

EXPNO: 01

DATE:04-01-2024

ODD OR EVEN NUMBER

AIM:

The aim of this program is to generate a php based odd or even number checker allowing users to input a numeric value and determining whether it is odd or even and displays the result

ALGORITHM:

STEP1: start the program

STEP2: HTML Structure:

- a. Open doctype and HTML tag.
- b. Inside head tag:
 - . Set title to "odd or even checker".
- c. Inside body tag
- d. Display title "Odd or Even number Checker" using <h1> tag.
- i. Create form with POST method:
 - Text input to enter a number.
 - Submit button with value "Check".

STEP3: Check if form is submitted:

- Retrieve entered number.
- Check if entered value is numeric.
- If numeric:
 - Check if number is even.
 - Display result.
- If not numeric:
 - Display error message.

STEP4: Close center ,body and html tags

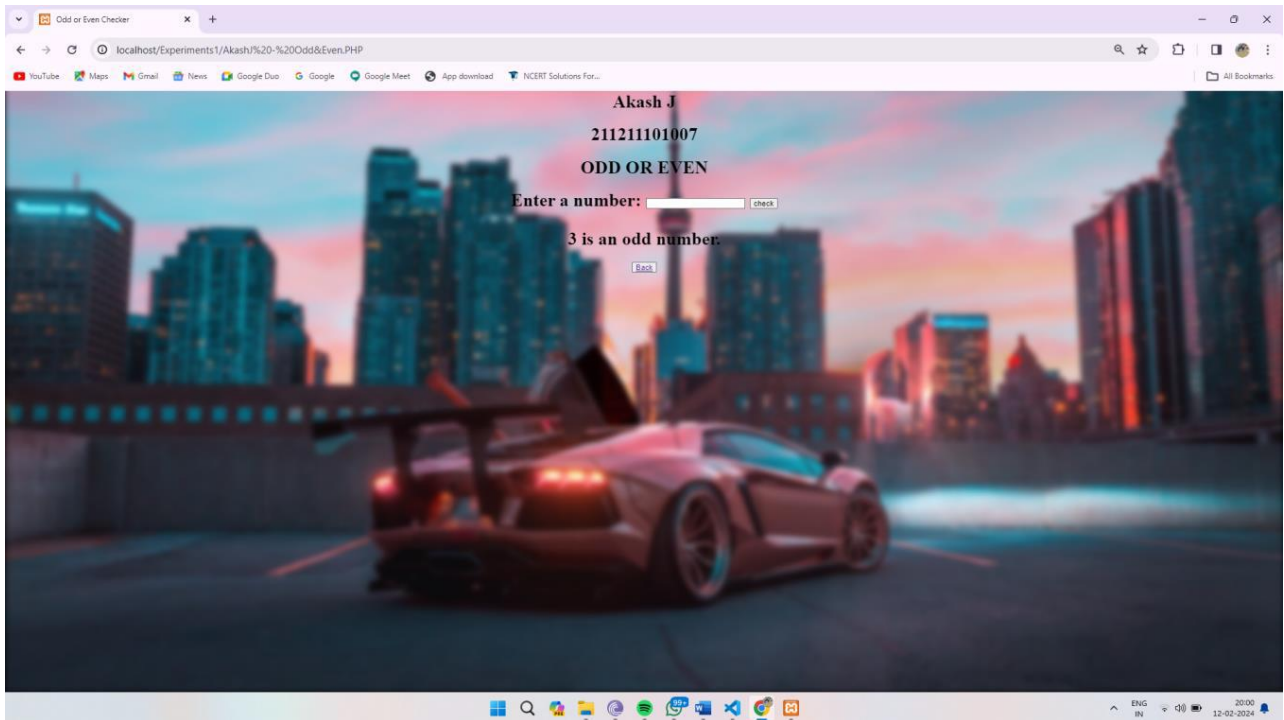
STEP5: End the program

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
<title>Odd or Even Checker </title>
<link rel="stylesheet" href="php1.css">
</head>
<style>
body{
color:white;
}
</style>
<body class="total">
<video class="videobackground" src="1.mp4" autoplay muted loop plays-inline ></video>
<p><h1><center> Akash J - 211211101007 <br><br> Odd & Even Number Checker </center></h1></p>
<form method="post" action="">
<center><h1>Enter a Number:</h1>
<input type="text" name="number" required>
<input type="submit" value="check"></H1>
</center>
</form>
<?php
if($_SERVER["REQUEST_METHOD"]=="POST") {
$number=$_POST["number"];
if(is_numeric($number)){
if($number%2==0){
echo"<p><center><h3> $number is an even number.</h3></p></center>";
}
else {
echo"<p><center><h3>$number is an odd number.</h3></p></center>";
}
}
else{
echo"<p><center><h3>Please enter a valid numeric value.</h3></p></center>";
}
}
?>
```

```
<center>  
<button class="backbtn" onclick="history.back()"> Back to Home </button>  
</center>  
</body>  
</html>
```

OUTPUT :



RESULT:

Thus the program to develop and execute odd or even number checker is verified and executed successfully.

SWITCH-CASE STATEMENT

AIM:

The aim is to create a web page that allows users to perform various arithmetic operations on two input numbers. The operations include addition, subtraction, division, multiplication, modulus, and finding the square root of the first number

ALGORITHM:

STEP1: start the program

STEP2: Define a php function 'perform arithmetic operation(\$num1,\$num2,\$operator)' to perform arithmetic operations

- a. Take three parameters:\$num1,\$num2 and \$operator
- b. Use a switch statement on \$operator to perform the respective arithmetic operation
 - for addition('+'):return '\$num1+\$num2'
 - for subtraction('-'):return '\$num1-\$num2'
 - for multiplication('*'):return '\$num1*\$num2'
 - for division('/'):check if '\$num2' is not zero,if not '\$num1/\$num2',otherwise return "division by zero is not allowed"
 - default case: return "invalid operator!"

STEP3: Check if the form is submitted (POST method):

- a. Get the values of num1, num2, and operator from \$_POST super global.
- b. Call the perform Arithmetic Operation(\$num1, \$num2, \$operator) function and store the result in \$result.

STEP4: Display the HTML form to the user:

- a. The form has:
 - Two input fields for num1 and num2.
 - A dropdown menu to select an arithmetic operation.
 - A submit button to trigger the calculation.

STEP5: Process the form submission and display the result:

- a. Check if \$_SERVER["REQUEST_METHOD"] is POST.
- b. Retrieve num1, num2, and operation values from \$_POST.
- c. Check if both num1 and num2 are numeric using is_numeric().
- d. Use a switch statement on operation to perform the respective arithmetic operation:

- For "add": $\$result = \$a + \$b$;
- For "subtract": $\$result = \$a - \$b$;
- For "divide": check if $\$b$ is not zero and compute $\$result = \$a / \$b$;, otherwise display an error message.
- For "multiply": $\$result = \$a * \$b$;
- For "modulus": $\$result = \$a \% \$b$;
- For "square root": $\$result = \$a * \$a$;
- e. Display the result or error message to the user.

STEP6: End the program

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
<title>Arithmetic Operations </title>
<link rel="stylesheet" href="php2.css">
</head>
<style>
body {
  color: white;
}
</style>
<body>
<video class="videobackground" src="gojo2.mp4" autoplay muted loop plays-inline ></video>
<p><center><h1> Akash J- 211211101007 <br><br> Arithmetic Operations </h1></center></p>
<form method="POST" action="">
<center>
Enter the first Number:</h4>
<input type="text" name="num1" required> <br><br><br>
Enter the second Number:
<input type="text" name="num2" required> <br><br><br>
Select an Operation:
<select name="operation">
<option value="add">Addition</option>
<option value="subtract">Subtraction</option>
<option value="multiply">Multiplication</option>
value="divide">Division</option>
<option value="modulus">Modulus</option>
</select>
<br><br>
<input type="submit" value="Calculate">
</center>
</form>
<?php

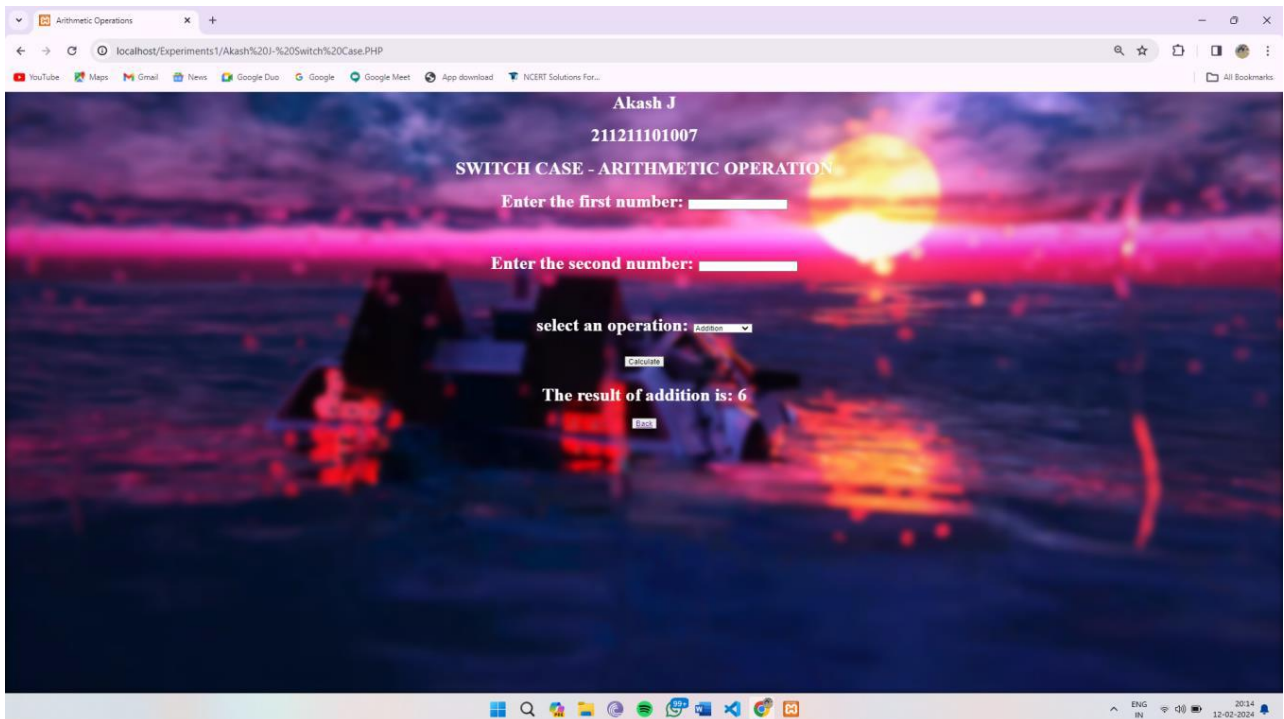
    error_reporting(E_ALL);
    ini_set('display_errors', 1);
    if($_SERVER["REQUEST_METHOD"] == "POST") {
```

```

$a=$_POST["num1"];
$b=$_POST["num2"];
$operation=$_POST["operation"];
if(is_numeric($a) && is_numeric($b)){
switch($operation){
case "add":
$result = $a + $b;
echo "<center><p><h1>The result of addition is:$result</h1></center></p>";
break;
case "subtract":
$result = $a - $b;
echo"<p><center>The result of subtraction is: $result</center></p>";
break;
case "multiply":
$result = $a * $b;
echo"<p><center>The result of Multiplication is: $result</center></p>";
break;
case "divide":
if($b != 0){
$result=$a / $b;
echo"<p><center>The result of Division is: $result</center></p>";
}
else {
echo"<p><center>Error: Divison by zero is not allowed.</center></p>";
}
break;
case "modulus":
$result=$a % $b;
echo"<p><center>The result of Modulus is: $result</center></p>";
break;
default:
echo"<p><center><h1> Please select a valid operation.</h1> </center></p>";
}
}
}
?>
<center><button class="backbtn" onclick="history.back()"> Back to Home </button> </center>
</body>
</html>

```


OUTPUT:



RESULT:

Thus the program to create a simple web based calculator for performing arithmetic operations are verified and executed successfully.

EXPNO: 03

DATE:11-01-2024

FIBONACCI SERIES GENERATOR

AIM:

The aim of the provided code is to create a web-based factorial calculator. Users can input a number, and upon clicking the "Calculate" button, the application will display a table showing the factorial of each number from 1 up to the entered number

ALGORITHM:

STEP1: Start the program

STEP2: Display Page Content:

- Display the web page with the following components
- Input field to enter a number
- Button to trigger the factorial calculation
- Table to display the factorial results
- Button to return to the previous page

STEP3:Function: calculate Factorial():

a. Input:

- Retrieve the entered number from the input field.
- Initialize a factorial variable to 1.

b.Clear Previous Entries:

- Clear the content of the factorial table.

c. Calculate Factorial:

- Loop from 1 to the entered number:
- Update factorial by multiplying it with the loop index i.
- Create a new table row.
- Populate the row with the loop index i and the updated factorial value.

STEP4: Display Factorial Results:

- Display the calculated factorial values in the factorial

STEP5: End the program

PROGRAM:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <style>

    table {
      border-collapse: collapse;
      width: 20%;
      margin: 20px auto;
      background-color: transparent;
    }
    th, td {
      border: 2px solid white;
      text-align: center;
      padding: 8px;
      color: white;
      font-size: 30px;
    }
    th {
      background-color: transparent;
    }

    body {
      color: white;
    }
    .text {
      font-size: 30px;
      font-weight: 100px;
    }

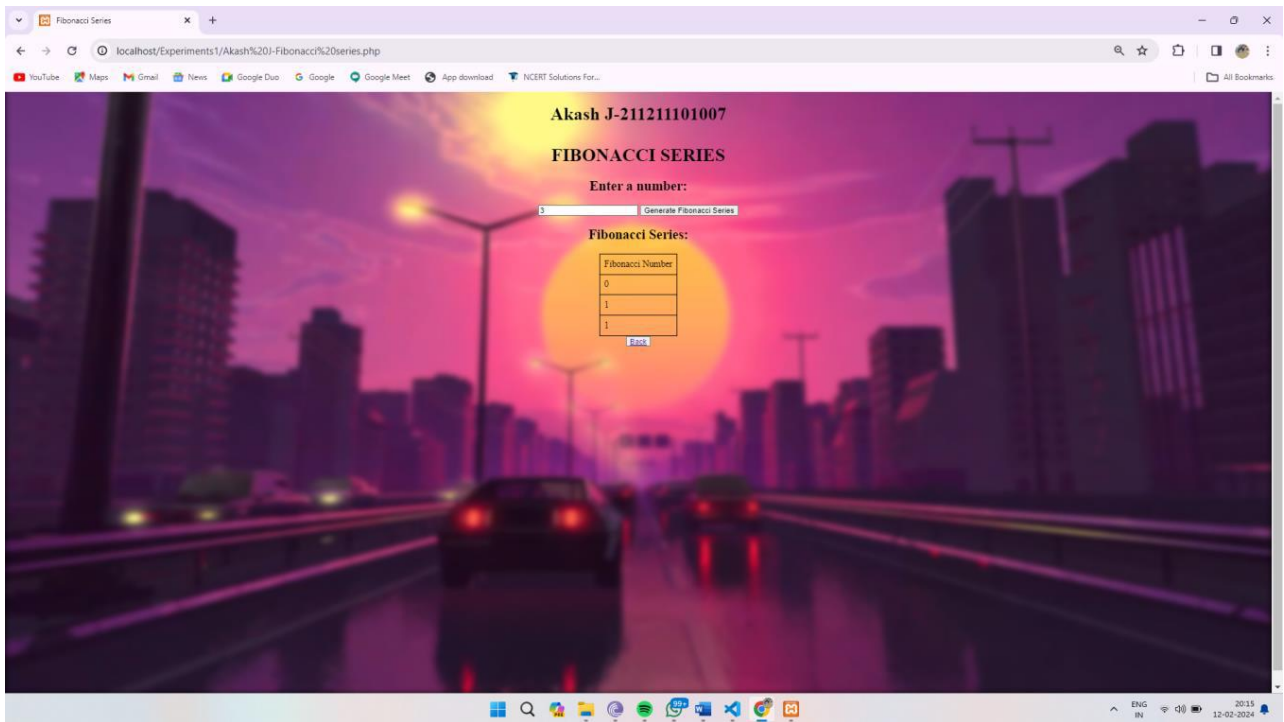
  </style>
  <center><h1>Akash J - 211211101007</h1>
  <title>Fibonacci Series</title></center>
  <link rel="stylesheet" href="php3.css">
</head>
<body>
<video class="videobackground" src="gojo3.mp4" autoplay muted loop plays-inline ></video>
```

```

<form><center>
<label class="text">Enter the fibonacci number</label>
<input type="number" name="Fibonacci">
<input type="submit" value="submit" name="submitForm"></br></br>
</form></center>
</body>
</html>
<?php
if(isset($_GET["submitForm"]))
{
    $num=$_GET['Fibonacci'];
    $a=0;
    $b=1;
    echo "<center><h2>Fibonacci series for $num elements:</h2></center>";
    echo "<center><table>";
    echo "<tr><th>Index</th><th>Fibonacci Number</th></tr>";
    echo "<tr><td>1</td><td>$a</td></tr>";
    echo "<tr><td>2</td><td>$b</td></tr>";
    $i=2;
    while($i<$num)
    {
        $next=$a+$b;
        echo "<tr><td>".($i+1)."</td><td>$next</td></tr>";
        $a=$b;
        $b=$next;
        $i++;
    }
    echo "</table></center>";
}
?>
<center><button class="backbtn" onclick="history.back()"> Back to Home </button></center>
</body>
</html>

```

OUTPUT:



RESULT:

Thus the program to find Fibonacci series of any number has been executed and verified successfully.

EXPNO: 04

DATE:18-01-2024

FACTORIAL OF A NUMBER

AIM:

the aim of the program is to find the factorial of a number

ALGORITHM:

STEP1: Start the program

STEP2: Initialization:

- Start PHP session..

- Initialize session variables for quantity, manufacture cost, and damaged products.

STEP3: Handle Form Submission:

- Check if "save" button is clicked:

- Store entered values in session variables.

STEP4: Display Page Content:

- Display Title, Name, Input fields, and buttons.

STEP5: Calculate Financial Metrics:

a. Input:

- Retrieve values from the form.

b. Create Product Class:

- Define class Product with attributes and methods.

c. Calculate Values:

- Calculate financial metrics.

STEP6: Display Calculated Values:

- Display financial metrics and damaged product details.

STEP7: Handle History Button Click:

- Check if "History" button is clicked:

- Display previous entered values from session.

STEP8: End the program.

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
  <title>Factorial Program</title>
  <link rel="stylesheet" href="php4.css">
<style>

  .num{
    color: white;
  }
  table {
    border-collapse: collapse;
    width: 30%;
    margin: auto;
    background-color:transparent;
    color:white

  }
  th, td {
    border: 1px solid black;
    padding: 8px;
    text-align: center;

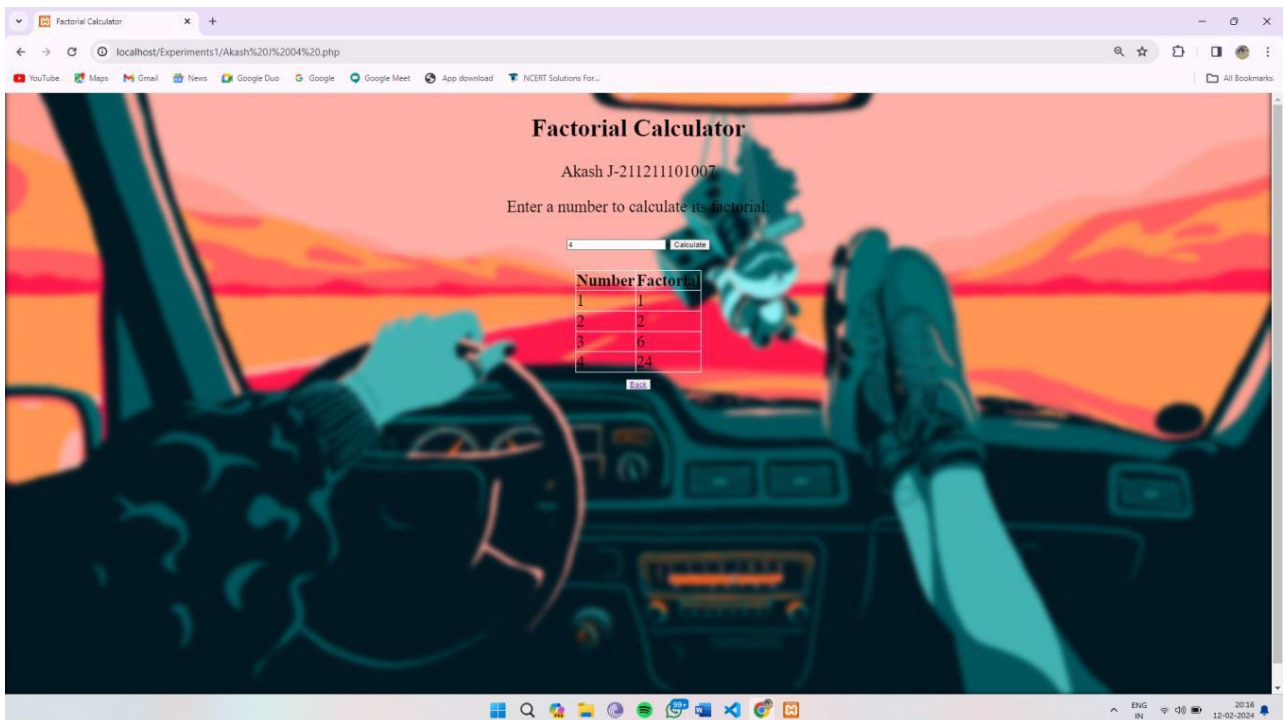
  }
  body{
    color:white;
  }
</style>
</head>
<body>
<center>
<div class="head">
  <h1>Akash J- 211211101007</h1>
  <h2>Factorial number</h2>
</div>
  <video src="luffy.mp4" autoplay muted loop plays-inline></video>
<form method="POST">
  <label class="num">Enter a number</label>
  <input type="number" name="Factorial">
```

```

        <input type="submit" value="Submit" name="submitForm"><br><br>
</form>
<?php
    // example to demonstrate factorial of a number using form
    if(isset($_POST["submitForm"])) {
        $input = $_POST["Factorial"];
        if(is_numeric($input) && $input >= 0) {
            echo '<table>';
            echo '<tr><th>Number</th><th>Factorial</th></tr>';
            for($i = $input; $i >= 1; $i--) {
                $fact = 1;
                for($j = $i; $j >= 1; $j--) {
                    $fact *= $j;
                }
                echo '<tr><td>' . $i . '</td><td>' . $fact . '</td></tr>';
            }
            echo '</table>';
        } else {
            echo '<br> Please enter a valid non-negative number.';
        }
    }
?>
<button class="backbtn" onclick="history.back()"> Back to Home</button>
</center>
</body>
</html>

```


OUTPUT:



RESULT:

Thus the execution of finding factorial of a number using php has been done and the output has been verified successfully.

CLASS AND OBJECTS**AIM:**

The aim of this program is to create an inventory system using classes and objects

ALGORITHM:

STEP1: Start the program

STEP2: Accept input for product quantity, manufacturing cost per product, and quantity of damaged products.

STEP3: Create a class named Product with private attributes quantity, manufacture Cost, and damage.

STEP4: Define a constructor method within the Product class to initialize the attributes.

STEP5: Define methods within the Product class to calculate:

- get Total Cost(): Calculates the total manufacturing cost.
- get Avg Benefit(): Calculates the average benefit.
- get Incl CGST(): Calculates the inclusive CGST.
- get Incl SGST(): Calculates the inclusive SGST.
- get Mar Benefit(): Calculates the marginal benefit.
- get Damaged product(): Returns the quantity of damaged products.

STEP6: Instantiate an object of the Product class with the provided input values.

STEP7: Calculate the various costs and benefits using the methods defined in the Product class.

STEP8: Display the results, including total manufacturing cost, average benefit, GST costs, marginal benefit, actual dealer price, total deal cost, selling price per product, total selling cost, and selling price for damaged products.

STEP9: End the program

PROGRAM:

```
<!DOCTYPE html>

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> Inventory System</title>
</head>
<style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;

}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}

.backbtn{
  position:center;
}
body{
  background-image:url('<?php echo $img;?>');
  background-size:cover;
  background-position:fixed;
  background-repeat:no-repeat;
}
```

```

h1{color:white}
p{color:white}
body{color:white}
</style>
<body>
<video autoplay loop muted plays-inline class="back-video"
    src="kid.mp4">
    </video>
    <center><p><center><h1> Akash J- 211211101007 <br> Inventory System</h1></center></p>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>"
    Enter the product quantity <input type="number" name="num1"> <br><br>
    Enter the manufacture cost per product <input type="number" name="num2"> <br> <br>
        Enter the damaged product <input type="number" name="num3"> <input type="submit"
value="Submit">
        <br><br>
</form>
    <?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $quantity = $_POST["num1"];
    $manufactureCost = $_POST["num2"];
    $dpq = $_POST["num3"];

class Product {
    private $quantity;
    private $manufactureCost;
    private $dpq;

    public function __construct($quantity, $manufactureCost,$dpq) {
        $this->quantity = $quantity;
        $this->manufactureCost = $manufactureCost;
        $this->damage = $dpq;
    }
    public function getTotalCost() {
        return $this->quantity * $this->manufactureCost;
    }

        public function getAvgBenefit() {
            return (($this->manufactureCost)*0.10);
        }
        public function getInclCGST() {
            return ($this->manufactureCost)*0.09;
        }

```

```

        public function getInclSGST() {
            return ($this->manufactureCost)*0.09;
        }

        public function getMarBenefit() {
            return ($this->manufactureCost)*0.20;
        }

        public function getDamagedproduct(){
            return($this->damage);
        }
    }

    $product = new Product($quantity, $manufactureCost,$dpq);
    $totalCost = $product->getTotalCost();
    $avgbenefit = $product->getAvgBenefit();
    $cgst = $product->getInclCGST();
    $sgst = $product->getInclSGST();
    $margin = $product->getMarBenefit();
    $dam = $product->getDamagedproduct();
    echo "MANUFACTURER PHASE<br>";
    echo "Total manufacture Cost: ₹" . $totalCost . "<br>";
    echo "Average Benefit : ₹" . $avgbenefit . "<br>";
    echo " CGST : ₹" . $cgst . "<br>";
    echo " SGST : ₹" . $sgst . "<br>";
    echo "Total GST Cost: ₹" .intval($sgst + $cgst). "<br>";
    echo "Marginal Benefit: ₹" . $margin . "<br>";
    echo " Actual Dealer Price: ₹" .intval($manufactureCost + $cgst + $sgst + $margin) . "<br>";
    echo " Total Deal Cost: ₹" .intval($manufactureCost + $cgst + $sgst + $margin)*$quantity . "<br>";
    echo "<br>DEALER PHASE<br>";
    echo "Selling price per product: ₹" .intval(((( $manufactureCost + $cgst + $sgst + $margin)*0.40) +
    ($manufactureCost + $cgst + $sgst + $margin)) . "<br>";
    echo "Total Selling Cost: ₹" .intval(((( $manufactureCost + $cgst + $sgst + $margin)*0.40) +
    ($manufactureCost + $cgst + $sgst + $margin)) * $quantity . "<br>";
    echo "<br>DAMAGED PRODUCT PHASE<br>";
    echo "Damaged product quantity : " . $dam . "<br>";
    echo "Each damaged product selling price : ₹" .intval(((( $manufactureCost + $cgst + $sgst +
    $margin)*0.40) + ($manufactureCost + $cgst + $sgst + $margin)/2). "<br>";
    echo "Total damaged product selling price : ₹" .intval(((( $manufactureCost + $cgst + $sgst +
    $margin)*0.40) + ($manufactureCost + $cgst + $sgst + $margin)/2)*$dam) . "<br>.<br>";
    }
?>

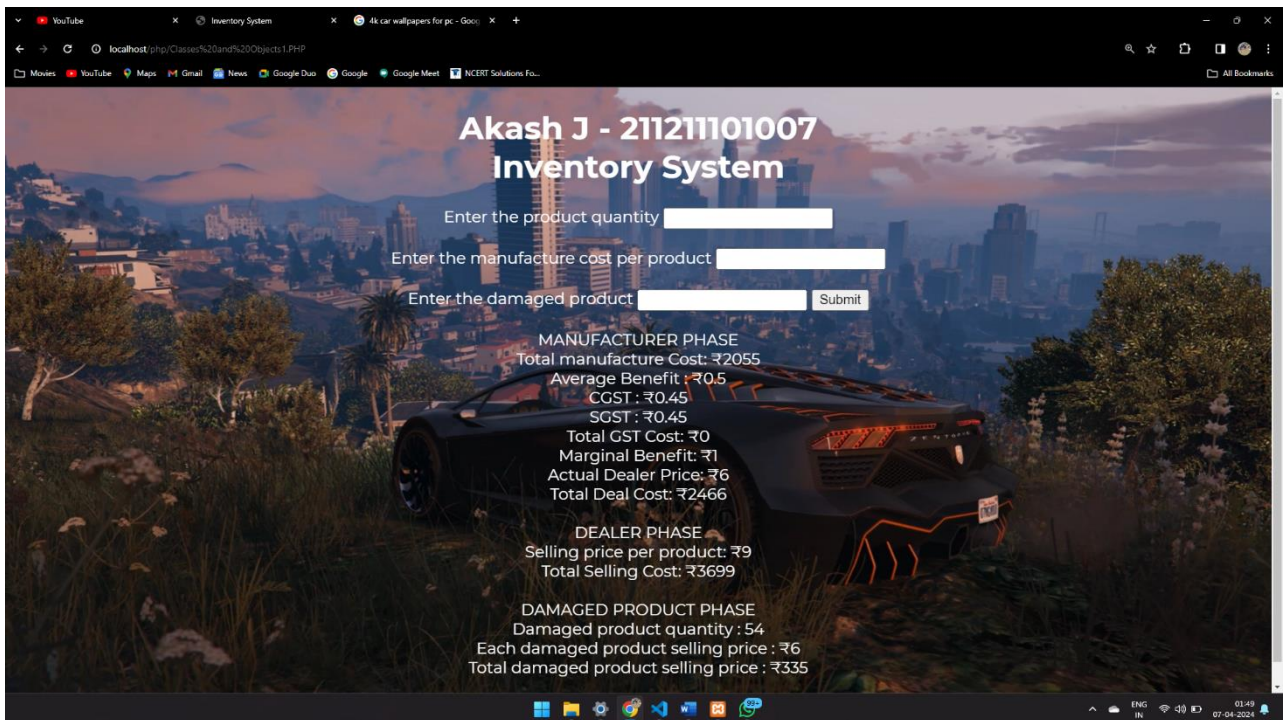
```

</center>

</body>

</html>

OUTPUT:



RESULT:

Thus execution of classes and objects in php has been done and output was verified successfully.

INHERITANCE

AIM:

The aim of inheritance in PHP is to promote code reusability and organization by allowing a subclass (child class) to inherit the characteristics (properties and methods) of a superclass (parent class)

ALGORITHM:

STEP1: Start the program

STEP2: Define the parent class using the class keyword followed by the class name.

STEP3: Declare properties and methods within the parent class.

STEP4: Define the child class using the extends keyword followed by the name of the parent class.

STEP5: Add additional properties and methods to the child class, if needed.

STEP6: Access the inherited properties and methods from the parent class within the child class using the parent:: keyword.

STEP7: Instantiate objects of the child class and use them to access both inherited and child-specific functionality.

STEP8: End the program

PROGRAM:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> INHERITANCE</title>
</head>
<style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;

}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}

.backbtn{
  position:center;
}

body{
  background-image:url('<?php echo $img;?>');
  background-size:cover;
  background-position:fixed;
  background-repeat:no-repeat;
  color:white;
}
```

```

h1 {color:white}
p {color:white}
body {color:white}
</style>
<body>
<video autoplay loop muted plays-inline class="back-video"
    src="sukuna.mp4">
    </video>
<center>
<script>
    function showCheckboxes() {
        var radioValue = document.querySelector('input[name="option"]:checked').value;
        var checkboxesContainer = document.getElementById('checkboxes-container');
        checkboxesContainer.innerHTML = "";
        if (radioValue === 'BMW') {
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="BMW1"> G310RR</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="BMW2"> F850GS</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="BMW3"> S1000RR</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="BMW4"> R nineT</label><br>';
        } else if (radioValue === 'JAWA') {
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="JAWA1"> 350</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="JAWA2"> PERAK</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="JAWA3"> BOBBER 42</label><br>';
        }
        else if (radioValue === 'YAMAHA') {
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="YAMA1"> MT-15</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="YAMA2"> RX100</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="YAMA3"> RAY ZR</label><br>';
            checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="YAMA4"> R15 M</label><br>';
        }
    }
}

```

```

    }
    else if (radioValue === 'ROYAL ENFIELD') {
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="RE1"> GT650</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="RE2"> METEOR 350</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="RE3"> CLASSIC 350</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="RE4"> ELECTRA</label><br>';
    }
    else if (radioValue === 'HARLEY DAVIDSON') {
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="HD1"> SPORTSTER</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="HD2"> PAN AMERICA</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="HD3"> ROAD GLIDE</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]"
value="HD4"> NIGHSTER</label><br>';
    }
    else if (radioValue === 'INDIAN MOTORCYCLES') {
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]" value="IM1">
FTR</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]" value="IM2">
CHIEF DARK HORSE</label><br>';
        checkboxesContainer.innerHTML += '<label><input type="checkbox" name="checkbox[]" value="IM3">
PURSUIT</label><br>';
    }
}
}
</script>
<center><h1> Akash J - 211211101007 </h1><br>
<h2> INHERITANCE </h2></center>
<form method="post" action="">
<input type="radio" name="option" value="BMW" onclick="showCheckboxes()"> BMW
<input type="radio" name="option" value="JAWA" onclick="showCheckboxes()"> JAWA
<input type="radio" name="option" value="YAMAHA" onclick="showCheckboxes()"> YAMAHA
<input type="radio" name="option" value="ROYAL ENFIELD" onclick="showCheckboxes()"> ROYAL
ENFIELD
<input type="radio" name="option" value="HARLEY DAVIDSON" onclick="showCheckboxes()"> HARLEY
DAVIDSON

```

```
<input type="radio" name="option" value="INDIAN MOTORCYCLES" onclick="showCheckboxes()"> INDIAN  
MOTORCYCLES
```

```
<div id="checkboxes-container">
```

```
</div>
```

```
<br>
```

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

```
</form>
```

```
</center>
```

```
<center>
```

```
<br>
```

```
<?php
```

```
class Motorcycle {
```

```
    protected $price;
```

```
    public function __construct($price) {
```

```
        $this->price = $price;
```

```
    }
```

```
    public function getPrice() {
```

```
        return $this->price;
```

```
    }
```

```
}
```

```
class BMW extends Motorcycle {
```

```
    public function __construct($price) {
```

```
        parent::__construct($price);
```

```
    }
```

```
}
```

```
class Jawa extends Motorcycle {
```

```
    public function __construct($price) {
```

```
        parent::__construct($price);
```

```
    }
```

```
}
```

```
class Yamaha extends Motorcycle {
```

```
    public function __construct($price) {
```

```
        parent::__construct($price);
```

```
    }
```

```
}
```

```
class RoyalEnfield extends Motorcycle {
```

```
    public function __construct($price) {
```

```
        parent::__construct($price);
```

```
    }
```

```
}
```

```

class HarleyDavidson extends Motorcycle {
  public function __construct($price) {
    parent::__construct($price);
  }
}

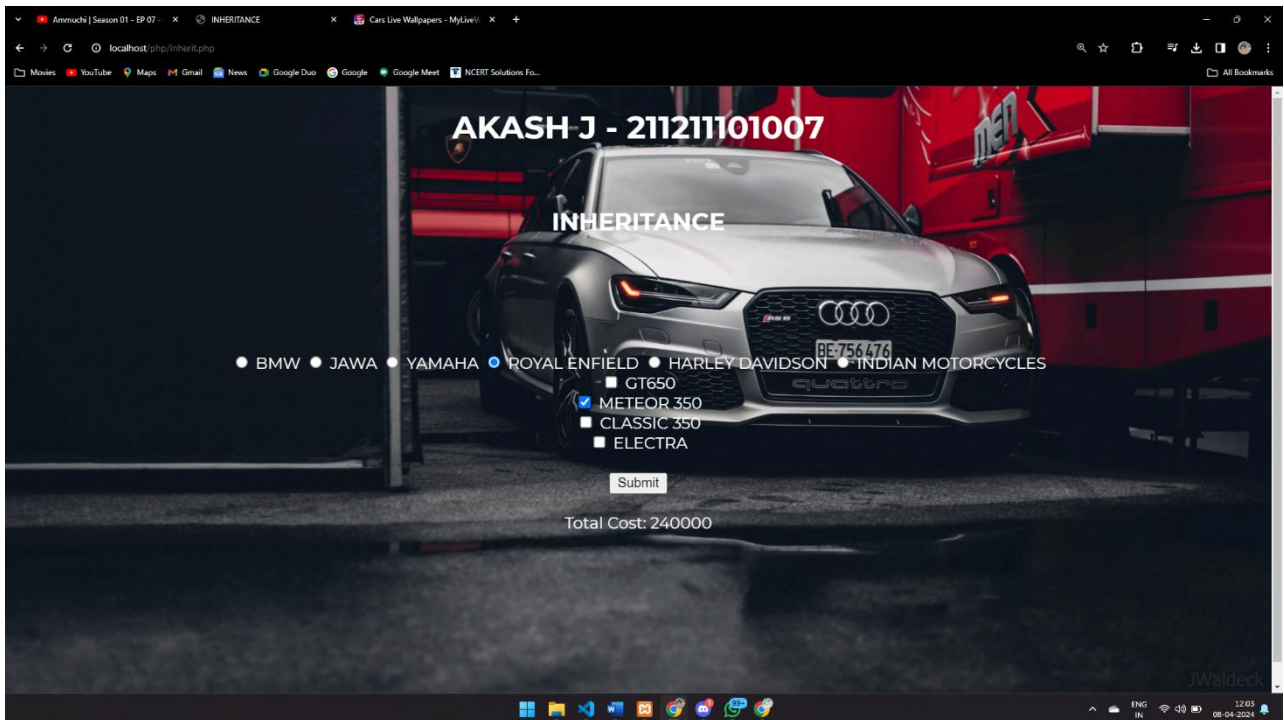
class IndianMotorcycles extends Motorcycle {
  public function __construct($price) {
    parent::__construct($price);
  }
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
  if(isset($_POST['checkbox'])) {
    $checkboxes = $_POST['checkbox'];
    $totalCost = 0;
    $prices = array(
      "BMW1" => 310000,
      "BMW2" => 850000,
      "BMW3" => 2000000,
      "JAWA1" => 180000,
      "JAWA2" => 190000,
      "JAWA3" => 200000,
      "YAMA1" => 150000,
      "YAMA2" => 40000,
      "YAMA3" => 76000,
      "YAMA4" => 180000,
      "RE1" => 400000,
      "RE2" => 240000,
      "RE3" => 250000,
      "RE4" => 220000,
      "HD1" => 1020000,
      "HD2" => 1900000,
      "HD3" => 3000000,
      "HD4" => 1200000,
      "IM1" => 1800000,
      "IM2" => 2500000,
      "IM3" => 3000000
    );
    foreach ($checkboxes as $bikeModel) {
      $totalCost += $prices[$bikeModel];
    }
  }
}

```

```
    echo "Total Cost: " . $totalCost;
  }
}
</center>
</body>
</html>
```

OUTPUT:



RESULT:

Thus execution of inheritance in php has been done and verified successfully.

CONSTRUCTOR AND DESTRUCTOR

AIM:

The aim of the code is to demonstrate the concept of interfaces and object-oriented programming in PHP. The code uses an Animal interface with construct and destruct methods and defines several classes that implement this interface.

ALGORITHM:

STEP 1: Start the program.

STEP 2: Set the background image URL (e.g., "kinji.jpg").

STEP 3: Display the HTML document structure with the following elements:

- <html> tag with the language attribute set to "en".
- <head> tag containing meta information and the title "1021-Aravind Interface".
- CSS styles for the body, label and box class.
- <body> tag with the background image set using PHP echo.
- Display "Aravind - 211211101021" and "Welcome to our Ice Cream Shop!" headings.

STEP 4: Create a form with method="POST" and action="".

- Display radio buttons for ice cream flavors.
- Display an input field for the number of scoops (1-10) and a submit button.
- PHP code block to process the form data:
- If the form is submitted (\$_POST['submit'] is set):

STEP 5: Retrieve the selected flavor and quantity from \$_POST.

- Define an interface Order Interface with a place Order method.
- Define a class Ice cream Shop that implements Order Interface:
- Constructor initializes flavor and quantity and displays the order message.
- Destructor thanks the customer for visiting.
- place Order method indicates successful order placement.
- Create an instance of Ice cream Shop with the selected flavor and quantity.

STEP 6: Call the place Order method on the instance.

- End the PHP code block.
- Close the form and the <div> container.

STEP 7: End the HTML document structure.

PROGRAM:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title> Interface</title>
</head>
<style>
  body {
    display: flex;
    flex-direction: column;
    align-items: center;
  }
  label{
    font-weight: bold;
  }
  .box {
    display: flex;
    flex-direction: column;
  }
</style>
<style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;
  }
@media (min-aspect-ratio: 16/9){

  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  211211101007
```

```

video{
    width: auto;
    height: 100%;
}

```

```

.backbtn{
    position:center;
}

```

```

body{
    background-image:url('<?php echo $img;?>');
    background-size:cover;
    background-position:fixed;
    background-repeat:no-repeat;
    color:white;
}

```

```
h1 {color:white}
```

```
p{color:white}
```

```
body{color:white}
```

```
</style>
```

```
<body>
```

```
<video autoplay loop muted plays-inline class="back-video"
```

```
    src="kinji.mp4">
```

```
    </video>
```

```
<h2>Aravind - 211211101021</h2>
```

```
<h2>Welcome to our Ice Cream Shop!</h2>
```

```
<div class="box">
```

```
<form method="POST" action="">
```

```
    <label>Select Flavor:</label><br>
```

```
        <input type="radio" name="flavor" value="Pumpkin">Pumpkin<br>
```

```
        <input type="radio" name="flavor" value="London Fog">London Fog<br>
```

```
        <input type="radio" name="flavor" value="Chunky Banana">Chunky Fruit<br>
```

```
        <input type="radio" name="flavor" value="Cream Cheese">Cream Cheese<br>
```

```
        <input type="radio" name="flavor" value="Cinnamon-Basil">Cinnamon-Basil<br>
```

```
        <input type="radio" name="flavor" value="Toasted Hazelnuts">Toasted Hazelnuts<br>
```

```

        <input type="radio" name="flavor" value="Lavender Peach Gelato">Lavender Peach Gelato<br>
        <input type="radio" name="flavor" value="Triple Chocolate Ricotta">Triple Chocolate ricotta
    <br><br>
</ul>
<label>Number of Scoops: </label>
<input type="number" name="quantity" min="1" max="100" required>
<input type="submit" name="submit" value="Place Order"><br><br>
</form>
</div>
<?php
if (isset($_POST['submit'])) {
    $flavor = $_POST['flavor'];

    $quantity = $_POST['quantity'];

    interface OrderInterface {
        public function placeOrder();
    }

    class IcecreamShop implements OrderInterface {
        public $flavor;
        public $quantity;

        public function __construct($flavor, $quantity) {
            $this->flavor = $flavor;
            $this->quantity = $quantity;
            echo "<center>You have selected $quantity scoop(s) of $flavor ice cream</center>.<br><br>";
        }

        public function __destruct() {

        }

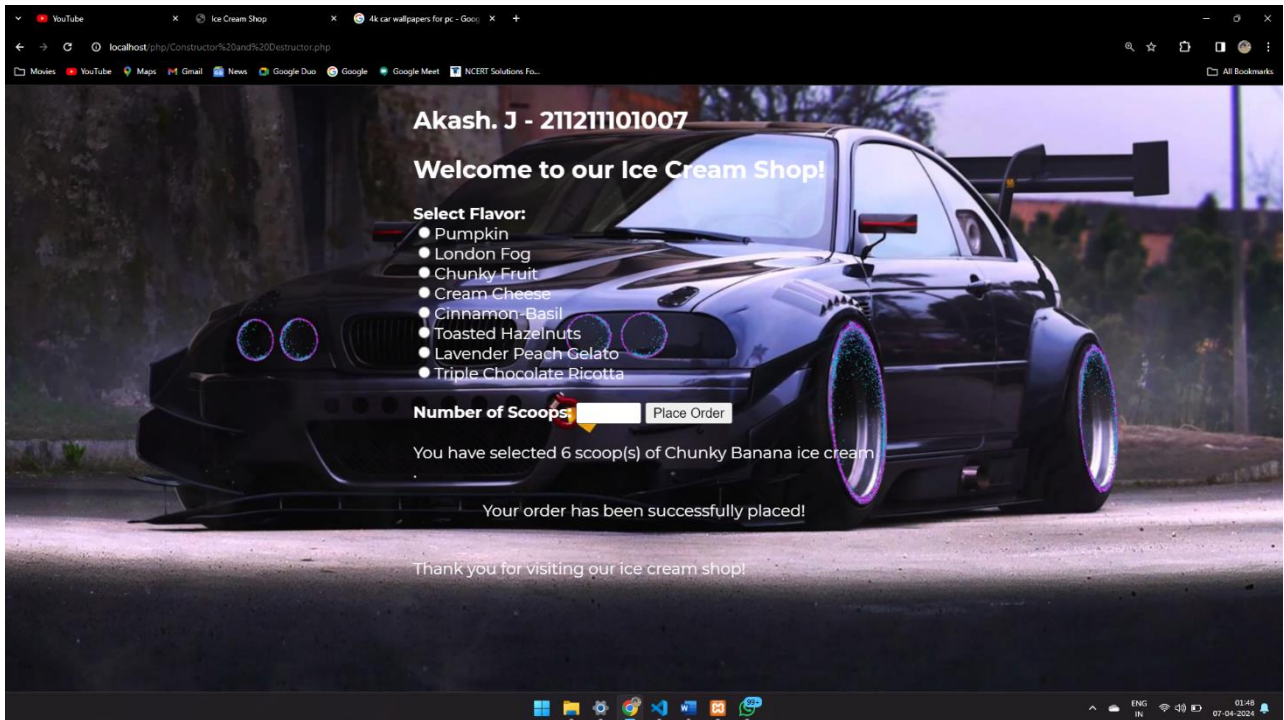
        echo "Thank you for visiting our ice cream shop!<br>";
    }

    public function placeOrder() {
        echo "<center>Your order has been successfully placed!</center><br><br>";
    }
}

```

```
$icecreamOrder = new IcecreamShop($flavor, $quantity);  
$icecreamOrder->placeOrder();  
}  
?>  
  
</body>  
</html>
```

OUTPUT:



RESULT:

Thus the execution of interface in php has been done and verified successfully .

MATRIX MULTIPLICATION

AIM:

The aim of the program is to perform matrix multiplication .the user is prompted to input the dimensions for two matrices ,and then fill in the values for each matrix upon submitting the matrices ,the code calculates their product and displays the result

ALGORITHM:

STEP1: start the program

STEP2: HTML Form for Matrix Dimensions

- Display a form to input the number of rows and columns for Matrix A and Matrix B.
- Use the POST method to submit the form data.

STEP3: PHP Code to Handle Form Submission

- Retrieve the dimensions of Matrix A and Matrix B from \$_POST.
- Check if the number of columns in Matrix A is equal to the number of rows in Matrix B for multiplication.
 1. If not, display an error message.
 2. If yes, display forms to input the values for Matrix A and Matrix B.

STEP4: HTML Form to Input Matrix Values

- Display forms to input the values for Matrix A and Matrix B using tables.
- Use random background colors for the table cells.

STEP5: PHP Code to Handle Matrix Multiplication

- Retrieve the input matrices from \$_POST.
- Calculate the product of Matrix A and Matrix B.
- Display the result matrix.

STEP6: end the program

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
  <title>Matrix Multiplication</title>
  <style>
    table {
      border-collapse: collapse;
      width: 50%;
    }
    th, td {
      border: 1px solid black;
      padding: 8px;
      text-align: center;
    }
    .row-color-odd td:nth-child(odd), .row-color-even td:nth-child(even) {
      background-color: lightgray;
    }
    .row-color-odd td:nth-child(even), .row-color-even td:nth-child(odd) {
      background-color: lightblue;
    }
  </style>
</style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;
}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}
```

```

    }
}

.backbtn{
    position:center;
}

body{
    background-image:url('<?php echo $img;?>');
    background-size:cover;
    background-position:fixed;
    background-repeat:no-repeat;
    color:white;
}
h1{color:white}
p{color:white}
body{color:white}
</style>
</head>
<body>
<video autoplay loop muted plays-inline class="back-video"
    src="tun.mp4">
    </video>

<center>
<h1>ARAVIND.A - 211211101021</h1>
<h2>Matrix Multiplication</h2>
<form method="post">
    <label>Enter the number of rows for Matrix A:</label>
    <input type="number" name="rows_a" required><br><br>
    <label>Enter the number of columns for Matrix A:</label>
    <input type="number" name="cols_a" required><br><br>

    <label>Enter the number of rows for Matrix B:</label>
    <input type="number" name="rows_b" required><br><br>
    <label>Enter the number of columns for Matrix B:</label>
    <input type="number" name="cols_b" required><br><br>

    <input type="submit" value="Generate Matrix">
</form>

```



```

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $rows_a = $_POST['rows_a'];
    $cols_a = $_POST['cols_a'];
    $rows_b = $_POST['rows_b'];
    $cols_b = $_POST['cols_b'];

    if ($cols_a != $rows_b) {
        echo "<p>Error: Number of columns in Matrix A must be equal to the number of rows in Matrix B for
multiplication.</p>";
    } else {
        echo "<h3>Matrix A</h3>";
        echo "<form method='post'>";
        echo "<table>";
        for ($i = 0; $i < $rows_a; $i++) {
            echo "<tr class='row-color-" . (($i % 2 == 0) ? "even" : "odd") . "'>";
            for ($j = 0; $j < $cols_a; $j++) {
                echo "<td style='background-color: #" . dechex(rand(0x000000, 0xFFFFFF)) . "'><input type='number'
name='matrix_a[$i][$j]' required></td>";
            }
            echo "</tr>";
        }
        echo "</table>";

        echo "<h3>Matrix B</h3>";
        echo "<table>";
        for ($i = 0; $i < $rows_b; $i++) {
            echo "<tr class='row-color-" . (($i % 2 == 0) ? "even" : "odd") . "'>";
            for ($j = 0; $j < $cols_b; $j++) {
                echo "<td style='background-color: #" . dechex(rand(0x000000, 0xFFFFFF)) . "'><input type='number'
name='matrix_b[$i][$j]' required></td>";
            }
            echo "</tr>";
        }
        echo "</table>";

        echo "<input type='hidden' name='rows_a' value='$rows_a'>";
        echo "<input type='hidden' name='cols_a' value='$cols_a'>";
        echo "<input type='hidden' name='rows_b' value='$rows_b'>";
        echo "<input type='hidden' name='cols_b' value='$cols_b'>";
        echo "<br><input type='submit' value='Multiply Matrices'>";
    }
}

```

```

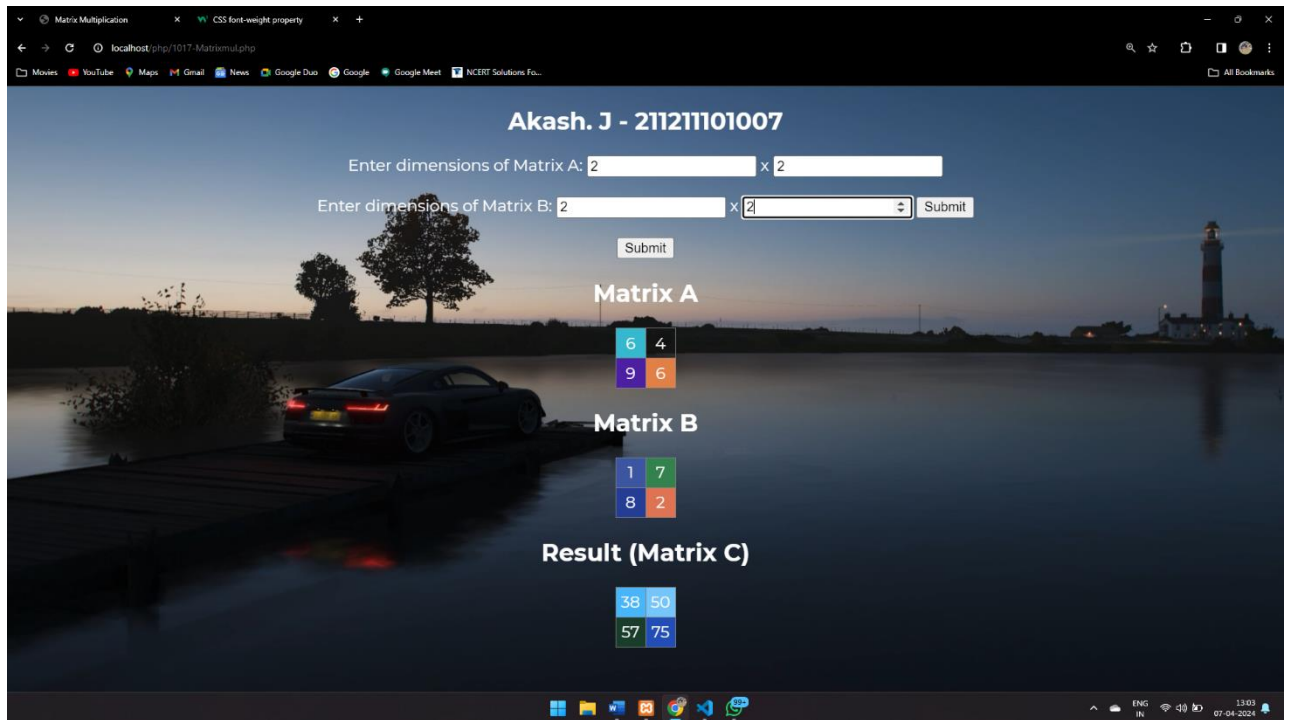
echo "</form>";

if (isset($_POST['matrix_a']) && isset($_POST['matrix_b'])) {
    $matrix_a = $_POST['matrix_a'];
    $matrix_b = $_POST['matrix_b'];

    echo "<h3>Result Matrix</h3>";
    echo "<table>";
    for ($i = 0; $i < $rows_a; $i++) {
        echo "<tr>";
        for ($j = 0; $j < $cols_b; $j++) {
            $result = 0;
            for ($k = 0; $k < $cols_a; $k++) {
                $result += $matrix_a[$i][$k] * $matrix_b[$k][$j];
            }
            echo "<td style='background-color: #" . dechex(rand(0x000000, 0xFFFFFF)) . "'>$result</td>";
        }
        echo "</tr>";
    }
    echo "</table>";
}
}
?>
</body>
</html>

```

OUTPUT:



RESULT:

The implementation of matrix multiplication that accepts user input of any dimension has been executed and verified successfully

SORTING OF AN ARRAY**AIM:**

To create a PHP program that allows users to input an array of integers and select a sorting method (Bubble Sort, Selection Sort, or Insertion Sort), and then displays the sorted array.

ALGORITHM:**BUBBLESORT ALGORITHM:**

STEP1: Start from the beginning of the array.

STEP2: Compare each element with the next one.

STEP3: If the current element is greater than the next one, swap them.

STEP4: Repeat this process until the end of the array.

STEP5: Continue iterating through the array until no more swaps are needed.

SELECTIONSORT ALGORITHM:

STEP1: Divide the array into two subarrays: sorted and unsorted.

STEP2: Find the minimum element in the unsorted subarray.

STEP3: Swap it with the first element of the unsorted subarray.

STEP4: Move the boundary between the sorted and unsorted subarrays one element to the right.

STEP5: Repeat until the entire array is sorted.

INSERTION ALGORITHM:

STEP1: Start with the second element of the array.

STEP2: Compare it with the elements to its left, moving them one position to the right until finding the correct position for insertion.

STEP3: Insert the element into its correct position in the sorted subarray.

STEP4: Repeat this process for each subsequent element in the array.

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
  <title>Array Sorting Program</title>
</head>
<style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;
}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}

.backbtn{
  position:center;
}

body{
  background-image:url('<?php echo $img;?>');
  background-size:cover;
  background-position:fixed;
  background-repeat:no-repeat;
  color:white;
}
h1{color:white}
p{color:white}
body{color:white}
</style>
<body><video autoplay loop muted plays-inline class="back-video"
  src="zor.mp4">
</video>
<center>
<h1>Akash J- 211211101007</h1>
<h2>Array Sorting Program</h2>
<form method="post">
  <label for="array">Enter array values (comma separated): </label><br>
  <input type="text" id="array" name="array"><br><br>
  <input type="radio" id="bubble" name="sort_method" value="bubble">
  <label for="bubble">Bubble Sort</label><br>
  <input type="radio" id="selection" name="sort_method" value="selection">
  <label for="selection">Selection Sort</label><br>
  <input type="radio" id="insertion" name="sort_method" value="insertion">
  <label for="insertion">Insertion Sort</label><br><br>
  <input type="submit" value="Sort">
</form>
```

```

<?php
function bubbleSort($arr) {
    $n = count($arr);
    for ($i = 0; $i < $n-1; $i++) {
        for ($j = 0; $j < $n-$i-1; $j++) {
            if ($arr[$j] > $arr[$j+1]) {
                $temp = $arr[$j];
                $arr[$j] = $arr[$j+1];
                $arr[$j+1] = $temp;
            }
        }
    }
    return $arr;
}

function selectionSort($arr) {
    $n = count($arr);
    for ($i = 0; $i < $n-1; $i++) {
        $min_index = $i;
        for ($j = $i+1; $j < $n; $j++) {
            if ($arr[$j] < $arr[$min_index]) {
                $min_index = $j;
            }
        }
        $temp = $arr[$i];
        $arr[$i] = $arr[$min_index];
        $arr[$min_index] = $temp;
    }
    return $arr;
}

function insertionSort($arr) {
    $n = count($arr);
    for ($i = 1; $i < $n; $i++) {
        $key = $arr[$i];
        $j = $i - 1;
        while ($j >= 0 && $arr[$j] > $key) {
            $arr[$j + 1] = $arr[$j];
            $j--;
        }
        $arr[$j + 1] = $key;
    }
    return $arr;
}

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $array = explode(',', $_POST["array"]);
    $sort_method = $_POST["sort_method"];

    $array = array_map('trim', $array);

    $array = array_filter($array);

    $array = array_map('intval', $array);

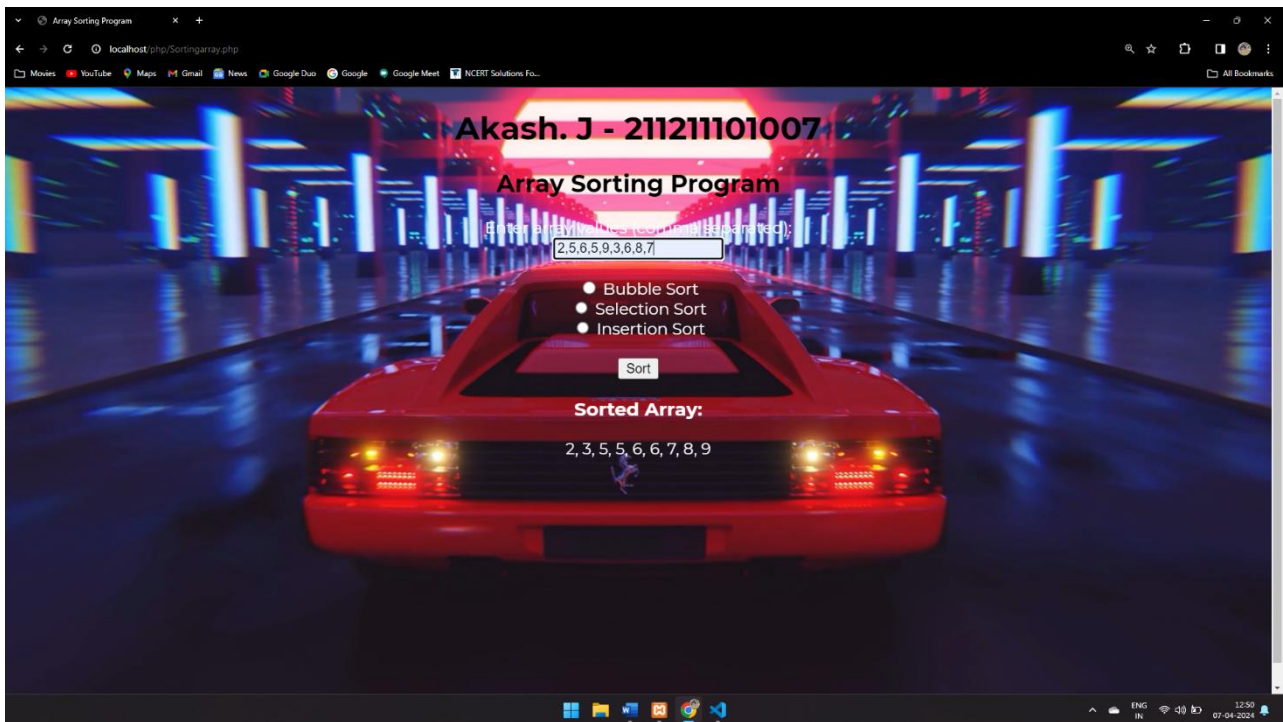
    switch ($sort_method) {
        case 'bubble':
            $sorted_array = bubbleSort($array);
            break;
    }
}

```

```
    case 'selection':
        $sorted_array = selectionSort($array);
        break;
    case 'insertion':
        $sorted_array = insertionSort($array);
        break;
    default:
        echo "Invalid sorting method selected.";
        break;
}

echo "<h3>Sorted Array:</h3>";
echo implode(' ', $sorted_array);
}
?>
</body>
</html>
```

OUTPUT:



RESULT:

Thus the implementation of different sorting algorithms in php has been executed and verified successfully

EXPNO: 10

DATE:16-02-2024

STUDENT REGISTRATION FORM

AIM:

The aim of the program is to create a student registration form. The user is prompted to input various details such as name, register number, gender, age, department, college name, mobile number, and email. Upon submitting the form, the details are stored in a MySQL database.

ALGORITHM:

STEP1: Start the program

STEP2: HTML Form for Student Registration

- Display a form to input student details:
- Use the POST method to submit the form data.

STEP3: PHP Code to Handle Form Submission and Database Connection

- Establish a connection to the MySQL database.
- Retrieve the student details from \$_POST.
- Create an SQL query to insert the student details into the database.
- Execute the SQL query.
 - 1.If the query is successful, display a success message.
 - 2.If the query fails, display an error message.

STEP4: HTML Form to View Student Details

- Display a button to view the student details.
- Use the GET method to redirect to the exp11.php page to view the details.

STEP5: end the program

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
  <title>Student Registration Form</title>
</head>
<style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;

}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}

.backbtn{
  position:center;
}

body{
  background-image:url('<?php echo $img;?>');
  background-size:cover;
  background-position:fixed;
  background-repeat:no-repeat;
  color:white;
}
```

```

h1 {color:white}
p {color:white}
body {color:white}
</style>
<body>
<video autoplay loop muted plays-inline class="back-video"
    src="thousand.mp4">
    </video>
    <center>
    <h2>Akash J- 211211101007</h2>
    <h2>Student Registration Form</h2>
    <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
        Name: <input type="text" name="name"><br><br>
        Register Number: <input type="number" name="register_number"><br><br>
        Gender: <input type="radio" name="gender" value="Male"> Male
                <input type="radio" name="gender" value="Female"> Female<br><br>
        Age: <input type="number" name="age"><br><br>
        Department: <input type="text" name="department"><br><br>
        College Name: <input type="text" name="college_name"><br><br>
        Mobile no: <input type="number" name="mobile"><br><br>
        Mail-id: <input type="text" name="mail"><br><br>
        <input type="submit" name="submit" value="Submit">
    </form><br>

    <form method="get" action="EX11.php">
        <input type="submit" name="view_details" value="View Details">
    </form>

    <?php

    $servername = "localhost";
    $username = "root";
    $password = "";
    $dbname = "student registration form";

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

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

```

```

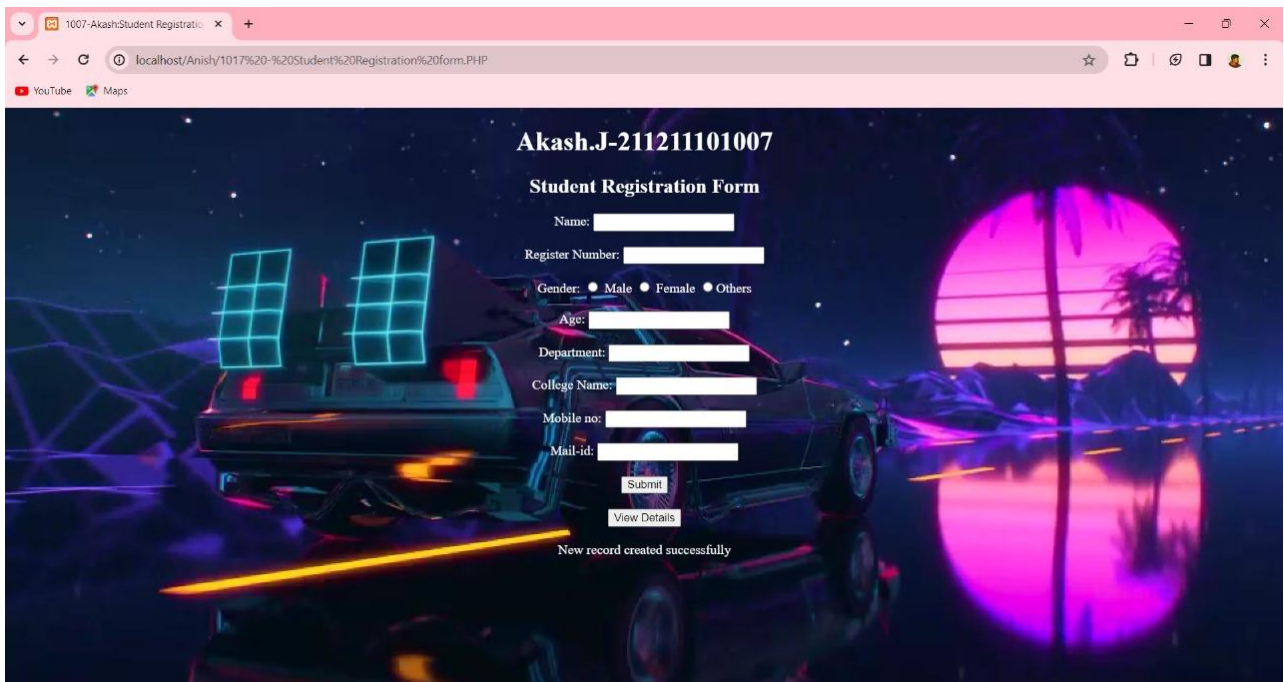
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = $_POST['name'];
    $register_number = $_POST['register_number'];
    $gender = $_POST['gender'];
    $age = $_POST['age'];
    $department = $_POST['department'];
    $college_name = $_POST['college_name'];
    $mobile = $_POST['mobile'];
    $mail = $_POST['mail'];
    $sql = "INSERT INTO students (name, register_number, gender, age, department, college_name, mobile, mail)
    VALUES ('$name', '$register_number', '$gender', '$age', '$department', '$college_name', '$mobile', '$mail')";

    if ($conn->query($sql) === TRUE) {
        echo "<br>New record created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $conn->error;
    }
}

$conn->close();
?>
</body>
</html>>

```

OUTPUT:



The screenshot displays a web browser window with a single tab titled '1007-AkashStudent Registrati...'. The address bar shows the URL 'localhost/Anishy/1017%20-%20Student%20Registration%20form.PHP'. The page features a dark, futuristic background with a car and a large orange sun. The form is titled 'Akash.J-211211101007' and 'Student Registration Form'. It includes input fields for Name, Register Number, Age, Department, College Name, Mobile no, and Mail-id. There are radio buttons for Gender (Male, Female, Others) and buttons for Submit and View Details. A message at the bottom states 'New record created successfully'.

Akash.J-211211101007

Student Registration Form

Name:

Register Number:

Gender: ☒ Male ☐ Female ☐ Others

Age:

Department:

College Name:

Mobile no:

Mail-id:

New record created successfully

RESULT:

Thus implementation of student registration form using php and my sql has been executed and verified successfully.

EXPNO: 11

DATE:22-02-2024

INSERTING RECORD INTO A DATABASE TABLE USING PHP/MYSQL

AIM:

The aim of the program is to display the details of registered students from the MySQL database in a table format on an HTML page. The student details include Name and Register Number, and additional details (Gender, Age, Department, College Name, and Mobile) are displayed in a hidden row.

ALGORITHM:

STEP1: start the program

STEP2: HTML Structure and Styling

2.1 Display a video background with styling.

2.2 Display a table to show student details:

1.Name 2. Register Number

STEP3: Define JavaScript Function to Toggle Details

3.1 Define function toggle Details(id)

3.1.1 Retrieve the element with the provided id

3.1.2 If details are hidden, display them

3.1.3 If details are displayed, hide them

STEP4: Retrieve Student Details from Database and Display

4.1 Establish MySQL database connection using \$server name, \$username, \$password, and \$db name

4.2 Check if \$_SERVER["REQUEST_METHOD"] == "GET"

4.3 Retrieve all student details using SQL query "SELECT * FROM students"

4.4 Display student details in a table format:

a. Display Name and Register Number as clickable links that trigger toggle Details(id)

b. Display additional details (Gender, Age, Department, College Name, Mobile) in a hidden row with class "details" and unique id "details_" followed by the register number

STEP5: End the program

PROGRAM:

```
<!DOCTYPE html>
<html>
<head>
  <title>Student Details</title>
  <style>
    table {
      border-collapse: collapse;
      width: 100%;
    }

    th, td {
      border: 1px solid #ddd;
      padding: 8px;
      text-align: left;
    }

    th {
      background-color: #f2f2f2;
    }

    .details {
      display: none;
    }
  </style>
  <script>
    function toggleDetails(id) {
      var details = document.getElementById(id);
      if (details.style.display === "none") {
        details.style.display = "table-row";
      } else {
        details.style.display = "none";
      }
    }
  </script>
</head>
<style>
video{
  position: absolute;
  right: 0;
```

```

    bottom: 0;
    z-index: -1;

}

@media (min-aspect-ratio: 16/9){
    video{
        width: 100%;
        height: auto;
    }
}

@media (max-aspect-ratio: 16/9){
    video{
        width: auto;
        height: 100%;
    }
}

.backbtn{
    position:center;
}

body{
    background-image:url('<?php echo $img;?>');
    background-size:cover;
    background-position:fixed;
    background-repeat:no-repeat;
    color:white;
}

h1{color:white}
p{color:white}
body{color:white}
</style>
<body>
<video autoplay loop muted plays-inline class="back-video"
    src="toji.mp4">
    </video>
    <h2>Akash J - 211211101007</h2>
<h2>Student Details</h2>
<table>
    <tr>

```



```

        <th>Name</th>
        <th>Register Number</th>
    </tr>

<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "student registration form";

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

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

$sql = "SELECT * FROM students";
$result = $conn->query($sql);

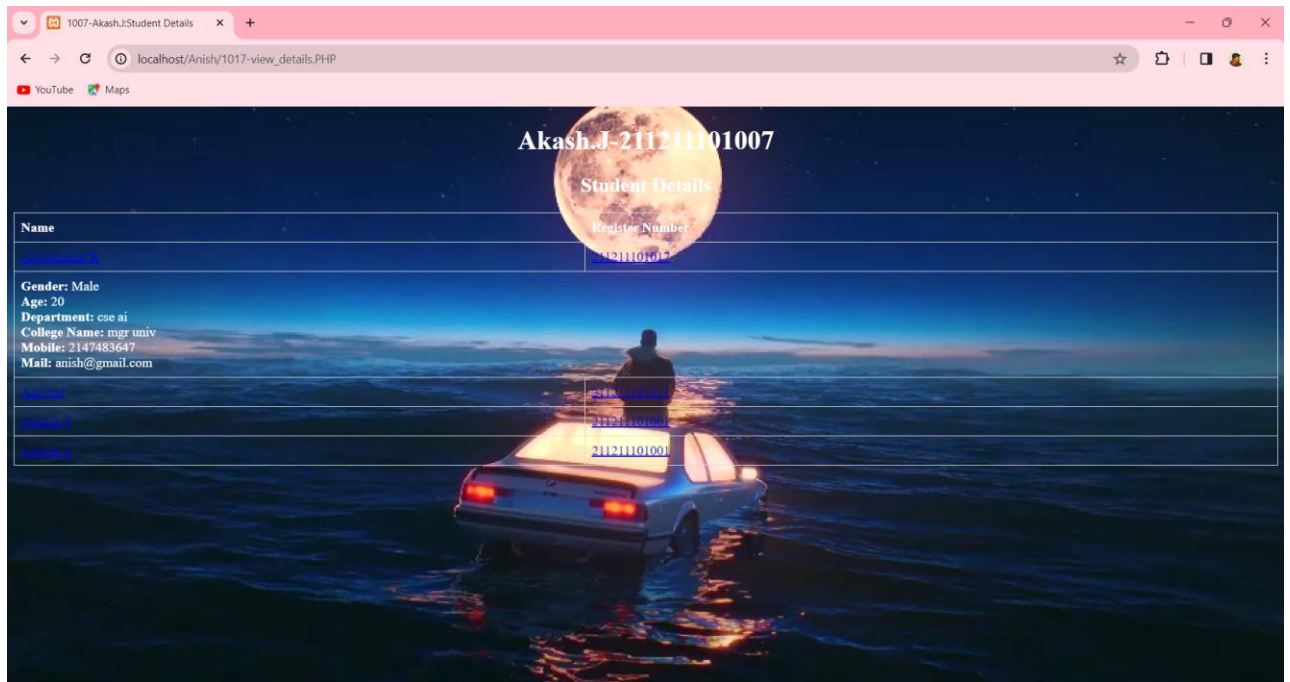
if ($result->num_rows > 0) {

    while($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo "<td><a href='javascript:void(0);' onclick='toggleDetails(\"details_\" . $row[\"register_number\"] . \"\");'>\" . $row[\"name\"] . \"</a></td>\";
        echo "<td><a href='javascript:void(0);' onclick='toggleDetails(\"details_\" . $row[\"register_number\"] . \"\");'>\" . $row[\"register_number\"] . \"</a></td>\";
        echo "</tr>";
        echo "<tr class='details' id='details_\" . $row[\"register_number\"] . \"'>\";
        echo "<td colspan='8'>\";
        echo "<strong>Gender:</strong> \" . $row[\"gender\"] . \"<br>\";
        echo "<strong>Age:</strong> \" . $row[\"age\"] . \"<br>\";
        echo "<strong>Department:</strong> \" . $row[\"department\"] . \"<br>\";
        echo "<strong>College Name:</strong> \" . $row[\"college_name\"] . \"<br>\";
        echo "<strong>Mobile:</strong> \" . $row[\"mobile\"] . \"<br>\";
        echo "<strong>Mail:</strong> \" . $row[\"mail\"] . \"<br>\";
        echo "</td>\";
        echo "</tr>\";
    }
} else {
    echo "0 results";
}

```

```
}  
    $conn->close();  
    ?>  
</table>  
</body>  
</html>
```

OUTPUT:



RESULT:

Thus implementing displaying the details of registered students from the MySQL database in a table format on an HTML page has been done and verified successfully.

EXPNO: 12

DATE:11-01-2024

TRACKING THE NUMBER OF VISITS MADE TO A PAGE

AIM:

The aim of the program is to display the total number of visits to webpage. each time the page is loaded ,the visit count is incremented and stored in a text file .the total number of visits is then displayed on the webpage.

ALGORITHM:

STEP1: start the program

STEP2: Define PHP Function to Increment Visit Count

2.1 Define function `increment Visit Count(\$file)`

2.1.1 Check if the file `\$file` exists

a. If not, set the count to 0

b. If yes, read the current count from the file

2.1.2 Increment the count by 1

2.1.3 Write the updated count back to the file

2.1.4 Return the updated count.

STEP3: Display Total Visit Count

3.1 Call the `increment Visit Count(\$visit_count_file)` function to get the updated visit count

3.2 Display the total visit count on the webpage

STEP4: End the program

PROGRAM:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Page Visit Count</title>
</head><style>
video{
  position: absolute;
  right: 0;
  bottom: 0;
  z-index: -1;

}
@media (min-aspect-ratio: 16/9){
  video{
    width: 100%;
    height: auto;
  }
}
@media (max-aspect-ratio: 16/9){
  video{
    width: auto;
    height: 100%;
  }
}

.backbtn{
  position:center;
}

body{
  background-image:url('<?php echo $img;?>');
  background-size:cover;
  background-position:fixed;
  background-repeat:no-repeat;
  color:white;
}
```

```

h1 {color:white}
p {color:white}
body {color:white}
</style>
<body>
<video autoplay loop muted plays-inline class="back-video"
src="fun.mp4">
</video>
<h2>Akash J - 211211101007</h2>
<?php

$visit_count_file = 'visit_count.txt';
function incrementVisitCount($file) {

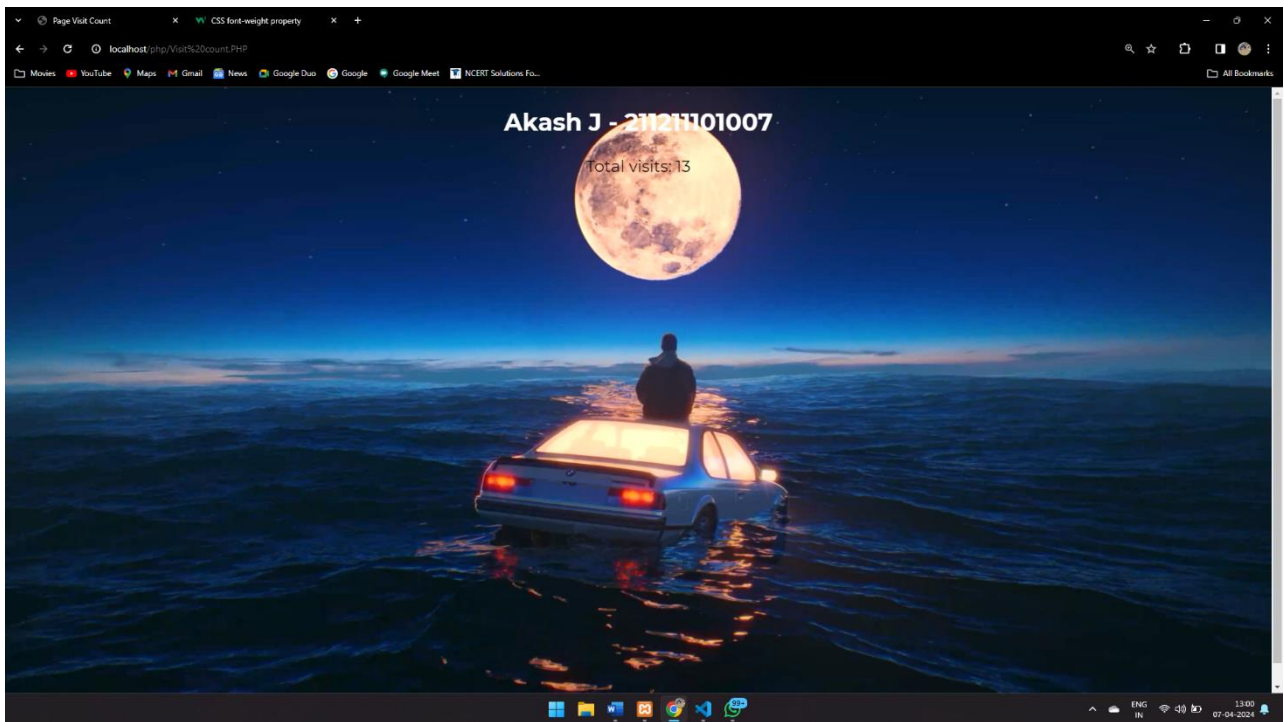
    if (!file_exists($file)) {
        $count = 0;
    } else {

        $count = (int)file_get_contents($file);
    }
    $count++;
    file_put_contents($file, $count);
    return $count;
}

$visit_count = incrementVisitCount($visit_count_file);
echo "Total visits: $visit_count";
?>
</body>
</html>

```

OUTPUT:



RESULT: Hence the php program to track the number of visits made to a page is executed and the output verified.