ASSIGNMENT 1

1. Create an HTML page showing the use of following: Paragraphs, Line breaking, Preserving multiple whitespaces, tabs, linebreaks, Headings, Block quoting, content based tags, Superscript and Subscript, various character entities, horizontal rules.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Example1</title>
    <style>
        /* Center the content */
        .center {
            text-align: center;
        }
    </style>
</head>
<body>
    <div class="center">
        <h1>Hello, welcome to the AWT lab</h1>

        <!-- Paragraphs -->
        <p>This is the 1st paragraph text.</p>
        <p>This is the 2nd paragraph text.<br>Line breaking example.</p>

        <!-- Horizontal Rule -->
        <hr>

        <!-- Block Quoting -->
        <blockquote>
            <p>"The only limit to our realization of tomorrow is our doubts of today." - Franklin D.
Roosevelt</p>
        </blockquote>

        <!-- Headings -->
        <h2>Mathematical Notation</h2>
        <h3>Examples:</h3>
        <h4>s<sup>2</sup> = This is Super Script</h4>
        <h4>s<sub>2</sub> This is Subscript</h4>

        <!-- Preserving Whitespace and Tabs -->
        <pre>
            This    is    text      with      multiple
            spaces and        tabs.
        </pre>

        <!-- Special Characters -->
        <p>Special characters: &amp; &lt; &gt; &copy;</p>

        <!-- Content Based Tags -->
        <address>
            Contact us at: <a href="https://www.vivianserrao.com">www.vivianserrao.com</a>
        </address>
    </div>
</body>
</html>
```

2. Create an HTML page with

- **a. An image**
- **b. Link to another page**
- **c. Link to section of the same page**
- **d. An image as a link to another page**

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Example2</title>
</head>
<body>
    <center>
        <h1>This is Our College</h1>

        <!-- Image as a link to another page -->
        <a href="./exmpl1.html">
            <img
src="https://daijiworld.ap-south-1.linodeobjects.com/iWeb/daijiworld/images3/asha_19621_toyca
thon.jpg" alt="Profile Image" width="300">
        </a>
        <br><br>

        <!-- Link to another page -->
        <a href="https://nmamit.nitte.edu.in" target="_blank">Click here to view Website</a>
        <br><br>

        <!-- Link to a section of the same page -->
        <a href="#new_page">Link to section of the same page</a>
        <br><br>

        <!-- Section to navigate to -->
        <section style="margin-top: 35%;" id="new_page">
            <h2>Second Section</h2>
            <p>This is the second section of the page.</p>
        </section>
    </center>
</body>
</html>
```

3. Create an HTML page for displaying Vision, Mission, PEOs, POs, PSOs of the MCA department as displayed in the lab. Try different types of numbering and bulleting.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Vision Misson</title>
    <style>
        h2
        {
            text-align: center;
        }
        h1{
            text-align: center;
            color: red;
        }
    </style>
</head>
<body>

    <h1>MASTER OF COMPUTER APPLIACTION</h1>
   <hr>
    <h2>VISION</h2>
<p>Equipping students with computing and programming domain
```

```
    expertise with the state of the art technology solutions to enable them
    to meet global professional challenges.</p>
    <h2>MISSION</h2>
    <p>The department strives to create an environment conducive to
        equipping students with teamwork ability, Professional Ethics. Sound
        Technical Knowledge and Skills to Handle Technological Challenges</p>
        <ul>
            <li>Professional Ethics</li>
            <li>Sound Technical Knowledge</li>
            <li>Skills to Handle Technological Challenges</li>
        </ul>
        <h2>Programme Educational Objectives (PEO's)</h2>
        <ul>
            <li>To think critically, work creatively, communicate effectively, and become
                technologically competent.</li>
                <li>To be able to update themselves in areas and technologies relevant to their
career.</li>
                <li>o function in supportive/ leadership roles with ethical responsibilities and high
regard
                    towards societal needs.</li>
                    <li>To develop team work ability, lead initiatives and manage task effectively.</li>
        </ul>
        <h2>Programme Outcomes (POs)</h2>
        <ol>
            <li>Apply knowledge of mathematics, science, and engineering.</li>
            <li>Identify, formulate, research literature, and analyze complex engineering
problems.</li>
            <li>Design a system, component, or process to meet desired needs within realistic
constraints such as economic, environmental, social, political, ethical, health and safety,
manufacturability, and sustainability.</li>
            <li>Design and conduct experiments, as well as analyze and interpret data.</li>
            <li>Apply appropriate techniques, resources, and modern engineering and IT tools to
complex engineering activities.</li>
            <li>Assess societal, health, safety, legal, and cultural issues.</li>
            <li>Understand the impact of engineering solutions in a global, economic,
environmental, and societal context.</li>
            <li>Understand professional and ethical responsibility.</li>
            <li>Function effectively as an individual, member, or leader on multidisciplinary
teams.</li>
            <li>Communicate effectively.</li>
            <li>Recognize the need for and engage in lifelong learning.</li>
            <li>Acknowledge contemporary issues.</li>
        </ol>
        <h2>Programme Specific outcomes (PSOs)</h2>
    <ol>
      <li>Empower students to exemplify themselves in the field of Information Technology, ensuring credibility, integrity, and ethical standards.</li>
      <li>Adopt research practices and trending technologies such as Artificial Intelligence and Machine Learning, Internet of Things, Cloud Computing, and Data Analytics to solve real-world problems.</li>
    </ol>

</body>
</html>
```

4. Create an HTML page with Abbreviations of courses offered at NMAMIT as terms and definitions.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Courses Abbreviations</title>
</head>
<body>
  <center>
    <h1>Courses Offered at NMAMIT</h1>

    <p><strong>B.Tech:</strong> Bachelor of Technology</p>
    <p><strong>M.Tech:</strong> Master of Technology</p>
    <p><strong>MBA:</strong> Master of Business Administration</p>
    <p><strong>M.Sc:</strong> Master of Science</p>
    <p><strong>Ph.D.:</strong> Doctor of Philosophy</p>
    <p><strong>B.Sc:</strong> Bachelor of Science</p>
    <p><strong>B.A.:</strong> Bachelor of Arts</p>
    <p><strong>M.A.:</strong> Master of Arts</p>
    <p><strong>IT:</strong> Information Technology</p>
    <p><strong>CS:</strong> Computer Science</p>
    <p><strong>EC:</strong> Electronics and Communication</p>
    <p><strong>ME:</strong> Mechanical Engineering</p>
  </center>
</body>
</html>
```

5. Create an HTML page with following nesting of lists:

To do List:

1. Shopping

a. Cloths

b. Cell Phone

2. Meeting

a. Friends

• Friend1

• Friend2

b. Teacher

◦ Teacher1

◦ Teacher2

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>To-Do List</title>
</head>
<body>
    <h1>To Do List</h1>
    <ol>
        <li>Shopping
            <ul style="list-style-type: lower-alpha;">
                <li>Cloths</li>
                <li>Cell Phone</li>
            </ul>
        </li>
        <li>Meeting
            <ul style="list-style-type: lower-alpha;">
                <li>Friends
                    <ul>
                        <li>Friend1</li>
                        <li>Friend2</li>
                    </ul>
                </li>
                <li>Teacher
                    <ul>
                        <li>Teacher1</li>
                        <li>Teacher2</li>
                    </ul>
                </li>
            </ul>
        </li>
    </ol>
</body>
</html>
```

6. Display your class time table.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Time Table</title>
    <style>
        table {
            width: 100%;
            border-collapse: collapse;
            margin: 20px 0;
        }
        th, td {
            padding: 10px;
            text-align: center;
        }
        th {
            background-color: #f2f2f2;
        }
        h1{
            text-align: center;
        }

    </style>
</head>
<body>
    <h1>MCA C Section Time Table</h1>
    <table border="1">
        <tr>
            <th>Days</th>
            <th>9.00- 9.50</th>
            <th>9.50-10.40</th>
            <th>11.10-12.05</th>
            <th>12.05-1.00</th>
            <th>1.00-1.55</th>
            <th>1.55-2.50</th>
            <th>2.50-3.40</th>
            <th>3.40-4.30</th>
        </tr>
        <tr>
            <td>Monday</td>
            <td>MC & AD</td>
            <td>AIML</td>
            <td colspan="2">Mini Project</td>
            <td rowspan="5">Lunch Break</td>
            <td>CC & BDA</td>
            <td>Placement Training</td>
            <td>-</td>
        </tr>
        <tr>
            <td>Tuesday</td>
            <td>AIML</td>
            <td>AWT</td>
            <td>Placement Training</td>
            <td>CC & BDA</td>
            <td colspan="3">LAB</td>
        </tr>
        <tr>
            <td>Wednesday</td>
            <td>AWT</td>
            <td>AIML</td>
            <td>Placement Training</td>
            <td>BCT/CS</td>
            <td>MC & AD</td>
            <td colspan="2">-</td>
        </tr>
        <tr>
            <td>Thursday</td>
            <td>CC & BDA</td>
            <td>AWT</td>
            <td>BCT/CS</td>
            <td>AIML</td>
            <td colspan="3">LAB</td>
        </tr>
        <tr>
            <td>Friday</td>
            <td>BCT/CS</td>
            <td>MC & AD</td>
```

```
            <td colspan="2">Mini Project</td>
            <td>AWT</td>
            <td colspan="2">Placement Training</td>
        </tr>
        <!-- <tr>
            <td>Saturday</td>
        </tr> -->
    </table>
</body>
</html>
```

7. Design a student registration form for student's enrolment to the course which collects personal and previous educational details. Include form elements: label, input, select, text area, fieldset, legend, datalist.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Student Registration Form</title>
    <style>
        .registration_form {
            width: 60%;
            margin: 0 auto;
            padding: 20px;
            border: 1px solid #ccc;
        }
        h1 {
            margin-bottom: 20px;
        }
        label {
            display: block;
            margin-top: 10px;
        }
        input[type="text"],textarea,select {
            width: 100%;
            padding: 8px;
        }
        input[type="submit"] {
            margin-top: 20px;
            padding: 10px 20px;
            background-color: #4CAF50;
        }
    </style>
</head>
<body>
    <form class="registration_form">
        <h1 style="text-align: center;">Student Registration Form</h1>

        <fieldset>
            <legend>Registration Form</legend>

            <label for="student_name">Student Name:</label>
            <input type="text" placeholder="Enter Student Name" required>

            <label for="father_name">Father's Name:</label>
            <input type="text" placeholder="Enter Father's Name" required>

            <label>Gender:</label>
            <input type="radio" name="gender" value="male">Male</input>
            <input type="radio" name="gender" value="female">Female</input>

            <label for="address">Address:</label>
            <textarea  rows="4" placeholder="Enter Address" required></textarea>

            <label for="previous_course">Previous Course:</label>
            <input type="text" placeholder="Enter Previous School" list="Course">
            <datalist id="Course">
                <option value="BCA">
                <option value="BSc">
                <option value="Others">
            </datalist>

            <label for="grade">Grade Achieved:</label>
            <select id="grade" name="grade" required>
                <option value="" disabled selected>Select Grade</option>
                <option value="A">A</option>
                <option value="B">B</option>
                <option value="C">C</option>
                <option value="D">D</option>
                <option value="E">E</option>
            </select>
        </fieldset>

        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

ASSIGNMENT 2

1. Create a web page with different levels of stylesheets. Also show the priority applied in these levels of stylesheets.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Stylesheet Example</title>
    <!-- External stylesheet -->
    <link rel="stylesheet" href="styles.css">
    <style>
        /* Internal stylesheet */
        .internal-style {
            color: green;
            font-size: 20px;
        }
        .background {
            background-color: lightblue;
        }
    </style>
    <style>
        /* Another internal stylesheet (later in the head) */
        .internal-style {
            color: darkgreen;
        }
    </style>
</head>
<body>
    <h1 class="external-style">This is a heading with external stylesheet</h1>
    <p class="internal-style background">This paragraph has internal and background styles.</p>
    <p style="color: red; font-size: 24px;">This paragraph has inline styles, which override internal and external styles.</p>
```

```
</body>
</html>
```

Style.css

```
/* external stylesheet */
.external-style {
    color: blue;
    font-size: 28px;
}
```

2. Create a page which shows the usage of different selectors for specifying style rules.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Selector</title>
    <style>
        /* Basedon element */
        p{
            color:blue;
        }
        /* Baased on ID */
        #div-paragraph
        {
            color: brown;
        }
        /* Based on class */
        .listing
        {
            color: darkgreen;
        }
        /* Based on Descendent */
        div p
        {
            color: gold;
        }
        /* Based on Attribute */
        [type="submit"] {
            color: purple;
        }
        /* Based on Child */
        ul>li{
            color:gray;
        }
        /* Based on Psedu-class  */
        a:hover
        {
            color: red;
        }

    </style>
</head>
<body>
    <h1>Selectors Demonstration</h1>

    <p>This is a text to demonstrate element selector</p>

    <div>
        <p id="div-paragraph">This is a text to demonstrate id selector</p>
    </div>

    <a class="listing" href="exmpl1.html">This is the example for class selector</a>

    <ul>
        <li>List item 1</li>
        <li>List item 2</li>
    </ul>

    <div>
        <p>This is the text to demonstrate Descendant selector.</p>
    </div>

    <input type="submit" value="Atribute Selector">
</body>
</html>
```

3. Create a page which includes application of font and text related style properties.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Font and Text</title>
    <style>
        h1{
            text-align: center;
            letter-spacing: 5px;
        }
        p{
            font-size: 20px;
        }
        .f1
        {
            font-family: 'Courier New', Courier, monospace;
            font-style: italic;
        }
        .f2
        {
            font-family:'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            font-size: 40px;
            font-weight: bold;
        }
        .f3
        {
            text-decoration: underline;
            text-transform: uppercase;
        }
    </style>
</head>
<body>
    <h1>Heading in center with letter spacing 5px</h1>
    <p>This is the font with 20px</p>
    <p class="f1">This is the Text with Courier New font and italic </p>
    <p class="f2">This is the Text with Segoe UI font and size is 40px with bold </p>
    <p class="f3">This is text with underline and transformed to uppercase </p>
</body>
</html>
```

4. Create a page which includes application of list related style properties.

Code:

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

```
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>List Styles</title>
    </head>
    <body>
        <h1>List Styles Example</h1>
        <h2>Unordered List</h2>
        <!-- By defualt it is Disc (list-style-type: disc;) -->
        <ul>
            <li>Round Bullet 1</li>
            <li>Round Bullet 4</li>
            <li>Round Bullet 3</li>
        </ul>
        <ul style="list-style-type: square;">
            <li>Round Bullet 1</li>
            <li>Round Bullet 4</li>
            <li>Round Bullet 3</li>
        </ul>
        <ul style="list-style-type: circle;">
            <li>Circle Bullet 1</li>
            <li>Circle Bullet 4</li>
            <li>Circle Bullet 3</li>
        </ul>
        <h2>Ordered List</h2>
        <ol>
            <li>Number Bullet 1</li>
            <li>Number Bullet 4</li>
            <li>Number Bullet 3</li>
        </ol>
        <ol style="list-style-type: lower-roman;">
            <li>Roman Bullet 1</li>
            <li>Roman Bullet 4</li>
            <li>Roman Bullet 3</li>
        </ol>
        <ul style="list-style-type: lower-alpha;">
            <li>Alphabet Bullet 1</li>
            <li>Alphabet Bullet 4</li>
            <li>Alphabet Bullet 3</li>
        </ul>
        <h2>Nested List</h2>
        <ul>
            <li>Parent List 1</li>
            <ul>
                <li>Child Item 1</li>
                <li>Child Item 2</li>
            </ul>
            <li>Parent List 2</li>
            <ul>
                <li>Child Item 1</li>
                <li>Child Item 2</li>
            </ul>
        </ul>
    </body>
    </html>
```

5. Show the use of Box Model properties.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Box Model Example</title>
    <style>
        .box {
            width: 300px;
            margin: 20px auto;
            padding: 20px;
            border: 5px solid #333;
            background-color: #f7e6b9;
            text-align: center;
        }
        .box p {
            padding: 10px;
            background-color: #e0e0e0;
        }
    </style>
</head>
<body>
    <div class="box">
        <p>This is a demonstration of the CSS Box Model.</p>
    </div>
</body>
</html>
```

6. Show the use of div and span tags.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Simple div and span Example</title>
    <style>
        .container {
            width: 80%;
            margin: 0 auto;
            padding: 10px;
            border: 1px solid #000000;
        }
        .box {
            margin-bottom: 20px;
            padding: 10px;
            border: 1px solid #ddd;
        }
        .highlight {
            color: red;
            font-weight: bold;
        }
        .inline-text {
            color: blue;
            padding: 2px 4px;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Using div and span Tags</h1>

        <div class="box">
            <h2>Example of div</h2>
            <p>The div element is a block-level container that can be used to group content and apply styles. Here's an example:</p>
            <div class="highlight">
                This is a styled <code>div</code> element.
            </div>
        </div>
    <div class="box">
      <h2>Example of span</h2>
      <p>The span element is an inline container used to style small sections of content.
Here's an example:</p>
      <p>Inline text with <span class="inline-text">highlighted part</span> using
<code>span</code> element.</p>
    </div>
  </div>
</body>
</html>
```

ASSIGNMENT 3

1. **Write a JavaScript to show the implicit and explicit type conversion.**

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Type Conversion Example</title>
</head>
<body>
  <h2>Implicit Conversion</h2>
  <div id="implicit-output"></div>
  <div id="implicit-output"></div>
  <h2>Emplicit Conversion</h2>
  <div id="explicit-output"></div>
  <script>
    const a=10;
    const b="20";
    const add= a + b;
    const sub=b-a
      document.getElementById('implicit-output').textContent = "Addition of two numbers:
"+add+" and Subtraction of two numbers is :"+sub;
    const s1="32"
    const s2= parseInt(s1)
    const s3= String(s2)
      document.getElementById('explicit-output').textContent = "First Number is " +s2;

  </script>
</body>
```

```
</html>
```

2. Show the use of String Object’s properties and methods.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>String Object Properties</title>
</head>
<body>
   <div id="output"></div>
   <script>
      const output =document.getElementById("output");
      const str="Welcome To AWT Lab";
      output.innerHTML +="Original String is: "+str+"<br>";
      output.innerHTML += "Character at index 6: " + str.charAt(6) + "<br>";
      output.innerHTML += "Unicode code at index 6: " + str.charCodeAt(6) + "<br>";
      output.innerHTML += "Concatenated String: " + str.concat(" How are you?") + "<br>";
      output.innerHTML += "Includes 'World': " + str.includes("World") + "<br>";
      output.innerHTML += "Index of 'World': " + str.indexOf("World") + "<br>";
      output.innerHTML += "Last index of 'o': " + str.lastIndexOf("o") + "<br>";
      output.innerHTML += "Match with regex 'World': " + str.match(/World/) + "<br>";
      output.innerHTML += "Replace 'World' with 'Universe': " + str.replace("World", "Universe")
+ "<br>";
      output.innerHTML += "Slice from index 6 to 11: " + str.slice(6, 11) + "<br>";
      output.innerHTML += "Split by space: " + str.split(" ")[1] + "<br>";
      output.innerHTML += "To lower case: " + str.toLowerCase() + "<br>";
      output.innerHTML += "To upper case: " + str.toUpperCase() + "<br>";
      output.innerHTML += "Trimmed (removes extra spaces): '" + str.trim() + "'<br>";
   </script>
</body>
</html>
```

3.Show the use of important methods of Date Object.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <title>Date Methods</title>
</head>
<body>
    <p id="output"></p>
    <script>
        const output = document.getElementById("output");
        const date = new Date();

        output.innerHTML += "<h2>Current Date and Time</h2>";
        output.innerHTML += "Date: " + date.toString() + "<br>";

        output.innerHTML += "<h2>Get Methods</h2>";
        output.innerHTML += "Year: " + date.getFullYear() + "<br>";
        output.innerHTML += "Month (0-11): " + date.getMonth() + "<br>";
        output.innerHTML += "Day of Month: " + date.getDate() + "<br>";
        output.innerHTML += "Hours (0-23): " + date.getHours() + "<br>";
        output.innerHTML += "Minutes (0-59): " + date.getMinutes() + "<br>";
        output.innerHTML += "Seconds (0-59): " + date.getSeconds() + "<br>";

        // Set methods (example)
        date.setFullYear(2025);
        date.setMonth(11);
        date.setDate(25);
        date.setHours(15);
        date.setMinutes(30);
        date.setSeconds(45);

        output.innerHTML += "<h2>Adjusted Date</h2>";
        output.innerHTML += "Adjusted Date: " + date.toString() + "<br>";
    </script>
</body>
</html>
```

4.Take required inputs from the keyboard and display the roots of quadratic equation.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Quadratic Equation Solver</title>
</head>
<body>
```

```
  <h1>Quadratic Equation Solver</h1>
  <p>
    a: <input type="number" id="a"><br><br>
    b: <input type="number" id="b"><br><br>
    c: <input type="number" id="c"><br><br>
    <button onclick="solveQuadratic()">Solve</button>
  </p>
  <p id="result"></p>
  <script>
    function solveQuadratic() {
      // Get values from the input fields
      const a= parseFloat(document.getElementById('a').value);
      const b= parseFloat(document.getElementById('b').value);
      const c= parseFloat(document.getElementById('c').value);

    if(a==0) {
      document.getElementById("result").innerText= "The value of 'a' cannot be zero.";
    return;
  }
  const discriminant= b *b - 4 * a * c;
  let resultText;
  if (discriminant > 0) {
        // Two real and distinct roots
        const root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
        const root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
        resultText = `The roots are real and distinct. Root 1: ${root1}, Root 2: ${root2}`;
      } else if (discriminant === 0) {
        // One real root (repeated)
        const root = -b / (2 * a);
        resultText = `The root is real and repeated. Root: ${root}`;
      } else {
        // Complex roots
        const realPart = -b / (2 * a);
        const imaginaryPart = Math.sqrt(-discriminant) / (2 * a);
        resultText = `The roots are complex. Root 1: ${realPart} + ${imaginaryPart}i, Root 2:
${realPart} - ${imaginaryPart}i`;
      }

      // Display the result
      document.getElementById('result').innerText = resultText;
    }
  </script>
</body>
</html>
```

5. Show the use of strict equality and loose equality operators in JavaScript.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Equality Operators in JavaScript</title>
</head>
<body>
    <h1>Equality Operators in JavaScript</h1>
    <p id="output"></p>

    <script>
        const output = document.getElementById("output");

        const a = 5;
        const b = '5';
        const c = 10;
        const d = '10';

        // Strict equality (===)
        const strictEquality1 = (a === b); // false, because 'a' is a number and 'b' is a string
        const strictEquality2 = (c === d); // false, because 'c' is a number and 'd' is a string

        // Loose equality (==)
        const looseEquality1 = (a == b); // true, because '5' (string) is coerced to number 5
        const looseEquality2 = (c == d); // false, because '10' (string) is not coerced to number 10

        // Display results
        output.innerHTML = `
            <h2>Strict Equality (===)</h2>
            <p>${a} === ${b} : ${strictEquality1}</p>
            <p>${c} === ${d} : ${strictEquality2}</p>

            <h2>Loose Equality (==)</h2>
            <p>${a} == ${b} : ${looseEquality1}</p>
            <p>${c} == ${d} : ${looseEquality2}</p>
        `;
    </script>
</body>
</html>
```

6. Write a JavaScript to find the largest of three inputted numbers.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Find the Largest Number </title>
</head>
<body>
    <h1>Find the Largest of Three Numbers</h1>
    <p>
        Number 1: <input type="number" id="num1"><br><br>
        Number 2: <input type="number" id="num2"><br><br>
        Number 3: <input type="number" id="num3"><br><br>
        <button onclick="findLargest()">Find Largest</button>
    </p>
    <p id="result"></p>
    <script>
        function findLargest() {
            // Get values from the input fields
            const num1 = parseFloat(document.getElementById('num1').value);
            const num2 = parseFloat(document.getElementById('num2').value);
            const num3 = parseFloat(document.getElementById('num3').value);

            const largest = Math.max(num1, num2, num3);
            document.getElementById('result').innerText = `The largest number is: ${largest}`;
        }
    </script>
</body>
</html>
```

7. Find the day of the week and accordingly show the menu of food items in table using switch statement. Apply different table border size for different tables.

Code:

```
<!DOCTYPE html>
<html>
<head>
    <title>Food Menu for Today</title>
    <style>
        table {
            border-collapse: collapse;
            margin: 10px;
        }

        table, th, td {
            border: 1px solid black;
        }

        th, td {
            padding: 8px;
        }
    </style>
</head>
<body>
    <script>
        // Get the current date and day of the week
        const currentDate = new Date();
        const currentDay = currentDate.getDay(); // 0 = Sunday, 1 = Monday, ..., 6 = Saturday

        let menuItems;
        switch (currentDay) {
            case 0:
                menuItems = ["Sunday Special: Biryani", "Raita", "Naan"];
                break;
            case 1:
                menuItems = ["Monday Menu: Fried Rice", "Spring Rolls", "Sweet and Sour Chicken"];
                break;
            case 2:
                menuItems = ["Tuesday Treat: Tacos", "Guacamole", "Nachos"];
                break;
            case 3:
                menuItems = ["Wednesday Delight: Pizza", "Garlic Bread", "Caesar Salad"];
                break;
            case 4:
                menuItems = ["Thursday Feast: Sushi", "Miso Soup", "Tempura"];
                break;
            case 5:
                menuItems = ["Friday Favorites: Burgers", "French Fries", "Milkshake"];
                break;
            case 6:
                menuItems = ["Saturday Special: BBQ Ribs", "Coleslaw", "Cornbread"];
                break;
            default:
                menuItems = ["No data for today"];
        }

        // Create and display a table for the menu items using document.write
        document.write("<table>");

        for (let i = 0; i < menuItems.length; i++) {
            document.write("<tr><td>" + menuItems[i] + "</td></tr>");
        }

        document.write("</table>");

        // Apply different table border size for different days
        if (currentDay === 0 || currentDay === 6) {
            document.write("<style>table { border: 2px solid black; }</style>");
        } else {
            document.write("<style>table { border: 1px solid black; }</style>");
        }
    </script>
</body>
</html>
```

8. Write a JavaScript to find the sum of first n natural numbers using while and for loop.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sum of First n Natural Numbers</title>
</head>
<body>
    <h1>Sum of First n Natural Numbers</h1>
    <label for="number">Enter a number for While Loop: </label>
    <input type="number" id="number" min="1"><br><br>
    <button onclick="calculateSumWhile()">Sum Using While Loop</button>
    <button onclick="calculateSumFor()">Sum Using For Loop</button>
    <p id="resultwhile"></p>
    <p id="resultfor"></p>
    <script>
```

```
        function calculateSumWhile() {
            const n = parseInt(document.getElementById('number').value);
            let sum = 0;
            let i = 1;
            while (i <= n) {
                sum += i;
                i++;
            }
            document.getElementById('resultwhile').innerText = `Sum  using while loop is: ${sum}`;
        }

        function calculateSumFor() {
            const n = parseInt(document.getElementById('number').value);
            let sum = 0;
            for (let i = 1; i <= n; i++) {
                sum += i;
            }
            document.getElementById('resultfor').innerText = `Sum using for loop is: ${sum}`;
        }
    </script>
</body>
</html>
```

9. Show the use of array methods in JavaScript.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Array Methods</title>
</head>
<body>
    <h1>Array Methods</h1>
    <p id="output"></p>
    <script>
        const array = [1, 2, 3, 4, 5];
        const outputElement = document.getElementById('output');
        let outputText = '';

        outputText += '<b>For Each()</b><br>';
        array.forEach((value, index) => {
            outputText += `Index: ${index} Value is: ${value}<br>`;
        });

        outputText += '<b>Map function</b><br>';
        const mappedArray = array.map(value => value * 2);
        outputText += `Mapped Array (values * 2): ${mappedArray}<br>`;

        outputText += '<b>Filter function</b><br>';
        outputText += `Filter Array (values > 3): ` + array.filter(value => value > 3);

        outputText += '<br><b>Find function</b><br>';
        // Returns the first element that matches the condition
        outputText += `Value found: ` + array.find(value => value == 4);

        outputText += '<br><b>Some function</b><br>';
        outputText += `Array contains an even number: ` + array.some(value => value % 2 == 0);

        outputText += '<br><b>Every function</b><br>';
        outputText += `All Array elements are greater than 2: ` + array.every(value => value > 2);

        outputText += '<br><b>Reduce function</b><br>';
        outputText += `Sum of all values: ` + array.reduce((acc, value) => acc + value, 0);

        outputText += '<br><b>Index function</b><br>';
        outputText += `Index of Element 4: ` + array.indexOf(4);

        outputText += '<br><b>Includes function</b><br>';
        outputText += `Array includes value 4: ` + array.includes(4);

        outputText += '<br><b>Slice function</b><br>';
        outputText += `Array slice from index 1 to 4: ` + array.slice(1, 4);

        outputElement.innerHTML = outputText;
    </script>
</body>
</html>
```

10. Write a JavaScript that uses function to find the sum 3 of inputted numbers.

Code:

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

```
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sum of 3 Numbers</title>
</head>
<body>
    <h1>Sum of Three Numbers</h1>
    <label for="num1">Number 1:</label>
    <input type="number" id="num1" required><br><br>

    <label for="num2">Number 2:</label>
    <input type="number" id="num2" required><br><br>

    <label for="num3">Number 3:</label>
    <input type="number" id="num3" required><br><br>

    <button onclick="calculateSum()">Calculate Sum</button>
    <p id="result"></p>
    <script>
        function calculateSum()
        {
            const num1 = parseInt(document.getElementById('num1').value);
            const num2 = parseInt(document.getElementById('num2').value);
            const num3 = parseInt(document.getElementById('num3').value);

            if (isNaN(num1) || isNaN(num2) || isNaN(num3)) {
                document.getElementById('result').innerText = 'Please enter valid numbers in all
fields.';
                return;
            }

            const sum = num1 + num2 + num3;

            document.getElementById('result').innerText = `The sum of ${num1}, ${num2}, and
${num3} is: ${sum}`;
        }
    </script>
</body>
</html>
```

11. Write a JavaScript to show the scope of variables.

Code:

```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Variable Scope Demo</title>
</head>
<body>
    <h1>Variable Scope Demonstration</h1>

    <p id="output" class="output"></p>

    <script>
        // Global Scope
        var globalVar = "I am a global variable";
        showScopes();

        function showScopes() {
            // Function Scope
            var functionVar = "I am a function variable";

            // Block Scope
            {
                let blockLet = "I am a block scoped variable (let)";
                var blockVar = "I am a block scoped variable (var)";

                document.getElementById('output').innerHTML =
                    `Inside Block: ${blockLet} <br> ${blockVar}`;
            }

            document.getElementById('output').innerHTML +=
                `<br>Inside Function: ${functionVar} <br> ${blockVar}`;

            document.getElementById('output').innerHTML +=
                `<br>Global: ${globalVar}`;
        }
    </script>
</body>
</html>
```

Assignment - 4

1. Create a basic HTML5 webpage structure. Include a header, navigation menu, main content section, and a footer. Use semantic HTML5 tags like `<header>`, `<nav>`, `<main>`, and `<footer>`.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>webpage structure</title>
   <style>
   body
   {
      text-align: center;
   }
      header
      {
         font-weight: bold;
         font-size: 25px;
      }
      nav ul{
         display: flex;
         justify-content: center;
         list-style: none;
      }
      nav li{
         padding: 10px 20px;
         font-weight: bold;
      }
      footer{
         bottom:0;
         left: 40%;
         position: fixed;
      }

   </style>
</head>
<body>
   <header>Welcome to My Portfolio</header>
   <nav>
```

```
        <ul>
            <li><a href="#header">About</a></li>
            <li><a href="#main">Education</a></li>
            <li><a href="#footer">Contact</a></li>
        </ul>
    </nav>
    <main>
        <h2 style="color: red;">Hey folks, I'm Vivian Serrao</h2>
        <h3>Tech enthusiast with a passion for learning, eager to contribute coding skills to innovative projects..</h3>
    </main>
    <footer>
        <p>Want to contact me?</p>
        <p>Reach out via email: vvnserrao2002@gmail.com</p>
        <h4>&copy 2024 Vivian Serrao. All rights reserved.</h4>
    </footer>
</body>
</html>
```

2. Design a registration form using HTML5 that includes the following fields: Full Name, Email, Password, Date of Birth, Gender (using radio buttons), and a "Submit" button. Use appropriate HTML5 form elements such as email, password, and date input types.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>
    <style>
        label
        {
            display: block;
            flex-direction: column;
            margin: auto;
            line-height: 43px;
        }
        input[type="text"],
        input[type="email"],
        input[type="password"],
        input[type="date"] {
            width: 100%;
            padding: 10px;
            box-sizing: border-box;
        }
        button {
            background-color: green;
            font-size: 16px;
            padding: 10px;
        }
    </style>
</head>
<body>
    <h1>Student Registration Form</h1>
    <form class="form1">
        <fieldset>
            <legend>Registration Form</legend>
        <label>Full Name:
        <input type="text" id="full_name" placeholder="Enter Full name" required></label>
        <label>Email:
        <input type="email" id="email" placeholder="Enter Full Email" required></label>
        <label>Password:
        <input type="password" id="password" placeholder="Enter Full Password"
required></label>
        <label>date of Birth:
        <input type="date" id="date" required></label>
        <label>Gender:
        <input type="radio" value="male" name="gender">Male
            <input type="radio" value="female" name="gender">Female
        </label>
        <button type="submit">Submit</button>
    </fieldset>
    </form>
</body>
</html>
```

3. Create a webpage that embeds both a video and an audio file using HTML5. Add controls for play, pause, and volume adjustment. Use the `<video>` and `<audio>` tags.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>embedding Media</title>
</head>
```

```
<body>
  <h1>Embedding Media</h1>
  <h2>Video Example</h2>
  <video controls style="width: 600px;">
    <source src="video.mp4" type="video/mp4" width="300px">
  </video>
  <h2>Audio Example</h2>
  <audio controls>
    <source src="audio.mp3" type="audio/mpeg">
    Your browser does not support the audio tag.
  </audio>
</body>
</html>
```

4. Design a responsive webpage using HTML5 and CSS. The webpage should have a header, a navigation bar, a content area with two columns (main content and a sidebar), and a footer. Make sure the layout adapts to different screen sizes (mobile, tablet, desktop).

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Responsive Webpage</title>
  <style>
    header {
      text-align: center;
    }
    nav ul {
      list-style: none;
      display: flex;
      justify-content: center;
      background-color: #333;
    }
    nav li {
      padding: 10px 20px;
    }
    nav a {
      color: white;
      text-decoration: none;
    }
    .container {
            display: flex;
            flex-wrap: wrap;
            padding: 20px;
        }
        main {
            flex: 2;
            padding: 10px;
        }
        aside {
            flex: 1;
            padding: 10px;
            background-color: #f4f4f4;
        }
        footer {
            text-align: center;
            background-color: chocolate;
            padding: 5px;
        }
        @media (max-width: 768px) {
            nav ul {
                flex-direction: column;
                align-items: center;
                padding: 0;
            }
            .container {
                flex-direction: column;
            }
        }
    </style>
</head>
<body>
    <header>
        <h1>Welcome to the Responsive Webpage</h1>
    </header>
    <nav>
        <ul>
            <li><a href="#">Home</a></li>
            <li><a href="#">About</a></li>
            <li><a href="#">Contact</a></li>
        </ul>
    </nav>
    <div class="container">
        <main>
            <h2>Main Content</h2>
            <p>This is the main content area.</p>
        </main>
        <aside>
            <h2>Sidebar</h2>
            <p>This is the sidebar content.</p>
        </aside>
    </div>
    <footer>
        <p>&copy; 2024 All rights reserved.</p>
    </footer>
</body>
</html>
```

5. Create a simple drawing on an HTML5 canvas element. Draw a rectangle, circle, and a line using JavaScript. Use the <canvas> tag and the associated JavaScript API.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Canvas Drawing</title>
    <style>
        canvas {
            border: 2px solid black;
        }
    </style>
</head>
<body>
    <h1>Simple Drawing on Canvas</h1>
    <canvas id="mycanvas" width="400" height="300"></canvas>
    <script>
        const canvas= document.getElementById('mycanvas');
        const ctx = canvas.getContext('2d');

         // Draw a rectangle
         ctx.fillStyle = 'lightblue';
        ctx.fillRect(50, 50, 150, 100);

        // Draw a circle
        ctx.beginPath();
        ctx.arc(300, 100, 50, 0, Math.PI * 2, true);
        ctx.fillStyle = 'lightgreen';
        ctx.fill();
        ctx.closePath();

        // Draw a line
        ctx.beginPath();
        ctx.moveTo(50, 200);
        ctx.lineTo(350, 200);
        ctx.strokeStyle = 'red';
        ctx.lineWidth = 5;
        ctx.stroke();
        ctx.closePath();
    </script>
</body>
</html>
```

Assignment - 5

1.Write a JavaScript to display the table of numbers from 5 to 15 and their squares and cubes using alert.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Number Table with Squares and Cubes</title>
</head>
<body>
    <h1>Number Table with Squares and Cubes</h1>
    <button onclick="displayTable()">Show Table</button>
    <script>
        function displayTable() {
            let msg = '';
            for (let i = 5; i <= 15; i++) {
                let square = i * i;
                let cube = i * i * i;
                msg += `Number: ${i} | Square: ${square} | Cube: ${cube}\n`;
            }
            alert(msg);
        }
    </script>
</body>
```

```
</html>
```

2. Write a script to input a number n that is the number of the Fibonacci numbers required as output and generate the Fibonacci series accordingly.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Fibonacci Series</title>
</head>
<body>
    <h1>Fibonacci Series Generator</h1>
    <label for="number">Enter the number of Fibonacci numbers:</label>
    <input type="number" id="number" min="1" placeholder="Enter a positive integer"><br><br>
    <button onclick="generateFibonacci()">Generate</button>
    <div id="output"></div>
    <script>
        function generateFibonacci() {
            n = parseInt(document.getElementById('number').value);

            if (isNaN(n) || n <= 0) {
                document.getElementById('output').innerHTML=`<br>Please Enter a valid positive
integer`;
                return;
            }
            let fib =[];
            let f1 = 0, f2 = 1;
            for (let i=0; i < n;i++)
            {
                fib.push(f1);
                f3= f1 + f2 ;
                f1 = f2;
                f2 = f3;
            }
        document.getElementById('output').innerHTML = `<br>Fibonacci Series : `+ fib;
        }
    </script>

</body>
</html>
```

3. Write a JavaScript function which receives a number as its parameter and returns the number with its digits in reverse order.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Reverse a Number</title>
</head>
<body>
    <h1>Reverse Digits</h1>
    <label for="number">Enter the number :</label>
    <input type="number" id="number" min="1" placeholder="Enter a positive integer"><br><br>
    <button onclick="callReverse()">Reverse Digits</button>
    <div id="result"></div>
    <script>
        function callReverse() {
            let n = parseInt(document.getElementById('number').value);
            if (isNaN(n) || n < 0) {
                document.getElementById('result').innerHTML = '<br>Please enter a valid positive
integer.';
                return;
            }
            reversenumber(n);
        }

        function reversenumber(n) {
            let numstr = n.toString();
            let revstr = numstr.split('').reverse().join('');
            let revnum = parseInt(revstr);
            document.getElementById('result').innerHTML = '<br>Reversed Number: ' + revnum;
        }
    </script>
</body>
</html>
```

4. Input a line of text from prompt and display the words of input text in alphabetical order.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sort Words Alphabetically</title>
</head>
<body>
    <h1>Sort Words Alphabetically</h1>
    <label for="textInput">Enter a line of text:</label>
    <input type="text" id="textInput" placeholder="Enter text here"><br><br>
    <button onclick="sortWords()">Sort Words</button>
    <div id="result"></div>
    <script>
        function sortWords() {
            let text = document.getElementById('textInput').value;
            let words = text.split(' ');
            words.sort();
            document.getElementById('result').innerHTML = '<br>Sorted Words: ' + words;
        }
    </script>
</body>
</html>
```

5. Write a script that includes a function which receives an array of names as parameter and returns the number of names in the given array that end in either ‘ie’ or ‘y’.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Program 5</title>
</head>
<body>
    <h1>Name Suffix Counter</h1>
    <p>Enter names separated by commas:</p>
    <input type="text" id="names"><br><br>
    <button onclick="onButtonClick()">Count Names</button>
    <p id="result"></p>
    <script>
        function onButtonClick() {
```

```
      let input = document.getElementById('names').value;
      let names = input.split(',').map(name => name.trim());  // Trim each name to remove
extra spaces
      let count = countNames(names);
      document.getElementById('result').innerText = `Number of names ending in 'ie' or 'y':
${count}`;
    }

    function countNames(names) {
      let count = 0;
      for (let i = 0; i < names.length; i++) {
        let name = names[i].toLowerCase();
        if (name.endsWith('ie') || name.endsWith('y')) {
          count++;
        }
      }
      return count;
    }
  </script>
</body>
</html>
```

6. Write a script that sorts an array with numbers. The order for sorting should be inputted from keyboard either as ascending or descending. Using switch array should be sorted according to inputted order.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Array Sorter</title>
</head>
<body>
  <h1>Array Sorter</h1>
  <label>Enter numbers separated by commas: </label>
  <input type="text" id="numbers"><br><br>
  <label>Enter the sorting order (ascending or descending): </label>
  <input type="text" id="order"><br><br>
  <button onclick="sortArray()">Sort Numbers</button>
  <p id="result"></p>
  <script>
```

```
        function sortArray() {
            let input = document.getElementById('numbers').value;
            const order = document.getElementById('order').value.trim().toLowerCase();
            // Convert the input string into an array of numbers
            let numbers = input.split(',').map(num => parseFloat(num.trim()));

            switch (order) {
                case 'ascending':
                    numbers.sort((a, b) => a - b);
                    break;
                case 'descending':
                    numbers.sort((a, b) => b - a);
                    break;
                default:
                    document.getElementById('result').innerText = 'Invalid sorting order. Please enter
"ascending" or "descending".';
                    return;
            }

            document.getElementById('result').innerText = `Sorted numbers in ${order} order:
${numbers}`;
        }
    </script>
</body>
</html>
```

Assignment - 6

1. Show the use four pattern methods provided by JavaScript.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>JavaScript Pattern Methods</title>
</head>
<body>
    <h1>JavaScript Pattern Methods</h1>
    <div id="output"></div>
    <script>
```

```
        const text = "Hello, World! This is NMAMIT college.";

        // 1. search() method
        document.write("First occurrence of 'World': " + text.search(/World/) + "<br>");

        // 2. match() method
        document.write("All occurrences of 'Hello': " + text.match(/Hello/) + "<br>");

        // 3. split() method
        document.write("Split by spaces: " + text.split(" ") + "<br>");

        // 4. replace() method
        const replacedText = text.replace(/World/, "Universe");
        document.write("Replace 'World' with 'Universe': " + replacedText + "<br>");

    </script>
</body>
</html>
```

2. Show how an element can be accessed in different ways.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Accessing Elements</title>
</head>
<body>
    <h1>Different ways of Accessing the elements</h1>
    <div class="myDivClass" id="myDivId">Hello, World!</div>
    <script>
        const elementById = document.getElementById('myDivId');
        document.write("Accessing by ID: " + elementById.innerHTML + "<br>");

        const elementByClass = document.getElementsByClassName('myDivClass')[0];
        document.write("Accessing by Class: " + elementByClass.innerHTML + "<br>");

        const elementByTagName = document.getElementsByTagName('div')[0];
        document.write("Accessing by Tag Name: " + elementByTagName.innerHTML + "<br>");
    </script>
</body>
</html>
```

3. Using an implicit array, display the number of checkboxes related to grocery items to be purchased that have been checked.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Grocery Items Checklist</title>
</head>
<body>
    <h1>Grocery Items</h1>
    <input type="checkbox" class="groceryItem">Milk</br>
    <input type="checkbox" class="groceryItem">Butter</br>
    <input type="checkbox" class="groceryItem">Bread</br>
    <input type="checkbox" class="groceryItem">Egg</br>

    <p id="result"></p>
    <script>
        const checkboxes = document.querySelectorAll('.groceryItem');
        checkboxes.forEach(checkbox => {
            checkbox.addEventListener('change', () => {
                const count = Array.from(checkboxes).filter(item => item.checked).length;
                document.getElementById('result').innerHTML = `You have selected ${count} grocery items`;
            });
        });
    </script>
</body>
</html>
```

4. Create an HTML document that includes five radio buttons, labelled red, blue, green, yellow, and orange. The event handlers for these buttons must produce messages stating the chosen favorite color. The event handler must be implemented as a function whose name must be assigned to the onclick attribute of the radio button elements. The chosen color must be sent to the event handler as a parameter.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Favorite Color Selector</title>
</head>
<body>
    <h1>Select Your Favorite Color</h1>
    <input type="radio" name="color" onclick="displayFavoriteColor('Red')">Red</br>
    <input type="radio" name="color" onclick="displayFavoriteColor('Blue')">Blue</br>
    <input type="radio" name="color" onclick="displayFavoriteColor('Green')">Green</br>
    <input type="radio" name="color" onclick="displayFavoriteColor('Yellow')">Yellow</br>
    <input type="radio" name="color" onclick="displayFavoriteColor('Orange')">Orange</br>

    <p id="result"></p>

    <script>
        function displayFavoriteColor(color) {
            document.getElementById('result').innerHTML = `Your favorite color is ${color}.`;
        }
    </script>
</body>
</html>
```

5. Rewrite the above exercise to handle the events using the DOM2 event model.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Favorite Color Selector</title>
</head>
<body>
    <h1>Select Your Favorite Color</h1>
    <input type="radio" name="color" value="Red">Red</br>
    <input type="radio" name="color" value="Blue">Blue</br>
    <input type="radio" name="color" value="Green">Green</br>
    <input type="radio" name="color" value="Yellow">Yellow</br>
    <input type="radio" name="color" value="Orange">Orange</br>

    <p id="result"></p>

    <script>
        // Select all radio buttons
        const radios = document.querySelectorAll('input[name="color"]');
        // Add event listener to each radio button
        radios.forEach(radio => {
            radio.addEventListener('change', () => {
                document.getElementById('result').innerHTML = `Your favorite color is
${radio.value}.`;
            });
        });
    </script>
</body>
</html>
```

6. Create a JavaScript for the following: When a page is loaded, it should display ‘Welcome to the Computation page’ in an alert; then two numbers are provided in text boxes (validate that they are numbers—if a non-numeric value is entered, an appropriate message should be displayed). In another text box, provide an operation to be carried out in the form of symbols (+, -, *, /, %). Validate this entry also. Include a button ‘Compute’ which, when clicked, should carry out the corresponding arithmetic operation and display the result in a text box (which should not be allowed to edit).

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Computation Page</title>
    <script>
        window.onload = function () {
            alert('Welcome to the Computation page');
        };

        function computeResult() {
            const num1 = document.getElementById('num1').value;
            const num2 = document.getElementById('num2').value;
            const operation = document.getElementById('operation').value;

            // Validate that num1 and num2 are numbers
            if (isNaN(num1) || isNaN(num2)) {
                alert('Please enter valid numbers.');
                return;
            }

            // Validate the operation input
            if (!['+', '-', '*', '/', '%'].includes(operation)) {
                alert('Please enter a valid operation (+, -, *, /, %).');
                return;
            }

            const number1 = parseFloat(num1);
            const number2 = parseFloat(num2);

            let result;
            switch (operation) {
                case '+':
                    result = number1 + number2;
                    break;
                case '-':
                    result = number1 - number2;
                    break;
                case '*':
                    result = number1 * number2;
                    break;
                case '/':
                    if (number2 === 0) {
                        alert('Cannot divide by zero.');
                        return;
                    }
                    result = number1 / number2;
                    break;
                case '%':
                    result = number1 % number2;
                    break;
            }
            document.getElementById('result').value = result;
        }
    </script>
</head>
<body>
    <h1>Computation Page </h1>
    <label>Enter First Number: </label>
    <input type="number" id="num1"><br><br>

    <label>Enter Second Number: </label>
    <input type="number" id="num2"><br><br>

    <label>Enter Operation (+, -, *, /, %): </label>
    <input type="text" id="operation"><br><br>
    <button onclick="computeResult()">Compute</button><br><br>

    <label>Result: </label>
    <input type="text" id="result" readonly><br><br>
</body>
</html>
```

7. Create an HTML document related to a registration to a service. Include appropriate JavaScript for the following validations:

- Input a name (Confirm that the first name, middle name, and last name are provided, with all of them being words that start with a capital letter).
- Input age (with a minimum value of 18 and a maximum value of 60).
- Input a mobile number (Confirm that it is an Indian mobile number).
- Input Aadhaar number.
- Set a password (letters, digits, and at least one special character are required).
- Confirm the password.

Display an appropriate message whenever a wrong entry is made.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>
    <style>
        .error {
            color: red;
        }
    </style>
</head>
<body>

<h2>Registration Form</h2>
<form id="registrationForm">
    <label>First Name:</label>
    <input type="text" id="firstName" required><br><br>

    <label>Middle Name:</label>
    <input type="text" id="middleName" required><br><br>
    <label>Last Name:</label>
    <input type="text" id="lastName" required><br><br>

    <label>Age:</label>
    <input type="number" id="age" min="18" max="60" required><br><br>

    <label>Mobile Number:</label>
    <input type="text" id="mobile" required><br><br>

    <label>Aadhaar Number:</label>
    <input type="text" id="aadhaar" required><br><br>

    <label>Password:</label>
    <input type="password" id="password" required><br><br>

    <label>Confirm Password:</label>
    <input type="password" id="confirmPassword" required><br><br>

    <button type="submit">Register</button>
</form>

<p id="message" class="error"></p>

<script>
    document.getElementById("registrationForm").onsubmit = function(event) {
        event.preventDefault(); // Prevent form submission
        let message = "";

        // Validate names
        const firstName = document.getElementById("firstName").value;
        const middleName = document.getElementById("middleName").value;
        const lastName = document.getElementById("lastName").value;

        if (!firstName[0].match(/[A-Z]/) || !middleName[0].match(/[A-Z]/) ||
!lastName[0].match(/[A-Z]/)) {
            message += "First, middle, and last names must start with a capital letter.\n";
        }

        // Validate age
        const age = Number(document.getElementById("age").value);
        if (age < 18 || age > 60) {
            message += "Age must be between 18 and 60.\n";
        }
        // Validate mobile number (Indian mobile number)
        const mobile = document.getElementById("mobile").value;
        if (mobile.length !== 10 || mobile[0] < '6' || mobile[0] > '9') {
            message += "Please enter a valid 10-digit Indian mobile number.\n";
        }

        // Validate Aadhaar number
        const aadhaar = document.getElementById("aadhaar").value;
        if (aadhaar.length !== 12 || isNaN(aadhaar)) {
            message += "Aadhaar number must be a 12-digit number.\n";
        }

        // Validate password
        const password = document.getElementById("password").value;
        if (password.length < 8 ||
            !/[A-Za-z]/.test(password) ||
            !/\d/.test(password) ||
            !/[!@#$%^&*]/.test(password)) {
            message += "Password must be at least 8 characters long and include letters, digits,
and a special character.\n";
        }

        // Confirm password
        const confirmPassword = document.getElementById("confirmPassword").value;
        if (password !== confirmPassword) {
            message += "Passwords do not match.\n";
        }

        // Display message
        document.getElementById("message").textContent = message || "Registration
successful!";
    };
</script>

</body>
</html>
```

8. Show the concept of event propagation in JavaScript event handling.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <title>Event Propagation Example</title>
</head>
<body>

<div id="parent" style="padding: 20px; background: lightblue;">
    Parent Div
    <div id="child" style="padding: 20px; background: lightcoral;">
        Child Div
    </div>
</div>

<script>
    // Event listener for the parent div
    document.getElementById("parent").addEventListener("click", function() {
        alert("Parent Div Clicked");
    });

    // Event listener for the child div
    document.getElementById("child").addEventListener("click", function(event) {
        alert("Child Div Clicked");
        event.stopPropagation(); // Prevents bubbling
    });
</script>

</body>
</html>
```

Assignment 7

1 . Show the use of various values of position property.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Position Property.</title>
    <style>
        /* Static position: The default value */
        .static {
            background-color:aqua;
            height: 50px;
            width: 100px;
            margin: 10px;
        }
        /* Relative position: Positioned relative to its normal position */
        .relative {
            background-color: brown;
            height: 50px;
            width: 100px;
            position: relative;
            top: 20px;
            left: 30px;
        }
        /* Absolute position: Positioned relative to the nearest positioned ancestor */
        .absolute {
            background-color:blueviolet;
            height: 50px;
            width: 100px;
            position: absolute;
            top: 100px;
            left: 300px;
        }
        /* Fixed position: Positioned relative to the viewport */
        .fixed {
            background-color:chocolate;
            height: 50px;
            width: 100px;
            position: fixed;
            top: 250px;
            left: 50px;
        }
        /* Sticky position: Sticks to the viewport once it scrolls past the top */
        .sticky {
            background-color:green;
            height: 50px;
            width: 100px;
            position: sticky;
            left: 200px;
        }
    </style>
</head>
<body>
    <h1>Position Property</h1>
    <div class="static">Static Position</div>

    <div class="relative">Relative Position</div>

    <div class="absolute">Absolute Position</div>

    <div class="fixed">Fixed Position</div>

    <div class="sticky">Sticky Position</div>

</body>
</html>
```

1. XML Document with CSS Styling

XML File (university.xml):

xml

Copy code

```
<?xml version="1.0" encoding="UTF-8"?>
<university>
  <name>ABC University</name>
  <address>
    <street>123 Elm Street</street>
    <city>Metropolis</city>
    <state>NY</state>
    <zipcode>12345</zipcode>
  </address>
  <departments>
    <department>
      <name>Computer Science</name>
      <courses>
        <course>Data Structures</course>
        <course>Machine Learning</course>
      </courses>
    </department>
    <department>
      <name>Mathematics</name>
      <courses>
        <course>Algebra</course>
        <course>Statistics</course>
      </courses>
    </department>
  </departments>
</university>
```

CSS File (university.css):

css

Copy code

```
university {
  display: block;
  font-family: Arial, sans-serif;
}

name {
  font-size: 24px;
  font-weight: bold;
  color: darkblue;
}

address, department {
  margin: 10px 0;
}

course {
  font-style: italic;
  color: darkgreen;
}
```

How to Apply: Use an XSLT stylesheet to link the XML and CSS if needed for rendering in a browser.

Output:

markdown

Copy code

```
ABC University

123 Elm Street

Metropolis, NY, 12345


Departments:

1. Computer Science
```

   - Data Structures
   - Machine Learning

2. Mathematics
   - Algebra
   - Statistics

Styling will make the university name bold and blue, courses italicized in green, etc.

---

2. PHP Numeric Functions

PHP Code:

php

Copy code

```php
<?php
echo "Absolute: " . abs(-5) . "<br>";
echo "Ceiling: " . ceil(4.3) . "<br>";
echo "Floor: " . floor(4.8) . "<br>";
echo "Square Root: " . sqrt(16) . "<br>";
echo "Random: " . rand(1, 100) . "<br>";
?>
```

Output:

mathematica

Copy code

```
Absolute: 5
Ceiling: 5
Floor: 4
Square Root: 4
Random: 37 (or any random number between 1 and 100)
```

---

3. PHP String Functions

PHP Code:

php

Copy code

```php
<?php
```

```
$str = "Hello World!";
echo "Length: " . strlen($str) . "<br>";
echo "Reverse: " . strrev($str) . "<br>";
echo "Substring: " . substr($str, 6) . "<br>";
echo "Uppercase: " . strtoupper($str) . "<br>";
echo "Lowercase: " . strtolower($str) . "<br>";
?>
```

Output:

yaml

Copy code

```
Length: 12
Reverse: !dlroW olleH
Substring: World!
Uppercase: HELLO WORLD!
Lowercase: hello world!
```

---

4. Type Conversion in PHP

PHP Code:

php

Copy code

```
<?php
// Implicit Conversion
$val = "10" + 5;
echo "Implicit Conversion: $val<br>";

// Explicit Conversion
$val2 = (int)"20" + 5;
echo "Explicit Conversion: $val2<br>";
?>
```

Output:

yaml

Copy code

```
Implicit Conversion: 15
Explicit Conversion: 25
```

---

5. Array Functions in PHP

PHP Code:

php

Copy code

```php
<?php
$arr = [1, 2, 3, 4, 5];
echo "Count: " . count($arr) . "<br>";
echo "Sum: " . array_sum($arr) . "<br>";
echo "Reverse: ";
print_r(array_reverse($arr));
?>
```

Output:

mathematica

Copy code

```
Count: 5
Sum: 15
Reverse: Array ( [0] => 5 [1] => 4 [2] => 3 [3] => 2 [4] => 1 )
```

---

6. Accessing Array Elements Sequentially

PHP Code:

php

Copy code

```php
<?php
$arr = ["a", "b", "c"];
foreach ($arr as $element) {
    echo $element . "<br>";
}
?>
```

Output:

css

Copy code

```
a
b
c
```

---

7. Sorting Arrays

PHP Code:

php

Copy code

```
<?php
$arr = [5, 2, 9, 1];
sort($arr);
echo "Ascending Order: ";
print_r($arr);
?>
```

Output:

mathematica

Copy code

```
Ascending Order: Array ( [0] => 1 [1] => 2 [2] => 5 [3] => 9 )
```

---

8. Pattern Matching

PHP Code:

php

Copy code

```
<?php
$str = "Hello, World!";
if (preg_match("/world/i", $str)) {
    echo "Match found!";
} else {
    echo "No match found!";
}
```

```
?>
```

Output:

sql

Copy code

```
Match found!
```

---

9. Variable Scope

PHP Code:

php

Copy code

```
<?php
function test() {
    global $globalVar;
    $globalVar = "I am global";
    static $staticVar = 0;
    $staticVar++;
    echo "Static: $staticVar<br>";
}
test();
test();
?>
```

Output:

makefile

Copy code

```
Static: 1
Static: 2
```

---

10. Reverse and Sum a Number

PHP Code:

php

Copy code

```
<?php
```

```
$num = 1234;
$rev = strrev($num);
$sum = array_sum(str_split($rev));
echo "Reversed: $rev, Sum: $sum";
?>
```

Output:

yaml

Copy code

```
Reversed: 4321, Sum: 10
```

---

11. First 20 Prime Numbers

PHP Code:

php

Copy code

```
<?php
function isPrime($num) {
    if ($num < 2) return false;
    for ($i = 2; $i <= sqrt($num); $i++) {
        if ($num % $i == 0) return false;
    }
    return true;
}

$primes = [];
for ($i = 2; count($primes) < 20; $i++) {
    if (isPrime($i)) $primes[] = $i;
}
print_r($primes);
?>
```

Output:

css

Copy code

```
Array ( [0] => 2 [1] => 3 [2] => 5 [3] => 7 [4] => 11 [5] => 13 [6] => 17 [7] => 19 [8] => 23 [9] =>
29 ... )
```

---

12. Find Second Most Frequent Element

PHP Code:

php

Copy code

```php
<?php
$arr = [1, 2, 2, 3, 3, 3, 4];
$freq = [];
foreach ($arr as $num) {
    $freq[$num] = isset($freq[$num]) ? $freq[$num] + 1 : 1;
}
arsort($freq);
$values = array_values($freq);
$secondFreq = $values[1];
foreach ($freq as $key => $val) {
    if ($val == $secondFreq) {
        echo "Second Most Frequent: $key<br>";
        break;
    }
}
?>
```

Output:

mathematica

Copy code

```
Second Most Frequent: 2
```

---

13. Max and Min Without Built-in Functions

PHP Code:

php

Copy code

```php
<?php
```

```
$arr = [5, 7, 2, 9];
$max = $min = $arr[0];
foreach ($arr as $num) {
    if ($num > $max) $max = $num;
    if ($num < $min) $min = $num;
}
echo "Max: $max, Min: $min";
?>
```

Output:

mathematica

Copy code

```
Max: 9, Min: 2
```

4o

ASSIGNMENT 1

1.Create an HTML page showing the use of following: Paragraphs, Line breaking, Preserving multiple whitespaces, tabs, linebreaks, Headings, Block quoting, content based tags, Superscript and Subscript, various character entities, horizontal rules.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>My Travel Blog</title>
  <style>
    body {
      font-family: 'Arial', sans-serif;
      background-color: lightgray;
    }

    h1 {
      color: darkslateblue;
      text-align: center;

    }
    h2,h3 {

      color: seagreen;
    }
    div {
      background-color: #ffffff;
      border-radius: 8px;
      box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
      padding: 10px;
    }
```

```
        pre {
            background-color: lightgray;
            border-left: 4px solid grey;
            padding:5px;
            border-radius: 4px;
        }

        blockquote {
            font-style: italic;
            padding: 5px;
            background-color: lightgray;
            border-left: 5px solid grey;
            border-radius: 4px;
        }

    </style>
</head>
<body>

    <h1>Welcome to Travel Blog</h1>
            <div>
        <p>Join me as I explore different corners of the world, sharing my experiences, tips, and travel insights</p>
    </div>
    <hr>
    <div>
        <h2>Trip plan</h2>
        <p>Date: 1<sup>st</sup> August 2024 - 3<sup>rd</sup> August 2024.<br> Note: Please bring H<sub>2</sub>O (water!!).</p>
        <pre>
            Day 1:      Arrival in Udupi
            Day 2:      Visit Shri Krishna Mutt
            Day 3:      Explore Manipal Museum
        </pre>
        <p>On the way plan:&nbsp;&nbsp; Malpe &nbsp;&nbsp;&nbsp;&nbsp;--
->&nbsp;&nbsp;&nbsp;&nbsp; Kunjalgiri &nbsp;&nbsp;&nbsp;&nbsp;--
->&nbsp;&nbsp;&nbsp;&nbsp; Manipal lake </p>
    </div>
    <hr>

    <div>
        <h2>You will experience:</h2>
        <p>Peaceful environment</p>
        <p>Enhanced knowledge</p>
        <p>Love with nature's beauty</p>
    </div>

    <hr>

    <div>
        <h3>Rate us:&nbsp;</h3>
        <p><b>Awesome </b>if rating &gt; 8 &amp; &lt; 10</p>
        <p><b>Good</b> if rating &gt; 5 &amp; &lt; 8</p>
        <p><b>Bad</b> if rating &gt; 3 &amp; &lt; 5</p>
        <p><b>Very bad </b>if rating &lt; 3</p>
    </div>
    <hr>
    <div>
        <blockquote cite="https://www.lonelyplanet.com">
            "Travel is the only thing you buy that makes you richer." - Unknown
        </blockquote>
    </div>
</body>
</html>
```

2.Create an HTML page with

1.a An image

1.b Link to another page

1.c Link to section of the same page

1.d An image as a link to another page

```
<!DOCTYPE html>
<html>
<head>
  <title>Travel Diaries</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: lightgrey;
      text-align: center;
    }
    h1 {
      color: darkslategrey;
      padding: 2px;
    }
    img {
      max-width: 400px;
      border: 2px solid grey;
      border-radius: 8px;
    }
    a {
      color: darkslategrey;
      text-decoration: none;
    }
    a:hover {
      text-decoration: underline;
    }

    h4,h3 {
      color: navy;
    }
```

```
        p {
            margin: -5px 0;
        }

         ul {
            list-style-type: disc;
            padding-left: 20px;
            text-align: left;
            margin: 0 auto;
            max-width: 400px;
        }
    </style>
</head>
<body>
    <h1>Welcome to the Travel Page</h1>

    <!-- Image on the page -->
    <img src="https://t4.ftcdn.net/jpg/03/41/40/71/360_F_341407103_ycPF1c11FnIsi5d6xL09xrECMdZ0DFs1.jpg" alt="Travel Image">

    <!-- Link to another page -->
    <h4>Interested in a world trip? Click the link below:</h4>
    <p><a href="https://www.lonelyplanet.com" target="_blank">Visit Lonely Planet</a></p>

    <br>

    <!-- Link to a section of the same page -->
    <p><a href="#section">Go to Travel Tips Section</a></p>

    <br><br>

    <h4>Want to book? Click the image below:</h4>
    <!-- Image as a link to another page -->
    <div>
        <a href="https://www.expedia.com" target="_blank">
            <img src="https://www.shutterstock.com/image-photo/dhaka-bangladesh-22-february-2024-260nw-2432040483.jpg" alt="Expedia Image Link">
        </a>
    </div>

    <br>

    <div id="section">
        <h3>Travel Tips Section</h3>

        <ul>
            <li>Plan your itinerary in advance.</li>
            <li>Check travel advisories and weather conditions.</li>
            <li>Pack light and carry essentials only.</li>
            <li>Learn a few phrases of the local language.</li>
        </ul>
    </div>
</body>
</html>
```

3. Create an HTML page for displaying Vision, Mission, PEOs, POs, PSOs of the MCA department as displayed in the lab. Try different types of numbering and bulleting.

```
<!DOCTYPE html>
<head>
            <title>MCA dept</title>
            <style>
            ul.asterisk_bullets {
                                    list-style-type: none;

                        }
```

```
        .asterisk_bullets li::before {
        content: '*';
      }
        </style>
</head>
<body>
        <h1>NMAMIT</h1>
        <hr>
        <h3>Vision</h3>
        <p>Equipping students with computing and programming domain expertise within the state of the art technology<br> solutions to enable them to meet global professional challenges</p>
        <br>

        <h3>Mission</h3>
        <p>The department strives to create a conductive environment equping students to work in teams and have</p>
        <ul>
          <li>Professional Ethics</li>
          <li>Sound Technical Knowledge</li>
          <li>Skills to Handle Technological Challenges</li>
        </ul>
        <br>

        <h3>Programme Educational Objectives (PEOs)</h3>
        <ul class="asterisk_bullets">
          <li>To think critically, work creatively, communicate effectively, and become technologically component</li>
          <li>To be able to update themselves in areas and technologies relevent to their career</li>
          <li> To develop teamwork ability, lead initiatives and manage tasks effectively</li>
        </ul>

        <br>
        <h3>Programme Outcoms(POs)</h3>

        <ol>

          <li>Ability to apply knowledge of mathematics, science and engineering</li>

          <li>Identify, formulate. research literature, and analyze complex engineering problems</li>

          <li>Ability to communicate effectively</li>

          <li>Recognition of the need for and ability to engage in lifelong learning</li>

          <li>A knowlege of conteporary issues</li>

        </ol>


        <h3>Programme Specific Outcoms(PSOs)</h3>

        <ol>

          <li>To empower the students tot exempify themselves in the field of information Technology ensuring credibility</li>

          <li>Adopt researchpractices and trending technologies such as AI and ML,IOT</li>

        </ol>




</body>

</html>
```

4.Create an HTML page with Abbreviations of courses offered at NMAMIT as terms and definitions.

```
<!DOCTYPE html>
<head>
        <title>Academic course Abbreviations</title>
</head>
<body>
        <h1>Abbreviations of courses offered at NMAMIT</h1>
        <p><dfn><abbr title="Master of Computer Applications">MCA</abbr></dfn>: is a postgraduate degree in computer science.</p>
        <p><dfn><abbr title="Master of Commerce">BCom</abbr></dfn>: is an postgraduate degree in commerce and business. </p>
        <p><dfn><abbr title="Master of Technology ">MTECH</abbr></dfn>: is a postgraduate degree in engineering and technology. </p>
        <p><dfn><abbr title="Information Science and Engineering">ISE</abbr></dfn>: focuses on study of information and system and technology.</p>
        <p><dfn><abbr title="Electronics and Communication Engineering">EC</abbr></dfn>: Deals with design and development of electronic system and communication technologies.  </p>
```

```
	<p><dfn><abbr title="Bachelor of Technology">BTECH</abbr></dfn>: is an undergraduate degree in engineering and technology. </p>
	<p><dfn><abbr title="Doctor of Philosophy">Ph.D</abbr></dfn>: is a doctoral degree awarded for advanced research in a specific field. </p>
	<p><dfn><abbr title="Bachelor of Computer Applications">BCA</abbr></dfn>: is an undergraduate degree in computer applications. </p>
	<p><dfn><abbr title="Bachelor of Commerce">BCom</abbr></dfn>: is an undergraduate degree in commerce and business. </p>

</body>
</html>
```

5.Create an HTML page with following nesting of lists:

To do List:

1. Shopping

  a. Cloths

  b. Cell Phone

2. Meeting

  a. Friends

  - **Friend1**
  - **Friend2**

  b. Teacher

  - **Teacher1**
  - **Teacher2**

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


    <title>To-Do List</title>
    <style>
        ul, ol {
            list-style-type: none;


        }
        ol.top-level {
            list-style-type: decimal;
        }

        ol.top-level > li > ol {
            list-style-type: lower-alpha;
            padding-left: 20px;
        }
        ol.top-level > li > ol > li > ul > li {
            list-style-type: disc;
            padding-left: 20px;
        }
    </style>
</head>
<body>
    <h1>To-Do List</h1>
    <ol class="top-level">
        <li>Shopping
            <ol>
                <li>Cloths</li>
                <li>Cell Phone</li>
            </ol>
        </li>
        <li>Meeting
            <ol>
                <li>Friends
                    <ul>
                        <li>Friend1</li>
                        <li>Friend2</li>
                    </ul>
                </li>
                <li>Teacher
                    <ul>
                        <li>Teacher1</li>
                        <li>Teacher2</li>
                    </ul>
                </li>
            </ol>
        </li>
    </ol>
</body>
</html>
```

6.Display your class time table.

```
<DOCTYPE html>
<head>
            <title>Time Table</title>
            <style>
                        table{
                                    width:100%;
                                    border-collapse: collapse;
                        }
                        th,td{
                                    border:1px solid black;
                                    padding:10px;
                                    text-align:center;
                        }
            </style>
</head>
<body>
            <h2>TimeTable</h2>
            <table>
                        <thead>
                                    <tr>
                                                <th>Day/Time</th>
                                                <th>9:00-9:55</th>
                                                <th>9:55-10:50</th>
```

```
                <th>11:10-12:05</th>
                <th>12:05-1:00</th>
                <th>1:00-1:55</th>
                <th>1:55-2:50</th>
                <th>2:50-3:40</th>
                <th>3:40-4:30</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>Monday</td>
                <td colspan="3"> AWT LAB </td>
                <td></td>
                <td rowspan="5">LunchBreak</td>
                <td>NLP</td>
                <td>AIML</td>
                <td>AWT</td>
            </tr>

            <tr>
                <td>Tuesday</td>
                <td> AWT</td>
                <td> AIML</td>
                <td>MC&AD</td>
                <td>NLP</td>
                <td colspan="2">Placement Training</td>
                <td></td>
            </tr>

            <tr>
                <td>Wednesday</td>
                <td> AWT</td>
                <td>AIML </td>
                <td>Placement Training</td>
                <td>NW&CS</td>
                <td colspan="3">AIML LAB </td>
            </tr>

            <tr>
                <td>Thursday</td>
                <td> NLP</td>
                <td>MC&AD </td>
                <td>NW&CS</td>
                <td></td>
                <td colspan="2">Placement Training</td>
                <td></td>
            </tr>

            <tr>
                <td>Friday</td>
                <td> NW&CS</td>
                <td>AWT </td>
                <td>AIML</td>
                <td>MC&AD</td>
                <td colspan="3">Mini Project/Journal club</td>
            </tr>
        </tbody>
    </table>
</body>

</html>
```

7.Design a student registration form for student’s enrolment to the course which collects personal and previous educational details. Include form elements: label, input, select, text area, fieldset, legend, datalist.

```
<!DOCTYPE html>
<head>
        <title> Student Registration form</title>
        <style>

     body {
        font-family: Arial;

     }
     .formgroup {
        margin-bottom: 15px;
     }
     form {
        max-width: 600px;
        margin: 0 auto;
        padding: 20px;
        border: 1px solid #ddd;
        border-radius: 8px;
        background-color: #f9f9f9;
     }
     fieldset {
        padding: 10px;
        margin-bottom: 20px;
        border-radius: 5px;
     }
     legend {
        font-weight: bold;
     }
     input[type="text"], input[type="email"], input[type="tel"], input[type="date"], select, textarea {
        width: 100%;
        padding: 8px;
        border: 1px solid #ccc;
        border-radius: 4px;
        box-sizing: border-box;
     }
     textarea {
        resize: vertical;
     }
        </style>
</head>
<body>
        <h1> Student Registration form</h1>
        <form action="" method="post">
        <fieldset>
                <legend>Personal Details</legend>

                <div class="formgroup">
                        <label for="fname">Full Name:</label>
                        <input type="text" id="fname" name="fname" required>
                </div>

                <div class="formgroup">
                        <label for="lname">Last Name:</label>
                        <input type="text" id="lname" name="lname" required>
                </div>

                <div class="formgroup">
                        <label for="dob">DOB:</label>
                        <input type="date" id="dob" name="dob" required>
                </div>

                <div class="formgroup">
                        <label for="gender">Gender:</label>
                        <select id="gender" name="gender" required>
                                <option value="" disabled selected>Select your gender</option>
                                <option value="male">Male</option>
                                <option value="female">Female</option>
                                <option value="other">Other</option>
                        </select>
                </div>

                <div class="formgroup">
                        <label for="email">Email:</label>
                        <input type="email" id="email" name="email" required>
                </div>

                <div class="formgroup">
                        <label for="phone">Phone No:</label>
                        <input type="tel" id="phone" name="phone" required>
                </div>
        </fieldset>

        <fieldset>
                <legend>Previous Education</legend>
                <div class="formgroup">
          <label for="course">Select Course:</label>
          <select id="course" name="course" required>
              <option value="" disabled selected>Select a course</option>
              <option value="bsc">B.Sc</option>
              <option value="bcom">BCom</option>
              <option value="bca">BCA </option>
              <option value="ba">BA</option>
          </select>
      </div>

      <div class="formgroup">
```

```
            <label for="clg">College:</label>
            <input type="text" id="clg" name="clg" required>
        </div>

        <div class="formgroup">
            <label for="passoutyear">Year of Passed out:</label>
            <input type="text" id="passyear" name="passyear" list="year-list" required>
            <datalist id="year-list">
              <option value="2020">
              <option value="2021">
              <option value="2022">
              <option value="2023">
              <option value="2024">
            </datalist>
        </div>

         <div class="formgroup">
            <label for="cgpa">CGPA:</label>
            <input type="number" id="cgpa" name="cgpa" required>
        </div>

        <div class="formgroup">
            <label for="achievements">Academic Achievements:</label>
            <textarea id="achievements" name="achievements" rows="4" placeholder="Enter any
academic achievements or awards"></textarea>
          </div>




          </fieldset>

          <fieldset>
          <legend>Course Enrollment</legend>
          <div class="formgroup">
```

```
                <label for="course">Select Course:</label>
                <select id="course" name="course" required>
                    <option value="" disabled selected>Select a course</option>
                    <option value="mca">Master of Computer Application</option>
                    <option value="msc">M.Sc. Computer Science</option>
                    <option value="msc">M.Sc. Data Science</option>
                    <option value="mba">MBA Management</option>
                    <option value="phd">Ph.D. in Engineering</option>
                </select>
            </div>
            <div class="formgroup">
                <label for="certificates">Certificates:</label>
                <textarea id="certificates" name="certificates" rows="4" placeholder="Enter any certificates which you completed"></textarea>
            </div>
        </fieldset>



</body>


</html>
```

ASSIGNMENT 2

1.Create a web page with different levels of stylesheets. Also show the priority applied in these levels of stylesheets.

```
<!DOCTYPE html>
<head>
    <title>Special Promotional Event</title>
    <!-- External Stylesheet -->
    <link rel="stylesheet" href="as2p1css.css">

    <style>
        /* Internal Stylesheet */
        .event-description {
            font-size: 18px;
            color: darkblue;  /*overridden by external stylesheet.highpriority than
external*/
            margin: 20px 0;
        }
        h1{
        color:black;             /*but inline style has more priority*/
        }
        h2{
        color:darkslategrey;  /*overridden by external stylesheet.highpriority than
external*/
        }

    </style>
</head>
<body>
    <!-- Inline Styles -->  <!--high priority -->
    <header style="background-color: black; text-align: center; padding: 10px;">
        <h1 style="color: white; font-size: 36px;">Join Us for Our Special Promotional
Event!</h1>
    </header>

    <main>
        <section>
            <h2>Event Details</h2>
            <p class="event-description">We're excited to invite you to our promotional
event where you'll get exclusive offers and a chance to win amazing prizes. Don't
miss out!</p>
```

```
            <p class="event-details">Date: August 15, 2024<br>
            Time: 6:00 PM - 9:00 PM<br>
            Venue: Downtown Convention Center</p>
        </section>
    </main>
</body>
</html>

--------
```

as2p1css.ccs

```
/* External Stylesheet */
h1 {
    color: navy; /* External style for h1, lower priority */
    font-size: 100px; /* External style for h1, lower priority */
}
h2{
    color:red;    /*low priority*/
   }
.event-description {   /*low priority*/
    color: darkred;
    font-size: 20px;
}
.event-details {
    color: brown;
    font-size: 18px;
}


/*
inline stylesheet  --high
internal stylesheet
external stylesheet  --low */
```

2.Create a page which shows the usage of different selectors for specifying style rules.

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Event Promotions</title>
<style>
  /* Element Selector */
  h2 {
    color: navy; /* Styles all <h2> elements */
    font-size: 28px;
  }
  h3{
   color:orange;
  }

  /* Class Selector */
  .highlight {
    background-color: yellow; /* Highlights important announcements */
    font-weight: bold;
    padding: 12px;
  }

  /* ID Selector */
  #special-offer {
    background-color: lightgreen; /* Special banner for special offers */
    color: darkgreen;
    text-align: center;
    padding: 20px;
    font-size: 22px;
  }

  /* Attribute Selector */
  a[href*="event"] {
    color: dodgerblue; /* Styles links with "event" in their URL */
    font-style: italic;
  }

  /* Pseudo-class Selector */
  a:hover {
    color: orange; /* Styles links on hover */
    text-decoration: underline;
  }

  /* Pseudo-element Selector */
  p::first-line {
    font-weight: bold; /* Styles the first line of each paragraph */
  }

  /* Descendant Selector */
  .event-description p {
    color: slategray; /* Styles <p> elements within the event-description section */
  }

  /* Child Selector */
  .event-list > li {
    color: darkslategray; /* Styles direct <li> children of .event-list */
  }

  /* Adjacent Sibling Selector */
  h3 + p {
    color: brown; /* Styles <p> elements immediately following an <h3> */
  }
```

```
        /* General Sibling Selector */
        h3 ~ p {
            font-style: italic;
            /* Styles all <p> elements following an <h3> */
        }
    </style>
</head>
<body>
    <header id="special-offer">
        **Exclusive Offer: Early Bird Discount - Save 25% on All Tickets!**
    </header>

    <main>
        <section class="event-description">
            <h2>Upcoming Shows</h2>
            <p>Experience unforgettable performances and activities!</p>
            <p class="highlight">Join our exclusive VIP event for a meet-and-greet with the stars!</p>
            <p>Find out what's new and trending in the world of entertainment.</p>
        </section>
        <hr>

        <section class="event-list">
            <h2>Event Categories</h2>
            <ul>
                <li>Concerts</li>
                <li>Workshops</li>
                <li>Exhibitions</li>
            </ul>
        </section>
```

```
          <hr>
        <section>
            <h3>Highlighted Events</h3>
            <p>Catch the best events of the month with special guest appearances.</p>
            <p>Secure your place early to avoid missing out!</p>
        </section>
                    <hr>
        <section>
            <h3>Recent Updates</h3>
            <p>Stay informed with the latest event news and updates.</p>
            <a href="http://example.com/event1">Exciting Event Highlights</a><br>
            <a href="http://example.com/news">Read Our Latest Blog Post</a>
        </section>
    </main>
</body>
</html>
```

3.Create a page which includes application of font and text related style properties.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Subscription Confirmed</title>
    <style>
        /* Font and Text Styles */
        body {
            font-family: 'Verdana', sans-serif;
            background-color: #e0f7fa;
            color: #00796b;
            text-align: center;
            margin: 0;
            padding: 20px;
            line-height: 1.6;
        }
        h1 {
            font-family:'Times New Roman';
            font-size: 1.8em;
            font-weight: bold;
            font-style: italic;
            font-variant: small-caps;
            color: brown;
            text-align: center;
            text-shadow: 12px 2px 4px rgba(200, 250, 20, 0.5);
        }
        p {
            font-family: 'Arial', sans-serif;
            font-size: 1em;
            font-weight: normal;
            color: #00796b;
            text-align: center;
            text-indent: 20px;
            line-height: 2;
            letter-spacing: 1px;
            word-spacing: 2px;
            text-transform: capitalize;
            text-decoration: none;
        }
        .highlight {
            color: orange;
            font-weight: bold;
            text-transform: uppercase;
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <h1>Thank You for Subscribing!</h1>
    <p>Your subscription to <span class="highlight">Daily Yoga Tips</span> has
been confirmed.</p>
    <p>we’re excited to share daily tips to enhance your yoga practice!</p>
</body>
</html>
```

4.Create a page which includes application of list related style properties.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Grocery Shopping List</title>
<style>
    body {
        font-family: 'Arial', sans-serif;
        background-color: lightgrey; /* Background color of the page */
        color: darkslategray; /* Default text color */
        padding: 20px;
    }
    h1 {
        text-align: center;
        color: midnightblue; /* Color of the main heading */
    }

    /* Ordered List Styles */
    ol {
        list-style-type: decimal; /* Numeric list items */
        padding-left: 100px; /* Indentation from left */
        margin-bottom: 20px; /* Space below the list */
        border: 1px solid red; /* Border around the list */
        border-radius: 5px; /* Rounded corners for the border */
        background-color: white; /* Background color */
    }
    ol li {
        margin-bottom: 10px; /* Space between items */
        padding: 10px; /* Padding inside list items */
        font-size: 1.1em; /* Slightly larger text */
        color: navy; /* Text color for ordered list items */
    }
    ol li:nth-child(odd) {
        background-color: lightyellow; /* Alternate row colors */
    }
    ol li:hover {
        background-color: lightcoral; /* Highlight on hover */
    }

    /* Unordered List Styles */
    ul {
        list-style-type: disc; /* Default bullets */
        padding-left: 30px; /* Indentation from left */
        margin-bottom: 5px; /* Space below the list */
        border: 1px solid black; /* Border around the list */
        border-radius: 10px; /* Rounded corners for the border */
        background-color: lightblue; /* Background color */
    }
    ul li {
      margin-bottom: 8px; /* Space between items */
      padding: 8px; /* Padding inside list items */
      font-size: 1em; /* Standard text size */
      color: darkgreen; /* Text color for unordered list items */

    }
    ul li:hover {
      background-color: slateblue; /* Highlight on hover */
    }

    /* Nested List Styles */
    ul ul {
      list-style-type: square; /* Square bullets for nested lists */
      background-color: skyblue;
      padding-left: 20px; /* Less indentation for nested items */
    }
    ol ol {
      list-style-type: lower-alpha; /* Lowercase alphabetic for nested ordered lists
*/
      padding-left: 20px; /* Less indentation for nested items */
    }
  </style>
</head>
<body>
  <h1>Grocery Shopping List</h1>

  <h2>Prioritize These Items</h2>
  <ol>
    <li>Milk</li>
    <li>Bread</li>
    <li>Eggs</li>
  </ol>

  <h2>Other Items</h2>
  <ul>
    <li>Fruits
      <ul>
        <li>Apples</li>
        <li>Bananas</li>
      </ul>
    </li>
    <li>Vegetables
      <ul>
        <li>Carrots</li>
```

```
                <li>Spinach</li>
            </ul>
        </li>
        <li>Snacks</li>
    </ul>
</body>
</html>
```

5.Show the use of Box Model properties.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Product Box</title>
    <style>
        body {
            font-family: 'Arial', sans-serif;
            background-color: lightgrey;

        }
        .product-box {
            width: 300px; /* Width of the product box */
            padding: 30px; /* Space inside the box, around the content */
            border: 2px solid black; /* Border around the box */
            border-radius: 8px; /* Rounded corners for the box */
            background-color: white; /* Background color of the box */
            margin: 50px; /* Space outside the box */
            box-sizing: border-box; /* Includes padding and border in the width */
            box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2); /* Shadow effect for depth */
        }
        .product-box img {
            width: 100%; /* Full width image inside the box */
            height: auto; /* Maintain aspect ratio of the image */
            border-bottom: 5px solid navy; /* Border below the image */
            margin-bottom: 10px; /* Space below the image */
            border-radius: 5px; /* Rounded corners for the image */
        }
        .product-box h3 {
            margin: 0; /* Remove default margin */
            padding: 0; /* Remove default padding */
            color: tomato; /* Text color */
            text-align: center; /* Center align text */
        }
        .product-box p {
            margin: 10px 0; /* Space above and below the paragraph */
            color: #777; /* Text color for the price */
            text-align: center; /* Center align text */
        }
    </style>
</head>
<body>
    <div class="product-box">
        <img src="https://cdn.pixabay.com/photo/2017/05/15/17/44/cup-
2315565_640.jpg" alt="Coffee Mug">
        <h3>Coffee Mug</h3>
        <p>300Rs</p>
    </div>
</body>
</html>
```

6.Show the use of div and span tags.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Special Offer</title>
    <style>
        body {

            padding: 20px;
        }
        .container {
            background-color: lightgrey; /* Background color for the container */
            padding: 20px;
            border: 1px solid black; /* Border color for the container */

        }
        .offer-header {
            color: royalblue; /* Color for the header text */
            font-size: 24px; /* Font size for the header */
            margin-bottom: 15px; /* Space below the header */
        }
        .offer-content {
            margin-bottom: 20px; /* Space below the content */
        }
        .offer-description {
            color: blueviolet; /* Color for the offer description text */
        }
        span {
            color: crimson; /* Color for highlighted text */
            font-weight: bold; /* Bold text for emphasis */
        }
        .offer-footer {
            text-align: center; /* Center-align text in the footer */
            font-size: 20px; /* Font size for the footer */
            color: green; /* Color for the price text */
            font-weight: bold; /* Bold text for emphasis */
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="offer-header">Special Summer Offer!</div>
        <div class="offer-content">
            <div class="offer-description">
                Get the best deals on your favorite products. <span >Save up to 50%</span> on selected items. Don't miss out on this amazing opportunity to <span >upgrade your summer essentials</span> at unbeatable prices!
            </div>
            <div class="offer-footer">
                Only $29.99!
            </div>
        </div>
    </div>
</body>
</html>
```

ASSIGNMENT 3

1. Write a JavaScript to show the implicit and explicit type conversion.
2. Show the use of String Object's properties and methods.
3. Show the use of important methods of Date Object.
4. Take required inputs from the keyboard and display the roots of quadratic equation.
5. Show the use of strict equality and loose equality operators in JavaScript.
6. Write a JavaScript to find the largest of three inputted numbers.
7. Find the day of the week and accordingly show the menu of food items in table using switch statement. Apply different table border size for different tables.
8. Write a JavaScript to find the sum of first n natural numbers using while and for loop.
9. Show the use of array methods in JavaScript.
10. Write a JavaScript that uses function to find the sum 3 of inputted numbers.
11. Write a JavaScript to show the scope of variables.

ASSIGNMENT 4

1 **Create a basic HTML5 webpage structure. Include a header, navigation menu, main content section, and a footer. Use semantic HTML5 tags like <header>, <nav>, <main>, and <footer>.**

```
<!DOCTYPE html>
<html lang="en">
<head>
        <title>My Portfolio</title>
        <style>
                body {
                font-family: Arial, sans-serif;
                margin: 0;
                padding: 0;
                }
                header {
                background-color: #333;
                color: #fff;
                padding: 10px 0;
                text-align: center;
                }
                nav {
                background-color: #444;
                padding: 10px;
                text-align: right;
                }
                #logo{
                margin-right: 900px;
                }
                nav a {
```

```
                color: white;
                margin: 0 15px;
                text-decoration: none;
                }
                main {
                padding: 20px;
                }
                footer {
                background-color: #333;
                color: white;
                text-align: center;
                padding: 10px 0;
                position: fixed; /* sticks it to end of screen*/
                width: 100%;
                bottom: 0;
                }
        </style>
</head>
<body>
        <header>
                <h1>Welcome to My Portfolio</h1>
        </header>
        <nav>
                <a id="logo" href="#">Nihal</a>
                <a href="#about">About Me</a>
                <a href="#projects">Projects</a>
                <a href="#contact">Contact</a>
        </nav>
        <main>
                <section id="about">
```

```
            <h2>About Me</h2>
            <p>Hello! I'm an MCA student with a passion for technology and
software development. I enjoy learning new programming languages and
building innovative projects</p>
            </section>
            <section id="projects">
            <h2>My Projects</h2>
            <p>Here are a few of the projects I have worked on recently:</p>
                  <ul>
                  <li>Washing machine management system using C#</li>
                  <li>To-dolist using react.js and spring boot </li>
                  </ul>
            </section>
            <section id="contact">
                  <h2>Contact Me</h2>
                  <p>You can reach me at <a

      href="mailto:nihalniya@gmail.com">nihalniya@gmail.com</a>.</p>
            </section>
      </main>
      <footer>
            &copy; 2024 Nihal. All rights reserved.
      </footer>
</body>
</html>
```

2 **Design a registration form using HTML5 that includes the following fields: Full Name, Email, Password, Date of Birth, Gender (using radio buttons), and a "Submit" button. Use appropriate HTML5 form elements such as email, password, and date input types.**

```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
        <title>Registration Form</title>
        <style>
                body{
                display: flex;
                flex-direction: column;
                align-items: center;
                }
                form{
                border: 2px solid black;
                padding : 5px;
                width:200px;
                background-color:lightblue;
                }
        </style>
</head>
<body>
        <h2>Registration Form</h2>
        <form action="/submit-form" method="post" border="1px solid black">
                <label for="fullname">Full Name:</label><br>
                <input type="text" id="fullname" name="fullname"
required><br><br>
                <label for="email">Email:</label><br>
                <input type="email" id="email" name="email" required><br><br>
                <label for="password">Password:</label><br>
                <input type="password" id="password" name="password"
required><br><br>
                <label for="dob">Date of Birth:</label><br>
                <input type="date" id="dob" name="dob" required><br><br>
                <label>Gender:</label><br>
                <input type="radio" id="male" name="gender" value="male" required>
                <label for="male">Male</label><br>
                <input type="radio" id="female" name="gender" value="female">
                <label for="female">Female</label><br>
                <input type="radio" id="other" name="gender" value="other">
                <label for="other">Other</label><br><br>
                <button type="submit">Submit</button>
                <button type="reset">Reset</button>
        </form>
</body>
</html>
```

3 **Create a webpage that embeds both a video and an audio file using HTML5. Add controls for play, pause, and volume adjustment. Use the <video> and <audio> tags.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Embedded Video and Audio</title>
</head>
<body>
    <h1>Multimedia Page</h1>
    <h2>Video</h2>
        <video width="400" controls>
        <source src="https://videos.pexels.com/video-files/4114797/4114797-sd_640_360_25fps.mp4" type="video/mp4">Your browser does not support the video tag.</video><br><br>
    <h2>Audio</h2>
    <audio controls>
```

```
        <source src="bekhayali.mp3" type="audio/mpeg">Your browser does not
support the audio element.</audio>
</body>
</html>
```

4 **Design a responsive webpage using HTML5 and CSS. The webpage should have a header, a navigation bar, a content area with two columns (main content and a sidebar), and a footer. Make sure the layout adapts to different screen sizes (mobile, tablet, desktop).**

```
<!DOCTYPE html>
<html lang="en">
<head>
        <title>Industry World</title>
        <style>
                body, header, nav, footer {
                margin: 0;
                padding: 0;
                font-family: Arial, sans-serif;
                background-color:black;
                color:white;
                }
                header {
                background: lightblue;
                color: white;
                padding: 10px;
                text-align: center;
                position: relative;
                }
                nav {
                background: #444;
```

```
display: flex;
justify-content: center;
padding: 10px;
position: fixed;
width: 100%;
top: 0;
left: 0;
z-index: 1000;
}
nav a {
color: white;
text-decoration: none;
margin: 0 10px;
}
.menu-btn {
display: none;
position: absolute;
top: 10px;
left: 10px;
font-size: 24px;
background: none;
border: none;
color: white;
cursor: pointer;
}
.mobile-nav {
position: fixed;
top: 0;
left: -250px;
width: 250px;
```

```
height: 100%;
background: #444;
transition: left 0.3s;
padding-top: 60px;
}
.mobile-nav a {
display: block;
padding: 10px;
color: white;
text-decoration: none;
border-bottom: 1px solid #333;
}
.mobile-nav a:hover {
background: #555;
}
.container {
display: flex;
flex-wrap: wrap;
padding: 20px;

}
.main-content {
flex: 70%;
}
footer {
background: lightblue;
color: white;
text-align: center;
bottom:0;
position: fixed;
```

```
        width: 100%;
        }
        .section {
        margin-bottom: 20px;
        }
        @media (max-width: 768px) {
        nav {
        display: none;
        }
        .menu-btn {
        display: block;
        }
        .container {
        flex-direction: column;
        }
        .main-content, .sidebar {
        flex: 100%;
        }
        .mobile-nav {
        left: 0;
        }
    </style>
</head>
<body>
    <header>
        <br><h1>Industry World</h1>
        <button class="menu-btn" onclick="toggleMenu()">☰ Menu</button>
    </header>
    <nav>
```

```
        <a href="#home">Home</a>
        <a href="#about">About Us</a>
        <a href="#services">Services</a>
        <a href="#contact">Contact</a>
    </nav>
    <div id="mobileNav" class="mobile-nav">
        <a href="javascript:void(0)" onclick="closeMenu()">Close ×</a>
        <a href="#home">Home</a>
        <a href="#about">About Us</a>
        <a href="#services">Services</a>
        <a href="#contact">Contact</a>
    </div>
    <div class="container">
        <div class="main-content" id="home">
            <h2>Welcome to Industry World</h2>
            <p>Find YouTube channel for tech, computer science, and IT content. Learn, grow,and stay updated with the latest in technology.</p>
        </div>
        <div id="about" class="section">
            <h2>About Us</h2>
            <p>Industry World is dedicated to providing high-quality tech education and resources.Our YouTube channel covers a wide range of topics in technology, computer science, and IT.</p>
        </div>
        <div id="services" class="section">
            <h2>Our Services</h2>
            <p>We offer tutorials, reviews, and insightful content to help you stay ahead in the tech world. Check out our videos and stay informed about the latest trends.</p>
        </div>
        <div id="contact" class="section">
            <h2>Contact Us</h2>
```

```
                              <p>For inquiries, collaborations, or feedback, please reach
out to us through the contact form below or via our social media channels.</p>
                    </div>
          </div>
          <footer>
                    <p>Industry World &copy; 2024</p>
          </footer>

          <script>
          function toggleMenu() {
          document.getElementById("mobileNav").style.left = "0";
          }
          function closeMenu() {
          document.getElementById("mobileNav").style.left = "-250px";
          }
          </script>
</body>
</html>
```

5 **Create a simple drawing on an HTML5 canvas element. Draw a rectangle, circle, and a line using JavaScript. Use the <canvas> tag and the associated JavaScript API.**

```
<!DOCTYPE html>
<html lang="en">
<head>
          <title>JS 2D Canvas</title>
</head>
<body>
          <h1>2D Shapes using JS Canvas</h1>
          <canvas id="myCanvas" width="400" height="250" style="border:1px
solid #000;"></canvas>
```

```
        <script>
                const canvas = document.getElementById('myCanvas');
                const ctx = canvas.getContext('2d');
                ctx.fillStyle = 'yellow'; //rect
                ctx.fillRect(200, 50, 150, 100); // (x, y, width, height)
                ctx.beginPath(); //circle
                ctx.arc(100, 100, 50, 0, 2 * Math.PI); // (x, y, radius, startAngle, endAngle)
                ctx.fillStyle = 'blue';
                ctx.fill();
                ctx.beginPath(); //line
                ctx.moveTo(50, 20); // Starting point (x, y)
                ctx.lineTo(350, 20); // Ending point (x, y)
                ctx.strokeStyle = 'green';
                ctx.lineWidth = 5;
                ctx.stroke();
        </script>
</body>
</html>
```

ASSIGNMENT 5

1 Write a JavaScript to display the table of numbers from 5 to 15 and their squares and cubes using alert.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Numbers, Squares, and Cubes</title>
    <script>
        function showTable() {
            let table = "Number    Square    Cube\n";
            table += "=================\n";

            for (let i = 5; i <= 15; i++) {
                let number = i.toString().padEnd(17);
                let square = (i * i).toString().padEnd(13);
                let cube = (i * i * i).toString().padEnd(10);

                table += number + square + cube + "\n";
            }

            alert(table);
        }
    </script>
</head>
<body onload="showTable()">
    <h1>Table for squares and cubes of number</h1>
</body>
</html>
```

2 Write a script to input a number n that is the number of the Fibonacci numbers required as output and generate the Fibonacci series accordingly.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Fibonacci series</title>
</head>
<body>
    <h1>Fibonacci Series Generator:</h1>
    <label for="fibnum">Enter number to genetor fibonacci series:</label>
    <input type="number" min="1" id="number" placeholder="Enter positive number">
    <button onclick="showfib()">Genete fibonacci</button>

    <h2>Fibonacci Series:</h2>
    <p id="result"></p>

    <script>
        function generatefib(n)
        {
            let fibseries=[];
            let f1=0,f2=1,f3;
            for(let i=1;i<=n;i++)
            {
                fibseries.push(f1);
                f3=f1+f2;
                f1=f2;
                f2=f3;
            }
            return fibseries;
        }
        function showfib()
        {
            let n=parseInt(document.getElementById("number").value);



            if(isNaN(n)||n<=0)
            {
                result.textContent="Please enter valid positive number";
            }
            else
            {
                let ans=generatefib(n);
                result.textContent=ans.join(",");
            }
        }
    </script>
</body>
</html>
```

3 Write a JavaScript function which receives a number as its parameter and returns the number with its digits in reverse order.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Reverse Number</title>
</head>
<body>
    <h1>Reverse a  Number</h1>
    <label for"revnum">Enter number to reverse it</label>
    <input type="number" id="num" placeholder="Enter a number">
    <button onclick="revNum()">Reverse</button>
    <p id="result"></p>
    <script>
        function revNum()
        {
            let num=document.getElementById("num").value;
            let isNegative=num<0;

            let reversed=Math.abs(num).toString().split('').reverse().join('');
            if(isNegative)
            {
                reversed='-'+reversed;
            }
            document.getElementById("result").textContent="Reversed Number "+reversed;
        }
    </script>
</body>

</body>
</html>
```

4 Input a line of text from prompt and display the words of input text in alphabetical order.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sort Words Alphabetically</title>

</head>
<body>
    <h1>Sort Words Alphabetically</h1>
    <div id="inputDisplay"></div>
    <div id="result"></div>

    <script>
        function sortWords() {
            let inputText = prompt("Enter a line of text:");

            let inputDisplay = document.getElementById("inputDisplay");
            if (inputText === null || inputText.trim() === '') {
                inputDisplay.textContent = "No input provided.";

                return;
            } else {
                inputDisplay.textContent = "Original Text: " + inputText;
            }

            let words = inputText.split(/\s+/);
            words.sort();
            let sortedText = words.join(' ');



            document.getElementById("result").textContent = "Sorted Words: " + sortedText;
        }



        window.onload = sortWords;
    </script>
</body>
</html>
```

5 Write a script that includes a function which receives an array of names as parameter and returns the number of names in the given array that end in either ‘ie’ or ‘y’

```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Count Names</title>
</head>
<body>
    <h1>Total no.of Names Ending with 'ie' or 'y'</h1>

    <label for="namesInput">Enter names separated by commas:</label>
    <input type="text" id="namesInput" placeholder="e.g., Charlie, Julie, Molly">
    <button onclick="countNames()">Count Names</button>

    <p id="result"></p>

    <script>
        function countNames() {

            let inputText = document.getElementById("namesInput").value;

            let namesArray = inputText.split(',').map(name => name.trim());

            let count = namesArray.filter(name => name.endsWith('ie') || name.endsWith('y')).length;

            document.getElementById("result").textContent = `Number of names ending with 'ie' or 'y': ${count}`;
        }
    </script>
</body>
</html>
```

6 **Write a script that sorts an array with numbers. The order for sorting should be inputted from keyboard either as ascending or descending. Using switch array should be sorted according to inputted order.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Sort Numbers</title>
</head>
<body>
    <h1>Sort Numbers</h1>

    <label for="numbersInput">Enter numbers separated by commas:</label>
    <input type="text" id="numbersInput" placeholder="e.g., 4, 2, 9, 1, 5">

    <label for="orderSelect">Select sorting order:</label>
    <select id="orderSelect">
        <option value="ascending">Ascending</option>
        <option value="descending">Descending</option>
    </select>

    <button onclick="sortNumbers()">Sort Numbers</button>

    <p id="result"></p>

    <script>
        function sortNumbers() {

            let numbersText = document.getElementById("numbersInput").value;
            let order = document.getElementById("orderSelect").value;



            let numbersArray = numbersText.split(',').map(num => parseFloat(num.trim()));
            switch (order) {
                case 'ascending':
                    numbersArray.sort((a, b) => a - b);
                    break;
                case 'descending':
                    numbersArray.sort((a, b) => b - a);
                    break;
            }

            document.getElementById("result").textContent = `Sorted Numbers: ${numbersArray.join(', ')}`;
        }
    </script>
</body>
</html>
```

ASSIGNMENT 6

1. Show the use four pattern methods provided by JavaScript.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Pattern Matching in JavaScript</title>
</head>
<body>
  <h1>Pattern Matching Examples</h1>

  <p id="searchResult"></p>
  <p id="matchResult"></p>
  <p id="replaceResult"></p>
  <p id="splitResult"></p>

  <script>
    // 1. search() method - finds the position of the first match of a pattern
    let text = "I have 10 apples and 5 oranges.";
    let searchPattern = /\d+/; // Regex to find the first number
    let searchResult = text.search(searchPattern);
    document.getElementById("searchResult").innerHTML = "Position of the first number: " + searchResult;

    // 2. match() method - returns an array of matches
    let matchPattern = /\d+/g; // Regex to find all numbers
    let matchResult = text.match(matchPattern);
    document.getElementById("matchResult").innerHTML = "Numbers found: " + matchResult;
```

```
        // 3. replace() method - replaces matched pattern with a new string
        let replacePattern = /apples/g; // Regex to find the word 'apples'
        let replacedText = text.replace(replacePattern, "bananas");
        document.getElementById("replaceResult").innerHTML = "After replacing 'apples' with 'bananas': " + replacedText;

        // 4. split() method - splits the string into an array based on the pattern
        let splitPattern = /\s+/; // Regex to split by spaces
        let splitResult = text.split(splitPattern);
        document.getElementById("splitResult").innerHTML = "Text split into words: " + splitResult.join(", ");
    </script>
</body>
</html>
```

2. Show how an element can be accessed in different ways.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Access Elements in Different Ways</title>
</head>
<body>

    <h1>Accessing HTML Elements with JavaScript</h1>

    <!-- Example form -->
    <form name="myForm">
        <label for="name">Name:</label>
        <input type="text" id="urname" name="nameInput" value="Rama">
        <br>
```

```
  <label for="email">Email:</label>
  <input type="email" id="uremail" name="emailInput" value="rama@gmail.com">
  <br>
  <input type="button" id="myButton" value="Submit">
</form>

<p id="output"></p>

<script>
  // 1. Access using forms and elements arrays (index-based)
  let nameField1 = document.forms[0].elements[0];
  let emailField1 = document.forms[0].elements[1];


  // 2. Access using the element's name attribute
  let nameField2 = document.myForm.nameInput;
  let emailField2 = document.myForm.emailInput;


  // 3. Access using getElementById
  let nameField3 = document.getElementById("urname");
  let emailField3 = document.getElementById("uremail");


  // Displaying output on the page
  document.getElementById("output").innerHTML =
    "Name and email accessed in three ways: <br>" +
    "1.a. forms[0].elements[0]: " + nameField1.value + "<br>" +
    "1.b. forms[0].elements[1]: " + emailField1.value + "<br>" +
    "2.a. document.myForm.name: " + nameField2.value + "<br>" +
    "2.b. document.myForm.email: " + emailField2.value + "<br>" +
    "3.a. getElementById('name'): " + nameField3.value+"<br>"+
    "3.b. getElementById('email'): " + emailField3.value;
  </script>

</body>
</html>
```

3. **Using an implicit array, display the number of checkboxes related to grocery items to be purchased have been checked.**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Checkboxes</title>
</head>
<body>

  <h1>Grocery Items to Purchase</h1>

  <!-- Form with grocery item checkboxes -->
  <form id="groceryForm">
    <label><input type="checkbox" name="groceryItem" value="Milk"> Milk</label><br>
    <label><input type="checkbox" name="groceryItem" value="Bread"> Bread</label><br>
    <label><input type="checkbox" name="groceryItem" value="Butter"> Butter</label><br>
    <label><input type="checkbox" name="groceryItem" value="Fruits"> Fruits</label><br>
    <label><input type="checkbox" name="groceryItem" value="Vegetables"> Vegetables</label><br>
  </form>

  <button onclick="countCheckedItems()">Count Checked Items</button>
```

```
    <p id="output"></p>

    <script>
        function countCheckedItems() {
            // Access the implicit array of checkboxes using the 'name' attribute
            let checkboxlist = document.getElementsByName("groceryItem");

            // Count how many checkboxes are checked
            let checkedCount = 0;
            for (let i = 0; i < checkboxlist.length; i++) {
                if (checkboxlist[i].checked) {
                    checkedCount++;
                }
            }

            // Display the result
            document.getElementById("output").innerHTML =
                "You have selected " + checkedCount + " item(s) for purchase.";
        }
    </script>

</body>
</html>
```

4. **Create an HTML document that includes five radio buttons, labelled red, blue, green, yellow, and orange. The event handlers for these button must produce messages stating the chosen favourite color. The event handler must be implemented as a function whose name must be assigned to the onclick attribute of the radio button elements. The chosen color must be sent to the event handler as a parameter.**

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Select Your Favorite Color</title>
</head>
<body>

    <h1>Choose Your Favorite Color</h1>

    <!-- Radio buttons for color selection -->
    <form>
        <label><input type="radio" name="color" onclick="displayColor('Red')">
Red</label><br>
        <label><input type="radio" name="color" onclick="displayColor('Blue')">
Blue</label><br>
        <label><input type="radio" name="color" onclick="displayColor('Green')">
Green</label><br>
        <label><input type="radio" name="color" onclick="displayColor('Yellow')">
Yellow</label><br>
        <label><input type="radio" name="color" onclick="displayColor('Orange')">
Orange</label><br>
    </form>

    <!-- Area to display the selected color -->
    <p id="output"></p>

    <script>
        // Function to display the selected color
        function displayColor(color) {
            // Display the message with the chosen favorite color
            document.getElementById("output").innerHTML = "Your favorite color is " + color
+ ".";
        }
    </script>

</body>
</html>
```

5. Rewrite the above exercise to handle the events using DOM2 event model.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Select Your Favorite Color</title>
</head>
<body>

    <h1>Choose Your Favorite Color</h1>

    <!-- Radio buttons for color selection -->
    <form>
        <label><input type="radio" name="color" id="red"> Red</label><br>
        <label><input type="radio" name="color" id="blue"> Blue</label><br>
        <label><input type="radio" name="color" id="green"> Green</label><br>
        <label><input type="radio" name="color" id="yellow"> Yellow</label><br>
        <label><input type="radio" name="color" id="orange"> Orange</label><br>
    </form>

    <!-- Area to display the selected color -->
    <p id="output"></p>

    <script>
        // Function to display the selected color
        function displayColor(event) {
            // Get the id of the radio button that triggered the event
            const color =  event.target.id;
            // Display the selected color
            document.getElementById("output").innerHTML = "Your favorite color is " + color + ".";
        }

        // Adding event listeners to all the radio buttons
        document.getElementById("red").addEventListener("click", displayColor);
        document.getElementById("blue").addEventListener("click", displayColor);
        document.getElementById("green").addEventListener("click", displayColor);
        document.getElementById("yellow").addEventListener("click", displayColor);
        document.getElementById("orange").addEventListener("click", displayColor);
    </script>

</body>
</html>
```

6. **Create a JavaScript for the following: When a page is loaded, it should display 'Welcome to the Computation page' in an alert; then two numbers are provided in text boxes (Validate that they are numbers-if nonnumeric value is displayed, appropriate message should be displayed). In another text box, provide an operation to be carried out in the form of symbols (+,-,*,/,%). Validate this entry also. Include a button 'Compute' which when clicked, corresponding arithmetic operation should be carried out and result should be displayed in a text box(which should not be allowed to edit)**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Computation Page</title>

  <script>
    // This function runs when the page loads
    window.onload = function() {
      // Display welcome alert when the page is loaded
      alert('Welcome to the Computation page');
    };
```

```
// Function to validate inputs and perform the computation
function computeResult() {
    // Get the values from the input fields
    let num1 = document.getElementById("num1").value;
    let num2 = document.getElementById("num2").value;
    let operator = document.getElementById("operator").value;

    // Validate if num1 and num2 are numbers
    if (isNaN(num1) || num1 === "" || isNaN(num2) || num2 === "") {
        alert("Please enter valid numeric values for both numbers.");
        return;
    }

    // Validate if the operator is one of the valid symbols
    if (!['+', '-', '*', '/', '%'].includes(operator)) {
        alert("Please enter a valid operator (+, -, *, /, %).");
        return;
    }

    // Convert num1 and num2 to numbers
    num1 = parseFloat(num1);
    num2 = parseFloat(num2);
    let result;

    // Perform the corresponding arithmetic operation
    switch (operator) {
        case '+':
            result = num1 + num2;
            break;
        case '-':
            result = num1 - num2;
            break;
```

```
                case '*':
                    result = num1 * num2;
                    break;
                case '/':
                    if (num2 === 0) {
                        alert("Division by zero is not allowed.");
                        return;
                    }
                    result = num1 / num2;
                    break;
                case '%':
                    result = num1 % num2;
                    break;
                default:
                    alert("Invalid operation.");
                    return;
            }

            // Display the result in the read-only text box
            document.getElementById("result").value = result;
        }
    </script>
</head>
<body>

    <h1>Computation Page</h1>

    <!-- Input fields for the numbers and operator -->
    <label for="num1">Enter first number:</label>
    <input type="text" id="num1" placeholder="Number 1"><br><br>

    <label for="num2">Enter second number:</label>
    <input type="text" id="num2" placeholder="Number 2"><br><br>
    <label for="operator">Enter operator (+, -, *, /, %):</label>
    <input type="text" id="operator" maxlength="1" placeholder="Operator"><br><br>

    <!-- Button to trigger computation -->
    <button onclick="computeResult()">Compute</button><br><br>

    <!-- Read-only text box to display the result -->
    <label for="result">Result:</label>
    <input type="text" id="result" readonly><br>

</body>
</html>
```

7. Create an HTML document related to a registration to a service. Include appropriate JavaScript for the following validations:

- **Input a name (Confirm that First name, Middle name, Last name is provided with all of them being words with first letter capital)**
- **Input Age(With min value 18 and maximum value 60)**
- **Input a Mobile number (Confirm that it is an Indian Mobile number)**
- **Input Adhar Number**
- **Set a Password(Letters, digits and at least a special character is required)**
- **Confirm the password**

Display appropriate message whenever wrong entry is made.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form</title>

    <script>
    // Function to validate the form
    function validateForm() {
      // Get the values from the form fields
      let firstName = document.getElementById("firstName").value;
      let middleName = document.getElementById("middleName").value;
      let lastName = document.getElementById("lastName").value;
      let age = document.getElementById("age").value;
      let mobile = document.getElementById("mobile").value;
      let aadhar = document.getElementById("aadhar").value;
      let password = document.getElementById("password").value;
      let confirmPassword = document.getElementById("confirmPassword").value;

      // Name validation (First letter of each name capitalized)
      let namePattern = /^[A-Z][a-z]+$/;
      if (!namePattern.test(firstName) || !namePattern.test(middleName) || !namePattern.test(lastName)) {
        alert("First Name, Middle Name, and Last Name must start with a capital letter and contain only letters.");
        return false;
      }

      // Age validation (18 <= age <= 60)
      if (age < 18 || age > 60) {
        alert("Age must be between 18 and 60.");
        return false;
      }

      // Mobile number validation (Indian mobile number: 10 digits starting with 7, 8, or 9)
      let mobilePattern = /^[6789]\d{9}$/;
      if (!mobilePattern.test(mobile)) {
        alert("Please enter a valid Indian mobile number (starting with 6,7, 8, or 9 and 10 digits long).");
        return false;
      }

      // Aadhar number validation (12 digits)
      let aadharPattern = /^\d{12}$/;
      if (!aadharPattern.test(aadhar)) {
        alert("Please enter a valid 12-digit Aadhar number.");
        return false;
      }

      // Password validation (at least one letter, one number, one special character)
      let passwordPattern = /^(?=.*[A-Za-z])(?=.*\d)(?=.*[@$!%*#?&])[A-Za-z\d@$!%*#?&]{8,}$/;
      if (!passwordPattern.test(password)) {
        alert("Password must contain at least one letter, one number, and one special character.");
        return false;
      }

      // Confirm password validation
      if (password !== confirmPassword) {
        alert("Passwords do not match.");
        return false;
      }

      // If all validations pass
      alert("Registration successful!");
      return true;
    }
  </script>
</head>
<body>

  <h1>Service Registration Form</h1>
```

```
    <form onsubmit="return validateForm()">
        <!-- Name fields -->
        <label for="firstName">First Name:</label>
        <input type="text" id="firstName" placeholder="First Name" required><br><br>

        <label for="middleName">Middle Name:</label>
        <input type="text" id="middleName" placeholder="Middle Name" required><br><br>

        <label for="lastName">Last Name:</label>
        <input type="text" id="lastName" placeholder="Last Name" required><br><br>

        <!-- Age field -->
        <label for="age">Age:</label>
        <input type="number" id="age" placeholder="Age" min="18" max="60" required><br><br>

        <!-- Mobile number field -->
        <label for="mobile">Mobile Number:</label>
        <input type="text" id="mobile" placeholder="Mobile Number" required><br><br>

        <!-- Aadhar number field -->
        <label for="aadhar">Aadhar Number:</label>
        <input type="text" id="aadhar" placeholder="Aadhar Number" required><br><br>

        <!-- Password fields -->
        <label for="password">Password:</label>
        <input type="password" id="password" placeholder="Password" required><br><br>

        <label for="confirmPassword">Confirm Password:</label>
        <input type="password" id="confirmPassword" placeholder="Confirm Password" required><br><br>
        <!-- Submit button -->
        <button type="submit">Register</button>
    </form>

</body>
</html>
```

8. Show the concept of event propagation in JavaScript event handling.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Event Propagation Example</title>
    <style>
        #grandparent {
            padding: 50px;
            background-color: lightblue;
        }
        #parent {
            padding: 30px;
            background-color: lightgreen;
        }
        #child {
            padding: 10px;
            background-color: lightcoral;
        }

        p {
            font-family: Arial, sans-serif;
        }
    </style>
</head>
```

```
<body>

<div id="grandparent">
  Grandparent
  <div id="parent">
    Parent
    <div id="child">
      Child
    </div>
  </div>
</div>

<p id="log"></p>

<script>
  function logEvent(element, phase) {


    // Display event details
    const log = document.getElementById('log');
    log.innerHTML += `${element.id} clicked - ${phase} phase<br>`;
  }

  // Capturing phase listener
  document.getElementById('grandparent').addEventListener('click', function(event) {
    logEvent(this, 'Capturing');
  }, true); // true enables capturing phase

  document.getElementById('parent').addEventListener('click', function(event) {
    logEvent(this, 'Capturing');
  }, true);

  document.getElementById('child').addEventListener('click', function(event) {
    logEvent(this, 'Capturing');
  }, true);

  // Bubbling phase listener
  document.getElementById('grandparent').addEventListener('click', function(event) {
    logEvent(this, 'Bubbling');
  });

  document.getElementById('parent').addEventListener('click', function(event) {
    logEvent(this, 'Bubbling');
  });

  document.getElementById('child').addEventListener('click', function(event) {
    logEvent(this, 'Bubbling');
  });

  // Stopping propagation on child
  document.getElementById('child').addEventListener('click', function(event) {
    event.stopPropagation(); // Prevents event from bubbling up
    logEvent(this, 'Propagation stopped');
  });
</script>

</body>
</html>
```

ASSIGNMENT 7

1. Show the use of various values of *position* property.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>CSS Position Property Examples</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 20px;
    }
    h1 {
      text-align: center;
    }
    .static {
      background-color: lightblue; /* Blue */
      padding: 10px;
      margin: 10px 0;
    }
    .relative {
      position: relative;
      top: 10px;   /* Moves down 10px */
      left: 20px;  /* Moves right 20px */
      background-color: lightgreen; /* Green */
      padding: 10px;
      margin: 10px 0;
    }
```

```
        .absolute-container {
            position: relative; /* Sets context for absolute positioning */
            border: 1px solid #ccc;
            padding: 20px;
            margin: 10px 0;
        }
        .absolute {
            position: absolute;
            top: 55px;   /* 50px from the top of the nearest positioned ancestor */
            left: 30px;  /* 30px from the left */
            background-color: lightcoral; /* Coral */
            padding: 10px;
        }
        .fixed {
            position: fixed;
            bottom: 10px; /* 10px from the bottom of the viewport */
            right: 10px;  /* 10px from the right of the viewport */
            background-color: lightgoldenrodyellow; /* Goldenrod */
            padding: 10px;
        }
        .sticky {
            position: sticky;
            top: 0; /* Sticks to the top when scrolling */
            background-color: lightpink; /* Pink */
            padding: 10px;
            margin: 10px 0;
        }
    </style>
</head>
<body>
    <h1>CSS Position Property Examples</h1>

    <!-- Static Positioning -->
    <div class="static">
        <h2>Static Positioning</h2>
        <p>This is a default positioned element. It flows naturally in the document.</p>
    </div>

     <!-- Sticky Positioning -->
    <div class="sticky">
        <h2>Sticky Positioning</h2>
        <p>This element sticks to the top of the viewport when you scroll down.</p>
    </div>

    <!-- Relative Positioning -->
    <div class="relative">
        <h2>Relative Positioning</h2>
        <p>This element is shifted 10px down and 20px right from its original position.</p>
    </div>

    <!-- Absolute Positioning -->
    <div class="absolute-container">
        <h2>Absolute Positioning</h2>
        <p>This is a container for absolute positioning.</p>
        <div class="absolute">
            This element is positioned 50px from the top and 30px from the left of its container.
        </div>
    </div>

    <!-- Fixed Positioning -->
    <div class="fixed">
        <h2>Fixed Positioning</h2>
        <p>This element is always visible at the bottom right of the viewport, even when scrolling.</p>
    </div>
```

```
  <div style="height: 800px;"></div> <!-- Extra height to enable scrolling -->
</body>
</html>
```

2. **Create an HTML document that includes a paragraph of text and a button. When button is clicked, paragraph should be moved to a place whose x and y coordinates are inputted in text boxes.**

```
      <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Move Paragraph Example</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }
    #movable {
      position: absolute; /* Allows the paragraph to be positioned based on coordinates
*/
      left: 100px; /* Initial position */
      top: 200px;  /* Initial position */
      background-color: lightblue;
      padding: 10px;
      transition: left 0.5s, top 0.5s; /* Smooth transition effect */
    }
  </style>
</head>
<body>
```

```
    <h1>Move the Paragraph</h1>
    <p id="movable">This is a movable paragraph. Enter coordinates to move me!</p>

    <label for="x">X Coordinate:</label>
    <input type="number" id="x" placeholder="Enter X" />
    <br /><br />
    <label for="y">Y Coordinate:</label>
    <input type="number" id="y" placeholder="Enter Y" />
    <br /><br />
    <button id="moveButton">Move Paragraph</button>

    <script>
        document.getElementById("moveButton").onclick = function() {
            // Get the values from input fields
            const x = document.getElementById("x").value;
            const y = document.getElementById("y").value;

            // Set the new position of the paragraph
            const paragraph = document.getElementById("movable");
            paragraph.style.left = x + 'px';
            paragraph.style.top = y + 'px';
        };
    </script>

</body>
</html>
```

3. **Create a form as specified in previous assignment (Without validation). Page should include a help box that displays appropriate message to assist in making entry to the form.**

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

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Registration Form with Help Box</title>
<style>
  body {
    font-family: Arial, sans-serif;
    background-color: lightgrey;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
  }

  .form-container {
    background-color: white;
    padding: 30px;  /* Increased padding for better spacing */
    border: 2px solid black;
    width: 300px;  /* Fixed width for the form */
  }

  h2 {
    margin: 5px;  /* Spacing below the heading */
    text-align: center;   /* Centered heading */
  }

  input[type="text"], input[type="number"], input[type="password"] {
    width: calc(100% - 22px);  /* Adjusted width for padding */
    padding: 10px;
    margin: 10px 0;  /*calc(100% - 22px);  Added margin for spacing */
    border: 1px solid #ccc;
  }
```

```
        input[type="submit"] {
            background-color: slateblue;
            color: white;
            padding: 10px;
            cursor: pointer;
            width: 100%;  /* Full width for the submit button */
        }

        input[type="submit"]:hover {
            background-color: darkblue;
        }

        #help-box {
            color: red;
            font-style: italic;
            margin-top: 10px;
            text-align: center;  /* Centered help box text */
        }
    </style>
</head>
<body>

<div class="form-container">
    <h2>Registration Form</h2>
    <form>
        Name: <input type="text" id="name" onfocus="showHelp('Enter your full name (First Middle Last)')" onblur="clearHelp()"><br>
        Age: <input type="number" id="age" onfocus="showHelp('Enter your age')" onblur="clearHelp()"><br>
        Mobile: <input type="text" id="mobile" onfocus="showHelp('Enter your 10-digit mobile number')" onblur="clearHelp()"><br>
        Aadhar: <input type="text" id="aadhar" onfocus="showHelp('Enter your 12-digit Aadhar number')" onblur="clearHelp()"><br>
        Password: <input type="password" id="password" onfocus="showHelp('Set a password (include letters, digits, and special characters)')" onblur="clearHelp()"><br>
        Confirm Password: <input type="password" id="confirmPassword"
onfocus="showHelp('Confirm your password')" onblur="clearHelp()"><br>
        <input type="submit" value="Submit">
    </form>

    <p id="help-box"></p>
</div>

<script>
function showHelp(message) {
    document.getElementById('help-box').innerHTML = message;
}

function clearHelp() {
    document.getElementById('help-box').innerHTML = '';
}
</script>

</body>
</html>
```

4. Display stacking of four paragraphs. When a paragraph is clicked, it should be displayed on top of other paragraphs.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Registration Form with Help Box</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: lightgrey;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
}

.form-container {
    background-color: white;
    padding: 30px;  /* Increased padding for better spacing */
    border: 2px solid black;
    width: 300px;  /* Fixed width for the form */
}

h2 {
    margin: 5px;  /* Spacing below the heading */
    text-align: center;   /* Centered heading */
}

input[type="text"], input[type="number"], input[type="password"] {
    width: calc(100% - 22px);  /* Adjusted width for padding */
    padding: 10px;
    margin: 10px 0;  /*calc(100% - 22px);  Added margin for spacing */
    border: 1px solid #ccc;
}

input[type="submit"] {
    background-color: slateblue;
    color: white;
    padding: 10px;
    cursor: pointer;
    width: 100%;  /* Full width for the submit button */
}
```

```
    input[type="submit"]:hover {
      background-color: darkblue;
    }

    #help-box {
      color: red;
      font-style: italic;
      margin-top: 10px;
      text-align: center;  /* Centered help box text */
    }
  </style>
</head>
<body>

<div class="form-container">
  <h2>Registration Form</h2>
  <form>
    Name: <input type="text" id="name" onfocus="showHelp('Enter your full name (First Middle Last)')" onblur="clearHelp()"><br>
    Age: <input type="number" id="age" onfocus="showHelp('Enter your age')" onblur="clearHelp()"><br>
    Mobile: <input type="text" id="mobile" onfocus="showHelp('Enter your 10-digit mobile number')" onblur="clearHelp()"><br>
    Aadhar: <input type="text" id="aadhar" onfocus="showHelp('Enter your 12-digit Aadhar number')" onblur="clearHelp()"><br>
    Password: <input type="password" id="password" onfocus="showHelp('Set a password (include letters, digits, and special characters)')" onblur="clearHelp()"><br>
    Confirm Password: <input type="password" id="confirmPassword" onfocus="showHelp('Confirm your password')" onblur="clearHelp()"><br>
    <input type="submit" value="Submit">
  </form>

  <p id="help-box"></p>
</div>
```

```
<script>
function showHelp(message) {
  document.getElementById('help-box').innerHTML = message;
}

function clearHelp() {
  document.getElementById('help-box').innerHTML = '';
}
</script>

</body>
</html>
```

5. **Create a page that includes a Heading 'Welcome to NMAMIT,Nitte'. Change the color and font properties of it as mouse is moved over it. Also include a paragraph, on which when mouse is moved mouse pointer's X and Y coordinates with respect to browser should be displayed in text boxes which are included in the page.**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Welcome to NMAMIT, Nitte</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color:lightgrey;
    }

    h1 {
      transition: color 0.3s ease, font-size 0.3s ease;
    }
```

```
        h1:hover {
            color: yellow;
            font-size: 3em;
            text-shadow: 2px 2px 5px rgba(0, 0, 0, 0.3);
        }

        p {
            margin: 20px 0;
            padding: 10px;

            background-color: white;
            cursor: pointer;
        }

        .coordinates {
            display: flex;
            flex-direction: column;
            margin-top: 20px;
        }

        .coordinates label {
            margin-top: 10px;
        }

        .coordinates input {
            margin: 5px 0;
            padding: 8px;
            width: 150px;
        }
    </style>
</head>
<body>
```

```
    <h1>Welcome to NMAMIT, Nitte</h1>
    <p id="coordinateArea">Hover over this paragraph to see the mouse coordinates.</p>

    <div class="coordinates">
        <label for="mouseX">Mouse X:</label>
        <input type="text" id="mouseX" placeholder="Mouse X" readonly>

        <label for="mouseY">Mouse Y:</label>
        <input type="text" id="mouseY" placeholder="Mouse Y" readonly>
    </div>

    <script>
        const coordinateArea = document.getElementById('coordinateArea');
        const mouseXInput = document.getElementById('mouseX');
        const mouseYInput = document.getElementById('mouseY');

        coordinateArea.addEventListener('mousemove', (event) => {
            const x = event.clientX;
            const y = event.clientY;

            mouseXInput.value = x;
            mouseYInput.value = y;
        });

        coordinateArea.addEventListener('mouseleave', () => {
            mouseXInput.value = '';
            mouseYInput.value = '';
        });
    </script>
</body>
</html>
```

6. **The document must have a small image of yourself, which must appear at the position of the mouse cursor when the mouse button is clicked, regardless of the position of the cursor at the time.**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Click to Show Image</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      display: flex;
      flex-direction: column;
      align-items: center;
      justify-content: center;
      height: 100vh;
      margin: 0;
      position: relative;
    }

    h1 {
      margin-bottom: 20px;
    }

    p {
      margin: 20px 0;
      padding: 10px;
      border: 1px solid #ccc;
      background-color: white;
      cursor: pointer;
    }
    #myImage {
      position: absolute;
      width: 300px; /* Adjust size as needed */
      height: 200px;
      display: none; /* Hidden by default */
    }
  </style>
</head>
<body>
  <h1>Click to Show Image</h1>
  <p>Click anywhere on this page to display your image!</p>

  <img id="myImage" src="/home/mca/Documents/AWT63/Assignment 7/girl.jpg" alt="My Image"> <!-- Replace with your image URL -->

  <script>
    const myImage = document.getElementById('myImage');

    // Show image at mouse position on click
    document.addEventListener('click', (event) => {
      myImage.style.display = 'block'; // Show the image
      myImage.style.left = `${event.pageX}px`; // Position it at the click X
      myImage.style.top = `${event.pageY}px`; // Position it at the click Y
    });
  </script>
</body>
</html>
```

7. Create an HTML document with text 'Welcome to NMAMAIT Nitte' which should be moved. The direction of movement is controlled by four buttons: Move Left, Move Down, Move Right and Move Up.

```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Move Text Example</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            display: flex;
            flex-direction: column;
            align-items: center;
            justify-content: center;
            height: 100vh;
            margin: 0;
        }

        #movingText {
            position: relative;
            font-size: 24px;
            padding: 10px;
            background-color: white;
            border: 1px solid #ccc;
            transition: transform 0.1s;
        }

        .button-container {
            margin-top: 20px;
        }

        button {
            margin: 5px;
            padding: 10px 15px;
            font-size: 16px;
            cursor: pointer;
        }
    </style>
</head>
<body>
    <div id="movingText">Welcome to NMAMIT Nitte</div>

    <div class="button-container">
        <button onclick="moveLeft()">Move Left</button>
        <button onclick="moveDown()">Move Down</button>
        <button onclick="moveRight()">Move Right</button>
        <button onclick="moveUp()">Move Up</button>
    </div>

    <script>
        const movingText = document.getElementById('movingText');
        let posX = 0;
        let posY = 0;

        function moveLeft() {
            posX -= 10;
            updatePosition();
        }

        function moveRight() {
            posX += 10;
            updatePosition();
        }

        function moveUp() {
            posY -= 10;
            updatePosition();
    }

    function moveDown() {
      posY += 10;
      updatePosition();
    }

    function updatePosition() {
      movingText.style.transform = `translate(${posX}px, ${posY}px)`;
    }
  </script>
</body>
</html>
```