**Tutorial 8 – PHP – part 2**

Repo link: <https://github.com/tinkerness/S7_Web_Programming/tree/main/tutorials/tut8>

Hosted link: <https://tinkerness.github.io/S7_Web_Programming/tutorials/tut8/index.html>

1. **Write a PHP program to store the name and rollno of 10 students in an Associative Array and Use foreach loop to process the array and Perform assort, rsort and ksort in the array. Illustrate with suitable output data.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

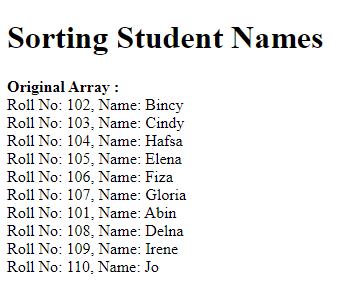
    <title>Sorting Student Names</title>

</head>

<body>

    <h1>Sorting Student Names</h1>

    <?php

    // associative array

    $students = array(

        '102' => 'Bincy',

        '103' => 'Cindy',

        '108' => 'Hafsa',

        '104' => 'Elena',

        '106' => 'Fiza',

        '107' => 'Gloria',

        '101' => 'Abin',

        '108' => 'Delna',

        '109' => 'Irene',

        '110' => 'Jo'

    );

    echo "<b>Original Array :</b><br>";

    foreach ($students as $rollno => $name) {

        echo "Roll No: $rollno, Name: $name <br/>";

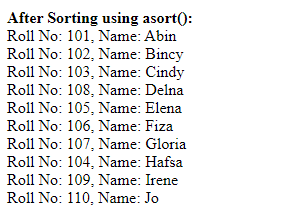
    }

    echo "<br/>";

    ?>

</body>

</html>



    // asort()

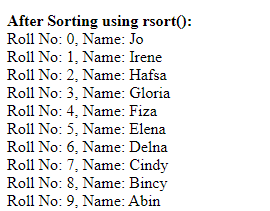
    asort($students);

    echo "<b>After Sorting using asort():</b><br>";

    foreach ($students as $rollno => $name) {

        echo "Roll No: $rollno, Name: $name <br/>";

    }



    // rsort() - reverse sort based on names

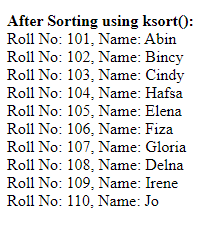
    rsort($students);

    echo "<b>After Sorting using rsort():</b><br>";

    foreach ($students as $rollno => $name) {

        echo "Roll No: $rollno, Name: $name <br/>";

    }

    // ksort() - sort based on keys

    ksort($students);

    echo "<b>After Sorting using ksort():</b><br>";

    foreach ($students as $rollno => $name) {

        echo "Roll No: $rollno, Name: $name <br/>";

    }

1. **Develop a PHP program to print the factorial of a number using a function that accepts number as input and returns the factorial?**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Factorial</title>

</head>

<body>

    <h1>Factorial</h1>

    <form method="post" action="factorial.php">

        Enter a number to find factorial: <input type="text" name="num" />

        <br/><br/>

        <input type="submit" value="Find Factorial" />

        <br/><br/>

    </form>

    <?php

    function calculatefactorial($num) {

        if ($num == 0 || $num == 1) {

            return 1;

        } else {

            return $num \* calculatefactorial($num - 1);

        }

    }

    if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

        if (isset($\_POST["num"])) {

            $number = $\_POST["num"];

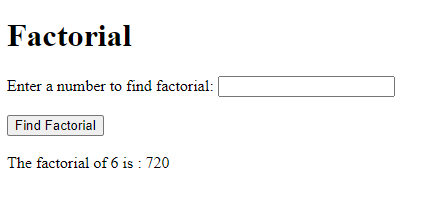
            if (is\_numeric($number)) {

                $factorial = calculatefactorial($number);

                echo "The factorial of $number is : $factorial";

            }

            else {

                echo "Please enter a valid number.";

            }

        }

    }

    ?>

</body>

</html>

1. **Declare an associative array named “items” to store the key-value pairs (“Bread”, 30), (“Butter”, 30), (“Jam”, 35), (“Cheese”, 32).**

**a. Print the array using for each loop**

**b. Sort the array according to values maintaining the key-value relationships and print the sorted key-value pairs.**

<!DOCTYPE html>

<html lang="en">

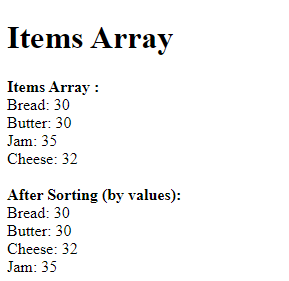
<head>

    <meta charset="UTF-8">

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

    <title>Items Array</title>

</head>

<body>

    <h1>Items Array</h1>

    <?php

    $items = array(

        'Bread' => 30,

        'Butter' => 30,

        'Jam' => 35,

        'Cheese' => 32

    );

    echo "<b>Items Array :</b><br/>";

    foreach ($items as  $product => $price) {

        echo "$product: $price <br/>";

    }

    echo "<br/>";

    asort($items);

    echo "<b>After Sorting (by values):</b><br/>";

    foreach ($items as $product => $price) {

        echo "$product: $price <br/>";

    }

    ?>

</body>

</html>

1. **What is the purpose of the implicit arrays $\_POST and $\_GET in PHP? Consider that a web page displays a form containing two text boxes (named num1 and num2), where the user enters numeric data. Write a PHP script which collects this form data, finds the sum, difference and the product of the two numbers and then displays the same with suitable messages. Assume that the script is to be embedded in the web page specified by the action attribute of the form and that the method used when the form is submitted is GET.**

In PHP, **$\_POST** and **$\_GET** are superglobal arrays that are used to collect form data submitted with the HTTP POST and GET methods, respectively.

* **$\_POST**: Used to collect form data when the HTTP POST method is used. It is commonly used when sensitive information, such as passwords, is being transmitted.
* **$\_GET**: Used to collect form data when the HTTP GET method is used. Data submitted with the GET method is visible in the URL, and it is suitable for non-sensitive data.

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>GET method</title>

</head>

<body>

    <h1>GET method</h1>

    <form method="get" action="get\_method.php">

        <label for="num1">Enter Number 1: </label>

        <input type="text" name="num1" id="num1" required />

        <br/><br/>

        <label for="num2">Enter Number 2:</label>

        <input type="text" name="num2" id="num2" required />

        <br/><br/>

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

        <br/><br/>

    </form>

    <?php

    if ($\_SERVER["REQUEST\_METHOD"] == "GET") {

        $num1 = isset($\_GET['num1']) ? $\_GET['num1'] : 0;

        $num2 = isset($\_GET['num2']) ? $\_GET['num2'] : 0;

        $num1 = is\_numeric($num1) ? $num1 : 0;

        $num2 = is\_numeric($num2) ? $num2 : 0;

        $sum = $num1 + $num2;

        $difference = $num1 - $num2;

        $product = $num1 \* $num2;

        echo "<h2>Results:</h2>";

        echo "<p>Number 1: $num1</p>";

        echo "<p>Number 2: $num2</p>";

        echo "<p>Sum: $sum</p>";

        echo "<p>Difference: $difference</p>";

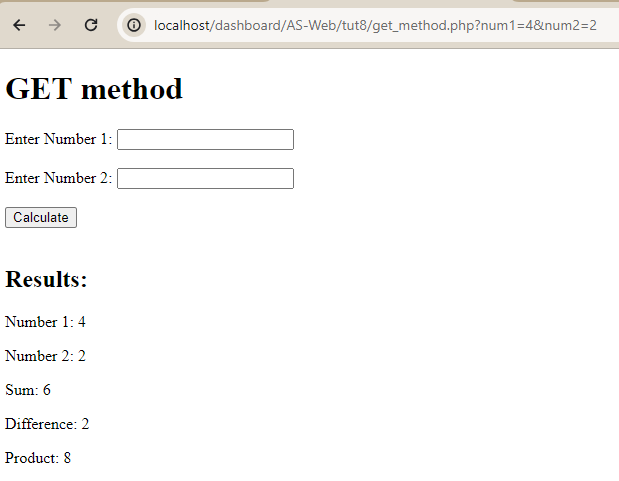
        echo "<p>Product: $product</p>";

    }

    ?>

</body>

</html>



1. **Design an HTML form for entering a number by the user. Write a PHP code to display a message indicating, whether the number is positive or negative, when clicking on the submit button.**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>positive or negative</title>

</head>

<body>

    <h1>positive or negative</h1>

    <form method="post" action="pos\_neg\_num.php">

        Enter a number : <input type="text" name="num" />

        <br/><br/>

        <input type="submit" value="Check" />

        <br/><br/>

    </form>

    <?php

    if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

        if (isset($\_POST["num"])) {

            $number = $\_POST["num"];

            if (is\_numeric($number)) {

                $message = ($number >= 0) ? "The number $number is positive." : "The number $number is negative.";

                echo "$message";

            }

            else {

                echo "Please enter a valid number.";

            }

        }

    }

    ?>

</body>

</html>

