1.Implement a program to create a session, update, and destroy the same session.

<?php

session\_start();

if (isset($\_POST['create'])) {

$\_SESSION['user'] = "John";

}

if (isset($\_POST['update'])) {

$\_SESSION['user'] = "Jane";

}

if (isset($\_POST['destroy'])) {

session\_unset();

session\_destroy();

}

?>

<!DOCTYPE html>

<html>

<body>

<h2>Session Demo (Simple)</h2>

<form method="post">

<input type="submit" name="create" value="Create Session">

<input type="submit" name="update" value="Update Session">

<input type="submit" name="destroy" value="Destroy Session">

</form>

<p>

<?php

if (isset($\_SESSION['user'])) {

echo "Session user: " . $\_SESSION['user'];

} else {

echo "No session set.";

}

?>

</p>

</body>

</html>

2. Implement a program to use validation for following controls

a) Username

b) Email address

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$username = $\_POST["username"];

$email = $\_POST["email"];

$usernamePattern = "/^[a-zA-Z0-9]{3,}$/";

$emailPattern = "/^[\w\-\.]+@([\w\-]+\.)+[a-zA-Z]{2,7}$/";

if (empty($username) || empty($email)) {

echo "All fields are required.";

} elseif (!preg\_match($usernamePattern, $username)) {

echo "Invalid username. Use letters and numbers only (min 3 chars).";

} elseif (!preg\_match($emailPattern, $email)) {

echo "Invalid email format.";

} else {

echo "Username: $username<br>Email: $email";

}

}

?>

<form method="post">

Username: <input type="text" name="username"><br><br>

Email: <input type="text" name="email"><br><br>

<input type="submit" value="Submit">

</form>

3.Develop a PHP program to Enter data into table and Retrieve and present data from

Table

<?Php

$servername="localhost";

$username="root";

$password="";

$db="Student";

$conn=new mysqli($servername,$username,$password,$db);

if($conn->connect\_error)

{

die("Connection is not established".$conn->connect\_error);

}

$sql="INSERT INTO student(name1,roll)VALUES('Mayuri',62)";

if($conn->query($sql)==TRUE)

{

echo " data inserted";

}

else

{

echo " data not inserted";

}

$sql="SELECT \* FROM student";

$result=$conn->query($sql);

if($result->num\_rows>0)

{

while($row=$result->fetch\_assoc())

{

echo "name1".$row["name1"]." roll no".$row["roll"];

}

}

else{

echo "Connection is not established".$conn->connect\_error;

}

$conn->close();

?>

4. Implement a simple PHP program to create a PDF documents by using Graphics

concepts.

<?php

require('fpdf.php');

$pdf = new FPDF();

$pdf->AddPage();

$pdf->SetFont('Arial', '', 12);

$text = "This is a simple example of using MultiCell in FPDF. It allows you to write long text in a block that wraps automatically when it reaches the edge of the page.";

$pdf->MultiCell(0, 10, $text);

$pdf->Output();

?>

5. Implement a PHP program to check prime number

<?php

$number = 7;

$isPrime = true;

if ($number <= 1) {

$isPrime = false;

} else {

for ($i = 2; $i < $number; $i++) {

if ($number % $i == 0) {

$isPrime = false;

break;

}

}

}

if ($isPrime) {

echo "$number is a Prime Number";

} else {

echo "$number is Not a Prime Number";

}

?>

6. Implement a simple PHP program to check that email id is valid or not.

<?php

$email = "example@mail.com";

if (preg\_match("/^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-z]{2,}$/", $email)) {

echo "$email is a valid email address.";

} else {

echo "$email is not a valid email address.";

}

?>

7. Implement a PHP program to demonstrate session management

<?php

session\_start();

if (!isset($\_SESSION['counter'])) {

$\_SESSION['counter'] = 1;

echo "Session started. Counter is set to 1.";

} else {

$\_SESSION['counter']++;

echo "Session counter: " . $\_SESSION['counter'];

}

if ($\_SESSION['counter'] > 5) {

session\_destroy();

echo "<br>Session destroyed after 5 counts.";

}

?>

8. Implement a PHP script to Implement any four String Functions

<?php

$text = "Hello World! PHP is Fun.";

// 1. strlen() – Get string length

echo "Length of string: " . strlen($text) . "<br>";

// 2. strrev() – Reverse the string

echo "Reversed string: " . strrev($text) . "<br>";

//3. strtoupper() – Convert to uppercase

echo "Uppercase: " . strtoupper($text) . "<br>";

//4. strtolower() – Convert to lowercase

echo “lowercase: " . strtolower($text) . "<br>";

//5.unword()-Capitalized word

echo "String with each word capitalized:".ucwords($text)."<br>";

?>

9. Implement a PHP script to Implement eight Math Functions

<?php

$num1 = 16;

$num2 = 4;

// 1. abs() – Absolute value of a number

echo "Absolute value of -$num1: " . abs(-$num1) . "<br>";

// 2. pow() – Power of a number (num1 raised to num2)

echo "$num1 raised to the power of $num2: " . pow($num1, $num2) . "<br>";

// 3. sqrt() – Square root of a number

echo "Square root of $num1: " . sqrt($num1) . "<br>";

// 4. max() – Maximum of two numbers

echo "Maximum of $num1 and $num2: " . max($num1, $num2) . "<br>";

// 5. min() – Minimum of two numbers

echo "Minimum of $num1 and $num2: " . min($num1, $num2) . "<br>";

// 6. round() – Round a number

echo "Round 3.14159: " . round(3.14159) . "<br>";

// 7. rand() – Generate a random number

echo "Random number between 1 and 100: " . rand(1, 100) . "<br>";

// 8. deg2rad() – Convert degrees to radians

echo "90 degrees in radians: " . deg2rad(90) . "<br>";

?>

10. Implement a PHP script to replace the first substring of the with second.

<?php

$string = "Hello world, welcome to the world of PHP!";

$search = "world";

$replace = "universe";

$new\_string = substr\_replace($string, $replace, strpos($string, $search), strlen($search));

echo $new\_string;

?>

11. Implement a PHP program for sending and receiving plain text message (email).

<?php

$to ="ushaborade212@gmail.com";

$subject = "Simple Test mail";

$message = "Hello, this is a simple email sent using PHP.";

$headers = "From:mayurimatsagar75@gamil.com";

if (mail($to, $subject, $message, $headers)) {

echo "Email sent successfully!";

} else {

echo "Email sending failed.";

}

?>

12. Implement a program to check positive or negative no.

<?php

// Input number

$number = -10; // You can change this value for testing

// Check if the number is positive, negative, or zero

if ($number > 0) {

echo "$number is a positive number.";

} elseif ($number < 0) {

echo "$number is a negative number.";

} else {

echo "The number is zero.";

}

?>

13.Implement a PHP code to insert data into employee table.

<?php

$servername = "localhost";

$username = "root";

$password = "";

$db = "Employee";

$conn = new mysqli($servername, $username, $password, $db);

if ($conn->connect\_error) {

die("Connection is not established: " . $conn->connect\_error);

}

$sql = "CREATE TABLE IF NOT EXISTS employee (

id INT AUTO\_INCREMENT PRIMARY KEY,

empname VARCHAR(100) NOT NULL,

empid INT NOT NULL

)";

if ($conn->query($sql) === TRUE) {

echo "Table 'employee' is ready!<br>";

} else {

echo "Error creating table: " . $conn->error . "<br>";

}

$sql = "INSERT INTO employee (empname, empid) VALUES ('Mayuri', 612)";

if ($conn->query($sql) === TRUE) {

echo "Data inserted successfully.";

} else {

echo "Data not inserted: " . $conn->error;

}

$conn->close();

?>

14. Implement a PHP Program to create a login page

<!DOCTYPE html>

<html>

<head>

<title>Login Page</title>

</head>

<body>

<h2>Login Form</h2>

<form action="login.php" method="post">

<label>Username:</label>

<input type="text" name="username" required><br><br>

<label>Password:</label>

<input type="password" name="password" required><br><br>

<input type="submit" value="Login">

</form>

</body>

</html>

<?php

$correct\_username = "admin";

$correct\_password = "1234";

$username = $\_POST['username'];

$password = $\_POST['password'];

if ($username === $correct\_username && $password === $correct\_password) {

echo "Login successful! Welcome, $username.";

} else {

echo "Invalid username or password.";

}

?>

15. Implement a Program to print each element of an array using foreach().

<?php

// Define an array

$fruits = array("Apple", "Banana", "Mango", "Orange", "Grapes");

// Print each element using foreach

echo "List of Fruits:<br>";

foreach ($fruits as $fruit) {

echo $fruit . "<br>";

}

?>

16. Implement a program to show day of the week using switch

<?php

// Day number (1 to 7)

$dayNumber = 3; // Change this number to test different days

// Use switch to determine the day

switch ($dayNumber) {

case 1:

echo "Sunday";

break;

case 2:

echo "Monday";

break;

case 3:

echo "Tuesday";

break;

case 4:

echo "Wednesday";

break;

case 5:

echo "Thursday";

break;

case 6:

echo "Friday";

break;

case 7:

echo "Saturday";

break;

default:

echo "Invalid day number. Please enter a number between 1 and 7.";

}

?>

17. Implement a PHP program factorial program using for loop in php

<?php

$number = 5;

$factorial = 1;

for ($i = 1; $i <= $number; $i++) {

$factorial \*= $i;

}

echo "Factorial of $number is: $factorial";

?>

18. Implement a PHP program simple calculator using switch case

<?php

// Input values

$num1 = 20;

$num2 = 10;

$operator = '+'; // Change to '+', '-', '\*', '/'

// Calculator logic using switch case

switch ($operator) {

case '+':

$result = $num1 + $num2;

echo "Result: $num1 + $num2 = $result";

break;

case '-':

$result = $num1 - $num2;

echo "Result: $num1 - $num2 = $result";

break;

case '\*':

$result = $num1 \* $num2;

echo "Result: $num1 \* $num2 = $result";

break;

case '/':

if ($num2 != 0) {

$result = $num1 / $num2;

echo "Result: $num1 / $num2 = $result";

} else {

echo "Error: Division by zero is not allowed.";

}

break;

default:

echo "Invalid operator. Please use +, -, \*, or /.";

}

?>

19. Implement a PHP program to reverse the string without string function

<?php

// Original string

$original = "Hello World";

$reversed = "";

$length = 0;

while (isset($original[$length])) {

$length++;

}

for ($i = $length - 1; $i >= 0; $i--) {

$reversed .= $original[$i];

}echo "Original String: $original<br>";

echo "Reversed String: $reversed";

?>

20. 20. Implement a PHP program to count the words in the string

<?php

$str = "PHP is a popular scripting language.";

$wordCount = str\_word\_count($str);

echo "Original String: $str<br>";

echo "Word Count: $wordCount";

?>

21. Implement a PHP program to check entered year is Leap Year or not.

<?php

$year = 2024;

if (($year % 4 == 0 && $year % 100 != 0) || $year % 400 == 0) {

echo "$year is a Leap Year.";

} else {

echo "$year is not a Leap Year.";

}

?>

22. Implement a PHP program to print first 30 even number

<?php

$evenCount = 30;

for ($i = 1; $i <= $evenCount; $i++) {

echo $i \* 2 . "<br>";

}

?>

23. Implement a PHP program to find reverse of a number

<?php

$number = 12345;

$reverse = 0;

while ($number > 0) {

$digit = $number % 10;

$reverse = ($reverse \* 10) + $digit;

$number = (int)($number / 10);

}

echo "Reversed Number: $reverse";

?>

24. Implement a PHP program to implement associative array

<?php

$person = array(

"name" => "John Doe",

"age" => 30,

"email" => "johndoe@example.com",

"city" => "New York"

);

echo "Name: " . $person["name"] . "<br>";

echo "Age: " . $person["age"] . "<br>";

echo "Email: " . $person["email"] . "<br>";

echo "City: " . $person["city"] . "<br>";

?>

25. Implement a PHP program to implode &amp; explode functions

<?php

$array = array("Apple", "Banana", "Cherry", "Date");

$imploded = implode(", ", $array);

echo "Imploded Array: $imploded<br>";

$string = "Apple, Banana, Cherry, Date";

$exploded = explode(", ", $string);

echo "Exploded String: ";

print\_r($exploded);

?>

**26. Implement a PHP program to count no of words in string without string function**

**<?php**

function countWords($str) {

    $count = 0;

    $inWord = false;

    for ($i = 0; $i < strlen($str); $i++) {

        if ($str[$i] != ' ') {

            if (!$inWord) {

                $count++;

                $inWord = true;

            }

        } else {

            $inWord = false;

        }

    }

    return $count;

}

$text = "This is a simple string.";

$wordCount = countWords($text);

echo "Word count: $wordCount";

?>

27. Implement a PHP program to implement Single inheritance

<?php

class Vehicle {

public $brand;

public function \_\_construct($brand) {

$this->brand = $brand;

}

public function drive() {

echo "The $this->brand vehicle is driving.<br>";

}

}

class Car extends Vehicle {

public function honk() {

echo "The $this->brand car honks.<br>";

}

}

$car = new Car("Toyota");

$car->drive(); // Outputs: The Toyota vehicle is driving.

$car->honk(); // Outputs: The Toyota car honks.

?>

28. Implement a PHP program to delete 1 record from Student Database where RollNo=2

<?Php

$servername="localhost";

$username="root";

$password="";

$db="Student";

$conn=new mysqli($servername,$username,$password,$db);

if($conn->connect\_error)

{

die("Connection is not established".$conn->connect\_error);

}

$sql = "CREATE TABLE IF NOT EXISTS student (

id INT AUTO\_INCREMENT PRIMARY KEY,

name1 VARCHAR(100) NOT NULL,

roll INT NOT NULL

)";

if ($conn->query($sql) === TRUE) {

echo "Table 'student' is ready!";

} else {

echo "Error creating table: " . $conn->error;

}

$sql="Delete from student where roll =2";

if($conn->query($sql)==TRUE)

{

echo "Record with RollNo 2 deleted successfully.";

}

else

{

echo "Error";

}

$conn->close();

?>

29. Implement a PHP program to update 1 record from Student Database. Update name of

Student having RollNo=1

<?php

$servername = "localhost";

$username = "root";

$password = "";

$db = "Student";

$conn = new mysqli($servername, $username, $password, $db);

if ($conn->connect\_error) {

die("Connection is not established: " . $conn->connect\_error);

}

$sql = "CREATE TABLE IF NOT EXISTS student (

id INT AUTO\_INCREMENT PRIMARY KEY,

name1 VARCHAR(100) NOT NULL,

roll INT NOT NULL

)";

if ($conn->query($sql) === TRUE) {

echo "Table 'student' is ready!<br>";

} else {

echo "Error creating table: " . $conn->error . "<br>";

}

$sql = "UPDATE student SET name1='Mayuri' WHERE roll=1";

if ($conn->query($sql) === TRUE) {

echo "Update successful.";

} else {

echo "Error: " . $conn->error;

}

// Close connection

$conn->close();

?>

31. Implement a PHP program update and delete data in the database(assume sutable

Database)

<?Php

$servername="localhost";

$username="root";

$password="";

$db="Student";

$conn=new mysqli($servername,$username,$password,$db);

if($conn->connect\_error)

{

die("Connection is not established".$conn->connect\_error);

}

// SQL query to create table if it does not exist

$sql = "CREATE TABLE IF NOT EXISTS student (

id INT AUTO\_INCREMENT PRIMARY KEY,

name1 VARCHAR(100) NOT NULL,

roll INT NOT NULL

)";

// Execute the query

if ($conn->query($sql) === TRUE) {

echo "Table 'student' is ready!";

} else {

echo "Error creating table: " . $conn->error;

}

$sql="INSERT INTO student(name1,roll)VALUES('Ovee',2)";

if($conn->query($sql)==TRUE)

{

echo " data inserted";

}

else

{

echo " data inserted";

}

$sql = "UPDATE student SET name1='Gaurav' WHERE roll=1";

if ($conn->query($sql) === TRUE) {

echo "Update successful.";

} else {

echo "Error: " . $conn->error;

}

$sql="Delete from student where roll =55";

if($conn->query($sql)==TRUE)

{

echo "Record with RollNo 2 deleted successfully.";

}

else

{

echo "Error";

}

$conn->close();

?>

32. Implement a PHP program set and modify cookies.

<?php

setcookie("user", "Mayuri", time() + 3600);

if (isset($\_COOKIE["user"])) {

echo "Cookie value is: " . $\_COOKIE["user"];

} else {

echo "Cookie is not set yet.";

}

?>

33. Implement a PHP program any Six Method of Introspection

<?php

class Student {

public $name;

private $roll;

function \_\_construct($name, $roll) {

$this->name = $name;

$this->roll = $roll;

}

public function display() {

echo "Name: $this->name";

}

}

$student = new Student("Yash", 101);

echo "1. Class Name: " . get\_class($student) . "<br>";

echo "2. Class Methods: ";

print\_r(get\_class\_methods($student));

echo "<br>";

echo "3. Class Properties: ";

print\_r(get\_class\_vars("Student"));

echo "<br>";

echo "4. Object Properties: ";

print\_r(get\_object\_vars($student));

echo "<br>";

echo "5. Method 'display' exists: " . (method\_exists($student, "display") ? "Yes" : "No") . "<br>";

echo "6. Property 'roll' exists: " . (property\_exists($student, "roll") ? "Yes" : "No") . "<br>";

?>

34. Implement a PHP program insert data in the database.

<?Php

$servername="localhost";

$username="root";

$password="";

$db="Student";

$conn=new mysqli($servername,$username,$password,$db);

if($conn->connect\_error)

{

die("Connection is not established".$conn->connect\_error);

}

$sql = "CREATE TABLE IF NOT EXISTS student (

id INT AUTO\_INCREMENT PRIMARY KEY,

name1 VARCHAR(100) NOT NULL,

roll INT NOT NULL

)";

if ($conn->query($sql) === TRUE) {

echo "Table 'student' is ready!";

} else {

echo "Error creating table: " . $conn->error;

}

$sql="INSERT INTO student(name1,roll)VALUES('Yash',55)";

if($conn->query($sql)==TRUE)

{

echo " data inserted";

}

else

{

echo " data inserted";

}

$conn->close();

?>

35. Create a class as “Percentage” with two properties length &amp; width. Calculate

area of rectangle for two objects.

<?php

class Percentage {

public $length;

public $width;

function \_\_construct($length, $width) {

$this->length = $length;

$this->width = $width;

}

function getArea() {

return $this->length \* $this->width;

}

}

$rect1 = new Percentage(10, 5);

echo "Area of Rectangle 1: " . $rect1->getArea() . "<br>";

$rect2 = new Percentage(8, 7);

echo "Area of Rectangle 2: " . $rect2->getArea() . "<br>";

?>

36. Implement a PHP program method overloading and overriding

**Method overriding**

<?php

class Animal {

    public function sound() {

        print "<br>Animal sound";

    }

}

class Dog extends Animal {

    public function sound() {

        Animal::sound();

        print "<br>Dog bark";

    }

}

class Cat extends Animal {

    public function sound() {

        Animal::sound();

        print "<br>Dog meow";

    }

}

$cat=new Cat();

$cat->sound(); // Output: Dog meow

// $dog = new Dog();

// $dog->sound(); // Output: Dog bark

// $animal = new Animal();

// $animal->sound(); // Output: Animal sound

?>

**Method Overloading**

**<?php**

class Person {

    public function \_\_call($method, $args) {

        if ($method == 'display') {

            if (count($args) == 2) {

                $name = $args[0];

                $age = $args[1];

                return "Name: $name, Age: $age";

            } elseif (count($args) == 1) {

                $name = $args[0];

                return "Name: $name, Age: Not provided";

            } else {

                return "Invalid number of arguments for display method";

            }

        }

        return "Method $method not found!";

    }

}

$person = new Person();

echo $person->display("John", 25) . "<br>";

echo $person->display("Alice") . "<br>";

echo $person->display() . "<br>";

?>

37. Implement a PHP program to demonstrate use of cookies.

<?php

if (!isset($\_COOKIE["user"])) {

setcookie("user", "Yash", time() + 3600);

echo "Cookie 'user' is set!<br>";

echo "Please refresh the page to see the stored value.";

} else {

echo "Welcome back, " . $\_COOKIE["user"] . "!";

}

?>

38. Implement a PHP program PDF document in PHP with image

<?php

require('fpdf.php');

$pdf = new FPDF();

$pdf->AddPage();

$pdf->SetFont('Arial', 'B', 16);

$pdf->Cell(0, 10, 'Welcome to PDF with Image!', 0, 1, 'C');

$pdf->Image('image.jpg', 10, 30, 50); // (file, x, y, width)

$pdf->SetY(90);

$pdf->SetFont('Arial', '', 12);

$pdf->MultiCell(0, 10, "This is a sample PDF file that includes an image and some text using FPDF in PHP.");

$pdf->Output();

?>

39. Implement a PHP program to

(i)Enter data into database

(ii) Retrieve and present data from database

<?Php

$servername="localhost";

$username="root";

$password="";

$db="Student";

$conn=new mysqli($servername,$username,$password,$db);

if($conn->connect\_error)

{

die("Connection is not established".$conn->connect\_error);

}

// SQL query to create table if it does not exist

$sql = "CREATE TABLE IF NOT EXISTS student (

id INT AUTO\_INCREMENT PRIMARY KEY,

name1 VARCHAR(100) NOT NULL,

roll INT NOT NULL

)";

// Execute the query

if ($conn->query($sql) === TRUE) {

echo "Table 'student' is ready!";

} else {

echo "Error creating table: " . $conn->error;

}

$sql="INSERT INTO student(name1,roll)VALUES('Yash',55)";

if($conn->query($sql)==TRUE)

{

echo " data inserted";

}

else

{

echo " data not inserted";

}

$sql="SELECT \* FROM student";

$result=$conn->query($sql);

if($result->num\_rows>0)

{

while($row=$result->fetch\_assoc())

{

echo "name1".$row["name1"]." roll no".$row["roll"];

}

}

else{

echo "Connection is not established".$conn->connect\_error;

}

$conn->close();

?>

40. Implement a PHP program Employee form like employee\_name, Address, Mobile\_No.,

Date of birth, Post and Salary using different form input element and display user

inserted values in new PHP form.

<!DOCTYPE html>

<html>

<head>

<title>Employee Form</title>

</head>

<body>

<h2>Employee Information Form</h2>

<form method="post" action="display\_employee.php">

<label>Employee Name:</label>

<input type="text" name="emp\_name" required><br><br>

<label>Address:</label>

<textarea name="address" required></textarea><br><br>

<label>Mobile No:</label>

<input type="text" name="mobile" required><br><br>

<label>Date of Birth:</label>

<input type="date" name="dob" required><br><br>

<label>Post:</label>

<select name="post">

<option value="Manager">Manager</option>

<option value="Engineer">Engineer</option>

<option value="Accountant">Accountant</option>

<option value="Clerk">Clerk</option>

</select><br><br>

<label>Salary:</label>

<input type="number" name="salary" required><br><br>

<input type="submit" value="Submit">

</form>

</body>

</html>

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

echo "<h2>Employee Details</h2>";

echo "Name: " . $\_POST["emp\_name"] . "<br>";

echo "Address: " . $\_POST["address"] . "<br>";

echo "Mobile No: " . $\_POST["mobile"] . "<br>";

echo "Date of Birth: " . $\_POST["dob"] . "<br>";

echo "Post: " . $\_POST["post"] . "<br>";

echo "Salary: " . $\_POST["salary"] . "<br>";

} else {

echo "No data submitted.";

}

?>