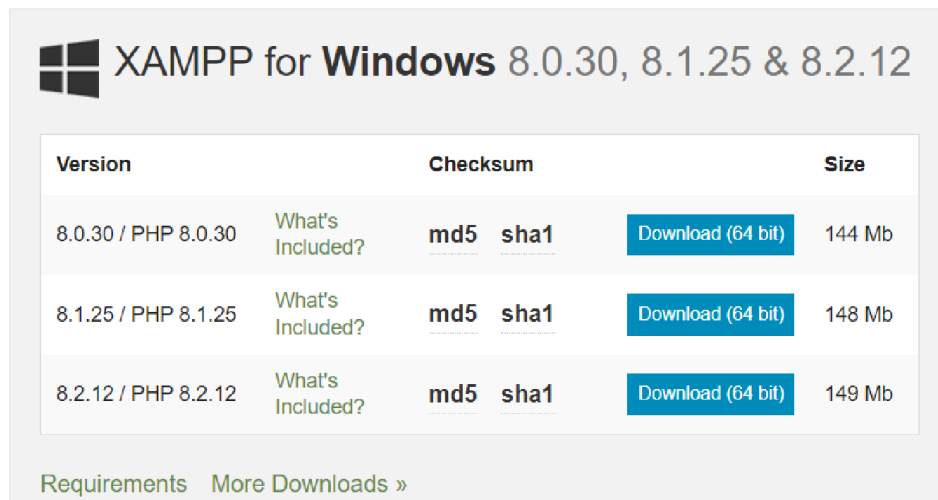


PRACTICAL - 01

Aim: - A] Install and configure PHP, Web Server and MySQL database Using XAMPP/WAMP/LAMP/MAMP.

STEP 1- Open any web browser and visit <https://www.apachefriends.org/index.html>. On the home page, you can find the option to download XAMPP for three platforms- Windows, MAC, and Linux. Click on **XAMPP for Windows**. The latest version available on the website is **7.4.5**.



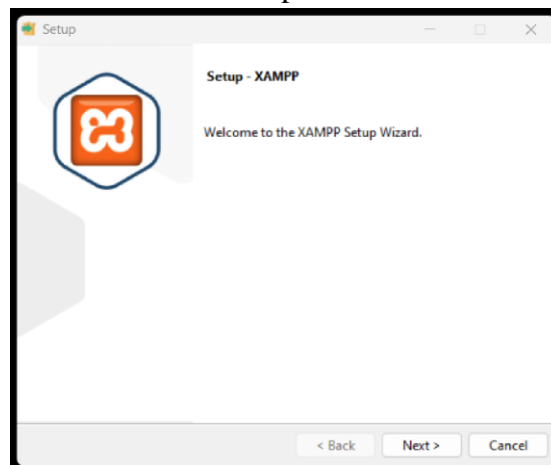
XAMPP for Windows 8.0.30, 8.1.25 & 8.2.12

Version	Checksum	Size
8.0.30 / PHP 8.0.30	md5 sha1	144 Mb
8.1.25 / PHP 8.1.25	md5 sha1	148 Mb
8.2.12 / PHP 8.2.12	md5 sha1	149 Mb

[Requirements](#) [More Downloads »](#)

STEP 2- After the download is completed, double click the .exe extension file to start the process of installation.

STEP 3- A pop-up screen with the message asking you to allow making changes on your desktop appears. Click "YES" to continue the process.

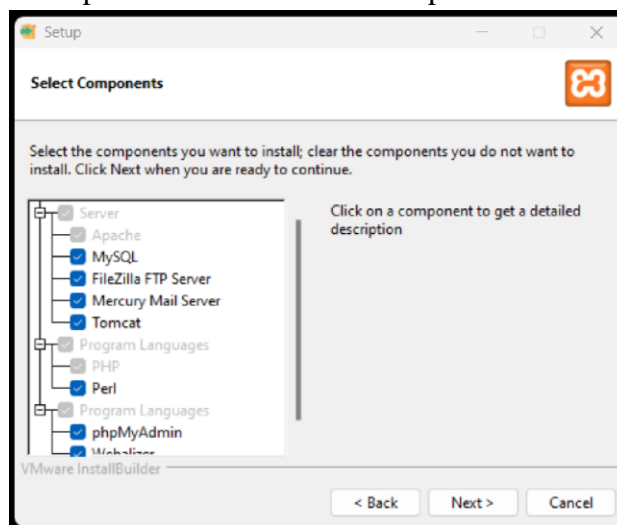


STEP 4- Click to **Allow access** or deactivate the firewall and any other antivirus software because it can hamper the process of installation. Thus, it is required to temporarily disable any antivirus software or security firewall till the time all the XAMPP components have been installed completely.

STEP 5- Just before the installation, a pop-up window appears with a warning to **disable UAC**. User Account Control (UAC) interrupts the XAMPP installation because it restricts the access to write to the C: drive. Therefore, it is suggested to disable it for the period of installation.

STEP 6- After clicking the .exe extension file, the XAMPP setup wizard opens spontaneously. Click on "NEXT" to start the configuration of the settings.

STEP 7- After that, a 'Select Components' panel appears, which gives you the liberty to choose amongst the separate components of the XAMPP software stack for the installation. To get a complete localhost server, it is recommended to install using the default options of containing all available components. Click "NEXT" to proceed further.



STEP 8- The setup is now ready to install, and a pop-up window showing the same appears on the screen. Click "NEXT" to take the process forward.

STEP 9- Select the location where the XAMPP software packet needs to be installed. The original setup creates a folder titled XAMPP under C:\ for you. After choosing a location, click "NEXT".

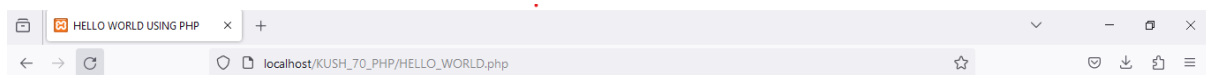
STEP 11- After the successful installation of the XAMPP setup on your desktop, press the "FINISH" button.

Aim: - B] Create a web page that displays “Hello World.”

CODE: -

```
<!DOCTYPE html>
<html>
<head>
    <meta charset='utf-8'>
    <meta http-equiv='X-UA-Compatible' content='IE=edge'>
    <title>HELLO WORLD USING PHP</title>
    <meta name='viewport' content='width=device-width, initial-
scale=1'>
    <style>
        body
        {
            font-size: 28px;
            text-align: center;
            color: red;
        }
    </style>
</head>
<body>
    <div class="container">
        <?php echo "<h1>Hello World!</h1>"; ?>
    </div>
</body>
</html>
```

OUTPUT:-



Hello World!

PRACTICAL - 02

Aim: - Create a web page that collects user information using a form and displays it when the user clicks the submit button.

CODE:-

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>USER INFORMATION FORM</title>
</head>
<body>
<div class="container">
    <h1>User Information</h1>
    <form action="" method = "POST" id="userForm">
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required placeholder =
"Kush Amit Shah"><br>
        <label for="age">Age:</label>
        <input type="number" id="age" name="age" min="0" required
placeholder = "17"><br>
        <label for="age">Phone Number:</label>
        <input type="number" id="pn" name="pn" min="0" required
placeholder = "2260703070"><br>
        <label for="age">Email:</label>
        <input type="emial" id="email" name="email" required placeholder
= "kush@bbit.ac.in"><br>
        <label for="age">City:</label>
        <input type="text" id="city" name="city" required placeholder =
"Nadiad"><br>
        <button type="submit">Submit</button>
    </form>
    <h2>KINDLY CHECK YOUR DETAILS:</h2>
    <?PHP
        if($_SERVER['REQUEST_METHOD'] == "POST"){
            $name_final = filter_var($_POST['name']);
            $age_final = filter_var($_POST['age'],
FILTER_SANITIZE_NUMBER_INT);
            $pn_final = filter_var($_POST['pn'],
FILTER_SANITIZE_NUMBER_INT);
```

```
$email_final = filter_var($_POST['email'],  
FILTER_SANITIZE_EMAIL);  
$city_final = filter_var($_POST['city']);  
echo "<p>NAME : " . $name_final. " </p>";  
echo "<p>AGE : " . $age_final. "</p>";  
echo "<p>NAME : " . $pn_final. " </p>";  
echo "<p>AGE : " . $email_final. "</p>";  
echo "<p>CITY : " . $city_final. " </p>";  
}  
?></div></body></html>
```

OUTPUT :-

The image shows a web form titled "User Information" with a dark background. The form contains several input fields with labels: "Name:" (Kush Amit Shah), "Age:" (17), "Phone Number:" (2260703070), "Email:" (kush@bbit.ac.in), and "City:" (Nadiad). A green "Submit" button is at the bottom. Below the form, the output is displayed in a white box with the heading "KINDLY CHECK YOUR DETAILS:". The output lists the entered details: NAME : Kush Amit Shah, AGE : 17, NAME : 226040307070, AGE : kush@bbit.ac.in, and CITY : Nadiad.

User Information

Name: Kush Amit Shah

Age: 17

Phone Number: 2260703070

Email: kush@bbit.ac.in

City: Nadiad

Submit

KINDLY CHECK YOUR DETAILS:

NAME : Kush Amit Shah

AGE : 17

NAME : 226040307070

AGE : kush@bbit.ac.in

CITY : Nadiad

PRACTICAL - 03

Aim: - A] Write a script to implement a simple calculator for mathematical operations.

CODE:-

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>SIMPLE CALCULATOR</title>
<?php
    $result = 0;
    if(isset($_POST['var'])){
        $num_1 = $_POST['num1'];
        $num_2 = $_POST['num2'];
        $operator = $_POST['var'];
        if ($operator == "+") {
            $result = $num_1 + $num_2;
        }
        else if ($operator == "-"){
            $result = $num_1 - $num_2;
        }
        else if ($operator == "*"){
            $result = $num_1 * $num_2;
        }
        else if ($operator == "/"){
            $result = $num_1 / $num_2;
        }
    }
?>
</head>
<body>
    <div class="container">
        <form action="" method="POST">
            <h1 style="color: red;">CALCULATOR</h1>
            <label for="num1">ENTER THE FIRST NUMBER</label>
            <input type="text" name="num1" placeholder="EG: 10"
required><br>

            <label for="num1">ENTER THE SECOND NUMBER</label>
            <input type="text" name="num2" placeholder="EG: 20"
required><br><br>
            <button type="submit" name="var" value="+" required> +
</button>
```

```
<button type="submit" name="var" value="-" required> -  
</button>  
<button type="submit" name="var" value="*" required> *  
</button>  
<button type="submit" name="var" value="/" required> /  
</button>  
<h3>RESULT </h3><input type="text" value = "<?php echo  
$result ?>" readonly>  
</form></div></body></html>
```

OUTPUT:-

The image displays two screenshots of a web-based calculator interface. The calculator has a dark gray background with white text and buttons. The title "CALCULATOR" is in red. The first input field is labeled "ENTER THE FIRST NUMBER" and contains the value "10". The second input field is labeled "ENTER THE SECOND NUMBER" and contains the value "20". Below the input fields are four buttons: a green button with a plus sign (+), a blue button with a minus sign (-), a green button with a multiplication sign (*), and a green button with a division sign (/). Below these buttons is a label "RESULT" and a text input field containing the result. In the top screenshot, the result is "-10". In the bottom screenshot, the result is "200".

Aim: - B] A company has following payment scheme for their staff:

- a. Net Salary = Gross Salary – Deduction**
- b. Gross Salary = Basic pay + DA + HRA + Medical**
- c. Deduction = Insurance + PF**
- d. DA (Dearness Allowance) = 50% of Basic pay**
- e. HRA (House Rent Allowance) = 10% of Basic pay**
- f. Medical = 4% of Basic pay**
- g. Insurance = 7% of Gross salary**
- h. PF (Provident Fund) = 5% of Gross salary**

Write a script to take the basic salary of an employee as input and calculate the net payment to any employee.

CODE:-

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>SALARY CALCULATOR</title>
    <?php
    if (isset($_POST['SUB'])) {
        $bs = $_POST['BS'];
        $da = $hra = $med = $gs = $in = $pf = $ded = $fs =
'';
        $da = $bs * 0.50;
        $hra = $bs * 0.10;
        $med = $bs * 0.04;
        $gs = $bs + $da + $hra + $med;

        $in = $gs * 0.07;
        $pf = $gs * 0.05;

        $ded = $in + $pf;
        $fs = $gs - $ded;
    }
    ?>
</head>
<body>
    <div class="container">
        <form action="" method="POST">
```



```
        <label for="age">BASIC SALARY (IN
RUPEES):</label>
        <input type="text" name="BS"
placeholder="BASIC SALARY" value="<?php echo isset($bs) ?
$bs : ''; ?>" required>
        <button type="submit"
name="SUB">CALCULATE</button>
</form>
</div>
    <div class="container">
        <label for="age">DEARNESS ALLOWANCE (IN
RUPEES):</label>
        <input type="text" name="DA"
placeholder="DEARNESS ALLOWANCE" value="<?php echo
isset($da) ? $da : ''; ?>" required>
        <label for="age">HOUSE RENT ALLOWANCE (IN
RUPEES):</label>
        <input type="text" name="HRA"
placeholder="HOUSE RENT ALLOWANCE" value="<?php echo
isset($hra) ? $hra : ''; ?>" required>
        <label for="age">MEDICAL ALLOWANCE (IN
RUPEES):</label>
        <input type="text" name="MED"
placeholder="MEDICAL ALLOWANCE" value="<?php echo
isset($med) ? $med : ''; ?>" required>
        <label for="age">INSURANCE (IN
RUPEES):</label>
        <input type="text" name="IN"
placeholder="INSURANCE" value="<?php echo isset($in) ? $in
: ''; ?>" required>
        <label for="age">PROVIDENT FUND (IN
RUPEES):</label>
        <input type="text" name="PF"
placeholder="PROVIDENT FUND" value="<?php echo isset($pf)
? $pf : ''; ?>" required>
        <label for="age">FINAL SALARY (IN
RUPEES):</label>
        <input type="text" name="FS"
placeholder="FINAL SALARY" value="<?php echo isset($fs) ?
$fs : ''; ?>" required>
    </div></body></html>
```

OUTPUT:-

BASIC SALARY (IN RUPEES):

DEARNESS ALLOWANCE (IN RUPEES):

HOUSE RENT ALLOWANCE (IN RUPEES):

MEDICAL ALLOWANCE (IN RUPEES):

INSURANCE (IN RUPEES):

PROVIDENT FUND (IN RUPEES):

FINAL SALARY (IN RUPEES):

BASIC SALARY (IN RUPEES):

DEARNESS ALLOWANCE (IN RUPEES):

HOUSE RENT ALLOWANCE (IN RUPEES):

MEDICAL ALLOWANCE (IN RUPEES):

INSURANCE (IN RUPEES):

PROVIDENT FUND (IN RUPEES):

FINAL SALARY (IN RUPEES):

PRACTICAL - 04

Aim: - A] Write a script that reads the name of the car and displays the name of the company the car belongs to as per the below table:

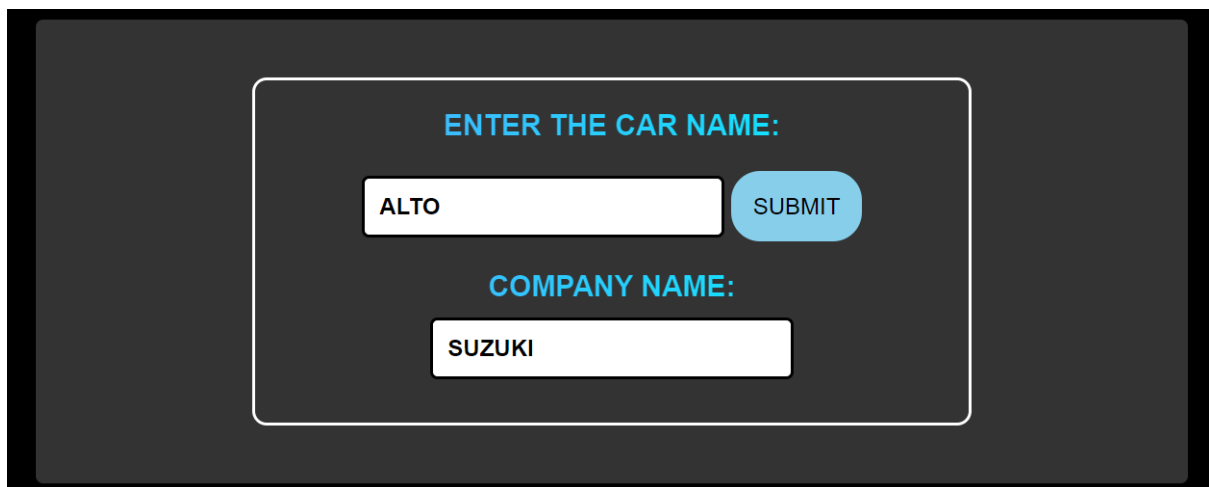
Car	Company
Safari, Nexon, Tigor, Tiago	Tata
XUV700, XUV300, Bolero	Mahindra
i20, Verna, Venue, Creta	Hyundai
Swift, Alto, Baleno, Brezza	Suzuki

CODE: -

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>CAR COMPANY FINDER</title>
<?php
    if (isset($_POST['car_name'])) {
        $name = strtoupper($_POST['car_name']);
        $comp = "";
        $name = $_POST['car_name'];

        if($name == "SAFARI" || $name == "NEXON" || $name
== "TIGOR" || $name == "TIAGO") {
            $comp = "TATA";
        } else if($name == "XUV700" || $name == "XUV300"
|| $name == "BOLERO") {
            $comp = "MAHINDRA";
        } else if($name == "I20" || $name == "VERNA" ||
$name == "VENUE" || $name == "CRETA") {
            $comp = "HYUNDAI";
        } else if($name == "SWIFT" || $name == "ALTO" ||
$name == "BOLENO" || $name == "BREZZA") {
            $comp = "SUZUKI";
        } else {
            $comp = "SORRY! NO DATA FOUND...";
        }
    }
}
```

```
    }  
  }  
?>  
</head>  
<body>  
<form action = "" method = "POST">  
  <div class="container">  
    <label for="car">ENTER THE CAR NAME:</label>  
    <input type="text" name="car_name"  
placeholder="EG: MERCEDES" required>  
    <button type="submit" name="SUB">SUBMIT</button>  
    <label for="comp_name">COMPANY NAME:</label>  
    <input type="text" name="res" placeholder="COMPANY  
NAME" value="<?php echo isset($comp) ? $comp : ''; ?>"  
readonly>  
  </div>  
</body>  
</html>
```

OUTPUT:-

The screenshot shows a web form with a dark gray background. It contains two input fields. The first input field is labeled "ENTER THE CAR NAME:" and contains the text "ALTO". To the right of this input field is a blue button labeled "SUBMIT". Below this is a second input field labeled "COMPANY NAME:" which contains the text "SUZUKI".

Aim: - B] Write a script to read the marks of 4 subjects and display the result as per the below instructions:

GTU GRADE	MARK-RANGE
AA	85-100
AB	75-84
BB	65-74
BC	55-64
CC	45-54
CD	40-44
DD	35-39
FF	<35 (FAIL)

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>GTU GRADE CALCULATOR</title>
    <?php
    if (isset($_POST['calculate'])) {
        $subjects = array("ADVANCED OBJECT ORIENTED
PROGRAMMING", "INTRODUCTION TO SOFTWARE ENGINEERING",
"COMPUTER NETWORKING", "INTRODUCTION TO WEB DEVELOPMENT");
        $marks = array($_POST['aoop'], $_POST['ise'],
$_POST['cn'], $_POST['iwd']);
        $grades = array();

        echo "<h3>RESULT</h3>";
        echo "<table>";
        echo
"<tr><th>SUBJECT</th><th>MARKS</th><th>GRADE</th></tr>";

        for ($i = 0; $i < count($subjects); $i++) {
```

```

$mark = $marks[$i];
if ($mark >= 85) $grade = "AA";
elseif ($mark >= 75) $grade = "AB";
elseif ($mark >= 65) $grade = "BB";
elseif ($mark >= 55) $grade = "BC";
elseif ($mark >= 45) $grade = "CC";
elseif ($mark >= 40) $grade = "CD";
elseif ($mark >= 35) $grade = "DD";
else $grade = "FF";
echo "<tr><td>" . $subjects[$i] . "</td><td>" .
$mark . "</td><td><input type='text' value='" . $grade .
"' readonly></td></tr>";
}
echo "</table>";
$overallResult = "CONGRATULATIONS! YOU HAVE PASSED
THE EXAM.";
$mark1 = $_POST['aop'];
$mark2 = $_POST['ise'];
$mark3 = $_POST['cn'];
$mark4 = $_POST['iwd'];
if ($mark1 < 35 || $mark2 < 35 || $mark3 < 35 ||
$mark4 < 35) {
    $overallResult = "NEEDS IMPROVEMENT";
}
echo "<h3>OVERALL RESULT: $overallResult</h3>";
}
?>
</head>
<body>
    <form action="" method="post">
<table id="mainTable">
<tr>
<td><label for="aop">ADVANCED OBJECT ORIENTED
PROGRAMMING</label></td>
        <td><input type="number" name="aop"
min="0" max="100" required></td>
</tr><tr>
<td><label for="ise">INTRODUCTION TO SOFTWARE
ENGINEERING</label></td>
        <td><input type="number" name="ise" min="0" max="100"
required></td><tr><tr>
<td><label for="cn">COMPUTER NETWORKING</label></td>

```

```

<td><input type="number" name="cn" min="0" max="100"
required></td></tr><tr>
<td><label for="iwd">INTRODUCTION TO WEB
DEVELOPMENT</label></td>
td><input type="number" name="iwd" min="0" max="100"
required></td></tr></table><br>
<button type="submit" name="calculate">CALCULATE
RESULT</button></form>
</body>
</html>

```

OUTPUT: -

ADVANCED OBJECT ORIENTED PROGRAMMING	100
INTRODUCTION TO SOFTWARE ENGINEERING	100
COMPUTER NETWORKING	100
INTRODUCTION TO WEB DEVELOPMENT	100

CALCULATE RESULT

RESULT		
SUBJECT	MARKS	GRADE
ADVANCED OBJECT ORIENTED PROGRAMMING	100	AA
INTRODUCTION TO SOFTWARE ENGINEERING	100	AA
COMPUTER NETWORKING	100	AA
INTRODUCTION TO WEB DEVELOPMENT	100	AA
OVERALL RESULT: CONGRATULATIONS! YOU HAVE PASSED THE EXAM.		

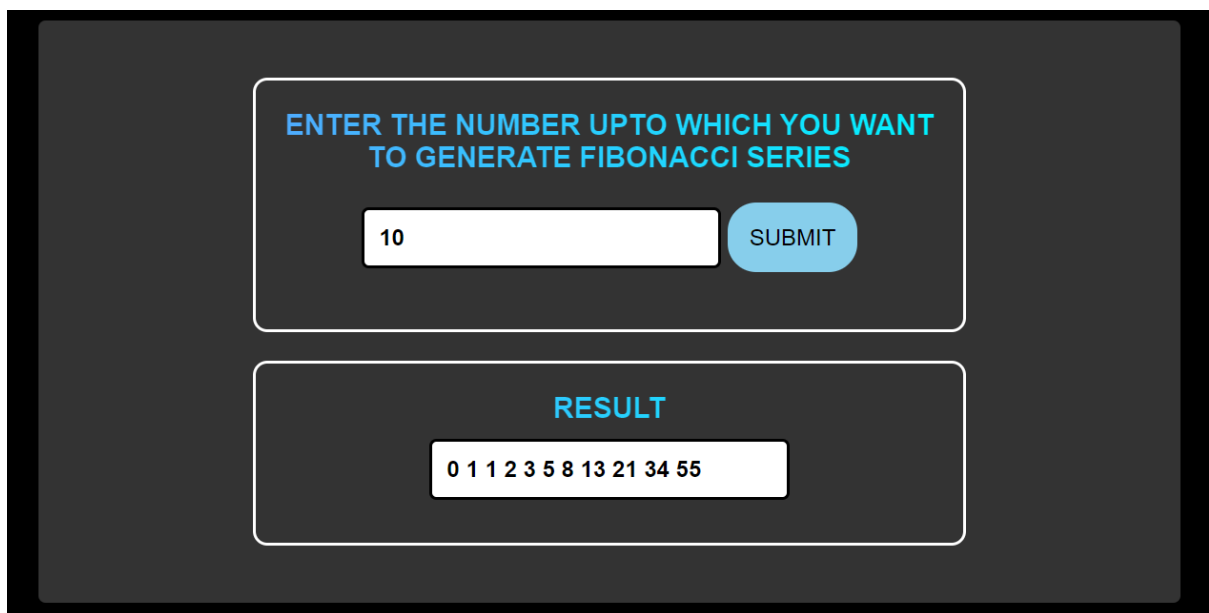
Aim: - C] Write a script to display Fibonacci numbers up to a given term.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>FIBONACCI CREATOR</title>
<?php
    $series = " ";
    if (isset($_POST['size'])) {
        $fibo = $_POST['size'];
        $a = 0;
        $b = 1;
        $series = "";
        $result = 0;
        for($i = 0 ; $i <= $fibo ; $i++){
            $series .= $a . " ";
            $result = $a + $b;
            $a = $b;
            $b= $result;
        }
    }
?>
</head>
<body>
<form action = '' method = 'POST'>
    <div class="container">
        <label for = "size">ENTER THE NUMBER UPTO WHICH
YOU WANT TO GENERATE FIBONACCI SERIES</label>
```



```
<input type = "number" name = "size" placeholder =  
" EG:- 10">  
    <button type="submit" name="SUB">SUBMIT</button>  
</div>  
<div class="container">  
    <label for = "size">RESULT</label>  
    <input type = "text" name = "res" readonly value =  
"<?php echo $series?>" placeholder = "RESULT">  
    </div>  
</form>  
</body>  
</html>
```

OUTPUT: -

ENTER THE NUMBER UPTO WHICH YOU WANT TO GENERATE FIBONACCI SERIES

10 SUBMIT

RESULT

0 1 1 2 3 5 8 13 21 34 55

Aim: - D] Write a script to display a multiplication table for the given number.

CODE: -

```
<!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>
    <?php
        $series = " ";
        if (isset($_POST['number'])) {
            $num = $_POST['number'];
            $result = 0;
            for ($i = 1 ; $i <= 10 ; $i++) {
                $result = $num * $i;
                $series .= "$num * $i = $result".PHP_EOL;
            }
        }
    ?>
</head>
<body>
<form action = '' method = 'POST'>
    <div class="container">
        <label for = "size">ENTER THE NUMBER OF WHICH YOU
WANT TO GENERATE MULTIPLICATION TABLE</label>
        <input type = "number" name = "number" placeholder
= " EG:- 10">
        <button type="submit" name="SUB">SUBMIT</button>
    </div>
    <div class="container">
        <label for = "size">MULTIPLICATION TABLE OF <?php
echo $num?>:</label>
        <textarea><?php echo $series;?> </textarea>
    </div>
</form>
</body>
</html>
```

OUTPUT: -

The screenshot displays a web interface on a dark background. At the top, a light blue rounded rectangle contains the text "ENTER THE NUMBER OF WHICH YOU WANT TO GENERATE MULTIPLICATION TABLE" in blue. Below this text is a white input field containing the number "13" and a light blue "SUBMIT" button. Below the input field, another light blue rounded rectangle contains the text "MULTIPLICATION TABLE OF 13:" in blue. Underneath this text is a white rectangular area with a scrollbar, displaying the multiplication table for 13 from 1 to 10. The table entries are as follows:

13	*	1	=	13
13	*	2	=	26
13	*	3	=	39
13	*	4	=	52
13	*	5	=	65
13	*	6	=	78
13	*	7	=	91
13	*	8	=	104
13	*	9	=	117
13	*	10	=	130

PRACTICAL - 05

Aim: - A] Write a script to calculate the length of a string and count the number of words in the given string without using string functions.

CODE: -

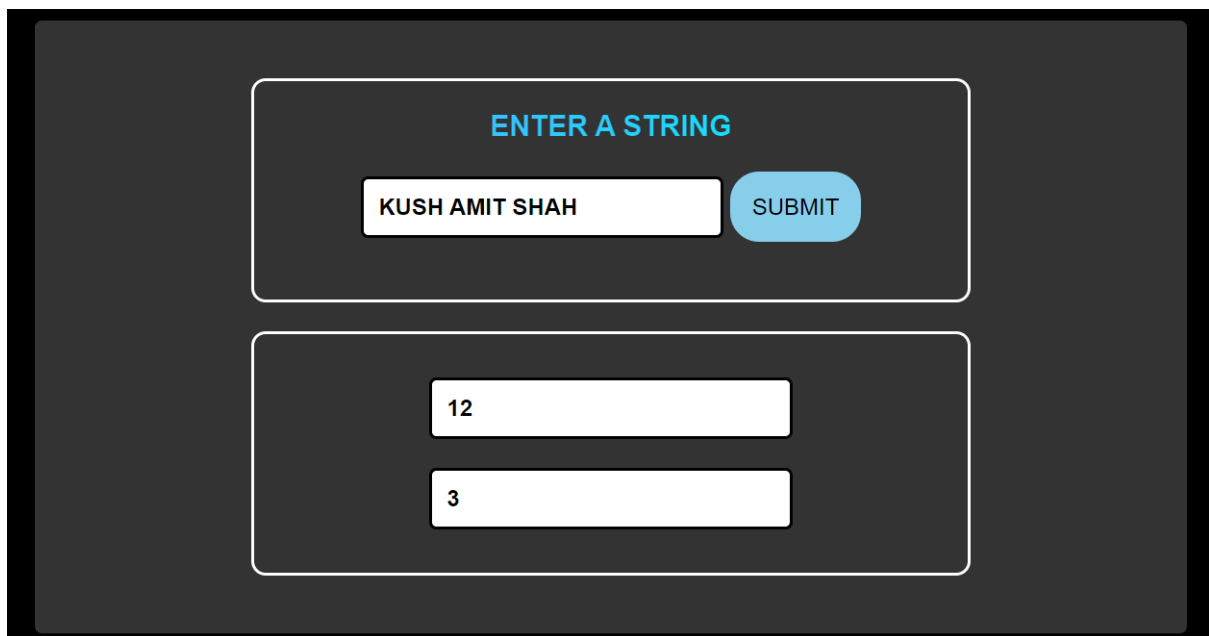
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>LENGTH AND WORDS COUNTER</title>
<?php
    $length = 0;
    $wordCount = 0;
    if (isset($_POST['string'])) {
        $string = $_POST['string'];

        for ($i = 0; $i < strlen($string); $i++) {
            if ($string[$i] != " ") {
                $length++;
            }
            if ($string[$i] == " " && $string[$i-1] != " ")
{
                $wordCount++;
            }
        }

        if ($length > 0 && $string[0] != " ") {
            $wordCount++;
        }
    }
?>
</head>
<body>
<form action = "" method = "POST">
    <div class="container">
        <label for = "string">ENTER A STRING</label>
        <input type = "text" name = "string" placeholder =
" EG: KUSH SHAH">
        <button type="submit" name="SUB">SUBMIT</button>
    </div>
```

```
<div class="container">
  <input type = "text" name = "length" value =
  "<?php echo $length?>" placeholder = " LENGTH" readonly>
  <input type = "text" name = "wordcount" value =
  "<?php echo $wordCount?>" placeholder = " WORD COUNT"
  readonly>
</div>
</form>
</body>
</html>
```

OUTPUT: -



The screenshot displays a web application interface on a dark background. At the top, a rounded rectangle contains the title 'ENTER A STRING' in cyan. Below the title is a text input field containing 'KUSH AMIT SHAH' and a blue 'SUBMIT' button. Below this, another rounded rectangle contains two output fields: the first shows '12' (the length of the string) and the second shows '3' (the word count of the string).

Aim: - B] Write a script to sort a given indexed array.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>SORTING OF ARRAY</title>
<?php
    $result = " ";
    $array = array();
    if (isset($_POST['SUB'])) {
        $array = array(10, 30, 20, 50, 40, 70, 60, 90, 80,
100);
        sort($array);
        $result = implode(", ", $array);
    }
?>
</head>
<body>
<form action = "" method = "POST">
    <div class="container">
        <label for="org">ORIGINAL ARRAY</label>
        <textarea name = "org"><?php echo "10, 30, 20, 50, 40,
70, 60, 90, 80, 100";?></textarea>
        <button type="submit" name="SUB">SUBMIT</button>
    </div>
    <div class="container">
        <label for = "result">RESULTANT ARRAY</label>
        <textarea name = "result"><?php echo
$result;?></textarea>
    </div>
</form>
</body>
</html>
```

OUTPUT: -

The screenshot shows a web application interface with a dark background. It features two main sections, each enclosed in a rounded rectangle with a white border. The top section is titled "ORIGINAL ARRAY" in cyan text. Below the title is a text input field containing the array elements "10, 30, 20, 50, 40, 70, 60, 90, 80, 100" on two lines. To the right of the input field is a light blue rounded button labeled "SUBMIT". The bottom section is titled "RESULTANT ARRAY" in cyan text. Below this title is a text input field containing the sorted array elements "10, 20, 30, 40, 50, 60, 70, 80, 90, 100" on two lines. The entire interface is set against a dark gray background within a black frame.

ORIGINAL ARRAY

10, 30, 20, 50, 40,
70, 60, 90, 80, 100

SUBMIT

RESULTANT ARRAY

10, 20, 30, 40, 50,
60, 70, 80, 90, 100

Aim: - C] Write a script to perform 3 x 3 matrix Multiplication.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>3 X 3 MATRIX MULTIPLICATION</title>
<?php
    $res = " ";
    if (isset($_POST['SUB1'])) {
        $arr1 = $_POST['arr1'];
        $arr2 = $_POST['arr2'];
        $result = array();
        for ($i = 0; $i <= 2; $i++) {
            for ($j = 0; $j <= 2; $j++) {
                $result[$i][$j] = 0;
                for ($k = 0; $k <= 2; $k++) {
                    $result[$i][$j] += $arr1[$i][$k] *
$arr2[$k][$j];
                }
            }
        }
        echo "<table>";
        for ($i = 0; $i <= 2; $i++) {
            echo "<tr>";
            for ($j = 0; $j <= 2; $j++) {
                echo "<td><input type='text' value='"
. $result[$i][$j] . "' readonly></td>";
            }
            echo "</tr>";
        }
        echo "</table>";

        $res = json_encode($result);
    }
?>
</head>
<body>
    <form action="" method="POST">
```



```

<div class="container">
  <table>
    <th>MATRIX - 01</th>
    <tr>
      <td><input type="text"
name="arr1[0][0]" placeholder="10" required></td>
      <td><input type="text"
name="arr1[0][1]" placeholder="20" required></td>
      <td><input type="text"
name="arr1[0][2]" placeholder="30" required></td>
    </tr>

    <tr>
      <td><input type="text"
name="arr1[1][0]" placeholder="10" required></td>
      <td><input type="text"
name="arr1[1][1]" placeholder="20" required></td>
      <td><input type="text"
name="arr1[1][2]" placeholder="30" required></td>
    </tr>

    <tr>
      <td><input type="text"
name="arr1[2][0]" placeholder="10" required></td>
      <td><input type="text"
name="arr1[2][1]" placeholder="20" required></td>
      <td><input type="text"
name="arr1[2][2]" placeholder="30" required></td>
    </tr>
    <tr>
      <td>
        <br>
        <button type="submit"
name="SUB1">SUBMIT</button>
      </td>
    </tr>
    <th>MATRIX - 02</th>
    <tr>
      <td><input type="text"
name="arr2[0][0]" placeholder="10" required></td>
      <td><input type="text"
name="arr2[0][1]" placeholder="20" required></td>

```

```
        <td><input type="text"
name="arr2[0][2]" placeholder="30" required></td>
    </tr>

    <tr>
        <td><input type="text"
name="arr2[1][0]" placeholder="10" required></td>
        <td><input type="text"
name="arr2[1][1]" placeholder="20" required></td>
        <td><input type="text"
name="arr2[1][2]" placeholder="30" required></td>
    </tr>

    <tr>
        <td><input type="text"
name="arr2[2][0]" placeholder="10" required></td>
        <td><input type="text"
name="arr2[2][1]" placeholder="20" required></td>
        <td><input type="text"
name="arr2[2][2]" placeholder="30" required></td>
    </tr>
    <tr>
        <td>
            <br>
            <button type="submit"
name="SUB">SUBMIT</button>
        </td>
    </tr>
</table>
</div>
</form>
</body>
</html>
```

OUTPUT: -

MATRIX - 01

1	2	3
4	5	6
7	8	9

SUBMIT

MATRIX - 02

9	8	7
6	5	4
3	2	1

SUBMIT

30	24	18
84	69	54
138	114	90

Aim: - D] Write a script to encode a given message into equivalent Morse code.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>MORSE CODE GENERATOR</title>
<?php
    $series = " ";
    if (isset($_POST['submit']))
    {
        $string = $_POST["string"];
        $string_main = strtolower($string);

        $array1 = array(
            "a"=>".-", "b"=>"-...", "c"=>"-.-.", "d"=>"-
..", "e"=>"..", "f"=>"..-.",
            "g"=>"--.", "h"=>"....", "i"=>"..", "j"=>"....",
            "k"=>"-.-", "l"=>"-.-.",
            "m"=>"--", "n"=>"-.", "o"=>"---", "p"=>"-.-.",
            "q"=>"--.-", "r"=>"-.-.",
            "s"=>"...", "t"=>"-", "u"=>"...-", "v"=>"...-",
            "w"=>"-.-", "x"=>"-.-.-",
            "y"=>"-.-.-", "z"=>"-.-.-", "0"=>"-----",
            "1"=>"-.-.-.-", "2"=>"-.-.-.-", "3"=>"-.-.-.-",
            "4"=>"-.-.-.-", "5"=>"-.-.-.-", "6"=>"-.-.-.-",
            "7"=>"-.-.-.-", "8"=>"-.-.-.-", "9"=>"-.-.-.-",
            "."=>"-.-.-.-", ","=>"-.-.-.-", "?"=>"-.-.-
..", "/"=>"-.-.-.-", " "=>" ");
        for($i=0;$i<strlen($string_main);$i++){
            foreach($array1 as $letter => $code)
            {
                if($letter == $string_main[$i])
                {
                    $series .= "$code";
                }
            }
        }
    }
}
```


PRACTICAL - 06

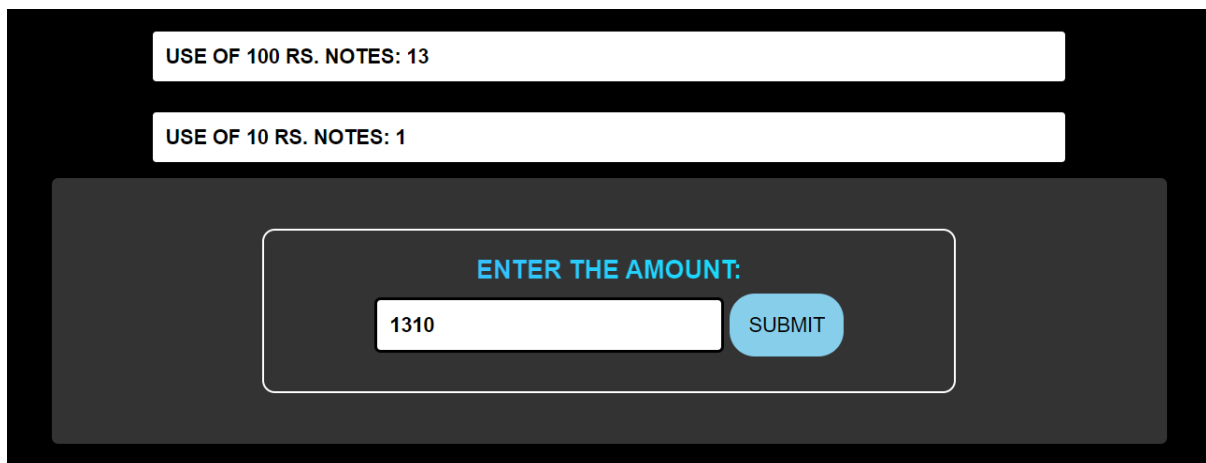
Aim: - A] Consider a currency system in which there are notes of 7 denominations, namely Rs. 1, Rs. 2, Rs. 5, Rs. 10, Rs. 20, Rs. 50 and Rs. 100. Write a function that computes the smallest number of notes that will combine for a given amount of money.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>NOTES CALCULATOR</title>
<?php
    $ans = " ";
    if (isset($_POST['submit'])){
        $amt = $_POST['amt'];
        function countnotes($amt){
            $var = 1;
            while ($var != 0){
                if ($amt >= 100){
                    $temp = 100;
                }else if ($amt >= 50){
                    $temp = 50;
                }else if ($amt >= 20){
                    $temp = 20;
                }else if ($amt >= 10){
                    $temp = 10;
                }else if ($amt >= 5){
                    $temp = 5;
                }else if ($amt >= 2){
                    $temp = 2;
                }else if ($amt >= 1){
                    $temp = 1;
                } else{
                    $ans .= $amt;
                }

                $amount = $amt / $temp;
                $notes = floor(floatval($amount));
```

```
    echo "<input type='text' name='res' value='USE OF $temp
RS. NOTES: $notes' readonly>";
    $amt = fmod($amt , $temp);
    if ($amt == 0){
        $var = 0;}}}}
    countnotes($amt);}
    ?>
</head>
<body>
<form action="" method="POST">
    <div class="container">
        <label for="amt">ENTER THE AMOUNT:</label>
        <input type="text" name="amt" placeholder="1310"
required>
        <button type="submit"
name="submit">SUBMIT</button>
    </div>
</form>
</body>
</html>
```

OUTPUT: -

USE OF 100 RS. NOTES: 13

USE OF 10 RS. NOTES: 1

ENTER THE AMOUNT:

1310 SUBMIT

Aim: - B] Write scripts using string functions:

- a. to check if the given string is lowercase or not.**
- b. to reverse the given string.**
- c. to remove white spaces from the given string.**
- d. to replace the given word from the given string.**

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>STRING FUNCTIONS</title>
</head>
<body>
<form action = "" method = "POST">
    <div class="container">
        <label for = "string">ENTER A STRING</label>
        <input type = "text" name = "string" placeholder =
" EG: KUSH SHAH" required>
        <button type="submit" name="SUB">SUBMIT</button>
    </div>
</form>
</body>
<?php
    $string_low = " ";
    $string_rev = " ";
    $string_trimmed = " ";
    $string_before = "HELLO WORLD!";
    $string_after = " ";
    if (isset($_POST['string'])) {
        $string = $_POST['string'];
        $string_low = strtolower($string);
        $string_rev = strrev($string);
        $string_trimmed = trim($string);
        $string_after = str_replace("WORLD" , "KUSH" ,
"HELLO WORLD!");
        echo "<input type = 'text' name = 'org' value =
'ORIGINAL STRING IS: $string' readonly>";
        if($string == $string_low){
```



```
        echo "<input type = 'text' name = 'lower'
value = 'THE STRING IS IN LOWERCASE.' readonly>";
    }else{
        echo "<input type = 'text' name = 'lower_not'
value = 'THE STRING IS NOT IN LOWERCASE.' readonly>";
    }
    echo "<input type = 'text' name = 'reversed' value
= 'THE REVERSED STRING IS: $string_rev' readonly>";
    echo "<input type = 'text' name = 'trimmed' value
= 'STRING AFTER REMOVING WHITESPACES IS: $string_trimmed'
readonly>";
    echo "<input type = 'text' name = 'before' value =
'String to replace: $string_before' readonly>";
    echo "<input type = 'text' name = 'after' value =
'REPLACED STRING IS: $string_after' readonly>";

}
?>
</html>
```

OUTPUT: -

The screenshot displays a web application interface with a dark background. At the top, there is a light gray rounded rectangle containing the text "ENTER A STRING" in blue. Below this, a white text input field contains "Kush Amit Shah", and next to it is a blue "SUBMIT" button. Below the input area, there are six white rectangular boxes, each displaying a string manipulation result: "ORIGINAL STRING IS: Kush Amit Shah", "THE STRING IS NOT IN LOWERCASE.", "THE REVERSED STRING IS: hahS timA hsuK", "STRING AFTER REMOVING WHITESPACES IS: Kush Amit Shah", "STRING TO REPLACE: HELLO WORLD!", and "REPLACED STRING IS: HELLO KUSH!".

Aim: - C] Write scripts using math functions:

- a. to generate a random number between the given range.**
- b. to display the binary, octal and hexadecimal of a given decimal number.**
- c. to display the sin, cos and tan of the given angle.**

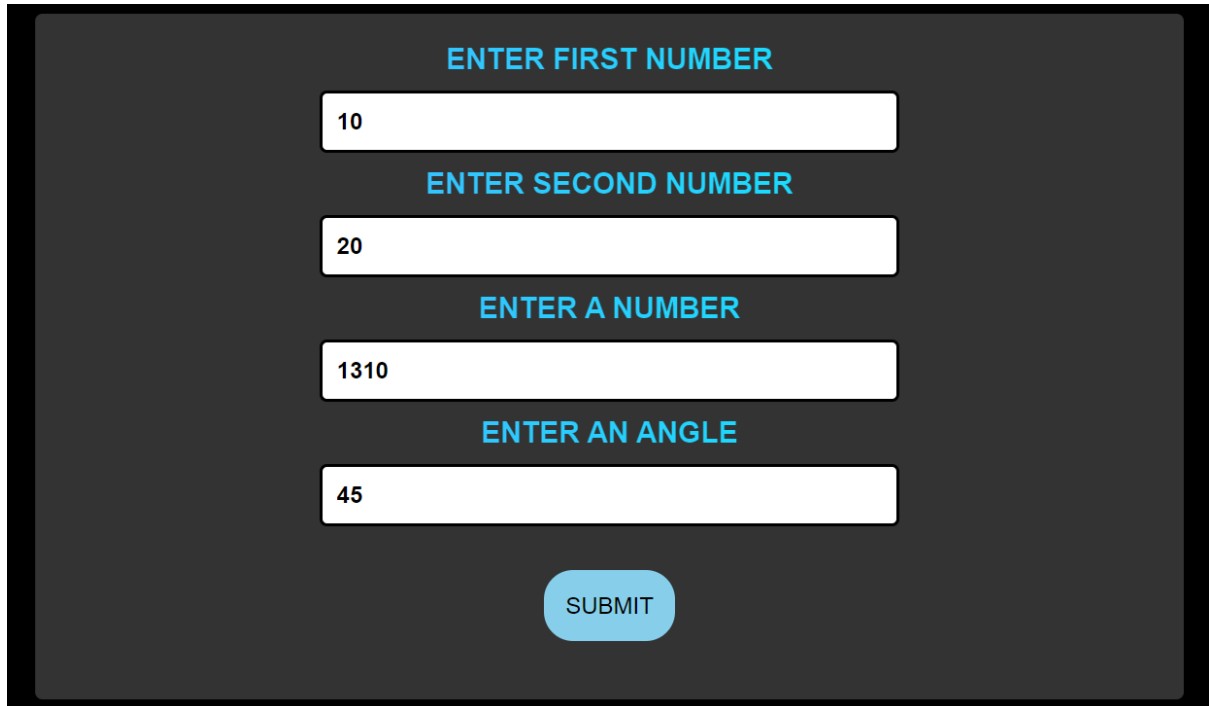
CODE: -

```

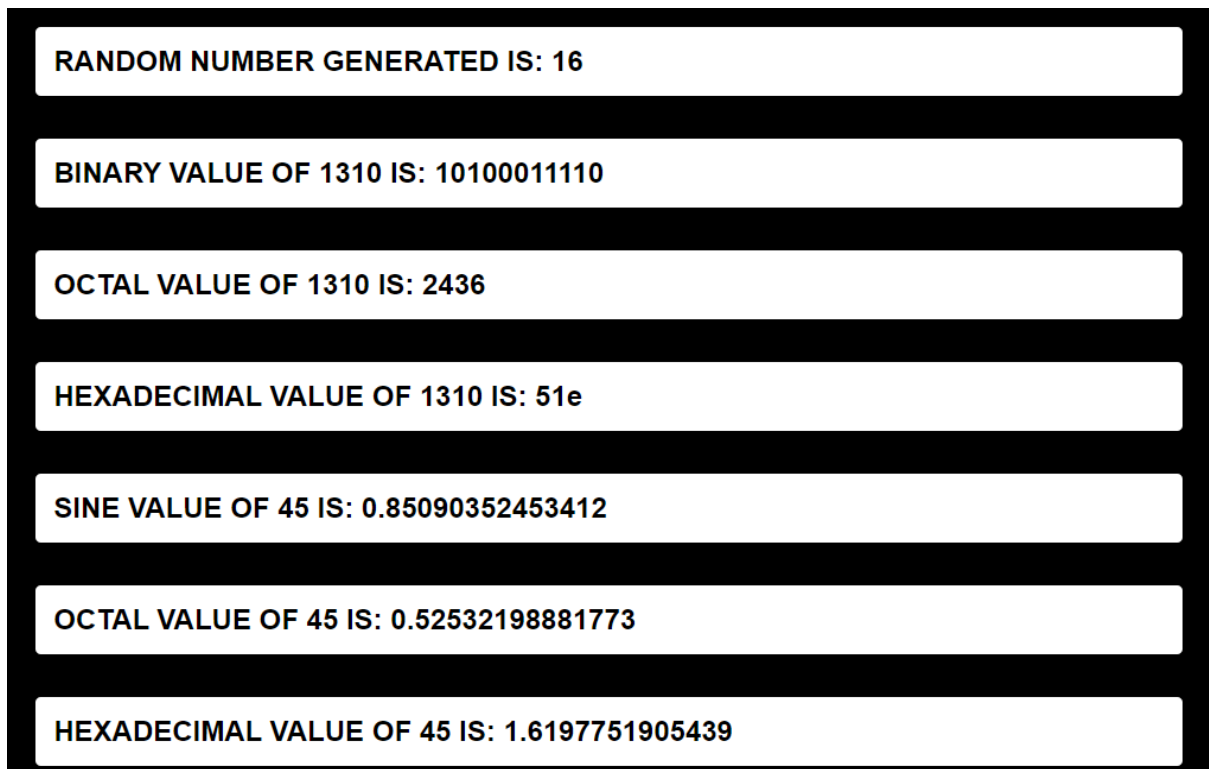
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>MATH FUNCTIONS</title>
</head>
<body>
<form action = "" method = "POST">
    <label for = "n1">ENTER FIRST NUMBER</label>
    <input type = "text" name = "n1" placeholder =
"START RANGE" required>
    <label for = "n2">ENTER SECOND NUMBER</label>
    <input type = "text" name = "n2" placeholder =
"END RANGE" required>
    <label for = "num">ENTER A NUMBER</label>
    <input type = "text" name = "num" placeholder =
"DECIMAL NUMBER TO CONVERT" required>
    <label for = "angle">ENTER AN ANGLE</label>
    <input type = "text" name = "angle" placeholder =
"EG: 80" required>
    <br>
    <button type="submit" name="SUB">SUBMIT</button>
</form>
</body>
<?php
    if (isset($_POST['SUB'])) {
        $num1 = $_POST['n1']; $num2 = $_POST['n2']; $num
= $_POST['num'];
        $angle = $_POST['angle'];
        $random_num = rand($num1, $num2);
        echo "<input type = 'text' name = 'random'
value = 'RANDOM NUMBER GENERATED IS: $random_num'
readonly>";
    }

```

```
        $num_bin = decbin($num); $num_oct =  
decoct($num); $num_hex = dechex($num);  
        echo "<input type = 'text' name = 'decbin'  
value = 'BINARY VALUE OF $num IS: $num_bin' readonly>";  
        echo "<input type = 'text' name = 'decoct'  
value = 'OCTAL VALUE OF $num IS: $num_oct' readonly>";  
        echo "<input type = 'text' name = 'dechex'  
value = 'HEXADECIMAL VALUE OF $num IS: $num_hex'  
readonly>";  
  
        $angle_sin = sin($angle); $angle_cos =  
cos($angle); $angle_tan = tan($angle);  
        echo "<input type = 'text' name = 'sin' value  
= 'SINE VALUE OF $angle IS: $angle_sin' readonly>";  
        echo "<input type = 'text' name = 'cos' value  
= 'OCTAL VALUE OF $angle IS: $angle_cos' readonly>";  
        echo "<input type = 'text' name = 'tan' value  
= 'HEXADECIMAL VALUE OF $angle IS: $angle_tan' readonly>";  
    }  
?>  
</html>
```

OUTPUT: -

A screenshot of a web form on a dark gray background. It contains five input fields, each with a label above it in cyan text. The labels are "ENTER FIRST NUMBER", "ENTER SECOND NUMBER", "ENTER A NUMBER", and "ENTER AN ANGLE". The input fields contain the values "10", "20", "1310", and "45" respectively. Below the input fields is a cyan button with the text "SUBMIT".



A screenshot of a web form showing the results of conversions. It consists of seven white rectangular boxes, each containing a line of text. The text in the boxes is as follows:

- RANDOM NUMBER GENERATED IS: 16
- BINARY VALUE OF 1310 IS: 10100011110
- OCTAL VALUE OF 1310 IS: 2436
- HEXADECIMAL VALUE OF 1310 IS: 51e
- SINE VALUE OF 45 IS: 0.85090352453412
- OCTAL VALUE OF 45 IS: 0.52532198881773
- HEXADECIMAL VALUE OF 45 IS: 1.6197751905439

Aim: - D] Write a script to display the current date and time in different formats.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>DATE FUNCTIONS</title>
</head>
<body>
<div class="container">
    <h2>DATE TIME FUNCTIONS</h2>
</div>
<?php
    $time = new DateTime();
    $format_1 = $time->format('Y-m-d H:i:s');
    $format_2 = $time->format('d-m-y');
    $format_3 = $time->format('l, F j, Y g:i A');
    $format_4 = $time->format('Y-m-d');
    $format_5 = $time->format('d-D-M-y');
    echo "<input type = 'text' value = 'DEFAULT FORMAT:
$format_1' readonly>";
    echo "<input type = 'text' value = 'CUSTOM FORMAT 1:
$format_2' readonly>";
    echo "<input type = 'text' value = 'CUSTOM FORMAT 2:
$format_3' readonly>";
    echo "<input type = 'text' value = 'CUSTOM FORMAT 3:
$format_4' readonly>";
    echo "<input type = 'text' value = 'CUSTOM FORMAT 4:
$format_5' readonly>";
?>
</body>
</html>
```

OUTPUT: -

DATE TIME FUNCTIONS

DEFAULT FORMAT: 2024-03-21 15:25:17

CUSTOM FORMAT 1: 21-03-24

CUSTOM FORMAT 2: Thursday, March 21, 2024 3:25 PM

CUSTOM FORMAT 3: 2024-03-21

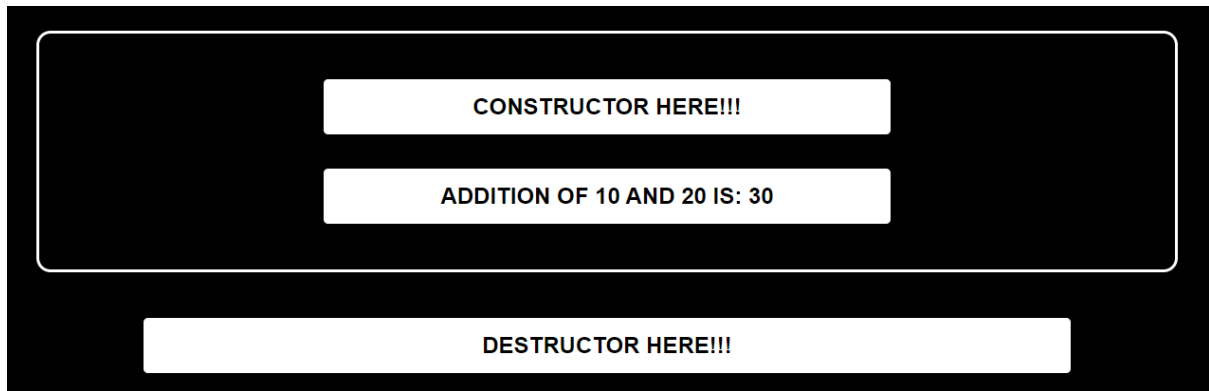
CUSTOM FORMAT 4: 21-Thu-Mar-24

PRACTICAL - 07**Aim: - A] Write a script to:**

- a. Define a class with constructor and destructor.**
- b. Create an object of a class and access its public properties and methods.**

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>CONSTRUCTOR AND DESTRUCTOR</title>
</head>
<body>
<div class="container">
    <?php
        class demo{
            function add(){
                $num1 = 10;
                $num2 = 20;
                $ans = $num1 + $num2;
                echo "<input type = 'text' value =
'ADDITION OF $num1 AND $num2 IS: $ans' readonly>";
            }
            function __construct(){
                echo "<input type = 'text' value =
'CONSTRUCTOR HERE!!!' readonly>";
            }
            function __destruct(){
                echo "<input type = 'text' value =
'DESTRUCTOR HERE!!!' readonly>";
            }
        }
        $ob = new demo();
        $ob->add();
    ?>
</div>
</body>
</html>
```

OUTPUT: -

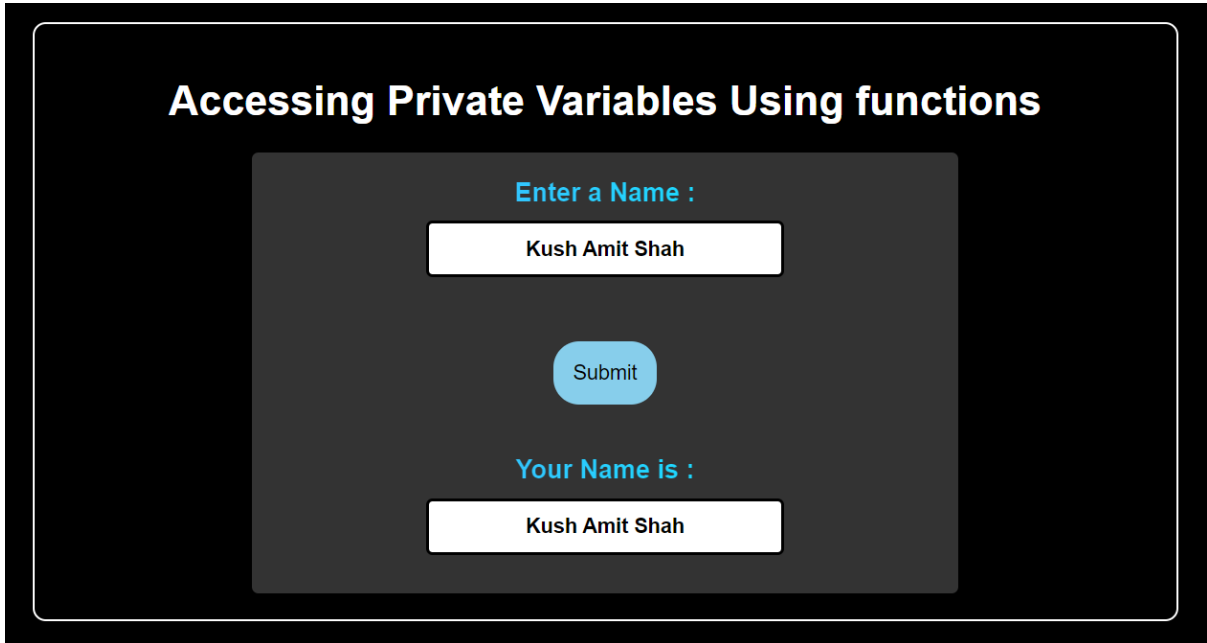
Aim: - B] Write a script that uses the set attribute and get attribute methods to access a class's private attributes of a class.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>SET AND GET ATTRIBUTES</title>
</head>
<body>
    <?php
        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            class privateattribute{
                private $user;
                function setUser($user){
                    $this->user = $user;
                }
                function getUser(){
                    echo $this->user;
                }
            }
            $use = new privateattribute();
            $user = $_POST["userName"];
            $use->setUser($user);
        }
    ?>
<div class="container">
    <h1>Accessing Private Variables Using functions</h1>
    <form action="" method="post">
        <div class="form-group">
            <label for="ans">Enter a Name :</label>
            <input type="text" name="userName"
required><br><br>
            <button type="submit">Submit</button><br><br>
            <label for="ans">Your Name is :</label>
            <input type="text" name="displayUser"
value="<?php $use->getUser(); ?>" readonly>
        </div>
    </form>
```

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

OUTPUT: -



Accessing Private Variables Using functions

Enter a Name :

Kush Amit Shah

Submit

Your Name is :

Kush Amit Shah

Aim: - C] Write a script to demonstrate single inheritance.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>SINGLE INHERITANCE</title>
</head>
<body>
<form action = "" method = "POST">
<div class="container">
    <input type = "text" name = "msg" value = "SINGLE
INHERITANCE" readonly></div>
<div class="container">
    <label for = "num1">ENTER FIRST NUMBER</label>
    <input type = "number" name = "num1" required>
    <label for = "num2">ENTER SECOND NUMBER</label>
    <input type = "number" name = "num2" required><br>

    <button type = "submit" name =
"submit">SUBMIT</button><br>
<?php
    if(isset($_POST['submit'])){
        class base{
            public $num1;
            public $num2;
            function __construct($num1, $num2){
                $this->num1 = $num1;
                $this->num2 = $num2;
            }
            function add(){
                $ans1 = $this->num1 + $this->num2;
                echo "ADDITION: " . $ans1;
            }
        }
        class child extends base{
            function sub(){
                $ans2 = $this->num1 - $this->num2;
                echo "SUBTRACTION: " . $ans2;}}
```

```
        $ob = new child($_POST['num1'],  
$_POST['num2']);  
    }  
    ?>  
        <input type = 'text' value = '<?php $ob-  
>add();?>' readonly>  
        <input type = 'text' value = '<?php $ob-  
>sub();?>' readonly>  
</div>  
</form>  
</body>  
</html>
```

OUTPUT: -

SINGLE INHERITANCE

ENTER FIRST NUMBER

13

ENTER SECOND NUMBER

10

SUBMIT

ADDITION: 23

SUBTRACTION: 3

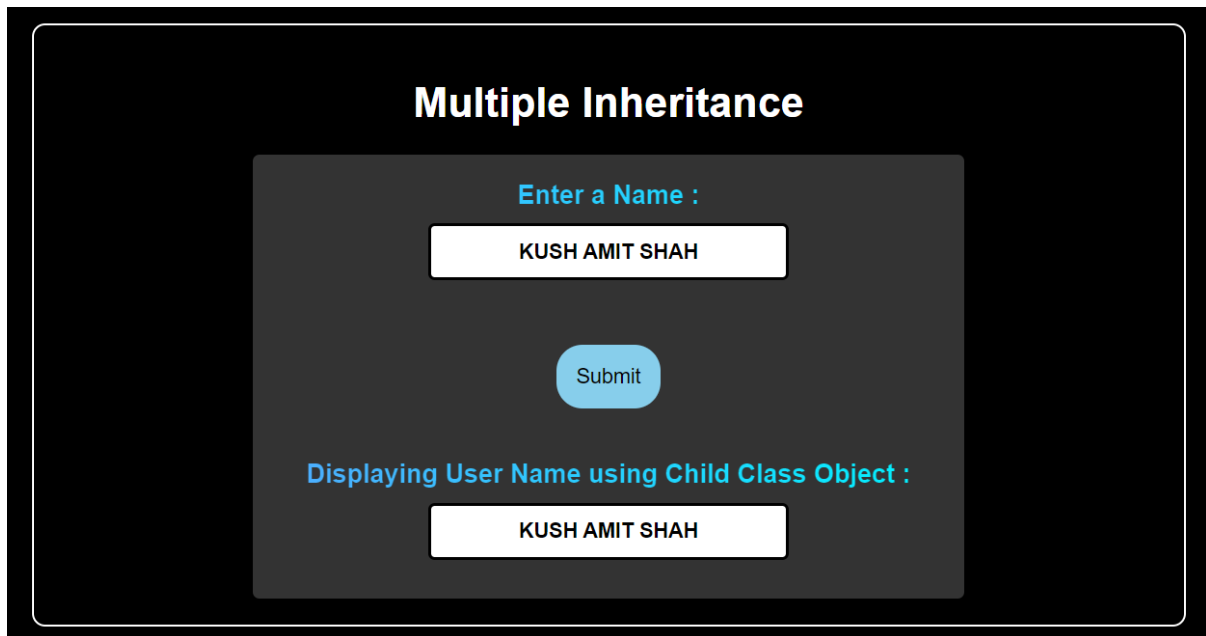
Aim: - D] Write a script to demonstrate multiple inheritance.

CODE: -

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>MULTIPLE INHERITANCE</title>
<body>
    <?php
        if ($_SERVER["REQUEST_METHOD"] == "POST") {
            interface A{
                function setUser($user);
            }
            interface B{
                function getUser();
            }
            class Userinfo implements A,B{
                public $user;
                function setUser($user){
                    $this->user = $user;
                }
                function getUser(){
                    echo $this->user;
                }
            }
            $use = new Userinfo();
            $user = $_POST["userName"];
            $use->setUser($user);
        }
    ?>
<div class="container">
    <h1>Multiple Inheritance</h1>
    <form action="#" method="post">
        <div class="form-group">
            <label for="ans">Enter a Name :</label>
            <input type="text" name="userName"
required><br><br>
            <button
type="submit">Submit</button><br><br>
```

```
<label for="ans">Displaying User Name  
using Child Class Object :</label>  
    <input type="text" name="displayUser"  
value="<?php $use->getUser(); ?>" readonly>  
    </div>  
</form>  
</body>  
</html>
```

OUTPUT: -



The screenshot shows a web application interface with a dark background. At the top, the title "Multiple Inheritance" is displayed in white. Below the title, there is a form area with a dark gray background. Inside this area, the text "Enter a Name :" is shown in blue. Below this text is a white text input field containing the text "KUSH AMIT SHAH". Below the input field is a blue rounded button labeled "Submit". Below the button, the text "Displaying User Name using Child Class Object :" is shown in blue. Below this text is another white text input field containing the text "KUSH AMIT SHAH".