site.

PHP programs are written using a text editor, such as Notepad, Simple Text, or vi (Visual editor (vi) is a screen oriented text editor), just like HTML pages. However, unlike HTML, PHP files end with a .php extension. This extension signifies to the server that it needs to parse the PHP code before sending the resulting HTML code to the viewer’s web browser. In PHP, on the fly method is adopted to publish the document. Hence, the PHP developer can generate not only web pages, but also other web embedding documents like PDF, PNG, GIF, etc. The PHP web environment is usually set with AMP (Apache, MySQL, and PHP/Perl/Python), which are linked together.

How PHP works?

When a user navigates in the browser to a page that ends with a .php extension, the request is sent to a web server, which directs the request to the PHP interpreter. As shown in the diagram below, the PHP interpreter processes the page, communicating with file systems, databases, and email servers as necessary, and then delivers a web page to the web server to return to the browser. Figure 9.1 shows the working of PHP.

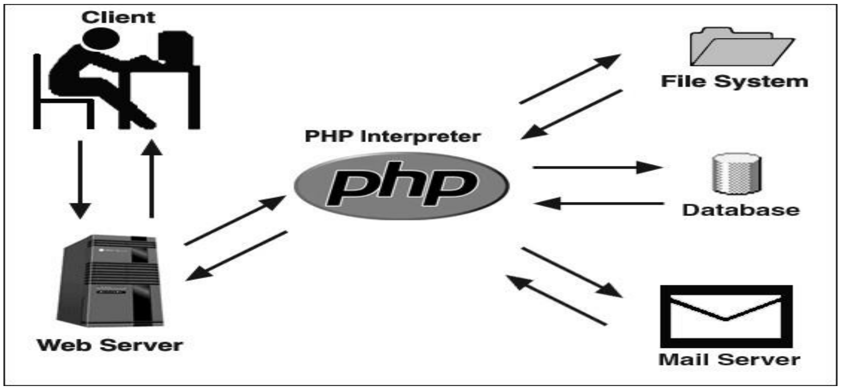

Figure 9.1: Working of PHP

PHP will allow us to:

- Reduce the time to create large websites.
- Create a customized user experience for visitors based on information that we have gathered from them.
- Open up thousands of possibilities for online tools. Check out PHP – HotScripts for examples of the great things that are possible with PHP.
- Allow creation of shopping carts for e-commerce websites**.**
- Display date, time, and other information in different ways.
- To make a survey form and ask people who visit your site to fill it out, send emails, save the information to a file, etc.

How PHP Fits with HTML?

PHP not only allows HTML pages to be created on the fly, but it is invisible to your web site visitors. The only thing they see when they view the source of your code is the resulting HTML output. In this respect, PHP gives you a bit more security by hiding your programming logic.

HTML can also be written inside the PHP code of your page, which allows you to format text while keeping blocks of code together. This will give you efficient and organized code. But the person, who is viewing the site, is unable to know the code difference.

PHP can also be written as a standalone program with no HTML at all. This is helpful for storing your connection variables, redirecting your visitors to another page of your site, or performing other functions.

How PHP Fits with MySQL?

MySQL easily fits into server-side programming languages, through a domain. Before your MySQL functions will be recognizable, make sure to enable MySQL in your php.ini file. You can use MySQL commands within PHP code almost as seamlessly as you do with HTML.

Self Assessment Questions

1. PHP stands for___________
2. PHP is a__________scripting language.

9.2 PHP Installation

Three vital components need to be installed on our system to develop and run PHP web pages. These are Webserver, database and PHP parser.

1) Web server

PHP will work with virtually all web server software, including Microsoft's Internet Information Server (IIS) but then most often used is freely available Apache server.

2) Database

PHP will work with virtually all database software, including Oracle and Sybase but most commonly used is freely available MySQL database.

3) PHP Parser

Inorder to process PHP script instructions a parser must be installed to generate HTML output that can be sent to the Web browser.

9.2.1 PHP Parser Installation

It is important to make sure that a proper environment setup is there in our machine to develop web programs using PHP. Type the address ***http://127.0.0.1/info.php*** into the browser's address box and if it displays a page showing the PHP installation related information then it means that PHP and webserver installed properly in our system. Otherwise we have to follow the procedures given below to install PHP.

9.2.2 PHP installation on Linux or UNIX with Apache

Given below are the steps to install Apache and PHP5 on a Linux or UNIX machine.

1) First, download the files and then unzip and untar Apache source distribution and save it in the location /usr/local.

   ```
    gunzip -c apache_1.3.x.tar.gz
   tar -xvf apache_1.3.x.tar
   ```

2) Then build the Apache server using the following steps,

   ```
   cd apache_1.3.x
   ./configure --prefix=/usr/local/apache --enable-so
   make
   make install
   ```

3) Configure and build PHP, assume that we are using MySQL database.

   ```
   ./configure --with-apxs=/usr/sbin/apxs\
               --with-mysql=/usr/bin/mysql
   make
   make install
   ```

4) Install the php.ini file and edit the file to get configuration directives

   ```
   cd ../../php-5.x
   cp php.ini-dist /usr/local/lib/php.ini
   ```

5) Then tell the Apache server, where we want to serve files from, and the extension for identifying PHP files. .php is the standard extension but we can use other extension also like .html, .phtml etc.

6) Then go to HTTP configuration files (/usr/local/apache/conf or whatever the path is) then open httpd.conf with a text editor. Then search for the word DocumentRoot and change both paths to the directory to serve files (in our case, /home/httpd). A home directory is recommend rather than the default /usr/local/apache/htdocs because it is more secure, butit doesn’t have to be in a home directory. We will keep all PHP files in this directory. Then add at least one PHP extension directive and in the second line, we have all HTML files parsed as PHP.

   ```
   AddType application/x-httpd-php .php
   AddType application/x-httpd-php .html
   ```

7) Then, restart server. After changing the HTTP configuration or php.ini files, we must stop and start server again.
   cd../bin
   . /apachectl start
8) Open a text editor. Type :<? Php phpinfo () ;?>. save this file in your Web server’s document root as info.php
9) Browse the file using any of the web browsers.

9.2.3 PHP installation on Mac OS X with Apache

MAC OS probably came with PHP and Apache preinstalled. But if you want to Install PHP, we can do this by using the following steps.

1) Open Apache config file in text editor as a root
   Sudo open –a TextEdit /etc/httpd/httpd.conf
2) Edit the file by uncommenting the following lines.
   Load module php5_module
   AddModule mod_php5.c
   Addtype application/x-httpd-php.php
3) We should also uncomment <Directory/home/*/sites>
4) Restart the server
5) Open a text editor. Type :<? Php phpinfo () ;?>. save this file in your Web server’s document root as info.php
6) Browse the file using any of the web browsers.

9.2.4 PHP installation on Windows with IIS

The installation of PHP on windows server is much simpler than on UNIX. The following are the installation steps.

1) Using unzip utility, extract the binary archive in to a location. One of the common location is C:\PHP
2) Next we should copy .dll files from PHP directory to our system directory. We need to copy php5ts.dll file and the file corresponding to webserver module c:\PHP\sapi\php5isapi.dll and few other files also if we need them.
3) Next, we must copy the php.ini-dist or php.ini-recommended file to windows directory and we should rename it as php.ini. Then open the file using text editor and edit to get configuration directives, and set error reporting to E_ALL. Under the path and directories section, the doc_root is the most important directive.

4) Then, stop and restart the WWW service. We can do this by first select the Start menu→settings→ control panel→ services and then scroll down the list and select IIS admin service and click stop. After that select World Wide Web publishing Service and click start.
5) Open a text editor. Type:<? Php phpinfo();?>. save this file in your Web server's document root as info.php
6) Browse the file using any of the web browser

9.2.5 PHP installation on Windows with Apache

The following are the installation steps to install Apache with PHP5 on windows.

1) Download Apache server which is the current stable version with no_src.msi extension. Double click on the installer file to install in a location (C:\program Files is the common location). Run Apache fromthe command line or DOS prompt.
2) Using the unzip utility, extract PHP binary archive in a location (C:\PHP is the common location)
3) Copy some .dll files from PHP directory to system directory. We need to copy php5ts.dll file and the file corresponding to webserver module c:\PHP\sapi\php5isapi.dll and few other files also if we need them.
4) Next, we must copy the php.ini-dist or php.ini-recommended file to windows directory and we should rename it as php.ini. Then open the file using text editor and edit to get configuration directives, and set error reporting to E_ALL.
5) Then go to HTTP configuration file and open using text editor httpd.conf file. Then search the word DocumentRoot and change both paths by giving the directory path from where files are serving.
6) Then, stop and restart the WWW service. We can do this by first select the Start menu→settings→control panel→ services and then scroll down the list and select IIS admin service and click stop. After that select World Wide Web publishing Service and click start.
7) Open a text editor. Type :<? Php phpinfo () ;?>. save this file in your Web server's document root as info.php
8) Browse the file using any of the web browsers.

9.2.6 First program: Hello world

Writing PHP program

Writing PHP on our computer is actually very simple. We don't need any special software, except for a text editor (like Notepad in Windows). Run this and we are ready to write our first PHP script.

Basic PHP Syntax

A PHP scripting block always starts with **<?php and ends with ?>.** A PHP scripting block can be placed anywhere in the document. On servers with shorthand support enabled can start a scripting block with <? and end with ?>. For maximum compatibility, we recommend that we use the standard form (<?php) rather than the shorthand form.

Syntax: <?PHP?>

Declaring PHP

PHP scripts are always enclosed in between two PHP tags. This tells our server to parse the information between them as PHP. A PHP file normally contains HTML tags, just like an HTML file, 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>
```

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 ".

How to Save Your PHP Pages

If we have PHP inserted into our HTML and want the web browser to interpret it correctly, then we must save the file with a **.php** extension, instead of the standard .html extension. So be sure to check that we are saving our files correctly. For example instead of data.html, it should be data.php if there is PHP code in the file. The PHP file must have a .php extension. If the PHP file has .html extension, the PHP code will not be executed.

Finishing and Testing our Script

Now we have finished your script save it as php data.php and upload it to our server in the normal way. Now, using your browser, go the URL of the script. If it has worked (and if PHP is installed on your server) we should get a huge page full of the information about PHP on our server.

If our script doesn't work and a blank page displays, we have either mistyped our code or our server does not support this. If, instead of a page being displayed, we are prompted to download the file, PHP is not installed on our server and we should either search for a new web host or ask our current host to install PHP.

9.2.7 Data types

Several types of data or values can be stored using PHP data types. PHP offers eight primitive data types that are further divided into three categories:

- Scalar Types (predefined)
- Types of Compounds (user-defined)
- Special Types

➢ Scalar Types

It simply stores one value. In PHP, there are 4 scalar data types.

- **boolean:** The simplest data type, booleans function as switches. There are only two possibilities: TRUE (1) or FALSE (0). It frequently appears alongside conditional phrases. It returns TRUE if the condition is true and FALSE otherwise.
- **integer:** An integer is a piece of numerical data that has a positive or negative sign. It solely stores whole numbers, which are those without fractional or decimal places.
- **float:** A number with a decimal point is referred to as a floating-point number. It may carry numbers with a fractional or decimal point, as well as a negative or positive sign, unlike integer.
- **string:** A string is a type of unsigned data. It can store any alphabet, numbers, and even special characters in addition to letters. You must surround string values either in single quotations or double quotes.

- **Compound Types**

In PHP, there are 2 compound data types. It can store a variety of values.

- array: A compound data type is an array. In a single variable, it is possible to hold numerous values of the same type.
- object: Instances of user-defined classes that may hold both values and functions are referred to as objects.

- **Special Types**

In PHP, there are 2 unique data types.

- **resource:** In PHP, resources are not a precise data type. They are essentially used to store references to external PHP resources or function calls. Consider making a database call. It is a third-party resource.
- **NULL:** Null is a unique data type with only one possible value, which is NULL. Since it is case sensitive, it is customary to write it in capital letters.A variable with no value was specified by the special data type NULL.

9.2.8 Variables

A variable is a means of storing a value, such as text string "ManipalUniversal Learning". A variable can then be reused throughout our code, instead of having to type out the actual value over and over again. In PHP you define a variable with the following form:

- **$variable_name = Value;**

If we forget that dollar sign at the beginning, it will not work. This is a common mistake for new PHP programmers!

Following is a program that contains a string, and a variable that contains a number:

```
<?php
        $txt="Manipal Universal Learning!"
        $x=16;
?>
```

There are a few rules that we need to follow when choosing a name for our PHP variables.

- PHP variables must start with a letter or underscore "_".
- PHP variables may only be comprised of alpha-numeric characters and

underscores. a-z, A-Z, 0-9, or _ .

- Variables with more than one word should be separated with underscores. $my_variable.
- Variables with more than one word can also be distinguished with capitalization. $myVariable.

PHP comments

Comments are useful for us to ignore the programming part that currently we don’t want to include in our program, but later on we can take it. While there is only one type of comment in HTML, PHP has two types.

- ***single line comment*** The single line comment tells the interpreter to ignoreeverything that occurs on that line to the right of the comment. To do asingle line comment type "//" or "#" and all text to the right will be ignored by PHP interpreter.

```
<?php
echo "Hello Manipal!"; // example of single line comment.
?>
```

Out of the above program will be Hello Manipal!

- ***multiple line comments*** Similar to the HTML comment, the multi-line PHP comment can be used to comment out large blocks of code or writing multiple line comments. The multiple line PHP comment begins with " /* " and ends with " */ ".

```
<?php
/* This Echo statement will print out my message to place in which I
resideon.  In other words, the World. */
echo "Hello World!";
/* echo "My name is Humperdinkle!";
echo "No way! My name is Uber PHP Programmer!";
*/
?>
```

Output of this program will be Hello World!.

9.2.9 Operators

In all programming languages, operators are used to manipulate or perform operations on variables and values. There are many operators used in PHP, so we have separated them into the following categories to make it easier to learn them all.

- Assignment Operators
- Arithmetic Operators
- Comparison Operators
- String Operators
- The Concatenation Operator

Assignment operators are used to set a variable equal to a value or set a variable to another variable's value. Such an assignment of value is done with the "=", or equal character. **Example:**

- $my_data = 2;
- $another_data = $my_data

Now both $my_data and $another_data contain the value 2. Assignments can also be used in conjunction with arithmetic operators.

Arithmetic operators are used to perform mathematical operations like addition, subtraction, multiplication, division and Modulus. Table 9.1 shows all the arithmetic operators.

Table 9.1: Arithmetic Operator

Operator	English	Example
+	Addition	4 + 4
-	Subtraction	6 - 4
*	Multiplication	5 * 4
/	Division	16 / 4
%	Modulus	43 % 10

Comparison operators: Comparisons are used to check the relationship between variables and/or values. Comparison operators are used inside conditional statements and evaluate to either true or false. Here are the most important comparison operators of PHP. Assume: $x = 4 and $y = 5; Table 9.2 shows all the comparison operators.

Table 9.2: Comparison Operator

Operator	Meaning	Example
==	Equal To	$x == $y
!=	Not Equal To	$x != $y
<	Less Than	$x < $y
>	Greater Than	$x > $y
<=	Less Than or Equal To	$x <= $y
>=	Greater Than or Equal To	$x >= $y

String Operators: As we have already seen in the Echo statement, the period "." is used to add two strings together, or more technically, the period is the concatenation operator for strings. By the following program you can see the function of string operator.

PHP Code:

```
$a_string = "Distance";
$another_string = " Education!";
$new_string = $a_string . $another_string;
echo $new_string . "!";
```

Output

```
Distance Education!
```

The Concatenation Operator

There is only one string operator in PHP. The concatenation operator (.) is used to put two string values together. To concatenate two string variables together, use the concatenation operator:

```
<? php $txt1="Hello World!";

$txt2="Distance education!";

echo $txt1 . " " . $txt2 ;?>
```

The output of the code above will be:

```
Hello World! Distance education!
```

9.2.10 Flow Control Structures

Mainly used decision making control structures are conditional structures and looping structures. Conditional structures are if, elseif…else and switch statements.

a) The if…else statement

The general syntax of if…else statement is given below.

```
If (condition)
 True block statements;
Else
 False block statements;
```

If the given condition is true, then the true block code will be executed and if condition is false then false block statement will execute.

b) Elseif statement

If we want to execute some code if one of the several conditions are true then we can use else if statement. The syntax is given below:

```
If (condition)
True block statements1;
Elseif (condition)
True block statements2;
else
False block statements;
```

If the given condition is true, then the true block statements1 will be executed. Or else it will check another condition and if it is true then the true block statements2 will be executed and if all the conditions are false then false block statement will execute.

c) Switch statement

Switch statement is used to select one of many statements of the code to be executed. The general syntax of this statement is given below.

```
Switch (expression)
    {
    Case label1: statements;
                            Break;
    Case label2: statements;
                           Break;
    Default: default statements
    }
```

In switch statement, first it evaluates the expression. Then it checks the resulting value with each case labels. If matching label found then its corresponding statements will execute. Otherwise default statements will be executed.

PHP Loop Types:

Loops are used to execute same block of code repeatedly. There are mainly four loops. They are,

- For
- While
- Do…while
- Foreach

a) The for loop statement

For loop is an entry controlled loop. First it will check the condition and after that it will execute the block statements. The general syntax of for loop is given below,

```
For (initialization; condition; incrementation/decrementation)
{
Code to be executed;
}
```

In this the initialization is used to set the start value for the counter of the number of loop iterations. Then it will check the condition. If the condition is true then it will execute the code once. After that incrementation or decrementation of the count value and again it will check the condition and if again the condition is true then code will be executed.

b) The While loop statement

While loop statement is also an entry controlled looping statement. In this also first, a condition will check and if it is true then the code will execute. Code will execute as many times as the condition is true. The general form is

```
While (condition)
{
 code to be executed;
Increment/ decrement;
}
```

In this, Initialization will do before entering in to the loop and incrementation or decrementation will do inside the loop.

c) The do…while loop statement

This is an exit controlled loop. That means it will execute the code once and then will check the condition. The general syntax is given below,

```
Do
```

```
{
Code to be executed;
} while (condition);
```

The code to be executed will execute at least once and then it will check the condition. Then it will repeat the loop as long as the condition is true.

d) The Foreach loop statement

The Foreach statement is used to loop through arrays. The general syntax is given below,

```
Foreach (array as value)
{
Code to be executed;
}
```

For each pass the value of the current array element is assigned to value and the array pointer is moved by one and in the next pass next element will be processed.

Self Assessment Questions

3. PHP scripts are always enclosed in between______PHP tags.
4. PHP variables must start with a letter or underscore "_". (true/false)

9.3 Functions

A function is a section of code that accepts another input in the form of a parameter, processes it, and then returns a value. A piece of code known as a PHP Function is reusable, receives parameter lists as input, and returns a value. Although there are built-in functions, PHP also allows you the opportunity to write your own functions.

Creating and Calling a PHP Function

```
<html>
<body>
<?php
//we create a function name my_function
function my_function()
{
        echo "Hello! Students?";
}
//we call our function like this when we want to use it
my_function(); ?>
</body>
</html>
```

In the above code, we create our own function name my_function() and we call that function by just typing the its name. The above code outputs "Hello! Students?";?".

9.3.1 PHP Functions – Adding parameters

To add more functionality to a function, we can add parameters. A parameter is just like a variable. Parameters are specified after the function name, inside the parentheses.

```
<html>
<body>
<?php
//we create a function named my_function
function my_function($first_name, $last_name, $message)
{
        echo "$first_name $last_name after that " . $message;
}
//we call our function like this when we want to use it
my_function("Shoney", "Kumar", '"It is the time of distance education."');
?>
</body>
</html>
```

In the above example we created a function with three parameters, $first_name, $last_name and $message. We used these parameters tosend our custom arguments (values) to our function. The above example *outputs*... Shoney Kumar after that “It is the time of distance education."

9.3.2 PHP Functions – Return values

To let a function return a value, use the return statement.

Example

```
<html>
 <body>
 <?php
        function add($a,$b)
        {
                $sum=$a+$b;
                return $sum;
        }
        echo "2 + 15 = " . add(2,15);
 ?>
</body>
 </html>
```

Output:

```
2 + 15 = 17
```

9.3.3 PHP Function Arguments

With PHP functions, we can pass data through parameters that are separated by commas.

Call by Value and Call by Reference are supported in PHP.

- **Call by value:** The real value is not changed if it is modified inside a function in PHP call by value. In the below example, the string "Call By Value mechanism" is concatenated with the variable $str before it is transmitted to the adder1 function. Yet, printing $str variable results 'Hello world' only. This is due to the fact that just the local variable $str2 is changed. It has no impact on the $str variable.

  Example:

  ```
  <?php  function adder1($str2)

  {
     $str2 .= 'Call By Value  mechanism';
  }
  $str = 'Hello world ';
  ```

```
adder1($str);
echo $str;
?>
```

- **Call By Reference**

When a function is called by reference in PHP, the real value is changed if the function makes a change to it. In this situation, the & (ampersand) symbol must be used with formal arguments. The variable's reference is represented by the &.

In the example, the string "Call By Reference mechanism" is concatenated with the variable $str before being provided to the adder1 function. This is a Call By Reference mechanism, as shown by the printing of the $str variable. This is so that modifications can be made to the real variable $str.

Example :

```
<?php
function adder1(&$str2)
{
   $str2 .= 'Call By Reference';
}
$str = 'This is ';
adder1($str);
echo $str;
?>
```

9.3.4 Using Get and Post Function

- **The GET Function**

The GET function is used to collect values from a form sent with method="get". Information sent from a form with the GET method is visible to everyone and has limits on the amount of information to send.

Example

```
<form action="welcome.php" method="get">
Name: <input type="text" name="Sname" />
```

```
Age: <input type="text" name="fee" />
 <input type="submit" />
 </form>
```

When the user clicks the "Submit" button, the URL sent to the server could look something like this:

http://www.manipal.com/welcome.php?Sname=Kumar&fee=20000

However, because the variables are displayed in the URL, it is possible to bookmark the page. This can be useful in some cases. The get method is not suitable for large variable values; the value cannot exceed 100 characters.

- ***The POST Function***

The POST function is used to collect values from a form sent with method="post". Information sent from a form with the POST method is invisible to others and has no limits on the amount of information to send.

Example

```
<form action="welcome.php" method="get">
Name: <input type="text" name="Sname" />
Age: <input type="text" name="fee" />
<input type="submit" />
</form>
```

When the user clicks the "Submit" button, the URL sent to the server could look something like this: **http://www.manipal.com/welcome.php.** So as we can see in this example that it will not show any information.

Self Assessment Questions

5. The_________function is used to collect values from a form sent with method="post".
6. The________function is used to collect values from a form sent with method="get".
7. To add more functionality to a function, we can add____________.

9.4 Summary

- PHP is one of the most popular server side scripting languages running today. It is used for creating dynamic webpages that interact with the user offering customized information.
- Three vital components need to be installed on our system to develop and run PHP web pages. Those are Webserver, database and PHP parser.

- A PHP scripting block always starts with <? Php and ends with?>.
- GET method in PHP will help us to send the encoded user information appended to the page request.
- In PHP, a function is a reusable section of code that executes a certain task. It receives user input in the form of parameters, does out specific operations, and outputs the results.

9.5 Terminal Questions

1. How PHP works? Explain with a suitable example?
2. Explain the different types of operators used in PHP with an example?
3. How we can add parameters and return values in PHP functions?
4. Briefly explain the Get and Post function?

9.6 Answers

Self Assessment Questions

1. Hypertext Pre-processor
2. Server- side
3. Two
4. True
5. POST
6. GET
7. Parameters

Terminal Questions

1. When a user navigates in their browser to a page that ends with a .php extension, the request is sent to a web server, which directs the request to the PHP interpreter. Figure 9.1 shows the working of PHP, how PHP works. (For more details refer section 9.1.2)
2. In all programming languages, operators are used to manipulate or perform operations on variables and values. There are many operators used in PHP, (For more details Refer section 9.2.9)
3. To add more functionality to a function, we can add parameters. A parameter is just like a variable. Parameters are specified after the function name, inside the parentheses. (For more details refer section 9.3.1)
4. . PHP offers two different ways for clients to communicate with servers. These include GET and POST. (For more details refer section 9.3.4)

9.7 References

- Larry Edward Ullman (2003). *PHP for the World Wide Web*. Third edition. Peachpit press
- Leon Atkinson, Zeev suraski. *Core PHP Programming.* Third edition
- Jesus castragentto (2004). *Professional Php programming*