

1. Write a PHP program to create a simple webpage.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>My Simple Webpage</title>
</head>
<body>
    <?php
        echo "<h1>Welcome to my webpage!</h1>";
    ?>

</body>
</html>
```

Output:

Welcome to my webpage!

2. Write a PHP program to display the current date and time.

Source code:

```
<?php  
  
    echo "Current date and time: " . date("Y-m-d H:i:s a");  
?>
```

Output:

Current date and time: 2023-03-18 12:01:38 am

3. Create a PHP program using if else conditional statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>If Else Example</title>
</head>
<body>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    <label for="number">Enter a number:</label>
    <input type="number" id="number" name="number">
    <input type="submit" value="Check">
</form>
<?php

    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        $number = $_POST["number"];

        if ($number > 0) {
            echo "$number is a positive number.";
        } else {
            echo "$number is a negative number.";
        }
    }

?>

</body>
</html>
```

Output:

Enter a number:

15 is a positive number.

4.Create a PHP program using nested if conditional statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Nested If Example</title>
</head>
<body>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $age = $_POST["age"];

            if ($age >= 18) {
                echo "You are an adult.";

                if ($age >= 65) {
                    echo " You are also a senior citizen.";
                }
            } else {
                echo "You are a minor.";
            }
        }

    ?>

    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="age">Enter your age:</label>
        <input type="number" id="age" name="age">
        <input type="submit" value="Check">
    </form>
</body>
</html>
```

Output:

You are a minor.

Enter your age:

Check

5. Create a PHP program using switch statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Switch Example</title>
</head>
<body>
    <?php
        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $grade = $_POST["grade"];

            switch ($grade) {
                case "A"://to add smaller case use another case "a"
                    echo "Excellent!";
                    break;
                case "B":
                    echo "Good job!";
                    break;
                case "C":
                    echo "You could do better.";
                    break;
                case "D":
                    echo "You need to work harder.";
                    break;
                case "F":
                    echo "You failed.";
                    break;
                default:
                    echo "Invalid grade entered.";
            }
        }

    ?>
    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="grade">Enter your grade:</label>
        <input type="text" id="grade" name="grade">
        <input type="submit" value="Check">
    </form>
</body>
```

</html>

Output:

You failed.

Enter your grade:

6.Create a PHP program using while loop statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>While Loop Example</title>
</head>
<body>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $countdown = $_POST["countdown"];
            while ($countdown > 0) {
                echo $countdown . "<br>";
                $countdown--;
            }
        }

    ?>

    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="countdown">Enter a number to count down from:</label>
        <input type="number" id="countdown" name="countdown">
        <input type="submit" value="Start Countdown">
    </form>
</body>
</html>
```

Output:

5
4
3
2
1

Enter a number to count down from:

Start Countdown

7. Create a PHP program using do while loop statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Do While Loop Example</title>
</head>
<body>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    <label for="number">Enter a number to calculate its factorial:</label>
    <input type="number" id="number" name="number">
    <input type="submit" value="Calculate">
</form>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $number = $_POST["number"];
            $factorial = 1;
            $i = 1;

            do {
                $factorial *= $i;
                $i++;
            } while ($i <= $number);

            echo "The factorial of " . $number . " is " . $factorial;
        }

    ?>

</body>
</html>
```

Output:

Enter a number to calculate its factorial:

The factorial of 5 is 120

8.Create a PHP program using for loop statement.

Source code :

```
<!DOCTYPE html>
<html>
<head>
    <title>For Loop Example</title>
</head>
<body>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $number = $_POST["number"];
            $sum = 0;

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

            echo "The sum of all numbers between 1 to " . $number . " is " . $sum;
        }

    ?>

    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="number">Enter a number:</label>
        <input type="number" id="number" name="number">
        <input type="submit" value="Calculate Sum">
    </form>
</body>
</html>
```

Output:

The sum of all numbers between 1 to 3 is 6

Enter a number:

Calculate Sum

9. Create a PHP program using for each statement.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Foreach Loop Example</title>
</head>
<body>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $input = $_POST["input"];
            $numbers = explode(",", $input);

            echo "<h2>Input Values:</h2>";
            foreach ($numbers as $num) {
                echo $num . "<br>";
            }
        }

    ?>

    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="input">Enter comma-separated values:</label>
        <input type="text" id="input" name="input">
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Output:

Input Values:

3
5
8
94

Enter comma-separated values:

Submit

10. Write a PHP program to display a numerical array.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Numerical Array Example</title>
</head>
<body>
    <?php

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            $input = $_POST["input"];
            $numbers = explode(",", $input);

            echo "<h2>Array Values:</h2>";
            foreach ($numbers as $number) {
                echo $number . "<br>";
            }
        }

    ?>

    <form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="input">Enter comma-separated numbers:</label>
        <input type="text" id="input" name="input">
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Output:

Array Values:

3

5

4

Enter comma-separated numbers:

Submit

11. Write a PHP program to display an associative array.

Source code:

```
<?php
// Check if form was submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Get input from form
    $keys = explode(",", $_POST["keys"]);
    $values = explode(",", $_POST["values"]);

    // Combine input into associative array
    $array = array_combine($keys, $values);

    // Display resulting array
    echo "<h2>Resulting Associative Array:</h2>";
    echo "<pre>";
    print_r($array);
    echo "</pre>";
}
?>

<!-- Display input form -->
<h2>Enter Key-Value Pairs for Associative Array:</h2>
<form method="post">
    <label for="keys">Keys (comma-separated):</label><br>
    <input type="text" id="keys" name="keys"><br>
    <label for="values">Values (comma-separated):</label><br>
    <input type="text" id="values" name="values"><br><br>
    <input type="submit" value="Submit">
</form>

</body>
</html>
```

Output:

Resulting Associative Array:

```
Array
(
    [car] => bmw
    [bike] => pulsar
    [boat] => speed
)
```

Enter Key-Value Pairs for Associative Array:

Keys (comma-separated):

Values (comma-separated):

12. Write a PHP program to display a multidimensional array.

Source code:

```
<html>
  <head>
    <title>Multi-Dimensional Array Example</title>
  </head>
  <body>
    <form method="post">
      <label for="input">Enter multi-dimensional array:(example:red,blue,green
[enter])</label><br>
      <textarea id="input" name="input" rows="10" cols="50"></textarea><br>
      <input type="submit" value="Submit">
    </form>

    <?php
      // Check if form was submitted
      if($_SERVER["REQUEST_METHOD"] == "POST") {
        // Get input from user
        $input = $_POST['input'];

        // Convert input string to array
        $arr = array();
        $lines = explode("\n", $input);
        foreach ($lines as $line) {
          $arr[] = explode(",", $line);
        }

        // Display array
        echo "<pre>";
        print_r($arr);
        echo "</pre>";
      }
    ?>

  </body>
</html>
```

Output:

Enter multi-dimensional array:(example:red,blue,green [enter])

Submit

Array

```
(  
  [0] => Array  
    (  
      [0] => bmw  
      [1] => volvo  
      [2] => audi  
    )  
  
  [1] => Array  
    (  
      [0] => TN05b8969  
      [1] => TN15a07  
      [2] => TN65n9999  
    )  
)
```


13. Write a PHP program to sort an array element.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Sort Array</title>
</head>
<body>
<h2>Sort Array</h2>
    <form method="post" action="<?php echo
    htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        <label for="numbers">Enter comma-separated numbers:</label>
        <input type="text" id="numbers" name="numbers"><br><br>
        <input type="submit" value="Sort">
    </form>

<?php
    // Check if form was submitted
    if ($_SERVER["REQUEST_METHOD"] == "POST") {
        // Retrieve form data using $_POST
        $numbers = $_POST["numbers"];

        // Convert input to array
        $numbers_array = explode(",", $numbers);

        // Trim whitespace from each number
        $numbers_array = array_map('trim', $numbers_array);

        // Convert each number to integer
        $numbers_array = array_map('intval', $numbers_array);

        // Sort the array in ascending order
        sort($numbers_array);

        // Output the sorted array
        echo "<h3>Sorted Numbers:</h3>";
        echo implode(", ", $numbers_array);
    }
?>
```

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

Output:

Sort Array

Enter comma-separated numbers:

Sorted Numbers:

1, 2, 2, 3, 4, 6, 8, 9

14. Write a PHP program to illustrate session.

Sourcecode:

```
<?php
    // Start a new session
    session_start();

    $_SESSION["username"]="deckard";
    echo $_SESSION["username"];

    if (isset($_SESSION["username"])){
        echo "<br>session created";
    }
    else{echo "<br>session not created";}

    // Destroy the session
    session_destroy();
?>
```

Output:

Deckard

Session created

15. Write a PHP program to illustrate cookies.

Source code:

```
<?php
    // Set the cookie with a name, value, expiration time, and path
    setcookie("username", "John", time() + 3600, "/");

    // Output a message if the cookie is set
    if (isset($_COOKIE["username"])) {
        echo "Welcome " . $_COOKIE["username"] . "!"<br>";
    } else {
        echo "Please set a cookie first.<br>";
    }

    // Update the cookie value
    setcookie("username", "Jane", time() + 30, "/");

    // Output the updated cookie value
    echo "Your username has been updated to " . $_COOKIE["username"] . ".";
?>
```

Output:

Welcome Jane!

Your username has been updated to Jane.

16. Write a PHP program to illustrate file manipulation.

Source code:

```
<?php
    // Create a new file and write some text to it
    $file = fopen("example.txt", "w");
    fwrite($file, "Hello, world!\n");
    fclose($file);

    // Read the contents of the file and output them to the user
    $file = fopen("example.txt", "r");
    echo fread($file, filesize("example.txt"));
    fclose($file);

?>
```

Output:

Hello world!

17. Create a simple application using the post method.

Source code:

1.postexample.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Post Example</title>
</head>
<body>
    <h1>Post Example</h1>
    <form method="post" action="post.php">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name"><br>
        <input type="submit" name="submit" value="Submit">
    </form>
</body>
</html>
```

2.post.php:

```
<?php
    if ($_SERVER['REQUEST_METHOD'] == 'POST') {
        $name = $_POST['name'];
        echo "<p>hello, $name!</p>";
    }
?>
```

Output:

Post Example

Name:

Hello, john!

18. Create a simple application using the get method.

Source code:

1.get.php

```
<?php
    if (isset($_GET['submit'])) {
        $name = $_GET['name'];
        echo "<p>Hello, $name!</p>";
    }
?>
```

2.getexample.html:

```
<!DOCTYPE html>
<html>
<head>
    <title>Get Example</title>
</head>
<body>
    <h1>Get Example</h1>
    <form method="get" action="get.php">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name"><br>
        <input type="submit" name="submit" value="Submit">
    </form>
</body>
</html>
```

Output:

Get Example

Name:

Hello , Eren!

19. Create an application with input validations.

Source code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Login Form</title>
</style>
.error{color:red}
</style>
</head>

<body>
    <?php
        $emailErr = $passwordErr = "";
        $email = $password = "";

        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            if (empty($_POST["email"])) {
                $emailErr = "Email is required";
            } else {
                $email = test_input($_POST["email"]);
                if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
                    $emailErr = "Invalid email format";
                }
            }

            if (empty($_POST["password"])) {
                $passwordErr = "Password is required";
            } else {
                $password = test_input($_POST["password"]);
                if (strlen($password) < 8) {
                    $passwordErr = "Password must be at least 8 characters
long";
                }
            }
        }

        function test_input($data) {
            $data = trim($data);
```

```

        $data = stripslashes($data);
        $data = htmlspecialchars($data);
        return $data;
    }
?>

<h2>Login Form</h2>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    <label>Email:</label>
    <input type="text" name="email" value="<?php echo $email;?>">
    <span class="error">* <?php echo $emailErr;?></span>
    <br><br>
    <label>Password:</label>
    <input type="password" name="password">
    <span class="error">* <?php echo $passwordErr;?></span>
    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

<?php
    if ($emailErr == "" && $passwordErr == "" && isset($_POST["submit"])) {
        echo "<h2>Login Details:</h2>";
        echo "<p>Email: $email</p>";
        echo "<p>Password: $password</p>";
    }
?>
</body>
</html>

```

Output:

Login Form

Email: *

Password: *

Login Details:

Email: rick@gmail.com

Password: 123321456

20.Create a table in PHP using MYSQL connectivity.

Source code:

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

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

// sql to create table
$sql = "CREATE TABLE MyGuests (
id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
firstname VARCHAR(30) NOT NULL,
lastname VARCHAR(30) NOT NULL,
email VARCHAR(50),
salary INT(25),
reg_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP
)";

if (mysqli_query($conn, $sql)) {
    echo "Table MyGuests created successfully";
} else {
    echo "Error creating table: " . mysqli_error($conn);
}

mysqli_close($conn);
?>
```


Output:

Table MyGuests created successfully

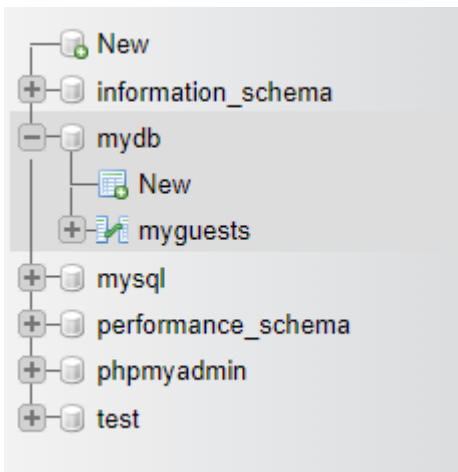


Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> myguests	Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_general_ci	16.0 KiB	-
1 table	Sum	0	InnoDB	utf8mb4_general_ci	16.0 KiB	0 B

☐ Check all With selected: ▼

21. Write a PHP program to insert row in table using MySQL connectivity.

Source code:

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

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




$sql = "INSERT INTO MyGuests (firstname, lastname, email, salary)
VALUES ('John', 'Doe', 'john@example.com', 25000)";

if (mysqli_query($conn, $sql)) {
    $last_id = mysqli_insert_id($conn);
    echo "New record created successfully. Last inserted ID is: " . $last_id;
} else {
    echo "Error: " . $sql . "<br>" . mysqli_error($conn);
}

mysqli_close($conn);
?>
```

Output:

New records created successfully

				id	firstname	lastname	email	reg_date
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	Jack	Hills	jack@example.com	2023-03-17 20:16:12

22. Write a PHP program to delete row in a table using MySQL connectivity.

Source code:

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

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

// sql to delete a record
$sql = "DELETE FROM MyGuests WHERE id=2";

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

mysqli_close($conn);
?>
```

Output:

Record deleted successfully

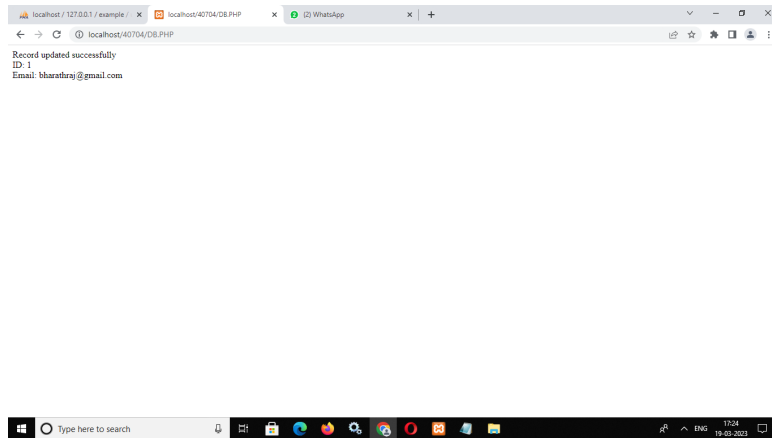
23. Write a PHP program to update and display the table using MySQL connectivity.

Source code:

```
<?php
// Connect to database
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "example";
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
// Update an existing row
$sql = "UPDATE users SET email='bharathraj@gmail.com' WHERE id=1";
if (mysqli_query($conn, $sql)) {
    echo "Record updated successfully". "<br>";
} else {
    echo "Error updating record: " . mysqli_error($conn);
}
// Display the updated row
$sql = "SELECT * FROM users WHERE id=1";
$result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
    $row = mysqli_fetch_assoc($result);
    echo "ID: " . $row["id"] . "<br>";
    echo "Name: " . $row["firstname"] . "<br>";
    echo "Email: " . $row["email"] . "<br>";
} else {
    echo "0 results";
}
// Close connection
mysqli_close($conn);
?>
```

Output:

Record updated successfully



24. Create a PHP program to apply any two aggregate functions using MySQL connectivity.

Source code:

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

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);

// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Example aggregate functions: COUNT() and MAX()
$sql = "SELECT COUNT(*) AS total_rows, MAX(salary) AS max_salary FROM
myguests";

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

if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
        echo "Total number of rows: " . $row["total_rows"]. "<br>";
        echo "Maximum salary: " . $row["max_salary"]. "<br>";
    }
} else {
    echo "0 results";
}

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


Output:

Total number of rows: 3

Maximum salary: 50000

25..Create a PHP program to search a record using MYSQL connectivity.

Source code:

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Get search keyword from user input
    if (isset($_POST['searchKeyword'])) {
        $searchKeyword = $_POST['searchKeyword'];
    } else {
        // Show a message to the user if the searchKeyword parameter is not set
        echo "Please enter a search keyword";
        exit();
    }

    $servername = "localhost";
    $username = "root";
    $password = "";
    $dbname = "myDB";

    // Create connection
    $conn = new mysqli($servername, $username, $password, $dbname);

    // Check connection
    if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
    }

    // Prepare SQL query
    $sql = "SELECT * FROM myguests WHERE firstname LIKE '%$searchKeyword%' OR
    email LIKE '%$searchKeyword%' OR id LIKE '%$searchKeyword%'";

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

    if ($result->num_rows > 0) {
        // Output data of each row
        while($row = $result->fetch_assoc()) {
            echo "id: " . $row["id"]. "<br>";
        }
    }
}
```

```
        echo "Name: " . $row["firstname"]. "<br>";
        echo "Email: " . $row["email"]. "<br>";
        echo "Salary: " . $row["salary"]. "<br>";
        echo "Registration Date: " . $row["reg_date"]. "<br>";
        echo "<br>";
    }
} else {
    echo "0 results";
}

$conn->close();
}
?>

<form method="post">
    <label for="searchKeyword">Search keyword:</label>
    <input type="text" name="searchKeyword" id="searchKeyword">
    <button type="submit">Search</button>
</form>
```

Output:

id: 3

Name: Julie

Email: julie@example.com

Salary: 50000

Registration Date: 2023-03-17 20:32:18

Search keyword: