PHP LAB

LABORATORY MANUAL

BCA

(III YEAR – VI SEM)

(2024 – 25)

DEPARTMENT OF COMPUTER SCIENCE

SWAMI DAYANANDA COLLEGE OF ARTS & SCIENCE

(Affiliated to Bharathidasan University , Tiruchirappalli – 24)
Accredited by NAAC 'B++' Grade(With CGPA 2.99 – I Cycle)

DEPARTMENT OF COMPUTER SCIENCE

Vision

- To empower student with the contemporary knowledge in computer science for the success in life.
- To create the most conductive environment for quality education in computer science.

Mission

- To provide learning ambience to enhance innovations, problem solving skills and leadership.
- To groom students with good attitude, team work and personality skills.
- To promote and research-based projects/activities in the emerging areas of technology convergence.

- To foster partnership with industry leaders and startups , ensuring that our programs remain aligned with industry trend and demands.

PROGRAM SPECIFIC OUTCOMES(PSOs)

After the completion of the course, BCA computer science , the graduates will have the following program specific outcomes:

1. **Fundamentals and critical knowledge of the computer system:-** Able to Understand the working principles of the computer system and its components, Apply the knowledge to build, access, and analyze the software and hardware aspects of it.

2. **The comprehensive and Applicative knowledge of Software Development:** Comprehensive skills of Programming Languages, Software process models, methodologies, and able to plan, develop, test, analyze, and manage the software and hardware intensive systems in heterogeneous platforms individually or working in teams.

3. **Applications of Computing Domain & Research:** Able to use the professional, managerial, interdisciplinary skill set, and domain specific tools in development processes, identify the research gaps, and provide innovative solutions to them.

OBJECTIVES:

· To develop an ability to design and implement static and dynamic website
· Choose best technologies for solving web client/server problems
· Create conforming web pages
· Use JavaScript for dynamic effects
· To prepare PHP scripts
· Use JavaScript & PHP to validate form input entry
· Understand, analyze and create XML documents and XML Schema
· Understand, analyze and build web applications using PHP
· Use appropriate client-side or Server-side applications
· Handling Cookies and Sessions using PHP, SERVLETS and JSP
· Manage normal and abnormal interactions with databases using JDBC.

OUTCOMES:

Upon successful completion of this course, the students will be able to:

· Design and implement dynamic websites with good aesthetic sense of designing and latest technical know-how's
· Create web pages using HTML and Cascading Styles sheets
· Analyze a web page and identify its elements and attributes

· Create dynamic web pages using JavaScript
· Develop interactive web applications using HTML forms and servlets.
· Have a Good grounding of Web Application Terminologies, Internet Tools, E – Commerce and other web services
· Use request and response objects provided to a servlet to read parameters and to produce an HTML response.

SOFTWARE REQUIREMENTS:

Editor	**:**	**Notepad, Visual Studio code**
Web browser	**:**	**IE, Mozilla**
Operating System	**:**	**Windows, Linux**

INDEX

7	**Write a program to download a file from server.**	
8	**Write a program to store the current date and time in a COOKIE and display the 'last visited' date and time on the web page.**	
9	**Write a program to store page views count in SESSION,to increment the count on each refresh and to show the count on the web page.**	
10	**Write a program to design a simple calculator.**	
11	**Design an authentication web page in PHP with MYSQL to check username and password.**	

EXERCISE 1:

Write a program to find the factorial of a number.

AIM : To write a program to find the factorial of a number.

PROCEDURE :

STEP 1 : Create an HTML form to take user input (a number).

STEP 2: On form submission, use PHP to retrieve and validate the number.

STEP 3: Define a recursive function factorial() to calculate the factorial.

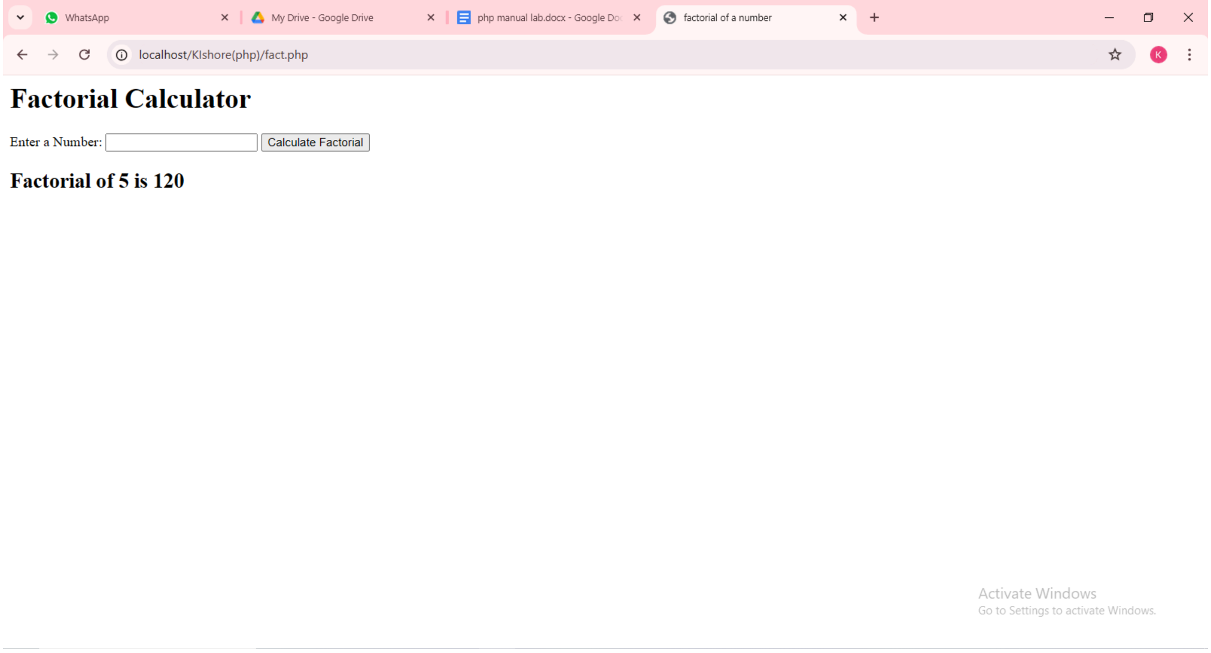
STEP 4: Check if the input number is non-negative and calculate the factorial.

STEP 5: Display the result or an error message based on the input value.

PROGRAM :

```
<html>
<head>
   <title>factorial of a number</title>
   </head>
<body>
   <h1>Factorial Calculator</h1>
<form method="POST" action="">
   <label for="number">Enter a Number:</label>
   <input type="number" id="number" name="number" required>
   <button type="submit">Calculate Factorial</button>
</form>
<?php
if($_SERVER["REQUEST_METHOD"]=="POST"){
   $number=intval($_POST['number']);
   function calculateFactorial($number){
      if($number==0 || $number==1){
         return 1;

      }
      return $number*calculateFactorial($number-1);
   }
   if($number>=0){
      $factorial=calculateFactorial($number);
      echo "<h2>Factorial of $number is $factorial</h2>";
   }

   else{
        echo"<h2>please enter valid number</h2>";
   }
   }

?>
   </html>
```

OUTPUT :



WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
factorial of a number
localhost/KIshore(php)/fact.php
Calculate Factorial
Activate Windows
Go to Settings to activate Windows.

Factorial Calculator

Enter a Number:

Factorial of 5 is 120

RESULT :

The above program entered and executed successfully.

EXERCISE 2 :

Write a program using conditional statement need a number N and check whether it is divisible by M..

AIM : To write a program using conditional statements.

PROCEDURE :

STEP 1 : Create an HTML form to input two numbers (the number and the divisor).

STEP 2 : On form submission, use PHP to retrieve and validate the numbers.

STEP 3 : Check if the divisor is zero and display an error message if true.

STEP 4 : Use the modulus operator to check if the first number is divisible by the second.

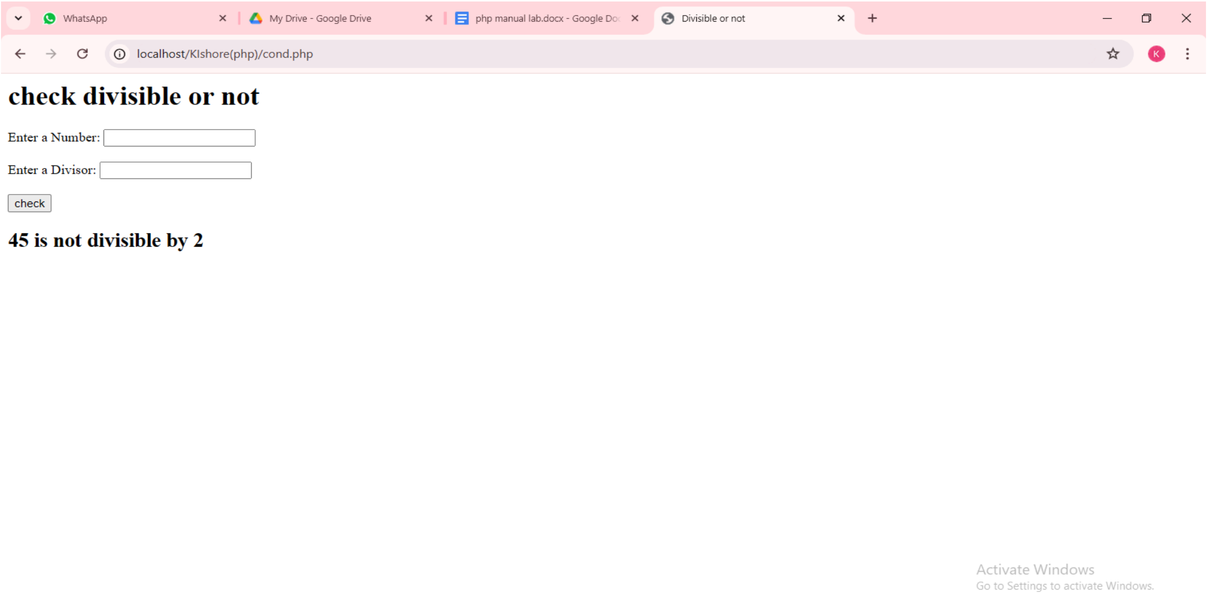
STEP 5 : Display the appropriate message based on the divisibility result.

PROGRAM :

```
<html>
   <head>
      <title>Divisible or not</title>
</head>
<body>
<h1>check divisible or not</h1>
   <form method="POST" action="">
```

```
        <label for="number1">Enter a Number:</label>
        <input type="number" name="number1" required><br> <br>
        <label for="number2">Enter a Divisor:</label>
        <input type="number" name="number2" required><br><br>
        <button type="submit">check</button>
</form>
<?php
if($_SERVER["REQUEST_METHOD"]=="POST"){
    $n1=intval($_POST['number1']);
    $n2=intval($_POST['number2']);
     if($n2==0){
        echo "<h2>Divisor will be greater than zero</h2>";
    }
        else if($n1%$n2==0){
            echo "<h2>$n1 is divisible by $n2</h2>";
        }
        else{
            echo "<h2>$n1 is not divisible by $n2</h2>";
        }
    }
?>
</body>
</html>
```

OUTPUT :



check divisible or not

Enter a Number:

Enter a Divisor:

check

45 is not divisible by 2

WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
Divisible or not
localhost/KIshore(php)/cond.php
Activate Windows
Go to Settings to activate Windows.

RESULT :

The above program entered and executed successfully.

EXERCISE 3 :

Write a program to find the maximum value in a given multi-Dimensional array.

AIM : To write a php program to find the maximum value in a multiDimensional array.

PROCEDURE :

STEP 1 : Create HTML Form: Provide a form with an input field for users to enter a comma-separated list of numbers.

STEP 2 : Capture User Input: Use PHP to capture the input when the form is submitted via the POST method.

STEP 3 : Convert Input to Array: Split the input string by commas and convert it into an array of integers using explode() and array_map().

STEP 4 : Find Maximum Value: Use a custom getmax() function to loop through the array and find the maximum value.

STEP 5 : Display Result: Output the maximum value on the web page using an echo statement.

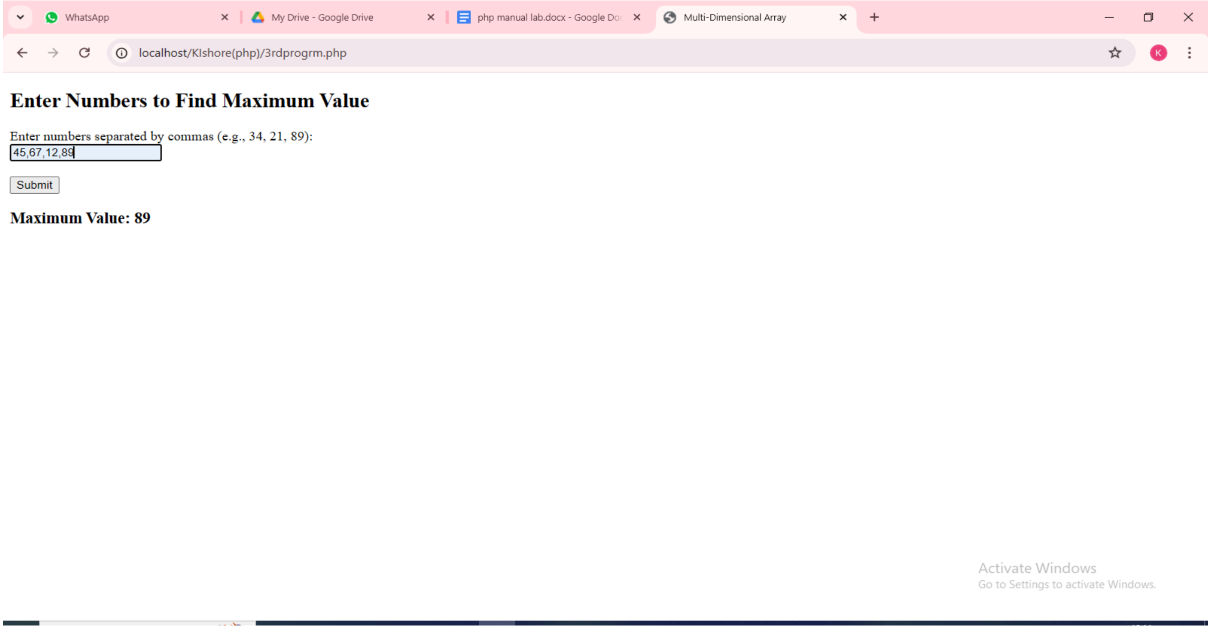
PROGRAM :

```
<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width,
initial-scale=1.0">

   <title>Multi-Dimensional Array</title>

</head>

<body>

   <h2>Enter Numbers to Find Maximum Value</h2>
```

```
<form method="post">
    <label for="numbers">Enter numbers separated by commas
(e.g., 34, 21, 89):</label><br>
    <input type="text" id="numbers" name="numbers"
required><br><br>
    <input type="submit" value="Submit">
</form>
<?php
function getmax($array) {
    $n = count($array);
    $max = $array[0];
    for ($i = 1; $i < $n; $i++) {
        if ($max < $array[$i]) {
            $max = $array[$i];
        }
    }
    return $max;
}
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $input = $_POST['numbers'];
    $array = array_map('intval', explode(',', $input));
```

```
        $max_value = getmax($array);

        echo "<h3>Maximum Value: $max_value</h3>";

       }

    ?>

</body>

</html>
```

OUTPUT :


WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
Multi-Dimensional Array
localhost/KIshore(php)/3rdprogrm.php
Activate Windows
Go to Settings to activate Windows.

Enter Numbers to Find Maximum Value

Enter numbers separated by commas (e.g., 34, 21, 89):
45,67,12,89

Submit

Maximum Value: 89

RESULT :

The above program entered and executed successfully.

EXERCISE 4 :

Write a program to find GCD of two numbers using a user-defined function.

AIM : To write a program to find GCD of two numbers using user-defined functions.

PROCEDURE :

STEP 1 : Create an HTML form to input two numbers.

STEP 2 : Implement a PHP function to compute the GCD using the Euclidean algorithm.

STEP 3 : Submit the form with the numbers to the server using POST.

STEP 4 : Retrieve the input values in PHP and call the GCD function.

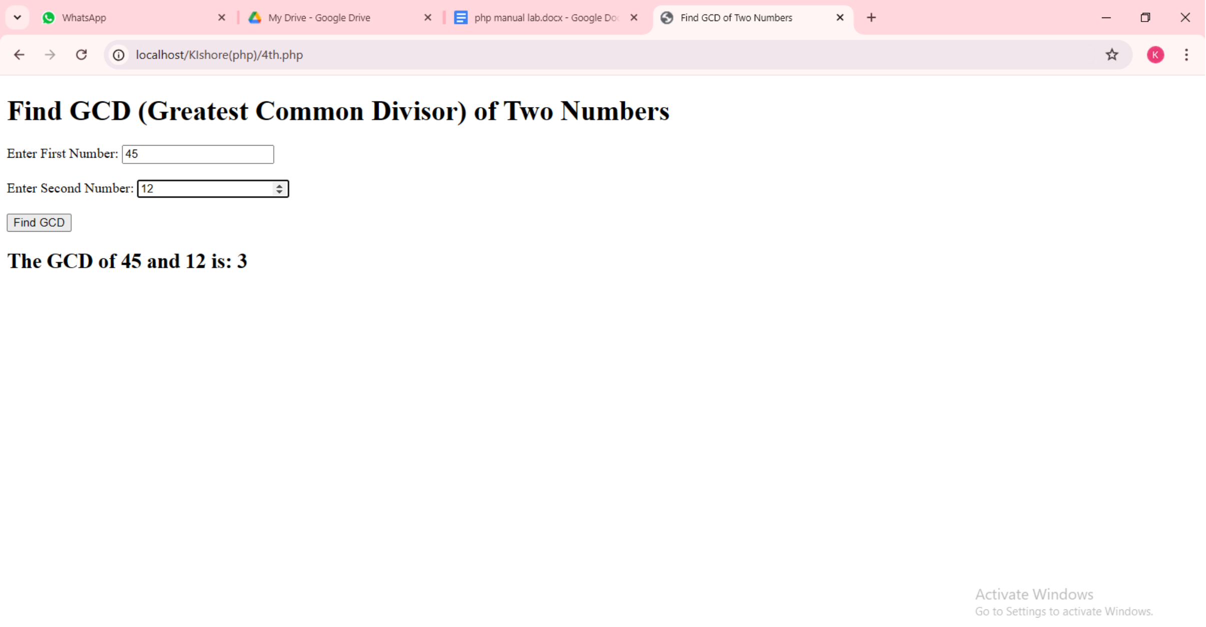
STEP 5 : Display the computed GCD result back on the webpage.

PROGRAM :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width,
initial-scale=1.0">
   <title>Find GCD of Two Numbers</title>
</head>
<body>
```

```
<h1>Find GCD (Greatest Common Divisor) of Two Numbers</h1>

<!-- Form to input the numbers -->
<form method="post" action="">
   <label for="num1">Enter First Number: </label>
   <input type="number" name="num1" id="num1"
required><br><br>

   <label for="num2">Enter Second Number: </label>
   <input type="number" name="num2" id="num2"
required><br><br>

   <input type="submit" value="Find GCD">
</form>

<?php
// Function to find GCD using Euclidean algorithm
function findGCD($a, $b) {
   // Loop until b becomes zero
   while ($b != 0) {
      $temp = $b;
      $b = $a % $b; // Update b to the remainder of a divided by b
      $a = $temp;    // Update a to the previous value of b
   }
   return $a;  // GCD is the last non-zero remainder
}

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
   // Get the numbers from the form input
   $num1 = $_POST['num1'];
   $num2 = $_POST['num2'];

   // Call the function to find the GCD
   $gcd = findGCD($num1, $num2);

   // Display the result
   echo "<h2>The GCD of $num1 and $num2 is: $gcd</h2>";
}
?>

</body>
</html>
```

OUTPUT :


WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
Find GCD of Two Numbers
localhost/KIshore(php)/4th.php
Activate Windows
Go to Settings to activate Windows.

Find GCD (Greatest Common Divisor) of Two Numbers

Enter First Number: 45

Enter Second Number: 12

Find GCD

The GCD of 45 and 12 is: 3

RESULT :

The above program entered and executed successfully.

EXERCISE 5 :

Design a simple web page to generate a multiplication table for a given number.

AIM : To write a program to generate a multiplication table.

PROCEDURE :

STEP 1 : Create an HTML form to input a number for generating the multiplication table.

STEP 2 : On form submission, use PHP to retrieve the entered number.

STEP 3 : Display the heading indicating the multiplication table for the entered number.

STEP 4 : Use a for loop to multiply the number by integers from 1 to 16.

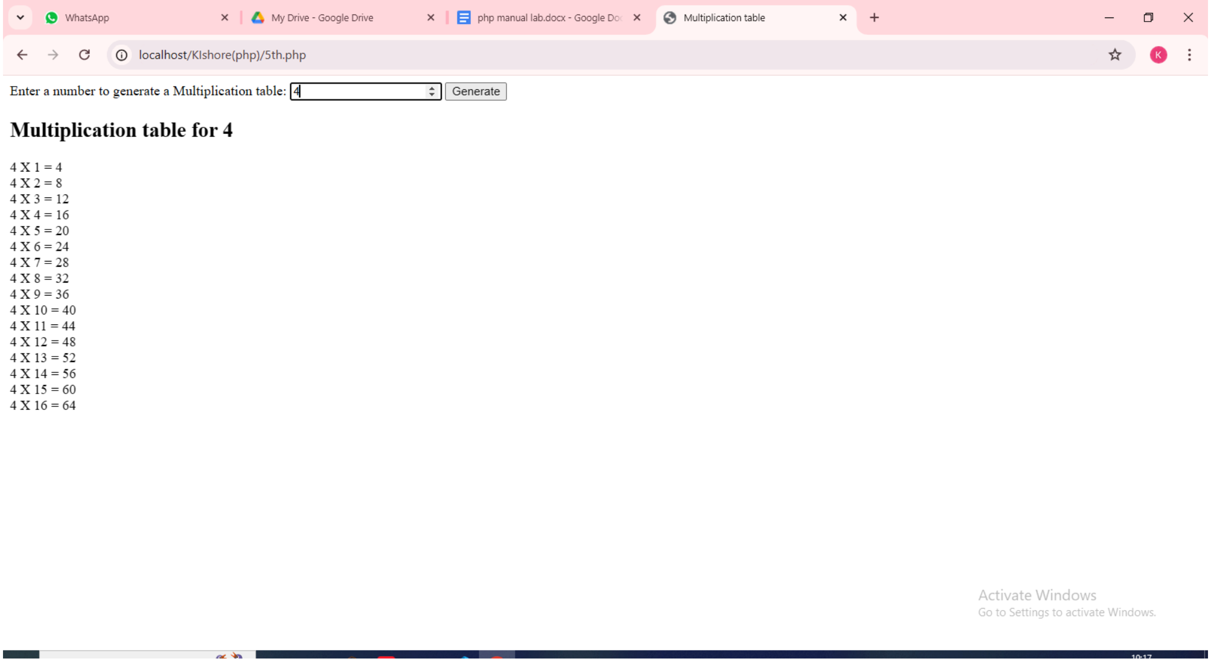
STEP 5 : Display each multiplication result in the format "number X i = result".

PROGRAM :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```

```
   <meta name="viewport" content="width=device-width,
initial-scale=1.0">
   <title>Multiplication table</title>
</head>
<body>
   <form method="post" action=" ">
      <label for="number">Enter a number to generate a
Multiplication table:</label>
      <input type="number" name="number">
      <button type ="submit">Generate</button>
</form>
<?php
 if($_SERVER['REQUEST_METHOD']=='POST'){
   $num = intval($_POST['number']);
   echo"<h2>Multiplication table for $num</h2>";
   for($i = 1; $i<=16; $i++){
      $res = $num * $i;
      echo"$num X $i = $res <br>";
   }
 }
?>

</body>
</html>
```

OUTPUT :


WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
Multiplication table
localhost/KIshore(php)/5th.php
Generate
Activate Windows
Go to Settings to activate Windows.

```
Enter a number to generate a Multiplication table: 4

Multiplication table for 4

4 X 1 = 4
4 X 2 = 8
4 X 3 = 12
4 X 4 = 16
4 X 5 = 20
4 X 6 = 24
4 X 7 = 28
4 X 8 = 32
4 X 9 = 36
4 X 10 = 40
4 X 11 = 44
4 X 12 = 48
4 X 13 = 52
4 X 14 = 56
4 X 15 = 60
4 X 16 = 64
```

RESULT :

The above program entered and executed successfully.

EXERCISE 6 :

Design a web page that should compute one's age on a given date .

AIM :

To write a program to calculate one's age on a given date.

PROCEDURE:

STEP 1 : Create an HTML form with a date input field to enter the date of birth.

STEP 2 : Use the POST method to submit the form data to the server.

STEP 3 : In PHP, retrieve the entered date of birth and convert it to a DateTime object.

STEP 4 : Calculate the age by finding the difference between the current date and the entered date of birth.

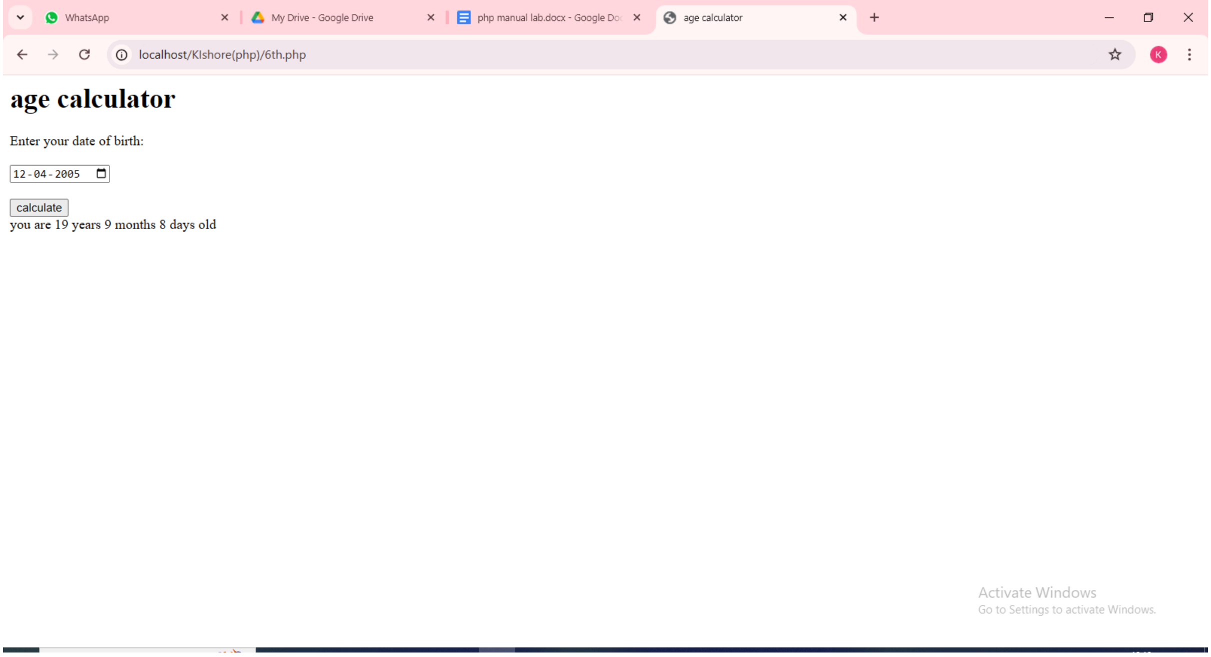
STEP 5 : Display the calculated age (in years, months, and days) or an error message if the date is invalid.

PROGRAM:

```
<html>
   <head><title>age calculator</title>
</head>
<body>
```

```
        <h1>age calculator</h1>
        <form method="POST">
            <label for="dob">Enter your date of birth:</label><br><br>
            <input type="date" name="dob"><br><br>
            <button type="submit">calculate</button>
            <?php
            if($_SERVER["REQUEST_METHOD"]=="POST"){
                $dob=$_POST['dob'];
                if($dob){
                $dobdate=new DateTime($dob);
                $currentdate=new DateTime();
                $age=$currentdate->diff($dobdate);
                if($dobdate>$currentdate){
                    echo "enter valid dob";
                }
                else{
                    echo "<br>you are ".$age->y." years ".$age->m." months
".$age->d." days old";

                }
                }


            }
?>
</body>
</html>
```

OUTPUT :



WhatsApp
My Drive - Google Drive
php manual lab.docx - Google Do
age calculator
localhost/KIshore(php)/6th.php
Activate Windows
Go to Settings to activate Windows.

age calculator

Enter your date of birth:

12-04-2005

calculate

you are 19 years 9 months 8 days old

RESULT :

The above program entered and executed successfully.

EXERCISE 7 :

Write a program to download a file from server.

AIM:

To write a program to download a file from a server.

PROCEDURE:

STEP 1 : Create an HTML link to trigger the file download via a URL with a file parameter.

STEP 2 : In PHP, check if the 'file' parameter is passed through the URL.

STEP 3 : Define the file path based on the directory and the file name passed in the URL.

STEP 4 : Verify if the file exists, then set headers to force the file to download.

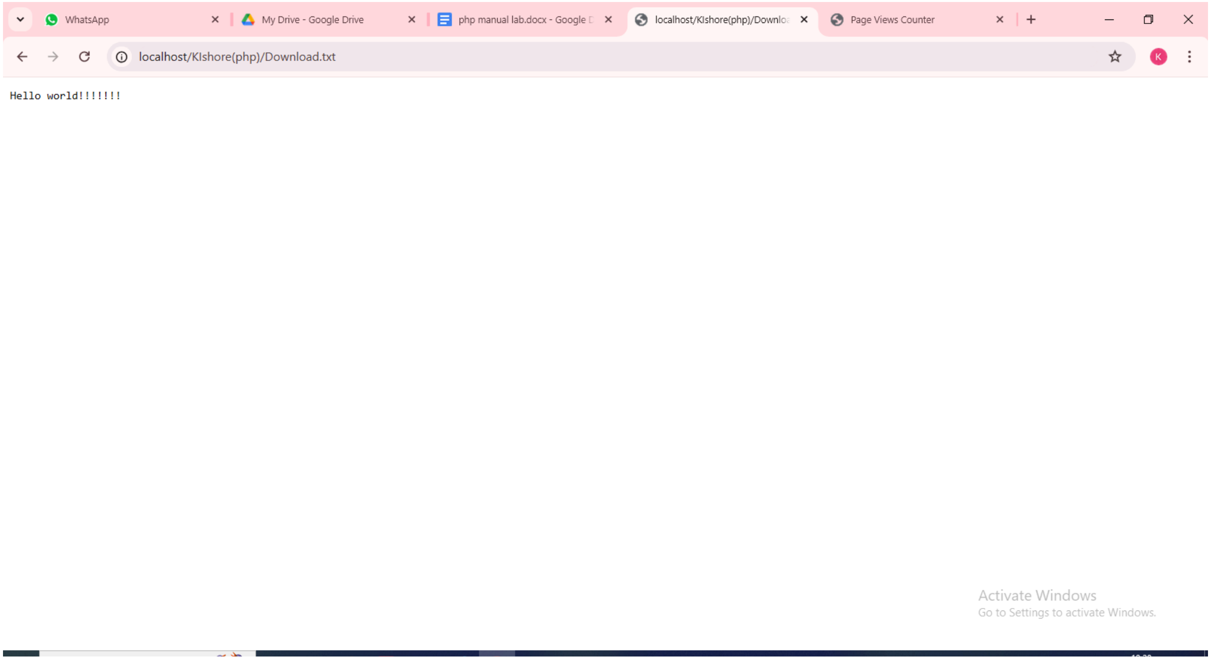
STEP 5 : Read and output the file to the browser, or show an error message if the file is not found.

PROGRAM :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width,
initial-scale=1.0">
   <title>Download File Example</title>
</head>
<body>

<h1>Download File</h1>

<!-- Link to download the file -->
<a href="download.txt">Download example.txt</a>

</body>
</html>
<?php
// Check if a 'file' parameter is passed in the URL
if (isset($_GET['file'])) {
   // Get the file name from the URL parameter
   $file = $_GET['file'];

   // Define the directory where the file is located (relative to the
current PHP script)
   $filePath = 'files/' . $file;  // assuming 'files' is the directory where
the file is stored

   // Check if the file exists
   if (file_exists($filePath)) {
      // Set the appropriate headers to force the file to be
downloaded
      header('Content-Description: File Transfer');
```

```
        header('Content-Type: application/octet-stream');
        header('Content-Disposition: attachment; filename="' .
basename($filePath) . '"');
        header('Content-Length: ' . filesize($filePath));

        // Clear the output buffer to avoid any issues with the file
download
        ob_clean();
        flush();

        // Read the file and output it to the browser
        readfile($filePath);
        exit; // End the script after the file is served
    } else {
        // If the file does not exist, show an error message
        echo 'File not found!';
    }
} else {
    // If no file is specified, show an error message
    echo 'No file specified!';
}
?>
</body>
<html>
```

OUTPUT :


WhatsApp
My Drive - Google Drive
php manual lab.docx - Google D
localhost/KIshore(php)/Downlo
Page Views Counter
localhost/KIshore(php)/Download.txt
Activate Windows
Go to Settings to activate Windows.

```
Hello world!!!!!!!
```

RESULT :

The above program entered and executed successfully.

EXERCISE 8 :

Write a program to store the current date and time in a COOKIE and display the ‘last visited date and time on the web page

AIM : To write a program to store the current date and time in a cookie and display the last visited date and time.

PROCEDURE :

STEP 1 : Check for Cookie: Verify if the last_visit cookie exists to determine if the user has visited before.

STEP 2 : Display Last Visit: If the cookie exists, display the stored last visit date and time on the page.

STEP 3 : Get Current Date/Time: Use PHP's date() function to retrieve the current date and time.

STEP 4 : Set the Cookie: Store the current date and time in a cookie using setcookie(), with an expiration time of 1 hour.

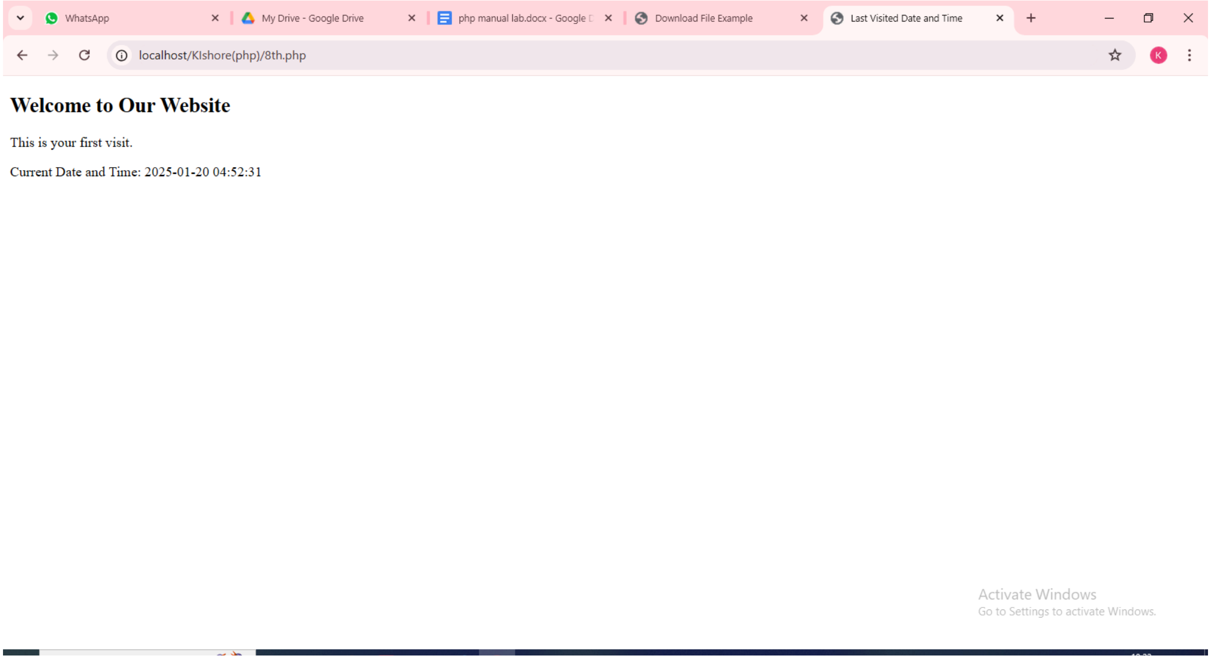
STEP 5 : Display Current Time: Show the current date and time on the page to the user.

PROGRAM :

```
<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width,
initial-scale=1.0">

   <title>Last Visited Date and Time</title>

</head>

<body>

   <h2>Welcome to Our Website</h2>
```

```
  <?php
  // Check if the cookie for last visited time exists
  if (isset($_COOKIE['last_visit'])) {
     // Display the last visited date and time
     echo "<p>Last visited: " . $_COOKIE['last_visit'] . "</p>";
  } else {
     // If the cookie doesn't exist, this is the first visit
     echo "<p>This is your first visit.</p>";
  }
  // Get current date and time
  $current_time = date('Y-m-d H:i:s');

  // Set a cookie with the current date and time, expires in 1 hour
  setcookie('last_visit', $current_time, time() + 3600);
  ?>
  <p>Current Date and Time: <?php echo $current_time; ?></p>
</body>
</html>
```

OUTPUT :



WhatsApp
My Drive - Google Drive
php manual lab.docx - Google D
Download File Example
Last Visited Date and Time
localhost/KIshore(php)/8th.php
Activate Windows
Go to Settings to activate Windows.

Welcome to Our Website

This is your first visit.

Current Date and Time: 2025-01-20 04:52:31

RESULT :

The above program entered and executed successfully.

EXERCISE 9:

Write a program to store page views count in SESSION , to increment the count on each refresh and to show the count on the web page.

AIM : To write a php program to store the views count on the web page and to increment the count on each refresh.

PROCEDURE :

STEP 1 : Start Session: Call session_start() to begin the session and store session data.

STEP 2 : Check for Existing Session Variable: Use isset() to check if the page_views session variable is set.

STEP 3 : Increment Count: If the session variable is set, increment the page_views count by 1.

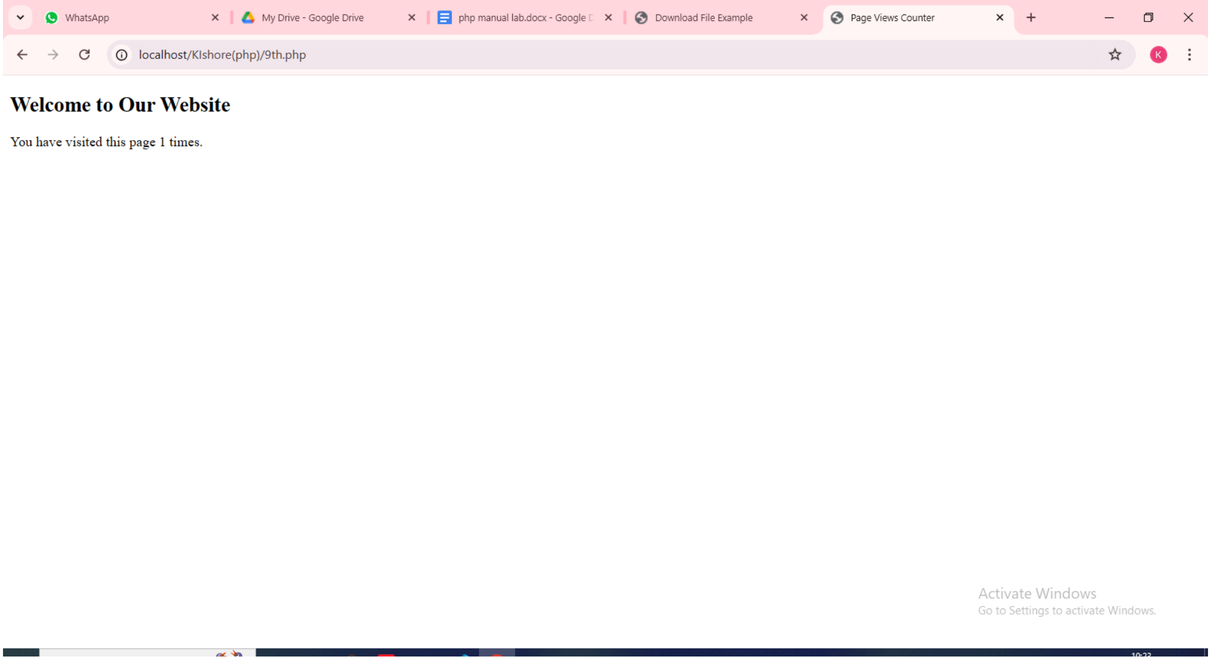
STEP 4 : Initialize Count: If the session variable is not set, initialize page_views to 1 for the first visit.

STEP 5 : Display Count: Echo the page views count to show the number of visits on the web page.

PROGRAM :

```
<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width,
initial-scale=1.0">

   <title>Page Views Counter</title>

</head>

<body>

   <h2>Welcome to Our Website</h2>


   <?php

   // Start the session
   session_start();

   // Check if the 'page_views' session variable is set
   if (isset($_SESSION['page_views'])) {
      // Increment the page views count
      $_SESSION['page_views']++;
   } else {
      // Initialize page views count to 1 for the first visit
      $_SESSION['page_views'] = 1;
   }

   // Display the page views count
   echo "<p>You have visited this page " . $_SESSION['page_views'] . "
times.</p>";
   ?>

</body>
</html>
```

OUTPUT :



WhatsApp
My Drive - Google Drive
php manual lab.docx - Google D
Download File Example
Page Views Counter
localhost/KIshore(php)/9th.php
Activate Windows
Go to Settings to activate Windows.

Welcome to Our Website

You have visited this page 1 times.

RESULT :

The above program entered and executed successfully.

EXERCISE 10 :

Write a program to design a simple calculator.

AIM : To write a php program to do simple arithmetic calculations .

PROCEDURE :

STEP 1 : Create an HTML form to input two numbers and select an operation (addition, subtraction, multiplication, division, modulus).

STEP 2 : On form submission, use PHP to retrieve the numbers and selected operation.

STEP 3 : Use a switch statement to determine the operation to perform.

STEP 4 : Perform the appropriate mathematical operation based on the selected operation.

STEP 5 : Display the result of the calculation or an error message for invalid operations or division by zero**.**

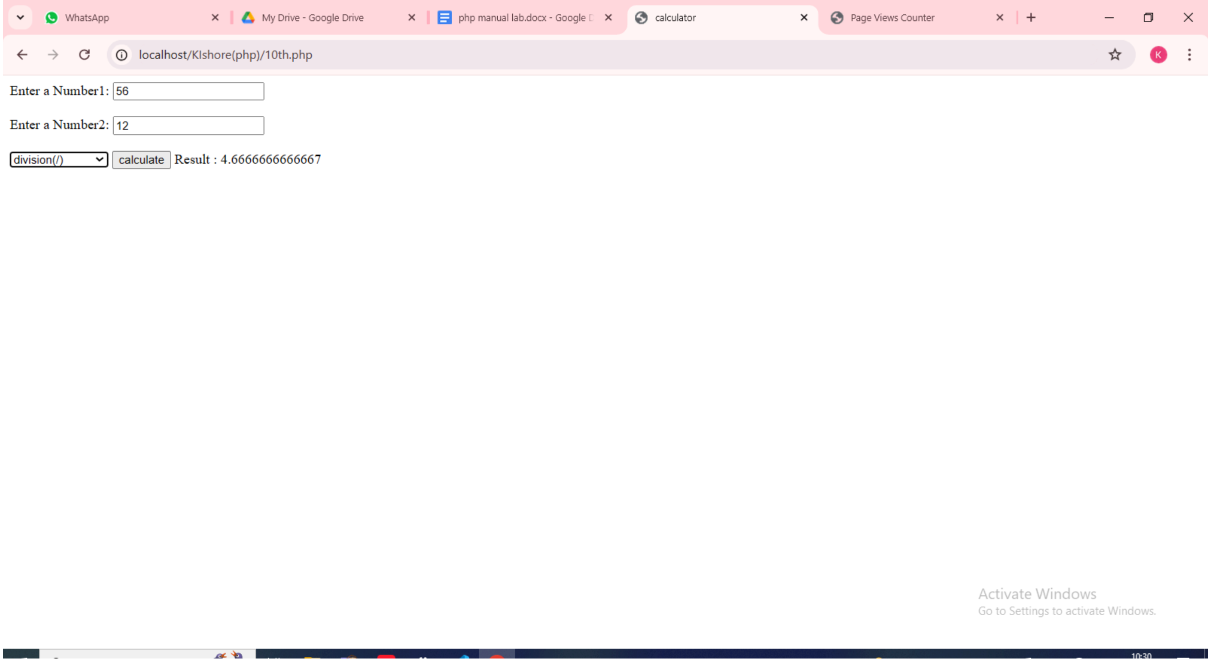
PROGRAM :

```
<html>
  <head>
    <title>calculator</title>
</head>
<body>
  <form method="POST" action="">
    <label for="number1">Enter a Number1:</label>
    <input type="number" name="number1"><br><br>
    <label for="number2">Enter a Number2:</label>
```

```
        <input type="number" name="number2"><br><br>
         <select name="operation">
            <option value="add">Addition(+)</option>
            <option value="sub">Subtraction(-)</option>
            <option value="mul">Multiplication(*)</option>
            <option value="div">division(/)</option>
            <option value="mod">Modulus(%)</option>
</select>
<button type="submit">calculate</button>
<?php
if($_SERVER["REQUEST_METHOD"]=="POST")
{
 $num1=intval($_POST['number1']);
 $num2=intval($_POST['number2']);
 $operation=$_POST['operation'];
 $result="";
 switch($operation){
    case "add":
        $result=$num1+$num2;
        break;

case "add":
    $result=$num1+$num2;
    break;
case "sub":
     $result=$num1-$num2;
    break;
case "mul":
    $result=$num1*$num2;
```

```
   break;
case "div":
   if($num2==0){
      echo "divisor should be greater than zero";
   }
   else{
      $result=$num1/$num2;
   }
   break;
case "mod":
   $result=$num1%$num2;
   break;
default:
$result="invalid input";
break;
}
echo "Result : $result";
}
?>
</body>
   </html>
```

OUTPUT :


WhatsApp
My Drive - Google Drive
php manual lab.docx - Google D
calculator
Page Views Counter
localhost/KIshore(php)/10th.php
Activate Windows
Go to Settings to activate Windows.

Enter a Number1: 56

Enter a Number2: 12

division(/) calculate Result : 4.6666666666667

RESULT :

The above program entered and executed successfully.

EXERCISE 11:

Design an authentication web page in php with MYSQL to check username and password.

AIM : To design a web page with mysql to check username and password.

PROCEDURE :

STEP 1 : Connect to MySQL database: Use mysqli_connect("localhost", "root", "", "example") to establish a connection.

STEP 2 : Check connection: Ensure the connection is successful with if (!$connection) { die('Could not connect...') }.

STEP 3 : Execute query: Run the query with mysqli_query($connection, "SELECT * FROM fruit").

STEP 4 : Display data: Loop through the result with mysqli_fetch_assoc() and output in HTML <table>.

STEP 5 : Close connection: After displaying the data, close the connection with mysqli_close($connection).

PROGRAM :

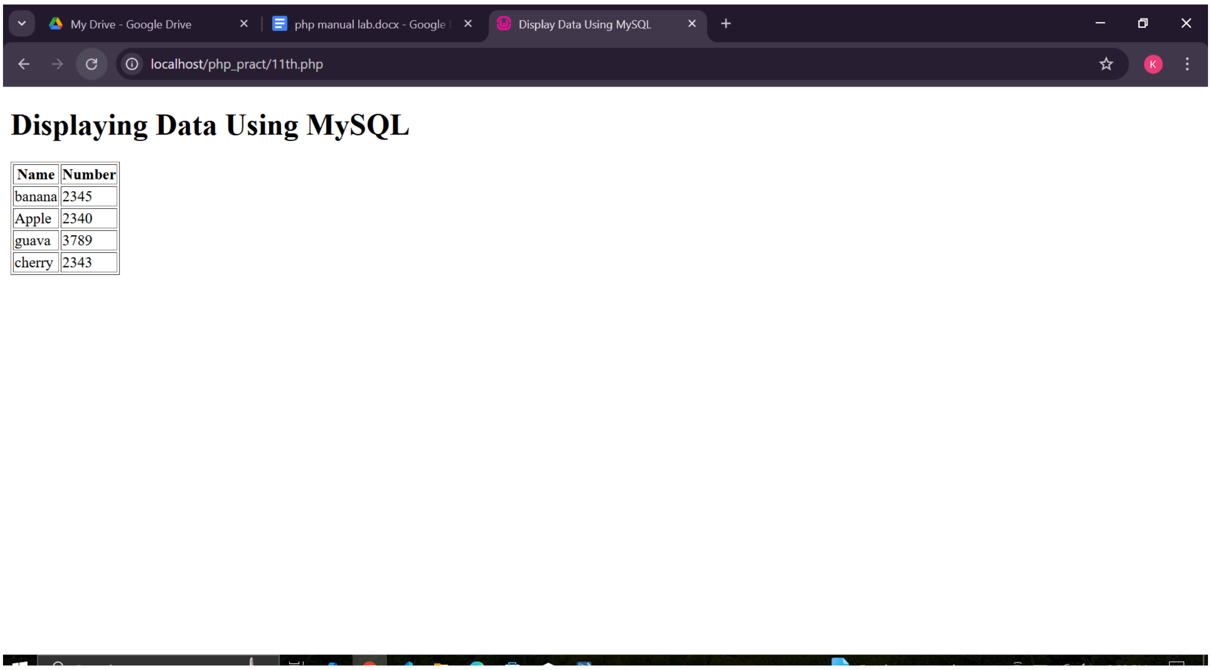
```
<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width,
initial-scale=1.0">

   <title>Display Data Using MySQL</title>
```

```
</head>
<body>
  <h1>Displaying Data Using MySQL</h1>
  <?php
  // Establish the connection to the MySQL database
  $connection = mysqli_connect("localhost", "root", "", "example");

  // Check if the connection was successful
  if (!$connection) {
    die('Could not connect to server: ' . mysqli_connect_error());
  }

  // Query to fetch data from the 'fruit' table
  $query = "SELECT * FROM fruit";
  $result = mysqli_query($connection, $query);

  // Check if the query was successful
  if (!$result) {
    die("Query failed: " . mysqli_error($connection));
  }
```

```
    // Start creating the HTML table
    echo "<table border='1'>";
    echo "<tr><th>Name</th><th>Number</th></tr>";

    // Fetch and display the data in the table
    while ($row = mysqli_fetch_assoc($result)) {
        echo "<tr>";
        echo "<td>" ,($row['Name']) , "</td><td>" , ($row['Number']) ,
"</td>";
        echo "</tr>";
    }

    // Close the table
    echo "</table>";

    // Close the MySQL connection
    mysqli_close($connection);
    ?>
</body>
```

```
</html>
```

OUTPUT :



Displaying Data Using MySQL

Name	Number
banana	2345
Apple	2340
guava	3789
cherry	2343

RESULT :

The above program entered and executed successfully.