V2V EDTECH LLP

Online Coaching at an Affordable Price.

OUR SERVICES:

- Diploma in All Branches, All Subjects
- Degree in All Branches, All Subjects
- BSCIT / CS
- Professional Courses

📞 +91 93260 50669

🌐 v2vedtech.com

▶ V2V EdTech LLP

📷 v2vedtech

1.Define Array. Expalin its type with example.

Ans:

Definition: An array is a special variable, which can hold more than one value at a time.

Example:

1)Indexed array:

$colors = array("Red", "Green", "Blue");

2)Associative array:

$student_one = array("Maths"=>95, "Physics"=>90, "Chemistry"=>96, "English"=>93, "Computer"=>98);

3)Multidimensional array :

$movies =array("comedy" =>array("Pink Panther", "John English", "See no evil hear no evil"),

"action" =>array("Die Hard", "Expendables","Inception"),

"epic" =>array("The Lord of the rings"));

2.Write down rules for declaring PHP variable

Ans:

a. A variable starts with the $ sign, followed by the name of the variable.

b. A variable name must start with a letter or the underscore character.

c. A variable name should not contain spaces. If a variable name is more than one word, it should be separated with an underscore ($first_name), or with capitalisation ($firstName).

d. Variables used before they are assigned have default values.

e. A variable name cannot start with a number.

f. A variable name can only contain alpha-numeric characters (AZ, a-z) and underscores.

g. Variable names are case-sensitive ($name and $NAME are two different variables)

h. Variables can, but do not need, to be declared before assignment. PHP automatically converts the variable to the correct data type, depending on its value.

i. Variables in PHP do not have intrinsic types - a variable does not know in advance whether it will be used to store a number or a string of characters.

3.Define Introspection and explain it with suitable example.

Ans:

• Introspection in PHP offers the useful ability to examine an object's characteristics, such as its name, parent class (if any) properties, classes, interfaces and methods.

• PHP offers a large number functions that can be used to accomplish the above task.

• Following are the functions to extract basic information about classes such as their name, the name of their parent class etc.

Function	Description
class_exists()	Checks whether a class has been defined.
get_class()	Returns the class name of an object.
get_parent_class()	Returns the class name of an object's parent class.
is_subclass_of()	Checks whether an object has a given parent class.
get_declared_classes()	Returns a list of all declared classes.
get_class_methods()	Returns the names of the class' methods.
get_class_vars()	Returns the default properties of a class.
interface_exists()	Checks whether the interface is defined.
method_exists()	Checks whether an object defines a method.

Example:
```php
<?php
class Test
{
function testing_one()
{
return(true);
}
function testing_two()
{
return(true);
}
function testing_three()
{
return(true);
}
//Class "Test" exist or not
if (class_exists('Test'))
{
$t = new Test();
echo "The class is exist.<br>";
}
else
{
echo "Class does not exist. <br>";
}
```

```
//Access name of the class
$p= new Test();
echo "Its class name is " ,get_class($p) , "<br>";
//Aceess name of the methods/functions
$method = get_class_methods(new Test());
echo "<br>List of Methods:</br><br>";
foreach ($method as $method_name)
{
echo "$method_name<br>";
}
?>
```
Output :
The class is exist.
Its class name is Test
List of Methods:
testing_one
testing_two
testing_three

4.Write difference between get() and post() method of form
Ans:

HTTP GET	HTTP POST
In GET method we cannot send large amount of data rather limited data is sent because the request parameter is appended into the URL.	In POST method large amount of data can be sent because the request parameter is appended into the body.
GET request is comparatively better than Post so it is used more than the Post request.	POST request is comparatively less better than Get so it is used less than the Get request.
GET request is comparatively less secure because the data is exposed in the URL bar.	POST request is comparatively more secure because the data is not exposed in the URL bar.

Request made through GET method are stored in Browser history.	Request made through POST methodis not stored in Browser history.
GET method request can be saved as bookmark in browser.	POST methodrequest cannot be saved as bookmark in browser.
Request made through GET method are stored in cache memory of Browser.	Request made through POST method are not stored in cache memory of Browser.
Data passed through GET method can be easily stolen by attackers.	Data passed through POST method cannot be easily stolen by attackers.
In GET method only ASCII characters are allowed.	In POST method all types of data is allowed.

5.Explain the use of break and continue statements.

Ans:

Break statement:-break keyword is used to terminate and transfer the control to the next statement when encountered inside a loop or switch case statement.

Syntax:

if (condition)

{ break; }

Example:

```
<?php
for ($a = 0; $a < 10; $a++)
{
if ($a == 7)
{
```

5

```
break; /* Break the loop when condition is true. */
}
echo "Number: $a <br>";
}
echo " Terminate the loop at $a number";
?>
```

ii)Continue Statement
It is used to skip the execution of a particular statement inside the loops.
```
if (condition)
{ continue; }
```
Example:
```
<?php
for ($i = 0; $i< 10; $i++)
{
if ($i == 5)continue;
{
echo " $i<br>";
}}
echo "end";
?>
```

6.Describe i) Start session
ii) Get session variables
Ans: **i) Start session**
PHP session_start() function is used to start the session. It starts a new or resumes existing session. It returns existing session if session is created already. If session is not available, it creates and returns new session
Syntax 1.
boolsession_start(void)
Example 1.session_start();
PHP $_SESSION is an associative array that contains all session variables. It is used to set and get session variable values.
Example: Store information
2. $_SESSION["CLASS"] = "TYIF STUDENTS"
Example: Program to set the session variable (demo_session1.php)
```
<?php
session_start();
?>
<html>
```

```
<body>
<?php
$_SESSION["CLASS"] = "TYIF STUDDENTS";
echo "Session information are set successfully.</br>";
?>
</body>
</html>
```

ii)Get Session variables
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:

Example:- program to get the session variable values(demo_session2.php)
```
<?php
session_start();
?>
<html>
<body>
<?php
echo "CLASS is: ".$_SESSION["CLASS"];
?>
</body>
</html>
```
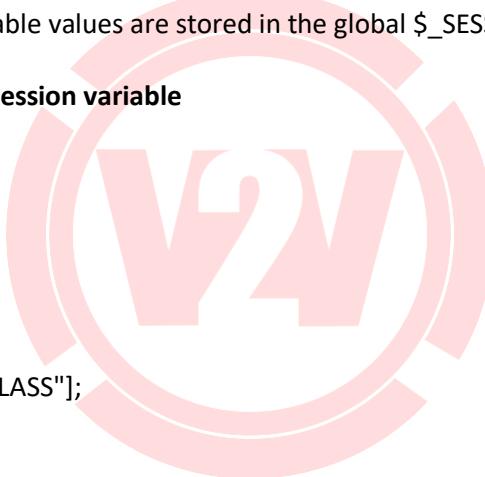
7.Explain two functions to scale the given image.
Ans:
imagecopyresized() function : It is an inbuilt function in PHP which is used to copy a rectangular portion of one image to another image and resize it. dst_image is the destination image, src_image is the source image identifier.
Syntax:
imagecopyresized(dst_image, src_image, dst_x, dst_y,src_x, src_y, dst_w,dst_h,src_w, src_h)
dst_image: It specifies the destination image resource.
src_image: It specifies the source image resource.
dst_x: It specifies the x-coordinate of destination point.

dst_y: It specifies the y-coordinate of destination point.
src_x: It specifies the x-coordinate of source point.
src_y: It specifies the y-coordinate of source point.
dst_w: It specifies the destination width.
dst_h: It specifies the destination height.
src_w: It specifies the source width.
src_h: It specifies the source height.

Example:
imagecopyresized($d_image,$s_image,0,0,50,50,200,200,$s_width, $s_height);

imagecopyresampled() function : It is used to copy a rectangular portion of one image to another image, smoothly interpolating pixel values thatresize an image.
Syntax:
imagecopyresampled(dst_image, src_image, dst_x, dst_y,src_x, src_y, dst_w,dst_h,src_w, src_h)

dst_image: It specifies the destination image resource.
src_image: It specifies the source image resource.
dst_x: It specifies the x-coordinate of destination point.
dst_y: It specifies the y-coordinate of destination point.
ṇsrc_x: It specifies the x-coordinate of source point.
src_y: It specifies the y-coordinate of source point.
dst_w: It specifies the destination width.
dst_h: It specifies the destination height.
src_w: It specifies the source width.
src_h: It specifies the source height.

Example:
imagecopyresampled($d_image,$s_image,0,0,50,50,200,200,$s_widt h,$s_height);

8.Explain class & Object and describe syntax to create class and object in PHP.
Ans:
Define a Class
A class is defined by using the class keyword, followed by the name of the class and a pair of curly braces ({}). All its properties and methods go inside the braces:
Syntax
```
<?php
class Fruit{
// code goes here…
}
?>
```

Below we declare a class named Fruit consisting of two properties ($name and $color) and two methods set_name() and get_name() for setting and getting the $name property:
Example

```php
<?php
class Fruit{
// Properties
public $name;
public $color;

// Methods
function set_name($name) {
$this->name = $name;
}
function get_name() {
return $this->name;
}
}
?>
```

Note: In a class, variables are called properties and functions are called methods!

Define Objects
Classes are nothing without objects! We can create multiple objects from a class. Each object has all the properties and methods defined in the class, but they will have different property values.

Objects of a class are created using the new keyword.In the example below, $apple and $banana are instances of the class Fruit:

Example

```php
<?php
class Fruit{
// Properties
public $name;
public $color;

// Methods
function set_name($name) {
$this->name = $name;
}
function get_name() {
return $this->name;
}
}
```

```php
$apple = new Fruit();
$banana = new Fruit();
$apple->set_name('Apple');
$banana->set_name('Banana');

echo $apple->get_name();
echo "<br>";
echo $banana->get_name();
?>
```

In the example below, we add two more methods to class Fruit, for setting and getting the $color property:

```php
Example
<?php
class Fruit{
// Properties
public $name;
public $color;

// Methods
function set_name($name) {
$this->name = $name;
}
function get_name() {
return $this->name;
}
function set_color($color) {
$this->color = $color;
}
function get_color() {
return $this->color;
}
}
$apple = new Fruit();
$apple->set_name('Apple');
$apple->set_color('Red');
echo "Name: " . $apple->get_name();
echo "<br>";
echo "Color: " . $apple->get_color();
?>
```

PHP - The $this Keyword
The $this keyword refers to the current object, and is only available inside methods.

9.State any four form controls to get user's input in PHP.

Ans:

1. Textbox control:It is used to enter data. It is a single line input on a web page.

Tag :<input type="text'>

2. Password control:It is used to enter data that appears in the form of special characters on a web page inside box. Password box looks like a text box on a wab page.

Tag:<input type="password" >

3. Textarea : It is used to display a textbox that allow user to enter multiple lines of text.

Tag :<textarea>...</textarea>

4. Checkbox:It is used to display multiple options from which user can select one or more options.

Tag:<input type="checkbox">

5. Radio / option button :These are used to display multiple options from which user can select only one option.

Tag :<input type="radio">

6. Select element (list) / Combo box / list box:

<select>...</select> : This tag is used to create a drop-down list box or scrolling list box from which user can select one or more options.

<option>...<</option> tag is used to insert item in a list.

10.Write steps to create database using PHP

Ans:

Steps using PHP Code:Creating database: With CREATE DATABASE query

Step 1: Set variables with values for servername, username, password.

Step 2: Set connection object by passing servername, username, password as parameters.

Step 3: Set query object with the query as "CREATE DATABASE dept";

Step 4: Execute query with connection object. Code (Optional)-

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";
$conn = new mysqli($servername, $username, $password);
if ($conn->connect_error)
{
die("Connection failed: " . $conn->connect_error);
}
$sql = "CREATE DATABASE ifdept";
if ($conn->query($sql) === TRUE)
{
```

11

```
echo "Database created successfully";
}
else
{
echo "Error creating database: " . $conn->error;
}
$conn->close ();
?>
```

OR

Steps using phpMyAdmin
Step 1: Click on Start and select XAMPP from the list. Open Xampp control panel by clicking on the option from the list. The Control Panel is now visible and can be used to initiate or halt the working of any module.
Step2: Click on the "Start" button corresponding to Apache and MySQL modules. Once it starts working, the user can see the following screen:
Step 3: Now click on the "Admin" button corresponding to the MySQL module. This automatically redirects the user to a web browser to the following address - http://localhost/phpmyadmin
Step 4: Screen with multiple tabs such as Database, SQL, User Accounts, Export, Import, Settings, etc. Will appear. Click on the "Database" tab. Give an appropriate name for the Database in the first textbox and click on create option.
Step 5 : In the created Database, click on the 'Structure' tab. Towards the end of the tables list, the user will see a 'Create Table' option. Give appropriate "Name" and "Number of Columns" for table and click on 'Go' button.
Step 6 : Give details of columns based on their type. Enter the names for each column, select the type, and the maximum length allowed for the input field. Click on "Save" in the bottom right corner. The table with the initialized columns will be created.

11.Write a program for cloning of an object. Explain narrow and deep cloning
Ans:

```
<?php
class MyClass {
  public
  $amount;
}

// Create an object with a reference
$value = 5;
$obj = new MyClass();
```

```
$obj->amount = &$value;

// Clone the object
$copy = clone $obj;
/ Change the value in the original object
$obj->amount = 6;

// The copy is
changed
print_r($copy);
?>
```

Shallow Clone:
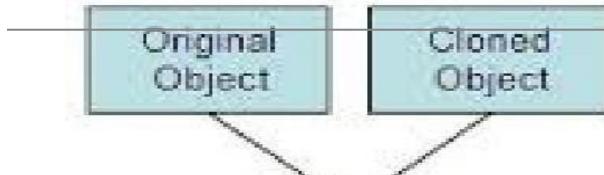
In **shallow copy** a new object is created.

The new object is an exact copy of the value in the original
object.It calls the object's " clone()" method.
It simply makes a copy of the reference to A to
B.It is copy of A's address.
The addresses of A and B will be same ie. they will be pointing to the same memory location.

Deep Clone:
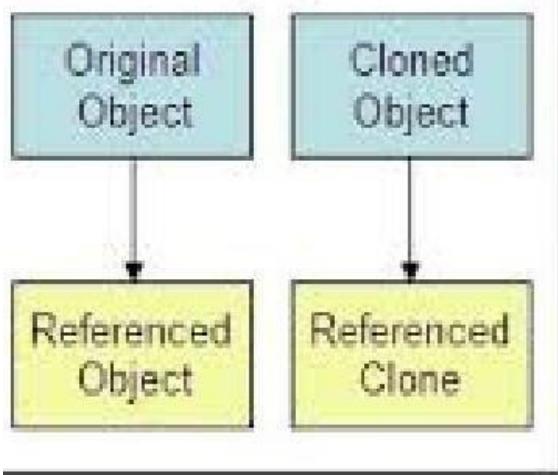
In this the data is actually completely copies.

In this everything is duplicated and all values are copies into a new instances.

Advantage of deep copy is that the A & B do not depend on each other but the process is relatively slower and more expensive.

In shallow copy B points to object A's memory location whereas in deep copy all things in object A's memory location get copied to object B's location.

12.Write a program to connect PHP with MYSQL.

Ans:

Solution1:

```php
<?php
$servername = "localhost";
$username = "root";
$password = "";
// Connection
$conn = new mysqli($servername,$username, $password);
// For checking if connection issuccessful or not
if ($conn->connect_error)
{
die("Connection failed: ". $conn->connect_error);
}
```

```
echo "Connected successfully";
?>
```
Output:
Connected successfully

<div align="center">OR</div>

Solution2:
Create login.php
```php
<?php
$hostname = 'localhost';
$username = 'root';
$password = '';
?>
```

Create db2.php file
```php
<?php
require_once 'login.php';
$conn = new mysqli($hostname, $username, $password);
//if ($conn->connect_error) die($conn->connect_error);
if ($conn->connect_error) {
die("Connection failed: "
. $conn->connect_error);
}
echo "Connected successfully";
?>
```
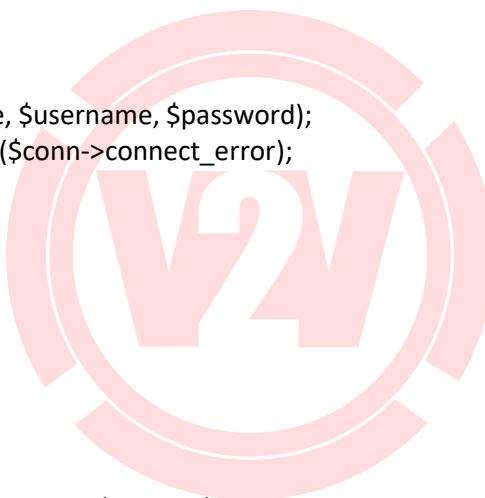
13.Illustrate class inheritance in PHP with example.
Ans:
• Inheritance is a mechanism of extending an existing class where a newly created or derived class have all functionalities of existing class along with its own properties and methods.
• The parent class is also called a base class or super class. And the child class is also known as a derived class or a subclass.
• Inheritance allows a class to reuse the code from another class without duplicating it.
• Reusing existing codes serves various advantages. It saves time, cost, effort, and increases a program's reliability.
• To define a class inherits from another class, you use the extends keyword.
• Types of Inheritance:
Single Inheritance
Multilevel Inheritance
Multiple Inheritance

Hierarchical Inheritance

Example:
(Any type of inheritance example shall be considered)

```php
<?php
class student {
var $var = "This is first var";
protected $fist_name;
protected $last_name;

// simple class method
function returnVar() {
echo $this->fist_name;
}
function set_fist_name($fname,$lname){
$this->fist_name = $fname;
$this->last_name = $lname;
}
}
class result extends student {
public $percentage;

function set_Percentage($p){
$this->percentage = $p;
}
function getVal(){
echo "Name:$this->fist_name $this->last_name";
echo "</br>";
echo "Result: $this->percentage %";
}
}
$res1 = new result();
$res1->set_fist_name("Rita","Patel");
$res1->set_Percentage(95);
$res1->getVal();
?>
```

Output:
Name:Rita Patel
Result: 95 %
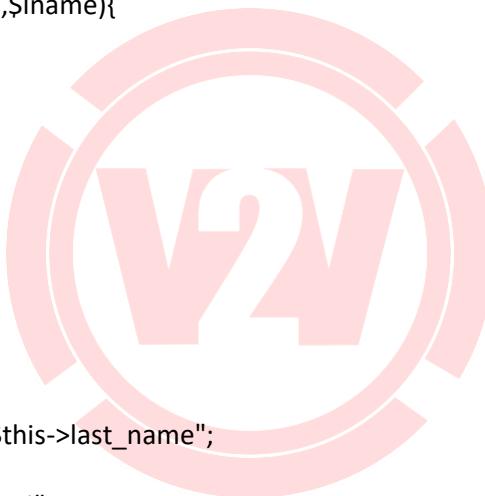
14.Enlist the attributes of cookies & Write a PHP program to set and modify cookies.

Ans:

PHP program to set cookies

```php
<html>
<body>
<?php
$cookie_name = "username";
$cookie_value = "abc";
setcookie($cookie_name, $cookie_value, time() + (86400 * 30), "/"); // 86400 = 1 day
if(!isset($_COOKIE[$cookie_name])) {
echo "Cookie name '" . $cookie_name . "' is not set!";
}
else
{
echo "Cookie '" . $cookie_name . "' is set!"<br>";
echo "Value is: " . $_COOKIE[$cookie_name];
}
?>
```

Output:

Cookie 'username' is set!

Value is: abc

PHP program to modify cookies

```php
<?php
setcookie("user","xyz");
?>
<html>
<body>
<?php
if(!isset($_COOKIE["user"]))
{
echo "Sorry, cookie is not found!";
} else {
echo "</br>Cookie Value: "
$_COOKIE["user"];
}
?>
</body>
</html>
```

Output:

Cookie Value: xyz

15.i) State the use of serialization.
ii) State the query to insert data in the database.
Ans:
i) Use of serialization.
Serializing an object means converting it to a bytestream representation that can be stored in a file. Serialization in PHP is mostly automatic, it requires little extra work from you, beyond calling the serialize () and unserialize( ) functions.

Serialize() :
• The serialize() converts a storable representation of a value.
• The serialize() function accepts a single parameter which is the data we want to serialize and returns a serialized string.
• A serialize data means a sequence of bits so that it can be stored in a file, a memory buffer or transmittedacross a network connection link. It is useful for storing or passing PHP values around without losing their type and structure.

Example:
```php
<?php
$s_data= serialize(array('Welcome', 'to', 'PHP'));
print_r($s_data . "<br>");
$us_data=unserialize($s_data);
print_r($us_data);
?>
```
Output:a:3:{i:0;s:7:"Welcome";i:1;s:2:"to";i:2;s:3:"PHP";}
Array ([0] => Welcome [1] => to [2] => PHP)

ii) Query to insert data in the database
```php
<?php
require_once 'login.php';
$conn = newmysqli($hostname,$username, $password,$dbname);
$query = "INSERT INTO studentinfo(rollno,name,percentage)
VALUES
('CO103','Yogita Khandagale',98.45)";
$result = $conn->query($query);
if (!$result)
die ("Database access failed: " . $conn->error);
```

else
echo "record inserted successfully";
?>

Output:
record inserted successfully

16.Define function. How to define user defined function in PHP?
Ans:
Definition: -A function is a block of code written in a program to perform some specific task. They take information as parameters, execute a block of statements or perform operations on these parameters and return the result. A function will be executed by a call to the function.

Define User Defined Function in PHP: A user-defined function declaration starts with the keyword function.

Syntax
function functionName() {
code to be executed;
}
Example
<?php
function writeMsg() {
echo "Welcome to PHP world!";
}
writeMsg(); // call the function
?>
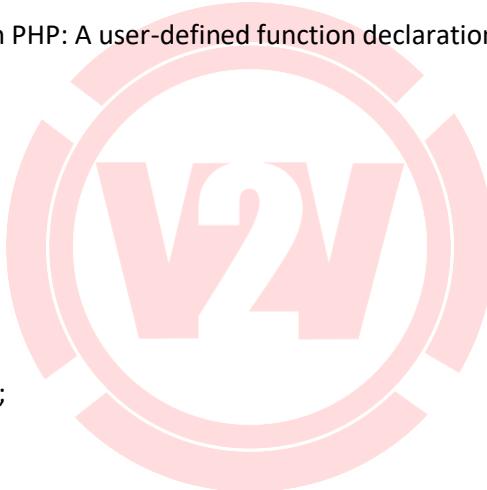
17.Explain method overloading with example.
Ans:
Function overloading or method overloading is the ability to create multiple functions of the same name with different implementations depending on the type of their arguments.
In PHP overloading means the behavior of a method changes dynamically according to the input parameter.
__call() is triggered when invoking inaccessible methods in an object context.
__callStatic() is triggered when invoking inaccessible methods in a static context.

__call():

If a class execute __call(), then if an object of that class is called with a method that doesn't exist then__call() is called instead of that method.

example:-

```php
<?php
// PHP program to explain function
// overloading in PHP
// Creating a class of type shape
class shape {
// __call is magic function which accepts
// function name and arguments

function __call($name_of_function, $arguments) {
// It will match the function name
if($name_of_function == 'area') {
switch (count($arguments)) {
// If there is only one argument
// area of circle
case 1:
return 3.14 * $arguments[0];

// IF two arguments then area is rectangle;
case 2:
return $arguments[0]*$arguments[1];
}
}
}
}
// Declaring a shape type object
$s = new Shape;

// Function call
echo($s->area(2));
echo "<br>";

// calling area method for rectangle
echo ($s->area(4, 2));
?>
```

Output:
9.426
48

Here the area() method is created dynamically and executed with the help of magic method __call() and its behavior changes according to the pass of parameters as object.

18.Explain delete operation of PHP on table data.
Ans:
Delete command is used to delete rows that are no longer required from the database tables. It deletes the whole row from the table.

The DELETE statement is used to delete records from a table:
DELETE FROM table_name WHERE some_column = some_value
[WHERE condition] is optional. The WHERE clause specifies which record or records that should be deleted. If the WHERE clause is not used, all records will be deleted.
Below is a simple example to delete records into the student.pridata table. To delete a record in any table it is required to locate that record by using a conditional clause. Below example uses name to match a record in student.pridata table.
Example:-
Assume Following Table
Database Name-student
Table name – pridata

name	age	gender	email	phone
amit	22	male	samplemail171019@gmail.com	2147483647
Raj	25	male	raj@gmail.com	1234567891
Swapna	36	female	swapna@gmail.com	2147483647

```php
<?php
    $server='localhost';
    $username='root';
    $password='';
    $con=mysqli_connect($server,$username,$password);if(!$con){
 die("Connection    to    this    database    failed    due    to"
 .mysqli_connect_error($mysqli));
    }
    $sql="DELETE FROM student.pridataWHERE name='amit'";if($con-
    >query($sql)==true){
       echo "Record deleted successfully";
       }
 else{
       "ERROR:error".$con->error();
    }
    $con->close();
```

?>

Output:-

Record deleted sucessfully
Table after Deletion

name	age	gender	email	phone
Raj	25	male	raj@gmail.com	1234567891
Swapna	36	female	swapna@gmail.com	2147483647

19.Write PHP script to sort & reverse any five numbers using array function.
Ans: sort numbers:

```php
<?php
    $a = array(1, 8, 9, 4, 5);sort($a);
 foreach($a as $i) {echo $i.' ';
    }
?>
```

OR

```html
<!DOCTYPE html>
<html>
<body>
<H1>Enter five numbers </H1>
<form action = 'sort.php' method = 'post'>
<input type = 'number' name = 'n1' placeholder = 'Number1...'><br><br>
<input type = 'number' name = 'n2' placeholder = 'Number2...'><br><br>
<input type = 'number' name = 'n3' placeholder = 'Number3...'><br><br>
<input type = 'number' name = 'n4' placeholder = 'Number4...'><br><br>
<input type = 'number' name = 'n5' placeholder = 'Number5...'><br><br>
<input type = 'submit' value = 'Submit'>
</form>
</body>
</html>
```

sort.php

```php
<?php
    if($_SERVER['REQUEST_METHOD'] == 'POST') {
```

```
    $a  =  array($_POST['n1'],  $_POST['n2'],  $_POST['n3'],
 $_POST['n4'], $_POST['n5']);
    sort($a);
    }
                          foreach($a as $i) {
                                echo $i.' ';
    }
  ?>
```

20.Create customer form like customer name, address, mobile no, date of birth using different form of input elements & display user inserted values in new PHP form
Ans:

```
 <!DOCTYPE html>
 <html>
 <body>
 <form action = 'data..php' method = 'post'>
 <input type = 'text' name = 'name' placeholder = 'CustomerName...'><br><br>
 <input  type  =  'text'  name  =  'address'  placeholder  ='Address...'><br><br>
 <input type = 'text' name = 'number' placeholder = 'MobileNumber...'><br><br>
 <label> Date of Birth: </label>
 <input type = 'date' name = 'dob'><br><br>
 <input type = 'submit' value = 'Submit'><br>
 </form>
 </body>
 </html>

 data.php
 <?php
   if($_SERVER['REQUEST_METHOD'] == 'POST') {
     echo '<html><body><form>
     Customer Name: '.$_POST['name'].'<br>Address:
     '.$_POST['address'].'<br>
     Mobile Number: '.$_POST['number'].'<br>Date of Birth:
     '.$_POST['dob'];
   }
?>
```

21.Inserting and retrieving the query result operations

Ans:

```php
<?php
$con = mysqli_connect('localhost', 'root', '', 'class');
        # Connecting to Database
$query = "insert into user values(1, 'Amit')";
        # Inserting Values
$result = mysqli_query($con, $query);
if($result) {
echo 'Insertion Successful <br>';
}
else {
echo 'Insertion Unsuccessful <br>';
}

$query = "select * from user";
        # Retrieving Values
$result = mysqli_query($con, $query);

foreach($result as $r) {
echo $r['roll_number'].' '.$r['name'];
}
?>
```

Output

Insertion Successful

1 Amit

Explanation

The above code connects with a database named „class‟‟.

The exam database has a table named „user‟ with 2 columns roll_number and name.

It executes an insert query on the user and checks whether the insertion was successful or not.

It executes a select query on the user and displays the information retrieved.

22.How do you validate user inputs in PHP.

Ans:

Invalid user input can make errors in processing. Therefore, validating inputs is a **must**.

1. Required field will check whether the field is filled or not in the proper way. Most of cases we will use the * symbol for requiredfield.
2. Validation means check the input submitted by the user.

There are two types of validation available in PHP.
Client-Side Validation − Validation is performed on the clientmachine web browsers.
Server Side Validation − After submitted by data, The data is sentto a server and performs validation checks in the server machine.

Some of Validation rules for field Field   Validation Rules
Name   Should required letters and white-spaces
Email   Should be required @ and .
Website   Should required a valid URL
Radio   Must be selectable at least once
Check Box   Must be checkable at least onceDrop Down menu   Must be selectable at least once

The preg_match() function searches a string for pattern, returningtrue if the pattern exists, and false otherwise.

To check whether an email address is well-formed is to use PHP's **filter_var() function.**

empty() function will ensure that text field is not blank it is with some data, function accepts a variable as an argument and returns TRUE when the text field is submitted with empty string, zero, NULL or FALSE value.

Is_numeric() function will ensure that data entered in a text field isa numeric value, the function accepts a variable as an argument andreturns TRUE when the text field is submitted with numeric value.

Example:-
Validations for: - name, email, phone no, website url
```
<!DOCTYPE html>
<body>
<?php
 $nerror = $merror = $perror = $werror = $cerror = "";
 $name = $email = $phone = $website = $comment = "";
$pattern   =   "^[_a-z0-9-]+(\.[_a-z0-9-]+)*  @[a-z0-9-]+(\.[a-z0-9-]+)*(\.[a-z]{2,3})$^";
if($_SERVER["REQUEST_METHOD"]=="POST") {
if(empty($_POST["name"])) {
$nerror = "Name cannot be empty!";
}
else {
```

```php
$name = test_input($_POST["name"]); if(!preg_match("/^[a-zA-Z-
']*$/",$name))
 {
$nerror = "Only characters and white spaces allowed";
 }
}
if(empty($_POST["email"])) {
$merror = "Email cannot be empty!";
}
else
{
$email = test_input($_POST["email"]);if(!preg_match($pattern,
$email)) {
$merror = "Email is not valid";
 }
}
if(empty($_POST["phone"])) {
$perror = "Phone no cannot be empty!";
}
else {
$phone = test_input($_POST["phone"]);
if (!preg_match ('/^[0-9]{10}+$/', $phone)) {
$perror = "Phn no is not valid";
}
}
if(empty($_POST["website"])) {
$werror = "This field cannot be empty!";
}
else {
$website = test_input($_POST["website"]);
if            (!preg_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-
9+&@#\/%?=~_|!:,.;]*[-a-z0-9+&@#\/%=~_|]/i",$website)) {
$werror = "URL is not valid";
}
 }
 if (empty($_POST["comment"])) {
$cerror = "";
 }
 else {
    $comment = test_input($_POST["comment"]);
 } }
```

```
function test_input($data)
{
$data = trim($data);
$data = stripslashes($data);
$data = htmlspecialchars($data);return $data;
}
?>
<p><span class="error">* required field </span></p>
<form          method="post"          action="<?php          echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
Name: <input type="text" name="name">
<span class="error">* <?php echo $nerror;?></span><br/><br/>E-mail: <input type="text"
name="email">
<span class="error">* <?php echo $merror;?></span><br/><br/>Phone no: <input
type="text" name="phone">
<span class="error">* <?php echo $perror;?></span><br/><br/>Website: <input
type="text" name="website">
<span class="error">* <?php echo $werror;?></span><br/><br/> Comment:      <textarea
            name="comment"                          rows="5"
cols="40"></textarea><br/><br/>
<input type="submit" name="submit" value="Submit"></form>
<?php
echo "<h2>Your Input:</h2>";echo $name; echo "<br>";
echo $email; echo "<br>"; echo $phone; echo "<br>";
echo $website; echo "<br>"; echo $comment;
?>
</body>
</html>
```

23.Explain different loops in PHP with example.

Ans:

> **while loop:** If the expression/condition specified with while
> evaluates to true then the statements inside loop executes and
> control is transferred back to expression/condition. The process of
> evaluation and execution
> **Example:**
> <?php
> $a=1;
> while($a<=5)

```
{
        echo " Iteration $a";
        $a++;
}
?>
```

OR

Example:
```
<?php
$a=1;
while($a<=5):
        echo " Iteration $a";
        $a++;
endwhile;
?>
```

do-while loop: All the statements inside the loop executes for the first time without checking any condition. The keyword „do‟ passes the flow of control inside loop. After executing loop for the first time, expression / condition is evaluated. If it evaluates to true then all statements inside loop again executes and if it evaluates to false then loop exits and flow of control passes to the next statement placed outside the loop. The process of execution and evaluation continues till expression / condition evaluates to true.

Example:
```
<?php
$a=1;
do
{
print("Iteration 1");
$a++;
}while($a<=0);
?>
```

for loop:It is used to execute same set of statements multiple times. In for loop variable initialization, condition and increment / decrement is placed in a single statement. Before starting first iteration a variable is initialized to specified value. Then condition is checked. If condition is true then statements inside the loop executes and variable is incremented or decremented. Control then passes to condition. If condition is false then control passes to the statement placed outside the loop. The process of condition checking, loop

statement execution and increment /decrement continues till condition is true.
Example :
```php
<?php
for ($a=1;$a<=5;$a++)
{
echo("Iteration $a");
}
?>
```

for each loop: This loop works with arrays and is used to traverse through values in an array. For each loop iteration, the value of the current array element is assigned to $value and the array pointer is moved by one, until it reaches the last array element.
Example :
```php
<?php
$arr=array("Apple","Banana","Orange"); foreach($arr as $fruit)
{
echo("$fruit");
}
?>
```

24.Explain any four string functions in PHP with example.
Ans:
1. str_word_count() function: This function is used to count thenumber of words in a string.
syntax : str_word_count(string,return,char); string : It indicates string to be checked.
return :It is optional. It specifies the return value of thefunction.

  0- default. Returns the number of words found.

   1- returns an array with the words from the string.

   2- returns an array where the key is the position ofthe word in the string, and value is the actual word.

  char : Optional. It specifies special characters to beconsidered as words.

Example:

```php
<?php
$str1="Welcome to WBP Theory & practical";echo " <br> Total words in
string str1= ".str_word_count($str1,0,"&");
?>
```

2. strlen() function : This function is used to find number ofcharacters in a string . While counting number characters fromstring, function also considers spaces between words.
syntax : strlen(string);
      - string specify name of the string from which charactershave to be counted.
Example :

```php
<?php
$str3="Hello,welcome to WBP";
echo "<br> Number of characters in a string '$str3' = "
.strlen($str3);
?>
```

3. strrev() function : This function accepts string as an argumentand returns a reversed copy of it.
Syntax : $strname=strrev($string variable/String);
Example :

```php
<?php
$str4="Polytechnic";
$str5=strrev($str4);
echo "Orginal string is '$str4' and reverse of it is '$str5'";
?>
```

4. strcmp() function : This function is used to compare two strings
with each other . It is a case sensitive comparison.
Syntax : $result= strcmp(string1,string2);
      - string1 and string2 indicates strings to be compared witheach other.
      -This function returns 0 if both the strings are equal. Itreturns a value <0 if string1 is less than string2 and >0 if string 1 is greater than string2
Example 1 :

```php
<?php
$str6="Welcome";
$str7="Welcome";
echo strcmp($str7,$str6);
?>
```

5. strpos() function : This function is used to find the position ofthe first occurrence of specified word inside another string. Itreturns False if word is not present in string. It is a case sensitivefunction. by default, search starts with 0th position in a string. **Syntax** : strpos(String,findstring,start);
      - string specify string to be searched to find another word
      - findstring specify word to be searched in specified firstparameter.
      - start is optional . It specifies where to start the search in a string. If start is a negative

number then it counts from the end of the string.

Example:
```php
<?php
$str8="Welcome to Polytechnic";
$result=strpos($str8,"Poly",0);echo $result;
?>
```

6. str_replace() function : This function is used to replace somecharacters with some other characters in a string.

Syntax : str_replace(findword,replace,string,count);
- Find word specify the value to find
- replace specify characters to be replaced with searchcharacters.

string specify name of the string on which find and replacehas to be performed.
- count is optional . It indicates number of occurrencesreplaced from a string.

Example 1:
```php
$str10="Welcome to poly";
$str11=str_replace("poly","msbte",$str10);echo $str11;
```

7. ucwords() function: This function is used to convert firstcharacter of each word from the string into uppercase.

Syntax : $variable=ucwords($Stringvar);

Example :
```php
<?php
$str9="welcome to poly for web based development";echo ucwords($str9);
?>
```

8. strtoupper() function :This function is used to convert anycharacter of string into uppercase.

Syntax : $variable=strtoupper($stringvar);

Example:
```php
<?php
$str9="POLYtechniC";echo strtoupper($str9);
?>
```

9. strtolower() function : This function is used to convert anycharacter of string into lowercase.

Syntax: $variable=strtolower($stringvar);

Example :
```php
<?php
$str9="POLYtechniC";echo strtolower($str9);
```

?>

25.Develop a simple program for sending and receiving plain text message. / Describe the procedure of sending e-mail

Ans:

Sending mail using PHP

The PHP mail function has the following basic syntax

```php
<?php
mail($to_email_address,$subject,$message,[$headers],[$parameters
]);
?>
```

HERE,

"$to_email_address" is the email address of the mail recipient

"$subject" is the email subject

"$message" is the message to be sent.

"[$headers]" is optional, it can be used to include information such as CC, BCC

CC is the acronym for carbon copy. It's used when you want to send a copy to an interested person i.e. a complaint email sent to a company can also be sent as CC to the complaints board. BCC is the acronym for blind carbon copy. It is similar to CC. The email addresses included in the BCC section will not be shown to the other recipients.

Simple Mail Transmission Protocol (SMTP)

PHP mailer uses Simple Mail Transmission Protocol (SMTP) to send mail.On a hosted server, the SMTP settings would have already been set.The SMTP mail settings can be configured from **"php.ini"** & "**sendemail.ini** " file in the PHP installation folder.

1. File :-Php.ini

```
//email function
smtp_port=465
sendmail_from = testmailkale@gmail.com
sendmail_path = "\"C:\xampp\sendmail\sendmail.exe\"
-t" mail.add_x_header=on
```

2. **File:- sendemail.ini**

smtp_server=smtp.gmail.c

om smtp_port=465

smtp_ssl=auto

error_logfile=error.log

auth_username=testmailkale@gmail.com

auth_password=

26. Explain constructor and destructor in PHP.

Ans:

To create and initialize a class object in a single step, PHP provides a special method called as Constructor, which is used to construct the object by assigning the required property values while creating the object.

And for destroying the object, the Destructor method is used.

Syntax for defining Constructor and Destructor

__construct() and __destruct().

```php
<?php
   class <CLASS_NAME> {

      // constructor
      function__construct() {
         // initialize the object properties
      }

      // destructor
      function__destruct() {
         // clearing the object reference
      }
  }
?
>
```

Constructor can accept arguments, whereas destructors won't have any argument because a destructor's job is to destroy the current object reference.

33

PHP Constructor:-Let's take the example of a class Person which has two properties, fname and lname, for this class we will define a constructor for initialising the class properties(variables) at the time of object creation.

```php
<?php
    class Person {
// first name of person
private $fname;

        // last name of
        personprivate
        $lname;

        // Constructor
        public function_construct($fname, $lname) {
            echo "Initialising the object...<br/>";
            $this->fname = $fname;
            $this->lname = $lname;
        }

        // public method to show
        namepublic function
        showName() {
            echo "The Legend of India: " . $this->fname . " " . $this->lname;
        }
    }

    // creating class object
    $j = new Person("Bhagat", "Singh");
    $j->showName();

?>
```

While earlier, we were using the -> operator to set values for the variables or used the setter methods, in case of a constructor method, we can assign values to the variables at the time of object creation.If a class has a constructor then whenever an object of that class is created, the

34

constructor is called.

PHP Destructor:-PHP Destructor method is called just before PHP is about to release any object from its memory. Generally, you can close files, clean up resources etc in the destructor method. Let's take an example,

```php
<?php
  class Person {
    // first name of
    personprivate
    $fname;
    // last name of
    personprivate
    $lname;

    // Constructor
    public function _construct($fname, $lname) {
      echo "Initialising the object...<br/>";
      $this->fname = $fname;
      $this->lname = $lname;
    }

    // Destructor
    public function __destruct(){
      // clean up resources or do something
      elseecho "Destroying Object...";
    }

    // public method to show
    namepublic function
    showName() {
      echo "The Legend of India: " . $this->fname . " " . $this->lname . "<br/>";
    }
  }

  // creating class object
  $j = new Person("Swami", "Vivekananda");
  $j->showName();

?>
```

As we can see in the output above, as the PHP program ends, just before it PHP initiates the release of the object created, and hence the destructor method is called.The destructor methodcannot accept any argument and is called just before the object is deleted, which happens either when no reference exist for an object or when the PHP script finishes its execution.

27.Differentiate between session and cookies.
Ans:

28.Develop a PHP program for overriding.
Ans:
```php
<?php
class
aicte {
function
helloWorld() {echo
"Parent"."<br>";
}
}
class msbte extends
aicte {function
helloWorld() { echo
"\nChild";
}
}
$p = new aicte;
$c= new msbte;
$p->helloWorld();
$c->helloWorld();
?>
```

29.Explain procedure to create PDF in PHP. Write example.

Ans:

FPDF is an open source library which is used for creating a PDF document.it is open source.Link to Download latest version of FPDF class: http://www.fpdf.org/en/download.php Features of fpdf

1)It is an open source package,hence freely available on internet

2)it provides the choice of measure unit,page format and margins for pdf page

3)it provides page header and footer management.

4)It provides automatic page breaks to the pdf document

5)it provides the support for various fonts,colors,encoding and image formate.

Example

```php
<?php
require('fpdf183/fpdf.php');
$pdf=new FPDF();
$pdf->AddPage();
$pdf->SetFont('Arial','B',16);
$pdf->Cell(60,10,'Hello PHP World!',1,1,'C');
$pdf->Output();
?>
```

30.Develop a PHP program to create constructor to initialize object of class.

Ans:

```php
<?php
class Person {
//first name of person
private $fname;
//last name of person
private $lname;
// Constructor
public function construct($fname, $lname) {
echo "Initialising the object...</br>";
$this->fname = $fname;
$this->lname = $lname;
}

// publicmethod to show name
public function showName() {
echo "The Legend of India: " . $this->fname . " " . $this->lname;
}
}
```

```
// creating class object
$j = new Person("Bhagat", "Singh");
$j->showName();
?>
```

31.State the advantages of PHP
Ans:
Advantages of PHP :
1. Most important advantage of PHP is that it's open source and freed from cost. It is often downloaded anywhere and readily available to use for web applications.
2. It is platform independent. PHP based applications can run on any OS like UNIX, Linux and windows, etc.
3. Applications can easily be loaded which are based on PHP and connected to a database. it's mainly used due to its faster rate of loading over slow internet and speed than other programming languages.
4. It has less learning curve, because it is straightforward and straightforward to use. If a private knows C programming can easily work on PHP.
5. It is more stable from a few years with assistance of providing continuous support to various versions.
6. It helps in reusing an equivalent code and no got to write lengthy code and sophisticated structure for the event of web applications.
7. It helps in managing code easily.
8. It has powerful library support to use various function modules for data representation.
9. PHP's built-in database connection modules help in connecting databases easily and reduce trouble and time for development of web applications and content based sites.
10. Popularity of PHP gave rise to various communities of developers, a fraction of which may be potential candidates for hire.

32.Differentiate between implode and explode functions

33.Create a web page using GUI components.
Ans:
```
<form method="post">
<h3>personal information form</h3>
user name:<input type="text" name="username"/>
<br/><br/>
address:<input type="text" name="address"/>
```

```
<br/><br/>
<input type="submit" name="submit_personal_info" value="submit"/>
<br/>
</form>
<form method="post">
<h3>Feedback Form</h3>
<textarea name="feedback" rows="5" cols="50"></textarea>
<br/>
<input type="submit" name="submit_feedback"/>
</form>
<?php
if(!empty($_POST['submit_personal_info'
]))
{

        echo "<h3>welcome user:".$_POST['username']."</h3>";
 }

if(!empty($_POST['submit_feedback']))
{
echo "<h3>we value your feedback:</h3>";
echo "your feedback is<br/>".$_POST["feedback"];
}

?>
```