Lab 5

Objectives:

This lab focuses on demonstrating core PHP functionalities through a series of practical tasks. It includes generating a multiplication table using loops and calculating the square of numbers using the foreach loop. It explores function overloading in PHP using func\_get\_args() to handle variable arguments for summing two or more numbers. File handling operations are performed such as creating a file, writing data, reading contents, appending the current date, and renaming the file. The lab also covers image file upload to a server directory using PHP, highlighting basic form handling and server-side validation. Lastly, session management is implemented by creating a session, maintaining its persistence across pages, and destroying it to ensure secure handling of user data. These exercises collectively enhance understanding of PHP syntax, data handling, and web programming concepts.

Q.1. Write a PHP script that generates a multiplication table.

Source Code:

multiplication.php

<!DOCTYPE *html*>

<html>

<head>

    <title>Multiplication Table</title>

    <style>

        table {

            border-collapse: collapse;

            width: 50%;

            text-align: center;

        }

        th,

td {

            border: 1px solid black;

            padding: 8px;

        }

        th {

            background-color: #f2f2f2;

        }

    </style>

</head>

<body>

    <h2>Multiplication Table (10 x 10)</h2>

    <table>

        <tr>

            <th>\*</th>

            <?php

            for ($i = 1; $i <= 10; $i++) {

                echo "<th>$i</th>";

            }

            ?>

        </tr>

        <?php

        for ($i = 1; $i <= 10; $i++) {

            echo "<tr>";

            echo "<th>$i</th>";

            for ($j = 1; $j <= 10; $j++) {

                echo "<td>" . ($i \* $j) . "</td>";

            }

            echo "</tr>";

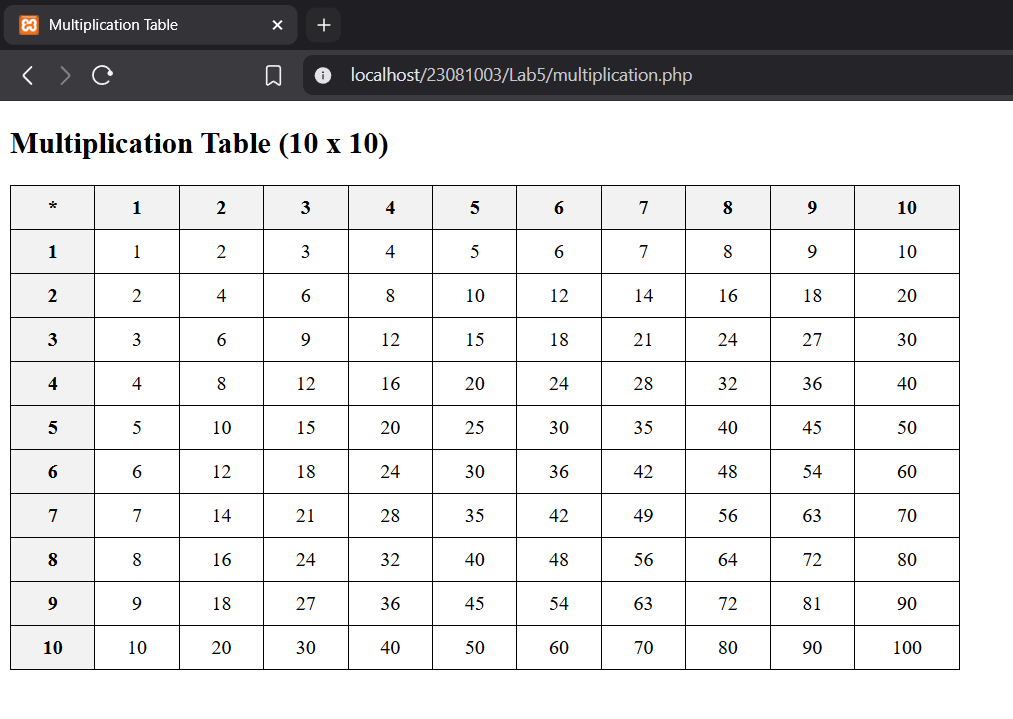
        }

        ?>

    </table>

</body>

</html>

Output:

Q.2. Assign an array of numbers to a variable. Find the square of the numbers by using foreach loop.

Source Code:

Square.php

<?php

// *Assigning array of numbers*

$numbers = array(2, 4, 6, 8, 10);

// *Finding squares using foreach*

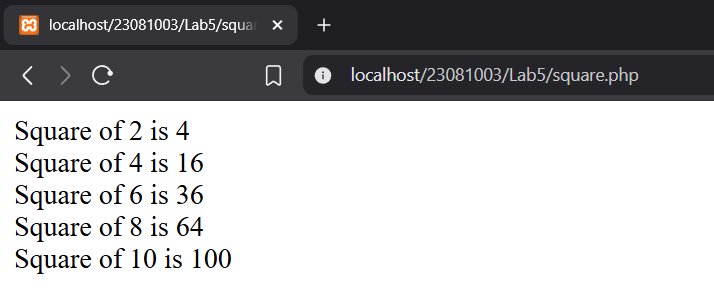
foreach ($numbers as $num) {

    $square = $num \* $num;

    echo "Square of $num is $square<br>";

}

Output:



Q.3. Verify if PHP supports function overloading. Create a function that generates the sum of two numbers. Use the same function to generate the sum of 3 numbers. Illustrate the use of func\_get\_args for the given scenario.

Source Code:

sum.php

<?php

function sum() {

    $args = func\_get\_args(); // *Get all passed arguments*

    $total = 0;

    foreach ($args as $num) {

        $total += $num;

    }

    return $total;

}

// *Sum of 2 numbers*

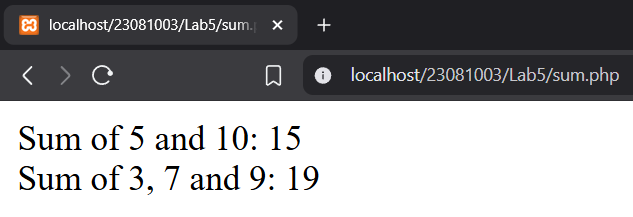
echo "Sum of 5 and 10: " . sum(5, 10) . "<br>";

// *Sum of 3 numbers*

echo "Sum of 3, 7 and 9: " . sum(3, 7, 9);

?>

Output:



Q.4. Perform the following file operations:

a. Create a file named file.txt.

b. Write your name and roll number in the file

c. Open the file to read its contents.

d. Add current date to its contents

e. Close the file

f. Rename the file as test.txt

Source Code:

<?php

// *a. Create a file named file.txt and open in write mode*

$file = fopen("file.txt", "w");

echo nl2br(file\_get\_contents("test.txt"));

// *b. Write name and roll number*

fwrite($file, "Name: Sulav Adhikari\n");

fwrite($file, "Roll Number: 23081003\n");

// *e. Close the file*

fclose($file);

// *c. Open the file to read its contents*

$file = fopen("file.txt", "r");

$content = fread($file, filesize("file.txt"));

fclose($file);

// *d. Add current date to its contents*

$currentDate = "Date: " . date("Y-m-d") . "\n";

$content .= $currentDate;

// *Write updated content back to the file*

$file = fopen("file.txt", "w");

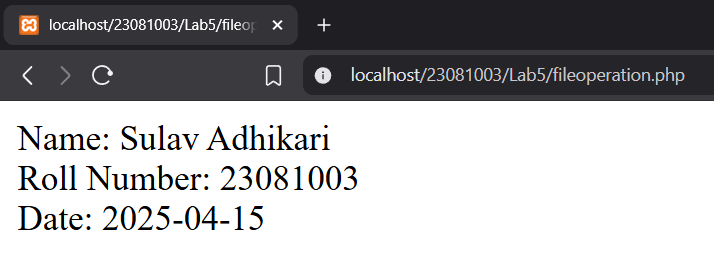
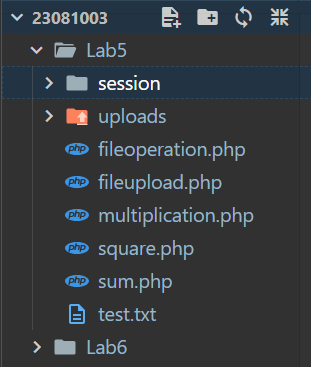
fwrite($file, $content);

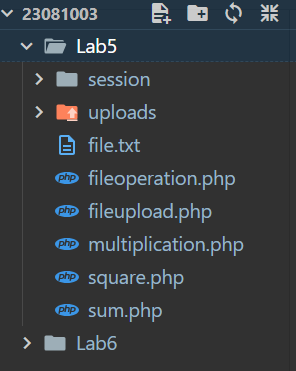
fclose($file);

// *f. Rename the file to test.txt*

rename("file.txt", "test.txt");

Output:





Q.5. Create a PHP script to upload an image to a directory on your server.

Source Code:

fileuplaod.php

<!DOCTYPE *html*>

<html>

<body>

<form *action*="" *method*="post" *enctype*="multipart/form-data">

    <h1>Select file to upload:</h1><br>

    <input *type*="file" *name*="fileToUpload" *id*="fileToUpload">

    <input *type*="submit" *value*="Upload File" *name*="submit">

</form>

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_FILES["fileToUpload"])) {

    $target\_dir = "uploads/";

    $target\_file = $target\_dir . basename($\_FILES["fileToUpload"]["name"]);

    $imageFileType = strtolower(pathinfo($target\_file, PATHINFO\_EXTENSION));

    // *Allow only PNG and JPG files*

    if (in\_array($imageFileType, ["jpg", "jpeg", "png"])) {

        if (move\_uploaded\_file($\_FILES["fileToUpload"]["tmp\_name"], $target\_file)) {

            echo "File uploaded successfully.";

        } else {

            echo "Error uploading file.";

        }

    } else {

        echo "Only JPG and PNG files are allowed.";

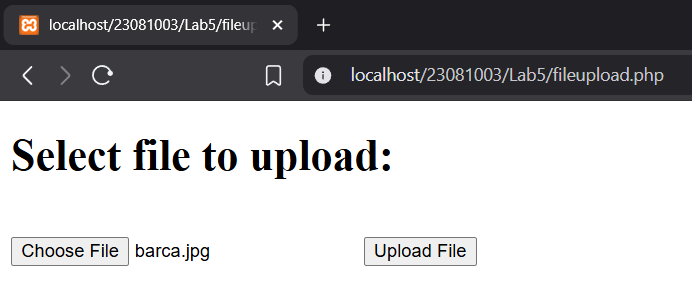
    }

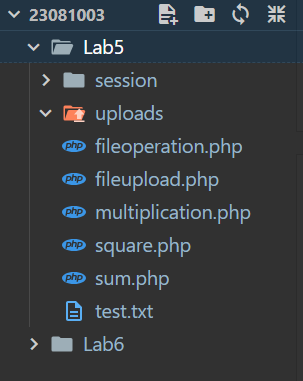
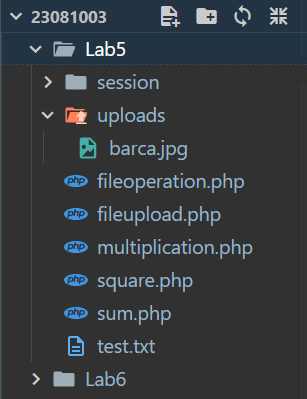
}

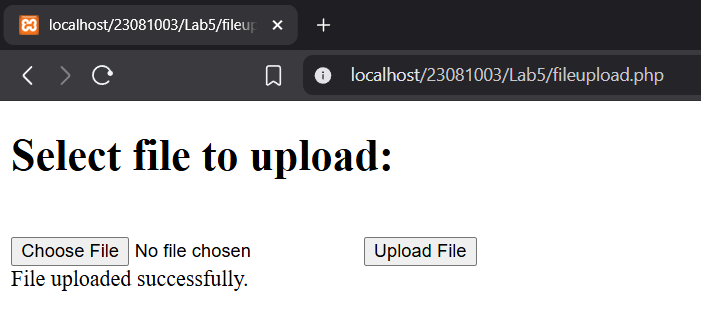
?>

</body>

</html>

Output:





Q.6. Write a PHP script to create a session. Verify the persistence of the session throughout various pages. Finally destroy the session.

Source Code:

check\_session.php

<?php

session\_start(); // *Resume the session*

if (isset($\_SESSION["username"])) {

    echo "Welcome, " . $\_SESSION["username"] . "<br>";

    echo "Roll Number: " . $\_SESSION["rollno"] . "<br>";

} else {

    echo "No session found.<br>";

}

echo "<a href='destroy\_session.php'>Destroy Session</a>";

create\_session.php

<?php

session\_start(); // *Create/start the session*

$\_SESSION["username"] = "Sulav Adhikari";

$\_SESSION["rollno"] = "23081003";

echo "Session started and values are set.<br>";

echo "<a href='check\_session.php'>Go to Check Session Page</a>";

destroy\_session.php

<?php

session\_start();

session\_unset(); // *Remove all session variables*

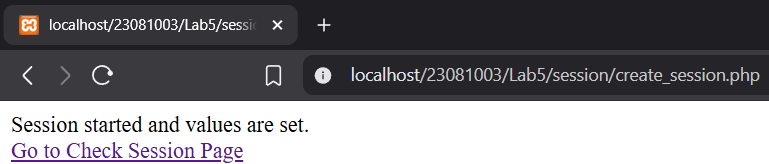
session\_destroy(); // *Destroy the session*

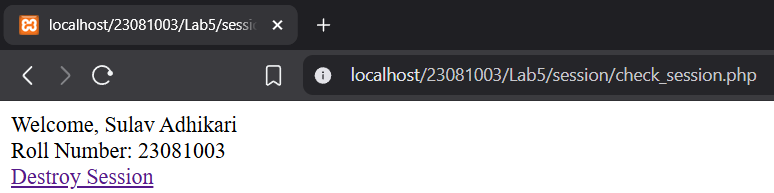
echo "Session destroyed.<br>";

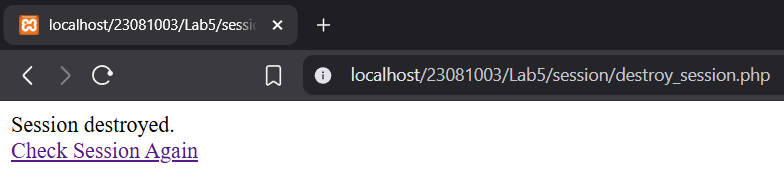
echo "<a href='check\_session.php'>Check Session Again</a>";

?>

Output:







Conclusion:

Through this lab, we successfully explored and implemented various fundamental concepts of PHP programming. We practiced working with loops, arrays, functions with variable arguments, file handling, image uploading, and session management. These tasks helped us understand how PHP interacts with the server, manages user data, and handles different types of operations efficiently. Overall, this lab strengthened our practical knowledge of PHP and its application in real-world web development scenarios.