

**Que 1: Write regular expressions including modifiers, operators, and metacharacters.**

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Contact Us Form</title>
  <style>
    body {
      background-color: #e89510;
      padding: 0;
      margin: 0;
    }
    form {
      background-color: #e89510;
      border-radius: 10px;
      margin: 20px auto;
      width: 80%;
      max-width: 500px;
    }
    .form_heading {
      text-align: center;
    }
    .cno_err {
      text-align: center;
    }

    input[type="submit"] {
      background-color: blue;
      color: white;
      padding: 10px 20px;
      border-radius: 5px;
      cursor: pointer;
    }
    input[type="text"],
    input[type="email"] {
      width: 55%;
      padding: 10px;
      margin: 5px 0;
      border: 1px solid grey;
      border-radius: 5px;
      box-sizing: border-box;
    }
    input[type="submit"]:hover {
      background-color: rgb(54, 54, 88); }
  </style>
</head>
<body>
```

```
</style>
</head>
<body>
  <h2 class="form_heading">Contact Us</h2>
  <form action="" method="post">
    <label for="name">Name:</label><br/>
    <input type="text" id="name" name="name" required><br><br>
    <label for="email">Email:</label><br/>
    <input type="email" id="email" name="email" required><br><br>
    <label for="contact">Contact Number:</label><br>
    <input type="text" id="contact" name="contact" required><br><br>
    <input type="submit" value="Submit">
  </form>

  <?php
  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = $_POST['name'];
    $email = $_POST['email'];
    $contact = $_POST['contact'];

    // Email Validation
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
      echo "Invalid email format.<br>";
    }

    // Contact Number Validation
    if (!preg_match("/^[0-9]{10}$/", $contact)) {
      echo "Invalid contact number format. Please enter 10 digits only.<br>";
    }
    echo "Note: This message will automatically disappear after correct insertion of
    Contact Number.";
  }
  ?>
</body>
</html>
```

## Output:

The screenshot shows a web browser window with the title 'Contact Us Form' and the address bar displaying 'http://localhost/assi/que1.php'. The page has an orange background and is titled 'Contact Us'. It contains a form with three input fields: 'Name' (filled with 'Nitesh Singh Bhati'), 'Email' (filled with 'tnitesh355@gmail.com'), and 'Contact Number' (filled with '850685260'). Below the 'Contact Number' field is a blue 'Submit' button. A red error message is displayed below the button: 'Invalid contact number format. Please enter 10 digits only. Note: This message will automatically disappear after correct insertion of Contact Number.' The Windows taskbar is visible at the bottom, showing the search bar and various application icons. The system clock in the bottom right corner indicates 9:31 PM on 4/8/2024.

**Contact Us**

Name:  
Nitesh Singh Bhati

Email:  
tnitesh355@gmail.com

Contact Number:  
850685260

Submit

Invalid contact number format. Please enter 10 digits only.  
Note: This message will automatically disappear after correct insertion of Contact Number.

Activate Windows  
Go to Settings to activate Windows.

**Que 2: Write a program to show the usage of nested if statement.**

**Solution:**

```
<html >
<head>
  <title>Job Eligibility Checker</title>
</head>
<body bgcolor="#0FFA6D">
<div align="center"> <br><br> <u><h1>Job Eligibility Checker</h1></u><br><br>
  <form action="" method="post">
    <label for="age">Age:</label>
    <input type="number" id="age" name="age" required>
    <br><br>
    <label for="income">Income (annual):</label>
    <input type="number" id="income" name="income" required>
    <br><br>
    <input type="submit" name="submit" value="Check Eligibility">
  </form>

  <?php
  if (isset($_POST['submit'])) {
    $age = $_POST['age'];
    $income = $_POST['income'];

    if ($age >= 18) {
      echo "You are eligible to apply for a job. ";

      if ($income >= 30000) {
        echo " You meet the minimum income requirement.";
      } else {
        echo " You do not meet the minimum income requirement.";
      }
    } else {
      echo "You are not eligible to apply for a job because you are under 18 years old.";
    }
  }
  ?>
</div>
</body>
</html>
```

## Output:

**Job Eligibility Checker**

Age:

Income (annual):

You are eligible to apply for a job. You meet the minimum income requirement.

Activate Windows  
Go to Settings to activate Windows.

Type here to search

9:31 PM  
4/8/2024

**Ques 3: Write a program to create a menu driven program and show the usage of switch-case.**

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Arithmetic Operations</title>
</head>
<body bgcolor="#FAA40F">
  <div align="center"> <br><br><u> <h1>Arithmetic Operations</h1></u><br>
  <form method="post" action="">
    <label for="num1">Enter first number:</label>
    <input type="number" id="num1" name="num1" required><br><br>

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

    <label for="operation">Select operation:</label>
    <select id="operation" name="operation">
      <option value="add">Addition</option>
      <option value="subtract">Subtraction</option>
      <option value="multiply">Multiplication</option>
      <option value="divide">Division</option>
    </select><br><br>

    <input type="submit" value="Calculate">
  </form>

  <?php
  // Check if form is submitted
  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve form data
    $num1 = $_POST['num1'];
    $num2 = $_POST['num2'];
    $operation = $_POST['operation'];

    // Perform arithmetic operation based on selected option
    switch ($operation) {
      case 'add':
        $result = $num1 + $num2;
        echo "<p>Result of addition: $result</p>";
        break;
      case 'subtract':
        $result = $num1 - $num2;
        echo "<p>Result of subtraction: $result</p>";
```

```
        break;
    case 'multiply':
        $result = $num1 * $num2;
        echo "<p>Result of multiplication: $result</p>";
        break;
    case 'divide':
        if ($num2 == 0) {
            echo "<p>Error: Cannot divide by zero</p>";
        } else {
            $result = $num1 / $num2;
            echo "<p>Result of division: $result</p>";
        }
        break;
    default:
        echo "<p>Error: Invalid operation</p>";
        break;
    }
}
?>
</div>
</body>
</html>
```

**Output:**

## Arithmetic Operations

Enter first number:

Enter second number:

Select operation:

Result of addition: 76

**Ques 4:** Write a program to show the usage of for/while/do while loop.

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Loop Examples with User Input</title>
</head>
<body align="center" bgcolor="#FF01D6">
  <h1><u>Loop Examples with User Input</u></h1><br><br>

  <form method="post">
    <label for="start">Enter start number:</label>
    <input type="number" id="start" name="start" required><br><br>

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

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

  <?php
  if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $start = $_POST['start'];
    $end = $_POST['end'];

    echo "<h2>For Loop Example:</h2>";
    echo "<p>Printing numbers from $start to $end using a for loop:</p>";
    echo "<ul>";
    for ($i = $start; $i <= $end; $i++) {
      echo "$i ";
    }
    echo "</ul>";

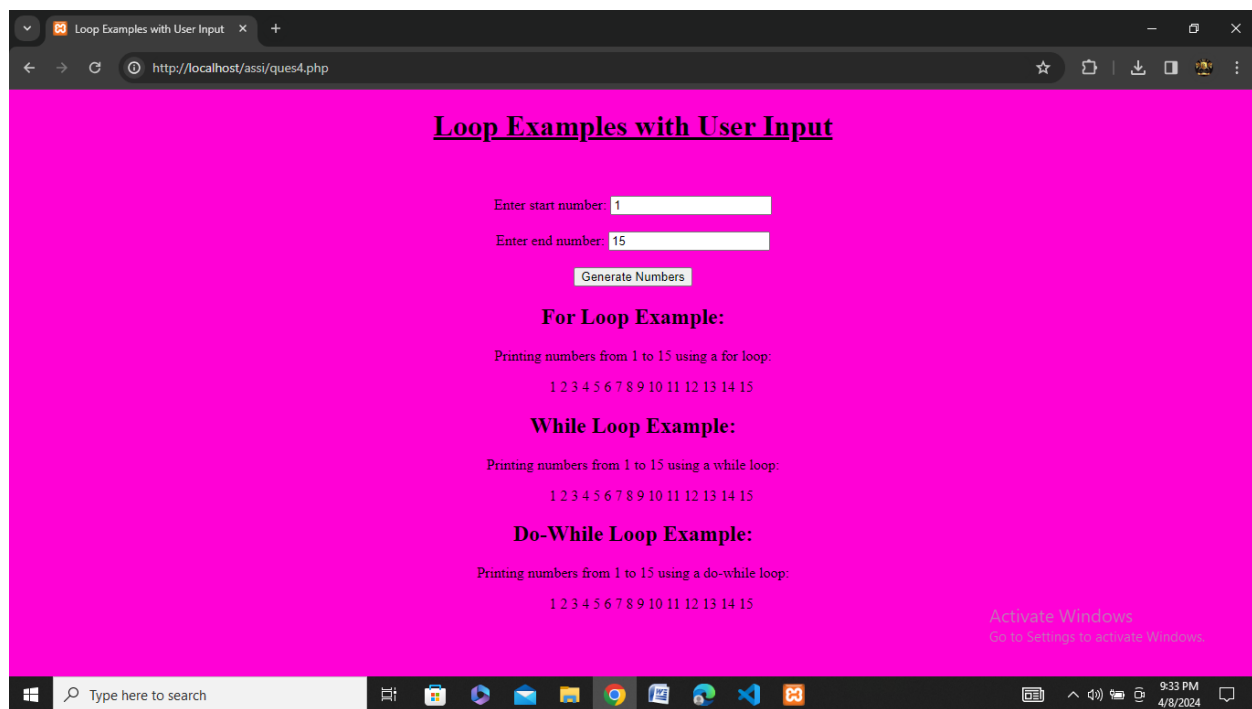
    echo "<h2>While Loop Example:</h2>";
    echo "<p>Printing numbers from $start to $end using a while loop:</p>";
    echo "<ul>";
    $j = $start;
    while ($j <= $end) {
      echo "$j ";
      $j++;
    }
    echo "</ul>";

    echo "<h2>Do-While Loop Example:</h2>";
```



```
echo "<p>Printing numbers from $start to $end using a do-while loop:</p>";
echo "<ul>";
$k = $start;
do {
    echo "$k ";
    $k++;
} while ($k <= $end);
echo "</ul>";
}
?>
</body>
</html>
```

### Output:



**Que 5: Write a program to perform all four types of sorting.**

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Sorting Algorithms</title>
</head>
<body bgcolor="#01B4FF">
  <h1>Sorting Algorithms</h1>

  <form method="post">
    <label for="numbers">Enter numbers (comma-separated):</label>
    <input type="text" id="numbers" name="numbers" required>
    <input type="submit" value="Sort">
  </form>

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

    // Remove any leading/trailing whitespace and convert strings to integers
    $numbers = array_map('trim', $numbers);
    $numbers = array_map('intval', $numbers);

    echo "<h2>Original Array:</h2>";
    printArray($numbers);

    // Bubble Sort
    $bubbleSorted = bubbleSort($numbers);
    echo "<h2>Bubble Sort:</h2>";
    printArray($bubbleSorted);

    // Selection Sort
    $selectionSorted = selectionSort($numbers);
    echo "<h2>Selection Sort:</h2>";
    printArray($selectionSorted);

    // Insertion Sort
    $insertionSorted = insertionSort($numbers);
    echo "<h2>Insertion Sort:</h2>";
    printArray($insertionSorted);

    // Quick Sort
```

```
$quickSorted = quickSort($numbers);  
echo "<h2>Quick Sort:</h2>";  
printArray($quickSorted);  
}
```

```
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++) {  
        $minIndex = $i;  
        for ($j = $i + 1; $j < $n; $j++) {  
            if ($arr[$j] < $arr[$minIndex]) {  
                $minIndex = $j;  
            }  
        }  
        $temp = $arr[$i];  
        $arr[$i] = $arr[$minIndex];  
        $arr[$minIndex] = $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 = $j - 1;  
        }  
        $arr[$j + 1] = $key;  
    }  
}
```

```
    return $arr;
}

function quickSort($arr) {
    $n = count($arr);
    if ($n <= 1) {
        return $arr;
    }
    $pivot = $arr[0];
    $left = $right = [];
    for ($i = 1; $i < $n; $i++) {
        if ($arr[$i] < $pivot) {
            $left[] = $arr[$i];
        } else {
            $right[] = $arr[$i];
        }
    }
    return array_merge(quickSort($left), [$pivot], quickSort($right));
}

function printArray($arr) {
    echo "<pre>";
    print_r($arr);
    echo "</pre>";
}
?>
</body>
</html>
```

**Output: (on next page=>)**

Sorting Algorithms

Enter numbers (comma-separated):

**Original Array:**

```
Array
(
    [0] => 2
    [1] => 4
    [2] => 88
    [3] => 7
    [4] => 3
    [5] => 45
)
```

**Bubble Sort:**

```
Array
(
    [0] => 2
    [1] => 3
    [2] => 4
    [3] => 7
    [4] => 45
    [5] => 88
)
```

**Selection Sort:**

```
Array
(
    [0] => 2
    [1] => 3
    [2] => 4
    [3] => 7
    [4] => 45
    [5] => 88
)
```

**Quick Sort:**

```
Array
(
    [0] => 2
    [1] => 3
    [2] => 4
    [3] => 7
    [4] => 45
    [5] => 88
)
```

**Selection Sort:**

```
Array
(
    [0] => 2
    [1] => 3
    [2] => 4
    [3] => 7
    [4] => 45
    [5] => 88
)
```

**Ques 6:** Write a program to implement Arraypad(), array\_slice(), array\_splice(), list()

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
<title>Array Functions</title>
<style>
body {
    font-family: Arial, sans-serif;
    padding: 20px;
    background-color: aquamarine;
}
h1 {
    text-align: center;
}
.container {
    display: flex;
    justify-content: center;
    margin-top: 20px;
}
.output {
    border: 1px solid #1a1a1a;
    padding: 10px;
    margin: 0 10px;
    background-color: azure;
}
</style>
</head>
<body>

<h1>Array Functions</h1>

<div class="container">
    <div class="output">
        <h2>Array Pad</h2>

        <?php
        $original_array = [1, 2, 3];
        $padded_array = array_pad($original_array, 5, 0);
        echo "<p>Original Array: " . implode(", ", $original_array) . "</p>";
        echo "<p>Array Pad: " . implode(", ", $padded_array) . "</p>";
        ?>
```

</div>

<div class="output">  
 <h2>Array Slice</h2>

```
<?php
$sliced_array = array_slice($original_array, 1, 2);
echo "<p>Original Array: " . implode(" ", $original_array) . "</p>";
echo "<p>Array Slice: " . implode(" ", $sliced_array) . "</p>";
?>
```

</div>

<div class="output">  
 <h2>Array Splice</h2>

```
<?php
$spliced_array = $original_array;
array_splice($spliced_array, 1, 2, [4, 5]);
echo "<p>Original Array: " . implode(" ", $original_array) . "</p>";
echo "<p>Array Splice: " . implode(" ", $spliced_array) . "</p>";
?>
```

</div>

<div class="output">  
 <h2>List</h2>

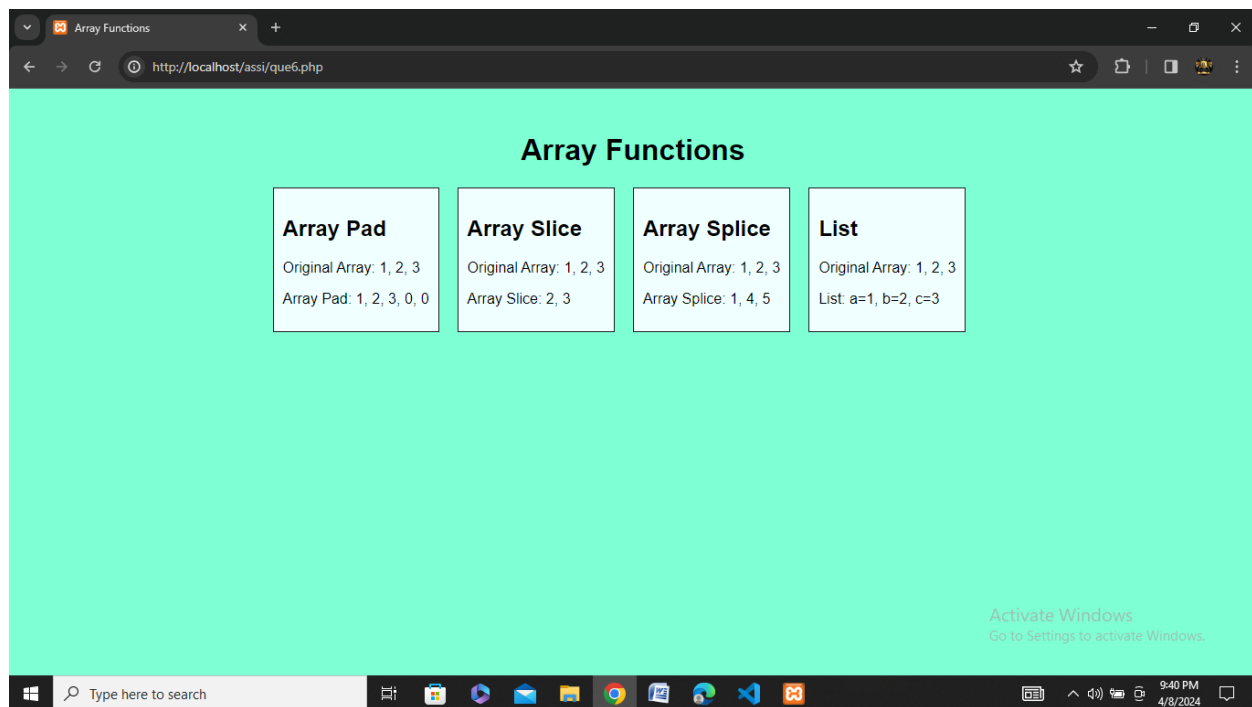
```
<?php
list($a, $b, $c) = $original_array;
echo "<p>Original Array: " . implode(" ", $original_array) . "</p>";
echo "<p>List: a=$a, b=$b, c=$c</p>";
?>
```

</div>

</div>

</body>  
</html>

## Output:





**Ques 7:** functions. (use foreach wherever applicable) .**Solution:**

```
<!DOCTYPE html>
<html>
<head>
<title>PHP Functions</title>
<link rel="stylesheet" href="foreach.css">
<style>
body {
    font-family: Arial, sans-serif;
    padding: 20px;
    background-color: #8fccf2;
}
h1 {
    text-align: center;
}
.container {
    display: flex;
    justify-content: center;
    margin-top: 50px;
}
.output {
    border: 2px solid #da5050;
    padding: 10px;
    margin: 0 10px;
    background-color: #2fcc3f;
}
</style>
</head>
<body>
<h1>PHP Functions</h1>
<div class="container">
    <div class="output">
        <h2>Factorial Calculation</h2>

        <?php
        function factorial($n) {
            if ($n == 0) {
                return 1;
            } else {
                return $n * factorial($n - 1);
            }
        }

        $number = 5;
```

```
echo "Factorial of $number is " . factorial($number);  
?>
```

```
</div>
```

```
<div class="output">  
  <h2>Reverse of Array</h2>
```

```
  <?php  
  function reverse_array($arr) {  
    return array_reverse($arr);  
  }
```

```
  $original_array = [1, 2, 3, 4, 5];  
  $reversed_array = reverse_array($original_array);
```

```
  foreach ($reversed_array as $element) {  
    echo $element . ",";  
  }  
  ?>
```

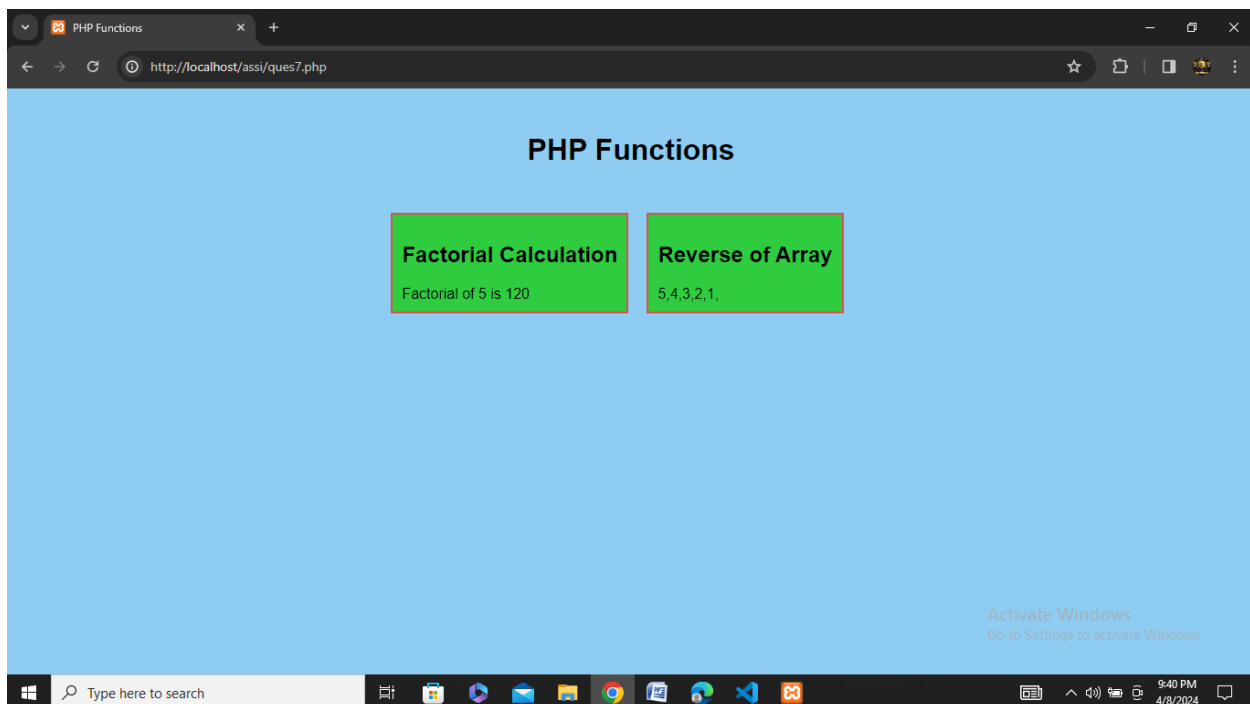
```
</div>
```

```
</div>
```

```
</body>
```

```
</html>
```

**Output:**



**Ques 8:** Write a program that Passes control to another page (include, require, exit and die functions).

**Solution:**

```
<!DOCTYPE html>
<html>
<head>
  <title>Control Transfer</title>
  <link rel="stylesheet" href="pass_control.css">
</head>
<body>
  <h1>Control Transfer</h1>
  <div class="container">
    <div class="output">
      <h2>Include</h2>
      <?php
        echo "Before include statement<br>";
        $included_text = "Included text";
        include('included_text');
        echo "$included_text<br>After include statement";
      ?>
    </div>

    <div class="output">
      <h2>Require</h2>
      <?php
        echo "Before require statement<br>";
        $required_text = "This is required text";
        require('required_text');
        echo "$required_text<br>After require statement";
      ?>
    </div>

    <div class="output">
      <h2>Exit</h2>
      <?php
        echo "Before exit statement<br>";
        exit("Exiting program");
        echo "<br><br> After exit statement ";
      ?>
    </div>
    <div class="output">
      <h2>Die</h2>
      <?php
```

```

Echo"Before die statement<br>";
Die("Exiting program with die");
Echo"<br>After die statement";
?>
</div>
</div>
</body>
</html>
Pass_control.css
Body {
    Font-family:Arial,sans-serif;
    Padding:20px
    background-color: #f5f5f5;}
H1{
    Text-align:center;}
.container{
    Display: flex;
    Justify-content:center;
    Margin-top:20px}
.output{
    Border:2px solid #da5050;
    Padding:10px
    Margin:0 10px;
    Background-colour:beige;
}

```

## Output:

