

## **EXP 9: Sorting an Indexed Array Using Form Input**

### **Aim**

To develop a PHP web application that accepts numbers through a form and sorts them in ascending and descending order.

### **Program**

```
<!DOCTYPE html>
<html>
<head>
    <title>Indexed Array Sorting</title>
</head>
<body>

<h2>Sort Indexed Array</h2>

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

<?php
if (isset($_POST['submit'])) {
    $input = $_POST['numbers'];
    $array = explode(",", $input);

    echo "<h3>Ascending Order:</h3>";
    sort($array);
    foreach ($array as $value) {
        echo trim($value) . " ";
    }

    echo "<h3>Descending Order:</h3>";
    rsort($array);
    foreach ($array as $value) {
        echo trim($value) . " ";
    }
}
?>

</body>
</html>
```

## EXP10:Min, Max, Sum and Average of Array Elements

### Aim

To develop a PHP program using an HTML form that accepts array elements from the user and calculates the **minimum**, **maximum**, **sum**, and **average** of the given elements.

```
<!DOCTYPE html>
<html>
<head>
    <title>Array Operations</title>
</head>
<body>
<h2>Min, Max, Sum and Average of Array Elements</h2>

<form method="post">
    Enter numbers (comma separated):
    <input type="text" name="elements" required>
    <br><br>
    <input type="submit" name="submit" value="Calculate">
</form>

<?php
if (isset($_POST['submit'])) {
    // Step 1: Convert input string to array
    $array = explode(",", $_POST['elements']);

    // Step 2: Remove extra spaces from each element
    $array = array_map('trim', $array);

    // Step 3: Convert array values to integers
    $array = array_map('intval', $array);

    // Step 4: Calculate min, max, sum, and average
    $min = min($array);
    $max = max($array);
    $sum = array_sum($array);
    $avg = $sum / count($array);

    // Display results
    echo "<h3>Results</h3>";
    echo "Minimum: $min <br>";
    echo "Maximum: $max <br>";
    echo "Sum: $sum <br>";
    echo "Average: $avg <br>";
}
?>
</body>
</html>
```

## EXP11:Searching Elements in an Array Using Form

### Aim

To develop a PHP web application that allows the user to add elements to an array and search for an element using a form.

### Program

```
<!DOCTYPE html>
<html>
<head>
    <title>Add and Search Array</title>
</head>
<body>

<h2>Add and Search Elements in Array</h2>
<form method="post">
    Enter elements (comma separated):<br><br>
    <input type="text" name="elements" required><br><br>

    Enter element to search:<br><br>
    <input type="text" name="search" required><br><br>
    <input type="submit" name="submit" value="Submit">
</form>
<?php
if (isset($_POST['submit'])) {

    $array = explode(",", $_POST['elements']);
    $search = trim($_POST['search']);

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

    echo "<h3>Array Elements:</h3>";
    foreach ($array as $value)
    {
        echo trim($value). " ";
    }

    echo "<br><br>";
    if (in_array($search, $array)) {
        echo "<p>Element <b>$search</b> found in the array.</p>";
    } else {
        echo "<p>Element <b>$search</b> not found in the array.</p>";
    }
}
?>
</body>
</html>
```

## EXP 12: Sorting an Associative Array Using Form Input

### Aim

To develop a PHP web application that accepts names and marks through a form and sorts the associative array.

### Program

```
<!DOCTYPE html>
<html>
<head>
    <title>Associative Array Sorting</title>
</head>
<body>

<h2>Sort Associative Array</h2>

<form method="post">
    Enter Names (comma separated):<br><br>
    <input type="text" name="names" required><br><br>

    Enter Marks (comma separated):<br><br>
    <input type="text" name="marks" required><br><br>

    <input type="submit" name="sort" value="Sort">
</form>

<?php
if (isset($_POST['sort'])) {

    $names = explode(",", $_POST['names']);
    $marks = explode(",", $_POST['marks']);

    // Create associative array
    $assocArray = array_combine($names, $marks);

    echo "<h3>Sorted by Marks (Ascending)</h3>";
    asort($assocArray);

    foreach ($assocArray as $name => $mark) {
        echo trim($name) . " : " . trim($mark) . "<br>";
    }
}
?>

</body>
</html>
```