WEB APPLICATION DEVELOPMENT

PGDCA 202

BLOCK 1:
INTRODUCTION TO PHP

Dr. Babasaheb Ambedkar Open University
Ahmedabad

WEB APPLICATION DEVELOPMENT

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Mr. K. J. Sharma

Language Editor
Mr. Vipul Shelke

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Mr. Prashant Tikone

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included may tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

WEB APPLICATION DEVELOPMENT

Contents

BLOCK 3: OO AND FILE HANDLING IN PHP

UNIT 1 OBJECT ORIENTED PHP

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

UNIT 2 FILE AND DIRECTORY HANDLING

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP

UNIT 1 PHP AND MYSQL DATABASE

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

UNIT 2 STATE MANAGEMENT IN PHP

Session, Cookies

Dr. Babasaheb
Ambedkar
Open University

PGDCA 202

Web Application Development

BLOCK 1: INTRODUCTION TO PHP

BLOCK 1: INTRODUCTION TO PHP

Block Introduction

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. PHP is Hypertext Pre-processor a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases.

In this block, we will detail about the basic understanding of the architecture of web application with knowledge on static and dynamic web application. The block will focus on features of installation of WAMP with study about their characteristics. The concept of Identifiers, Variables, Constants and Expressions are also explained.

In this block, you will make to learn and understand about basic of outputting data to web browser and its techniques. The concept related to web application as interactive system allowing users to interact will also be explained to you. You will be demonstrated practically about String Interpolation.

Block Objective

After learning this block, you will be able to understand:

- The architecture of a web application
- Knowledge about static and dynamic web application
- Features of installation of WAMP
- Idea about Outputting Data to the Browser
- Knowledge related to PHP Supported Data types
- Basic of Identifiers, Variables, Constants and Expressions
- Concept of Control Structures

Block Structure

Unit 1: Introduction to Web Application

Unit 2: PHP Basics

UNIT 1: INTRODUCTION TO WEB APPLICATION

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About architecture of web application
- About Static and dynamic web application
- About WAMP

1.1 Introduction

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. In this, the components can be internal or external to an organization, which are used to communicate using Internet-based protocols

which can be Hyper Text Transfer Protocol or else. These services run on servers and carried out large variety of data which users want to access quickly and easily.

As seen, the famous programming model where individual web services are joined or mixed in order to frame functional results in service-oriented architecture. It is an architectural model, where service consumer put on the requests to service provider using standard connection. In this, request and successive response are explained in a way which is understandable to consumer and provider.

Many web services make use of Extensible Markup Language (XML) in order to define format of request and response messages. This feature is tagged structure which shows required flexibility for changing of information that exists among disparate components.

Simple Object Access Protocol shows standard for persistence and sending of web service messages which is part of XML messaging specification that shows message format with certain rules applied in exchanging of data in required sequence that appears among structured data types and arrays.

1.2 The architecture of a web application

A web application is an interactive system which allows its users to do business logic which is inside a server and helps in seeing the results of particular logic using web browser on client workstation. In this the defining factor makes the system a web application where server and client are able to communicate over the Internet. Such web applications makes data to process using web services which is present for users for fast access using web browsers.

Web applications are built on client/server architecture where business logic is present in the application that works on web server and uses HTTP in order to communicate with clients over Internet. Here the web server helps the application by passing requests from clients to application which will further return with responses to the client.

On client side, web application can be visible using browser. The application's user interface with the use of HyperText Markup Language pages gets interpreted and displayed with the help of browser. Apart from this, such HTML pages carry web forms, image files, audio and video clips and several other displayable data types.

Fig 1.1 describes the basic architecture of web components and services which are commonly applied to host websites and build web applications.

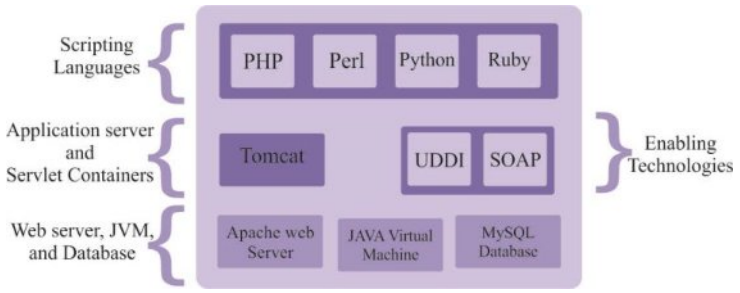


Fig 1.1 Architecture of Web application

The structure will be shown in fig 1.2

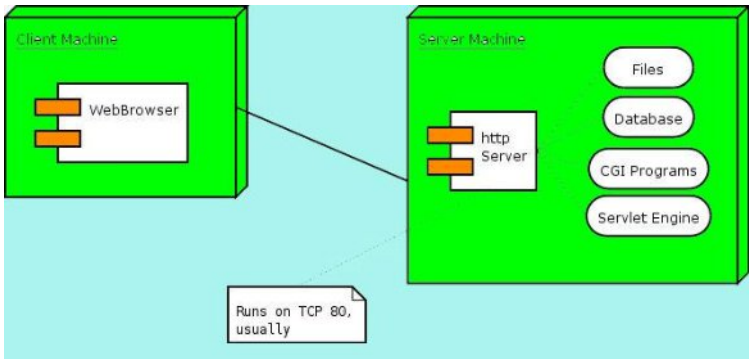


Fig 1.2 Structure of Web application

Fetching Static Content

Client sends HTTP request like

GET /hostpage.html HTTP/1.1CRLF

Host: www.mypage.comCRLF

CRLF

Server response as:

HTTP/1.1 200 OKCRLF

Server: XXXCRLF

Date: XXXCRLF

Content-type: XXXCRLF

...CRLF

CRLF

<html>

...

</html>

Here the browser will read the response from server and provide the page. If the page has fixed content which can be images, then the client will request for it. In HTTP/1.0, requests will be on different connections where as HTTP/1.1 and later carries same connection.

Server-Side Dynamic Content

In server side, applications are not so important if they just have static pages which can be responded by the user input. The server will construct the content of HTTP responses on fly in response to user input.

It is noted that fig 1.3 describes the Architecture Diagram of PHP web application that shows how every components in a system gets joined or connected among each other and how data flows across them.

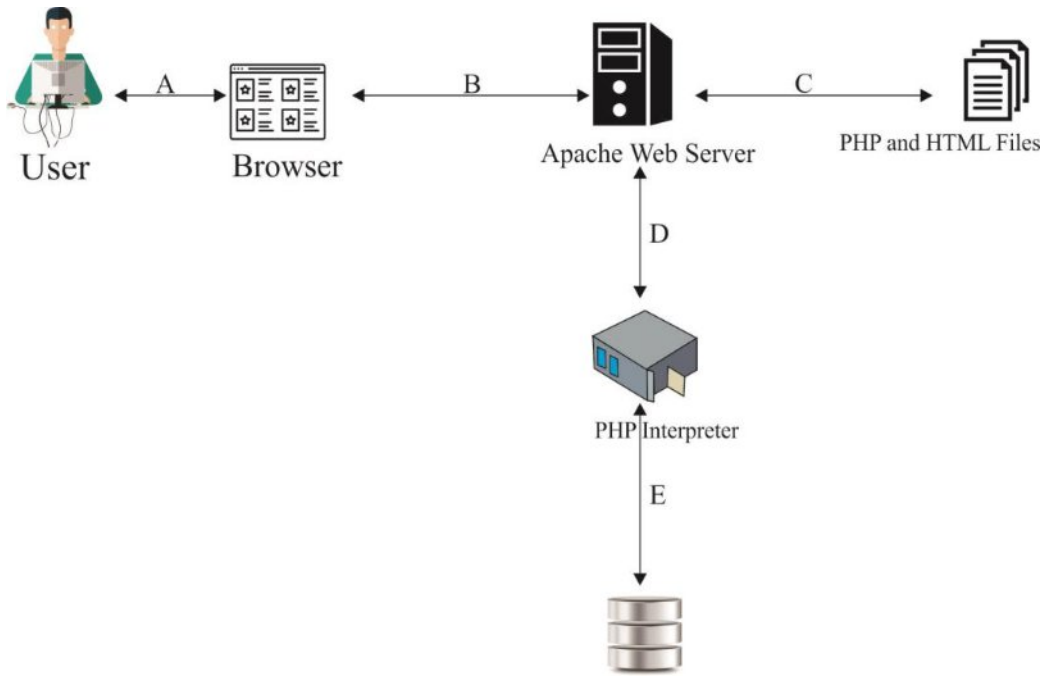


Fig 1.3 PHP web application

From the figure we see that initially the user will access the website using the browser where user types the URL of website in browser and click on go option.

Then the page request on browser will reach to Web Server.

After this, the web server will collect requested page from its document root.

When a static element such as HTML, CSS, image or Script file appears then web server will send it directly to browser and in such case the browser will leave it to user on screen.

When the PHP file appears then the web browser will send content of file to PHP Interpreter which further will interprets such code and start execute.

After executing the code, the PHP interpreter will generate an output and sends it to web server. Further, the web server will send the required content to browser which provides it to users' screen.

We see all static components which can be HTML, CSS files, Image Files, Scripts etc will not require interpreter. In such case the web browser will provide them and shows on screen.

Only if requested page is in PHP, then the web server will send it to PHP interpreter in order to do translation and execution. For such reason, those listed static components will remain inside the server and are consider as part of User Interface which provides at user's browser which is a Client side.

Due to this, the PHP files are referred to as Server side components since they carry out dependencies on other with this, we see that:

1. PHP files are kept on Server - Server Side
2. PHP Interpreter will interprets PHP language and perform calculation of instructions as per required code without any compilation.

Check your progress 1

1. PHP scripts are executed on_____.

   a. Client Side

   b. Server side

   c. Both of these

   d. None of these

2. The extension of php file is_____.

   a. html

   b. .ph

   c. .xml

   d. .php

1.3 Static and dynamic web application

Web server process out every piece of information to users so as to compose the web pages. It is seen that web page structure can of many pieces, which result in sum that affects series of performance metrics which can be in terms of bandwidth consumption; user perceived latency without relevance to workload required so as to arrange all information of particular web pages together. It describes whether a web page needs large data processing or queries to permanent store so as to put together.

A web application's content can be of two types:

- Dynamic
- Static

Dynamic content is what needs some type of processing that can be applied where as static content is that that never changes. When a request is made for either type of content, in such case the web server will do execution which is required to send out dynamic content present in the file. This process is shown in figure 1.4.

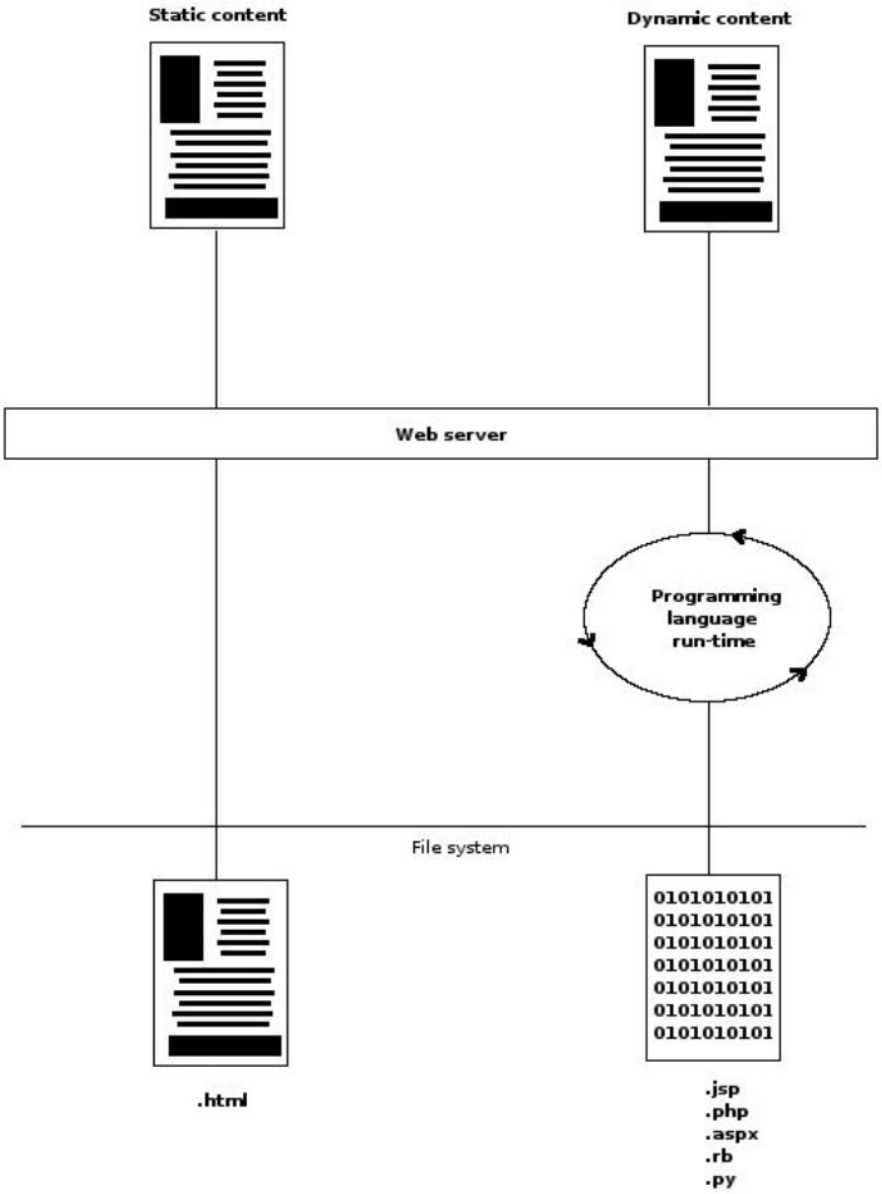


Fig 1.4 Process execution

It is noticed that the art of dispatching dynamic or static content will take place when a particular load on web server using dynamic content places heavier load as compared to static content. It describes inefficient allocation of resources on web server that is required to send dynamic and static content. So it is analysed that decoupling of static content from application is considered as steps in order to upgrade the performance on web servers. The problem with static content separation is not clear of having full static web page.

When a user's browser receives the original application page of the elements they are declared in the content where browser start requesting for referenced content. Figure 1.5 shows such multiple request process.

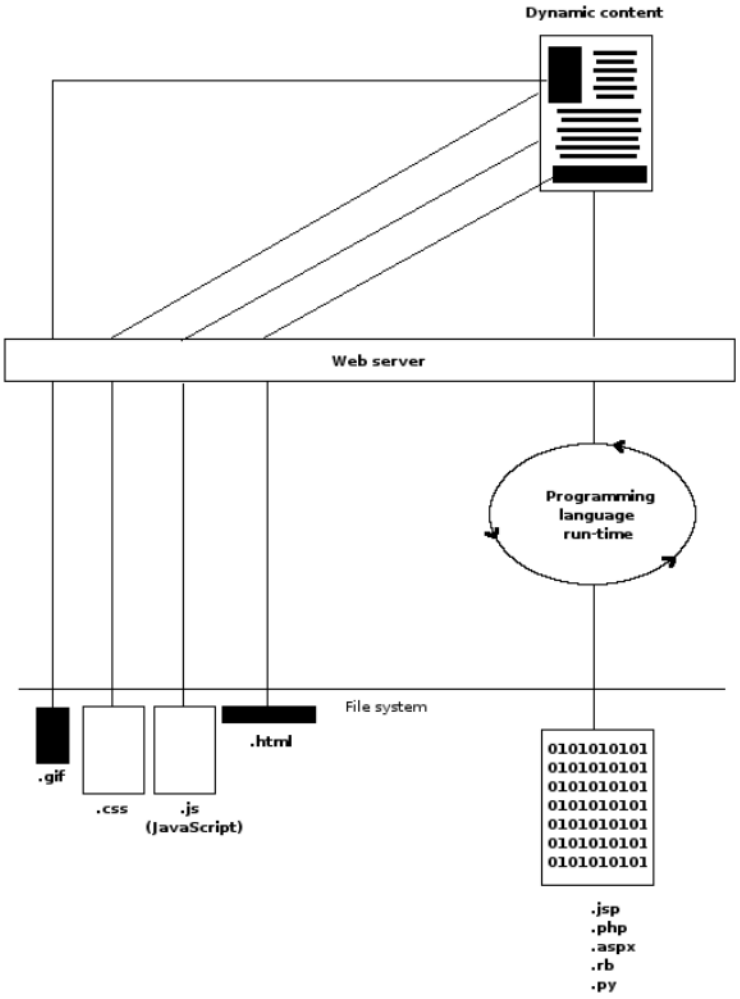


Fig 1.5 Referenced content requests

As observe in figure 1.5, the extra requests generates incremental load on web server. As this process is inevitable, it can be designed to be more performance friendly. Rather than having all content send across by same web server, referenced content can also be send by separate web server which alleviates potential web server load.

As seen, in the last listing, domain URLs applied in HTML elements for static content references which allow content to be decoupled and attended by a different web server on a different sub-domain. Thus lowering the burden on the primary web server used for original dynamic content. This process is illustrated in figure 1.6.

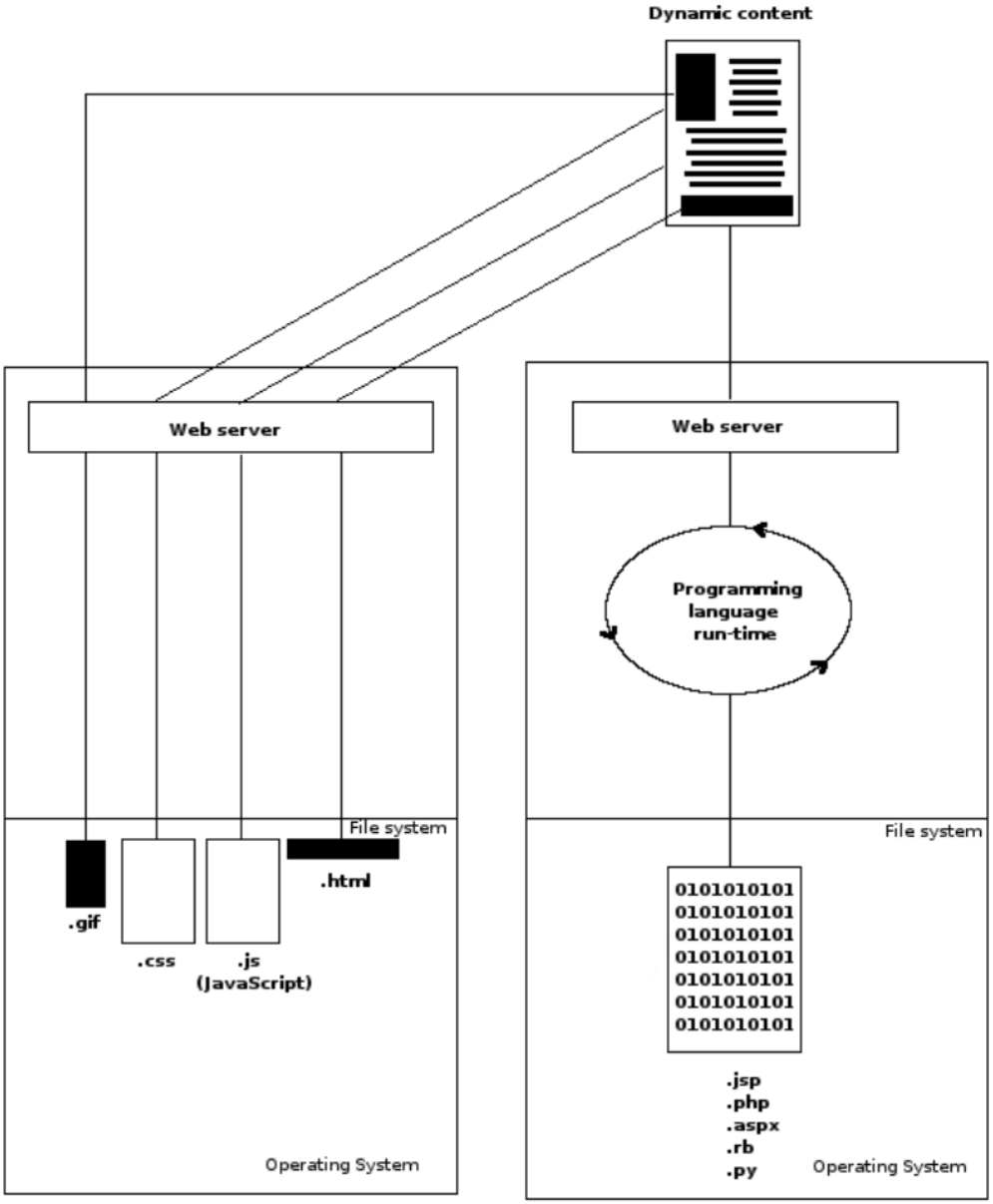


Fig 1.6 Different web servers for dynamic and static content.

Check your progress 2

1. What is meant by dynamic web content?

   a. Content that cannot change without a developer editing its source code,

   b. Different content from the same source code.

   c. Both of these

   d. None of these

1.4 Installing WAMP

WAMP allows you to move between different versions of PHP if it is installed properly. There are certain steps to be followed while installing Wamp Server.

To start installing, you have to open the folder where you have saved your file and double click on installer file. After double clicking, a security warning window will open which will ask you whether to run this file or not. If you select on Run you will find that the installation process started and you will see Welcome To WampServer Setup Wizard screen on which you have choose next to forward with the installation.

Fig 1.7 Welcome screen

After this, the License Agreement screen will appear which will ask you to accept all agreements by selecting the radio button. Selecting the radio button and clicking on next will allow moving to next screen.

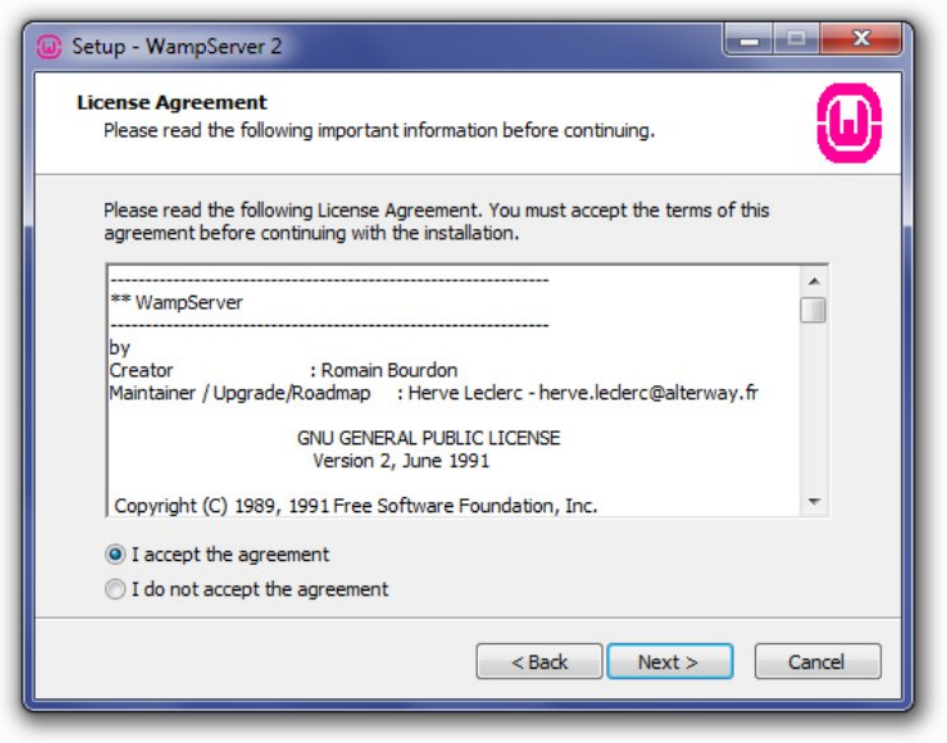


Fig 1.8 License Agreements

This screen will ask for the location where you have to select Destination Location. Unless you would like to install WampServer on another drive, you should not need to change anything. Click Next to continue.

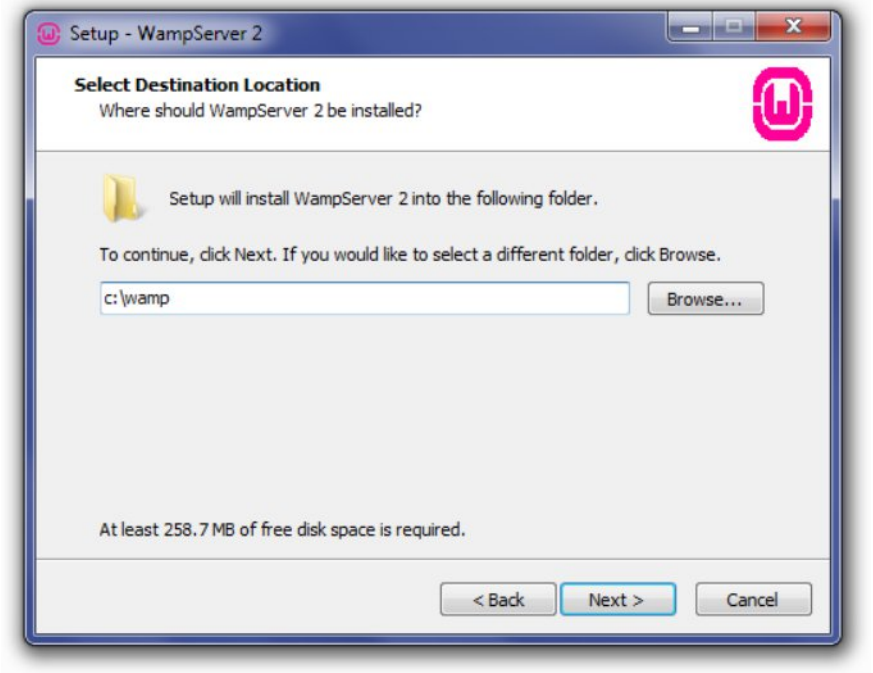


Fig 1.9 Destination location

After this, the next screen shows Select Additional Tasks screen where you need to select whether you would like Quick Launch icon to be added to taskbar or Desktop icon after the installation. After making the selections, click Next to continue.

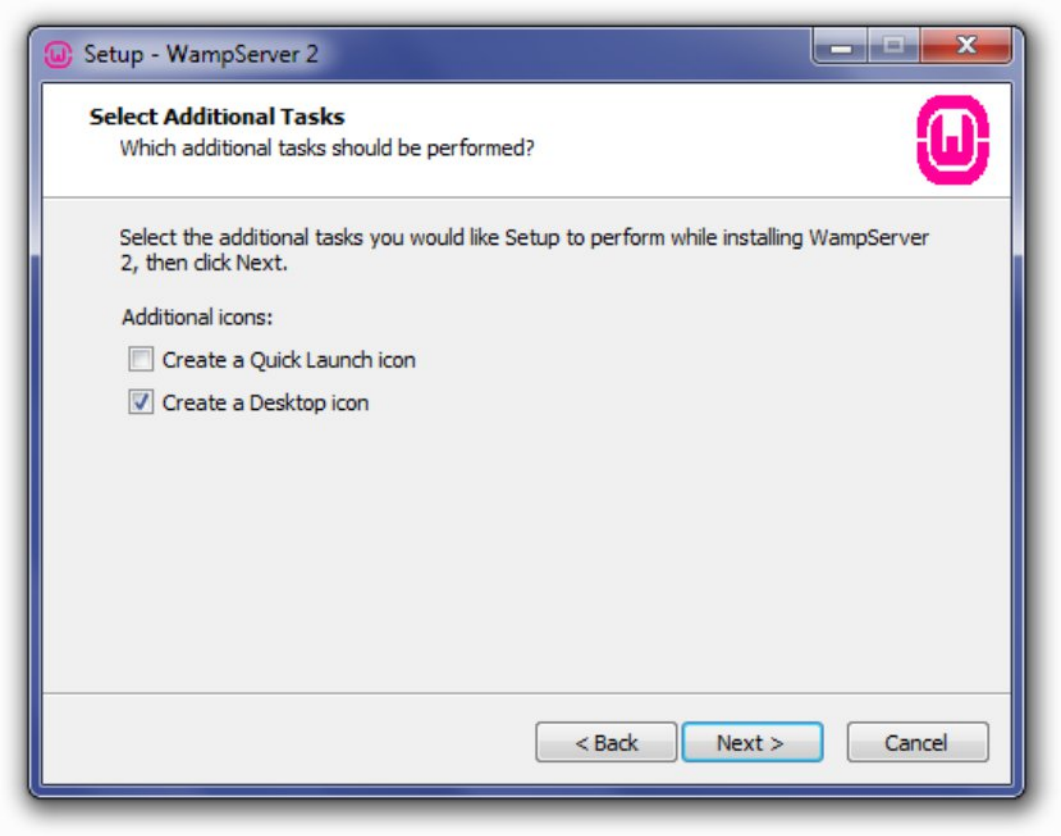


Fig 1.10 Task selection

Clicking on Next will take you to Ready To Install screen where you can review your setup choices and change any by clicking Back to required screen. Once you have reviewed your choices, click Install to continue.

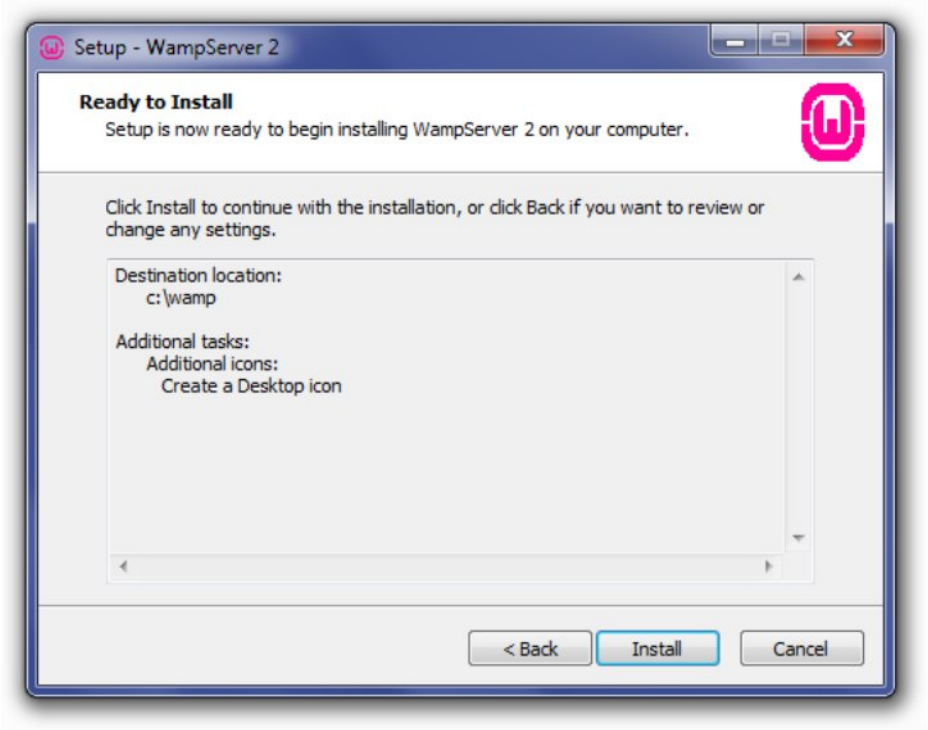


Fig 1.11 Ready Install screen

After this, the WampServer will start extracting files to required selected location.

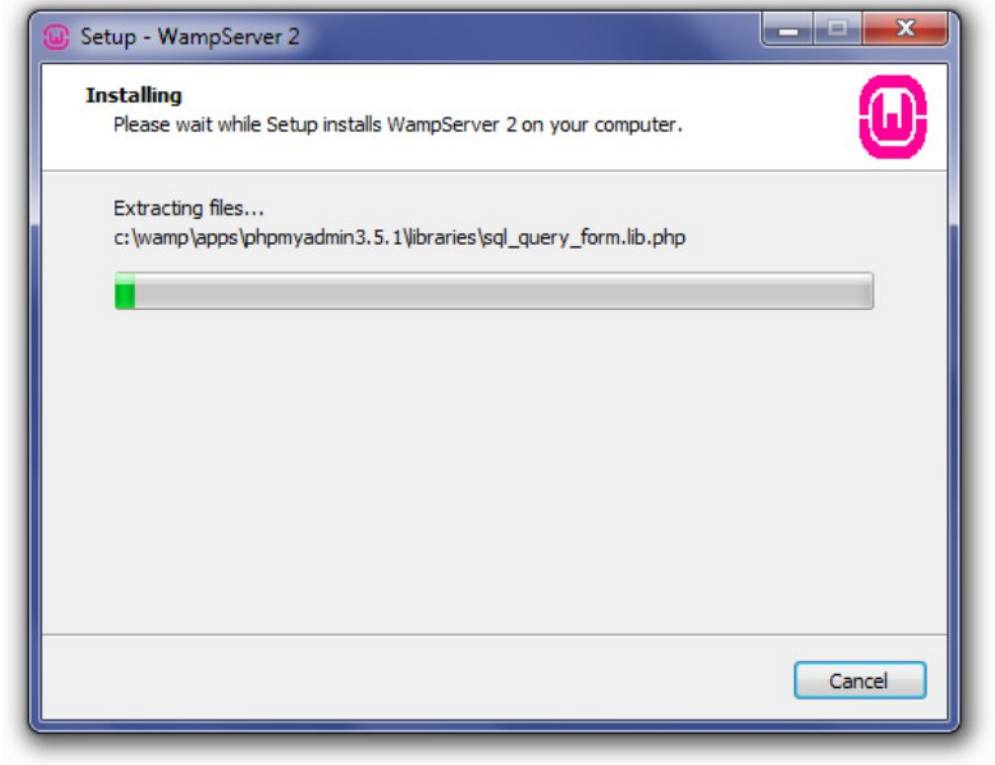


Fig 1.12 Installation Screen

After extraction of files, you will be asked to select default browser. Wamp Server defaults to Internet Explorer upon opening the local file browser window. If your default browser is not an Internet Explorer, then find required .exe file:

- Opera: C:\Program Files (x86)\Opera\opera.exe
- Firefox: C:\Program Files (x86)\Mozilla Firefox\firefox.exe
- Safari: C:\Program Files (x86)\Safari\safari.exe
- Chrome: C:\Users\xxxxx\AppData\Local\Google\Chrome\Application\chrome.exe

Now select default browser's .exe file and click Open to continue.

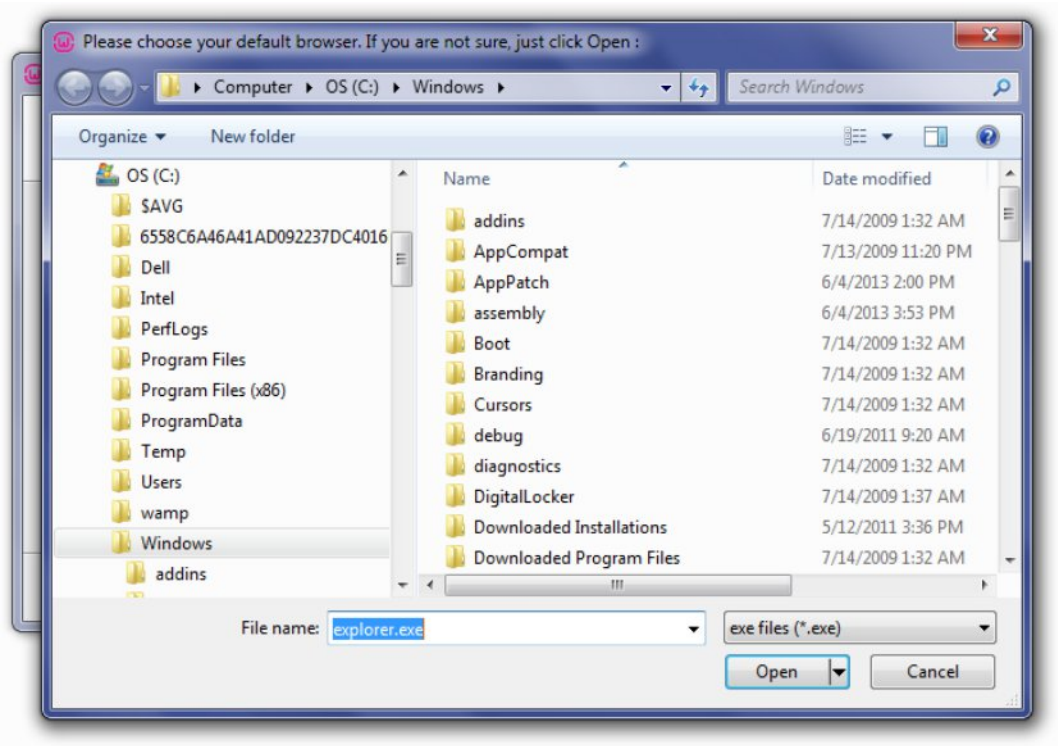


Fig 1.13 Selecting default browser

On clicking you will find that a Windows Security Alert window will appear prompting that Windows Firewall has blocked some features of program. Here you need to check whether you want to allow Apache HTTP Server to communicate on private or public network and if this and then click Allow Access. You will see that the setup screen will appear showing status of installation process.

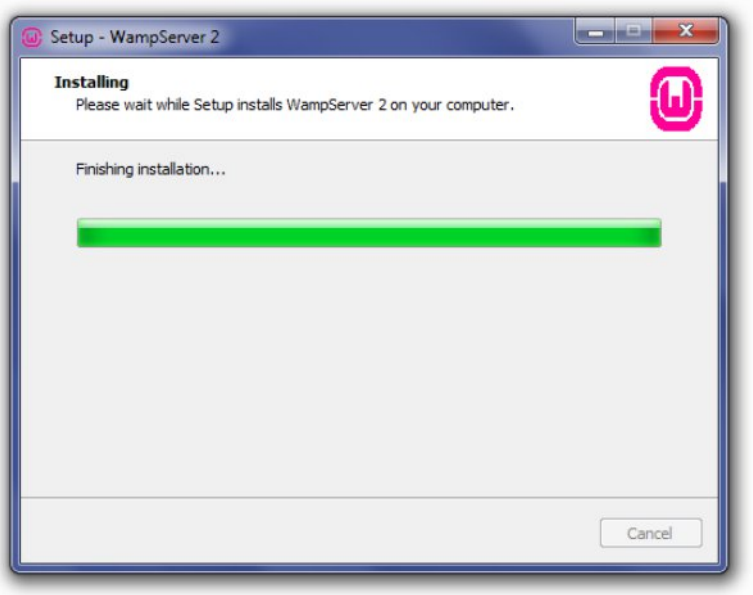


Fig 1.14 Installing process

You will that the progress bar will work and once it is completely green, then PHP Mail Parameters screen will appear. Here you have to leave SMTP server local host and change email address to one of your selection. Click Next to continue.

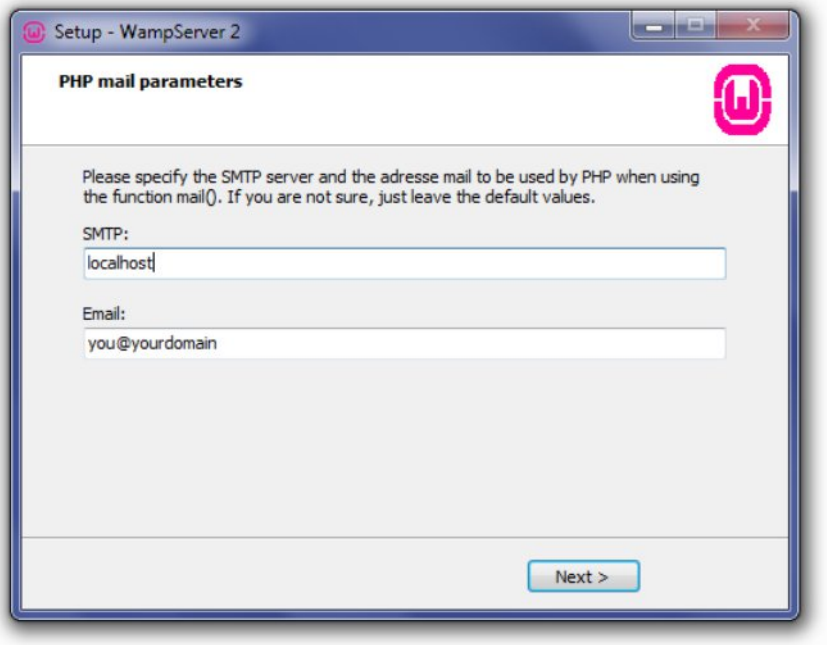


Fig 1.15 Local host parameters

After this you will find that the Installation complete screen will appear where you have to select Launch WampServer Now and click on Finish to complete installation.

Fig 1.16 Complete setup wizard

Finally you will find WampServer icon in systray on right side of taskbar.

Check your progress 3

1. Full form of WAMP is_____.

   a. Windows, Apache, MySQL and PHP

   b. Word, Adobe, MySQL and PHP

   c. Windows, Apache Mainstream Program

   d. None of these

2. Which of the following is needed to be installed on computer as to run PHPscript?

   i) Adobe Dreamweaver
   ii) PHP
   iii) Apache
   iv) IIS

   a. All of these

   b. Only ii)

   c. ii) and iii)

   d. ii), iii) and iv)

1.5 Let Us Sum Up

In this unit we have learnt that Web application are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. Simple Object Access Protocol shows standard for persistence and sending of web service messages which is part of XML messaging specification that shows message format with certain rules applied in exchanging of data in required sequence that appears among structured data types and arrays.

A web application is interactive system allowing its users to do business logic which is inside a server and helps in seeing the results of particular logic using web browser on client workstation. In server side, applications are not so important if they just have static pages which can be responded by the user input.

Web server process out every piece of information to users so as to compose the web pages. In WAMP, you can move between different versions of PHP if it is installed properly with various steps.

1.6 Answers for Check Your Progress

Check your progress 1

Answers: (1 –b), (2-d)

Check your progress 2

Answers: (1 –b)

Check your progress 3

Answers: (1 –a), (2-d)

1.7 Glossary

1. **Web application -** Services used for business logic components connected across with exchange data to do work.
2. **Simple Object Access Protocol -** Standard for persistence and sending of web service messages.

3. **Server side applications -** Applications performed at server end on static responded by user input.
4. **Web server -** Server running web applications data process information to users through web pages.
5. **WAMP -** Version in PHP where you can move in-between the process.

1.8 Assignment

Design a process to install PHP web application.

1.9 Activities

Create an activity on WampServer.

1.10 Case Study

Discuss about the architecture of a web application.

1.11 Further Readings

1. Web Applications in PHP, Jim Smith, Oxford, 2006
2. Applications in WAMP by Coulouris and Kindberg, 2010

UNIT 2: PHP BASICS

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About PHP Code
- About Outputting Data to the Browser
- About Variables and Constants

2.1 Introduction

PHP which is Hypertext Preprocessor is a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases. It is normally used for creating and developing web based software applications.

2.2 Embedding PHP Code in Your Web Pages

PHP is an important coding technique which can be easily embedded directly in HTML. For doing this, the page must be passed to PHP engine for interpretation. Since web server doesn't pass every web page but simply passes those pages which are located by particular file extension. By selectively passing certain pages to engine is inefficient for engine in order to consider every line as potential PHP command. So engine needs certain means to find at the same time which areas of page are PHP enabled. There are four delimitation of variants such as:

Default Syntax

It is a default delimiter syntax which opens with <?php and results with ?> as:

```
<h3>Hi!</h3>
<?php
echo "<p>K. J. Sharma</p>";
?>
<p>Engineer</p>
```

Output:

Hi!

K. J. Sharma

Engineer

Short-Tags

It is a shorter delimiter syntax where syntax gives up php location that is needed in default syntax. For this you have to allow PHP's short_open_tag directive as shown:

```
<?
print "He is an Author";
?>
```

Since short-tag delimiters are simple and convenient so it is not applicable while creating PHP driven software which is needed for redistribution as this feature potentially gets disabled in php.ini file.

Such tag when enabled will allow you to quickly escape from PHP to output with small dynamic text thereby deleting statements using output variation:

```
<?=" He is an Author.";?>
```

This is functionally equivalent to both variations:

```
<? echo " He is an Author."; ?>
<?php echo " He is an Author.";?>
```

Script

It is noted that many programmers has problems with PHP escape syntax variants, so support for another mainstream delimiter variant, <script>, is offered:

```
<script language="php">
print " He is an Author.";
</script>
```

ASP Style

If you are having experience on ASP, the delimiting strategy/escape syntax PHP supports it as shown:

```
<%
print " He is an Author.";
%>
```

It is noted that ASP Style and Script delimiting variants are not commonly used and should be avoided till you have reasons for doing.

Embedding Multiple Code Blocks

You can escape to and from PHP as many times as needed within a page. Take an example:

```
<html>
```

```
<head>
<title><?php echo "Academic Solution";?></title>
</head>
<body>
<?php
$date = "21th September 2015";
?>
<p>Today's date is <?=28th Feburary 2016?></p>
</body> </html>
```

It is noted that any variables can be declared a prior code block that are remembered for later blocks.

Check your progress 1

1. Choose the correct way of writing short-open tag.
   a. <? echo”hello” ?>
   b. <% echo”hello” ?>
   c. <? echo”hello” ?>
   d. None of these
2. Choose the correct way of writing ASP tag.
   a. <? echo”world” ?>
   b. <% echo”world” %>
   c. <? echo”world” >
   d. None of these

2.3 Commenting Your Code

Comment in PHP code is a line which is not read as part of program but to read by person editing the code. If you are working with team using your script, then comments allows the programmers to understand what you were doing during the process each time. It makes the easier for programmers to work and can do editing if require.

It is noted that programmers having experience in editing their work and can re-figure will introduce comment just to remind about thinking in order to avoid decoding work.

You can add comments in PHP code by many ways. Initially you can use // to comment a line as shown:

```
<?php  echo "Hi";  //this is a comment  echo " How are you";  ?>
```

If you have single line comment, so there is another option of introducing comment by using a # sign as shown:

```
<?php  echo "Hi";  #this is a comment  echo " How are you";  ?>
```

If you have no multi line comment, then you can comment by using /* */ as shown in example:

```
<?php  echo "Hi";  /*  Using this  you can create big block of text that comments as  */  echo " How are you";  ?>
```

Check your progress 2

1. We can write comment in PHP by using_____ .

   a. #

   b. /* */

   c. //

   d. All of these

2.4 Outputting Data to the Browser

It is seen that in PHP, the data can be outputted to the browser with various methods which are discussed below:

The print() Statement

The print() statement will outputs data when passed to it which looks like:

int print(argument)

All of following are plausible print() statements:

```
<?php
print("<p>Hello World</p>");
?>
<?php
$author = "K. J. Sharma";
print "<p>I like $author.</p>";
?>
```

The output of above statements will be:

I like K. J. Sharma.

We see that official syntax calls for use of parentheses will enclose the argument, and are not applied as print() is not technical function but language construct. The print() statement's will return false value as it return 1 immaterial of outcome.

The echo() Statement

Apart from print, you can use echo() statement for similar work. But both echo() and print() statements are different and is not of many used to readers and so are avoided. The example of echo()'s prototype will be:

```
void echo(string argument1 [, ...string argumentN])
```

To use echo(), provide with:

```
echo " I like K. J. Sharma.";
```

From the prototype, echo() is able to output many strings. Study the example shown:

```
<?php
$he = "Rohit";
$she = "Vandana";
echo $he, " and ", $she, " are great couple.";
?>
```

This code produces the following:

Rohit and Vandana are great couple.

On executing following variation of above syntax will generate output as:

echo "$he and $she are great couple.";

The printf() Statement

The printf() statement is good in order to blend output with static text and dynamic information which is kept in one or more variables. This statement separates static and dynamic data in two different sections which allow easy maintenance and also does large control over dynamic information required to screen in terms of type, precision, alignment and position as shown:

integer printf(string format [, mixed args])

To insert single dynamic integer value otherwise static string:

printf("what is the time %d hours", 21);

On executing this command will produce:

What is the time 21 hours

We see that %d is placeholder which is type specifier and d is an integer value that is kept in that position. When printf() statement work, lone argument, 100, will be inserted in placeholder.

The sprintf() Statement

The sprintf() statement is similar in function as printf() but in this output assigns to string instead of rendering to browser as shown:

string sprintf(string format [, mixed arguments])

An example follows:

$value = sprintf("$%.2f", 55.5); // $value = $55.50

Check your progress 3

1. What is the difference between printf and sprint?

   a. Both are same

   b. Both are different

   c. Outputs are assigned to strings in sprintf, Unlike printf in which it is assigned to browser.

   d. None of these

2. What is the difference between echo and print?

a. Echo has no return value while print has a return value of 1

b. Echo can take multiple parameters while print can take one argument.

c. Echo is faster than print.

d. All of these

2.5 PHP's Supported Data types

Variables store data of many types which does different work. PHP supports following types of data types:

- String
- Integer
- Float (floating point numbers - also called double)
- Boolean
- Array
- Object
- NULL
- Resource

PHP String

A string is sequence of characters that can be "K. J. Sharma" which is inside quotes or double quotes as shown in example:

```
<?php
$x = "K. J. Sharma";
$y = 'K. J. Sharma';
echo $x;
echo "<br>";
echo $y;
?>
```

Output

K. J. Sharma

PHP Integer

The integer data type is without decimal that ranges from -2,147,483,648 and 2,147,483,647 and carries certain rules:

- An integer should have at least 1 digit
- An integer should have 1 decimal point
- An integer can be positive or negative
- An integers specifies formats such as decimal (10-based), hexadecimal (16-based) or octal (8-based)

The PHP var_dump() function returns the data type and value as shown in function example:

```
<?php
$x = 1234;
var_dump($x);
?>
```

Output:

Int(1234)

PHP Float

Float is a floating point number having decimal point or exponential form. In an example shown, $x is float where PHP var_dump() function returns data type as:

```
<?php
$x = 12.345;
var_dump($x);
?>
```

Output:

float(12.345)

PHP Boolean

A Boolean has only two values which can be TRUE or FALSE.

```
$x = true;
$y = false;
```

Booleans are often used in conditional testing.

PHP Array

Array stores many values in single variable. From the example, $scooters is an array which returns PHP var_dump() function as:

```
<?php
$cars = array("Bajaj","Priya","LML");
var_dump($scooters);
?>
```

Output:

```
array(3) { [0]=> string(5) "Bajaj" [1]=> string(3) "Priya" [2]=> string(6) "LML" }
```

PHP Object

Object are data type that keep data and information about data processing. It is noted that in PHP, object should clearly be declared. Initially we have to declare a class of object. A class is a structure that has properties and methods such as:

```
<?php
class Scooter {
   function Scooter() {
      $this->model = "Priya 112";
   }
}
// create an object
$herbie = new Scooter();
// show object properties
echo $herbie->model;
?>
```

Output:

```
Priya 112
```

PHP NULL Value

Null is special data type having single value: NULL. A variable of data type NULL is variable having no assigned value. If variable is without a value, it is directly given with NULL value. Example

```
<?php
$x = "Name!";
$x = null;
var_dump($x);
?>
```

Output:

null

PHP Resource

It is a different type of resource which is not real data type which stores reference to functions and resources that is external to PHP.

Check your progress 4

1. How can we declare a variable in PHP?
   a. $num
   b. #num
   c. $21
   d. None of these
2. What is resource in PHP?
   a. It is special data type having value: NULL
   b. They are special variables that hold references to resources external to PHP
   c. are instances of programmer-defined classes
   d. All of these

2.6 Identifiers

Identifier is a name used to name variables, functions, constants and classes. In this the first character be either ASCII letter, underscore or any characters between ASCII 0x7F and ASCII 0xFF. After initial character, such characters and digits 0-9 are valid.

Variable names

Variable names begin with dollar sign ($) and are case-sensitive such as:

$copy

$head_count

$MaxArea

$I_HEART_PHP

$_self

$_int

There exist certain uncommon variable names like:

$not valid

$|

$3wa

Function names

Function names are not case-sensitive such as:

tally

list_all_users

deleteTclFiles

Class names

Class names have standard rules for identifiers that are not case-sensitive like:

People

Account

Constants

Constants are identifier for simple scalar values which can be Boolean, integer, double and string constants.

Keywords

It is a reserved word by language for its main function that cannot assign variable, function, class or constant.

Check your progress 5

1. Which of the following statement is true?
   a. Variable names are not case sensitive.
   b. Function names are not case sensitive.
   c. Class names are case sensitive.
   d. All of these.

2.7 Variables

A variable in PHP serves as containers which used for keeping information. It starts with $ sign followed by name of variable as shown in example:

```
<?php
$txt = "Solving Integers";
$x = 11;
$y = 16.7;
?>
```

Output:

Solving Integers

11

16.7

On executing above program, variable $txt will keep the value Solving Integers, where variable $x will keep value 11 while variable $y keep value 16.7. Variable have short name or can be descriptive.

Rules for PHP variables:

- Variable starts with $ sign followed by name of variable
- Variable name to begin with letter or underscore

- Variable name cannot begin with number
- Variable name only has alpha-numeric characters and underscores
- Variable names are case-sensitive

Output Variables

The PHP echo statement is mainly applied to output data to screen as shown in example:

```
<?php
$txt = "K. J. Sharma Books";
echo "I like!";
?>
```

Output:

I like K. J. Sharma Books

The program shown will generate sum of two variables, as shown:

```
<?php
$x = 8;
$y = 8;
echo $x + $y;
?>
```

Output:

16

In above example we see that we have not instructed PHP about data type. PHP will directly convert variable to correct data type as per its value.

PHP Variables Scope

In PHP, variables are declared in the script. The scope of variable is part of script where variable can be used. PHP has three different variable scopes:

- local
- global
- static

Global and Local Scope

A variable declared outside function has GLOBAL SCOPE and can be accessed outside a function as seen in example:

```
<?php
$x = 10; // global scope
function myTest() {
   // using x inside this function will generate an error
   echo "<p>Variable x inside function is: $x</p>";
}
myTest();
echo "<p>Variable x outside function is: $x</p>";
?>
```

Output:

Variable x inside function is:

Variable x outside function is: 10

A variable that is declared in a function contains LOCAL SCOPE which can be accessed in a function as shown in example:

```
<?php
function myTest() {
   $x = 12; // local scope
   echo "<p>Variable x inside function is: $x</p>";
}
myTest();
// using x outside the function will generate an error
echo "<p>Variable x outside function is: $x</p>";
?>
```

Output:

Variable x inside function is: 12

Variable x outside function is:

The global Keyword

The global keyword is applied in order to access global variable within a function. To do this, use global keyword before the variables such as:

```
<?php
$x = 12;
$y = 18;
function myTest() {
   global $x, $y;
   $y = $x + $y;
}
myTest();
echo $y; // outputs 30
?>
```

Output:

30

It is noted that PHP stores global variables in array $GLOBALS[index] where index keeps name of variable. Such array is accessible in functions and applied to adjust global variables as shown in example:

```
<?php
$x = 9;
$y = 19;
function myTest() {
   $GLOBALS['y'] = $GLOBALS['x'] + $GLOBALS['y'];
}
myTest();
echo $y; // outputs 28
?>
```

Output:

28

The static Keyword

In PHP when functions executes, all variables gets vanished. The local variables are required for further work as shown in example:

```
<?php
function myTest() {
   static $x = 5;
   echo $x;
   $x++;
}
myTest();
myTest();
myTest();
?>
```

Output:

5

6

7

Here every time the function is called with variable having information contained from last when function was called.

Check your progress 6

1. What is the output of following code?

```
<?php
$num1 = 7;
$num2 = 15;
function test() {
   global $num1, $num2;
   $num2 = $num2 + $num1;
}
test();
echo $num2;
?>
```

   a. 8

   b. -8

   c. 0

   d. None of these

2. What is the output of the following code?

```
<?php
   $a = 'alen';

   $b = "My name is $a";
   echo $b;
    ?>
```

   a. "My name is $a"

   b. “"My name is alen"

   c. My name is alen

   d. None of these

2.8 Constants

Constants are variables which cannot be defined unless changed or undefined. It is an identifier (name) for simple value which cannot be changed during script. So valid constant name begins with letter or underscore. To create a constant, you require to define() function.

Syntax

define(name, value, case-insensitive)

Parameters:

name: Specifies name of constant

value: Specifies value of constant

case-insensitive: Shows whether constant name should be case-insensitive where default is not correct. The example below will form constant with case-sensitive name such as:

```
<?php
define("GREETING", "Welcome to my Library of Books!");
echo GREETING;
?>
```

Output:

Welcome to my Library of Books|

Further we see that an example below describes case-insensitive name:

```
<?php
define("GREETING", "Welcome to my Library of Books!", true);
echo greeting;
?>
```

Output:

Welcome to my Library of Books!

Constants are direct global applied across the script. The example uses constant in a function, even if it is defined outside the function:

```
<?php
define("GREETING", " Welcome to my Library of Books!");
```

```
function myTest() {
    echo GREETING;
}
myTest();
?>
```

Output:

Welcome to my Library of Books!

Check your progress 7

1. What is the output of the following code?

```
<?php
  define("HELLO", 100);
  echo HELLO;
  ?>
```

a. Hello
b. HELLO
c. 100
d. None of these

2.9 Expressions

Expressions are bits of PHP which calculates to generate a value. It carries simple expressions having literal values and variables that shows literal value calculates it as variable calculates value stored in variable.

In PHP, expressions are important building stones which can be applied for writing an expression. The simplest expression is anything that has value. The basic feature of expressions results are constants and variables.

It is noted that when you type "$a = 10", then it shows that '10' assigns to $a. '10', has value 10, or we see that '10' is an expression with value of 10.

After an assignment, $a's value as 10, hence if we wrote $b = $a, then $b = 10. In other words, $a is expression with value 10. It is noted that PHP takes expressions much further in same way as other languages. In an example '$a = 10' it is noted that there exists two values, value of integer constant '10' and value of $a that updates to 10 as well. In practice, we see that '$a = 10' is an expression having value 10. So writing like '$b = $a = 10' is like '$a = 10; $b = 10;'. As assignments are arranged from right to left, so we can write '$b = $a = 10'.

Check your progress 8

1. Expression for adding two number is_____.
   a. a+b
   b. $a+ $b
   c. @a+@b
   d. None of these

2.10 String Interpolation

In PHP when it is required to control code to be worked and should want to beautify code, in such case these string interprets which provide with complete flexibility while working with string values such as

The $car moved over the breaker.\n

In this, we see that $car is a variable and \n is newline character so both be interpreted. If you want to output string exactly as written or want newline to render but want variable to display in literal form($car) or vice versa.

Double Quotes

Strings in double quotes are applied in PHP scripts as they give flexibility since both variables and escape sequences are parsed as shown in example:

```
<?php
$book = "Electronics";
echo "Sanjay's favourite book is $book.";
?>
```

This example returns the following:

Sanjay's favorite book is Electronics.

Single Quotes

Keeping a string in single quotes is important while interpreted string exactly similar as stated that has variables and escape sequences not interpreted when string is parsed as shown:

```
print 'This string will $print exactly as it\'s \n declared.';
```

It shows output as:

This string will $print exactly as it's \n declared.

We see that single quote gets out which delete backslash escape character and result in syntax error as shown:

```
print 'Find other string.\\';
```

It shows output as:

Find other string.\

In above example we see that backslash appears at conclusion of string which has escaped else PHP will parser that understand single quote that escaped.

```
print 'Find \another string.';
```

It shows output as:

Find \another string.

Check your progress 9

1. Choose correct statement to display the line as it is Hello World.HaVe a nIce dAy
   a. echo " Hello World..HaVe a nIce dAy"
   b. echo 'Hello World.HaVe a nIce dAy'
   c. Syntax error
   d. None of these

2.11 Control Structures

In PHP, control structures are loops, conditionals and if which are familiar to javascript programmer such as:

- if and else
- elseif
- while
- do...while
- for
- foreach
- break and continue
- switch
- Reference

if and else

It is same as javascript like:

```
if ($a==$b) {
print "The same";
}
else {
print "They aren't the same";
}
```

elseif

elseif also else if gives a second if-statement where first if-condition if met run other elseif conditions even if not run.

```
$a="Sanjay";
if($a=="Sanjay"){
print"Hello Sanjay";}
elseif($a=="Sharma"){
print"Hello Sharma";
```

```
}
else if ($a=="Sanjay"){
print "Hello Sanjay 2";
}
else {
print"I don't know you!";
}
//result: Hello Sanjay
```

while

While loop is similar as in javascript:

```
$i=1;
while ($i<10){
print "Hello";
$i++;
}
```

do..while

do..while loop is same as while loop but while loop will not run if condition does not exists. The do..while loop always run at least once as shown in example:

```
$i=1;
do{
print "The number is $i";
}
while($i>100);
//result: The number is 1
```

for

The for loop in PHP is same as in javascript's:

```
for($i=1;$i<=2;$i++){
print "This is $i<br>";
}
```

```
//This is 5
//This is 6
```

The for loop can be written in various ways. One additional way is as follows:

```
for($i=1;$i<=3;print "This time it's $i ",$i++);
//result: This time it's 5 This time it's 6 This time it's 7
```

foreach

This control structure access items in array:

```
$arr=array("cow","deer","gost");
foreach($arr as $value){
echo"Value: $value<br>\n";
//Value: cow
//Value: deer
//Value: gost
```

This example shows the keys and values of an associative array:

```
$a=array("fruit"=>"pineapple","meat"=>"gost","vegetables"=>"beans","sweet"=>
"ice cream");
foreach($a as $k=>$v){
print"\$a[$k]=>$v.\n";
};
//result: $a[fruit]=>pineapple. $a[meat]=>gost. $a[vegetables]=>beans.
$a[sweet]=>ice cream.
```

break and continue

Break and continue structures are common in javascript that exits for, while, or switch structure and exits in loop and begins with next.

switch

The switch statement will continue as shown which print all statements after finding as true as shown:

```
$i=1;
switch($i){ //variable in brackets and cases in curly brackets
case 0:
```

```
print "zero";
break;
case 1:
print "one";
break;
default: //You can do default case!
print "whatever it is, it isn't 0, 1 or 2";
}
//result: one
```

Check your progress 10

1. What is the output of the following code?

```
<?php
$a = 25;
if ($a < "30") {
   echo "Hello World!";
}
?>
```

   a. Hello World!
   b. 25
   c. 30
   d. None of these

2. What is the output of the following code?

```
<?php
$x = "1";
switch ($x)
{
case 1:
print "hello";
```

```
case 2:
print "to";
default:
print "world";
}
?>
```

a. helloworld

b. hellotoworld

c. syntax error

d. None of these

2.12 Let Us Sum Up

In this unit we have learnt that PHP is Hypertext Preprocessor applied by web developers to frame dynamic content which is easy to interact with databases. It is found that PHP is an important coding technique that can be easily embedded in HTML which uses page to be passed through PHP engine for interpretation.

It is known that comment in PHP code is a line that is not part of program but to read by person editing the code. Identifier are name used to name variables, functions, constants and classes where first character is ASCII letter, underscore or characters between ASCII 0x7F and ASCII 0xFF.

It is noted that constants are variables that are not defined unless changed or undefined which serves as an identifier for simple value that cannot be changed during script. As seen, expressions are bits of PHP that calculates value having simple expressions with literal values and variables.

2.13 Answers for Check Your Progress

Check your progress 1

Answers: (1 –c), (2-b)

Check your progress 2

Answers: (1 –d)

Check your progress 3

Answers: (1 –c), (2-d)

Check your progress 4

Answers: (1 –a), (2-b)

Check your progress 5

Answers: (1 –b)

Check your progress 6

Answers: (1 –a), (2-c)

Check your progress 7

Answers: (1 –c)

Check your progress 8

Answers: (1 –b)

Check your progress 9

Answers: (1 –b)

Check your progress 10

Answers: (1 –a), (2-b)

2.14 Glossary

1. **Variables -** They are containers used for keeping information.
2. **Constants -** Variables that are not defined till it changes or undefined and also serves as identifier.
3. **Expression -** These are bits in PHP that finds generated value.

2.15 Assignment

Describe the structure of control statements.

2.16 Activities

Write short note on PHP's Supported Data types.

2.17 Case Study

How to embed PHP Code in Web Pages?

2.18 Further Readings

1. Programming with PHP, Programmer’s Guide by Ronald.
2. Building Secure PHP Apps by Coulouris, Dollimore, Kindberg.

Block Summary

In this block, you will understand about the basic of identifiers with concept about ASCII letter, underscore or characters as ASCII 0x7F and ASCII 0xFF. The block gives an idea on architecture of PHP supported data types with study about their working characteristics. The examples related to Control Structures and Comments in PHP code are also discussed.

In this block, you will understand about the basic of remote procedure calls and its techniques. The concept related to basic of Identifiers, Variables, Constants and Expressions are also detailed. You will be demonstrated practically about important coding technique that can be easily embedded in HTML.

Block Assignment

Short Answer Questions

1. Write a short note on PHP datatypes.
2. Define the architecture of a web application.
3. Explain decision making in PHP using examples.

Long Answer Questions

1. Discuss PHP constants and variables in detail with the help of examples.
2. Write a PHP code to add two numbers and display the output.
3. Write a program to check whether the given number is odd or even using if-else loop.
4. How static web pages are different from dynamic web page? Explain in detail

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____ _____
Language and Style	☐	☐	☐	☐	_____ _____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____ _____
Conceptual Clarity	☐	☐	☐	☐	_____ _____
Check your progress Quest	☐	☐	☐	☐	_____ _____
Feed back to CYP Question	☐	☐	☐	☐	_____ _____

3. Any Other Comments

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

WEB APPLICATION DEVELOPMENT

PGDCA 202

BLOCK 2:
Array, Function and Expression

Dr. Babasaheb Ambedkar Open University
Ahmedabad

WEB APPLICATION DEVELOPMENT

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Mr. Sanjay Thapar

Language Editor
Mr. K. J. Sharma

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Mr. Prashant Tikone

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included may tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

WEB APPLICATION DEVELOPMENT

Contents

BLOCK 3: OO AND FILE HANDLING IN PHP

UNIT 1 OBJECT ORIENTED PHP

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

UNIT 2 FILE AND DIRECTORY HANDLING

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP

UNIT 1 PHP AND MYSQL DATABASE

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

UNIT 2 STATE MANAGEMENT IN PHP

Session, Cookies

Dr. Babasaheb
Ambedkar
Open University

PGDCA 202

Web Application Development

BLOCK 2: ARRAY, FUNCTION AND EXPRESSION

BLOCK 2: ARRAY, FUNCTION AND EXPRESSION

Block Introduction

Arrays are basically applied in order to store and organize data efficiently and quickly within no time. Hence an array having numeric index is called an indexed array while array having name is called an associative array.

In this block, we will detail about the basic of Arrays with their features associated with storing and organizing of data with required efficiency. The block will focus on the study and concept of Error handling and Exception handling concept in respect to code execution with knowledge on exceptional conditions. The students will give an idea on Perl style regular expressions with comparison with POSIX counterparts.

In this block, you will make to learn and understand about the basic of PHP in built functions with their features. The concept related to structure of POSIX regular expression with knowledge about typical arithmetic expression is well explained to you. You will be demonstrated practically about POSIX regular expression implementation.

Block Objective

After learning this block, you will be able to understand:

- About File systems-structure and partition
- Basic of Disk space allocation
- Features of Disk scheduling
- Concept of I/O Hardware and Drivers
- Detailed about DMA controlled I/O

Block Structure

Unit 1: Array in PHP

Unit 2: Function in PHP

Unit 3: Expression

UNIT 1: ARRAY IN PHP

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About array creation
- About adding array element
- About finding array size

1.1 Introduction

Array is an important part of any programming language since it carries important facilities and built-in functionality which is required by any programming language. In PHP, there exists strong set of efficient ways that will help in dealing with such arrays. Arrays are basically applied in order to store and organize data efficiently and quickly within no time. It serves as an important data

a type which is present in any programming language as it is easily described as ordered list of elements. Here you can access single elements simply by referring their index position which is available inside an array. Its position is shown by numeral or alphabet. Hence an array having numeric index is called an indexed array while array having name is called an associative array.

1.2 Creating an array

Array serves as a map since every key is mapped with some value. In an programming language, the role of arrays is to store data in cases when variable results in many values. The array() function is applied to create an array. We can create array in PHP by two ways::

Way 1. Creating an array by passing the value in array function

```
<?php
$arr = array(
   “Sanjay”,
   “Rohit”,
   “Amit”
);
?>
```

We can also create array as:

```
<?php
$arr = array(
   “Sanjay”,
   Rohit”,
   “Amit”,
   array(
      “Delhi”,
      “Mumbai”,
      “Pune”
   );
```

```
?>
```

Way 2. Creating an array by creating an index

```
<?php
$arr[] = “Sanjay”;
$arr[] = “Rohit”;
$arr[] = “Amit”;
?>
```

We see that we can create three types of Arrays available in PHP which are:

Indexed array

In PHP, an indexed array is a type of array which is created inside an index. It can be created in two ways as shown.

```
<?php
$arr = array(
   “Neha”,
   “Preeti”,
   “Anika”
);
Print_r($arr);
?>
```

Associative array

In PHP, associative array is a type of array having a key that is associated with every value where a string can be used as index of array.

```
<?php
$employee = array(
   “name” => “Sanjay”,
   “address” => “Pune”,
   “batch” => “1998”
);
Print_r($employee);
```

```
?>
```

Multidimensional array

In PHP, multidimensional array is a type of array having another array as value as shown:

```
<?php
// A two-dimensional array:
$scooters = array(
  array(
    "Mahindra",
    100,
    80
  ) ,
  array(
    "Bajaj",
    80,
    60
  ) ,
  array(
    "LML",
    100,
    90
  )
);
?>
```

Example for an Indexed array:

```
<?php
echo "Index Array";
$arr = array(
  "Sanjay",
```

```
    "Rohit",
    "Amit"
);
Print_r($arr);
$array[] = "Sanjay";
$array[] = "Rohit";
$array[] = "Amit";
Print_r($array);
?>
```

Output:

```
<?php
IndexArray Array(
    [0] => Sanjay[1] => Rohit[2] => Amit
) Array(
    [0] => Sanjay[1] => Rohit[2] => Amit
)
?>
```

Example for an Associative array:

```
<?php
echo "Associative Array";
$array1 = array(
    "name" => "Sanjay",
    "address" => "Pune",
    "batch" => "1998"
);
print_r($array1);
echo "";
foreach ($array1 as $key => $value) {
    echo "array1[$key] = $value";
```

```
}
echo "";
?>
```

Output:

```
<?php
AssociativeArray Array(
   [name] => Sanjay[address] => Pune[batch] => 1998
) array1[name] = Sanjayarray1[address] = Punearray1[batch] = 1998
?>
```

Check your progress 1

1. Array contains:
   a. string
   b. number
   c. data
   d. all of above

1.3 Adding and removing array element

While working with arrays, it not certain that the particular will fit or is not required. Every time you need to check by adding or removing elements to the array. In PHP, we see that there are certain built-in functions which are framed that will attach or detach elements right from beginning or ends in numerically indexed array.

Adding an Element in an array

Initially we will start with how to add elements to an array. We can add elements in an array by two simple ways. First we can “push” element onto array with the help of array_push function as it is needed when you want to add multiple elements to end of array at the same time. To use array_push function, you need to have the following code:

view plaincopy to clipboardprint?

```
$myarray = array(‘cow’,‘bull’);
```

array_push($myarray,‘snake’);

With the help of above code, we can create a new array which is $myarray in which we add single element. After adding, we used array_push function so as to add another new element to an array. We see that the new elements gets added to the end of array in similar order as listed in the function. Hence the order of elements in an array will now be ‘cow’, ‘bull’, ‘snake’.

On adding single new element to existing array, simply apply empty brackets, hence rather than applying array_push($myarray,’snake’) we use $myarray[] = ‘snake’.

Removing an Element from an array

To make the last element out of an array, you need array_pop function. With the help of array_pop function, you can easily remove last element in array and will able to keep particular value in new variable. Consider an example for this:

view plaincopy to clipboardprint?

```
$filename = ‘stye.css’;
$filename = explode(‘.’,$filename);
$extension = array_pop($filename);
```

The above code will convert string stye.css into an array and gets split up wherever dot comes in the string. Once that appears we will finish with array having elements ‘stye’ and ‘css’ in particular order. After it we use array_pop function so as to remove ‘css’ from array and keep it as new variable $extension.

Shifting and Unshifting an Element in an array

By now you will be able to have an idea regarding the adding of new elements and removing of elements from end of an array. Many times there arte possibilities that you need to add element at the beginning of array. In PHP, there are certain options available where there exists two functions which can be applied in order to add new elements or remove elements from beginning of array.

For this, you will apply PHP built-in function array_unshift in order to add element in the start of array as shown in example below.

view plaincopy to clipboardprint?

```
$myarray = array(‘cow’,‘bull’);
array_unshift($myarray,‘cat’);
```

From the above example, we see that new element gets added at the beginning of array and it will result as ‘cat’, ‘cow’, ‘bull’.

You can remove the first element from an array using array_shift function as shown in example:

view plaincopy to clipboardprint?

```
$myarray = array(‘cow’,‘bull’, ‘cat’);

array_shift($myarray)
```

By using array_shift function, you will find that first element cow gets removed and the array is left with ‘bull’, ‘cat’.

Check your progress 2

1. Array array_push function is required in order to add:
   a. single element
   b. two elements
   c. three elements
   d. many elements
2. Array_shift function is required to remove:
   a. first element
   b. middle elements
   c. second elements
   d. last elements

1.4 Determining array size

In PHP, the sizeof() function will return number of elements in an array which is alias of count() function. The syntax of size array is sizeof(array,mode). The sizeof function in an array is used to count number of words which is available in the paragraph. It can be determined by:

- Storing paragraph of text in string variable
- Breaking variable and creating array using split function with space as delimiter

- Counting number of elements available in an array

Consider an example program where the arrays are counted recursively:

```
<?php
$fourwheelers=array
 (
 "Jeep"=>array
 (
 "JC60",
 "JC90"
 ),
 "Toyota"=>array
 (
 "X7",
 "X9"
 ),
 "Maruti"=>array
 (
 "Cidan"
 )
 );
echo "Normal count: " . sizeof($fourwheelers)."<br>";
echo "Recursive count: " . sizeof($fourwheelers,1);
```

If you run the above program, you will get an output as:

Normal count: 7

Recursive count: 16

Check your progress 3

1. In an array, sizeof() function is used to:

   a. add number of elements in an array

   b. return number of elements in an array

   c. overwrite number of elements in an array

   d. none of above

1.5 Uniqueness sorting array

In PHP, the array_unique() function will able to remove double values from an array. It is noted that when two or more array values are similar, then first will be kept while other will be removed. The returned array will hold the initial array item's key type. The syntax is array_unique(array).

Consider an example program of removing duplicate values from an array:

```
<?php
$a=array("a"=>"table","b"=>"chair","c"=>"table");
print_r(array_unique($a));
?>
```

On running the above program, we will get an output as:

```
Array ( [a] => table [b] => chair )
```

Check your progress 4

1. In an array, array_unique() function is used to:

   a. add double values from an array

   b. remove double values from an array

   c. add single value in an array

   d. remove single value from an array

1.6 Let Us Sum Up

In this unit we have learnt that arrays stores and organizes data efficiently and quickly within no time and carries numeric index and name index. It is seen that array serves as map as every key is mapped with value which gets stored in an array.

In PHP, we see that there are certain built-in functions which are framed that will attach or detach elements right from beginning or ends in numerically indexed array.

The sizeof() function is applied in order to return the number of elements in an array which is same as count() function with syntax as sizeof(array,mode). It is found that array_unique() function removes double values from an array where on having two or more array values, the first will be there while other gets removed.

1.7 Answers for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-d), (2-a)

Check your progress 3

Answers: (1-b)

Check your progress 4

Answers: (1-b)

1.8 Glossary

1. **Array -** A homogeneous container of numerical elements that occupies fixed amount of memory.
2. **Attribute -** It is the property of an object which will able to access by using obj.attribute,

3. **Instance -** It is a class definition which shows blueprint for framing object.
4. **Sizeof() function -** In an array, sizeof() function returns number of elements.

1.9 Assignment

Write short note on arrays?

1.10 Activities

Study about the importance of sizeof() function in an array.

1.11 Case Study

Discuss the importance of array_unique() function in an array.

1.12 Further Readings

1. Basic of Arrays, Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, 1996.
2. Array Programming in PHP by Dan Sydow, 2003.
3. Introduction to PHP, Chandresh Shah,2010.
4. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
5. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.

UNIT 2: FUNCTION IN PHP

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About PHP functions.
- The features of Argument function.
- Various built-in functions in PHP.

2.1 Introduction

In PHP, function resembles with the name given by us for bunch of code which to be executed whenever it is required by us. By using functions in the code, you will save lot of time and effort and can easily be able to write a code which is much readable. On creating a function, initially you require to give a name to it as my Company Mission. With the help of function name, you can simple able to call it which makes simple and easy to type and understand.

It is found that the real syntax for creating a function is self-explanatory as initially you have to tell PHP about creating a function. For this, you need to type the keyword function which can be along function name.

2.2 In built functions

In PHP, there appears many functions which are built-in. Apart from these, there are many functions which requires particular PHP extensions, failure of which lead to fatal errors. It is noted that in case of imagecreatetruecolor() function, PHP will compiled with GD support or with MySQL support. The list below contains different function categories such as:

Array Functions

Calender Functions

Class/Object Functions

Character Functions

Date & Time Functions

Directory Functions

Error Handling Functions

File System Functions

MySQL Functions

Network Functions

ODBC Functions

Array, function and expression

String Functions

Simple XML Functions

XML Parsing Functions

2.2.1 String functions

There are many string functions that come with PHP which are not to be installed, such as:

addcslashes()	It returns string with backslashes in front of specified characters
addslashes()	It returns string with backslashes in front of predefined characters
bin2hex()	It converts string of ASCII characters to hexadecimal values
chop()	It removes whitespace characters from right end of string
chr()	It returns character from particular ASCII value
chunk_split()	It splits string into series of tiny parts
count_chars()	It return information about characters used in string
crc32()	It calculates 32-bit CRC for string
crypt()	It is one-way string hashing
echo()	It gives one or more strings
explode()	It breaks string into an array
fprintf()	It writes formatted string to particular output stream
hebrev()	It converts Hebrew text to visual text
hebrevc()	It converts Hebrew text to visual text and new lines (\n) into
hex2bin()	It converts string of hexadecimal values to ASCII characters
htmlentities()	It converts characters to HTML entities

implode()	It returns string from elements of an array
lcfirst()	It converts first character of string to lowercase
md5()	It will find MD5 hash of string
md5_file()	It will find MD5 hash of file
metaphone()	It will find metaphone key of a string
nl_langinfo()	It returns particular local information
ord()	It returns ASCII value of first character of a string
parse_str()	It will parses query string inside variables
print()	It will give one or more strings
printf()	It will give formatted string
setlocale()	It will set locale information
sha1_file()	It will show SHA-1 hash of a file
similar_text()	It will give similarity between two strings
soundex()	It will find soundex key of a string
sprintf()	It will write formatted string to variable
sscanf()	It will parses input from string as per format
str_repeat()	It will repeat string particular times
str_replace()	It changes characters in a string
str_rot13()	It will do ROT13 encoding on string
str_shuffle()	It shuffles all characters in string
str_split()	It splits string into an array
str_word_count()	It will count number of words in string
strcasecmp()	It will compare two strings
strip_tags()	It will strips HTML and PHP tags from string
strlen()	it will return length of string

strpbrk()	It searches string for any set of characters
strrchr()	It locates last occurrence of string in another string
strrev()	It will reverse the string
strtok()	It splits the string in tiny strings
strtolower()	It changes string to lowercase letters
strtoupper()	It changes string to uppercase letters
strtr()	It will translate characters in a string
substr()	It will return part of string
ucfirst()	It changes first character of string to uppercase
ucwords()	It will changes first character of every word to uppercase
vfprintf()	It will writes formatted string to particular output stream
vprintf()	It gives out formatted string
vsprintf()	It will write formatted string to variable
wordwrap()	It will wraps string to given number of characters

2.2.3 Array functions

In PHP, array functions are part of the software where again there will be no need to install these functions.

array()	This will creates an array
array_chunk()	It will splits array into portions of arrays
array_diff()	It will compare arrays values
array_fill()	It will fills array with values
array_filter()	It will filters values of array by callback function
array_keys()	It will return all keys of an array
array_merge()	It will merge one or more arrays in an array
array_pop()	It will erase last element of an array
array_push()	It will insert one or more elements to end of array
array_rand()	It will return one or more random keys from array

array_reduce()	It returns array as string with user-defined function
array_reverse()	It returns an array in the reverse order
array_search()	It will find array for given value and returns the key
array_slice()	It will return selected parts of an array
array_splice()	It will remove and replaces particular elements of array
array_sum()	It returns sum of values in array
array_unique()	It erase duplicate values from array
array_unshift()	It will add one or more elements to start of an array
array_values()	It will return all values of array
array_walk()	It applies user function to all member of array
compact()	It will form array having variables with their values
count()	It will return number of elements in array
current()	It will return present element in array
each()	It will return present key and value pair from array
end()	It will set internal pointer of array with last element
extract()	It takes variables in present symbol table from array
in_array()	It will check if particular value appears in an array
key()	It fetches key from an array
ksort()	It will sort associative array in increasing order, as per the key
list()	It will assigns variables on expectation of an array
natsort()	It will sort array by natural order algorithm
next()	It will advance internal array pointer of array
prev()	It rewinds internal array pointer
range()	It forms array having range of elements
reset()	It will set internal pointer of array to its first element
shuffle()	It will jumble the array
sort()	It will sort indexed array in increasing order
uasort()	It will sort array by values using user-defined function
usort()	It sort array by user-defined comparison function

2.2.4 Mathematical functions

In PHP there are built-in math functions which will take care of values that appear inside range of integer and float types. Following are the list of math functions in PHP.

abs()	It will return positive value of a number
acos()	It will return arc cosine of a number
acosh()	It will return inverse hyperbolic cosine of number
asin()	It will return arc sine of a number
asinh()	It will return inverse hyperbolic sine of number
atan()	It will return arc tangent of number in radians
atan2()	It will return arc tangent of two variables x and y
atanh()	It will return inverse hyperbolic tangent of number
base_convert()	It will convert number from one number base to another
bindec()	It will convert binary to decimal number
ceil()	It will round number to nearest integer
cos()	It will return cosine of a number
cosh()	It will returns hyperbolic cosine of a number
decbin()	It will change decimal to binary number
dechex()	It will change decimal to hexadecimal number
decoct()	It will change decimal to octal number
deg2rad()	It will change degree to radian value
exp()	It will find exponent of e
expm1()	It will returns exp(x) - 1
floor()	It will round number down to nearest integer
fmod()	It will return remainder of x/y
getrandmax()	It will return largest possible value returned by rand()
hexdec()	It will change hexadecimal to decimal number

hypot()	It will change hypotenuse of right-angle triangle
is_finite()	It will check for finite value
is_infinite()	It will check for infinite value
is_nan()	It will check for value of number
lcg_value()	It will return pseudo random number in range from 0 to1
log()	It will return natural logarithm of a number
log10()	It will return base-10 logarithm of a number
log1p()	It will return log(1+number)
mt_rand()	It will form random integer by Mersenne Twister algorithm
octdec()	It will change octal to decimal number
pi()	It will return value of PI
pow()	It will return x^y
rad2deg()	It will change radian to degree value
rand()	It will form random integer
round()	It will round floating-point number
sin()	It will return sine of a number
sinh()	It will return hyperbolic sine of a number
sqrt()	It will return square root of a number

2.2.5 File System functions

In PHP, file system functions are used in programming which comes with buit-in features. The list below shows file system functions such as

basename	Returns trailing name component of path
chgrp	Changes file group
chmod	Changes file mode
chown	Changes file owner
clearstatcache	Clears file status cache
copy	Copies file

:

Array, function and expression

delete	See unlink or unset
dirname	Returns a parent directory's path
fclose	Closes an open file pointer
feof	Tests for end-of-file on a file pointer
fflush	Flushes the output to a file
fgetc	Gets character from file pointer
fgetcsv	Gets line from file pointer and parse for CSV fields
fgetc	Gets character from file pointer
fgetcsv	Gets line from file pointer and parse for CSV fields
fgets	Gets line from file pointer
file	Reads entire file into an array
fileatime	Gets last access time of file
fileinode	Gets file inode
filesize	Gets file size
filetype	Gets file type
fopen	Opens file or URL
fread	Binary-safe file read
fseek	Seeks on a file pointer
glob	Find pathnames matching a pattern
is_dir	Tells whether the filename is a directory

is_executable	Tells whether the filename is executable
is_file	Tells whether the filename is a regular file
is_link	Tells whether the filename is a symbolic link
is_readable	Tells whether a file exists and is readable
link	Create a hard link
linkinfo	Gets information about a link
mkdir	Makes directory
pathinfo	Returns information about a file path
pclose	Closes process file pointer
popen	Opens process file pointer
readfile	Outputs a file
readlink	Returns the target of a symbolic link
rename	Renames a file or directory
rewind	Rewind the position of a file pointer
rmdir	Removes directory
stat	Gives information about a file
symlink	Creates a symbolic link
tempnam	Create file with unique file name
tmpfile	Creates a temporary file
touch	Sets access and modification time of file
unlink	Deletes a file

2.2.6 Date and Time Functions

In PHP, date and time functions are built-in that are used in PHP scripts. Such functions will be applied to format date and time that appears in many ways. The PHP date() function will format timestamp with syntax as date(format,timestamp)

Array, function and expression

checkdate()	Validates a Gregorian date
date_create()	Returns new DateTime object
date_date_set()	Sets the date
date_format()	Returns date formatted according to given format
date_modify()	Alters the timestamp
date_sunrise()	It will return time of sunrise for given day / location
date_sunset()	It will return time of sunset for given day / location
date_time_set()	It will set the time
date()	It will format local time/date
gettimeofday()	It will returns array having current time information
gmdate()	It will format GMT/UTC date/time
gmmktime()	It will return Unix timestamp for GMT date
gmstrftime()	It will format GMT/UTC time/date as per locale settings
idate()	It will formats local time/date as integer
microtime()	It will return microseconds for current time
mktime()	It will return Unix timestamp for a date
strftime()	It will format local time/date according to locale settings
strptime()	It will parse time/date generated with strftime()
time()	It will return current time as Unix timestamp
timezone_open()	It will returns new DateTimeZone object

2.2.7 Miscellaneous Functions

In PHP there are many built-in miscellaneous functions that are:

connection_aborted	It will check for client disconnection
connection_status	It will return connection status bitfield
constant	It will return value of a constant
define	It will explain named constant
defined	It will check for given named constant
die	It is same as exit
eval	It will find a string as PHP code
exit	It will show message and terminate current script
get_browser	It direct about user's browser capacity
highlight_file	It shows syntax by highlighting a file
pack	It will pack data into binary string
sleep	It will delay execution
uniqid	It will generate unique ID
unpack	It will unpack data from binary string
usleep	It will delay execution in microseconds

2.2.8 User Defined Functions

In PHP, there are more than 1,000 in-built functions which are called as per requirement. Such functions are known as User Defined Function. The syntax of this function is:

```
function function-name()
{
  statement 1 :
  statement 2 :
  statement 3 :
   ......
}
```

It is found that function declaration begins with function.

Name of function:

- It is defined by user.
- In this, valid function begins with letter or underscore with letters, numbers, or underscores.
- In this, function names are case-insensitive.

Opening and Closing curly braces ({ }):

It is noted that function body is enclosed in pair of braces that has variable names and actual function code. Opening of curly bracket ({) shows start of function code while closing of curly bracket (}) shows termination of function.

Example: PHP function

view plaincopy to clipboardprint?

```
<?php
function myfunction()
{
echo "How are you";
}
myfunction();
?>
```

Output:

When we call this function it will display:

How are you

Example: Functions within functions

```
view plaincopy to clipboardprint?
<?php
function function1()
{
function function2()
{
echo "How are you <br>";
}
}
function1();
function2();
?>
```

In this program we see that function1() is declared and function2() is declared inside function1(). So first we have to execute function1() and then function2() which will show "How are you".

By executing function1() makes funtion2() workable, hence we cannot call function2() directly without calling function1().

2.2.9 Arguments passing by reference

In PHP, we see that by default, function arguments are passed by value. By allowing function to change the arguments, the arguments must be passed by reference.

Also, you will able to pass variable by reference to function hence function will able to modify the variable. The syntax of argument passing by reference is:

```
<?php
function foo(&$var)
{
   $var++;
}
$a=8;
foo($a);
```

```
// $a is 9 here
?>
```

Now, you can also pass an argument to function as reference by adding ampersand (&) character before variable name as shown.

view plaincopy to clipboardprint?

```
<?php
function cube(&$x)
{
$x = $x * $x * $x;
}
$result = 9;
cube($result);
echo $result;
?>
```

Output

729

2.2.10 Argument passing by value

In PHP, argument is any valid expression that is calculated and its value being assigned to required variable in a function. Consider a program, where function $a is assigned to value 20 and function $b is assigned to value 25:

```
function add($a, $b)
{
...
}
add(20, 25);
```

Check your progress 1

1. What is the use of explode() function?

   a. It breaks string into an array.

   b. It gives one or more string

   c. It spits string into parts.

   d. None of these

2. flush () is a type of _____.

   a. String function

   b. Date function

   c. File system function

   d. None of these

3. What is the output of the following code?

```
<?php
function write() {
    echo "Programming is a fun!";
}
write();
?>
```

   a. Write

   b. echo "Programming is a fun!"

   c. "Programming is a fun!"

   d. Programming is a fun!

2.3 Let Us Sum Up

While studying this unit, we have learnt that functions in PHP resembles with the name given by us for bunch of code which to be executed whenever it is required by us. It is noted that by applying functions in code, you will save lot of time and effort and can easily be able to write a code which is much readable.

In PHP, there appears many functions which are built-in that requires particular PHP extensions, failure of which lead to fatal errors.

2.4 Answers for Check Your Progress

Check your progress 1

Answers: (1 –a), (2 –c), (3-d)

2.5 Glossary

1. **Structured data type -** It is a type of data type which is made of other datatypes.
2. **Declaration -** A syntactical element describing name and type of one or more variables, functions, structures, unions, or enumerations.
3. **Object -** Piece of data that which is manipulated by a program using simple variable, array, structure.

2.6 Assignment

Write short note on Argument passing by value.

2.7 Activities

Try to implement all string functions.

2.8 Case Study

Write a program by using functions and pass parameters by using call by reference and call by value.

2.9 Further Readings

1. Introduction to PHP, Chandresh Shah, 2010.
2. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
3. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.

UNIT 3: Expression

Unit Structure

3.0 Learning Objectives

After learning this Unit, you will be able to understand:

- About Regular Expression.
- About Error Handling Regular Expressions.
- About Exception handling.

3.1 Introduction

In PHP expressions plays an important building stones which describes what you write is an expression. It can be called as an expression is anything having a value. Important forms of expressions are constants and variables. If you type "$x = 10", it means you have assigned '10' into $x. '10' obviously has value 10 or it can be '10' an expression having value 10. If we see $x's value to be 10, so $y = $a, also $y = 10. We see that, $x is an expression having value of 10.

3.2 Regular Expression

Regular expressions are strong tool which alter and changes text. These expressions having general pattern notation is like programming language that describes parse text which will search for patterns in string. Such types of expressions are slower as compared to normal expressions but are strong enough and are applied for particular use.

In PHP, these expressions are nothing but series or arrangement of characters which shows foundation for pattern matching functionality. With this, you can locate for particular string which available inside another string that can change one string with another and gets splited into pieces.

Regular expression has characters that will match among themselves as on searching "tra" in a string "Trains are fast," as you have a match since "tra" occurs in particular string. It is seen that many characters carry special meanings in regular expressions. It is noted that dollar sign ($) is applied to match strings which ends with required pattern. Also, caret (^) character if applied at start of regular expression will show that it will match starting of string. It is noted that regular expression contains own set of functions as:

preg_filter()	Performs a Regular Expression Search & Replace
preg_grep()	Returns Array Entries That Match the Pattern
preg_last_error()	Returns the Error Code of the Last PCRE Regex Execution
preg_match_all()	Performs a Global Regular Expression Match
preg_match()	Performs a Regular Expression Match
preg_quote()	Quote Regular Expression Characters
preg_replace()	Performs a Regular Expression Search & Replace
preg_split()	Splits a String by a Regular Expression

PHP offers functions that are particular to two sets of regular expression functions having certain type of regular expression which can be:

- POSIX Regular Expressions
- PERL Style Regular Expressions

POSIX Regular Expressions

The structure of POSIX regular expression is different of typical arithmetic expression where operators are combined to form complex expressions.

Brackets

Brackets ([]) carry special meaning if applied in context of regular expressions and are applied to have a range of characters as shown:

[0-9] It matches any decimal digit from 0 through 9.

[a-z] It matches any character from lower-case a through lowercase z.

[A-Z] It matches any character from uppercase A through uppercase Z.

[a-Z] It matches any character from lowercase a through uppercase Z

Quantifiers

The frequency or position of bracketed character sequences and single characters be shown by special character having particular connotation as +, *, ?, {int. range}, and $ flags:

p+ It matches any string having at least one p.

p* It matches any string having zero or more p's.

p? It matches any string having zero or more p's.

p{N} It matches any string having sequence of N p's

p{2,3} It matches any string having sequence of two or three p's.

p{2, } It matches any string having sequence of at least two p's.

pS It matches any string with p at end of it.

^p It matches any string with p at beginning of it.

PERL Style Regular Expressions

Perl style regular expressions are same as POSIX counterparts. In this, the syntax can be applied by changing with Perl style regular expression functions.

Meta characters

Meta characters are alphabetical character which comes before backslash and gives mixture of special meaning. On locating for large money sums with '\d' meta character /([\d]+)000/, \d will locate for string of numerical character. The list below shows meta characters applied in PERL Style Regular Expressions.

. single character

\s whitespace character (space, tab, newline)

\S	non-whitespace character
\d	digit (0-9)
\D	non-digit
\w	word character (a-z, A-Z, 0-9, _)
\W	non-word character

[aeiou] matches a single character in the given set

[^aeiou] matches a single character outside the given set

Modifiers

There are modifiers present which makes work with regexps easily such as case sensitivity or locating in multiple lines.

i	Makes the match case insensitive
m	Specifies newline/carriage for string characters
o	Evaluates expression only once
s	Allows use of . to match a newline character
x	Allows you to use white space in the expression for clarity
g	Globally finds all matches
cg	Allows a search to continue even after a global match fails

Check your progress 1

1. What are meta characters?
   a. alphabetical character which comes before backslash
   b. alphabetical character which comes after backslash
   c. Special character
   d. None of these
2. What is POSIX?
   a. Portable Operating System Interface for Linux
   b. Portative Operating System Interface for Unix
   c. Portable Operating System Interface for Unix
   d. Portative Operating System Interface for Linux

3.3 Error Handling Regular Expressions

Error handling is the process of catching errors raised by your program and then taking appropriate action. If you would handle errors properly then it may lead to many unforeseen consequences. It's very simple in PHP to handle errors. You can write your own function to handling any error. PHP provides you a framework to define error handling function. This function must be able to handle a minimum of two parameters (error level and error message) but can accept up to five parameters. Its syntax is:

error_function(error_level,error_message, error_file,error_line,error_context);

error_level	Shows error report level for user-defined error in terms of value.
error_message	Shows error message for user-defined error
error_file	Shows file name of error
error_line	Shows line number where error occurred
error_context	Shows array having variable and values in use during error

Using die() function

In a PHP program, check possible error condition before going forward and take correct action as required. Consider an example:

```
<?php
  if(!file_exists("/tmp/test.txt")) {
    die("File not found");
  }else {
    Sfile = fopen("/tmp/test.txt","r");
    print "Opend file sucessfully";
  }
  // Test of the code here.
?>
```

From the above, you will visualise that in such a way you can write good code and with this the program can be stopped with any errors.

Check your progress 2

1. What is the use of error_context parameter?

a. It Specifies the error message for the user-defined error

b. Specifies an array containing every variable, and their values, in use when the error occurred

c. Specifies the filename in which the error occurred

d. None of these

3.4 Exception handling

Exception handling is applied in order to change normal writing of code execution when exceptional conditions take place which is termed as exception. It is seen that when an exception is thrown, the code will not work and with this the PHP tries to locate matching "catch" block. When exception is trapped, in such case fatal error takes place that is taken by Uncaught Exception message. In this, error can be avoided by framing proper code which can take care of exception that will cover:

Try:

A function carrying exception be placed in try block. If exception fails to trigger, the code will continue as normal or on triggering, will be thrown out.

Throw:

This is how you trigger an exception. Each throw must have at least one catch.

Catch:

A catch block retrieves an exception and creates an object containing the exception information.

Once the exception is thrown, code that follows the statement will not be executed, and PHP will attempt to find the first matching catch block. If an exception is not caught, a fatal error will be given with catching the exception. An exception can be thrown, and caught in PHP where code is surrounded in try block. Each try must have at least one corresponding catch block. Multiple catch blocks can be used to catch different classes of exceptions. Exceptions can be thrown (or re-thrown) within a catch block.

Example

```
<?php
  try {
    $error = 'Always throw this error';
    throw new Exception($error);

    // Code following an exception is not executed.
    echo 'Never executed';

  }catch (Exception $e) {
    echo 'Caught exception: ',  $e->getMessage(), "\n";
  }

  // Continue execution
  echo 'Hello World';
?>
```

From the above example, we see that $e->getMessage function is applied to have an error message. Study the functions applied from Exception class.

getMessage() – message of exception

getCode() – code of exception

getFile() – source filename

getLine() – source line

getTrace() – n array of the backtrace()

getTraceAsString() – formated string of trace

Creating Custom Exception Handler

You can define your own custom exception handler by applying user-defined exception handler function as:

string set_exception_handler (callback $exception_handler)

Here exception_handler is the name of the function to be called when an uncaught exception occurs. This function must be defined before calling set_exception_handler().

Example

```
<?php
  function exception_handler(Sexception) {
    echo "Uncaught exception: " , Sexception->getMessage(), "\n";
  }
  set_exception_handler('exception_handler');
  throw new Exception('Uncaught Exception');
  echo "Not Executed\n";
?>
```

Check your progress 3

1. Which version of PHP was incorporated with Exception handling?

a. PHP 4

b. PHP 6

c. PHP 5

d. None of these

2. Which of the following statement is used to call exception class?

a. throw new Exception();

b. throws new Exception();

c. new Exception();

d. None of these

3.5 Let Us Sum Up

MS-DOS and OS/2 use another variation on linked list called FAT. Index allocation addresses many of the problems of contiguous and chained allocation.

C-Scan Scheduling is a type of scheduling, where the processes get arranged by using particular circular order list.

Round Robin is a type of scheduling where the time of CPU is shared into equal numbers which is called as Quantum Time.

3.6 Answers for Check Your Progress

Check your progress 1

Answers: (1 -a), (2 –c)

Check your progress 2

Answers: (1 -b)

Check your progress 3

Answers: (1 -c), (2 –a)

3.7 Glossary

1. **Declaration -** A syntactical element describing name and type of one or more variables, functions, structures, unions, or enumerations.
2. **Object -** Piece of data that which is manipulated by a program using simple variable, array, structure.
3. **Regular expressions -** Strong tool required to change and ammend text.
4. **Quantifiers -** They are frequency bracket character sequences or single characters highlighted by special character.
5. **Error handling -** Getting errors raised by program and applying required action.

3.8 Assignment

Explain Error handling with the help of an example?

3.9 Activities

Study regular exceptions in detail and try to implement.

3.10 Case Study

Create a custom exception handler.

3.11 Further Readings

1. Features of Arrays and Function with Integrated Approach by B. Rahim, 2004.
2. Programming and Principles in PHP, Oxford, 2010, Jerry Smith.

Block Summary

In this block, you have learnt and understand about the sizeof() function and array_unique() functions in PHP. The block gives an idea on array having numeric index and array having name index. You have been well explained with the concepts of application of syntax with respect to Perl style regular expression functions.

The block detailed about the Study about Error Handling Regular Expressions and features of Exception handling. The concept related to Uniqueness sorting array and determining array size will be well explained to students. You will be demonstrated practically about Regular Expression.

Array, function and expression

Block Assignment

Short Answer Questions

1. What is an associative array? Give example using code snippet
2. Explain the String functions?
3. Write a short note on exception handling?
4. Explain user defined functions?

Long Answer Questions

1. Write any program and try to solve error using error handling.
2. Write a program that throws multiple exceptions.
3. Write a program for multidimensional array.

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

WEB APPLICATION DEVELOPMENT

PGDCA 202

BLOCK 2:
OO AND FILE HANDLING IN PHP

Dr. Babasaheb Ambedkar Open University
Ahmedabad

WEB APPLICATION DEVELOPMENT

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Mr. Sanjay Thapar

Language Editor
Mr. K. J. Sharma

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Mr. Prashant Tikone

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included may tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

WEB APPLICATION DEVELOPMENT

Contents

BLOCK 3: OO AND FILE HANDLING IN PHP

UNIT 1 OBJECT ORIENTED PHP

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

UNIT 2 FILE AND DIRECTORY HANDLING

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP

UNIT 1 PHP AND MYSQL DATABASE

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

UNIT 2 STATE MANAGEMENT IN PHP

Session, Cookies

Dr. Babasaheb
Ambedkar
Open University

PGDCA 202

Web Application Development

BLOCK 3: OO AND FILE HANDLING IN PHP

BLOCK 3: OO AND FILE HANDLING IN PHP

Block Introduction

Object-Oriented Programming is a type of programming which can be modular and reusable web applications. It is a methodology which frames an application that could be any sort of web based or windows based. In OOP, all will be around objects and class.

In this block, we will detail about the basic of Object-Oriented Programming and its modelling techniques. The block will focus on declaration of variables, functions, structures, unions, or enumerations with study about their syntax. The concept of structured data type and class in PHP are also explained.

In this block, the student will make to learn and understand about the basic of Comma Separated Values with their usability and necessity in programming the script. The concept related to directory support functions and knowledge related to read and operate directories and entries will also be explained to students. The student will be demonstrated practically about storing tabular data in plain text format.

Block Objective

After learning this block, you will be able to understand:

- The basic of OOP Concepts.
- Features about Creating and Using class properties.
- Basic about Constructors and Destructors Methods.
- Features regarding Cloning and comparing objects.
- Idea about Class Inherit.
- Understanding about Namespaces.
- Idea about Read/Write entire/part of file.
- Features of read and write CSV data.
- Basic of Copy, rename and delete file.
- Idea about File uploading in PHP.

Block Structure

Unit 1: Object Oriented PHP

Unit 2: File and Directory Handling

UNIT 1: OBJECT ORIENTED PHP

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About OOP
- About Constructors and Destructors Methods
- About Creating and Using Object class constant
- About Inspecting an object
- About protected access modifier

1.1 Introduction

Object-Oriented Programming is a type of programming which can be modular and reusable web applications. In PHP, OOP will help the code to remain flexible by defining at many places. It is the latest way of software development where all languages such as Java, PERL, PHP, C#, Ruby use in programming.

1.2 The benefits of OOP

Object oriented programming methodology frames an application which could be any type as web based or windows based applications. In OOP, all will be around objects and class. Use of OOP in PHP will form modular web application and can do activity in object model structure. It carries many benefits.

OOP invokes usage of classes in order to arrange data and structure of an application. It is noted that large programmers avoids the use of OOP as programming theory. It is noted that objects in programming is similar to real word object where all programming object carries properties and behaviours.

Check your progress 1

1. Object oriented programming methodology is:

   a. web based

   b. windows based

   c. both a and b

   d. neither a nor b

1.3 Key OOP Concepts

In terms of programming, object-oriented is hard concept having difficult syntax and roadblocks. Object-oriented programming serves as unique coding style which allows developers to assemble similar work into classes. The concept related to OOP is DRY programming which describes a piece of information when changes in a program, normally required so as to update the programming code. OOP shows threatening to developers since it introduces new syntax and appears to be complex as simple procedural or inline code.

Check your progress 2

1. OOP is:

   a. comfortable for users

   b. simple for user

   c. threatening for user

   d. none of above

1.4 Create and Use class properties

While defining a class in PHP, you can add properties in it. It is hard to imagine that to create native class having regular non-typed public PHP properties. To add data to class, properties or class-specific variables are used which works just like regular variables. In order to add property to MyPlant, write the code as:

```
<?php
class MyPlant
{
  public $prop1 = "Rose is my Plant!";
}
$obj = new MyPlant;
var_dump($obj);
?>
```

In this code, keyword public will describe visibility of property, while property is named by standard variable syntax with certain value. If you have to read such property and give it to browser, then you need to reference the object from place where you read and read as:

```
echo $obj->prop1;
```

As multiple instances of class exist, when individual object is not referenced. In such case, the script is unable to find object which will read. You can alter the script in test.php so as to read the property instead of dumping whole class by simply changing the code as shown:

```
<?php
class MyPlant
{               public $prop1 = "My Plant is Rose!";
}
$obj = new MyPlant;
echo $obj->prop1; // Output the property
?>
```

Output:

My Plant is Rose!

Check your progress 3

1. Which functions shows existence of class?
   a. exist()
   b. exist_class()
   c. class_exist()
   d. __exist()
2. _____keyword describes properties or methods in class itself?
   a. exist
   b. public
   c. protected
   d. $this

1.5 Constructors and Destructors Methods

To start with an object, it is advisable to set few things. To handle this, PHP gives with magic method __construct() that is automatic when an object is created. To have an idea of constructors, you have to add constructor to MyClass which will output message when new instance of class is created:

```
<?php
class MyClass
{
  public $prop1 = "I'm a class property!";
  public function __construct()
  {
     echo 'The class "', __CLASS__, '" was initiated!<br />';
  }
  public function setProperty($newval)
  {
     $this->prop1 = $newval;
  }
  public function getProperty()
  {
     return $this->prop1 . "<br />";
  }
}
// Create a new object
$obj = new MyClass;
// Get the value of $prop1
echo $obj->getProperty();
// Output a message at the end of the file
echo "End of file.<br />";
?>
```

In this, we see that __CLASS__ returns the name of class in which it is called which is magic constant. On reloading file in browser, we have the following result:

The class "MyClass" was initiated!

I'm a class property!

End of file.

To call a function when object is damaged, the __destruct() magic method is available which is good for class cleanup. The message will be outputted when object gets destroyed by explaining magic method as:

__destruct() in MyClass:

```
<?php
class MyClass
{
  public $prop1 = "I'm a class property!";
  public function __construct()
  {
     echo 'The class "', __CLASS__, '" was initiated!<br />';
  }
  public function __destruct()
  {
     echo 'The class "', __CLASS__, '" was destroyed.<br />';
  }
```

```
    public function setProperty($newval)
    {
        $this->prop1 = $newval;
    }
    public function getProperty()
    {
        return $this->prop1 . "<br />";
    }
}
// Create a new object
$obj = new MyClass;
// Get the value of $prop1
echo $obj->getProperty();
// Output a message at the end of the file
echo "End of file.<br />";
?>
```

Output:

The class "MyClass" was initiated!

I'm a class property!

End of file

The class "MyClass" was destroyed.

"When the end of a file is reached, PHP automatically releases all resources."

To explicitly trigger the destructor, you can destroy the object using the function unset():

```
<?php
class MyClass
{
 public $prop1 = "I'm a class property!";
 public function __construct()
 {
    echo 'The class "', __CLASS__, '" was initiated!<br />';
 }
 public function __destruct()
 {
    echo 'The class "', __CLASS__, '" was destroyed.<br />';
 }
 public function setProperty($newval)
 {
    $this->prop1 = $newval;
 }
 public function getProperty()
 {
    return $this->prop1 . "<br />";
 }
}
// Create a new object
$obj = new MyClass;
 // Get the value of $prop1
 echo $obj->getProperty();
 // Destroy the object
 unset($obj);
 // Output a message at the end of the file
 echo "End of file.<br />";
 ?>
```

Output:

```
The class "MyClass"  was initiated!
I'm a class property!
The class "MyClass"  was destroyed.
End of file.
```

Check your progress 4

1. Which statements is true about Constructors?

i PHP has class constructors.

ii Constructors accept parameters.

iii Constructors call class methods.

iv Constructors call other constructors.

a. ii) and iii)

b. All of above

c. i) and iii)

d. ii), iii) and iv)

2. PHP identifies constructors by:

a. classname()

b. _construct()

c. function _construct()

d. function __construct()

1.6 Create and Use Object class constant

You can define constant values on per-class basis keeping it same and unchangeable. Constants differ from normal variables where no use of $ symbol is there in order to declare.

Example #1 Defining and using a constant

```
<?php
class MyClass
```

```
{
    const CONSTANT = 'constant value';
    function showConstant() {
        echo  self::CONSTANT . "\n";
    }
}

echo MyClass::CONSTANT  . "\n";
$classname = "MyClass";
echo $classname::CONSTANT  . "\n"; // As of PHP 5.3.0
$class = new MyClass();
$class->showConstant();
echo $class::CONSTANT."\n";  // As of PHP 5.3.0
?>
```

Example #2 Static data example

```
<?php
class foo {
    // As of PHP 5.3.0
    const BAR = <<<'EOT'
bar
EOT;
    // As of PHP 5.3.0
    const BAZ = <<<EOT
baz
EOT;
}
?>
```

Example #3 Constant expression example

```
<?php
const ONE = 1;
class foo {
   // As of PHP 5.6.0
   const TWO = ONE * 2;
   const THREE = ONE + self::TWO;
   const SENTENCE = 'The value of THREE is '.self::THREE;
}
?>
```

Check your progress 5

1. Creation of objects based on predefined classes is:
   a. class object
   b. object class
   c. object instance
   d. class instantiation
2. Identify correct way to define constant?
   a. constant G = “9.18”;
   b. const $G = “9.18”;
   c. constant G = ‘9.18’;
   d. const G = ‘9.18’;

1.7 Static properties and method

In PHP, static methods and properties is an important feature which is directly accessible without creating object of class. Static properties of class are directly available from class using scope resolution operator. Also, you can make property static using static keyword as shown in static variable in php class:

```
class test
{
public static $a;//Static variable
}
test::$a = 5;
echo test::$a;
```

It is noted that regular property cannot be applied by static way, and if so, will result in fatal error. It is noted that inside a class you can access static property by self keyword. If you are accessing parent class property then, apply parent keyword as shown:

```
class testParent
{
public static $var1;
}
class testChild  extends testParent
{
public static $var2;
public $abc =2;
function testFunction()
{
self::$var2 = 3;
parent::$var1 = 5;
}
}
echo testChild::$abc;  //throw fatal error
```

It is found that static variable or property is the good means to save value of variable in the context of various instances.

Check your progress 6

1. Regular property in static way will lead to:

   a. data error

   b. runtime error

   c. fatal error

   d. all of above

1.8 Loop through an object's properties

PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example a for each statement. By default, all visible properties will be used for the iteration.

Example Simple Object Iteration

```
<?php
class MyClass
{
   public $var1 = 'value 5';
   public $var2 = 'value 6';
   public $var3 = 'value 7';
   protected $protected = 'protected var';
   private   $private   = 'private var';
   function iterateVisible() {
      echo "MyClass::iterateVisible:\n";
      foreach ($this as $key => $value) {
         print "$key => $value\n";
      }
   }
}
$class = new MyClass();
```

```
foreach($class as $key => $value) {
   print "$key => $value\n";
}
echo "\n";
$class->iterateVisible();
?>
```

On running above program:

```
var1 => value 5
var2 => value 6
var3 => value 7
MyClass::iterateVisible:
var1 => value 5
var2 => value 6
var3 => value 7
protected => protected var
private => private var
```

In the output, each iterated properties get accessed where iterator interface gets worked out.

Check your progress 7

1. The output of the program will be:

```
<?php
$colors = array("white","safron","magenta","brown");
foreach ($colors as $value)
{
   echo "$value <br>";
}
?>
```

a)white

safron

magenta

a. brown
b. white
c. brown
d. error

1.9 Clone and compare objects

Having a copy of object carrying replicated properties is not always required. In a copy constructor an object having GTK window will holds the resource of GTK window once duplicate or new window carrying similar properties exists. An object copy is created by using the clone keyword (which calls the object's __clone() method if possible). An object's __clone() method cannot be called directly.

$copy_of_object = clone $object;

Once the object is cloned, PHP will do shallow copy of object's properties which are referenced to other variables as tcremain references. Its syntax is:

void __clone (void)

After cloning is over and once clone() method is defined, then newly created object's __clone() method gets called which allow required properties for change.

Example Cloning an object

```
<?php
class SubObject
{
   static $instances = 0;
   public $instance;

   public function __construct() {
      $this->instance = ++self::$instances;
   }

    public function __clone() {
       $this->instance = ++self::$instances;
    }
}
class MyCloneable
{
   public $object1;
   public $object2;

   function __clone()
   {
      // Force a copy of this->object, otherwise
      // it will point to same object.
      $this->object1 = clone $this->object1;
   }
}
```

```
$obj = new MyCloneable();

$obj->object1 = new SubObject();
$obj->object2 = new SubObject();

$obj2 = clone $obj;

print("Original Object:\n");
print_r($obj);

print("Cloned Object:\n");
print_r($obj2);
?>
```

The above example will output:

Original Object:

```
MyCloneable Object
(
    [object1] => SubObject Object
        (
            [instance] => 1
        )

    [object2] => SubObject Object
        (
            [instance] => 2
        )

)
```

Cloned Object:

```
MyCloneable Object
(
    [object1] => SubObject Object
        (
            [instance] => 3
        )

    [object2] => SubObject Object
        (
            [instance] => 2
        )

)
```

Check your progress 8

1. By cloning:
   a. object itself gets cloned
   b. object is cloned with its associations
   c. both a and b
   d. neither a nor b

1.10 Inspect an object

Inspecting object in good panel or a panel which locates through optional argument panel Name. Its syntax is inspect(object[, panelName]).

Consider an example:

```
class Person
  attr_accessor :name, :age
  def initialize(name, age)
    @name, @age = name, age
```

```
  end
end
bob= Person.new("Sanjay Mathur", 45)
p bob
```

Outputs:

#<Person:0x1a58d8 @age=45, @name="Sanjay Mathur">

Check your progress 9

1.The syntax of object inspection involves:

a. alphabets

b. numbers

c. characters

d. all of these

1.11 Inherit a class

Class inheritance is little complicated as inheritance serves as a connection between a child and its parent. To declare one class inherits code from other class, in such case we have to apply keyword.

```
class Parent {
  // The parent's class code
}
class Child extends Parent {
  // The  child can use the parent's class code
}
```

In the above code we see that child class makes use of non-private methods and a property which tends to inherit from parent class which allows writing code once in parent and can be applied in parent and child classes.

Consider an example:

//The parent class

```
class Vehicle {
  // Private property inside the class
  private $model;
  //Public setter method
  public function setModel($model)
  {
    $this -> model = $model;
  }
  public function hello()
  {
    return "horn! I am a <i>" . $this -> model . "</i><br />";
  }
}
//The child class inherits the code from the parent class
class SportsVehicle extends Vehicle {
  //No code in the child class
}
//Create an instance from the child class
$sportsVehicle1 = new SportsVehicle();
// Set the value of the class' property.
// For this aim, we use a method that we created in the parent
$sportsVehicle1 -> setModel('Maruti');
//Use another method that the child class inherited from the parent class
echo $sportsVehicle1 -> hello();
```

Output:

horn! I am Maruti

Check your progress 10

1. Which class from which child class inherits?

   i) Child class

   ii) Parent class

   iii) Super class

   iv) Base class

   a. (i)

   b. (ii)

   c. (iii)

   d. (ii) and (iv)

1.12 Use the protected access modifier

In PHP, access modifiers are applied to give access rights to PHP classes and their members which define functions and variables in the class.

PHP Access Control Modifiers

Following are the PHP keywords used as access modifiers:

Public:

It is a class or its members having access modifier that are used publicly from anywhere from outside scope of class. It is used where data is available for all public.

Private

The class members with this gets access inside class itself which saves members from outside class access with reference of class instance.

Protected

It is similar as private, since it allows sub classes to access protected super class members.

Abstract

It is applied only for PHP classes and its functions having abstract functions with abstract class.

Final

It saves sub classes which override super class members having final keyword.

Example: Private & Public Access Modifier in PHP

```
<? class Computer {
private $Model = "HP 1150";
private      $HardDrive = 800;
private             $Ram = 34;
public function Specification(){
return $this->Model;
}
}
class User extends Computer{
public $UserName = "Sanjay";
public $UserExperince = "Engineering";
public function UserInformation(){
echo $this->UserName." has a ";
echo $this->Specification();
}
}
$obj = new User;
echo $obj->UserInformation();
?>
```

Example: Protected Access Modifier in PHP

```
<?
class Computer {
protected $Name = "Wipro Satellite Pro";
```

```
public function NameDisplay1(){
echo "Accessing inside the Parent Class: ".$this->Name;
}
}
class Display extends Computer{
public function NameDisplay2(){
echo "<br>Accessing inside the child Class: ".$this->Name;
}
}
$obj = new Computer;
echo $obj->NameDisplay1();
$obj = new Display;
echo $obj->NameDisplay2();
?>
```

Check your progress 11

1. What will be the class relationship shown in the program?

```
class Employee
{
private int empid;
private String ename;
public double getBonus()
{
Accounts acc = new Accounts();
return acc.calculateBonus();
}
}
*
```

```
class Accounts
{
public double calculateBonus(){//method's code}
}
```

a. Aggregation
b. Simple Association
c. Dependency
d. Composition

1.13 Create abstract classes and methods

Abstract class is such class which is not completely implemented that mostly applied as base class for other classes in order to take from complete implementation. It is better in interface as it allows adding common elements of implementation which can be shared by subclasses not simply specifying the interface of subclasses. Normally, we look at creation of abstract class with abstract keyword:

```
<?php
abstract class MyAbstractClass {}
```

Abstract class contains no body, so it is less importance since it fails to define interface for subclasses so as to inherit and serves as legal class.

```
<?php
abstract class MyAbstractClass  {}
$obj = new MyAbstractClass();
/*
```

Output:

Fatal error:

It is noted that an abstract class is partially implemented, so it is fine for it to contain normal methods:

```
<?php
abstract class MyAbstractClass  {
   public function doSomething() {
      echo "MyAbstractClass::doSomething()\n";
   }
}
```

In abstract class, you can declare method as abstract that is not implemented but any non-abstract subclass implements it. The body of abstract method:

```
<?php
abstract class MyAbstractClass  {
   public abstract function doSomething();
}
```

It is noticed that an abstract class need not require having any abstract methods. Also if the class is not declared as abstract, it has no abstract method as shown:

```
<?php
class MyClass {
   public abstract function doSomething();
}
/*
```

Output:

If we run the above program, we see that it will generate fatal error:

Check your progress 12

1. Abstract class:
   a. should contain abstract method
   b. need not contain abstract method
   c. may or may not contain abstract method
   d. all of above

1.14 Create final classes and methods

Final Class

A final class is a class that cannot be extended and can be declared by prefixing 'class' keyword along with 'final'.

Example:

```
final class BaseClass  {
   public function myMethod() {
      echo "BaseClass method called";
   }
}
//this will cause Compile error
class DerivedClass  extends  BaseClass  {
   public function myMethod() {
      echo "DerivedClass method called";
   }
}
$c = new DerivedClass();
$c->myMethod();
```

We see that BaseClass here is declared as final and will not inherit. In this, DerivedClass tries to extend from BaseClass and compiler compile an error.

Final Method

It is a method that cannot be overruled. In order to declare method as final, you need to prefix function name with 'final' keyword as shown in example:

```
class BaseClass  {
   final public function myMethod() {
      echo "BaseClass method called";
   }
}
```

```
class DerivedClass  extends  BaseClass  {
   //this will cause Compile error
   public function myMethod() {
      echo "DerivedClass method called";
   }
}
$c = new DerivedClass();
$c->myMethod();
```

In the above example, DerivedClass extends from BaseClass.

Check your progress 13

1. BaseClass has myMethod() declared as final which can be:
   a. can be ouveruled
   b. cannot be overruled
   c. given fatal error
   d. none of above

1.15 Work with interfaces

Interfaces are abstract classes which has no guts and cannot be worked out as they are designed in OOP allowing qualities in classes. Interface classes are defined by using the keyword interface. All the methods in the interface must be public as this is the nature of an interface.

```
// Declare the interface 'iTemplate'
interface iTemplate
{
   public function setVariable($name, $var);
   public function getHtml($template);
}
```

Example of class using interface:

```
class dog implements sound{
```

```
function sound() {
echo “sa, re, ga, ma …”;
}
/* the interface methods/functions implements in class */
function sound() { echo “piano has sound …”;}
function playing() { echo “guitar is playing …”;}
}
/*
It is seen that when a class implements interface, it should explain all methods or functions of interface else php engine will result in error.
*/
```

Check your progress 14

1. The abstract class carries:
   a. empty concrete method
   b. working concrete method
   c. both a and b
   d. neither a nor b

1.16 Introducing Namespaces

In PHP, there are namespacing which is added to language. There are no two classes having similar name. They are different as if you are using another party library having user as class named, in such case you fail to have your own class. With this, PHP uses namespaces so as to get out of such issue. Consider the simple program describing Simple Namespacing.

```
<?php
namespace Handle;
// app/models/another.php
class add
{
```

```
}
```

In this, add class with small change shows namespace directive. The line namespace Handle will tell PHP about Handle namespace. It says if classes created in such file will be there in 'Handle' namespace. So using Handle class once again.

```
<?php
// app/routes.php
$add = new add();
```

Example:

```
This\Namespace\And\Class\Combination\Is\Silly\But\Works
```

Example to use namespace

```
<?php
namespace app\a{
   class one{
      public static function _1(){
       echo 'a one _1<br>';
      }
   }
}
namespace app\b{
   class one{
      public static function _2(){
         echo 'b one _2<br>';
      }
   }
}
namespace app{
   echo a\one::_1();
   echo b\one::_2();
```

```
    echo a\two::_1();
}
namespace app\a{
    class two{
        public static function _1(){
         echo 'a two _1<br>';
        }
    }
}
```

prints

a one _1

b one _2

a two _1

Check your progress 15

1. In PHP two classes will have:
   a. similar names
   b. no similar names
   c. both clsses will have single alphabet
   d. none of above

1.17 Let Us Sum Up

In this unit we have learnt that Object-Oriented Programming is a type of programming which can be modular and reusable web applications. Object oriented programming methodology frames an application which could be any type as web based or windows based applications. In OOP, all will be around objects and class.

Object-oriented programming serves as unique coding style which allows developers to assemble similar work into classes. While defining a class in PHP,

you can add properties in it. It is hard to imagine that to create native class having regular non-typed public PHP properties.

In PHP, static methods and properties is an important feature which is directly accessible without creating object of class. PHP provides a way for objects to be defined so it is possible to iterate through a list of items, with, for example for each statement. An object copy is created by using the clone keyword (which calls the object's __clone() method if possible). An object's __clone() method cannot be called directly.

1.18 Answers for Check Your Progress

Check your progress 1

Answers: (1-c)

Check your progress 2

Answers: (1-c)

Check your progress 3

Answers: (1-c), (2-d)

Check your progress 4

Answers: (1-b), (2-d)

Check your progress 5

Answers: (1-d), (2-d)

Check your progress 6

Answers: (1-c)

Check your progress 7

Answers: (1-a)

Check your progress 8

Answers: (1-a)

Check your progress 9

Answers: (1-d)

Check your progress 10

Answers: (1-d)

Check your progress 11

Answers: (1-c)

Check your progress 12

Answers: (1-b)

Check your progress 13

Answers: (1-b)

Check your progress 14

Answers: (1-c)

Check your progress 15

Answers: (1 –b)

1.19 Glossary

1. **Public -** It is a property or method which can be worked upon from anywhere on the script.
2. **Private -** It is a method which cannot be accessed from everywhere since used by class or object.
3. **Protected -** It is a property which can be used by code in class which is part of.
4. **Abstract -** It is a property that subclasses and not applied directly.

1.20 Assignment

What is Object-oriented programming?

1.21 Activities

What are Namespace in PHP?

1.22 Case Study

How to create Abstract Class in PHP?

1.23 Further Readings

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22
2. "Using PHP from command line", PHI, Dixit, 2009-09-11

UNIT 2: FILE AND DIRECTORY HANDLING

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About directory listing.
- About Read and write file.
- About Read and write part of file.

2.1 Introduction

PHP carries full set of directory support functions which describes different functions that will read and operate directories and entries in it. With other

directory or file parts, it performs similar functions as with other languages and can do similar work with more simplified versions.

2.2 Get a directory listing

The directory listing in PHP is in shape of Directory Listing Script which can be configured easily and allows a programmer to simply in uploading of files on web directories that turns into formatted, mobile friendly directory browser. It is seen that with the ease of version 4, there appears new features in PHP directories which are:

- Support to mobile browser.
- Uploading of many files with certain restrictions on allowed file-types.
- Gives access to restricted access to script with the application of password or IP Address.
- Supports formation of new directories and sub-directories.
- Able to upload zip files and can able to extract them directly having an option to delete zip file once extracted.
- Will able to hide certain file types, names or extensions and directories.
- Ability to sort file listings by name, size or last modified date.

Single Directory Listing

In order to start with single directory listing, consider a simple function which will return list of files, directories and properties from single directory:

```
<?PHP
 function getFileList($dir)
 {
  // array to hold return value
  $retval = array();
  // add trailing slash if missing
  if(substr($dir, -1) != "/") $dir .= "/";
  // open pointer to directory and read list of files
  $d = @dir($dir) or die("getFileList: Failed opening directory $dir for reading");
  while(false !== ($entry = $d->read())) {
   // skip hidden files
   if($entry[0] == ".") continue;
   if(is_dir("$dir$entry")) {
    $retval[] = array(
"name" => "$dir$entry/",
     "type" => filetype("$dir$entry"),
     "size" => 0,
     "lastmod" => filemtime("$dir$entry")
    );
  } elseif(is_readable("$dir$entry")) {
   $retval[] = array(
    "name" => "$dir$entry",
    "type" => mime_content_type("$dir$entry"),
    "size" => filesize("$dir$entry"),
    "lastmod" => filemtime("$dir$entry")
     );
    }
  }
  $d->close();
  return $retval;
 }
```

?>

You can use this function as follows:

```
<?PHP
  // list files in the current directory
  $dirlist = getFileList(".");
  $dirlist = getFileList("./");
  // a subdirectory of the current directory called images
  $dirlist = getFileList("images");
  $dirlist = getFileList("images/");
  $dirlist = getFileList("./images");
  $dirlist = getFileList("./images/");
  // using an absolute path
  $dirlist = getFileList("{$_SERVER['DOCUMENT_ROOT']}/images");
  $dirlist = getFileList("{$_SERVER['DOCUMENT_ROOT']}/images/");
```



```
?>
```

Example:

```
<?PHP
  $dirlist = getFileList("images");
  echo "<pre>",print_r($dirlist),"</pre>";
/* sample output
  Array
  (
    [0] => Array
       (
```

```
            [name] => images/background0.jpg
            [type] => image/jpeg
            [size] => 86920
            [lastmod] => 1077461701
        )
    [1] => ...
)
*/
?>
```

Check your progress 1

1. In PHP, the directory feature will not:

   a. support mobile browser

   b. have restrictions on uploading file types

   c. give access to restricted access

   d. supports formation of new directories

2.3 Read and write an entire file

In PHP, you can easily read and write full of existing files and can able to delete the file. We will initially start with reading a file as:

```
$myFile = "sampleFile.txt";
$fh = fopen($project, 'r');
$projectContents = fread($fh, 21);
fclose($fh);
echo $projectContents;
```

To do this, you need to initially set the file with file name which could be anything as $project which can be set to "sampleFile.txt" as shown. Using link with $myFileLink variable, set file name and set instruction for the file. Now see for line with carrying contents of file with variable $projectContents. Now applying fread function, and pass variable link as first parameter followed by

number of characters which you want to read. You see that PHP will read the files in the same way as you read.

Just like reading the file, PHP will able to write. With filesize function having file size that shows number of characters present in the file. Using this in fread function will tell PHP to read till the end of the file.

In PHP, writing to file is not hard as compared to reading of file. For this, consider an example shown:

```
$project2 = "testFolder/sampleFile2.txt";
$projectLink2 = fopen($project2, 'w+') or die("Can't open file.");
$newContents = "You wrote on me...";
fwrite($projectLink2, $newContents);
fclose($projectLink2);
```

It is noted that writing to file will vanish it immediately. On using write function you need to run it on some test content before taking all files out. For this, you need to have written permission and if it is not enabled, in such case server will tell PHP that it cannot write to file. On opening file link, we set variable $newContents to hold content that to be written to file. Finally, apply fwrite function in order to write $newContents to $projectLink2.

Check your progress 2

1. In PHP, fread function will instruct to read:
   a. first two lines
   b. last two lines
   c. half of the program
   d. from top to bottom of file

2.4 Read and write part of file

You can read and write part of the file in PHP. The fopen() function in PHP is applied to open a file which needs two arguments that initially shows file name and mode of operation. You can make the change to opened file which will be

closed using fclose() function. The fclose() function needs file pointer as its argument which returns true when closure succeeds or false if it fails.

Reading a file

Once a file is opened using fopen() function it can be read with a function called fread() that makes use of two arguments. These arguments should be file pointer having file length in bytes. It is noted that files length be found using filesize() function that takes file name as argument and returns file size expressed in bytes. Consider the steps shown in reading of file in PHP:

- Open a file using fopen() function.
- Have file's length using filesize() function.
- Read file's content using fread() function.
- Close file with fclose() function.

Consider an example that contains text file to a variable which shows required text on web page.

```
<html>
  <head>
  <title>Reading a file using PHP</title>
</head>
<body>
  <?php
    $filename = "tmp.txt";
    $file = fopen( $filename, "r" );
      if( $file == false ) {
        echo ( "Error in opening file" );
        exit();
 }
 $filesize = filesize( $filename );
 $filetext = fread( $file, $filesize );
 fclose( $file );
 echo ( "File size : $filesize bytes" );
 echo ( "<pre>$filetext</pre>" );
```

```
    ?>
  </body>
</html>
```

It will produce the following result:

File size : 278 bytes

Writing a file

In order to write text in existing file with PHP fwrite() function, it is seen that this function uses two arguments that specifies file pointer and string of data which is to be written. You can also include third integer argument to specify length of data to write. The impact of third argument is that, it will stop writing after reaching to particular length. Consider an example showing text insertion in file by writing short text heading inside it. After closing this file its existence is confirmed using file_exist() function which takes file name as an argument

```
<?php
  $filename = "/home/user/guest/newfile.txt";
  $file = fopen( $filename, "w" );
  if( $file == false ) {
    echo ( "Error in opening new file" );
    exit();
  }
  fwrite( $file, "This is a simple test\n" );
  fclose( $file );
?>
<html>
  <head>
    <title>Writing a file using PHP</title>
  </head>
  <body>
```

```
<?php
  $filename = "newfile.txt";
  $file = fopen( $filename, "r" );
  if( $file == false ) {
    echo ( "Error in opening file" );
    exit();
  }
$filesize = filesize( $filename );
$filetext = fread( $file, $filesize );
fclose( $file );
echo ( "File size : $filesize bytes" );
echo ( "$filetext" );
echo("file name: $filename");
?>
 </body>
</html>
```

It will produce the following result:

Error in opening new file

Check your progress 3

1. The PHP fopen() function uses _____arguments to open a file.
   a. one
   b. two
   c. three
   d. four
2. The fclose() function needs _____as its argument.
   a. arrays
   b. strings
   c. file pointer
   d. none of above

2.5 Read and write CSV data

CSV data in PHP is Comma Separated Values that are file that stores tabular data in plain text form with sequence of characters having no data that has to be interpreted as binary numbers. It carries any number of records, with line breaks having fields, separated by character or string with literal comma or tab. In this, the records have identical sequence of fields. The structure of CSV file:

one, two

example1, example2

data1, data2

test1, test2

You can read from a CSV file using php function "fgetcsv" which reads content of csv file as shown:

```
<?php
function readCSV($csvFile){
 $file_handle = fopen($csvFile, 'r');
 while (!feof($file_handle) ) {
  $line_of_text[] = fgetcsv($file_handle, 1024);
 }
 fclose($file_handle);
 return $line_of_text;
}

 // Set path to CSV file
 $csvFile = 'test_data.csv';
 //calling the function
 $csv = readCSV($csvFile);
 if(!empty($csv)){
    foreach($csv as $file){
        //inserting into database
```

```
        $query_insert = "insert into csv_data_upload set
           name   =  '".$file[0]."',
           value  =  '".$file[1]."'";
           echo $query_insert;
        $insert = mysql_query($query_insert);
    }
  }else{
     echo 'Csv is empty';
}
?>
```

Once the code is executed, data gets stored in database.

To write into CSV File we require "fputcsv" which will format a line as csv and write it to particular file handle.

Example

```
<?php
$list = array (
   array('dd', 'mm', 'yy','),
   array('12', '12', '2016'),
   array('"dd"', '"mm"')
);
$fp = fopen('file.csv', 'w');
foreach ($list as $fields) {
   fputcsv($fp, $fields);
}
fclose($fp);
?>
```

Following code will write into CSV File.

Check your progress 4

1. In PHP, fputcsv fuinction helps:

   a. To read CSV

   b. To write CSV file

   c. To print CSV file

   d. None of above

2.6 Copy, rename and delete file

PHP copying a file

Simply, to copy a file, use copy() function which initially requires file type to copy that passes file path to first parameter of copy() function. Apart from this, you need to show file path so as to copy file. It is seen that copy() function will return true when file gets copied completely, else will returns false. In the following program, there are specific codes written which shows how to copy the test.txt file to test2.txt file.

```
<?php
        $fn = './test.txt';
        $newfn = './test2.txt';

        if(copy($fn,$newfn))
                echo 'The file was copied successfully';
        else
                echo 'An error occurred during copying the file';
<?php
 $fn = './test.txt';
 $newfn = './test2.txt';

 if(copy($fn,$newfn))
 echo 'The file was copied successfully';
 else
```

```
echo 'An error occurred during copying the file';
```

PHP renaming a file

You can rename a file using rename() function which makes you to move a file from one directory to different directories. Consider the program where renaming of test.txt file to test.bak file exists with code:

```
<?php
$fn = './test.txt';
$newfn = './test.bak';
if(rename($fn,$newfn)){
        echo sprintf("%s was renamed to %s",$fn,$newfn);
}else{
        echo 'An error occurred during renaming the file';
 }
 <?php
 $fn = './test.txt';
 $newfn = './test.bak';
 if(rename($fn,$newfn)){
  echo sprintf("%s was renamed to %s",$fn,$newfn);
}else{
 echo 'An error occurred during renaming the file';
}
```

We see that rename() function will return true when file gets successfully renamed, else returns false. In the following program, it describes how to apply rename() function so as to move test.bak to backup directory:

```
<?php
$fn = './test.bak';
$newfn = './backup/test.bak';
if(rename($fn,$newfn)){
        echo sprintf("%s was moved to %s",$fn,$newfn);
}else{
        echo 'An error occurred during moving the file';
```

```
}
<?php
$fn = './test.bak';
$newfn = './backup/test.bak';
if(rename($fn,$newfn)){
 echo sprintf("%s was moved to %s",$fn,$newfn);
}else{
 echo 'An error occurred during moving the file';
}
```

PHP deleting a file

Similarly you can delete a file using unlink() function where on applying the particular file will result in deletion using unlink() function. Consider the program showing use of unlink() function in file deletion.

```
<?php
$fn = './backup/test.bak';
if(unlink($fn)){
    echo sprintf("The file %s deleted successfully",$fn);
}else{
    echo sprintf("An error occurred deleting the file %s",$fn);
}
<?php
$fn = './backup/test.bak';
if(unlink($fn)){
 echo sprintf("The file %s deleted successfully",$fn);
}else{
 echo sprintf("An error occurred deleting the file %s",$fn);
}
```

We see that copy(), rename() and unlink() functions raise warning-level errors if the file cannot be found therefore it is good practice to check the file exists using the file_exists() function before copying, renaming or deleting it.

Check your progress 5

1. The unlink() functions is used in:

a. read of file

b. writing of file

c. deleting of file

d. none of above

2.7 File uploading

It is seen that normally files get uploaded in temporary directory which further will send across targeted destination with the help of PHP script. Information in phpinfo.php page will show temporary directory which is required for file uploading using upload_tmp_dir with max allowed size of files as upload_max_filesize. Such parameter gets set in PHP configuration file php.ini.

The process of uploading requires following steps −

- On opening HTML form having text files along with browse button and submit button.
- User on clicking browse button and choosing uploading of file from local PC.
- Complete path along with selected file will appear in text filed which the user will click on submit button.
- On pressing submit, selected file gets sent to temporary directory on server.
- The PHP script specified as form handler in form's action will check for arriving of file which further copies file in intended directory.
- The PHP script further will make sure about success to user.

Creating Upload File Script

The "upload.php" file contains the code for uploading a file:

```
<?php
$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES["fileToUpload"]["name"]);
$uploadOk = 1;
$imageFileType = pathinfo($target_file,PATHINFO_EXTENSION);
// Check if image file is a actual image or fake image
if(isset($_POST["submit"])) {
    $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
    if($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File is not an image.";
        $uploadOk = 0;
    }
}
?>
```

In this, we see that:

$target_dir = "uploads/": This will specify directory about the place of file

$target_file: It shows path of file which to be uploaded

$imageFileType: It will keep file extension of file

Check your progress 6

1. The correct form of uploading files in PHP is:
   a. upload.php
   b. upl.php
   c. uploa.php
   d. upload.php/

2.8 Let Us Sum Up

In this unit we have learnt that PHP carries full set of directory support functions which describes different functions that will read and operate directories and entries in it. In PHP, fopen() function is used to open a file. It requires two arguments stating first the file name and then mode in which to operate. After making a changes to the opened file it is important to close it with the fclose() function.

CSV data in PHP is Comma Separated Values that are file that stores tabular data in plain text form with sequence of characters having no data that has to be interpreted as binary numbers.

2.9 Answers for Check Your Progress

Check your progress 1

Answers: (1-b)

Check your progress 2

Answers: (1-d)

Check your progress 3

Answers: (1-b), (2-c)

Check your progress 4

Answers: (1-b)

Check your progress 5

Answers: (1-c)

Check your progress 6

Answers: (1-a)

2.10 Glossary

1. **Class -** HTML elements can have one or more classes, separated by spaces. You can style elements using CSS by selecting them with their classes.
2. **Usage -** Almost all content belongs inside the body tag. The main exceptions are script and style tags, as well as the page title tag.
3. **Div -** A block level container for content with no semantic meaning.

2.11 Assignment

Explain the CSV data in PHP.

2.12 Activities

Write a program stated the application of PHP fopen() function.

2.13 Case Study

Compile and run the program in PHP which will upload files from computer?

2.14 Further Readings

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

Block Summary

In this block, you will understand about the basic of Use the protected access modifier with knowledge about creating abstract classes and methods. The block gives an idea on creating final classes and methods along with concept of Interface of class are well detailed. The examples related to concept of Namespaces are also discussed.

In this block, you will understand about the basic of Copy, rename and deleting file along with its techniques. The concept related to reading and writing complete or part of file in PHP is well detailed. The student will be demonstrated practically about Comma Separated Values.

Block Assignment

Short Answer Questions

1. What is directory listing in PHp?
2. What is object's __clone() method?
3. What are the advantages and drawbacks of directories in PHP?
4. What is declaration of object in PHP?
5. Explain attribute with example?

Long Answer Questions

1. Explain the steps involved in reading and writing part of file in PHp?
2. What are the features of CSV data files in PHP?
3. Explain Copy, rename and delete file in PHP?

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

WEB APPLICATION DEVELOPMENT

PGDCA 202

BLOCK 2:
DATABASE AND STATE MANAGEMENT IN PHP

Dr. Babasaheb Ambedkar Open University
Ahmedabad

WEB APPLICATION DEVELOPMENT

Knowledge Management and
Research Organization
Pune

Editorial Panel

Author
Mr. Sanjay Thapar

Language Editor
Mr. K. J. Sharma

Graphic and Creative Panel
Ms. K. Jamdal
Ms. Lata Dawange
Mr. Prashant Tikone

Acknowledgment
Every attempt has been made to trace the copyright holders of material reproduced in this book. Should an infringement have occurred, we apologize for the same and will be pleased to make necessary correction/amendment in future edition of this book.
The content is developed by taking reference of online and print publications that are mentioned in Bibliography. The content developed represents the breadth of research excellence in this multidisciplinary academic field. Some of the information, illustrations and examples are taken "as is" and as available in the references mentioned in Bibliography for academic purpose and better understanding by learner.'

ROLE OF SELF INSTRUCTIONAL MATERIAL IN DISTANCE LEARNING

The need to plan effective instruction is imperative for a successful distance teaching repertoire. This is due to the fact that the instructional designer, the tutor, the author (s) and the student are often separated by distance and may never meet in person. This is an increasingly common scenario in distance education instruction. As much as possible, teaching by distance should stimulate the student's intellectual involvement and contain all the necessary learning instructional activities that are capable of guiding the student through the course objectives. Therefore, the course / self-instructional material are completely equipped with everything that the syllabus prescribes.

To ensure effective instruction, a number of instructional design ideas are used and these help students to acquire knowledge, intellectual skills, motor skills and necessary attitudinal changes. In this respect, students' assessment and course evaluation are incorporated in the text.

The nature of instructional activities used in distance education self-instructional materials depends on the domain of learning that they reinforce in the text, that is, the cognitive, psychomotor and affective. These are further interpreted in the acquisition of knowledge, intellectual skills and motor skills. Students may be encouraged to gain, apply and communicate (orally or in writing) the knowledge acquired. Intellectual-skills objectives may be met by designing instructions that make use of students' prior knowledge and experiences in the discourse as the foundation on which newly acquired knowledge is built.

The provision of exercises in the form of assignments, projects and tutorial feedback is necessary. Instructional activities that teach motor skills need to be graphically demonstrated and the correct practices provided during tutorials. Instructional activities for inculcating change in attitude and behavior should create interest and demonstrate need and benefits gained by adopting the required change. Information on the adoption and procedures for practice of new attitudes may then be introduced.

Teaching and learning at a distance eliminates interactive communication cues, such as pauses, intonation and gestures, associated with the face-to-face method of teaching. This is particularly so with the exclusive use of print media. Instructional activities built into the instructional repertoire provide this missing interaction between the student and the teacher. Therefore, the use of instructional activities to affect better distance teaching is not optional, but mandatory.

Our team of successful writers and authors has tried to reduce this.

Divide and to bring this Self Instructional Material as the best teaching and communication tool. Instructional activities are varied in order to assess the different facets of the domains of learning.

Distance education teaching repertoire involves extensive use of self-instructional materials, be they print or otherwise. These materials are designed to achieve certain pre-determined learning outcomes, namely goals and objectives that are contained in an instructional plan. Since the teaching process is affected over a distance, there is need to ensure that students actively participate in their learning by performing specific tasks that help them to understand the relevant concepts. Therefore, a set of exercises is built into the teaching repertoire in order to link what students and tutors do in the framework of the course outline. These could be in the form of students' assignments, a research project or a science practical exercise. Examples of instructional activities in distance education are too numerous to list. Instructional activities, when used in this context, help to motivate students, guide and measure students' performance (continuous assessment)

PREFACE

We have put in lots of hard work to make this book as user-friendly as possible, but we have not sacrificed quality. Experts were involved in preparing the materials. However, concepts are explained in easy language for you. We have included may tables and examples for easy understanding.

We sincerely hope this book will help you in every way you expect.

All the best for your studies from our team!

WEB APPLICATION DEVELOPMENT

Contents

BLOCK 3: OO AND FILE HANDLING IN PHP

UNIT 1 OBJECT ORIENTED PHP

The benefits of OOP, Key OOP Concepts, Create and Use class, properties, Constructors and Destructors, Methods, Create and Use Object, class constant, static properties and method, loop through an object's properties, clone and compare objects, inspect an object, inherit a class, use the protected access modifier, create abstract classes and methods, create final classes and methods, work with interfaces, Introducing Namespaces

UNIT 2 FILE AND DIRECTORY HANDLING

Get a directory listing, read and write an entire file, read and write part of a file, read and write CSV data, copy, rename and delete a file, file uploading

---

BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP

UNIT 1 PHP AND MYSQL DATABASE

Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension), database connection, select data, insert, update and delete data in PHP using MySQL

UNIT 2 STATE MANAGEMENT IN PHP

Session, Cookies

Dr. Babasaheb
Ambedkar
Open University

PGDCA 202

Web Application Development

BLOCK 3: DATABASE AND STATE MANAGEMENT IN PHP

BLOCK 4: DATABASE AND STATE MANAGEMENT IN PHP

Block Introduction

PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie. It is noted that cookie serves as piece of information which is send to server which further sends it to Web browser. In case, a web browser enters to subsequent page within similar domain, it will show server access with similar cookie which is quiet easy to implement. It is found that cookies are restricted to how information gets stored with lots of security problems. Apart from cookies, another state management technique in PHP is session. In PHP, session will keep all data on server. With this, sessions get more versatile which helps in securing cookies. Among the entire biggest potential problem related to sessions is that it depends on session ID which serves as an identifier which points to every unique set of session data. Such session ID gets access on every page through passing it from page to page or by keeping it in cookie.

In this block, we will detail about the basic of database software as Oracle or Sybase with respect to MYSQL database. The block will focus on MySQL extension and MySQL systems versions 4.1.3 with study about their characteristics. The concept of installation of host computer with WAMP, MAMP or XAMPP is also explained.

In this block, you will make to learn and understand about the basic of session cookies and their related techniques. The concept related to Database connectivity with MySQL and characteristics working of script in PHP using MySQL are also explained to you. You will be demonstrated practically about cookies in local host computer.

Block Objective

After learning this block, you will be able to understand:

- About ways to use PHP in MySQL.
- Basic of Database connectivity with MySQL.
- Feature of selecting data.

- Characteristics about working in PHP using MySQL.
- Concept of Session.
- Idea about Cookies.

Block Structure

Unit 1: PHP and MySQL Database

Unit 2: State Management in PHP

UNIT 1: PHP AND MYSQL DATABASE

Unit Structure

1.0 Learning Objectives

After learning this unit, you will be able to understand:

- About MySQL
- About Database connection

1.1 Introduction

It is seen that PHP will work with the help of database software which can be Oracle or Sybase which commonly present in MySQL database. To run any website using dynamic information, you need to have a database so that the information can be kept inside it. There are many types of things you can perform while interacting with PHP and MySQL. To run any PHP script, you only need three items that can be accessed by MySQL databases. Initially, you require web

server which can be a computer or can be web host along with web server software compatible with PHP and MySQL.

1.2 Three ways to use PHP to work with MySQL (PDO, mysqli extension, MySQL extension)

PHP will able to handle MySQL with the application of its extension. Many projects compose and written across the globe using PHP and MySQL. In a spell of time, PHP depreciate MySQL extension and was removed with alternatives as MySQLi which is improved MySQL version and PDO which is PHP Data Objects. There are three main API options when considering connecting to MySQL database server:

- PHP's MySQL Extension
- PHP's mysqli Extension
- PHP Data Objects (PDO)

It is found that MySQLi and PDO are object oriented and support Prepared Statements which can be support Transactions, Stored Procedures and many. The Prepared Statements are important for use of web application security since they can save from SQL injection. With the application of Prepared Statements, there will be no possibility escape strings before insertion in Database. It is found that PDO supports many databases apart from MySQL.

PHP's MySQL Extension

PHP MySQL Extension is an original extension which is created to help in development of PHP applications which interacts with MySQL database. In this, mysql extension will show a process interface which is planned for applied only with MySQL versions. It can also be applied with MySQL 4.1.3 or newer versions, with restriction on latest MySQL server features. It is found that mysql extension source code is present in PHP extension directory of ext/mysql.

PHP's mysqli Extension

PHP mysqli extension is an improved extension of MySQL which was created in order to make use of latest features of available in MySQL systems versions 4.1.3 and above that. This extension is covered with PHP versions 5 and later. The mysqli extension carries many advantages over key mysql extension as:

- Having Object-oriented interface

- Supporting for Prepared Statements
- Supporting for Multiple Statements
- Supporting for Transactions
- Advanced debugging capabilities
- Advanced server help

The mysqli extension is created with the help of PHP extension framework having its source code present in the directory ext/mysqli.

Example:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

PDO

PDO is PHP Data Objects which is type of database abstraction layer particularly designed for PHP applications. It provides steady API for certain PHP application immaterial of database type server that connects to it. Normally it is found that when applying PDO API, selection of database is upto you which can be from Firebird to MySQL with small alteration in order to apply in PHP code.

With certain advantages related to simplicity, easy, portable API, POD has certain drawbacks as it fails to allow in using several advanced features which is present in current MySQL server version. PDO is carried out with the help of PHP extension framework where its source code available in directory ext/pdo.

Example

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";


try {
   $conn = new PDO("mysql:host=$servername;dbname=myDB", $username, $password);
   // set the PDO error mode to exception
   $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
   echo "Connected successfully";
   }
catch(PDOException $e)
   {
   echo "Connection failed: " .$e->getMessage();
   }
?>
```

Check your progress 1

1. Advance MySQL extension released with PHP 5 is:
   a. MySQL
   b. mysql
   c. mysqli
   d. mysqil
2. Which version carries PHP mysqlnd:
   a. PHP 5.0
   b. PHP 4.1
   c. PHP 4.2
   d. PHP 5.3

1.3 Database connection

In order to work in PHP and MySQL, you need to initially establish a connection to MySQL database which is an important part as without script interconnection with database will fail all queries related to database.

It is good way to use database is initially to apply username, password and should give name to database at the starting of script code which can be altered anytime with no effort.

```
$username="your_username";$password="your_password";$database="your_dat
abase";
```

In the above code, you should change "your_username", "your_password" and "your_database" with required MySQL username, password and database which can be used with your script.

Here, you will not worried about security risk of keeping your password in a file as PHP source code gets processed by server before it will; send data to browser which will help in safe guarding from visitors. Further, you need to connect PHP script to database using mysql_connect PHP function which is:

```
mysql_connect(localhost,$username,$password);
```

The above script will direct the PHP to connect to MySQL database server at localhost. Once the connection is made, then choose the database which you want to use where your username had access which can be done by using the following command:

```
@mysql_select_db($database) or die( "Unable to select database");
```

This command will tell PHP so as to select database which is kept in variable $database or in "your_database". If the script fails to connect in such case it will stop executing and displays an error message as Unable to select database. Apart from these, there is another feature of PHP function which is:

```
mysql_close();
```

The above function is important since it closes the connection to database server. This script is required in order to stop the script from running. It is noted that many open MySQL connections will generate problem in the account. So it is better to close MySQL connection after all queries are done. After this, you can connect to server and select database which you wish to work with and start querying about it.

So it is noted that in order to run database connection, you have to initially create the database and add database user along with MySQL user before running the script on server. Many times you require connecting PHP website to database which can be easily performed with content management systems using config.php file as shown in example below:

```
<?php
//Sample Database Connection Script
//Setup connection variables, such as database username
//and password
$hostname="localhost";
$username="your_dbusername";
$password="your_dbpassword";
$dbname="your_dbusername";
$usertable="your_tablename";
$yourfield = "your_field";

//Connect to the database
$connection = mysql_connect($hostname, $username, $password);
mysql_select_db($dbname, $connection);

//Setup our query
$query = "SELECT * FROM $usertable";

//Run the Query
$result = mysql_query($query);

//If the query returned results, loop through
// each result
if($result)
{
```

```
 while($row = mysql_fetch_array($result))
 {
  $name = $row["$yourfield"];
  echo "Name: " . $name;

 }
}

?>
```

In the above PHP script which is connected to database showing relevant fields for particular table is specified in the code above. To run database script on local computer, set the computer with Apache, MySQL, and PHP which can be easily done by installing WAMP, MAMP or XAMPP.

Check your progress 2

1. The function which diagnose and display information about MySQL connection error is:
   a. connect_errno()
   b. connect_error()
   c. mysqli_connect_errno()
   d. mysqli_connect_error()

1.4 Select data

Till now we have seen creating and starting of database. Once the data is inserted into database, you can easily retrieve it. In SQL, you have to made a query with the help of SELECT statement which you have to execute by passing it to mysqli_query() function so as to retrieve data from database or data table. The basic syntax of SELECT query is:

SELECT column_name(s) FROM table_name

With SQL query using SELECT statement, now you will execute SQL query by passing it to mysqli_query() function in order to select tables records. For this you consider the data table showing details of persons:

```
+-----------+------------+-----------+---------------------+
| person_id | first_name | last_name | email_address       |
+-----------+------------+-----------+---------------------+
|         1 | Sanjay     | Mathur    | Sanjaymathur@gmail.com |
|         2 | Amit       | Charan    | amitcharan@gmail.com  |
|         3 | Renu       | Kapoor    | renukapoor@gmail.com  |
|         4 | Rajiv      | Sood      | rajivsood@gmail.com   |
|         5 | Amkrish    | Bajaj     | amrishbajaj@gmail.com |
+-----------+------------+-----------+---------------------+
```

We see that the PHP code in above example selects all data which is stored in persons table.

```
<?php
/* Attempt MySQL server connection. Assuming you are running MySQL
server with default setting (user 'root' with no password) */
$link = mysqli_connect("localhost", "root", "", "demo");

// Check connection
if($link === false){
    die("ERROR: Could not connect. " . mysqli_connect_error());
}
```

```
// Attempt select query execution
$sql = "SELECT * FROM persons";
if($result = mysqli_query($link, $sql)){
    if(mysqli_num_rows($result) > 0){
        echo "<table>";
            echo "<tr>";
                echo "<th>person_id</th>";
                echo "<th>first_name</th>";
                echo "<th>last_name</th>";
                echo "<th>email_address</th>";
            echo "</tr>";
        while($row = mysqli_fetch_array($result)){
            echo "<tr>";
                echo "<td>" . $row['person_id'] . "</td>";
                echo "<td>" . $row['first_name'] . "</td>";
                echo "<td>" . $row['last_name'] . "</td>";
                echo "<td>" . $row['email_address'] . "</td>";
            echo "</tr>";
        }
        echo "</table>";
        // Close result set
        mysqli_free_result($result);
    } else{
        echo "No records matching your query were found.";
    }
} else{
    echo "ERROR: Could not able to execute $sql. " . mysqli_error($link);
}

// Close connection
mysqli_close($link);
?>
```

From the above example we see that, data returned by mysqli_query() function gets stored in $result variable. It is seen that every time mysqli_fetch_array() is called upon and returns next record from result set array. In this, the while loop is used to loops all the records in result set. Lastly, value of individual fields gets accessed from record either by passing field index or field name to $row variable such as:

- $row['person_id'] or $row[0]
- $row['first_name'] or $row[1]
- $row['last_name'] or $row[2]
- $row['email_address'] or $row[3]

Check your progress 3

1. The compound datatype is:

   a. Array

   b. String

   c. Float

   d. Boolean

1.5 Insert, update and delete data in PHP using MySQL

Delete data:

DELETE statement is applied in order to delete records from table. The syntax is:

DELETE FROM table_name

WHERE some_column = some_value

We see that WHERE clause in DELETE syntax describes which record to be deleted. On removing the WHERE clause, you will see that all records gets deleted. Consider "Students" table:

id	firstname	lastname	email	reg_date
1	Sanjay	Mathur	Sanjay@gmail.com	2014-11-21 14:26:15
2	Monica	Goel	monica@gmail.com	2014-11-22 10:22:30
3	Naine	Malik	naine@gmail.com	2014-11-25 10:48:23

In the above example we see that the record of student with id=3 gets deleted in "Students" table:

Example:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
// sql to delete a record
$sql = "DELETE FROM Students WHERE id=3";
if ($conn->query($sql) === TRUE) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . $conn->error;
}
$conn->close();
?>
```

Example: MySQLi Procedural

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
```

```
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
// sql to delete a record
$sql = "DELETE FROM Students WHERE id=3";
if (mysqli_query($conn, $sql)) {
    echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Example (PDO)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
```

```
try {
   $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
   // set the PDO error mode to exception
   $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
   // sql to delete a record
   $sql = "DELETE FROM Students WHERE id=3";
   // use exec() because no results are returned
   $conn->exec($sql);
   echo "Record deleted successfully";
   }
catch(PDOException $e)
   {
   echo $sql . "<br>" . $e->getMessage();
   }
$conn = null;
?>
```

After the record is deleted, the table will look like this:

id	firstname	lastname	email	reg_date
1	Sanjay	Mathur	Sanjay@gmail.com	2014-11-21 14:26:15
2	Monica	Goel	monica@gmail.com	2014-11-22 10:22:30

Update Data:

To update data in MySQL Table with the help of MySQLi and PDO, consider an UPDATE statement which will update present records in the table. Consider the synex:

UPDATE table_name

SET column1=value, column2=value2,...

WHERE some_column=some_value

In the above UPDATE syntax, WHERE clause is applied in UPDATE syntax in order to describe which record to be updated. To erase WHERE clause, you will find that all records get updated. For this, consider Students table:

id	firstname	lastname	email	reg_date
1	Sanjay	Mathur	Sanjay@gmail.com	2014-11-21 14:26:15
2	Monica	Goel	monica@gmail.com	2014-11-22 10:22:30

id	firstname	lastname	email	reg_date
1	John	Doe	john@example.com	2014-10-22 14:26:15
2	Mary	Moe	mary@example.com	2014-10-23 10:22:30

We see from the this is that the record with id=2 in students table get adjusted.

Example (MySQLi Object-oriented)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$sql = "UPDATE Students SET lastname='Doe' WHERE id=2";
if ($conn->query($sql) === TRUE) {
    echo "Record updated successfully";
} else
{
```

```
    echo "Error updating record: " . $conn->error;
}

$conn->close();
?>
```

Example (MySQLi Procedural)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

$sql = "UPDATE Students SET lastname='Doe' WHERE id=2";

if (mysqli_query($conn, $sql)) {
    echo "Record updated successfully";
} else {
    echo "Error updating record: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

Example (PDO)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
try {
   $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
   // set the PDO error mode to exception
   $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
   $sql = "UPDATE Students SET lastname='Doe' WHERE id=2";
   // Prepare statement
   $stmt = $conn->prepare($sql);
   // execute the query
   $stmt->execute();
      // echo a message to say the UPDATE succeeded
      echo $stmt->rowCount() . " records UPDATED successfully";
      }
   catch(PDOException $e)
      {
      echo $sql . "<br>" . $e->getMessage();
      }
   $conn = null;
   ?>
```

Once the record is updated, the table will look like:

id	firstname	lastname	email	reg_date
1	Sanjay	Mathur	Sanjay@gmail.com	2014-11-21 14:26:15
2	Monica	Goel	monica@gmail.com	2014-11-22 10:22:30

Insert Data:

Once the database and table is framed, now we can insert data by following certain syntax rules:

- Query in SQL should be quoted in PHP
- String values in SQL query should be quoted
- Numeric values be excluded from quoted
- NULL word should not be quoted

We see that INSERT INTO statement is applied in order to add new records to MySQL table:

INSERT INTO table_name (column1, column2, column3,...)

VALUES (value1, value2, value3,...)

The examples given below will add new record to Students table:

Example (MySQLi Object-oriented)

```
<?php
$servername = "localhost"
$username = "username";
$password = "password";
$dbname = "myDB";


// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
```

```
$sql = "INSERT INTO Students (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if ($conn->query($sql) === TRUE) {
    echo "New record created successfully";
} else
```

```
{
    echo "Error: " . $sql . "<br>" . $conn->error;
}
$conn->close();
?>
```

Example (MySQLi Procedural)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
```

```
$sql = "INSERT INTO Students (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com')";
if (mysqli_query($conn, $sql)) {
   echo "New record created successfully";
} else {
   echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}
mysqli_close($conn);
?>
```

Example (PDO)

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";
try {
   $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
   // set the PDO error mode to exception
   $conn->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
   $sql = "INSERT INTO Students (firstname, lastname, email)
   VALUES ('Nishit', 'Mathur', nishit@gmail.com')";
   // use exec() because no results are returned
   $conn->exec($sql);
   echo "New record created successfully";
   }
```

```
catch(PDOException $e)
   {
   echo $sql . "<br>" . $e->getMessage();
   }
$conn = null;
?>
```

Check your progress 4

1. In SQL statement, inserting new data is done by:
   a. INSERT
   b. UPDATE
   c. ADD
   d. INSERT NEW
2. Which commands creates database named “school”:
   a. CREATE ?I school
   b. CREATE DATABASE school
   c. DATABASE /school
   d. DATABSE school

1.6 Let Us Sum Up

In this unit we have learnt that PHP will work using database software that can be Oracle or Sybase which is available in MySQL database. It seen that PHP will handle MySQL with application of its extension as certain projects carried and written across globe using PHP and MySQL.

It is found that PHP depreciate MySQL extension and was removed with alternatives as MySQLi which is improved MySQL version and PDO which is PHP Data Objects. The PHP mysqli extension is improved extension of MySQL created to make use of latest features of available in MySQL systems. PHP Data Objects is type of database abstraction layer designed for PHP applications that

gives steady API for certain PHP application immaterial of database type server that connects to it.

It is noted that to run database script on local computer, you need to set computer with Apache, MySQL, and PHP by installing WAMP, MAMP or XAMPP.

1.7 Answers for Check Your Progress

Check your progress 1

Answers: (1-c), (2-d)

Check your progress 2

Answers: (1-c)

Check your progress 3

Answers: (1-a)

Check your progress 3

Answers: (1-a), (2-b)

1.8 Glossary

1. **MySQL -** It is a database server applied in PHP.
2. **localhost -** It can be a computer at home or in office used to run particular web pages for websites.
3. **Database -** It is a collection of information that get stored in server and can be taken on calling it.

1.9 Assignment

What is an improved extension of MySQL?

1.10 Activities

Create an activity on PHP Data Objects.

1.11 Case Study

Does your institute carries database designed in PHP? Comment

1.12 Further Readings

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

UNIT 2: STATE MANAGEMENT IN PHP

Unit Structure

2.0 Learning Objectives

After learning this unit, you will be able to understand:

- About Session.
- About Cookies.

2.1 Introduction

In PHP, there are certain sessions that take care and keep record of genuine users. It serves as building block for websites with many user activities. In absence of sessions, everyone would be an unknown visitor. With fast evolution of Web programming, stateless nature of protocol brings problems to Web applications which maintain state across protocol requests. With this the role of session management emerges as straight response to certain problem with present method being applied by PHP developers in context to Web application requirement to track state during occurrence of various HTTP requests.

2.2 Session

In PHP, session is a way by which you can keep your information that is applied across multiple pages. Similar to cookie, in session, the information is not stored on user's computer. On working with application and opening it for certain alterations further can be closed. It is same as Session where computer will able to identify you. It knows when you start an application and when you close. In case of web there appears one problem that web server is unable to find you and your activities as protocol address fails to maintain the state.

It is noted that session variables will able to solve problem with storage of user information that can be applied across various pages. By default, session variables last till user closes the browser. Hence, session variables keep the information of single user that is present to all pages in single application. We see that session starts with session_start() function where session variables are set using PHP global variable: $_SESSION.

Consider a new page "project_session1.php" where we start with new PHP session and set session variables as:

```
<?php
// Session Start
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Set session variables
$_SESSION["favpicture"] = "junun";
$_SESSION["favplant"] = "rose";
echo "Session variables are set.";
?>
</body>
</html>
```

Output:

Session variables are set

Creating PHP session:

Now we will see another page "project1_session2.php" where from we can access session information set on first page ("project_session1.php"). We see that session variables are not passed singly to every new page rather than are recovered from session what gets open at start of every page (session_start()). Further, all session variable values are kept in global $_SESSION variable as shown in the code:

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Echo session variables that were set on previous page
echo "Favorite picture is " . $_SESSION["favpicture"] . ".<br>";
echo "Favorite plant is " . $_SESSION["favplant"] . ".";
?>
</body>
</html>
```

Output:

Favorite picture is junun.

Favorite plant is rose.

Changing PHP Session Variable:

You can change the session variable simply by overwriting it as shown in the code below:

```
<?php
session_start();
```

```
?>
<!DOCTYPE html>
<html>
<body>
<?php
// to change session variable, overwrite it
$_SESSION["favpicture"] = "junun";
print_r($_SESSION);
?>
</body>
</html>
```

Output:

On running the above code, we will get the output as:

Array ([favpicture] => junun [favplant] => rose)

Check your progress 1

1. Session data can be kept in _____ways.
   a. 2
   b. 4
   c. 3
   d. 5
2. The default PHP session name is:
   a. PHPSESSID
   b. PHPSESID
   c. PHPSESSIONID
   d. PHPIDSESSION

3. _____function is used in starting of a session.

a. start_session()

b. session_start()

c. session_begin()

d. start__begin,session()

2.3 Cookies

In PHP, cookies also is used to find a user which is a type of small file where server embossed on user's computer. In this, every time, similar computer will requests for page having a browser which sends the cookies. PHP helps in creating and retrieving cookie values. A cookie is created by setcookie() function. The Syntax of cookies is:

setcookie(name, value, expire, path, domain, secure, httponly);

In this only name parameter are used while others are optional.

Create/Retrieve a Cookie

You can create and retrieve cookies in PHP. The example below creates cookie as "user" having value "Sanjay Mathur" that expires after 30 days. Here "/" shows that cookie is present in full website. Then the value of cookie "user" is retrieved using isset() function to find when cookie is set as:

```
<?php
$cookie_name = "user";
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/"); //
?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
   echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
```

```
    echo "Cookie '" . $cookie_name . "' is set!<br>";
    echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
</body>
</html>
```

Output:

Cookie named 'user' is not set!

We see that here we will reload the page in order to see the value of cookie which automatically encoded while sending cookies which further will decoded when received.

Modify a Cookie Value

In PHP, you can modify a cookie simply by setting cookie by setcookie() function as shown in following code:

```
<?php
$cookie_name = "user";
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/");
?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
    echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
    echo "Cookie '" . $cookie_name . "' present<br>";
    echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
```

```
</body>
</html>
```

Output:

Cookie 'user' present

Value is: Sanjay Mathur

From this, we see that we can reload the page anytime to see new value of cookie.

Delete a Cookie

Further in PHP, you can easily delete a cookie with setcookie() function having expiration date in past:

```
<?php
// set the expiration date
setcookie("user", "", time() - 36);
?>
<html>
<body>
<?php
echo "Cookie 'user' is deleted.";
?>
</body>
</html>
```

Output:

Cookie 'user' is deleted.

Check your progress 2

1. Which among the following environment variable is used in accessing cookies:
   a. HTTP_COOKIE_VARS
   b. COOKIE_HTTP_VARS
   c. VARS_COOKIE_HTTP
   d. COOKIE_VARS_HTTP

2. The correct declare Session syntax of cookies is:

a. $_mark[‘username’]=’SESSION';

b. $_SESSION[‘username’]=’mark';

c. $_SESSION[‘declrusername’]=’mark';

a. d. $_mark[‘usernamedeclr’]=’SESSION';

2.4 Let Us Sum Up

In this unit we have learnt that there are certain sessions in PHP which takes care and keep record of genuine users and serves building block for websites with many user activities. In PHP, session is a way by which you can keep your information that is applied across multiple pages which is same as cookie where information is not stored on user’s computer.

It is noted that in PHP, cookies are used to find user that serves as small file where server is embossed on user's computer.

2.5 Answers for Check Your Progress

Check your progress 1

Answers: (1 -c), (2 –d), (3 -d), (4 –a)

Check your progress 2

Answers: (1 -d), (2 -d)

Check your progress 3

Answers: (1 –a), (2 –d), (3 –a), (4 –d)

2.6 Glossary

1. **Cookies -** It is a program that keeps information and is stored in user's browser memory.
2. **Session -** It is a logical object that will save data across subsequent http requests.

2.7 Assignment

Explain the advantages of Cookies.

2.8 Activities

Write the program which describes session in PHP.

2.9 Case Study

Compile and run the program which shows deletion of Cookies.

2.10 Further Readings

1. "PHP: Basic syntax", PHI, Robin Smith. 2008-02-22.
2. "Using PHP from command line", PHI, Dixit, 2009-09-11.

Block Summary

In this block, you will understand about the basic of PHP Data Objects with concept related to database abstraction layer. The block gives an idea on ways to use PHP in MySQL with study about their characteristics. The examples related to concept of selecting data with their working characteristics are also discussed.

In this block, you will understand about the basic of handling MySQL with application of its extension. The concept related to inserting, updating and deleting of data in PHP using MySQL is also detailed. You will be demonstrated practically about data selection and its coding.

Block Assignment

Short Answer Questions

1. What is session cookie?
2. What do you mean by database in MySQL?
3. What is the function of fopen() in PHP?
4. Highlight predefined classes in PHP?
5. Explain database abstraction layer?

Long Answer Questions

1. How can we destroy a session in PHP?
2. How do you define a constant?
3. What are the different errors in PHP?

Enrolment No. ☐

1. How many hours did you need for studying the units?

Unit No	**1**	**2**	**3**	**4**
Nos of Hrs				

2. Please give your reactions to the following items based on your reading of the block:

Items	Excellent	Very Good	Good	Poor	Give specific example if any
Presentation Quality	☐	☐	☐	☐	_____
Language and Style	☐	☐	☐	☐	_____
Illustration used (Diagram, tables etc)	☐	☐	☐	☐	_____
Conceptual Clarity	☐	☐	☐	☐	_____
Check your progress Quest	☐	☐	☐	☐	_____
Feed back to CYP Question	☐	☐	☐	☐	_____

3. Any Other Comments

……

……

……

……

……

……

……

……

……..

Education is something which ought to be brought within the reach of every one.

- Dr. B. R. Ambedkar

Dr. Babasaheb Ambedkar Open University
Jyotirmay' Parisar, Opp. Shri Balaji Temple, Sarkhej-Gandhinagar Highway, Chharodi,
Ahmedabad-382 481.

Post Graduate Diploma in Computer Application

Lab Manual

For

Web Application Development

Course Code: PGDCA – 202

Dr. Babasaheb Ambedkar Open University

Gujarat

Contents

22	**Lab 17: Write a PHP program to select data and show into table format.**
23	**Lab 18: Update the data present in MYSQL database using web form**
24	**Lab 19 : To create PHP session**
25	**Lab 20 :Write a program to change session variable**
26	**Lab 21: Write a program to create ,modify and deleting cookies.**

Introduction of Web Application Development

Web application or services are applicable in variety of ways as it serves business logic components which can be connected across the exchange data to do meaningful work. PHP is Hypertext Pre-processor a programming language which is applied by web developers in order to frame dynamic content which is easy to interact with databases. Many web services make use of Extensible Markup Language (XML) in order to define format of request and response messages. This feature is tagged structure which shows required flexibility for changing of information that exists among disparate components. In PHP, there exists strong set of efficient ways that will help in dealing with such arrays. Arrays are basically applied in order to store and organize data efficiently and quickly within no time. It serves as an important data a type which is present in any programming language as it is easily described as ordered list of elements.

PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie. It is noted that cookie serves as piece of information which is send to server which further sends it to Web browser. In case, a web browser enters to subsequent page within similar domain, it will show server access with similar cookie which is quiet easy to implement. It is found that cookies are restricted to how information gets stored with lots of security problems. Apart from cookies, another state management technique in PHP is session.

Importance

- Web applications are built on client/server architecture where business logic is present in the application that works on web server and uses HTTP in order to communicate with clients over Internet.
- PHP is an important coding technique which can be easily embedded directly in HTML.
- You will make to learn and understand about the basic of PHP in built functions with their features.
- PHP carries various state management features which will overcome certain problems which are taken from stateless Web which can be cookie.
- You will make to learn and understand about the basic of session cookies and their related techniques.

Objective of Lab Manual

After completion of this lab, learners will:

- Learn and understand about basic of outputting data to web browser and its techniques.
- Knowledge related to PHP Supported Data types
- Learn basic of Identifiers, Variables, Constants and Expressions
- Learn the concept of Control Structures.
- Learn basic of Arrays with their features associated with storing and organizing of data with required efficiency.
- Learn basic of database connectivity with MySql.
- Learn features of selecting data.
- Learn to create, modify and delete cookies.

Introduction of LAB

There are 50 systems installed in computer Lab. Their configurations are as follows:

Processor	:	Pentium IV 2.4 GHz
RAM	:	512 MB
Hard Disk	:	40 GB
Mouse	:	Optical Mouse
Operating System	:	Windows XP (or latest version)
Software	:	PHP, MySql
Network Interface card	:	Present

Guidelines Related To Lab

Guidelines to learners:

- Equipment in the lab for the use of learner's community. Learners need to maintain a proper decorum in the computer lab.
- Learners must use the equipment with care. Any damage is caused is punishable.
- Learners are required to carry their observation / programs book with completed exercises while entering the lab.
- Learners are supposed to occupy the machines allotted to them and are not supposed to talk or make noise in the lab. The allocation is put up on the lab notice board.
- Lab can be used in lab time decided by lab-in charge.
- Lab records need to be submitted on or before date of submission.
- Learners are not supposed to use any USB or other devices.
- Use of computer network is encouraged.

Practicals

&

Their solutions

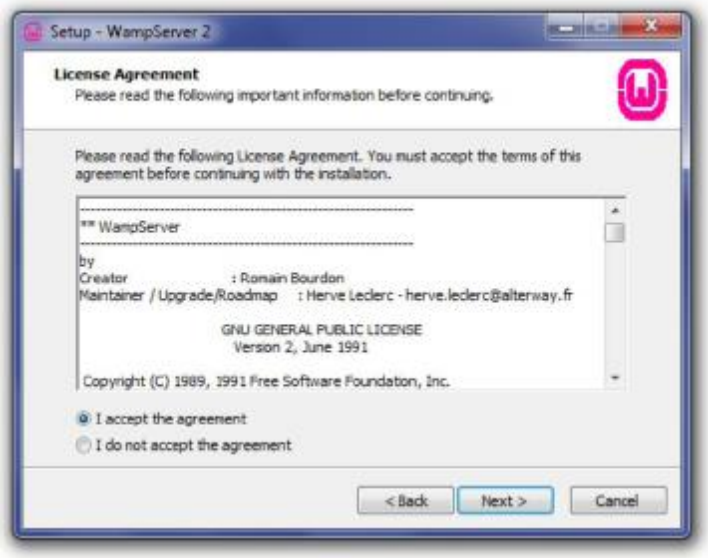
Lab 1: To Install PHP web Application.

Solution:

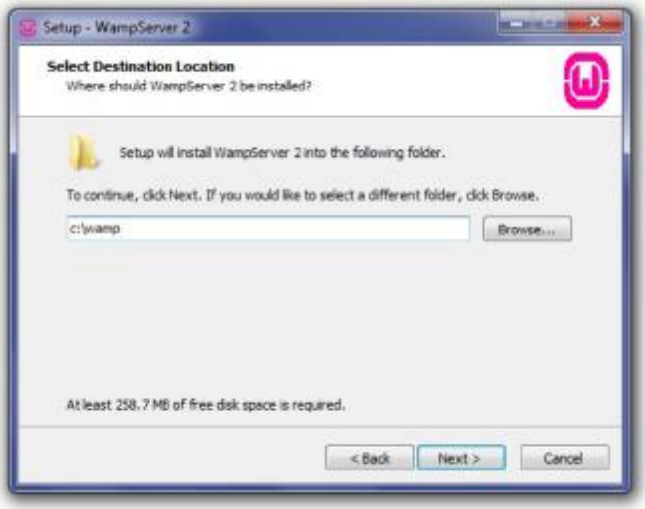
WAMP allows you to move between different versions of PHP if it is installed properly. There are certain steps to be followed while installing Wamp Server.

To start installing, you have to open the folder where you have saved your file and double click on installer file. After double clicking, a security warning window will open which will ask you whether to run this file or not. If you select on Run you will find that the installation process started and you will see Welcome To WampServer Setup Wizard screen on which you have choose next to forward with the installation.

After this, the License Agreement screen will appear which will ask you to accept all agreements by selecting the radio button. Selecting the radio button and clicking on next will allow moving to next screen.

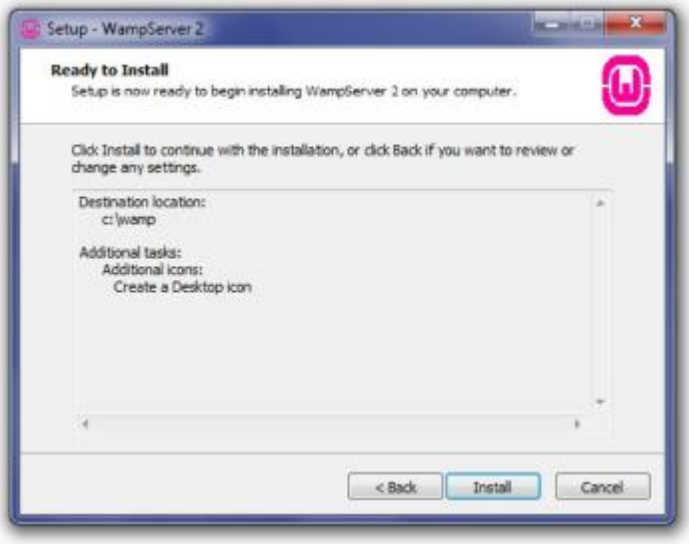


This screen will ask for the location where you have to select Destination Location. Unless you would like to install WampServer on another drive, you should not need to change anything. Click Next to continue.

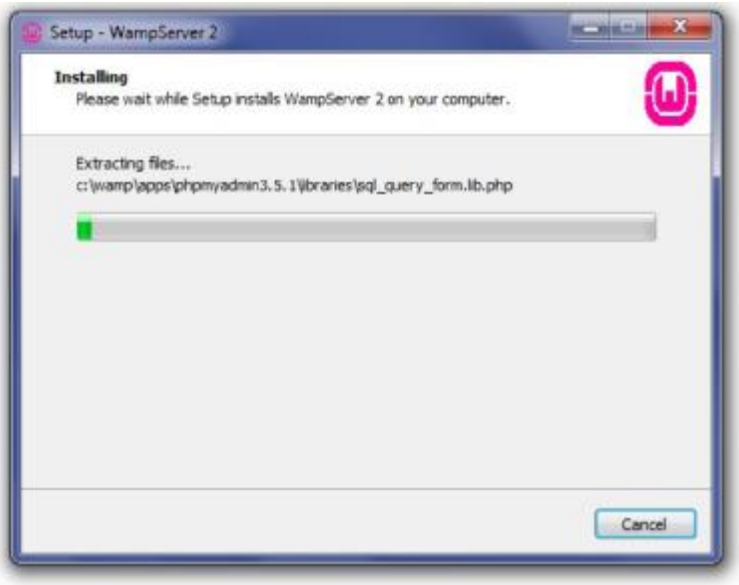


After this, the next screen shows Select Additional Tasks screen where you need to select whether you would like Quick Launch icon to be added to taskbar or Desktop icon after the installation. After making the selections, click Next to continue.

Clicking on Next will take you to Ready To Install screen where you can review your setup choices and change any by clicking Back to required screen. Once you have reviewed your choices, click Install to continue.



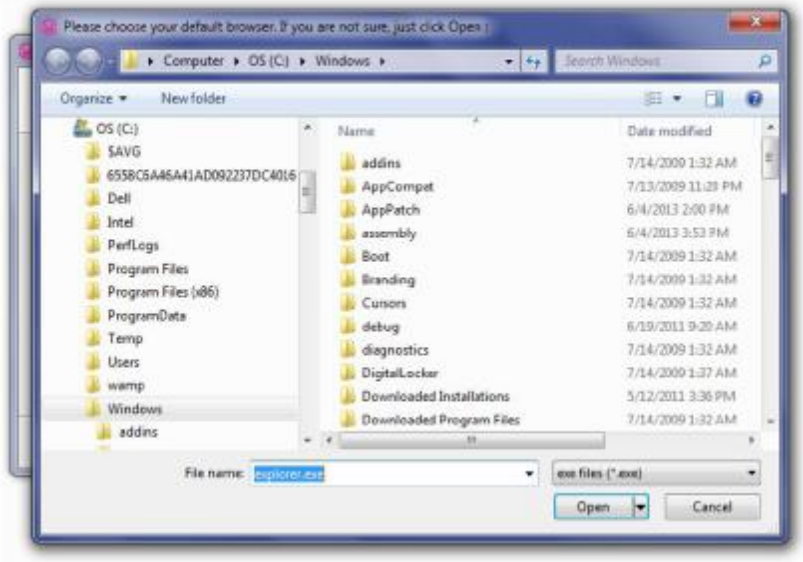
After this, the WampServer will start extracting files to required selected location.



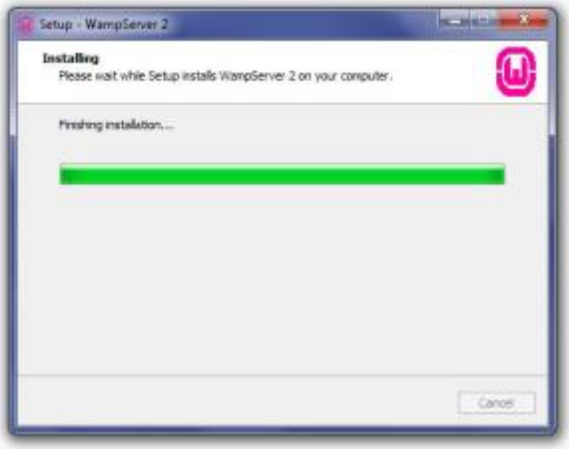
After extraction of files, you will be asked to select default browser. Wamp Server defaults to Internet Explorer upon opening the local file browser window. If your default browser is not an Internet Exsplorer, then find required .exe file:

- Opera: C:\Program Files (x86)\Opera\opera.exe
- Firefox: C:\Program Files (x86)\Mozilla Firefox\firefox.exe
- Safari: C:\Program Files (x86)\Safari\safari.exe
- Chrome:
  C:\Users\xxxxx\AppData\Local\Google\Chrome\Application\chrome.exe

Now select default browser's .exe file and click Open to continue.



On clicking you will find that a Windows Security Alert window will appear prompting that Windows Firewall has blocked some features of program. Here you need to check whether you want to allow Apache HTTP Server to communicate on private or public network and if this and then click Allow Access. You will see that the setup screen will appear showing status of installation process.



You will that the progress bar will work and once it is completely green, then PHP Mail Parameters screen will appear. Here you have to leave SMTP server local host and change email address to one of your selection. Click Next to continue.

After this you will find that the Installation complete screen will appear where you have to select Launch WampServer Now and click on Finish to complete installation.

Finally you will find WampServer icon in systray on right side of taskbar.

Lab 2: To Create a php webpage and print "hello world".

```
<Html>
<Head>
<Title> My Simple Program </Title>
</head>
<Body>
   <?php
      echo "Hello World"
   ?>
</Body>
</Html>
```

Output:

Hello World

Lab 3: To create a php program to find odd or even number from given number.

```
<Html>
<Head>
<Title> Find out odd or even number. . ! </title>
</Head >
<body>
   <?php
 $a = 10;
 if ($a % 2==0)
          {
  echo "Given number is" . "<br>" . "<b>EVEN<b>" ;
          }
 else
  {
  echo "Given number is" . "<br>" . "<b>ODD<b>";
  }
   ?>
</body> </html>
```

Output:

Given number is

EVEN

Lab 4: To write a PHP program to swap two numbers.

```
<html>
<head>
<title>
 Swapping of two numbers. . !
</title>
</head>
<Body>
    <?php
 $a = 10;
 $b = 20;
  echo "a = $a"."<br>"."b = $b"."<br>";
$a = $a + $b;
 $b = $a - $b;
 $a = $a - $b;
  echo "<b>After Swapping"."<br>"." a = $a"."<br>"."b = $b<b>";
?>
</body>
</html>
```

Output:

a =10

b =20

After Swapping

a =20

b =10

Lab 5: Give PHP Example to calculate the area of the circle

Solution:

```
<html>
    <body>
     <?php
    // defining constant value PI = 3.14
     define("PI","3.14");
     $radius=15;
    $area=PI*$radius*$radius;
    echo "Area=".$area;
     ?>
    </body>
 </html>
```

Output

Area=706.5

Lab 6: To declare multiple variables in for loop.

Solution:

```
<html>
  <body>
   <?php

    for ($x=0,$y=1,$z=2;$x<=3;$x++) {
     echo "x = $x,  y = $y,  z = $z <br>";
    }
   ?>
  </body>
</html>
```

Output:

x = 0, y = 1, z = 2 x = 1, y = 1, z = 2 x = 2, y = 1, z = 2 x = 3, y = 1, z = 2

Lab 7: To declare a user defined function.

Solution:

```
<html>
  <body>
  <?php

   // Function definition
   function myFunction()
   {
    echo "Hello world";
   }

   // Function call
   myFunction();
  ?>
</body>
</html>
```

Output:

Hello world

Lab 8: Example of Numeric Array.

Solution:

```
<html>
  <body>
   <?php
   /* An array $flower_shop is created with three
   Values - rose, daisy,orchid */
    $flower_shop = array (
       "rose",
       "daisy",
       "orchid"
    );
   echo "Flowers: ".$flower_shop[0].",
     ".$flower_shop[1].", ".$flower_shop[2]."";
   ?>
  </body>
</html>
```

Output:

Flowers: rose, daisy, orchid

Lab 9: Give the example of multiple dimensional array.

Solution:

```
<html>
 <body>
  <?php
   $flower_shop = array(
    "rose" => array( "5.00", "7 items", "red" ),
    "daisy" => array( "4.00", "3 items", "blue" ),
    "orchid" => array( "2.00", "1 item", "white" ),
   );
/* in the array $flower_shop['rose'][0], ‘rose’ indicates row and  ‘0’ indicates column */
   echo "rose costs ".$flower_shop['rose'][0].
      ", and you  get ".$flower_shop['rose'][1].".<br>";

   echo "daisy costs ".$flower_shop['daisy'][0].
    ", and you get ".$flower_shop['daisy'][1].".<br>";

   echo "orchid costs ".$flower_shop['orchid'][0].
    ", and you get ".$flower_shop['orchid'][1].".<br>";
  ?>
 </body>
</html>
```

Output:

rose costs 5.00, and you get 7 items. daisy costs 4.00, and you get 3 items. orchid costs 2.00, and you get 1 item.

Lab 10: Give the example of string function: substr():

Solution:

```
<html>
<head>
<title> String Function: Substr():1 </title>
</head>

<body>
  <b>If the start is non-negative, the returned string will start at the start'th position in string, start from 0.</b><br/><br/>
   <?php
 echo "Substring with positive start: “substr("abcdef",2)."<br/>";?><br/>
  <b>If the start is negative, the returned string will start at the start'th character in string, from the end of the string.</b><br/><br/>
   <?php
 echo "Substring with negative start: “substr("abcdef",-2)."<br/>";?><br/>
  <b>If the start is less than or equal to start characters long, false will return</b><br/><br/>
   <?php
 echo "start is <= string”. Substr("abcdef",7)."<br/><br/>";
 echo "Finish";
    ?>
</body>
</html>
```

O/P:

If the start is non-negative, the returned string will start at the start'th position in string, start from 0. Substring with positive start:cdef **If the start is negative, the returned string will start at the start'th character in string, from the end of the string.** Substring with negative start:ef **If the start is less than or equal to start characters long, false will return** start is <= string Finish

Lab 11: Give the example of string function: strcmp()

```
<html>
<head> <title> String Function:strcmp() </title> </head>
<body>
   <?php
 $str1 = 'a';
 $str2 = 'b';
 echo strcmp($str1, $str2)."<br/>";
 $str3 = 'b';
 $str4 = 'a';
 echo strcmp($str3, $str4)."<br/>";
 $str5 = "Anil";
 $str6 = "anil";
 echo strcmp($str5, $str6)."<br/>";
        ?>
</body></html>
```

O/P:

-1 1 -1

Lab 12: Write a PHP program to create a database using MySQL.

Solution:

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
  <?php
 $con = mysql_connect("localhost","root","");
 if(!$con)
  {
     die("not opened");
 }
 echo "Connection open"."</br>";

      $query = "create database std";
 $crdb = mysql_query($query,$con);
 if(!$crdb)
 {
   die("not created. .!".mysql_error());
 }
 echo "database created.. !";

   ?>
</body>
</html>
```

O/P:

Connection open database created.. !

Lab 13: Write a program to use namespace

Solution:

```
<html>
<body>
<?php
namespace app\a{
class one{
public static function _1(){
echo 'a one _1<br>';
}
}
}
namespace app\b{
class one{
public static function _2(){
echo 'b one _2<br>';
}
}
}
namespace app{
echo a\one::_1();
echo b\one::_2();
echo a\two::_1();
}
namespace app\a{
class two{
public static function _1(){
echo 'a two _1<br>';
}
}
}
</body>
</html>
```

Output:

a one _1

b one _2

a two _1

Lab 14: Write a PHP program to create a table in MySQL.

Solution:

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
  <?php
$con = mysql_connect("localhost","root","");
if(!$con)
  {
     die("not opened");
 }
echo "Connection open"."</br>";

$db = mysql_select_db("studinfo",$con);
if(!$db)
 {
   die("Database not found".mysql_error());
 }
echo "Database is selected"."</br>";

      $query = "create table computer(id INT not null,name varchar(50),branch varchar(50))";
$crtb = mysql_query($query,$con);
if(!$crtb)
 {
   die(" table not created. .!".mysql_error());
```

```
 }
echo "table created.. !"."</br>";


  ?>
</body>
</html>
```

Output:

Connection open Database is selected

table created.. !

Lab 15: Write a PHP program to insert record into a table using MySQL.

Solution

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
  <?php
$con = mysql_connect("localhost","root","");
if(!$con)
  {
     die("not opened");
 }
echo "Connection open"."</br>";

$db = mysql_select_db("studinfo",$con);
if(!$db)
 {
   die("Database not found".mysql_error());
 }
echo "Database is selected"."</br>";
```

```
$query = "insert into computer values(7009,'Anil J Basantani','Sadhana colony Jamnagar')";

$insrtb = mysql_query($query,$con);
if(!$insrtb)
 {
   die("Record not inserted.".mysql_error());
 }
echo "Record inserted successfully. . .!"."</br>";
   ?>
</body>
</html>
```

Output:

Connection open Database is selected Record inserted successfully. . .!

Lab 16: Write a PHP program to drop table using MySQL.

Solution:

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
  <?php
 $con = mysql_connect("localhost","root","");
 if(!$con)
  {
     die("not opened");
 }
 echo "Connection open"."</br>";

 $db = mysql_select_db("studinfo",$con);
 if(!$db)
 {
   die("Database not found".mysql_error());
 }
 echo "Database is selected"."</br>";

      $query = "drop table ce";
 $crtb = mysql_query($query,$con);
 if(!$crtb)
 {
   die(" table not droped. .!".mysql_error());
 }
 echo "table droped.. !"."</br>";

 ?>
</body>
</html>
```

Output:

Connection open Database is selected

table droped.. !

Lab 17: Write a PHP program to select data and show into table format.

Solution:

```
<html>
<head>
<title>Create Database. </title>
</head>
<body>
  <?php
$con = mysql_connect("localhost","root","");
if(!$con)
  {
     die("not opened");
 }
echo "Connection open"."</br>";

$db = mysql_select_db("studinfo",$con);
if(!$db)
 {
   die("Database not found".mysql_error());
 }
echo "Database is selected"."</br>";

      $query = "select * from computer";
$sldt = mysql_query($query,$con);
if(!$sldt)
 {
   die("data not selected".mysql_error());
 }
echo "<table border='1'>
```

```
 <tr>
 <th>ID</th>
 <th>Name</th>
 <th>Branch</th>
</tr>";
 while($row = mysql_fetch_array($sldt))
 {
 echo "<tr>";
 echo "<td>".$row['id']."</td>";
 echo "<td>".$row['name']."</td>";
 echo "<td>".$row['branch']."</td>";
 echo "</tr>";
 }
echo "</table>";

  ?>
</body>
</html>
```

Output:

Connection open Database is selected

ID	Name	Branch
9	Anil J Basantani	CE
9	Anil J Basantani	CE
9	Anil J Basantani	CE

Lab 18: Update the data present in MYSQL database using web form

Solution:

```
<html>
 <body>
  <?php
    // Open MYSQL server connection
    $db = mysql_connect("localhost", "root","q1w2e3r4/");
    // Select the database using MYSQL server connection
    mysql_select_db("mydb",$db);
    if ($id) {
      if ($submit) {
     // Write UPDATE query and assign to $sql Variable
        $sql = "UPDATE employees SET
          first='$first', last='$last',
     address='$address',
               position='$position'
     WHERE id=$id";
      // Execute the query
   $result = mysql_query($sql);
         echo "Thank you! Information updated.";
}
else
{

     // Write query to SELECT data from table
     $sql = "SELECT * FROM employees WHERE id=$id";
     // Execute the query
     $result = mysql_query($sql);
     // Fetch the values
     $myrow = mysql_fetch_array($result);
 ?>
<form method="post" action="<?php echo $PHP_SELF?>">
   <input type=hidden name="id" value="<?php echo
       $myrow["id"] ?>">
```

```
First name:<input type="Text" name="first"
        value="<?php echo $myrow["first"] ?>"><br>
Last name:<input type="Text" name="last"
        value="<?php echo $myrow["last"] ?>"><br>
Address:<input type="Text" name="address"
        value="<?php echo $myrow["address"]?>"><br>
Position:<input type="Text" name="position"
        value="<?php echo $myrow["position"]?>"><br>
<input type="Submit" name="submit" value="Enter
        information">
</form>
  <?php
   }
  }
  else
  {
   // display list of employees
   $result = mysql_query("SELECT * FROM
                    employees",$db);
   while ($myrow = mysql_fetch_array($result)) {
      printf("<a href=\"%s?id=%s\">%s %s</a><br>",
         $PHP_SELF, $myrow["id"],$myrow["first"],
         $myrow["last"]);
   }
  }
 ?>
 </body>
</html>
```

Lab 19 : To create PHP session

Solution:

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// Echo session variables that were set on previous page
echo "Favorite picture is " . $_SESSION["favpicture"] . ".<br>";
echo "Favorite plant is " . $_SESSION["favplant"] . ".";
?>
</body>
</html>
```

Output:

Favorite picture is junun.

Favorite plant is rose.

Lab 20 :Write a program to change session variable

Solution:

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// to change session variable, overwrite it
$_SESSION["favpicture"] = "junun";
print_r($_SESSION);
?>
```

```
</body>
</html>
```

Output:

On running the above code, we will get the output as:

Array ([favpicture] => junun [favplant] => rose)

Lab 21: Write a program to create ,modify and deleting cookies.

Solution:

Create cookies

```
<?php
$cookie_name = "user";
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/"); //
?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
echo "Cookie '" . $cookie_name . "' is set!<br>";
echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
</body>
</html>
```

Output:

Cookie named 'user' is not set!

Modify cookies

```
<?php
$cookie_name = "user";
```

```
$cookie_value = "Sanjay Mathur";
setcookie($cookie_name, $cookie_value, time() + (30), "/");
?>
<html>
<body>
<?php
if(!isset($_COOKIE[$cookie_name])) {
echo "Cookie named '" . $cookie_name . "' is not set!";
} else {
echo "Cookie '" . $cookie_name . "' present<br>";
echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
</body>
</html>
```

Output:

Cookie 'user' present

Value is: Sanjay Mathur

Delete a cookie:

```
<?php
// set the expiration date
setcookie("user", "", time() - 36);
?>
<html>
<body>
<?php
echo "Cookie 'user' is deleted.";
?>
</body>
</html>
```

Output:

Cookie 'user' is deleted.