Institute of Computer Technology

B. Tech. Computer Science and Engineering

Semester: IV

Sub: Front End Technologies

Course Code: 2CSE401

Practical Number:5

Objective:

To learn about function (named function, arrow function) and form validation concept.

Practical Definition:

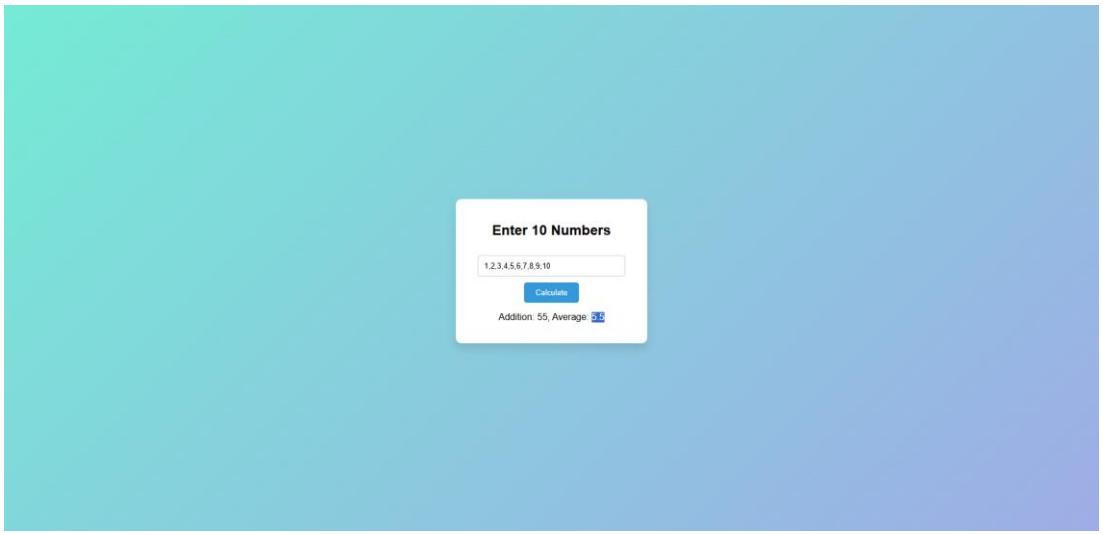
1. Make an appropriate program, where you have to read any random 10 number from the user, and the have to find addition and average of given number by using function with and without parameter.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Addition and Average</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      background: linear-gradient(135deg, #74ebd5, #9face6);
    }
    .container {
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);
      text-align: center;
    }
    input {
```

```
      width: 80%;
      padding: 10px;
      margin: 10px 0;
      border: 1px solid #ccc;
      border-radius: 5px;
    }
    button {
      padding: 10px 20px;
      background: #3498db;
      color: white;
      border: none;
      border-radius: 5px;
      cursor: pointer;
      transition: background 0.3s;
    }
    button:hover {
      background: #2980b9;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>Enter 10 Numbers</h2>
    <input type="text" id="numbers" placeholder="Enter 10 numbers separated by commas">
    <button onclick="calculate()">Calculate</button>
    <p id="result"></p>
  </div>

  <script>
    function calculate() {
      let input = document.getElementById("numbers").value;
      let numbers = input.split(",").map(Number);

      if (numbers.length !== 10) {
        alert("Please enter exactly 10 numbers.");
        return;
      }

      const sum = (nums) => nums.reduce((acc, num) => acc + num, 0);
      const addition = sum(numbers);
      const average = addition / numbers.length;

      document.getElementById("result").innerHTML =
        `Addition: ${addition}, Average: ${average}`;
    }
  </script>
</body>
</html>
```

Output :-



2. Write an appropriate program, where you have to read any one random number between (1 to 9) from the user, and then have to find factorial of 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>Factorial</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      background: linear-gradient(135deg, #fbc2eb, #a6c1ee);
    }
    .container {
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);
      text-align: center;
    }
```

```
    input, button {
      padding: 10px;
      margin: 10px 0;
      border-radius: 5px;
      border: 1px solid #ccc;
    }
    button {
      background: #8e44ad;
      color: white;
      cursor: pointer;
      transition: background 0.3s;
    }
    button:hover {
      background: #732d91;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>Enter a Number (1 to 9)</h2>
    <input type="number" id="number" min="1" max="9">
    <button onclick="findFactorial()">Find Factorial</button>
    <p id="factorialResult"></p>
  </div>

  <script>
    function factorial(num) {
      if (num <= 1) return 1;
      return num * factorial(num - 1);
    }

    function findFactorial() {
      let num = parseInt(document.getElementById("number").value);

      if (isNaN(num) || num < 1 || num > 9) {
        alert("Please enter a number between 1 and 9.");
        return;
      }

      let result = factorial(num);
      document.getElementById("factorialResult").innerHTML =
        `Factorial of ${num} is ${result}`;
    }
  </script>
</body>
</html>
```

Output :-

3. Demonstrate one program, where you have to read any random one three-digit number from the user, and then have to check, whether the given input number is Armstrong or Palindrome number. [Note: Perform this program by using function with parameter, and then, you have to call Armstrong function on button onClick event, and then, have to call palindrome function on ondblClick event.]

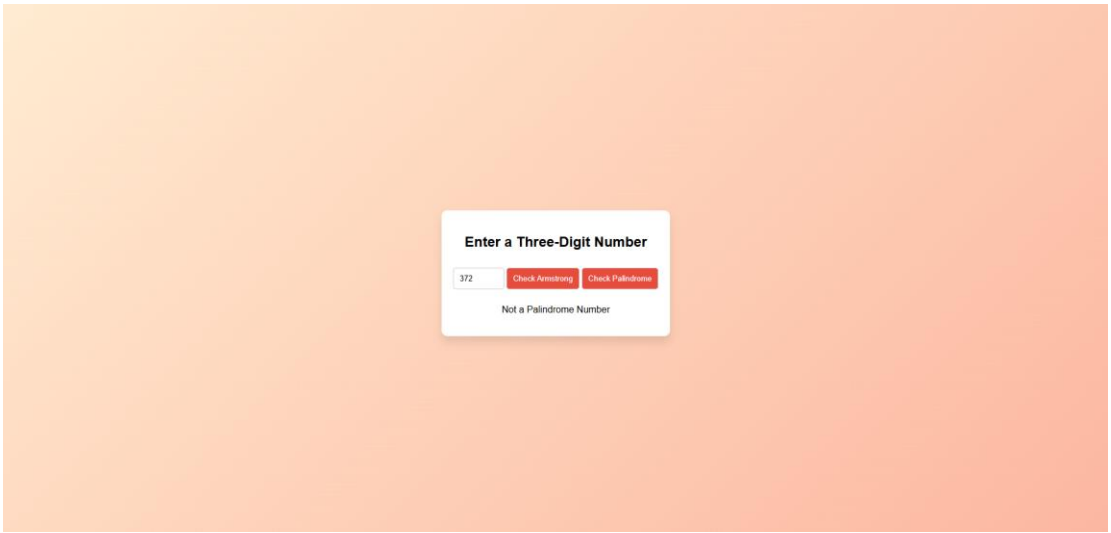
Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Armstrong and Palindrome Check</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
      background: linear-gradient(135deg, #ffecd2, #fcb69f);
    }
    .container {
      background: white;
      padding: 20px;
      border-radius: 10px;
      box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);
```

```
      text-align: center;
    }
    input, button {
      padding: 10px;
      margin: 10px 0;
      border-radius: 5px;
      border: 1px solid #ccc;
    }
    button {
      background: #e74c3c;
      color: white;
      cursor: pointer;
      transition: background 0.3s;
    }
    button:hover {
      background: #c0392b;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>Enter a Three-Digit Number</h2>
    <input type="number" id="threeDigit" min="100" max="999">
    <button onclick="checkArmstrong()">Check Armstrong</button>
    <button ondblclick="checkPalindrome()">Check Palindrome</button>
    <p id="checkResult"></p>
  </div>

  <script>
    const isArmstrong = (num) => {
      let sum = 0;
      let temp = num;
      while (temp > 0) {
        let digit = temp % 10;
        sum += digit ** 3;
        temp = Math.floor(temp / 10);
      }
      return sum === num;
    }

    const isPalindrome = (num) => {
      let str = num.toString();
      return str === str.split("").reverse().join("");
    }

    function checkArmstrong() {
      let num = parseInt(document.getElementById("threeDigit").value);

      if (isNaN(num) || num < 100 || num > 999) {
```

```
            alert("Please enter a three-digit number.");
            return;
        }

        let result = isArmstrong(num) ? "Armstrong Number" : "Not an Armstrong Number";
        document.getElementById("checkResult").innerHTML = result;
    }

    function checkPalindrome() {
        let num = parseInt(document.getElementById("threeDigit").value);

        if (isNaN(num) || num < 100 || num > 999) {
            alert("Please enter a three-digit number.");
            return;
        }

        let result = isPalindrome(num) ? "Palindrome Number" : "Not a Palindrome Number";
        document.getElementById("checkResult").innerHTML = result;
    }
  </script>
</body>
</html>
```

Output :-



4. Design an appropriate employee registration web page, where you have to implement the concept of function, and then have to validate all required field according to their information in the given form.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8" />
 <title>Employee Registration Form</title>
 <style>
  body {
   font-family: Arial, sans-serif;
   margin: 20px;
   background-color: #f9f9f9;
  }
  h1 {
   text-align: center;
   margin-bottom: 20px;
  }
  form {
   width: 95%;
   margin: 0 auto;
   background: #fff;
   padding: 20px;
   border-radius: 8px;
   box-shadow: 0 0 15px rgba(0, 0, 0, 0.1);
  }
  table {
   width: 100%;
   border-collapse: collapse;
  }
  th {
   text-align: left;
  }
  td {
   border: 1px solid #ccc;
   padding: 8px;
   vertical-align: middle;
  }
  td input[type="text"],
  td input[type="number"],
  td input[type="date"],
  td input[type="file"],
  td select {
   width: 98%;
   box-sizing: border-box;
   padding: 5px;
   margin: 0;
  }
  .section-title {
   background: #eee;
```

```
    font-weight: bold;
    text-align: center;
   }
   .full-width-input {
    width: 100% !important;
   }
   .radio-group label {
    margin-right: 10px;
   }
   button {
    display: inline-block;
    padding: 10px 20px;
    background: #3498db;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    margin-top: 10px;
    font-size: 16px;
   }
   button:hover {
    background: #2980b9;
   }
   /* Responsive: for smaller screens, reduce table layout */
   @media (max-width: 768px) {
    table,
    tr,
    td {
     display: block;
     width: 100%;
    }
    td {
     border: none;
     border-bottom: 1px solid #ccc;
    }
    td input,
    td select {
     width: 100% !important;
    }
    .section-title {
     text-align: left;
    }
   }
  </style>
 </head>
 <body>
  <h1>Employee Registration Form</h1>
  <form id="employeeForm">
   <table>
```

```
<!-- Row 1 -->
<tr>
 <td>Name</td>
 <td><input type="text" name="name" /></td>
 <td>Gender</td>
 <td>
  <select name="gender">
   <option value="">Select</option>
   <option>Male</option>
   <option>Female</option>
   <option>Other</option>
  </select>
 </td>
 <td>Date of Birth</td>
 <td><input type="date" name="dob" /></td>
 <td>Photo</td>
 <td><input type="file" name="photo" /></td>
</tr>

<!-- Row 2 -->
<tr>
 <td>Birthplace</td>
 <td><input type="text" name="birthplace" /></td>
 <td>Political Status</td>
 <td><input type="text" name="politicalStatus" /></td>
 <td>Marital Status</td>
 <td>
  <select name="maritalStatus">
   <option value="">Select</option>
   <option>Single</option>
   <option>Married</option>
   <option>Divorced</option>
  </select>
 </td>
 <td>Major</td>
 <td><input type="text" name="major" /></td>
</tr>

<!-- Row 3 -->
<tr>
 <td>Education</td>
 <td><input type="text" name="education" /></td>
 <td>Job Title</td>
 <td><input type="text" name="jobTitle" /></td>
 <td>Height</td>
 <td><input type="number" name="height" /></td>
 <td>ID Number</td>
 <td><input type="text" name="idNumber" /></td>
</tr>
```

```
<!-- Row 4 -->
<tr>
 <td>Graduated School</td>
 <td><input type="text" name="graduatedSchool" /></td>
 <td>Graduation Time</td>
 <td><input type="date" name="graduationTime" /></td>
 <td>Contact Number</td>
 <td><input type="text" name="contactNumber" /></td>
 <td>Telephone Number</td>
 <td><input type="text" name="telephoneNumber" /></td>
</tr>

<!-- Row 5: Address -->
<tr>
 <td>Address</td>
 <td colspan="7"><input type="text" name="address" class="full-width-input" /></td>
</tr>

<!-- Emergency Contact Section -->
<tr>
 <td colspan="8" class="section-title">Emergency Contact</td>
</tr>
<tr>
 <td>Name</td>
 <td><input type="text" name="emergencyName" /></td>
 <td>Relationship</td>
 <td><input type="text" name="emergencyRelationship" /></td>
 <td>Tel</td>
 <td><input type="text" name="emergencyTel" /></td>
 <td colspan="2"></td>
</tr>

<!-- Skills Section -->
<tr>
 <td colspan="8" class="section-title">Skills</td>
</tr>
<tr>
 <td>Language Skills</td>
 <td><input type="text" name="languageSkills" /></td>
 <td>Type</td>
 <td class="radio-group">
  <label><input type="radio" name="type" value="A" />A</label>
  <label><input type="radio" name="type" value="B" />B</label>
  <label><input type="radio" name="type" value="C" />C</label>
  <label><input type="radio" name="type" value="F" />F</label>
 </td>
 <td>Driving License</td>
 <td><input type="text" name="drivingLicense" /></td>
```

```
      <td>Computer Skills</td>
      <td><input type="text" name="computerSkills" /></td>
     </tr>
     <tr>
      <td>Skill Level</td>
      <td><input type="text" name="skillLevel" /></td>
      <td>Date of Initial Certification</td>
      <td><input type="date" name="certDate" /></td>
      <td>Other Skills</td>
      <td colspan="3"><input type="text" name="otherSkills" class="full-width-input" /></td>
     </tr>

     <!-- Family Member Section -->
     <tr>
      <td colspan="8" class="section-title">Family Member</td>
     </tr>
     <tr>
      <td>Name</td>
      <td><input type="text" name="familyName" /></td>
      <td>Relationship</td>
      <td><input type="text" name="familyRelationship" /></td>
      <td>Unit and Position</td>
      <td><input type="text" name="familyUnitPosition" /></td>
      <td>Contact Number</td>
      <td><input type="text" name="familyContact" /></td>
     </tr>

     <!-- Submit Button -->
     <tr>
      <td colspan="8" style="text-align: center;">
       <button type="submit">Submit</button>
      </td>
     </tr>
    </table>
   </form>

<!-- Output Section -->
<div id="output" style="margin-top: 20px; padding: 15px; background: #e8f4ff; border-radius:
8px;">
  <h2>Submitted Information</h2>
  <p><strong>Name:</strong> <span id="outputName"></span></p>
  <p><strong>Gender:</strong> <span id="outputGender"></span></p>
  <p><strong>Date of Birth:</strong> <span id="outputDob"></span></p>
  <p><strong>Contact Number:</strong> <span id="outputContact"></span></p>
</div>

<script>
  document.getElementById('employeeForm').addEventListener('submit', function (e) {
    e.preventDefault();
```

```
  // Get all form data
  const formData = new FormData(this);
  let formObject = {};
  formData.forEach((value, key) => {
   formObject[key] = value;
  });

  // Store all data in localStorage
  localStorage.setItem('employeeData', JSON.stringify(formObject));

  // Show only the required fields in the output section
  document.getElementById('outputName').textContent = formObject.name;
  document.getElementById('outputGender').textContent = formObject.gender;
  document.getElementById('outputDob').textContent = formObject.dob;
  document.getElementById('outputContact').textContent = formObject.contactNumber;
 });

 // On page load, check if there's saved data and display it
 window.addEventListener('load', function() {
  const savedData = JSON.parse(localStorage.getItem('employeeData'));
  if (savedData) {
   document.getElementById('outputName').textContent = savedData.name;
   document.getElementById('outputGender').textContent = savedData.gender;
   document.getElementById('outputDob').textContent = savedData.dob;
   document.getElementById('outputContact').textContent = savedData.contactNumber;
  }
 });
</script>


</body>
</html>
```

Output :-