A Workbook on PHP with MySQL

Table of Contents

3-	**Operators** • Arithmetic, Assignment, Comparison, Logical Operators, Concatenation, Bitwise, Error Suppression, Increment & Decrement operators. **Control Structures** • Execution Control Statement, Conditional Statements, Looping statements with Real Time Examples

4-	**Functions** • Creating Functions • Types of Functions • Passing Arguments by Value • Passing Arguments by Reference • Some pre-defined functions **Include Mechanism** • Include(), Include_once() • Require(), Require_once()
5-	**Arrays** • What is an Array. • Types of arrays • How to create an array • Array Functions **Date and Time Functions** **Pre-Defined Functions** **Strings** • Introduction to string • Creating and Working with String. • Introduction of String Function. • Working with string functions.
6-	**Object Oriented Programming in PHP** • Classes, Objects
7-	Fields, Properties, Constants, Method Encapsulation

8-	Inheritance and Types Polymorphism Constructor and Destructor Static Members and Instance Member, Helper Function
9-	**Error and Exception** • Handling Error, Logging Configuration, Directives PHP's, Exception ClassThrow New Exception, Custom Exceptions
10-	**Working with Files and Operating System** • File Streams (Input, Output, Temp) • Creating and deleting a file • Reading and writing test file • Checking for existence of file • Determining file size • Opening a file for writing, reading or appending
	• Writing data to the file • Reading characters
11-	**Session** • Introduction to Session • Starting a Session • Working with Session variables • Destroying session
12-	**File Uploading** • Upload a file from a form • Make a target Folder • Store the file into targeted folder • File Downloading
13-	**Introduction to MY SQL** • PHP MySQL Introduction • PHP MySQL Connect • PHP MySQL Create Database • PHP MySQL Create Table • PHP MySQL Insert • PHP MySQL Prepared

14-	PHP MySQL Last Inserted ID PHP MySQL Select PHP MySQL Where PHP MySQL Limit PHP MySQL Order By PHP MySQL Update PHP MySQL Delete PHP MySQL CRUD Application
15-	SMS API Integration Google Map Integration Live Chatting Payment Gateway

PHP Lecture – 1 (Introduction to Web Applications and PHP)

Long Answer Questions: -

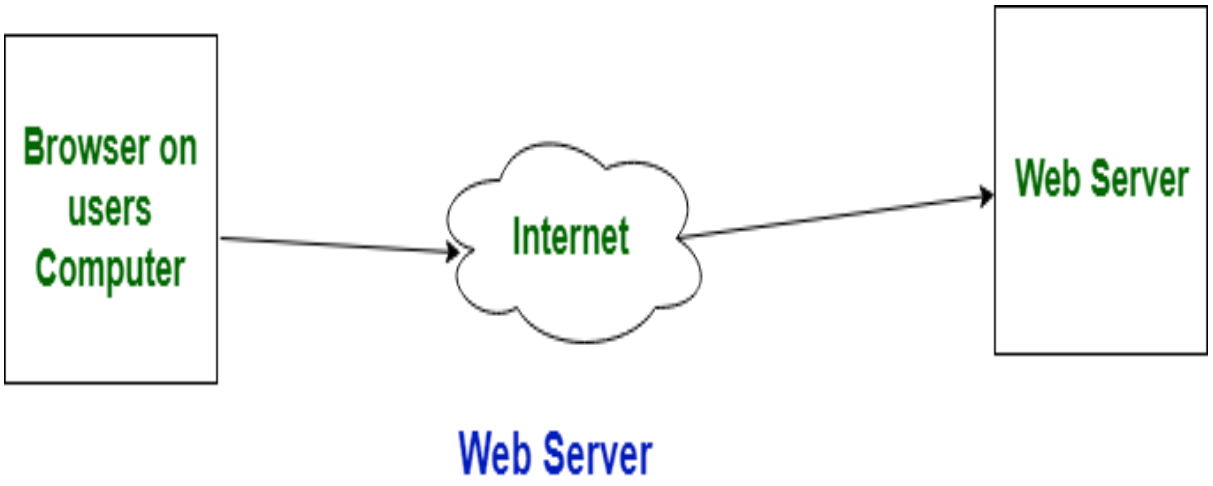
Q.1) What is the difference between web applications and webserver?

Ans.) A server is a central repository where information and computer programs are held and accessed by the programmer within the network. **Web server** and **Application server** are kinds of the server which employed to deliver sites and therefore the latter deals with application operations performed between users and back-end business applications of the organization.

Web Server: It is a computer program that accepts the request for data and sends the specified documents. Web server may be a computer where the online content is kept. Essentially internet server is employed to host sites however there exist different web servers conjointly like recreation, storage, FTP, email, etc.

Example of Web Servers:

- Apache Tomcat
- Resin

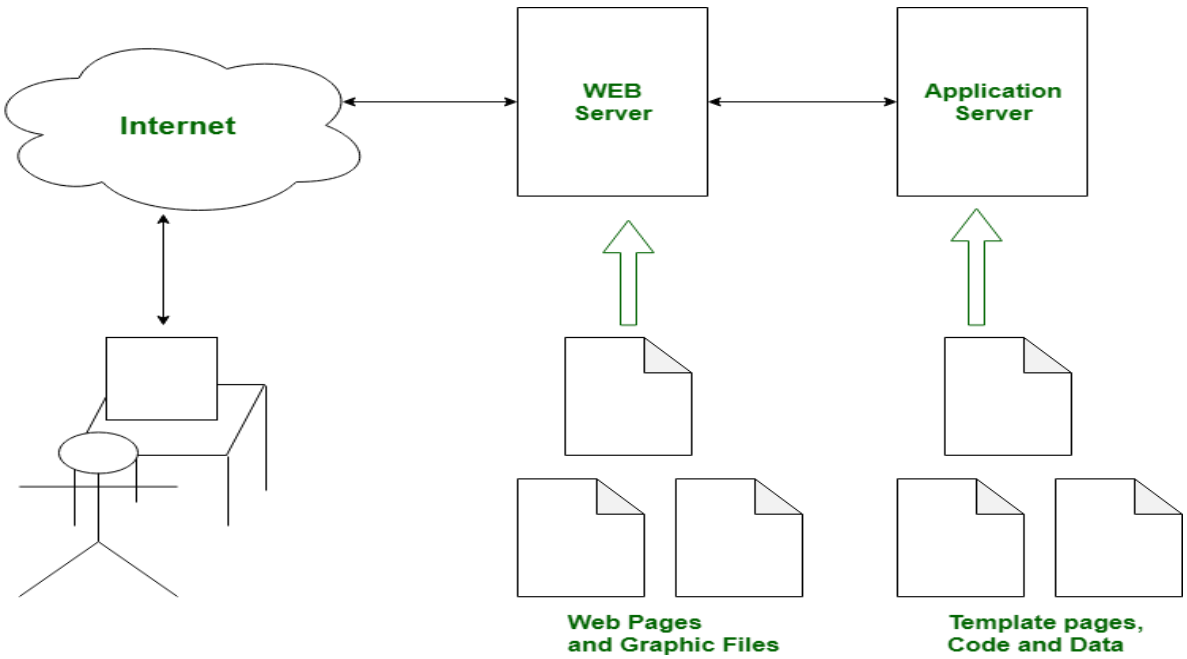


Web Server

Application server: It encompasses Web container as well as EJB container. Application servers organize the run atmosphere for enterprises applications. Application server may be a reasonably server that mean how to put operating system, hosting the applications and services for users, IT services and organizations. In this, user interface similarly as protocol and RPC/RMI protocols are used.

Examples of Application Server:

- Weblogic
- JBoss
- Websphere



Difference between web server and application server:

S. No.	Web Server	Application Server
1.	Web server encompasses web container only.	While application server encompasses Web container as well as EJB container.
2.	Web server is useful or fitted for static content.	Whereas application server is fitted for dynamic content.
3.	Web server consumes or utilizes less resources.	While application server utilizes more resources.
4.	Web servers arrange the run environment for web applications.	While application servers arrange the run environment for enterprises applications.
5.	In web servers, multithreading is not supported.	While in application server, multithreading is supported.
6.	Web server's capacity is lower than application server.	While application server's capacity is higher than web server.
7.	In web server, HTML and HTTP protocols are used.	While in this, GUI as well as HTTP and RPC/RMI protocols are used.

Q.2) What is HTTP protocol?

Ans.) HTTP is a **Hypertext Transfer Protocol.** HTTP works similar to other application services like SMTP and FTP. Like the performance of FTP, it transfers a document using service of TCP port. But it uses just one TCP connection i.e. at data link and no individual Control Connection is used.

HTTP is a protocol which fetches resources such as HTML documents. It is used for exchanging data on the Web and is a client-server protocol which means requests are initiated by the recipient usually the Web browser.
The controls from the client-side delivered in a request message into the webserver. The web server sends the requested content at a response message. The HTTP doesn't provide any security and makes use of SSL (Secure Socket layer) to club security in its communication.

How HTTP works: -HTTP provides users a way by distributing hypertext messages between servers and clients to interact. HTTP clients generally use Transmission Control Protocol (TCP) connection to communicate with servers.

HTTP uses various tasks to be performed by request methods which are enlisted as below –

- **GET** = Requests a specific source in its entirety.
- **HEAD** = A specific resource with no body content.
- **POST** = Adds articles, messages and information to another page under an existing web resource.

- **PUT** = Directly modifies a current web source and creates a new URL if need be.
- **DELETE** = Eliminates a specified source.
- **TRACE** = Shows users any modifications or additions made to an internet resource.
- **OPTIONS** = Show user's which HTTP methods are available for a particular URL.
- **CONNECT** = Transforms the requested link.
- **PATCH** = Partially modifies an internet resource HTTP servers use the GET and HEAD methods.

Q.3) What is TCP?

Ans.) TCP is a connection-oriented protocol which states a connection is established and maintained until the application data at each end have finished exchange. TCP breaks application data into packets. This packet delivers to the transport layer. Layer 4 manages flow control and provide error free data transmission and handles retransmission of dropped or garbled packets and acknowledges all packets that arrive. In the OSI model, TCP covers parts of Layer 4 the transport layer and Layer 5 covers the session layer.

How TCP Works: - TCP's job is to ensure that all data sent in a stream moves from client to server in a correct order and is intact. TCP uses a technique known as positive acknowledgement with retransmission, requiring the receiving end of a transmission to give a response as to what data has been received. The bytes sent can exactly match the bytes received. No data is altered or lost along the way.

Connection is established and a 3-way handshake is made. First, the source sends a SYN request packet to the server in order to start session establishment process. Then, the server sends a SYN-ACK packet to agree to the process. Lastly, the source sends an ACK packet to the target to confirm the process, after which the data can be sent.

Q.4) Comparison between HTTP and TCP.

Ans.) The comparison between HTTP and TCP are as follows: -

1. HTTP is a Hypertext Transfer Protocol, whereas TCP full form is Transmission Control Protocol.
2. HTTP is utilized to access websites, while TCP is a session establishment protocol between client and server.
3. HTTP uses port 80 and TCP uses no port.
4. HTTP doesn't need authentication, whereas, TCP uses the TCP-AO.

5. HTTP is Stateless but not session less. In contrast, TCP is a Connection-Oriented Protocol.

6. HTTP is a One-way communication system, while on the other hand, TCP is a 3-Way Handshake (SYN, SYN-ACK, ACK).

7. In case you're using HTTP, HTTP appears in the URL of the site, and if you're using TCP, TCP appears in IP.

8. HTTP establishes data link communication only but TCP establishes session connection.

9. HTTP is useful for transfer of smaller documents like webpages, on the other hand, TCP is useful to setup connection for data transfer.

10. HTTP is faster in comparison to TCP, which is slower.

Q.5) What is scripting language?

Ans.) All scripting languages are programming languages. The scripting language is basically a language where instructions are written for a run time environment. They do not require the compilation step and are rather interpreted. It brings new functions to applications and glue complex system together. A scripting language is a programming language designed for integrating and communicating with other programming languages.

Advantages of scripting languages:

- **Easy learning:** The user can learn to code in scripting languages quickly, not much knowledge of web technology is required.

- **Fast editing:** It is highly efficient with the limited number of data structures and variables to use.

- **Interactivity:** It helps in adding visualization interfaces and combinations in web pages. Modern web pages demand the use of scripting languages. To create enhanced web pages, fascinated visual description which includes background and foreground colors and so on.

- **Functionality:** There are different libraries which are part of different scripting languages. They help in creating new applications in web browsers and are different from normal programming languages.

Application of Scripting Languages: Scripting languages are used in many areas:

- Scripting languages are used in web applications. It is used in server side as well as client side. Server-side scripting languages are: JavaScript, PHP, Perl etc. and client-side scripting languages are: JavaScript, AJAX, jQuery etc.

- Scripting languages are used in system administration. For example: Shell, Perl, Python scripts etc.

- It is used in Games application and Multimedia.

- It is used to create plugins and extensions for existing applications.

Q.6) What is Client-Side Scripting and Server-Side Scripting?

Ans.) Client-side scripting:

Web browsers execute client-side scripting. It is used when browsers have all code. Source code is used to transfer from webserver to user's computer over the internet and run directly on browsers. It is also used for validations and functionality for user events.

It allows for more interactivity. It usually performs several actions without going to the user. It cannot be basically used to connect to databases on a web server. These scripts cannot access the file system that resides in the web browser. Pages are altered on basis of the user's choice. It can also be used to create "cookies" that store data on the user's computer.

Server-side scripting:

Web servers are used to execute server-side scripting. They are basically used to create dynamic pages. It can also access the file system residing at the webserver. A server-side environment that runs on a scripting language is a web server.

Scripts can be written in any of a number of server-side scripting languages available. It is used to retrieve and generate content for dynamic pages. It is used to require to download plugins. In this load times are generally faster than client-side scripting. When you need to store and retrieve information a database will be used to contain data. It can use huge resources of the server. It reduces client-side computation overhead. The server sends pages to the request of the user/client.

Difference between client-side scripting and server-side scripting:

Client-side scripting	Server-side scripting

Source code is visible to the user.	Source code is not visible to the user because its output of server-side is an HTML page.
It usually depends on the browser and its version.	In this any server-side technology can be used and it does not depend on the client.
It runs on the user's computer.	It runs on the webserver.
There are many advantages linked with this like faster response times, a more interactive application.	The primary advantage is its ability to highly customize, response requirements, access rights based on user.
It does not provide security for data.	It provides more security for data.
It is a technique used in web development in which scripts run on the client's browser.	It is a technique that uses scripts on the webserver to produce a response that is customized for each client's request.
HTML, CSS, and JavaScript are used.	PHP, Python, Java, Ruby is used.

Q.7) Give a brief of PHP history.

Ans.) PHP as it's known today is actually the successor to a product named PHP/FI. Created in 1994 by **Rasmus Lerdorf**, the very first incarnation of PHP was a simple set of Common Gateway Interface (CGI) binaries written in the C programming language.

- Originally used for tracking visits to his online resume, he named the suite of scripts "Personal Home Page Tools," more frequently referenced as "PHP Tools."

- Over time, more functionality was desired, and Rasmus rewrote PHP Tools, producing a much larger and richer implementation.

- This new model was capable of database interaction and more, providing a framework upon which users could develop simple dynamic web applications such as guestbooks.

- In June of 1995, Rasmus » released the source code for PHP Tools to the public, which allowed developers to use it as they saw fit.

- This also permitted - and encouraged - users to provide fixes for bugs in the code, and to generally improve upon it.

- In September of that year, Rasmus expanded upon PHP and - for a short time - actually dropped the PHP name.

- Now referring to the tools as FI (short for "Forms Interpreter"), the new implementation included some of the basic functionality of PHP as we know it today.

- It had Perl-like variables, automatic interpretation of form variables, and HTML embedded syntax.

- The syntax itself was similar to that of Perl, albeit much more limited, simple, and somewhat inconsistent.

- In fact, to embed the code into an HTML file, developers had to use HTML comments. Though this method was not entirely well-received, FI continued to enjoy growth and acceptance as a CGI tool --- but still not quite as a language.

- However, this began to change the following month; in October, 1995, Rasmus released a complete rewrite of the code. Bringing back the PHP name, it was now (briefly) named "Personal Home Page Construction Kit," and was the first release to boast what was, at the time, considered an advanced scripting interface.

- The language was deliberately designed to resemble C in structure, making it an easy adoption for developers familiar with C, Perl, and similar languages.

Q.8) Describe different versions of PHP.

Ans.) PHP FI stands for Personal Home Page and Form Interpreter. It deals with processing forms, sending SQL statements and Flow control. Though objects were not in the perspective, there was support for associative arrays, variables, and functions. PHP/FI was also proficient with receiving file uploads from RFC-1867 compliant web browsers. It enabled people to upload both binary as well as text files. The access control and logical functions of PHP/FI offered full control over who is permitted to upload a file and what actions to be taken with the uploaded file. This PHP version has supported the MySQL database package. It allowed programmers to enter details into a database and access those details via simple embedded SQL queries.

In November 1997, PHP/FI was rewritten and given a version 2.0 status. Though the initial versions of PHP don’t include enough features and lived a very short development life, they have defined a benchmark for the forthcoming PHP versions to receive further advancements.

Features of PHP/FI

Built-in support for

- MySQL

- DBM
- Postgres95 databases
- User-defined functions
- Cookies

Features of PHP/FI 2

- Form Handling
- HTML Embedding

PHP 3

Zeev Suraski and Andi Gutmans realized that PHP was lacking some common features. They decided to rewrite the scripting language, so they teamed up with Rasmus to launch PHP 3. PHP 3 comes with limited object-oriented support that added extra fuel to the growth of PHP. Another attractive feature of PHP 3 was its extensibility. It was implemented with a modular approach that made it hassle-free for programmers to extend functionality. Moreover, the combination of PHP3 with Apache quickly resulted in the widespread adoption of the scripting language and PHP 3 powered about 10 % of all sites on the internet.

Features of PHP 3

- API Extension
- Object-Oriented Support
- Introduction of Zend Engine

PHP 4

PHP 4 was launched in 2000, comprising major changes to PHP version 3 in all aspects. It ensured backward compatibility which made upgrading to PHP 4 from PHP 3 seem much smoother than the upgrade from PHP/FI 2 to PHP 3. PHP 4 also comes with the web server abstraction that enables PHP to run on Microsoft's IIS, Apache 2, Zeus and more. The performance of this version has taken a huge leap upward because of the two main factors:

1. PHP 4 uses "Compile first, execute later" paradigm whereas other previous versions employed the "execute while interpreting" paradigm
2. PHP 4 hosted multi-threading feature; you can run the lengthy but non-critical functions independently, which further streamlines execution.

Features of PHP 4

- Enhanced run time execution
- Introduced Super global
- Web server abstraction layer
- Object overloading support
- Output buffering support
- Improved resource handling

PHP 5

In August 2008, it has been uncovered that no further effort would be added in fixing bugs, patching or improving security holes discovered in PHP version 4, so the development effort moved to PHP version 5. This version is primarily focused on language maturity and provides an array of new functionalities like 64-bit support, built-in debugging, an opcode cache, generators, namespaces, traits and more. With PHP 5, developers can declare how their objects can be used. It enables developers to work on each other's code effortlessly. In addition, object-oriented scripts with a broad variety of functions made this version more flexible and simpler to work with.

PHP version 5 has undergone a huge number of improvements with the following subversions:

Features of PHP 5. 1

- Support for custom auto loading
- Over 400 various bug fixes
- Significant performance improvements
- Improved time zone support

Features of PHP 5.2

- Improved memory manager
- Increased default memory limit
- New extensions with JSON, Zip
- Input filtering extension
- Upgraded bundle SQLite

Features of PHP 5.3

- Support for namespaces
- Support for jump labels
- Dynamic access to static methods
- Support for Late Static Bindings
- A garbage collector for circular references

Features of PHP 5.4

- Support for traits
- Tracking the upload progress of files
- Support for Binary number format
- Function array dereferencing
- Improved parse error messages

Features of PHP 5.5

- empty() supports arbitrary expressions
- Generators
- password_hash()
- Introduced finally keyword
- array & string literal dereferencing

Features of PHP 5.6

- Constant expressions
- SSL/TLS improvements
- Default character encoding
- pgsql async support
- Introduced phpdbg, an interactive debugger
- Operator overloading
- Gost-crypto hash algorithm

PHP 6

PHP version 6 was supposed to feature native Unicode support, but that extension never succeeded. When PHP 6 under development, the UTF-8 wasn't widely adopted. As such, PHP becomes a false start and development was gradually stopped in favor of PHP version 7.

PHP 7

The latest PHP version made a revolution in the way we build applications that influences everything from the website & mobile to organizations & the cloud. PHP version 7 comes with features like Null Coalescing Operator, Scalar Type Declarations, Return Type declarations, Spaceship operator and many more. PHP 7 offers impressive performance, and better error handling along with scalar type hints, return type declaration and many other syntax modifications.

When compared to PHP 5.6, this latest PHP version ensures a substantial reduction in the demands on the server. The following graph illustrates how the performance of PHP 7 made a benchmark on the WordPress CMS, and Laravel & Zend Frameworks compared to PHP 5.6.

As such, it becomes an environment-friendly and cost-effective choice, because servers require less energy to run PHP 7 applications.

Here are the new features with each release of PHP 7 subversions.

Features of PHP 7.1

- Void functions
- Nullable Types

- Asynchronous Signal handling
- Class Constant visibility
- Symmetric Array Destructuring

Features of PHP 7.2

- Abstract Method Overriding
- New Object Type
- Password hashing with Argon 2

Features of PHP 7.3

- Trailing Comma
- Flexible Heredoc and Nowdoc Syntaxes
- PCRE2 Migration
- JSON_THROW_ON_ERROR

PHP is widely used over other competitors like Python, Ruby and Lamp Stack for its abovementioned advantages. Each latest version of PHP is 100 times better than its previous releases. As such, many top companies favor this scripting language for web application development. No matter of the number of competitors for PHP, still it is highly popular as well as a highly demanded programming language in emerging web application needs.

Short Answer Questions: -

Q.1) What is PEAR in PHP?

Ans.) PEAR is a **framework** and repository for reusable PHP components. PEAR stands for **PHP Extension and Application Repository**. It contains all types of PHP code snippets and libraries. It also provides a command line interface to install “packages” automatically. **Q.2) What are the common uses of PHP?**

Ans.)

- It performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- It can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You can add, delete, modify elements within your database with the help of PHP.
- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website and also encrypt data. **Technical Tasks: -**

1. Download and Install server to run PHP file.

2. Write a program to print “Hello World!” and run it in server.
3. What is a difference between echo and print statement?

Interview Questions: -

1. What is difference between Server-side scripting and Client-side scripting?
2. Define HTTP and HTTPS?
3. What are the layers of TCP/IP?
4. Define Servers and discuss various servers like Apache, WAMP, XAMP, LAMP, MAMP, and IIS servers.
5. What is HTTPCONFIG file in WAMP server?

Multiple Choice Questions: -

1. Which of the following is cross platform AMP stack?
   a) Wamp
   b) Xampp
   c) Lamp
   d) Mamp

2. Can your own computer can be reserved for a dedicated server?
   a) True
   b) False

3. What is IP address for domain name server localhost:80?
   a) 192.168.1.1:80
   b) 127.0.0.0.1
   c) 127.0.0.1:80
   d) 192.168.1.2

4. Which of the file in xampp-stack is responsible for all configuration related to php? a) Http.conf
   b) http-ssl.conf
   c) php.ini
   d) none of the above

5. Who is creator of php?
   a) Rasmus Ledorf
   b) Dennis M Ritchie
   c) Kris Morgan
   d) Ken Thompson

6. PHP is developed using which environment and by implementing which extension? a) Linux and C
   b) Mac and PHP
   c) Unix and PHP
   d) Unix and C

7. The only function used to see all the configuration of php installed
   a) php_version()
   b) php info()
   c) php_config()
   d) php_ini()

Answer Key: -

1. b	2. a	3. b	4. c
5. a	6. d	7. b	

PHP Lecture – 2 (PHP Basics, Variables, Constants)

Long Answer Questions: -

Q.1) Write the basic syntax of PHP.

Ans.) A PHP script is executed on the server, and the plain HTML result is sent back to the browser.

A PHP script can be placed anywhere in the document.

A PHP script **starts with <?php and ends with?>:**

```
<?php
// PHP code goes here ?>
```

The default file extension for PHP files is ".php".

A PHP file normally contains HTML tags, and some PHP scripting code.

Below, we have an example of a simple PHP file, with a PHP script that uses a built-in PHP function "echo" to output the text "Hello World!" on a web page:

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

<h1>My first PHP page</h1>
 <?php
echo "Hello World!";
?>
</body>
</html>
```

PHP statements end with a semicolon (;). In PHP, keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are not case-sensitive.

Note: However; all variable names are case-sensitive!

Look at the example below; only the first statement will display the value of the $color variable! This is because $color, $COLOR, and $coLOR are treated as three different variables.

Q.2) What are PHP comments?

Ans.) A comment in PHP code is a line that is not executed as a part of the program. Its only purpose is to be read by someone who is looking at the code.

Comments can be used to:

- Let others understand your code
- Remind yourself of what you did - Most programmers have experienced coming back to their own work a year or two later and having to re-figure out what they did. Comments can remind you of what you were thinking when you wrote the code

Syntax for single-line comments:

```
<!DOCTYPE html>
<html>
<body>
 <?php
// This is a single-line comment

# This is also a single-line comment
?>
</body> </html>
```

Syntax for multiple-line comments:

```
<!DOCTYPE html>
<html>
<body>
 <?php
/*
This is a multiple-lines comment block that
spans over multiple
lines */
?>

</body> </html>
```

Q.2) How many types of output functions are there in PHP?

Ans.) PHP is a server-side scripting language for creating dynamic web pages. There are so many functions available for displaying output in PHP. The basic functions for displaying output in PHP are as follows:

- Print() Function
- Echo() Function
- Printf() Function
- Sprintf() Function
- Var_dump() Function
- Print_r() Function

Print() Function

Using this function, we can display the outputs in the browser. This function returns the Boolean value true. We cannot print the multiple statements using this function. The print function plays the same role as the echo function.

Example1

```
<html>
<body bgcolor="pink">
<?php
print "Welcome Vineet Kumar Saini !!";
?>
</body> </html>
```

Example2

```
<html>
<body bgcolor="pink">
<?php
$rval="Welcome Vineet Kumar Saini is MCA Qualified !!";
print $rval; ?>
</body> </html>
```

Example3

```
<html>
<body bgcolor="pink">
<?php
print "Welcome Vineet Kumar Saini is "," ","MCA Qualified !!"; ?>
</body>
</html>
```

Echo() Function

The echo() function outputs one or more strings. Using this function we can display multiple statements in the browser. The echo() function is slightly faster than print(). Because it won't return a value.

Example1

```
<html>
<body bgcolor="pink">
<?php
echo "Welcome Vineet Kumar Saini !!";
?>
</body> </html>
```

Example2

```
<html>
<body bgcolor="pink">
<?php
echo "Welcome Vineet Kumar Saini is "," ","MCA Qualified !!"; ?>
</body>
```

```
</html>
```

Printf() Function

The printf() function is also used in C, C++. The printf() function outputs a formatted string. Using this function we can display the output with the help of the formats specified.

Example

```
<html>
<body bgcolor="pink">
<?php
$name="Vineet Saini";
$age=24;
printf("The age of %s is %d years.",$name,$age);
?>
</body> </ html>
```

sprintf() Function

The sprintf() function writes a formatted string to a variable. This is the same as printf, but instead of displaying the output on the web page, it returns that output. sprintf() prints the result to a string.

Example

```
<html>
<body bgcolor="pink">
<?php
$name="Vineet Saini";
$age=24;
$rv=sprintf ("The age of %s is %d years.",$name, $age);
echo $rv; ?>
```

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

var_dump() Function

The var_dump() function displays information of a variable that includes its type and value. This function displays the variable value along with the variable data type. The var_dump() function is used to display structured information (type and value) about one or more variables.

Example

```
<html>
<body bgcolor="pink">
<?php
$name="Vineet Saini";
$age=24; var_dump($name);
var_dump($age);
?>
</body> </html>
```

print_r() Function

The Print_r() PHP function is used to return an array in a human readable form. This function displays the elements of an array and properties of an object. The print_r() function displays human-readable information about a variable.

Example

```
<html>
<body bgcolor="pink">
<?php
$arr=array ("Vineet","Kumar","Saini");
$arr1=array(10,20,30);
print_r($arr); print_r($arr1);
?>
</body>
</html>
```

Q.3) What are data types? How many data types does PHP have?

Ans.) Variables can store data of different types, and different data types can do different things.

PHP supports the following data types:

- String
- Integer

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

PHP String

A string is a sequence of characters, like "Hello world!".

A string can be any text inside quotes. You can use single or double quotes:

```
<!DOCTYPE html>
<html>
<body>
<?php
$x = "Hello world!";
$y = 'Hello world!';
echo $x; echo
"<br>";  echo $y;
?>

</body>
</html>
```

O/P- Hello world!
Hello world!

PHP Integer

An integer data type is a non-decimal number between **-2,147,483,648** and **2,147,483,647**.

Rules for integers:

- An integer must have at least one digit
- An integer must not have a decimal point
- An integer can be either positive or negative

- Integers can be specified in: decimal (base 10), hexadecimal (base 16), octal (base 8), or binary (base 2) notation

In the following example $x is an integer. The PHP var_dump () function returns the data type and value:

```
<!DOCTYPE html>
<html>
<body>
<?php   $x =
5985;
var_dump($x);
?>
</body>
</html>
```

O/P- int (5985)

PHP Float

A float (floating point number) is a number with a decimal point or a number in exponential form.

In the following example $x is a float. The PHP var_dump() function returns the data type and value:

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

<?php
$x = 10.365; var_dump($x);
?>
```

```
</body>

</html>
```

O/P- float(10.365)

PHP Boolean

A Boolean represents two possible states: TRUE or FALSE.

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

Booleans are often used in conditional testing. You will learn more about conditional testing in a later chapter of this tutorial.

PHP Array

An array stores multiple values in one single variable.

In the following example $cars is an array. The PHP var_dump() function returns the data type and value:

```
<!DOCTYPE html>

<html>

<body>


<?php

$cars = array("Volvo","BMW","Toyota"); var_dump($cars);

?>


</body>

</html>
```

O/P- array (3) {
  [0] =>
  String (5) "Volvo"
[1] =>
  String (3) "BMW"

```
  [2] =>
  String (6) "Toyota"
}
```

PHP NULL Value

Null is a special data type which can have only one value: NULL.

A variable of data type NULL is a variable that has no value assigned to it.

Tip: If a variable is created without a value, it is automatically assigned a value of NULL.

Variables can also be emptied by setting the value to NULL:

```
<!DOCTYPE html>

<html>

<body>
<?php

$x = "Hello world!";

$x = null; var_dump($x);

?>

</body>

</html>
```

O/P- NULL

Q.3) How many types of errors are there?

Ans.) A **PHP Error** occurs when something is wrong in the PHP code. The error can be as simple as a missing semicolon, or as complex as calling an incorrect variable.

To efficiently resolve a PHP issue in a script, you must understand what kind of problem is occurring.

The four types of PHP errors are:

1. Warning Error
2. Notice Error
3. Parse Error

4. Fatal Error

Warning Error

A **warning error in PHP** does not stop the script from running. It only warns you that there is a problem, one that is likely to cause bigger issues in the future.

The most common causes of warning errors are:

- Calling on an external file that does not exist in the directory
- Wrong parameters in a function

For instance:

```
<?php
echo "Warning error"';
include ("external_file.php");
?>
```

As there is no “external_file”, the output displays a message, notifying it failed to include it. Still, it doesn’t stop executing the script.

Notice Error

Notice errors are minor errors. They are similar to warning errors, as they also don’t stop code execution. Often, the system is uncertain whether it’s an actual error or regular code. Notice errors usually occur if the script needs access to an undefined variable.

```
<?php
$a="Defined error";
echo "Notice error";
echo $b; ?>
```

In the script above, we defined a variable (**$a**), but called on an undefined variable (**$b**). PHP executes the script but with a notice error message telling you the variable is not defined.

Parse Error (Syntax)

Parse errors are caused by misused or missing symbols in a syntax. The compiler catches the error and terminates the script.

Parse errors are caused by:

- Unclosed brackets or quotes
- Missing or extra semicolons or parentheses
- Misspellings

For example, the following script would stop execution and signal a parse error:

```
<?php echo
"Red"; echo
"Blue"; echo
"Green"
?>
```

It is unable to execute because of the missing semicolon in the third line.

Fatal Error

Fatal errors are ones that crash your program and are classified as critical errors. An undefined function or class in the script is the main reason for this type of error.

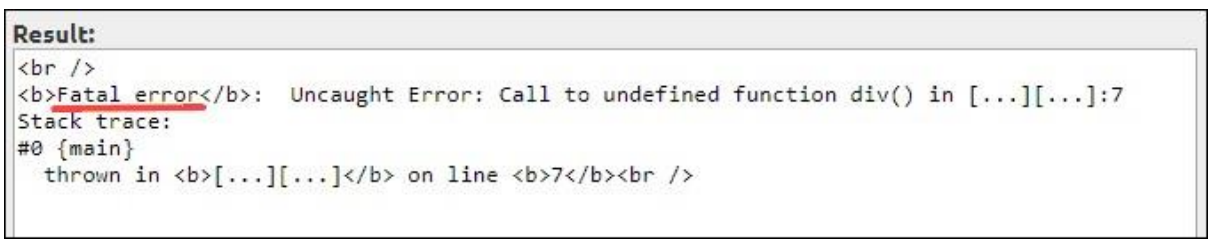
There are three (3) types of fatal errors:

1. **Start-up fatal error** (when the system can't run the code at installation)
2. **Compile time fatal error** (when a programmer tries to use non - existent data)
3. **Runtime fatal error** (happens while the program is running, causing the code to stop working completely)

For instance, the following script would result in a fatal error:

```
<?php
function sub ()
{
$sub=6-1;
echo "The sub= ".$sub;
}
div();
?>
```

The output tells you why it is unable to compile, as in the image below:



Q.3) What is variable? How can we declare variable in PHP? How many types of PHP variables are there?

Ans.) Variables are "containers" for storing information. The main way to store information in the middle of a PHP program is by using a variable.

Creating (Declaring) PHP Variables

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

```
<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;
?>
```

After the execution of the statements above, the variable $txt will hold the value Hello world!, the variable $x will hold the value 5, and the variable $y will hold the value 10.5.

Note: When you assign a text value to a variable, put quotes around the value.

Note: Unlike other programming languages, PHP has no command for declaring a variable. It is created the moment you first assign a value to it.

PHP Variables

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total_volume).

Rules for PHP variables:

- A variable starts with the $ sign, followed by the name of the variable
- A variable name must start with a letter or the underscore character
- A variable name cannot start with a number
- A variable name can only contain alpha-numeric characters and underscores (A-z, 09, and _)
- Variable names are case-sensitive ($age and $AGE are two different variables)

Remember that PHP variable names are case-sensitive!

PHP has a total of eight data types which we use to construct our variables −

- **Integers** − are whole numbers, without a decimal point, like 4195.
- **Doubles** − are floating-point numbers, like 3.14159 or 49.1.
- **Booleans** − have only two possible values either true or false.
- **NULL** − is a special type that only has one value: NULL.
- **Strings** − are sequences of characters, like 'PHP supports string operations.'

- **Arrays** − are named and indexed collections of other values.
- **Objects** − are instances of programmer-defined classes, which can package up both other kinds of values and functions that are specific to the class.
- **Resources** − are special variables that hold references to resources external to PHP (such as database connections).

PHP Variables Scope

In PHP, variables can be declared anywhere in the script.

The scope of a variable is the part of the script where the variable can be referenced/used.

PHP has three different variable scopes:

- local
- global
- static

Global and Local Scope

A variable declared **outside** a function has a GLOBAL SCOPE and can only be accessed outside a function:

Example

Variable with global scope:

```
<?php
$x = 5; // 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>"; ?>
```

O/P- Variable x inside function is: 5

Variable x outside function is:

A variable declared **within** a function has a LOCAL SCOPE and can only be accessed within that function: **Example**

Variable with local scope:

```
<?php
function myTest() {
  $x = 5; // 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>"; ?>
```

O/P- Variable x inside function is: 5

Variable x outside function is:

PHP The global Keyword

The global keyword is used to access a global variable from within a function.

To do this, use the global keyword before the variables (inside the function):

Example

```
<?php $x
= 5;
$y = 10;

function myTest() {
global $x, $y;
  $y = $x + $y;
}

myTest();
echo $y; // outputs 15 ?>
```

O/P- 15

PHP also stores all global variables in an array called $GLOBALS[*index*]. The *index* holds the name of the variable. This array is also accessible from within functions and can be used to update global variables directly.

PHP The static Keyword

Normally, when a function is completed/executed, all of its variables are deleted. However, sometimes we want a local variable NOT to be deleted. We need it for a further job.

To do this, use the static keyword when you first declare the variable:

Example

```
<?php function
myTest() {   static
$x = 0;   echo $x;
  $x++;
}

myTest();
myTest();
myTest(); ?>
```

O/P- 0
1
2

Then, each time the function is called, that variable will still have the information it contained from the last time the function was called.

Note: The variable is still local to the function.

Q.4) What do you mean by super global variables?

Ans.) Super globals were introduced in PHP 4.1.0, and are built-in variables that are always available in all scopes.

PHP Global Variables – Super Globals

Some predefined variables in PHP are "super globals", which means that they are always accessible, regardless of scope - and you can access them from any function, class or file without having to do anything special.

The PHP super global variables are:

- $GLOBALS
- $_SERVER
- $_REQUEST
- $_POST
- $_GET
- $_FILES
- $_ENV
- $_COOKIE
- $_SESSION

PHP $GLOBALS

$GLOBALS is a PHP super global variable which is used to access global variables from anywhere in the PHP script (also from within functions or methods).

PHP stores all global variables in an array called $GLOBALS[*index*]. The *index* holds the name of the variable.

The example below shows how to use the super global variable $GLOBALS:

Example

```
<?php
$x = 75;
$y = 25;

function addition() {
  $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
}

addition();
echo $z; ?>
```

O/P – 100

PHP $_SERVER

$_SERVER is a PHP super global variable which holds information about headers, paths, and script locations.

The example below shows how to use some of the elements in $_SERVER:

Example

```
<?php
echo $_SERVER['PHP_SELF']; echo
"<br>";
echo $_SERVER['SERVER_NAME'];
echo "<br>";
echo $_SERVER['HTTP_HOST']; echo
"<br>";
echo $_SERVER['HTTP_REFERER'];
echo "<br>";
echo $_SERVER['HTTP_USER_AGENT'];
echo "<br>";
echo $_SERVER['SCRIPT_NAME']; ?>
```

O/P - demo/demo_global_server.php

35.194.26.41
35.194.26.41 https://tryphp.w3schools.com/showphp.php?filename=demo_global_server Mozilla/5.0 (Windows NT 10.0; rv:97.0) Gecko/20100101 Firefox/97.0 /demo/demo_global_server.php

PHP $_REQUEST

PHP $_REQUEST is a PHP super global variable which is used to collect data after submitting an HTML form.

The example below shows a form with an input field and a submit button. When a user submits the data by clicking on "Submit", the form data is sent to the file specified in the action attribute of the <form> tag. In this example, we point to this file itself for processing form data. If you wish to use another PHP file to process form data, replace that with the filename of your choice. Then, we can use the super global variable $_REQUEST to collect the value of the input field:

Example

```
<html>
<body>

<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>
 <?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // collect value of input field
$name = $_REQUEST['fname'];
  if (empty($name)) {
    echo "Name is empty";
  } else {
    echo $name;
  }
} ?>
</b
ody
>
</h
tml
>
```

PHP $_POST

PHP $_POST is a PHP super global variable which is used to collect form data after submitting an HTML form with method="post". $_POST is also widely used to pass variables.

The example below shows a form with an input field and a submit button. When a user submits the data by clicking on "Submit", the form data is sent to the file specified in the action attribute of the <form> tag. In this example, we point to the file itself for processing form data. If you wish to use another PHP file to process form data, replace that with the filename of your choice. Then, we can use the super global variable $_POST to collect the value of the input field:

Example

```
<html>
<body>

<form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
  Name: <input type="text" name="fname">
  <input type="submit">
</form>

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  // collect value of input field
$name = $_POST['fname'];   if
(empty($name)) {
    echo "Name is empty";
  } else {
    echo $name;
  }
} ?>

</body> </html>
```

PHP $_GET

PHP $_GET is a PHP super global variable which is used to collect form data after submitting an HTML form with method="get".
$_GET can also collect data sent in the URL.

Assume we have an HTML page that contains a hyperlink with parameters:

```
<html>
<body>

<a href="test_get.php?subject=PHP&web=W3schools.com">Test $GET</a>
```

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

When a user clicks on the link "Test $GET", the parameters "subject" and "web" are sent to "test_get.php", and you can then access their values in "test_get.php" with $_GET.

The example below shows the code in "test_get.php":

Example

```
<html>
<body>
 <?php
echo "Study " . $_GET['subject'] . " at " . $_GET['web'];
?>

</body> </html>
```

Short Answer Questions: -

Q1. How to execute a PHP script from the command line?

Ans.) To execute a PHP script, use the **PHP Command Line Interface (CLI)** and specify the file name of the script in the following way:

```
php script.php
```

Q2. Is PHP a case sensitive language?

Ans.) PHP is **partially** case sensitive. The variable names are case-sensitive but function names are not. If you define the function name in lowercase and call them in uppercase, it will still work. User-defined functions are not case sensitive but the rest of the language is case-sensitive.

Q3. What is the meaning of 'escaping to PHP'?

The PHP parsing engine needs a way to **differentiate** PHP code from other elements in the page. The mechanism for doing so is known as 'escaping to PHP'. Escaping a string means to reduce **ambiguity** in quotes used in that string.

Q4. What are the rules for naming a PHP variable?

Ans.) The following rules are needed to be followed while naming a **PHP variable**:

- Variable names must begin with a **letter** or **underscore** character.
- A variable name can consist of numbers, letters, underscores but you cannot use **characters** like + , – , % , (,) . & , etc.

Q5. What are the two most common ways to start and finish a PHP block of code? Ans.)

The two most common ways to **start** and **finish** a PHP block of code are:

```
<?php [ --- PHP code---- ] ?>

<? [--- PHP code ---] ?>
```

Q6. Types of PHP Tags?

Ans.) There are only four types of Tags available in PHP.

1. Universal Style Tag
2. Short Open Tag
3. ASP Style Tag
4. Script Style Tag

Q7. What is GET and POST method in PHP?

The GET method sends the **encoded** user information appended to the page request. The page and the encoded information are separated by the **? character**.

For example –

```
<a
href="http://www.test.com/index.htm?name1=value1&name2=value2">http://www.
test.com/index.htm?name1=value1&name2=value2

</a>
```

The POST method transfers information via **HTTP** headers. The information is encoded as described in case of GET method and put into a header called **QUERY_STRING**.

Q8. What is the difference between GET and POST method? Ans.)

GET	POST
• The GET method is restricted to send upto 1024 characters only.	• The POST method does not have any restriction on data size to be sent.
• GET can't be used to send binary data, like images or word documents, to the server.	• The POST method can be used to send ASCII as well as binary data.

• The data sent by GET method can be accessed using QUERY_STRING environment variable.	• The data sent by POST method goes through HTTP header so security depends on HTTP protocol.
• The PHP provides $_GET associative array to access all the sent information using GET method.	• The PHP provides $_POST associative array to access all the sent information using POST method.

Q9. What is the difference between "echo" and "print" in PHP?

- PHP **echo** output one or more string. It is a language construct not a function. So use of parentheses is not required. But if you want to pass more than one parameter to echo, use of parentheses is required. Whereas, PHP **print** output a string. It is a language construct not a function. So use of parentheses is not required with the argument list. Unlike echo, it always returns 1.
- **Echo** can output one or more string but **print** can only output one string and always returns 1.
- **Echo** is faster than print because it does not return any value.

Technical Tasks: -

1. Write a program having different variables of different datatypes and display the output.
2. Write a program to show types of error in PHP.

Interview Questions: -

1. Is PHP a strongly typed language?
2. What is meant by variable variables in PHP?
3. What are the differences between echo and print?
4. List down all Super Global Vaiables.
5. List down all the errors in PHP and their solutions.
6. What is the difference between $message and $$message?

Multiple Choice Questions: -

1. Predict the output of the following

```
<?php

 $a = "xyz"
```

```
 $$a = 300
 $a=100;
 echo $xyz;
?>
```

a) 100
b) Undefined index
c) Xyz
d) 300

2. Predict the output of the following

```
<?php
$a= “Raja Ram Mohan Roy”; echo
‘$a is Raja Ram Mohan Roy ’
?>
```

a) Raja Ram Mohan Roy is Raja Ram Mohan Roy
b) $a is Raja Ram Mohan Roy
c) undefined Variable $a
d) None of the above

3. which of these functions can be used to print resource

a) echo
b) print
c) print_r
d) None of the Above

4. Echo can be used to print multiple argument in a single line

c) True
d) False

5. Which of the following statement about echo or print are not correct?

a) echo can be used to print multiple argument
b) print is language construct
c) echo is not a language construct
d) print and echo cannot be used to print array

6. The only primitive data type that do not render anything

a) bool
b) integer
c) object
d) Array

7. The only function used to output the internal data structure of any variable or constant in terms of data value is a) die()

b) dump()
c) var_dump()
d) args_var()

8. **The only function used to terminate execution of php script** a) die()
   b) Kill()
   c) terminate()
   d) None of the Above

Answer Key: -

1. d	2. b	3. c	4. a
5. c	6. a	7. c	8. a

PHP Lecture – 3 (Operators & Control Structures)

Long Answer Questions: -

Q1. What are Conditional Statements in PHP?

Ans.) Conditional Statements performs different computations or actions depending on conditions. In PHP, the following are conditional statements.

a) **if** statement - The if statement is used to test a specific condition. If the condition is true, a block of code (if-block) will be executed. **Syntax -**

```
if (condtion)
{
        statements
}
```

b) **if - else** statement - The if-else statement provides an else block combined with the if statement which is executed in the false case of the condition.

Syntax –

```
if (condtion)
{
        statements
} else
{
        statements
}
```

c) **if - elseif - else** statement - The elseif statement enables us to check multiple conditions and execute the specific block of statements depending upon the true condition among them. **Syntax –**

```
if (condtion1)
{
        statements
}
else if (condtion2)
{
        statements
}
else if (condtion3)
{
        statements
} .
.
else
{
        statements
}
```

d) **switch** statement - The switch statement enables us to execute a block of code from multiple conditions depending upon the expression. **Syntax –**

```
switch (expression)
```

```
{
case 1: statements
                break;
case 2: statements
                break;
case 3: statements
                break;
. . default: statements }
```

Q2. Explain while loop with proper example?

Ans.) With the while loop we can execute a set of statements as long as a condition is true. The while loop is mostly used in the case where the number of iterations is not known in advance. Syntax –

```
while (condition)
{
        statements
}
```

Example - whileloopdemo.php

```
<html>
<head>
<title>While Demo</title>
</head>
<body>
<h1>While Demo</h1>
<?php $n=1;
while($n<=5)
{
    echo "$n <br/>";
    $n++;
}
?>
</body>
</html>
```

Q3. Explain Jump Statements with proper examples?

Ans.) Jump statements in PHP are used to alter the flow of a loop like you want to skip a part of a loop or terminate a loop.

In PHP, the following are jump statements

- break statement
- continue statement

Break Statement - The break is a keyword in php which is used to bring the program control out of the loop. i.e. when a break statement is encountered inside a loop, the loop is terminated and program control resumes at the next statement following the loop.

The break statement breaks the loops one by one, i.e., in the case of nested loops, it breaks the inner loop first and then proceeds to outer loops. The break is commonly used in the cases where we need to break the loop for a given condition.

Example - Breakdemo.php

```
<html>
<head>
<title>Break Demo</title>
</head>
<body>
<h1>Break Demo</h1>
<?php
for($i=1;$i<=10;$i++)
{
    if($i==5)     {
        break;         //terminates the current loop
    }
    echo "$i <br/>";
}
echo "Loop is Over !";
?>
</body>
</html>
```

Continue Statement - The continue statement in php is used to bring the program control to the beginning of the loop. i.e. when a continue statement is encountered inside the loop, remaining statements are skipped and loop proceeds with the next iteration.

The continue statement skips the remaining lines of code inside the loop and start with the next iteration. It is mainly used for a particular condition inside the loop so that we can skip some specific code for a particular condition.

Example – continuedemo.php

```
<html>
<head>
<title>Continue Demo</title>
</head>
<body>
<h1>Continue Demo</h1>
<?php
for($i=1;$i<=10;$i++)
{
    if($i%2==0)
    {
        continue;    //terminates the current Iteration and moves to
Next
}
    echo "$i <br/>";
}
echo "Loop is Over !";
?>
</body>
</html>
```

Short Answer Questions: -

Q1. What are Control Structures.

Ans.) The Control Structures (or) Statements used to alter the execution process of a program.

These statements are mainly categorized into following:

- Conditional Statements
- Loop Statements
- Jump Statements

Q2. What are Loop Statements.

Ans.) Sometimes we may need to alter the flow of the program. If the execution of a specific code may need to be repeated several numbers of times, then we can go for loop statements.

In PHP, the following are loop statements

- while loop
- do - while loop

- for loop

Q3. Define operators and its types.

Ans.) An operator, in computer programming, is a symbol that usually represents an action or process. These symbols were adapted from mathematics and logic. An operator is capable of manipulating a certain value or operand. Simple answer can be given using expression *4 + 5 is equal to 9*. Here 4 and 5 are called operands and + is called operator.

PHP language supports following type of operators.

- Arithmetic Operators
- Comparison Operators
- Logical (or Relational) Operators
- Assignment Operators
- Conditional (or ternary) Operators

Q4. What is the purpose of break and continue statement?

Ans.) Break – It terminates the **for loop** or **switch** statement and transfers execution to the statement immediately following the for loop or switch.
Continue – It causes the **loop** to skip the remainder of its body and immediately retest its condition prior to reiterating.

Q5. What are the data types in PHP?

Ans.) PHP support 9 **primitive data types**:

Scalar Types	Compound Types	Special Types
• Integer • Boolean • Float • String	• Array • Object • Callable	• Resource • Null

Q6. Explain the syntax for 'foreach' loop with example.

Ans.) The foreach statement is used to **loop** through **arrays**. For each pass the value of the current array element is assigned to $value and the array pointer is moved by one and in the next pass next element will be processed.

Example –

<?php

```
$colors = array("blue", "white", "black");
foreach ($colors as $value) { echo
"$value
";
}
?>
```

Technical Tasks: -

1. Write a program in PHP to find greatest no. among three number.
2. Write a program in PHP to check the no. is odd or even.
3. Write a program in PHP to print a table of given number.
4. Develop a Program in PHP to calculate bill after taking input from user. Take the following logic to calculate the electricity bill ; For 1 to 150 units rate is Rs. 2.40, for next 150 (150 to 300) units rate is Rs. 3.00 and Rs. 3.20 for units above 300.
5. Write a program in PHP to find a no. is prime or not.
6. Write a program in PHP to check the no. is Armstrong or not.
7. Write a program in PHP to print prime numbers between 1 to 100.
8. Write a program in PHP to accept an integer and print it in reverse order.
   Input – 123
   Output – 321

Interview Questions: -

1. What are the different loops in PHP?
2. What is the difference between "echo" and "print" in PHP?
3. What is the difference between logical operator and bitwise operator?

Multiple Choice Questions: -

1. **In php 7 use of : colon in case decision control is used for**
   a) Decision block
   b) Suit block
   c) Opening Block
   d) Closing suit

2. **End keyword is added in php 7 for**
   a) To define closing block
   b) To define ending of statement
   c) To define Code Termination

d) None of Above

3. Which of the following can be used to run matched case out of given cases a) switch

b) Else if ladder

c) Switch with no default case

d) Nested if else

4. Predict the output of the following

```
$code = false; die(‘ This code is

samplecode’); If($code):

        echo ‘else will execute’;

else:

        echo ‘if will execute’;

endif;
```

a) If will execute

b) else will execute

c) neither if nor else will execute

d) Error in code

5. Understand the requirement? Suppose you are working in a Good Software company and your boss asked you to insert data of 5000 registered employee in database what will do you in such case

a) Use insert query and insert data one by one

b) Use decision control to insert data repeatedly

c) Use loops control to insert the data at single go

d) Leave the job

6. What will be the output of the following PHP code?

```
<?php
$colors = array("red","green","blue","yellow");  foreach
($colors as $value)
{
   echo "$value <br>";
}
?>
```

a) red green blue yellow

b) red
c) no output
d) error

7. What will be the output of the following PHP code? <?php

```
for ($x = 0; $x <= 10; $x++)
{
   echo "The number is: $x <br>";
}
?>
```

a) The number is : 0 The number is : 1
The number is : 2
The number is : 3
The number is : 4
The number is : 5
The number is : 6
The number is : 7
The number is : 8
The number is : 9
b) The number is: 0
c) no output
d) error

8. What will be the output of the following PHP code?

```
<?php
for ($x = 0; $x <= 10; print ++$x)
{
   print ++$x;
} ?>
```

a) 123456789101112
b) 12345678910
c) 1234567891011
d) infinite loop

9. What will be the output of the following PHP code?

```
<?php
for ($x = 1; $x < 10;++$x)
{          print
"*\t";
}
?>
```

a) **********
b) *********

c) **********
d) infinite loop

10. What will be the output of the following PHP code?

```
<?php
    for ($x = -1; $x < 10;--$x)
    {       print
    $x;
    } ?>
```

a) 123456789101112
b) 12345678910
c) 1234567891011
d) infinite loop

11. What will be the output of the following PHP code?

```
<?php $x;
for ($x = -3; $x < -5; ++$x)
{
   print ++$x;
} ?>
```

a) -3-4-5
b) -3-4
c) infinite loop
d) no output

12. What will be the output of the following PHP code?

```
<?php
for ($i++; $i == 1; $i = 2)
print "In for loop ";
print "After loop\n";

?>
```

a) In for loop
b) After for loop
c) In for loopAfter for loop
d) Infinite loop

13. What will be the output of the following PHP code?

```
<?php
```

```
for (1; $i == 1; $i = 2)
print "In for loop "; print
"After loop\n";
?>
```

a) In for loop
b) After for loop
c) In for loopAfter for loop
d) Infinite loop

14. What will be the output of the following PHP code?

```
<?php
for ($i == 2; ++$i == $i; ++$i)
   print "In for loop "; print
"After loop\n";
?>
```

a) In for loop in for loop in for loop in for loop …….infinitely
b) After for loop After for loop After for loop……..infinitely
c) In for loop After for loop In for loop After for loop In for loop After for loop…..infinitely d) After for loop

15. Which of the following keyword is use to terminate the loop a) exit
b) die()
c) Break
d) continue

16. Which of the following loop are used to print 2 dimensional pattern of order 2x2 a) for
b) nested for
c) while
d) nested if

17. Which of the following keyword is used to by pass the block of code

a) break
b) continue
c) default
d) else

18. Out of three condition in for loop which parameter position cannot be changed a) i=0

b) i++
c) i<=9
d) echo i

19. True or false can body of the loop be terminated or not

a) True
b) False

20. Among three loops which loop is the loop which runs minimum once

a) while
b) for
c) do-while
d) for each

21. in for each which operator is optional

a) $ Doller
b) => Arrow Operator
c) -> Object Operator
d) : Suit Operator

22. **For each loop can be used to traverse which type resources in php**

a) Associative arrays
b) numerical Arrays
c) Both Arrays
d) variables

23. **In which of the following case break is not used for terminating**

a) switch
b) while loops
c) for loops
d) do-while loops

24. If i=1 and i <=9 and i++ how many times will for loop execute

a) 9 times
b) 10 times
c) 8 times
d) None of the above

25. Which loops is used when no of iteration are not known

a) do while loop

b) foreach loop
c) while loop
d) nested loops

Answer Key: -

1. c	2. a	3. a	4. c	5. c
6. a	7. a	8. a	9. b	10. d
11. d	12. c	13. b	14. a	15. c
16. b	17. b	18. c	19. a	20. c
21. b	22. c	23. d	24. b	25. c

PHP Lecture – 4 (Functions & Error Handling)

Long Answer Questions: -

Q1. What is Function and How to create a function in PHP?

Ans.) PHP functions are similar to other programming languages. A function is a piece of code which takes one more input in the form of parameter and does some processing and returns a value. You already have seen many functions like **array()** and **string()** etc. They are built-in functions but PHP gives you option to create your own functions as well.

There are two parts which should be clear to you:

- Creating a PHP Function (Called Function)
- Calling a PHP Function (Calling Function)

In fact, you hardly need to create your own PHP function because there are already more than 1000 built-in library functions created for different areas and you need to call them according to your requirements.

Creating PHP Function:
Its very easy to create your own PHP function. Suppose you want to create a PHP function that will simply write a simple message on your browser when you call it. Following example creates a function called writeMessage() and then calls it just after creating it.

Note that while creating a function its name should start with keyword **function** and all the PHP code should be put inside { and } braces as shown in the following example below:

Example – FunSimple.php

```
<html>
<head>
<title>Function Example</title>
</head>
<body>
<h1>Simple Function Example</h1>
<?php
function simple() //Function Definition
{
    echo "Welcome to php";
}
simple();//Function calling
?>
</body>
</html>
```

Q2. What do you mean by Parameterized Function?

Ans.) A Parameterized functions are functions with parameters'. These functions allow us to pass one or more parameters inside the function.

We can pass the information in function through arguments (or) parameters which are separated by comma (,).

These passed arguments (or) parameters acts as variables inside the function. **Example –**

```
<html>
<head>
<title>Function Example</title>
</head>
<body>
<h1>Parameterized Function Example</h1>
<?php
function add($a,$b) //function defn
{
    $sum = $a + $b;
    echo "Sum is :$sum <br/>";
}
add(23,34);//calling function

function sub($a,$b) //function defn
{
    $sub = $a - $b;
echo "Sub is :$sub";
}
sub(63,34);//calling function
?>
</body>
</html>
```

Q3. Define Call by value & Call by reference and What is difference between them?

Ans.) Call by Value - In case of call by value, the actual value is not modified even if it is modified inside the function.

Example –

```
<html>
<head>
<title>Call by Value</title>
</head>
<body>
```

```
<h1>Call by Value Demo</h1>
<?php //Example1
echo "Example1 : ";
function adder($str)
{
    $str .= 'Call By Value';
}
$str = 'Hello ';
adder($str);
echo $str;  echo
"<br/>";

//Example2 echo
"Example2 : ";
function increment($i)
{
    $i++;
}   $i = 10;
increment($i);
echo $i;  ?>
</body>
</html>
```

Call by Reference - In case of call by reference, the actual value is modified if it is modified inside the function.

In this case, we need to use **& (ampersand)** symbol with formal arguments or parameters. And **&** (ampersand) symbol represents reference of a variable.

```
<html>
<head>
<title>Call by Reference</title>
</head>
<body>
<h1>Call by Reference Demo</h1>
<?php //Example1 echo
"Example1 : ";
function adder(&$str2)
{
    $str2 .= 'Call By Reference';
}
$str = 'Hello ';   adder($str);
```

```
echo $str;  echo
"<br/>";

//Example2 echo
"Example2 : "; function
increment(&$i)
{
    $i++;
}   $i = 10;
increment($i);
echo $i;   ?>
</body>
</html>
```

Parameters	Call by value	Call by reference
Definition	While calling a function, when you pass values by copying variables, it is known as “Call By Values.”	While calling a function, in programming language instead of copying the values of variables, the address of the variables is used it is known as “Call By References.
Arguments	In this method, a copy of the variable is passed.	In this method, a variable itself is passed.
Effect	Changes made in a copy of variable never modify the value of variable outside the function.	Change in the variable also affects the value of the variable outside the function.
Alteration of value	Does not allow you to make any changes in the actual variables.	Allows you to make changes in the values of variables by using function calls.
Passing of variable	Values of variables are passed using a straightforward method.	Pointer variables are required to store the address of variables.
Value modification	Original value not modified.	The original value is modified.
Memory Location	Actual and formal arguments will be created in different memory location	Actual and formal arguments will be created in the same memory location
Safety	Actual arguments remain safe as they cannot be modified accidentally.	Actual arguments are not Safe. They can be accidentally modified, so you need to handle arguments operations carefully.
Default	Default in many programming languages like C++.PHP. Visual Basic NET, and C#.	It is supported by most programming languages like JAVA, but not as default.

Q4. What is Recursive Function and How to write a recursive function in PHP?

Ans.) Broadly speaking, recursion occurs when something contains, or uses, a similar version of itself. That similar version then contains or uses another similar version of itself, and so on. Sometimes this process can go on forever, such as when you hold 2 mirrors directly opposite each other, creating an infinite series of reflections. More often, though, the number of repetitions, or "depth" of the recursion, is limited by some sort of end condition.

How to write a recursive function in PHP

In general terms, a recursive function works like this:

1. The calling code calls the recursive function.
2. The function does any processing or calculations required.
3. If the base case has not yet been reached, the function calls itself to continue the recursion. This creates a new instance of the function.
4. If the base case is reached, the function just returns control to the code that called it, thereby ending the recursion.

Q4. What is a Recursive Function and How to write a recursive function in PHP?

Ans.) Broadly speaking, recursion occurs when something contains, or uses, a similar version of itself. That similar version then contains or uses another similar version of itself, and so on. Sometimes this process can go on forever, such as when you hold 2 mirrors directly opposite each other, creating an infinite series of reflections. More often, though, the number of repetitions, or "depth" of the recursion, is limited by some sort of end condition.

How to write a recursive function in PHP

In general terms, a recursive function works like this:

1. The calling code calls the recursive function.
2. The function does any processing or calculations required.
3. If the base case has not yet been reached, the function calls itself to continue the recursion. This creates a new instance of the function.
4. If the base case is reached, the function just returns control to the code that called it, thereby ending the recursion.

Short Answer Questions: -

Q1. Advantages of Call by Value

- The method doesn’t change the original variable, so it is preserving data.

- Whenever a function is called it, never affect the actual contents of the actual arguments.
- Value of actual arguments passed to the formal arguments, so any changes made in the formal argument does not affect the real cases.

Q2. Advantages of Call by Reference

- The function can change the value of the argument, which is quite useful.
- It does not create duplicate data for holding only one value which helps you to save memory space.
- In this method, there is no copy of the argument made. Therefore it is processed very fast.
- Helps you to avoid changes done by mistake
- A person reading the code never knows that the value can be modified in the function.

Q3. What is an Exception?

Ans.) Exception handling is used to change the normal flow of the code execution if a specified error (exceptional) condition occurs. This condition is called an exception.

Q4. What normally happens when an exception is triggered

Ans.)

- The current code state is saved
- The code execution will switch to a predefined (custom) exception handler function
- Depending on the situation, the handler may then resume the execution from the saved code state, terminate the script execution or continue the script from a different location in the code

Q5. Mention different Exception handling methods.

Ans.) We will show different error handling methods:

- Basic use of Exceptions
- Creating a custom exception handler
- Multiple exceptions
- Re-throwing an exception
- Setting a top level exception handler

Q6. Define Try, Throw and Catch.

Ans.) To avoid the error from the example above, we need to create the proper code to handle an exception.

Proper exception code should include:

1. **try** - A function using an exception should be in a "try" block. If the exception does not trigger, the code will continue as normal. However if the exception triggers, an exception is "thrown"
2. **throw** - This is how you trigger an exception. Each "throw" must have at least one "catch"
3. **catch** - A "catch" block retrieves an exception and creates an object containing the exception information

Q7. Rules for Exception.

Ans.)

- Code may be surrounded in a try block, to help catch potential exceptions
- Each try block or "throw" 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) in a catch block within a try block

A simple rule: If you throw something, you have to catch it.

Technical Tasks: -

1. Develop a program to find area & perimeter of rectangle.
2. Develop a program to find the simple interest of given input P=Rs.2000, R=10% & T=2Years.
3. Develop a program to find prime number using function
4. Develop a program to generate Fibonacci sequence up to n terms using Recursion.
5. Develop a program to find a Random Number from a Specified range using function.
6. Make a mini project **CURRENCY CONVERTOR.**
7. Develop a program to print “Welcome to Eucoders in Summer Training” with help of Function.
8. Develop a program to add two number using function.
9. Develop a program to find greater no. of given 3 numbers
10. Develop a program to find greatest number in two numbers using function with return value.

Interview Questions: -

1. What is the use of header() function in PHP?
2. What does isset() function?
3. Explain PHP parameterized functions. 4. Explain PHP variable length argument function
5. What are the different types of errors in PHP?

Multiple Choice Questions: -

1. **How to define a function in PHP?**
   a) function {function body}
   b) data type functionName(parameters) {function body}
   c) functionName(parameters) {function body}
   d) function functionName(parameteres){function body}

2. **Type Hinting was introduced in which version of PHP?**
   a) PHP 4
   b) PHP 5
   c) PHP 5.3
   d) PHP 6

3. **Which type of function call is used in line 8 in the following PHP code?**

```
<?php
function calc($price, $tax)
{
    $total = $price + $tax;
}
$pricetag = 15; $taxtag
= 3;
calc($pricetag, $taxtag);
?>
```

   a) Call By Value
   b) Call By Reference
   c) Default Argument Value
   d) Type Hinting

4. **What will be the output of the following PHP code?**

```
<?php
function calc($price, $tax="")
{
  $total = $price + ($price * $tax);
echo "$total";
}
calc(42);
?>
```

   a) Error

b) 0
c) 42
d) 84

5. Which of the following are valid function names?

a) function()
b) €()
c) function()
d) $function()
e) a) and b)

6. What will be the output of the following PHP code?

```
<?php
function a()
  {
    function b()
    {
      echo 'I am b';
 }
    echo 'I am a';
  }
a();
a();
  ?>
```

a) I am a
b) I am bI am a
c) Error
d) I am a Error

7. What will be the output of the following PHP code?

```
<?php    $op2 =
"blabla";
  function foo($op1)
  {
    echo        $op1;
echo $op2;
  }
  foo("hello");
  ?>
```

a) Helloblabla
b) Error
c) Hello
d) helloblablablabla

8. A function in PHP which starts with __ (double underscore) is known as

a) Magic Function
b) Inbuilt Function
c) Default Function
d) User Defined Function

9. What will be the output of the following PHP code?

```
<?php
  function foo($msg)
  {
    echo "$msg";
  }
  $var1 = "foo";
  $var1("will this work");
  ?>
```

a) Error
b) $msg
c) 0
d) Will this work

10. A function automatically converted to method as soon as

a) Signature of function is called using this.
b) signature of function is added to class
c) As soon as signature is access using data member
d) as soon as data member as initialized using $this

11. which function is used to destroy the variable

a) Iseet()
b) Unset()
c) Var_destroy()
d) __destruct()

12. Which of the operator can be used to hide the notice error

a) # operator
b) $$ operator
c) @operator

d) @@ operator

13. Which argument allows you to supply any number of arguments to functions a) Default

b) Array is argument

c) Var arg Length

d) Formal argument

Answer Key: -

1. d	2. b	3. a	4. c	5. b
6. a	7. c	8. a	9. d	10. a
11. b	12. c	13. c		

PHP Lecture – 5 (Arrays & Strings)

Long Answer Questions: -

Q1. Define Array and its Types with proper examples?

Ans.) An array is a data structure that stores one or more similar type of values in a single value. For example, if you want to store 100 numbers then instead of defining 100 variables it's easy to define an array of 100 length.

Or

Array is the collection of homogeneous (simmilar data type) as well as heterogeneous (dissimilar data type) element.

How to create a array

Array() :- By using this function we can create array variable.

Syntax :

Variable_name= array (e1,e2,e3,e4……..en);

Example:

```
$arr = array(10,20,30,40);

$arr_new = array( 10,20.5,'Eucoders','a');
```

How to access array values

Once we will create an array with some elements or values by default indexes will be assign for each and every element. The starting index of an array is zero and the last index of an array is (n – 1).

Using this index, we can read array values.

Example: WAP to find the sum of an array elements .

```
<?PHP

$arr = array(10,20,30,40);

$sum =$arr[0] + $arr[1] + $arr[2] + $arr[3]
```

```
Echo $sum;
```

There are three different kind of arrays and each array value is accessed using an ID's which is called array index.

- **Numeric array** - An array with a numeric index. Values are stored and accessed in linear fashion.
- **Associative array** - An array with strings as index. This stores element values in association with key values rather than in a strict linear index order.
- **Multidimensional array** - An array containing one or more arrays and values are accessed using multiple indices.

Q2. Define Numeric array and Associative array with program.

These arrays can store numbers, strings and any object but their index will be prepresented by numbers. By default, array index starts from zero.

Example

Following is the example showing how to create and access numeric arrays.
Here we have used **array()** function to create array. This function is explained in function reference.

```
<?PHP
/* First method to create array. */

$numbers = array( 1, 2, 3, 4, 5);

foreach( $numbers as $value )

{
echo "Value is $value <br />";
}
/* Second method to create array. */
$numbers[0] = "one";

$numbers[1] = "two";

$numbers[2] = "three";
$numbers[3] = "four";  $numbers[4] = "five";  foreach( $numbers as $value )
{  echo "Value is $value <br />";  }
?>
```

Output of Program

Value is 1
Value is 2
Value is 3
Value is 4
Value is 5 Value is one
Value is two Value is three Value is four
Value is five

Associative Arrays

The associative arrays are very similar to numeric arrays in term of functionality but they are different in terms of their index. Associative array will have their index as string so that you can establish a strong association between key and values.

To store the salaries of employees in an array, a numerically indexed array would not be the best choice. Instead, we could use the employees' names as the keys in our associative array, and the value would be their respective salary.

NOTE: Don't keep associative array inside double quote while printing otherwise it would not return any value.

```
<?PHP
/* First method to associate create array. */ $salaries = array("mohammad"
=> 2000,"qadir" => 1000,"zara" =>
500); echo "Salary of mohammad is ". $salaries['mohammad'] . "<br />"; echo
"Salary of qadir is ". $salaries['qadir']. "<br />"; echo "Salary of zara is ".
$salaries['zara'].
"<br />";
/* Second method to create array. */
$salaries['mohammad'] = "high";
$salaries['qadir'] = "medium"; $salaries['zara'] = "low"; echo "Salary of
mohammad is ". $salaries['mohammad'] . "<br />"; echo "Salary of qadir is ".
$salaries['qadir']. "<br />"; echo "Salary of zara is ". $salaries['zara']. "<br />";
?>
```

Output of the Program

Salary of mohammad is 2000
Salary of qadir is 1000

Salary of zara is 500
Salary of mohammad is high Salary of qadir is medium Salary of zara is low

Q3. What is Multidimensional Array with Example?

Ans.) Multidimensional Arrays

A multi-dimensional array each element in the main array can also be an array. And each element in the sub-array can be an array, and so on. Values in the multi-dimensional array are accessed using multiple index.

Example

In this example we create a two-dimensional array to store marks of three students in three subjects:
This example is an associative array, you can create numeric array in the same fashion.

```
<?PHP
 $marks = array(
"mohammad" => array

 ("physics" => 35,
"maths" => 30,
 "chemistry" => 39
),
 "qadir" => array
(
 "physics" => 30,
"maths" => 32,
 "chemistry" => 29
),
 "zara" => array
(

 "physics" => 31,"maths" => 22,
"chemistry" => 39
 ));
   /* Accessing multi-dimensional array values */ echo "Marks for
mohammad in physics : " ; echo $marks['mohammad']['physics'] .
   "<br />"; echo "Marks for qadir in maths : ";  echo $marks['qadir']['maths'] .
"<br />"; echo "Marks for zara in
chemistry : " ; echo $marks['zara']['chemistry'] . "<br />";
 ?>
```

Q4. Mention few Arrays Function with description.

Ans.)

Function	Description
array()	Creates an array
array_change_key_case()	Changes all keys in an array to lowercase or uppercase
array_chunk()	Splits an array into chunks of arrays
array_column()	Returns the values from a single column in the input array
array_combine()	Creates an array by using the elements from one "keys" array and one "values" array
array_count_values()	Counts all the values of an array
array_diff()	Compare arrays, and returns the differences (compare values only)
array_diff_assoc()	Compare arrays, and returns the differences (compare keys and values)
array_diff_key()	Compare arrays, and returns the differences (compare keys only)
array_diff_uassoc()	Compare arrays, and returns the differences (compare keys and values, using a user-defined key comparison function)
array_diff_ukey()	Compare arrays, and returns the differences (compare keys only, using a user-defined key comparison function)
array_fill()	Fills an array with values
array_fill_keys()	Fills an array with values, specifying keys

array_filter()	Filters the values of an array using a callback function
array_flip()	Flips/Exchanges all keys with their associated values in an array
array_intersect()	Compare arrays, and returns the matches (compare values only)
array_intersect_assoc()	Compare arrays and returns the matches (compare keys and values)

array_intersect_key()	Compare arrays, and returns the matches (compare keys only)
array_intersect_uassoc()	Compare arrays, and returns the matches (compare keys and values, using a user-defined key comparison function)
array_intersect_ukey()	Compare arrays, and returns the matches (compare keys only, using a user-defined key comparison function)
array_key_exists()	Checks if the specified key exists in the array
array_keys()	Returns all the keys of an array
array_map()	Sends each value of an array to a user-made function, which returns new values
array_merge()	Merges one or more arrays into one array
array_merge_recursive()	Merges one or more arrays into one array recursively
array_multisort()	Sorts multiple or multi-dimensional arrays
array_pad()	Inserts a specified number of items, with a specified value, to an array
array_pop()	Deletes the last element of an array

array_product()	Calculates the product of the values in an array
array_push()	Inserts one or more elements to the end of an array
array_rand()	Returns one or more random keys from an array
array_reduce()	Returns an array as a string, using a user-defined function
array_replace()	Replaces the values of the first array with the values from following arrays
array_replace_recursive()	Replaces the values of the first array with the values from following arrays recursively
array_reverse()	Returns an array in the reverse order
array_search()	Searches an array for a given value and returns the key

array_shift()	Removes the first element from an array, and returns the value of the removed element
array_slice()	Returns selected parts of an array
array_splice()	Removes and replaces specified elements of an array
array_sum()	Returns the sum of the values in an array
array_udiff()	Compare arrays, and returns the differences (compare values only, using a user-defined key comparison function)
array_udiff_assoc()	Compare arrays, and returns the differences (compare keys and values, using a built-in function to compare the keys and a userdefined function to compare the values)

array_udiff_uassoc()	Compare arrays, and returns the differences (compare keys and values, using two user-defined key comparison functions)
array_uintersect()	Compare arrays, and returns the matches (compare values only, using a user-defined key comparison function)
array_uintersect_assoc()	Compare arrays, and returns the matches (compare keys and values, using a built-in function to compare the keys and a userdefined function to compare the values)
array_uintersect_uassoc()	Compare arrays, and returns the matches (compare keys and values, using two user-defined key comparison functions)
array_unique()	Removes duplicate values from an array
array_unshift()	Adds one or more elements to the beginning of an array
array_values()	Returns all the values of an array
array_walk()	Applies a user function to every member of an array
array_walk_recursive()	Applies a user function recursively to every member of an array
arsort()	Sorts an associative array in descending order, according to the value
asort()	Sorts an associative array in ascending order, according to the value

compact()	Create array containing variables and their values
count()	Returns the number of elements in an array
current()	Returns the current element in an array
each()	Returns the current key and value pair from an array

end()	Sets the internal pointer of an array to its last element
extract()	Imports variables into the current symbol table from an array
in_array()	Checks if a specified value exists in an array
key()	Fetches a key from an array
krsort()	Sorts an associative array in descending order, according to the key
ksort()	Sorts an associative array in ascending order, according to the key
list()	Assigns variables as if they were an array
natcasesort()	Sorts an array using a case insensitive "natural order" algorithm
natsort()	Sorts an array using a "natural order" algorithm
next()	Advance the internal array pointer of an array
pos()	Alias of current()
prev()	Rewinds the internal array pointer
range()	Creates an array containing a range of elements
reset()	Sets the internal pointer of an array to its first element
rsort()	Sorts an indexed array in descending order
shuffle()	Shuffles an array
sizeof()	Alias of count()
sort()	Sorts an indexed array in ascending order
uasort()	Sorts an array by values using a user-defined comparison function
uksort()	Sorts an array by keys using a user-defined comparison function
usort()	Sorts an array using a user-defined comparison function

Q5. Define Strings with proper example.

Ans.) They are sequences of characters, like "PHP supports string operations". Following are valid examples of string.

```
$string_1 = "This is a string in double quotes";

$string_2 = "This is a somewhat longer, singly quoted string"; $string_39 = "This string has

thirty-nine characters"; $string_0 = ""; // a string with zero characters
```

Singly quoted strings are treated almost literally, whereas doubly quoted strings replace variables with their values as well as specially interpreting certain character sequences.

```
<?
$variable = "name";
$literally = 'My $variable will not print!\\n'; print($literally);
$literally = "My $variable will print!\\n"; print($literally);
?>
```

Output of the Program

```
My $variable will not print!\n  My name will print
```

There are no artificial limits on string length - within the bounds of available memory, you ought to be able to make arbitrarily long strings.
Strings that are delimited by double quotes (as in "this") are preprocessed in both the following two ways by PHP:

- Certain character sequences beginning with backslash (\) are replaced with special characters.
- Variable names (starting with $) are replaced with string representations of their values.

The escape-sequence replacements are:

- \n is replaced by the newline character
- \r is replaced by the carriage-return character
- \t is replaced by the tab character
- \$ is replaced by the dollar sign itself ($)
- \" is replaced by a single double-quote (")
- \\ is replaced by a single backslash (\)

Q6. Define Some String Functions and Description.

Ans.) The PHP string functions are part of the PHP core. No installation is required to use these functions.

Function	Description
addcslashes()	Returns a string with backslashes in front of the specified characters
addslashes()	Returns a string with backslashes in front of predefined characters
bin2hex()	Converts a string of ASCII characters to hexadecimal values
chop()	Removes whitespace or other characters from the right end of a string
chr()	Returns a character from a specified ASCII value
chunk_split()	Splits a string into a series of smaller parts
convert_cyr_string()	Converts a string from one Cyrillic character-set to another
convert_uudecode()	Decodes a uuencoded string
convert_uuencode()	Encodes a string using the uuencode algorithm
count_chars()	Returns information about characters used in a string
crc32()	Calculates a 32-bit CRC for a string
crypt()	One-way string hashing
echo()	Outputs one or more strings
explode()	Breaks a string into an array

fprintf()	Writes a formatted string to a specified output stream
get_html_translation_table()	Returns the translation table used by htmlspecialchars() and htmlentities()

hebrev()	Converts Hebrew text to visual text
hebrevc()	Converts Hebrew text to visual text and new lines (\n) into
hex2bin()	Converts a string of hexadecimal values to ASCII characters
html_entity_decode()	Converts HTML entities to characters
htmlentities()	Converts characters to HTML entities
htmlspecialchars_decode()	Converts some predefined HTML entities to characters
htmlspecialchars()	Converts some predefined characters to HTML entities
implode()	Returns a string from the elements of an array
join()	Alias of implode()
lcfirst()	Converts the first character of a string to lowercase
levenshtein()	Returns the Levenshtein distance between two strings
localeconv()	Returns locale numeric and monetary formatting information
ltrim()	Removes whitespace or other characters from the left side of a string

md5()	Calculates the MD5 hash of a string
md5_file()	Calculates the MD5 hash of a file
metaphone()	Calculates the metaphone key of a string
money_format()	Returns a string formatted as a currency string
nl_langinfo()	Returns specific local information
nl2br()	Inserts HTML line breaks in front of each newline in a string
number_format()	Formats a number with grouped thousands
ord()	Returns the ASCII value of the first character of a string
parse_str()	Parses a query string into variables

print()	Outputs one or more strings
printf()	Outputs a formatted string
quoted_printable_decode()	Converts a quoted-printable string to an 8-bit string
quoted_printable_encode()	Converts an 8-bit string to a quoted printable string
quotemeta()	Quotes meta characters
rtrim()	Removes whitespace or other characters from the right side of a string
setlocale()	Sets locale information

sha1()	Calculates the SHA-1 hash of a string
sha1_file()	Calculates the SHA-1 hash of a file
similar_text()	Calculates the similarity between two strings
soundex()	Calculates the soundex key of a string
sprintf()	Writes a formatted string to a variable
sscanf()	Parses input from a string according to a format
str_getcsv()	Parses a CSV string into an array
str_ireplace()	Replaces some characters in a string (case-insensitive)
str_pad()	Pads a string to a new length
str_repeat()	Repeats a string a specified number of times
str_replace()	Replaces some characters in a string (case-sensitive)
str_rot13()	Performs the ROT13 encoding on a string
str_shuffle()	Randomly shuffles all characters in a string
str_split()	Splits a string into an array
str_word_count()	Count the number of words in a string
strcasecmp()	Compares two strings (case-insensitive)

strchr()	Finds the first occurrence of a string inside another string (alias of strstr())
strcmp()	Compares two strings (case-sensitive)
strcoll()	Compares two strings (locale based string comparison)
strcspn()	Returns the number of characters found in a string before any part of some specified characters are found
strip_tags()	Strips HTML and PHP tags from a string
stripcslashes()	Unquotes a string quoted with addcslashes()
stripslashes()	Unquotes a string quoted with addslashes()
stripos()	Returns the position of the first occurrence of a string inside another string (case-insensitive)
stristr()	Finds the first occurrence of a string inside another string (case-insensitive)
strlen()	Returns the length of a string
strnatcasecmp()	Compares two strings using a "natural order" algorithm (caseinsensitive)
strnatcmp()	Compares two strings using a "natural order" algorithm (casesensitive)
strncasecmp()	String comparison of the first n characters (case-insensitive)
strncmp()	String comparison of the first n characters (case-sensitive)

strpbrk()	Searches a string for any of a set of characters
strpos()	Returns the position of the first occurrence of a string inside another string (case-sensitive)
strrchr()	Finds the last occurrence of a string inside another string
strrev()	Reverses a string
strripos()	Finds the position of the last occurrence of a string inside another string (case-insensitive)
strrpos()	Finds the position of the last occurrence of a string inside another string (case-sensitive)
strspn()	Returns the number of characters found in a string that contains only characters from a specified charlist
strstr()	Finds the first occurrence of a string inside another string (case-sensitive)
strtok()	Splits a string into smaller strings
strtolower()	Converts a string to lowercase letters
strtoupper()	Converts a string to uppercase letters
strtr()	Translates certain characters in a string
substr()	Returns a part of a string
substr_compare()	Compares two strings from a specified start position (binary safe and optionally case-sensitive)

substr_count()	Counts the number of times a substring occurs in a string
substr_replace()	Replaces a part of a string with another string
trim()	Removes whitespace or other characters from both sides of a string
ucfirst()	Converts the first character of a string to uppercase
ucwords()	Converts the first character of each word in a string to uppercase
vfprintf()	Writes a formatted string to a specified output stream
vprintf()	Outputs a formatted string
vsprintf()	Writes a formatted string to a variable
wordwrap()	Wraps a string to a given number of characters

Short Answer Questions: -

Q1. What is an Array?

Ans.) An array is a special variable, which can hold more than one value at a time. In other words, an array can hold many values under single name, and we can access the values by referring to an index.

Q2. Difference between arrays in PHP and other programming language.

Ans.) In PHP array is nothing but heterogeneous element (we can store any type of values), where as in another programming languages just like (C, C++, Java) arrays in nothing but homogeneous elements (same group of elements we can store but not other type.)

Q3. What is String?

Ans.) In PHP, A String is a sequence of characters which are enclosed with single quotes or double quotes.

There are 2 ways to specify string

- Single Quotes Ex: $str=' Hello PHP ';
- Double Quotes Ex: $str=" Hello PHP ";

Q4. What is the difference between single quoted string and double quoted string? Ans.) Singly quoted strings are treated almost literally, whereas doubly quoted strings replace variables with their values as well as specially interpreting certain character sequences. For example –

```
<?php
$variable = "name";
 $statement = 'My $variable will not print!n';
print($statement);  print " ;"
$statement = "My $variable will print!n"
 print($statement);
?>
```

Output of the Program

```
My $variable will not print!

My name will print
```

Q5. How to concatenate two strings in PHP?

Ans.) To concatenate two string variables together, we use the **dot (.)** operator.

```
<?php $string1="Hello softpro"; $string2="123"; echo $string1 . " "
. $string2; ?>
```

Technical Tasks: -

1. Write a PHP script to sort the following associative array: array("Sophia"=>"31","Jacob"=>"41","William"=>"39","Ramesh"=>"40") in a. Ascending order sort by value
   b. Ascending order sort by key
   c. Descending order sort by value
   d. Descending order sort by key
2. Search a no in the array.
3. Sum and average of a numeric array.
4. Develop a web page using PHP to check whether two strings are equal or not.
5. Develop a PHP script to check the strength of given Password.
6. Develop a PHP script which should take two strings and display the percentage of matched characters.

7. Write a program to enter n no. in an array and calculate the addition and average of the array and display the calculation.
8. Develop a web page using PHP to calculate highest no. of an array.
9. Develop a web page using PHP to calculate lowest no. of an array.
10. Develop a web page using PHP to take username as input and display username in Uppercase and Lowercase Letters.
11. Develop a web page using PHP to copy one string to another.

Interview Questions: -

1. What is the array in PHP?
2. How many types of array are there in PHP?
3. Explain some of the PHP array functions?
4. What is the difference between indexed and associative array?
5. How to get the length of string?
6. Explain some of the PHP string functions?
7. What are the methods to submit form in PHP?
8. How can you submit a form without a submit button?

Multiple Choice Questions: -

1. PHP's numerically indexed array begin with position

a) 1
b) 2
c) 0
d) -1

2. Which of the following are correct ways of creating an array?

a) state[0] = "karnataka";
b) $state[] = array("karnataka");
c) $state[0] = "karnataka";
d) $state = array("karnataka");
e) c) and d)

3. What will be the output of the following PHP code?

```
<?php
 $states = array("Karnataka" => array
 ("population" => "11,35,000", "capital" => "Bangalore"),
"Tamil Nadu" => array( "population" => "17,90,000",
"capital" => "Chennai") );
```

```
echo $states["Karnataka"]["population"];
 ?>
```

a) Karnataka 11,35,000
b) 11,35,000
c) population 11,35,000
d) Karnataka population

4. Which of the following PHP function will return true if a variable is an array or false if it is not an array?

a) this_array()
b) is_array()
c) do_array()
d) in_array()

5. Which in-built function will add a value to the end of an array?

a) array_unshift()
b) into_array()
c) inend_array()
d) array_push()

6. What will be the output of the following PHP code?

```
<?php
$state = array ("Karnataka", "Goa", "Tamil Nadu",
"Andhra Pradesh");
echo (array_search ("Tamil Nadu", $state) );
?>
```

a) True
b) 1
c) False
d) 2

7. What will be the output of the following PHP code?

```
<?php
$fruits = array ("apple", "orange", "banana");
echo (next($fruits));     echo (next($fruits));
?>
```

a) Orangebanana
b) Appleorange
c) Orangeorange
d) Appleapple

8. Which of the following function is used to get the value of the previous element in an array?

a) last()
b) before()
c) prev()
d) previous()

9. What will be the output of the following PHP code?

```
<?php
$fruits = array ("apple", "orange", array ("pear", "mango"),
"banana");
echo (count($fruits, 1));
?>
```

a) 3
b) 4
c) 5
d) 6

10. Which function returns an array consisting of associative key/value pairs? a) count()
b) array_count()
c) array_count_values()
d) count_values()

11. Which function in array are used to break the larger array into chunks

a) array_chunk()
b) array_split()
c) array_break
d) array_divide()

12. Which function is used to break and convert case of key not value

a) array_key_upper()
b) array_key_case()
c) array_change_key_case()
d) array_key_case_Change()

13. Which unique symbol is used represent the day of the week

a) d
b) m
c) y
d) I

14. Which function is used to change the data and time of time zone

a) set_date_time_zone()
b) date_set_timezone()
c) timezone_set()
d) date default timezone function set()

15. Date by default is in built class and Date and time is predefined object hence we can make object of class as $date = new Date() then which function is used to create your own date

a) set_date()
b) create_Date()
c) date_create()
d) date_set()

16. In php how many ways you can create String Literal

a) 1
b) 2
c) 3
d) 4

17. True or false in php every character sequence from 0 to 256-character set can be included in string literal

a) True
b) False

18. This special symbol is used to allow the single quotation used in single quotes a) \
b) \\
c) @
d) “”

19. Variable declared using $ will be processed in which type of string

a) Single String
b) Double String
c) Here doc string
d) New here doc string

20. <<< symbol is create which type of string

a) Here Doc
b) New doc
c) Both

d) None of the these

21. Which function is used break the large String into array of character by a unique symbol

a) Explode()
b) Implode
c) Sub_str
d) Str_split()

22. Ucfirst function is used

a) Making first character uppercase
b) Every character uppercase
c) Every space separated symbol uppercase
d) First word upper case

23. The only function used to get length or size of the string is

a) Count()
b) Str_size
c) Str_Len()
d) Len()

24. The only function of string used to generated 32-64 character hashcode generally used for encrypting

Password and cannot be decrypted

a) Md5()
b) Bcrypt()
c) Crypt()
d) Base64_encode()

25. The only function that can be used to get the ascii character value a) Char()
b) Getchar()
c) Ord()
d) None of the above

26. Which function used to count number words separated by space in long String a) Space_count()
b) Str_space_count
c) Str_word_count()
d) Word_count()

27. What is meaning of string interpolation

a) Using {} brackets inside doubles quotes
b) Using {} to show user that $variable inside double quotes will be parsed
c) Its just a convention
d) None of the above

Answer Key: -

1. c	2. e	3. b	4.	5. d	6. d
7. a	8. c	9. d	10. c	11. a	12. c
13. d	14. c	15. c	16. d	17. a	18. a
19. a	20. a	21. a	22. a	23. c	24. a
25. c	26. c	27. b			

PHP Lecture – 6 (Object Oriented Programming In PHP)

Long Answer Questions: -

Q1. What is Object Oriented Programming. Define Class and Objects with Examples?

Ans.) Object-Oriented Programming (OOP) is a programming model that is based on the concept of classes and objects. As opposed to procedural programming where the focus is on writing procedures or functions that perform operations on the data, in object-oriented programming the focus is on the creations of objects which contain both data and functions together. Object-oriented programming has several advantages over conventional or procedural style of programming.

The most important ones are listed below:

- It provides a clear modular structure for the programs.
- It helps you adhere to the "don't repeat yourself" (DRY) principle, and thus make your
- Code much easier to maintain, modify and debug.
- It makes it possible to create more complicated behavior with less code and shorter
- Development time and high degree of reusability.
- The following sections will describe how classes and objects work in PHP.

Understanding Classes and Objects

Classes and objects are the two main aspects of object-oriented programming. A class is a self-contained, independent collection of variables and functions which work together to perform one or more specific tasks, while objects are individual instances of a class.

A class acts as a template or blueprint from which lots of individual objects can be created. When individual objects are created, they inherit the same generic properties and behaviors, although each object may have different values for certain properties.

For example, think of a class as a blueprint for a house. The blueprint itself is not a house, but is a detailed plan of the house. While, an object is like an actual house built according to that blueprint. We can build several identical houses from the same blueprint, but each house may have different paints, interiors and families inside, as shown in the illustration below.

Blueprint *Houses built according to the blueprint*

A class can be declared using the *class* keyword, followed by the name of the class and a pair of curly braces **({})**, as shown in the following example.
Let's create a PHP file named Rectangle. PHP and put the following example code inside it so that our class code should be separated from rest of the program. We can then use it wherever it's needed by simply including the Rectangle. PHP file.

Example:

```
<?PHP
class Rectangle
{
  //  Declare  properties
public $length = 0;
  public $width = 0;

  // Method to get the perimeter
public function getPerimeter(){
    return (2 * ($this->length + $this->width));
  }
  // Method to get the area
public function getArea(){
    return ($this->length * $this->width);
  }
}
?>
```

The `public` keyword before the properties and methods in the example above, is an **access modifier**, which indicates that this property or method is accessible from anywhere. We will learn more about this a little later in this chapter.

Note: Syntactically, variables within a class are called *properties*, whereas functions are called *methods*. Also class names conventionally are written in PascalCase i.e. each concatenated word starts with an uppercase letter (e.g. MyClass).

Short Answer Questions: -

Q1. What is the main difference between asp net and PHP?

Ans.) PHP is a **programming language** whereas **ASP.NET** is a Programming **framework**. Websites developed by ASP.NET may use C#, but also other languages such as J#. ASP.NET is compiled whereas PHP is interpreted. ASP.NET is designed for windows machines, whereas PHP is platform free and typically runs on Linux servers.

Q2. How can you compare objects in PHP?

Ans.) We use the operator '==' to test if two objects are **instanced** from the same class and have same attributes and equal values. We can also test if two objects are referring to the same instance of the same class by the use of the identity operator '==='.

Q3. What are constructor and destructor in PHP?

Ans.) PHP constructor and destructor are special type functions which are automatically called when a PHP **class object** is **created** and **destroyed**. The constructor is the most useful of the two because it allows you to send parameters along when creating a new object, which can then be used to initialize variables on the object.

Q4. What is the relation between Classes and Objects?

Ans.) They look very much same but are not same.

1. A class is a definition, while an object is an instance of the class.
2. A class is a blueprint while objects are actual objects existing in the real world.

Suppose we have a class Person which has attributes and methods like name, age, height, weight, color etc.

Class Person is just a prototype, now we can create real-time objects of class Person.

#Example: Ajay is real time object of class **Person**, which have name=Ajay, age=23, height=170cm, weight=60kg and color=black etc.

Q5. What is Member Variable and Member function?

Ans.)

Member Variable − These are the variables defined inside a class. This data will be invisible to the outside of the class and can be accessed via member functions. These variables are called attribute of the object once an object is created.

Member function − These are the function defined inside a class and are used to access object data.

Q6. **What is different types of Visibility? OR What are access modifiers?**

Ans.) Each method and property has its visibility. There are three types of visibility in PHP.

Types of visibility:

1. **public:** Public method or variable can be accessible from anywhere, Means a public method or variable of a class can be called outside of the class or in a subclass.
2. **protected:** A protected method or variable can only be called in that class & it's subclass.
3. **private:** A private method or variable of a class can only be called inside that class only in which it is declared.

NOTE: By default, in PHP, a class member is public unless declared private or protected.

Technical Tasks: -

1. Write a short note on Class and objects
2. Difference between structured programming technique and Modular Programming.
3. Develop a program to create a class USER and print the value with the help of object.
4. Develop a program to find the area of the circle using class & Object.
5. Make a web page to convertor (1km to 1000m) using class and objects.
6. Develop a registration page to insert the values using class and object.
7. Write PHP program to show that private member of a super class Cannot be accessed from derived classes.

Interview Questions: -

1. **What is meant by the term OOPs?**

OOPs refers to Object-Oriented Programming. It is the programming paradigm that is defined using objects. Objects can be considered as real-world instances of entities like class, that have some characteristics and behaviors.

2. **What is the need for OOPs?**

There are many reasons why OOPs is mostly preferred, but the most important among them are:

- OOPs helps users to understand the software easily, although they don't know the actual implementation.
- With OOPs, the readability, understandability, and maintainability of the code increase multifold.
- Even very big software can be easily written and managed easily using OOPs

3. **What are some major Object Oriented Programming languages?**

The programming languages that use and follow the Object-Oriented Programming paradigm or OOPs, are known as Object-Oriented Programming languages. Some of the major ObjectOriented Programming languages include:

- Java
- C++
- Javascript
- Python
- PHP

And many more.

6. **What are the main features of OOPs?**

OOPs or Object Oriented Programming mainly comprises of the below four features, and make sure you don't miss any of these:

- Inheritance
- Encapsulation
- Polymorphism
- Data Abstraction

7. **What are some advantages of using OOPs?**

- OOPs is very helpful in solving very complex level of problems.
- Highly complex programs can be created, handled, and maintained easily using objectoriented programming.
- OOPs, promote code reuse, thereby reducing redundancy.
- OOPs also helps to hide the unnecessary details with the help of Data Abstraction.
- OOPs, are based on a bottom-up approach, unlike the Structural programming paradigm, which uses a top-down approach.
- Polymorphism offers a lot of flexibility in OOPs

8. **Why is OOPs so popular?**

OOPs programming paradigm is considered as a better style of programming. Not only it helps in writing a complex piece of code easily, but it also allows users to handle and maintain them easily as well. Not only that, the main pillar of OOPs - Data Abstraction,

Encapsulation, Inheritance, and Polymorphism, makes it easy for programmers to solve complex scenarios. As a result of these, OOPs is so popular.

9. **What is a class?**

A class can be understood as a template or a blueprint, which contains some values, known as member data or member, and some set of rules, known as behaviors or functions. So when an object is created, it automatically takes the data and functions that are defined in the class. Therefore the class is basically a template or blueprint for objects. Also one can create as many objects as they want based on a class.

For example, first, a car's template is created. Then multiple units of car are created based on that template.

10. **What is an Object?**

An object refers to the instance of the class, which contains the instance of the members and behaviors defined in the class template. In the real world, an object is an actual entity to which a user interacts, whereas class is just the blueprint for that object. So the objects consume space and have some characteristic behavior. For example, a specific car.

Multiple Choice Questions: -

1. **Object oriented is secure way of making Software application**
   a) True
   b) False
2. **Which of the following is not used pillars in oops**
   a) Inheritance
   b) Abstraction
   c) Polymorphism
   d) Subroutimes Implementation
3. **In procedural way of coding program is divided into functions where as in case of oops**

   The program are divided into which entity
   a) Classes
   b) Objects
   c) Data member
   d) Member functions
4. **Oops follows which type of approach**
   a) Left to right
   b) Right to left
   c) Top to bottom
   d) Bottom to top

5. **Objects can called as blue print of the class is it true statement or false** a) False
   b) True
6. **Name of function which can be so internal signature or definition of class.** a) Instanceof
   b) gettype()
   c) echo ()
   d) var_dump()
7. **Where do objects get memory allocation**
   a) In Heap
   b) In Stack
   c) In Both
   d) No where
8. **The only object used to refer the current object is**
   a) $obj
   b) new keyword
   c) $this
   d) None of the above
9. **Which operator is used to check if given object is part of a class or not** a) Is
   b) Is_object
   c) $this
   d) Instanceof
10. **Data Member are just properties or state of object these are same as _____ in procedural programming** a) Functions
   b) Constant
   c) Variables
   d) Objects
11. **Which keyword is used to perform inheritance**
   a) $this
   b) Traits
   c) Extends
   d) Implements
12. **Which among of these inheritance is not supported by php**
   a) Multiple
   b) Multi-level
   c) Hierarchy
   d) Single

13. **Which of the following pillars can be used to implement security?**
   a) Encapsulation
   b) Inheritance

c) Data Abstraction
d) Polymorphism
e) 1.What is the special type of method ____

14. Which is automatically called as the class is instantiated

a) a:method
b) var object
c) class object
d) constructor

15. Which is the special prefix added before adding the constructor a) _
b) __
c) $
d) function

16. What is return of the constructor

a) void
b) bool
c) resource
d) no return

17. ____Constructor donot allow the parameter to pass

a) parameterized constructor()
b) default constructor()
c) destructor()
d) $this

18. In which of the following conditions 2nd syntaxes is important

a) in case method calling
b) in case initialising data member
c) in case of constructor calling
d) in case of using setter function

19. Using setter & getter is better than constructor _____

a) True
b) False

20. For what reason constructor is better than setter

a) because constructor can be called at the run time
b) becuase php does not support getter or setter
c) because it break the rule of encapsulation

d) none of the above

Answer Key: -

1 a	2 d	3 b	4 d	5 a
6 d	7 a	8 c	9 d	10 c
11 c	12 a	13 c	14 d	15 b
16 d	17 b	18 c	19 a	20 c

PHP Lecture – 7 (Encapsulation)

Long Answer Questions: -

Q1. What is Encapsulation?

Ans.) Wrapping up member variables and methods together into a single unit (i.e. Class) is called Encapsulation.

1. Encapsulation is used to hide the values or state of a structured data object inside a class, preventing unauthorized parties' direct access to them.
2. Visibility is the mechanism for encapsulation.

```
class Person
 {
private $name;

public function setName($name)
 {
$this->name = $name;
 } public function getName($name)

{
 return $this->name;
}
 }
$personObj = new Person();

$personObj->setName('Eucoders');
 $personObj->getName();
```

Short Answer Questions: -

Q1. What is Data hiding?

Ans.) Encapsulation is the process of hiding unwanted information, such as restricting access to any member of an object.

Q2. What is Data Binding?

Encapsulation is the process of binding the data members and the methods together as a whole, as a class.

Q3. Write down the advantages of Encapsulation.

Ans.) Here is the list of advantages

- Data Hiding and Abstraction
- Data security
- Reduces complexity
- Reusability
- Reliability
- Easier testing of code
- Increased flexibility

Technical Tasks: -

1. Write a program in PHP to update password and check the course of the student using encapsulation.

Interview Questions: -

1. **How much memory does a class occupy?**

Classes do not consume any memory. They are just a blueprint based on which objects are created. Now when objects are created, they actually initialize the class members and methods and therefore consume memory.

2. **Is it always necessary to create objects from class?**

No. An object is necessary to be created if the base class has non-static methods. But if the class has static methods, then objects don't need to be created. You can call the class method directly in this case, using the class name.

3. **What is a Constructor?**

Constructors are special methods whose name is the same as the class name. The constructors serve the special purpose of initializing the

objects. For example, suppose there is a class with the name “MyClass”, then when you instantiate this class, you pass the.

syntax:
MyClass myClassObject = new MyClass();

Multiple Choice Questions: -

1. **Which one of the following keyword is used to inherit our subclass into a superclass?** a. extends
   b. implements
   c. inherit
   d. include
2. **Which keyword is used to refer to properties or methods within the class itself?** a. Private
   b. Public
   c. Protected
   d. $this
3. **Which keyword allows class members (methods and properties) to be used without needing to instantiate a new instance of the class?**
   a. Protected
   b. Final
   c. Static
   d. Private
4. **Which of the following advanced OOP features is/are not supported by PHP?**

   i) Method overloading ii)
   Multiple Inheritance iii)
   Namespaces
   iv) Object Cloning

   a. All of the mentioned
   b. None of the mentioned
   c. i) and ii)
   d. iii) and iv)

5. **Which version of PHP introduced the advanced concepts of OOP?**
   a. PHP 4
   b. PHP 5
   c. PHP 5.3

d. PHP 6

6. Which one of the following is the right way to clone an object?

a. _clone(targetObject);
b. destinationObject = clone targetObject;
c. destinationObject = _clone(targetObject);
d. destinationObject = clone(targetObject); **Answer Key: -**

1. a	2. d	3. c	4. c	5. b	6. b

PHP Lecture – 8 (Inheritance, Polymorphism, & Constructor)

Long Answer Questions: -

Q1. Define Inheritance and its types with examples.

Ans.) Inheritance in OOP = When a class derives from another class.

The child class will inherit all the public and protected properties and methods from the parent class. In addition, it can have its own properties and methods.

An inherited class is defined by using the extends keyword.

Let's look at an example:

```php
<?php class
 Fruit {
  public $name;
   public $color;
  public function __construct($name, $color) {     $this-
 >name = $name;
    $this->color = $color;
   }   public function
intro() {
     echo "The fruit is {$this->name} and the color is {$this
>color}.";
   }
}

// Strawberry is inherited from Fruit class
 Strawberry extends Fruit {
  public function message() {
     echo "Am I a fruit or a berry? ";
  }
 }
$strawberry = new Strawberry("Strawberry", "red");
 $strawberry->message();
$strawberry->intro();
 ?>
```

Example Explained

The Strawberry class is inherited from the Fruit class.

This means that the Strawberry class can use the public $name and $color properties as well as the public __construct() and intro() methods from the Fruit class because of inheritance.
The Strawberry class also has its own method: message().

Types of PHP Inheritance

Generally, inheritance has three types, single, multiple and multi-level inheritance. But, PHP supports single inheritance and multi-level inheritance. That means the subclass will be derived from a single parent class. Even though PHP is not supporting any multiple inheritances, we can simulate it by using PHP interfaces.

In PHP, inheritance can be done by using extends keyword, meaning that, we are extending the derived class with some additional properties and methods of its parent class. The syntax for inheriting a class is as follows.

```
Child_class_name extends Parent_class_name {
...
```

Q2. Explain about Polymorphism and its types?

Ans.) It is simply "One thing, can use in different forms". Technically, it is the ability to redefine methods for derived classes.

#Example: One Class (Car) can extend two classes (Audi & BMW).

Note - Polymorphism describes a pattern in object oriented programming in which classes have different functionality while sharing a common interface.

Types of Polymorphism

Polymorphism could be static and dynamic both. Overloading is static polymorphism while, overriding is dynamic polymorphism.

1. Compile time polymorphism (Static) - Method Overloading
2. Runtime time polymorphism (Dynamic) - Method Overriding

1. Overloading is defining functions/methods that have same signatures with different parameters in the same class.
2. Overriding is redefining parent class functions/methods in child class with same signature. So, basically the purpose of overriding is to change the behavior of your parent class method.

Note - The overloading methods are invoked when interacting with properties or methods that have not been declared or are not visible in the current scope. The rest of this section will use the terms "inaccessible properties" and "inaccessible methods" to refer to this combination of declaration and visibility.

Q3. What is the key difference between concrete class and abstract class?

Ans.) Concrete classes are those classes which has to declare body of abstract methods which extends or implements from abstract class or interface

OR

Abstract classes usually have partial or no implementation. On the other hand, Concrete classes always have full implementation of its behavior. Unlike Concrete classes, Abstract classes cannot be instantiated.

Q4. What is the difference between Abstract class and Interface?

Ans.)

Abstract class	Interface
In abstract class a method must be declared as abstract. Abstract methods doesn't have any implementation.	In interface all the methods by default are abstract.
Abstract class can also contain member variables and concrete functions/methods.	Interfaces cannot contain any member variables and concrete functions/methods except constants.
An Abstract methods can be declare with access modifiers like public, protected etc. Concrete Class which is extending the abstract class must be defined with the same or visibility.	All methods declared in an interface must be public.
A class can Inherits only one Abstract class and Multiple inheritance is not possible for Abstract class.	A class can implements many interfaces and Multiple interface inheritance is possible.
Only complete member of abstract class can be static.	Memebrs of Interface cannot be static.

Q5. Can Class Properties be Abstract?

Ans.) No. there is no need for a class property to be abstract. Class properties and constants are not implemented, instead, they are declared, which in a way already makes them abstract. They follow the same rules and are treated the same way when declared in an abstract class as they would any other class.

Note: All the remaining classes which are not abstract are called **Concrete Classes**

Q6. What is the need of abstract class?

Ans.) Suppose we were modeling the behavior of animals, by creating a class hierarchy that started with a base class called Animal.

Animals are capable of doing different things like flying, digging and walking, but there are some common operations as well like eating and sleeping.

Short Answer Questions: -

1. **What is Polymorphism?**

Ans.) Polymorphism is composed of two words - “poly” which means “many”, and “morph” which means “shapes”. Therefore, Polymorphism refers to something that has many shapes.

In OOPs, Polymorphism refers to the process by which some code, data, method, or object behaves differently under different circumstances or contexts. Compile-time polymorphism and Run time polymorphism are the two types of polymorphisms in OOPs languages.

2. **What is Compile time Polymorphism and how is it different from Runtime Polymorphism?**

Ans.)

Compile Time Polymorphism: Compile time polymorphism, also known as Static Polymorphism, refers to the type of Polymorphism that happens at compile time. What it means is that the compiler decides what shape or value has to be taken by the entity in the picture.

Runtime Polymorphism: Runtime polymorphism, also known as Dynamic Polymorphism, refers to the type of Polymorphism that happens at the run time. What it means is it can't be decided by the compiler. Therefore, what shape or value has to be taken depends upon the execution. Hence the name Runtime Polymorphism.

3. **What is meant by Inheritance?**

Ans.) The term “inheritance” means “receiving some quality or behavior from a parent to an offspring.” In object-oriented programming, inheritance is the mechanism by which an object or class (referred to as a child) is created using the definition of another object or class (referred to as a parent). Inheritance not only helps to keep the implementation simpler but also helps to facilitate code reuse.

4. **What is Abstraction?**

Ans.) You only want to know how the software solves your problem. Abstraction is the method of hiding unnecessary details from the necessary ones. It is one of the main features of OOPs. For example, consider a car. You only need to know how to run a car, and not how the wires are connected inside it. This is obtained using Abstraction.

5. What is a Destructor?

Ans.) But destructors free up the resources and memory occupied by an object. Destructors are automatically called when an object is being destroyed.

6. What is an Interface?

Ans.) Only the declaration of methods is allowed inside an interface. To use an interface, you cannot create objects. Instead, you need to implement that interface and define the methods for their implementation.

7. What is meant by static polymorphism?

Ans.) Static Polymorphism is commonly known as the Compile time polymorphism. Static polymorphism is the feature by which an object is linked with the respective function or operator based on the values during the compile time. Static or Compile time Polymorphism can be achieved through Method overloading or operator overloading.

8. What is meant by dynamic polymorphism?

Ans.) Dynamic Polymorphism or Runtime polymorphism refers to the type of Polymorphism in OOPs, by which the actual implementation of the function is decided during the runtime or execution. The dynamic or runtime polymorphism can be achieved with the help of method overriding.

9. What is the difference between overloading and overriding?

Ans.) Overloading is a compile-time polymorphism feature in which an entity has multiple implementations with the same name. For example, Method overloading and Operator overloading.
Whereas Overriding, is a runtime polymorphism feature in which an entity has the same name, but its implementation changes during execution. For example, Method overriding.

10. How is data abstraction accomplished?

Ans.) Data abstraction is accomplished with the help of abstract methods or abstract classes.

11. What is an abstract class?

Ans.) An abstract class is a special class containing abstract methods. The significance of abstract class is that the abstract methods inside it are not implemented and only declared. So as a result, when a subclass inherits the abstract class and needs to use its abstract methods, they need to define and implement them.

Technical Tasks: -

1. Any integer is input by the user. Write a program to find out whether it is an odd number or even number.
2. Develop a program to create a class **USER** and print the value with the help of object.
3. Develop a program to find the area of the circle using class & Object
4. Write a PHP class that calculates the factorial of an integer
5. Write a PHP Calculator class which will accept two values as arguments, then add them, subtract them, multiply them together, or divide them on request
6. Write a program to sum of digits of given integer number.
7. Make a web page to convertor (1km to 1000m) using class and objects.
8. Develop a registration page to insert the values using class and object. Cannot be accessed from derived classes.
9. Design a class to represents a bank account. Include the following members

   Data Members

   i.) Name of the depositor ii.)

   Account Number iii.) Type of

   Account iv.) Balance Amount in

   Account

   Methods

   i.) To assign initial values ii.) To deposit an account

   iii.) To withdraw an amount after checking balance

   iv.) To display the name and balance

Interview Questions: -

7. What is Polymorphism?
8. What is Compile time Polymorphism and how is it different from Runtime Polymorphism?
9. What is meant by Inheritance?
10. What is Abstraction?
11. What is a Destructor?
12. What is an Interface?
13. What is meant by static polymorphism?
14. What is meant by dynamic polymorphism?
15. What is the difference between overloading and overriding?
16. How is data abstraction accomplished?
17. What is an abstract class?

PHP Lecture – 9 (Error & Exception)

Long Answer Questions: -

Q1. What is Error and Exception explain in brief.

Ans.)

Error - A **PHP Error** occurs when something is wrong in the PHP code. The error can be as simple as a missing semicolon, or as complex as calling an incorrect variable.

To efficiently resolve a PHP issue in a script, you must understand what kind of problem is occurring.

Exception - **Dictionary Meaning:** Exception is an abnormal condition.

In Programming world, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime.

This is what normally happens when an exception is triggered:

- The current code state is saved

- The code execution will switch to a predefined (custom) exception handler function
- Depending on the situation, the handler may then resume the execution from the saved code state, terminate the script execution or continue the script from a different location in the code

We will show different error handling methods:

- Basic use of Exceptions
- Creating a custom exception handler
- Multiple exceptions
- Re-throwing an exception
- Setting a top level exception handler

Note: Exceptions should only be used with error conditions, and should not be used to jump to another place in the code at a specified point.

Q2. What is Exception Handling with examples?

Ans.) PHP 5 has an exception model similar to that of other programming languages. Exceptions are important and provide a better control over error handling.

Let's explain the new keyword related to exceptions.

- **Try** − A function using an exception should be in a "try" block. If the exception does not trigger, the code will continue as normal. However if the exception triggers, an exception is "thrown".
- **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.

When an exception is thrown, code following the statement will not be executed, and PHP will attempt to find the first matching catch block. If an exception is not caught, a PHP Fatal Error will be issued with an "Uncaught Exception ...

- An exception can be thrown, and caught ("catched") within PHP. Code may be surrounded in a 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

Following is the piece of code, copy and paste this code into a file and verify the result.

```
```

```
<?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';
?>
```

In the above example $e->getMessage function is used to get error message. There are following functions which can be used from **Exception** class.

- **getMessage()** − message of exception
- **getCode()** − code of exception
- **getFile()** − source filename
- **getLine()** − source line
- **getTrace()** − n array of the backtrace()
- **getTraceAsString()** − formatted string of trace

Short Answer Questions: -

Q1. What are called Exceptions and why it comes.

Q2. What are the correct ways to handle an exceptions?

Q3. Short Notes on

- Try
- Catch
- Finally

Technical Tasks: -

1. Write a php code to demonstrate the divide by zero exception.
2. Write a php code to generate a user defined exception.
3. Write a php code to demonstrate multiple catch blocks for exception handling.

Interview Questions: - 1.

What is an Exception?

An exception can be considered as a special event, which is raised during the execution of a program at runtime, that brings the execution to a halt. The reason for the exception is mainly due to a position in the program, where the user wants to do something for which the program is not specified, like undesirable input.

2. **What is meant by Exception Handling?**

No one wants its software to fail or crash. Exceptions are the major reason for software failure. The exceptions can be handled in the program beforehand and prevent the execution from stopping. This is known as exception handling. So exception handling is the mechanism for identifying the undesirable states that the program can reach and specifying the desirable outcomes of such states. Try-catch is the most common method used for handling exceptions in the program.

Multiple Choice Questions: -

1. Are error and Exception same: _____

a) False
b) true

2. Which type of Error occurs due to typecasting

a) Syntax Error
b) Fatal Error
c) Warning Error
d) Logical Error

3. which Error is generated due to programmer mistake

a) syntax Error
b) Fatal Error
c) warning Error
d) Logical Error

4. This particular Error stopes the execution of the php script

a) logical Error
b) Fatal Error
c) warning Error
d) Notice Error

5. These Error are used for debugging but they donot terminate the script these Error can suppress by (Suppress Operator @)

a) warning Error
b) Logical Error
c) Fatal Error
d) Parse Error

6. in Exception handling every Exception is handled using which Block a)
try
b) catch
c) finally
d) throw

7. This particular keyword error is used to raise the error and caught by the catch exeception block. a) Try
b) Catch
c) Finally
d) throw

8. instead of using supress operator at every line we can use
a) display_error(1)
b) Error_Reporting(1)
c) Error.Reporting(0)
d) Error_Reporting(ALL)

Answer key: -

1. a	2. d	3. a	4. b	5. a
6. b	7. d	8. c		

PHP Lecture – 10 (Files & Operating System)

Long Answer Questions: -

Q1. What is File Handling in PHP?

Ans.) In PHP File System allows us to create file, read file line by line, read file character by character, write file, append file, delete file and close file.

PHP supports following functions to handle files.

- **fopen()**
- **fread()**
- **fwrite()**
- **fclose()**
- **unlink()**

fopen - In PHP, The fopen() function is used to open a file in read mode or write mode or in both read and write modes. And this function accepts two arguments: **$filename** and **$mode**.

Syntax: fopen(string $filename , string $mode)

The **$filename** represents the file to be opended and **$mode** represents the file mode for example read-only, read-write, write-only etc.

The **fopen()** supports the following modes:

r - Read Only w - Write Only a - Append r+ - Read & Write. w+ - Read & Write.

```
<?php
$filename = "E:\\studyglance\\simple.txt";
$fp = fopen($filename, "r");//open file in read mode ?>
```

Note : The above code will open the file in in read mode.

fread - In PHP, The fread() function is used to read data of the file. It requires two arguments: **file resource** and **file size**.

Syntax: fread(resource $fp,int $length)

Where **$fp** represents file pointer that is created by fopen() function. And **$length** represents length of file (bytes)to be read.

```
<?php
$filename = "D:\\studyglance\\file.txt";
$fp = fopen($filename, "r");//open file in read mode
$content = fread($fp, filesize($filename));//read file
echo "$content";//printing data of file
fclose($fp);//close file     ?>
```

Note : To read content of file, we need to open file with **r , r+, w+** mode.

fwrite - In PHP, The fwrite() function is used to write the content (string) into file.And It requires two arguments: **file resource** and **data(string)**

If the file opened in write(**w**) mode, the fwrite() function will erase the previous data of the file and writes the new data or If the file opened in append(**a**) mode, the fwrite() function appends the new data at the end of the previous data.

Syntax: fwrite(resource $fp,string $data)

Where **$fp** represents file pointer that is created by fopen() function.And **$data** represents the data to be written.

If the file opened in **write(w) mode**:

```
<?php
$filename = "D:\\studyglance\\file.txt";
$fp = fopen($filename, "w");//open file in write mode
fwrite($fp,'PHP is a Scripting Language'); echo "Data
written successfully.."; fclose($fp);//close file
?>
```

fclose - In PHP, The fclose() function is used to close an open file.And It requires one argument i.e. "**File Pointer**"

Syntax: fclose(resource $fp)

Where **$fp** represents file pointer that is created by fopen() function.

```php
<?php
$filename = "D:\\studyglance\\file.txt";
$fp = fopen($filename, "r");//open file in write mode
// To close the file  fclose($fp);
?>
```

unlink - In PHP, The unlink() function is used to delete any file.And It requires one argument i.e. "**File Name**"

Syntax: unlink(string $filename)

Where **$filename** represents name of the file to be remove.

```php
<?php
$status=unlink('file.txt'); if($status){
echo "File deleted successfully";
}else{ echo "Sorry!";
}
?>
```

Technical Tasks: -

1. Write a php code to store information in to a file.
2. Develop a program to create a .doc file with the help of file handling.
3. Make a program to make a notice on text file using file handling.
4. Develop a program to read a .doc file and write on it.
5. Write a php code to read the contents of a file convert information into uppercase and display it on web page.
6. Write a php code to copy contents of one file into another file.
7. Create a file named **file.doc** and following some operations
    a. Write some text into it.
    b. Read the written text.
    c. Display the text on browser.
8. Create a registration page atleast 6 fields and save the records on file and fetch all the data from the file.

Interview Questions: -

1. **Write syntax to open a file in PHP?**

PHP fopen() function is used to open file or URL and returns resource. It accepts two arguments: $filename and $mode.

Syntax:

resource fopen (string $filename , string $mode [, bool $use_include_path = false [, resourc e $context]])

2. How to read a file in PHP?

PHP provides various functions to read data from file. There are different functions that allow you to read all file data, read data line by line and read data character by character.

PHP file read functions are given below:

- fread()
- fgets()
- fgetc()

3. How to write in a file in PHP?

PHP fwrite() and fputs() functions are used to write data into file. To write data into file, you need to use w, r+, w+, x, x+, c or c+ mode.

4. How to delete file in PHP?

The unlink() function is used to delete file in PHP.

bool unlink (string $filename)

Multiple Choice Questions: -

1. **_____function is used to open the file**
   a) Open
   b) fileopen()
   c) file()
   d) fopen()

2. **fopen() takes two argument 1st argument is filename and 2nd argument is** a) filemode
   b) read mode
   c) Write Mode
   d) (b) and (c)

3. **fopen() can be also used to data from the url also _____**

a) true
b) false

4. **The only mode in which data can be written without overriding the file** a) r
   b) w
   c) a
   d) x+

5. **$handler = fopen('user.txt','r'); here $handler is**
   a) Resource
   b) data from user.txt
   c) filehandler
   d) filepointer

6. **How to check weather file pointer is reached EOF**
   a) by reading the End Character
   b) By reading null in End of File
   c) By reading \n
   d) By using feof()

7. **What is does the eof return**
   a) false
   b) true
   c) -1
   d) null

8. **_____function can be used to execute the data from the user in command line** a) input()
   b) fgets()
   c) fread()
   d) readline()

9. **This particular function can be used to read the complete file at once** a) fread()
   b) file()
   c) file_get_contents()
   d) file_read()

10. **How to read complete and convert into array**
    a) file_arr()
    b) file()

c) file_read()
d) read()

11. **This particular function can be used to get size of the file**
    a) size()
    b) filesize()
    c) file_size()
    d) sizefile()

12. **start() function is used to**
    a) get the file mode
    b) get the file information
    c) get the file status
    d) only works for unix based machine

13. **if you want to copy the content from one file paste to another then which method you would use**
    a) fread() and fwrite()
    b) file_get_contents and file_put_contents
    c) copy()
    d) All of the above

14. **_____ is used to check weather file is folder or not**
    a) Dir
    b) __dir__
    c) is_dir()
    d) is_file

Answer Key: -

1. d	2. d	3. a	4. c	5. d
6. d	7. c	8. b	9. c	10. b
11. b	12. b	13. d	14. c	

PHP Lecture – 11 (Session)

Long Answer Questions: -

Q1. What is a PHP Session and How to start session, receive session variable and destroy session with examples?

Ans.) When you work with an application, you open it, do some changes, and then you close it. This is much like a Session. The computer knows who you are. It knows when you start the application and when you end. But on the internet there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favorite color, etc). By default, session variables last until the user closes the browser.

So; Session variables hold information about one single user, and are available to all pages in one application.

Start a PHP Session

A session is started with the `session_start()` function.

Session variables are set with the PHP global variable: $_SESSION.

Now, let's create a new page called "demo_session1.php". In this page, we start a new PHP session and set some session variables:

```
<?php
  // Start the session  session_start(); ?>
<!DOCTYPE html>
<html>
<body>

<?php
// Set session variables
$_SESSION["favcolor"] = "green";
  $_SESSION["favanimal"] = "cat";  echo "Session variables are
set.";
?>

</body>
</html>
```

Note: *The session_start() function must be the very first thing in your document. Before any HTML tags.*

Get PHP Session Variable Values - Next, we create another page called "demo_session2.php". From this page, we will access the session information we set on the first page ("demo_session1.php").

Notice that session variables are not passed individually to each new page, instead they are retrieved from the session we open at the beginning of each page (session_start()).

Also notice that all session variable values are stored in the global $_SESSION variable:

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

Destroy a PHP Session - To remove all global session variables and destroy the session, use session_unset() and session_destroy()

```
<?php session_start();
?>
<!DOCTYPE html>
<html>
<body>
<?php
// remove all session variables session_unset();

// destroy the session session_destroy();
?>
</body>
</html>
```

Q2. Define Cookies with examples?

Ans.) A cookie is often used to identify a user. A cookie is a small file that the server embeds on the user's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With PHP, you can both create and retrieve cookie values.

Create Cookies With PHP - A cookie is created with the setcookie() function. **Syntax**

–

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

PHP Create/Retrieve a Cookie

The following example creates a cookie named "user" with the value "John Doe". The cookie will expire after 30 days (86400 * 30). The "/" means that the cookie is available in entire website (otherwise, select the directory you prefer).

We then retrieve the value of the cookie "user" (using the global variable $_COOKIE). We also use the isset() function to find out if the cookie is set:

```
<?php
$cookie_name = "user"; $cookie_value
= "John Doe";
setcookie($cookie_name, $cookie_value, time() +
(86400 * 30), "/"); // 86400 = 1 day
?>
<html>
<body>
 <?php
if(!isset($_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>
```

Note: The setcookie() function must appear BEFORE the <html> tag.

Note: The value of the cookie is automatically URLencoded when sending the cookie, and automatically decoded when received (to prevent URLencoding, use setrawcookie() instead).

Short Answer Questions: -

Q1. What is the use of session and cookies in PHP?

A session is a global variable stored on the server. Each session is assigned a unique id which is used to retrieve stored values. Sessions have the capacity to store relatively large data compared to cookies. The session values are automatically deleted when the browser is closed.

Technical Tasks: -

1. Design a session-based login page.
2. Develop a web program to create a session on a page and fetch the value of the session.
3. Develop a web program to destroy a session of above program.
4. Develop a web program to create a cookie on browser and fetch it value of cookie.
5. Develop a web program to delete the cookie of above program.
6. Develop a login page using session, validate username and password and also validate the pages and also get the login and logout time using session.
7. Develop a webpage of login, use remember me with checkbox and save the data of login form in cookie.

Interview Questions: -

1. Explain setcookie() function in PHP?

PHP setcookie() function is used to set cookie with HTTP response. Once cookie is set, you can access it by $_COOKIE superglobal variable.

Syntax:

bool setcookie (string $name [, string $value [, int $expire = 0 [, string $path

[, string $domain [, bool $secure = false [, bool $httponly = false]]]]])

2. How can you retrieve a cookie value?

echo $_COOKIE ["user"];

3. What is a session?

PHP Engine creates a logical object to preserve data across subsequent HTTP requests, which is known as session.

Sessions generally store temporary data to allow multiple PHP pages to offer a complete functional transaction for the same user.

Simply, it maintains data of an user (browser).

4. What is the method to register a variable into a session?

```
<?php
Session_register($ur_session_var);
?>
```

5. What is $_SESSION in PHP?

PHP $_SESSION is an associative array that contains all session variables. It is used to set and get session variable values.

6. What is PHP session_start() and session_destroy() function?

PHP session_start() function is used to start the session. It starts a new or resumes the existing session. It returns the existing session if session is created already. If session is not available, it creates and returns new sessions.

7. What is the difference between session and cookie?

The main difference between session and cookie is that *cookies are stored on user's computer in the text file* format while *sessions are stored on the server side*.

Cookies can't hold multiple variables on the other hand Session can hold multiple variables.

You can manually set an expiry for a cookie, while session only remains active as long as browser is open.

Multiple Choice Questions: -

1. **How many types of cookies are there**
    a. 1 Types
    b. types
    c. types
    d. types
2. **Cookies are part of which system _____Over HTTP response**
    a. ssl system
    b. protocal system

c. header system
d. none of the above

3. **What is url for getting reports tracking in case of google chrome**
    a. chrome://settings/advance/cookies
    b. chrome://settings/advance/resources
    c. chrome://storage/application/cookies
    d. chrome://settings/content/cookies
4. **How to set the cookies in php**
    a. $_Cookies['key']='value'
    b. set['cookies']='value'
    c. setcookie('key','value')
    d. $ENV
5. **Cookies that remains permanent until expiry date is reached.**
    a. session Cookies
    b. Persistent Cookies
    c. manifest cookies
    d. None of the above
6. **Which attribute can be used to specify the particular url in the cookies**
    a. uripath attribute
    b. domain attribute
    c. expiry attribute
    d. none of the above
7. **How to enable cookies only using secure channel**
    a. setcookie('key','secure',1)
    b. setcookie('key','value',expiry,'path',domain,1)
    c. setcookie('key','http_mode',1)
    d. setcookie('key','url_mode','http',1)
8. **Which super global varible is used for cookies**
    a. getcookie()
    b. $_Cookies['url']
    c. $_COOKIE['path']
    d. $_COOKIE['key']
9. **What is best way to delete the cookies**
    a. setcookie('username',time()-3600);
    b. setcookie('username',null)
    c. setcookie('username','')
    d. unset($_COOKIE['username'])
10. **While dealing with the session we need to setup the session which function is used to do it.**
    a. session_start()

b. session_set(true)
c. start_session()
d. start_session(true)

11. Out of All super global variable this variable is used to access the value from one page to any page

a. $_COOKIE
b. $_ENV
c. $_SERVER
d. $_SESSION

12. Captcha uses of which super global variable

a. $_COOKIE
b. $_SESSION
c. $_SERVER
d. $GLOBALS

13. Remember me or keep me signed is implemented using

a. $_SESSION
b. $_COOKIE
c. $_SERVER['token']
d. $_ENV

14. Only function used to unset all the session at once

a. session_unset()
b. unset($_SESSION)
c. session_destroy()
d. setNull($_SESSION)

15. How to check weather user is logged in or not

a. session_exist()
b. session_status()
c. isset($_SESSION)
d. session_isset()

Answer Key: -

1 b	2 c	3 d	4 c	5 b
6 b	7 b	8 d	9 a	10 a
11 d	12 b	13 b	14 c	15 c

PHP Lecture -12 (File Uploading)

Long Answer Questions: -

Q1. Concept of File Uploading in PHP?

Ans.) PHP allow you to upload any type of a file i.e. image, binary or text files.etc..,PHP has one in built global variable i.e. **$_FILES**, it contains all information about file.By the help of **$_FILES** global variable, we can get file name, file type, file size and temparary file name associated with file.

- **$_FILES['filename']['name']** - returns file name.
- **$_FILES['filename']['type']** - returns MIME type of the file.
- **$_FILES['filename']['size']** - returns size of the file (in bytes).
- **$_FILES['filename']['tmp_name']** - returns temporary file name of the file which was stored on the server.

In HTML, File Upload control can be created by using following:

```
<input type="file" name="fileupload"/>
```

In PHP, To upload the file we have to use the function called **move_uploaded_file().**

move_uploaded_file() function

The move_uploaded_file() function is used to move the uploaded file to a new location. It moves the file only if it is uploaded through the POST request.

Syntax: move_uploaded_file (string $filename , string $destination)

First Configure the **php.ini** File by ensure that PHP is configured to **allow file uploads**. In your **php.ini** file, search for the **file_uploads** directive, and set it to **On** i.e. **file_uploads = On**

```
<html>
<head>
<title>File Upload</title>
</head>
<body>
<form method="post" enctype="multipart/form-data">
    Select File: <input type="file" name="fileToUpload"/>
    <input type="submit" value="Upload Any File" name="submit"/>
</form>
```

```php
<?php
if(isset($_POST["submit"]))
{
$target_path = "D:/";
$target_path=$target_path.basename($_FILES['fileToUpload'] ['name']
);

if(move_uploaded_file($_FILES['fileToUpload']['tmp_name'], $target_p
ath))  {
    echo "File uploaded successfully!";
}
else {
    echo "Sorry, file not uploaded, please try again!";
}
}
?>
```

Technical Tasks: -

1. Develop a webpage to upload file.
2. Develop a webpage to download file
3. Develop a web program to create a web page using input type file and upload image in a folder.
4. Develop a web program to show the image uploaded by above program and download it.

Interview Questions: -

1. What is the method to execute a PHP script from the command line?

You should just run the PHP command line interface (CLI) and specify the file name of the script to be executed as follows.

2. How to upload file in PHP?

The move_uploaded_file() function is used to upload file in PHP. move_uploaded_file

(string $filename , string $destination)

3. How to download file in PHP?

The readfile() function is used to download file in PHP.

readfile (string $filename)

4. How can you send email in PHP?

The mail() function is used to send email in PHP.

mail($to,$subject,$message,$header)

Multiple Choice Questions: -

1. inorder to check weather your php virtual machine in apache support file writing/Uploading can be checked using while configuration file

a) php.conf
b) php.ini
c) http.conf
d) filesmod.ini

2. The only attribute which is need to be added in order to enable file uploading

a) enctype="multipart/form-data"
b) enctype="www//urlencode"
c) enctype="urlencode/xml+utf8"
d) enctype="formdata/urlencode"

3. what is fullform of mime type

a) Multipartition Internet Media Extensions
b) Multipartition Internet Media Encryption
c) Multipurpose Internet Mail Extensions
d) Multipurpose Internet media Extensions

4. what is default foldername where file is uploaded by default

a) file folder
b) htdocs folder
c) uploads folder
d) tmp folder

5. the function is used to get the filename from the path url a) file()
b) basename()
c) $_FilES['file']['name']
d) file_name()

6. This particular function is used to check weather file exist

a) doexist()
b) fileexist()
c) file_exist()
d) file_check()

7. This Constant can be used get the path information

a) PATH_INFO
b) PATH_EXTENSION
c) PATH_FILE
d) PATHINFO_EXTENSION

8. Suppose my requirement is get the dimension of uploaded file which is actually image as size this can be done using

a) $_FilES['file']['size']
b) $_FilES['file']['mime']
c) file_size()
d) getImagesize()

9. This powerful function can be used to move the particular file from tmp folder to specific(Target) Folder

a) move_uploaded_file()
b) file_put_contents()
c) fwrite()
d) copy()

10. The only function we is used to force download file in php

a) header()
b) readfile()
c) flush()
d) spl_force_pull()

Answer Key: -

1. b	2. a	3. d	4 d	5 b
6 c	7 d	8 d	9 a	10 b

PHP Lecture -13 (Introduction to MySQL)

Long Answer Questions: -

Q1. What is MySQL.

Ans.) MySQL is one of the most popular relational database systems being used on the Web today. It is freely available and easy to install, however if you have installed Wampserver it already there on your machine. MySQL database server offers several advantages:

- MySQL is easy to use, yet extremely powerful, fast, secure, and scalable.
- MySQL runs on a wide range of operating systems, including UNIX or Linux, Microsoft Windows, Apple Mac OS X, and others.
- MySQL supports standard SQL (Structured Query Language).
- MySQL is ideal database solution for both small and large applications.
- MySQL is developed, and distributed by Oracle Corporation.
- MySQL includes data security layers that protect sensitive data from intruders.

MySQL database stores data into tables like other relational database. A table is a collection of related data, and it is divided into rows and columns.

Each row in a table represents a data record that are inherently connected to each other such as information related to a particular person, whereas each column represents a specific field such as *id*, *first_name*, *last_name*, *email*, etc. The structure of a simple MySQL table that contains person's general information may look something like this:

```
+----+---------------+---------------+-------------------------------------+
| id | first_name | last_name | email             |
+----+---------------+---------------+-------------------------------------+
| 1 | Peter      | Parker        | peterparker@mail.com   |
| 2 | John       | Rambo         | johnrambo@mail.com     |
| 3 | Clark      | Kent          | clarkkent@mail.com     |
| 4 | John       | Carter        | johncarter@mail.com    |
| 5 | Harry      | Potter        | harrypotter@mail.com   |
+----+---------------+---------------+-------------------------------------+
```

NOTE : Websites like Facebook, Twitter, Wikipedia uses MySQL for their storage need. So you can easily understand what MySQL is capable of.

Q2. What is SQL and How many commands used in SQL?

Ans.) SQL, the Structured Query Language, is a simple, standardized language for communicating with relational databases like MySQL. With SQL you can perform any database-related task, such as creating databases and tables, saving data in database tables, query a database for specific records, deleting and updating data in databases.

Look at the following standard SQL query that returns the email address of a person whose first name is equal to 'Peter' in the persons table:

```
SELECT email FROM persons WHERE first_name="Peter"
```

If you execute the SQL query above it will return the following record:

```
peterparker@mail.com
```

We have divided this chapter in the following sections −

- Connecting to MySQL database − Learn how to use PHP to open and close a MySQL database connection.
- Create MySQL Database Using PHP − This part explains how to create MySQL database and tables using PHP.
- Delete MySQL Database Using PHP − This part explains how to delete MySQL database and tables using PHP.
- Insert Data To MySQL Database − Once you have created your database and tables then you would like to insert your data into created tables. This session will take you through real example on data insert.
- Retrieve Data From MySQL Database − Learn how to fetch records from MySQL database using PHP.
- Using Paging through PHP − This one explains how to show your query result into multiple pages and how to create the navigation link.
- Updating Data Into MySQL Database − This part explains how to update existing records into MySQL database using PHP.
- Deleting Data From MySQL Database − This part explains how to delete or purge existing records from MySQL database using PHP.

- Using PHP To Backup MySQL Database – Learn different ways to take backup of your MySQL database for safety purpose.

Based on SQL commands the SQL is categorized in three parts:-

1. DDL (Data Definition Language)
2. DML (Data Manipulation Language)
3. DCL (Data Control Language)

Short Answer Questions: -

Q1. Explain DDL in brief?

Ans.) DDL stands for "Data Definition Language." A DDL is a language used to define data structures and modify data. For example, DDL commands can be used to add, remove, or modify tables within in a database. DDLs used in database applications are considered a subset of SQL, the Structured Query Language. However, a DDL may also define other types of data, such as XML.

A Data Definition Language has a pre-defined syntax for describing data. For example, to build a new table using SQL syntax, the CREATE command is used, followed by parameters for the table name and column definitions. The DDL can also define the name of each column and the associated data type. Once a table is created, it can be modified using the ALTER command. If the table is no longer needed, the DROP command can be used to delete the table.

Commands	Working
Create	The create command is used to create a new database object.
Alter	The alter command is used to modify database object.
Truncate	The truncate command is used to delete all data from database object but structure remains same.
Drop	The drop command is used to delete database object.

Q2. Explain DML in brief?

Ans.) Tables and formulas are useful when interacting with data held in a database via SQL up to a point, but there comes a time when you really want to perform some pretty complex interactions with data. In that case, you'll likely need **Data Manipulation Language**. Data Manipulation Language is a way of telling a database exactly what you want it to do by speaking in a way that it is built from the ground up to understand. When it comes to working within existing data, whether it is to add, move, or delete data, Data Manipulation Language offers an effective way of doing it.

Commands	Working
Insert	The insert command is used to insert record into a database object.
Select	The select command is used to select data from database object.
Update	The update command is used to modify data of database object.
Delete	The delete command is used to delete the data from database object.

Q3. Explain DCL in brief?

Ans.) DCL is short name of Data Control Language which includes commands such as GRANT and mostly concerned with rights, permissions and other controls of the database system.

Commands	Working
Grant	The grant command is used to give rights to database user.
Revoke	The revoke is just opposite to grant command . It is used to take off rights from database user.
Rename	The rename command is used to change the name of database object.

Technical Tasks: -

1. Write a SQL statement to create a simple table countries including columns country_id, country_name and region_id.

2. Write a SQL statement to create a simple table countries including columns country_id, country_name and region_id which is already exists.
3. Write a SQL statement to insert a record with your own value into the table countries against each columns.
4. Write a SQL statement to create duplicate of countries table named country_new with all structure and data.
5. Write a SQL statement to change the email column of employees table with 'not available' for all employees.

Interview Questions: -

1. How do you connect MySQL database with PHP?

There are two methods to connect MySQL database with PHP. Procedural and object oriented style.

2. How to create connection in PHP?

The mysqli_connect() function is used to create connection in PHP.

mysqli_connect (server, username, password)

3. How to create database connection and query in PHP?

Since PHP 4.3, mysql_reate_db() is deprecated. Now you can use following 2 alternatives.

- mysqli_query()
- PDO::_query()

Multiple Choice Questions: -

1. The Collection of columns is called as

a) Cardinality
b) Entity
c) Degree
d) Cartesian

2. Attributes are Alias name to

a) Schema
b) Table
c) Column
d) Rows

3. which of the following is not the db objects with reference to mysql a) view

b) table

c) trigger

d) meta dictionary

4. Which Command deal with Transaction commit permission

a) DML

b) DCL

c) DDL

d) TCL

5. _____is a set of rules that defines how to compare and sort character strings. a) Charset

b) Collation

c) Commands

d) queries

6. which command can be used to show query for created database

a) show

b) select

c) desc

d) use

7. How to select the database

a) By select

b) By Use

c) By Connect

d) By Root

8. Who is super user in Mysql

a) Root

b) System

c) Admin

d) None of the Above

9. _____ is used to check if database already available

a) if Not db

b) use Not Db

c) Exists
d) if Object Not Db

10. Dropping Means

a) Deleting ALL Record
b) Deleting the Schema
c) Deleting db Entity
d) Deleting Table

11. That constraints is used to define that field cannot be blank

a) Primary key
b) Default
c) Not Null
d) Null

12. which keyword to Add Column before last columns

a) Before last columns
b) after 2nd last column
c) first
d) None

13. Rename in Alter can be used rename db Entity_____

a) True
b) False

14. can alter command use to alter the dbname

a) False
b) True

15. Only Command that works same for field and database and table

a) Select
b) Show
c) Drop
d) alter

16. _____Command is used to create database objects

a) alter
b) select
c) use
d) create

17. _____This particular command is used to describe table schema.

a) Use
b) Show
c) Desc
d) select

18. Extra Columns in the Schema defines

a) Keys
b) Constraints
c) Relations
d) None of the Above

19. what is the name of the keyword used to create alias

a) view
b) As
c) On
d) [space]

20. which sql keyword is optional while inserting all the records

a) column_name
b) into
c) values
d) None of the above

21. which data set can be used to insert multiple record at same time. a) List
b) Array
c) Tuple
d) string

22. which wild card character is used to fetch record from all fields(columns) a) as
b) (field1,field2,feild3...field_n)
c) *
d) %

23. Selection with All(*) clause is called as

a) Selection
b) Projection
c) Viewing
d) joining

24. set is keyword is used to field values to the multiple columns is a part of which query

a) update
b) select
c) insert
d) None of the Above

25. if where class is not used in update then

a) We will get Error
b) query will not be fired
c) all record will be Updated
d) First Record will be Updated

26. while deleting the record primary key added with where clause to find the unique record and delete if we want to delete multiple records then which keyword will be used

a) *
b) All
c) Limit
d) in

27. This keyword is used to uniquely identify the record and remove any duplicates if any by _____

a) Unique
b) Index
c) Distinct
d) REPLACE

28. The abstracted or virtual table can be created for the real table by using a) Join
b) View
c) Aliasing
d) projection

29. In order to organize the record from latest date to recent date we can use

a) order by date Asc
b) Order by date Desc
c) Limit max(date),min min(date)
d) None of the above

30. Pagination is concept of fetching the finite no of record per request can be done using

a) Limit
b) Limit Offset
c) Limit offset,Limit
d) No Limit

31. How comments are applied in mysql

a) #
b) /**/
c) - -
d) All of the above

32. _____used to connect with database

a) mysql_connect()
b) new mysqli()
c) mysqli_connect()
d) mysqli_connect_db()

33. What is Ip of the host while connecting to the database

a) Localhost
b) 129.0.0.1
c) ::1
d) 198.168.0.0.1

34. _____function is used to check for the connection error

a) mysqli_error()
b) mysqli_connection_error()
c) mysqli_connect_error()
d) mysqli_connect_errorno()

35. mysqli_query is used for

a) firing the query
b) grab the Result set
c) Show the Resource or Error if any
d) All the Above

36. mysqli_num_rows() can be used to the no of

a) rows

b) matched rows
c) matched columns
d) None of the Above

37. This perticular function takes result set as argument and can be used to fetch the resource and convert into associative array

a) mysqli_fetch_array()
b) mysqli_fetch_rows()
c) mysqli_fetch_assoc()
d) mysqli_fetch_num()

38. mysqli_error return any generated error related to last query statement

a) TRUE
b) False

39. which of the following query is not committable query

a) update
b) insert
c) join
d) None of the Above

40. while() loops can be used for single record instead of if but query has to be optimized by

a) where clause
b) Limit
c) Less Columns
d) None of the Above

41. which APi level is most widely used in php for database connection with Oracle

a) Prepared statement
b) PDO
c) Mysql
d) mysqli

Answer Key: -

1. a	2. c	3. c	4. d	5. b	6. a
7. b	8. a	9. c	10. c	11. c	12. b
13. a	14. a	15. c	16. d	17. c	18. b
19. b	20. b	21. c	22. c	23. b	24. a

25. c	26. d	27. c	28. b	29. b	30. c
31. c	32. c	33. b	34. c	35. d	36. b
37. c	38. a	39. c	40. b	41. b	

PHP Lecture -14 (MySQL Commands)

Long Answer Questions: -

Q1. Explain CREATE Command and how to use create command.

Ans.) The table is a database object which have its rows and columns. The rows are also called tuples and columns are called attributes or Fields. **Syntax To Create table:-** create table <tablename>

(

<field_name> <data_type>,

<field_name> <data_type>

);

Create the table Employee with following structure

empid	int
empname	varchar2(20)
department	varchar2(20)
salary	int

```
create table employee
( empid int, empname
varchar2(20), department
varchar2(20),
salary int

);
```

This query display the message Table is created..

Q2. Use of insert command to insert record into the table.

Ans.) Insert Command is used to insert the actual values in respective column of the table.

Syntax - insert into <table_name> values(<value1>,<value2>,<value3>);

For Example - insert into employee values(1001,'Brijesh

Mishra','Development',80000);

When we execute above query, then "1 row created" message will displayed.

In the similar way insert the following records:-

Insert into employee values(1002,'Rajat Verma','Development',85000);

Insert into employee values(1003,'Priya Shukla','HR',50000);

Insert into employee values(1004,'Aman Verma','Marketing',80000);

If you want to insert values in specified columns follow this syntax -

Insert into <table_name> (<column_name>,<column_name>) values(<value1>,<value2>);

E.g.

Insert into employee (empid,empname,department) values(1005,'Rajesh','Development');

When you execute this command, it will display "1 row created"..

Q3. Use of SELECT Command with example?

Ans.) The select command of sql is used to select record from database object like table.

Use of select command to display all rows and columns.

Syntax -

Select * from <table_name>;

For example -

Select * from employee;

EMPID	EMPNAME	DEPARTMENT	SALARY
1001	Brijesh	Development	80000
1002	Rajat Verma	Development	85000
1003	Priya Shukla	HR	50000
1004	Aman Verma	Marketing	80000
1005	Rajesh	Development	

Use of select command to display specified columns.

Syntax -

Select <column_name>,<column_name>,<column_name> from <table_name>;

For example -

Select empid,empname,department from employee;

It will give following result

EMPID	EMPNAME	DEPARTMENT
1001	Brijesh	Development
1002	Rajat Verma	Development
1003	Priya Shukla	HR
1004	Aman Verma	Marketing
1005	Rajesh	Development

Where Clause with SELECT Command

The where clause in sql is used to specify the condition in query. The syntax of select operation with where clause is given below.

Syntax -

Select * from <table_name> where <condition>;

For example -

Select * from employee where empid=1001;

It will give following result

EMPID	EMPNAME	DEPARTMENT	SALARY
1001	Brijesh	Development	80000

Q4. Define Operators in SQL?

Ans.) An operator is a reserved word or a character used primarily in an SQL statement's WHERE clause to perform operation(s), such as comparisons and arithmetic operations. These Operators are used to specify conditions in an SQL statement and to serve as conjunctions for multiple conditions in a statement.

- Arithmetic operators
- Comparison operators
- Logical operators

Arithmetic Operators

Operator	Names
+	Addition.
-	Substarction
*	Multiplication
/	Division
%	Modulo (Remainder Collector)

Comparison Operators

Operator	Description
=	Checks if the values of two operands are equal or not, if yes then condition becomes true.
!=	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.
<>	Checks if the values of two operands are equal or not, if values are not equal then condition becomes true.
>	Checks if the value of left operand is greater than the value of right operand, if yes then condition becomes true.
<	Checks if the value of left operand is less than the value of right operand, if yes then condition becomes true.
>=	Checks if the value of left operand is greater than or equal to the value of right operand, if yes then condition becomes true.
<=	Checks if the value of left operand is less than or equal to the value of right operand, if yes then condition becomes true.

Logical Operators

Operator	Description
ALL	The ALL operator is used to compare a value to all values in another value set.
AND	The AND operator allows the existence of multiple conditions in an SQL statement's WHERE clause.
ANY	The ANY operator is used to compare a value to any applicable value in the list as per the condition.
BETWEEN	The BETWEEN operator is used to search for values that are within a set of values, given the minimum value and the maximum value.
EXISTS	The EXISTS operator is used to search for the presence of a row in a specified table that meets a certain criterion.
IN	The IN operator is used to compare a value to a list of literal values that have been specified.
LIKE	The LIKE operator is used to compare a value to similar values using wildcard operators.

Short Answer Questions: -

Q1. Use of Truncate Command?

Ans.) Truncate Command:-The truncate command of SQL is used to delete all data from table. The Syntax of truncate command is given below - truncate table <table_name>;

For Example - truncate

table employee;

Q2. Use of Drop Command?

Ans.) The drop command of SQL is used to delete database object. The Syntax of drop command to drop table is given below -

drop table <table_name>;

For Example - drop table

employee;

Technical Tasks: -

1. Develop a php module for task information having fields taskid, taskname, taskdesc, submittedby, submissiondate. Store the task information into the taskinfo table.
2. Develop a php code to access all task information and take a search button to search the task information based on taskid.
3. Apply delete and update operations in task information module.
4. Create a registration page atleast 10 fields and save the records into database.
5. Display all the records in tabular format.
6. Update the records.
7. Delete records one by one.

Interview Questions: -

1. What does SQL in MySQL stand for?

The SQL in MySQL stands for Structured Query Language. This language is also used in other databases such as Oracle and Microsoft SQL Server.

2. How can you interact with MySQL?

There are three main ways you can interact with MySQL:

- using a command line
- via a web interface
- through a programming language

3. What are MySQL Database Queries?

A query is a specific request or a question. One can query a database for specific information and have a record returned.

4. How to create an Index in MySQL?

In MySQL, there are different index types, such as a regular INDEX, a PRIMARY KEY, or a FULLTEXT index. You can achieve fast searches with the help of an index. Indexes speed up performance by either ordering the data on disk so it's quicker to find your result or, telling the SQL engine where to go to find your data.

Example: Adding indexes to the history table:

```
ALTER TABLE history ADD INDEX (author (10));
ALTER TABLE history ADD INDEX (title (10));
ALTER TABLE history ADD INDEX (category (5));
ALTER TABLE history ADD INDEX (year);
DESCRIBE history;
```

5. What is MySQL "Views"?

In MySQL, a view consists of a set of rows that is returned if a particular query is executed. This is also known as a 'virtual table'. Views make it easy to retrieve the way of making the query available via an alias.

The advantages of views are:

- Simplicity
- Security
- Maintainability

6. How do you create and execute views in MySQL?

Creating a view is accomplished with the CREATE VIEW statement. As an example:

```
CREATE
  [OR REPLACE]
  [ALGORITHM = {MERGE | TEMPTABLE | UNDEFINED }]
  [DEFINER = { user | CURRENT_USER }]
  [SQL SECURITY { DEFINER | INVOKER }]
  VIEW view_name [(column_list)]
  AS select_statement
  [WITH [CASCADED | LOCAL] CHECK OPTION]
```

7. What are MySQL Triggers?

A trigger is a task that executes in response to some predefined database event, such as after a new row is added to a particular table. Specifically, this event involves inserting, modifying, or deleting table data, and the task can occur either prior to or immediately following any such event.

Triggers have many purposes, including:

- Audit Trails
- Validation

- Referential integrity enforcement

8. How many Triggers are possible in MySQL?

There are six Triggers allowed to use in the MySQL database:

- Before Insert
- After Insert
- Before Update
- After Update
- Before Delete
- After Delete

9. What is the MySQL server?

The server, mysqld, is the hub of a MySQL installation; it performs all manipulation of databases and tables.

Multiple Choice Questions: -

1. This particular function is used to handle checkboxes or multiple selection a) array()

b) implode()

c) explode()

d) None of the Above

2. what is url for working with mysqldb

a) http://localhost:70/mysqli

b) http://localhost:70/mariadb

c) http://localhost:70/phpmyadmin

d) http://localhost:70/phpmysql

3. What is meaning AI in Primary key constraints while working with table scheme in PHPMYADMIN

a) ActiveIncrement

b) Artificial intelligence

c) AllIndex

d) Autoincrement

4. while sending data to the server it is recommended to use secure data protocol this can be simply handled using

a) method="get"

b) method="post"

c) method=""

d) enctype="true"

5. This super global variable can be used to accept the values from form independent of form method type

a) $_POST
b) $_GET
c) $_REQUEST
d) $_PUT

6. The only variable which acts as resource object to build connection in case crud with phpmyadmin

a) $_SERVER
b) $_REQUEST
c) $con
d) $query

7. This function is used to convert the record into associative returned from the database

a) mysqli_fetch_array()
b) mysqli_fetch_assoc()
c) mysqli_query()
d) mysqli_num_assoc()

8. To redirect the page the php specific function is

a) window.location.href
b) header()
c) redirect()
d) href()

9. The record return from the database when select query is fired in case of showing record the data object is called as

a) result_array
b) result_set
c) record_Set
d) associative_array

10. Events in php can binded with query string in url by for update and delete

a) <form action>
b) window.location
c) <a href>
d) header(“location:”);

11. The part of url which is generated due to get request is called_____ a) Header

b) Querystring

c) Queryheader

d) formdata

12. Which operator uniquely identify the query_string

a) =

b) &

c) ?

d) None of the above

13. The only tag attribute used to embed the sensitive information within the form is a) csrf_token

b) hidden

c) enctype

d) None of the above

14. The values in the query string are transmitted as pair

a) variables and values

b) key and value pair

c) key and hash

d) data array

15. Which part of the query string is encrypted

a) key is encrypted

b) hash is encrypted

c) value is encrypted

d) data is encrypted

16. How to pass multiple value pairs from url

a) using &

b) using =

c) using ?

d) None of the above

17. _____function can be used to inherit the code from dbconnection a) Require

b) include

c) Both
d) None of the above

18. _____attribute use to bind the value to the html input tags

a) Type
b) Value
c) Required
d) hidden

19. How to bind js inside php

a) using echo
b) using <script></script>
c) using print_r
d) None of the above

Answer Key: -

1. b	2. c	3. a	4. b	5. c	6. c
7. b	8. b	9. b	10. c	11. b	12. c
13. b	14. b	15. c	16. a	17. c	18.
19.b					

PHP Lecture – 15 (API, Map, & Payment Gateway Integrations)

Interview Questions: -

1. What is an API?

An API (Application Programming Interface) is a software intermediary that enables two applications to communicate with each other. It comprises a number of subroutine definitions, logs, and tools for creating application software.

In an API testing interview, you could be asked to give some API examples, here are the wellknown ones: Google Maps API, Amazon Advertising API, Twitter API, YouTube API, etc.

2. What are main differences between API and Web Service?

- All Web services are APIs but not all APIs are Web services.
- Web services might not contain all the specifications and cannot perform all the tasks that APIs would perform.
- A Web service uses only three styles of use: SOAP, REST and XML-RPC for communication whereas API may be exposed to in multiple ways.
- A Web service always needs a network to operate while APIs don't need a network for operation.

3. What are some architectural styles for creating a Web API?

This is one of the fundamental Web API interview questions. Bellows are four common Web API architectural styles:

- HTTP for client-server communication
- XML/JSON as formatting language
- Simple URI as the address for the services
- Stateless communication

4. What is API Testing?

API testing is a kind of software testing that determines if the developed APIs meet expectations regarding the functionality, reliability, performance, and security of the application.

5. What are the common API testing types?

While there are certainly specialty tests, and no list can be asked to be comprehensive in this realm, most tests fit broadly into these following nine categories that you should remember before attending in an API testing interview.

1. Validation Testing
2. Functional Testing
3. UI testing
4. Load testing
5. Runtime/ Error Detection
6. Security testing
7. Penetration testing
8. Fuzz testing
9. Interoperability and WS Compliance testing

6. What are common API errors that often founded?

Not only API fundamental questions, the interviewer also determine your knowledge and experience by asking about the API errors in a Web API testing interview. So the most common ones are:

- Missing module errors
- Documentation errors
- Parameter validation errors

- And some standard error expectations as if the result is not so predicted then the occurrence of errors can be seen and for the same warnings are specified in the form of a message. There can be one or more warnings within an individual module.

7. What is REST?

REST (Representational State Transfer) is an architectural style for developing web services which exploit the ubiquity of HTTP protocol and uses HTTP method to define actions. It revolves around resource where every component being a resource that can be accessed through a shared interface using standard HTTP methods.

In REST architecture, a REST Server provides access to resources and REST client accesses and makes these resources available. Here, each resource is identified by URIs or global IDs, and REST uses multiple ways to represent a resource, such as text, JSON, and XML. XML and JSON are nowadays the most popular representations of resources.