10. Apply the usage While Loop, External JavaScript Files and do Loop in JavaScript

P10.htm

```
<!DOCTYPE html>
<html>
<head>
  <title>Loop Demo with While & Do-While</title>
 <style>
  body {
    font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    background: linear-gradient(to right, #83a4d4, #b6fbff);
    margin: 0;
    padding: 50px 20px;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
  }
  .container {
    background: #fff;
    padding: 40px;
    border-radius: 16px;
    box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);
    width: 100%;
    max-width: 500px;
    text-align: center;
  }
  h2 {
    color: #333;
    margin-bottom: 20px;
  }
```

```
input[type="text"] {
  padding: 12px;
  font-size: 16px;
  width: 80%;
  max-width: 200px;
  border: 2px solid #ccc;
  border-radius: 6px;
  transition: border-color 0.3s;
}
input[type="text"]:focus {
  border-color: #007bff;
  outline: none;
}
button {
  padding: 12px 20px;
  margin: 15px 10px 0 10px;
  background-color: #007bff;
  color: white;
  border: none;
  border-radius: 6px;
  font-size: 16px;
  cursor: pointer;
  transition: background-color 0.3s, transform 0.2s;
}
button:hover {
  background-color: #0056b3;
  transform: translateY(-2px);
}
.output {
  margin-top: 25px;
```

```
      background-color: #f1f1f1;
      padding: 20px;
      border-radius: 8px;
      font-size: 16px;
      color: #333;
      box-shadow: inset 0 0 10px rgba(0,0,0,0.05);
      white-space: pre-wrap;
    }
    label {
      font-weight: bold;
      color: #222;
      display: block;
      margin-bottom: 10px;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>While & Do-While Loop Demo</h2>
    <label>Enter a number (1 to 10):</label>
    <input type="text" id="numInput" placeholder="e.g. 5" />
    <div>
      <button onclick="generateFibonacci()">Fibonacci (While)</button>
      <button onclick="calculateFactorial()">Factorial (Do-While)</button>
    </div>
    <div class="output" id="output"></div>
  </div>
  <script src="s10.js"></script>
</body>
</html>
```

S10.js

```
function getValidNumber() {
  const input = document.getElementById("numInput").value.trim();
  const num = parseInt(input);
  if (!/^\d+$/.test(input) || isNaN(num) || num < 1 || num > 10) {
    alert("Please enter a valid whole number between 1 and 10.");
    return null;
  }
  return num;
}
// Fibonacci using WHILE loop
function generateFibonacci() {
  const n = getValidNumber();
  if (n === null) return;
  let output = "Fibonacci series (While loop):<br>";
  let a = 0, b = 1, count = 0;
  while (count < n) {
    output += a + " ";
    let next = a + b;
    a = b;
    b = next;
    count++;
  }
  document.getElementById("output").innerHTML = output;
}
// Factorial using DO-WHILE loop
function calculateFactorial() {
  const n = getValidNumber();
  if (n === null) return;
  let factorial = 1;
```

```
    let i = 1;
    do {
        factorial *= i;
        i++;
    } while (i <= n);
    document.getElementById("output").innerHTML =
        `Factorial of ${n} (Do-While loop):<br>${factorial}`;
}
```

11. Apply the usage of Radio Buttons, Checkboxes, for Loop - fieldset and legend Elements in JavaScript

P11.htm

```
<!DOCTYPE html>
<html>
<head>
  <title>Styled Form Example</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: #f0f4f8;
      padding: 30px;
    }
    form {
      background: #ffffff;
      padding: 25px 30px;
      border-radius: 12px;
      box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
      max-width: 400px;
      margin: auto;
    }
    fieldset {
      border: 2px solid #4a90e2;
      border-radius: 8px;
      margin-bottom: 20px;
      padding: 15px;
    }
    legend {
      font-size: 1.2em;
      font-weight: bold;
      color: #4a90e2;
      padding: 0 10px;
    }
    label {
      display: block;
      margin-bottom: 8px;
      font-size: 1em;
      color: #333;
    }
    button {
      background-color: #4a90e2;
      color: white;
      padding: 10px 20px;
      border: none;
      border-radius: 6px;
      font-size: 1em;
      cursor: pointer;
      transition: background 0.3s ease;
    }
    button:hover {
      background-color: #357ab8;
    }
    #output {
      margin-top: 20px;
      font-size: 1.1em;
      color: #2e2e2e;
      background-color: #e9f5ff;
      padding: 10px 15px;
      border-left: 4px solid #4a90e2;
      border-radius: 6px;
    }
  </style>
</head>
<body>
<form id="userForm">
  <fieldset>
    <legend>Select Your Gender:</legend>
    <label><input type="radio" name="gender" value="Male"> Male</label>
    <label><input type="radio" name="gender" value="Female"> Female</label>
    <label><input type="radio" name="gender" value="Other"> Other</label>
  </fieldset>
  <fieldset>
    <legend>Select Your Hobbies:</legend>
    <label><input type="checkbox" name="hobby" value="Reading"> Reading</label>
    <label><input type="checkbox" name="hobby" value="Traveling"> Traveling</label>
    <label><input type="checkbox" name="hobby" value="Gaming"> Gaming</label>
    <label><input type="checkbox" name="hobby" value="Cooking"> Cooking</label>
  </fieldset>
  <button type="button" onclick="showSelections()">Submit</button>
</form>
<p id="output"></p>
<script>
function showSelections() {
  let gender = document.querySelector('input[name="gender"]:checked');
  let hobbies = document.querySelectorAll('input[name="hobby"]:checked');
  let output = "Selected Gender: ";
  output += gender ? gender.value : "None";
  output += "<br>Selected Hobbies: ";
  for (let i = 0; i < hobbies.length; i++) {
    output += hobbies[i].value;
    if (i < hobbies.length - 1) {
      output += ", ";
    }
  }
  document.getElementById("output").innerHTML = output;
}
</script>
</body>
</html>
```

12. Use Pull-Down Menus, List Boxes with Event Handler and Listener in JavaScript

P12.htm

```
<!DOCTYPE html>
<html>
<head>
  <title>Dropdown & List Box with Event Handler and Listener</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: linear-gradient(to right, #e0c3fc, #8ec5fc);
      margin: 0;
      padding: 40px;
      display: flex;
      justify-content: center;
      align-items: center;
      min-height: 100vh;
    }
    .container {
      background: white;
      padding: 30px;
      border-radius: 10px;
      box-shadow: 0 8px 20px rgba(0, 0, 0, 0.1);
      width: 400px;
    }
    h2 {
      text-align: center;
      color: #333;
    }
    label {
      font-weight: bold;
      margin-top: 15px;
      display: block;
    }
    select {
      width: 100%;
      padding: 10px;
      margin: 10px 0;
      font-size: 16px;
      border-radius: 5px;
      border: 1px solid #ccc;
    }
    .output {
      margin-top: 20px;
      padding: 10px;
```

```
      background-color: #f9f9f9;
      border-left: 5px solid #007bff;
      font-size: 16px;
    }
  </style>
</head>
<body>
  <div class="container">
    <h2>Choose a Color and Fruit</h2>

    <label for="colorSelect">Select a Color:</label>
    <select id="colorSelect">
      <option value="">-- Select Color --</option>
      <option value="Red">Red</option>
      <option value="Green">Green</option>
      <option value="Blue">Blue</option>
    </select>
    <label for="fruitList">Select a Fruit:</label>
    <select id="fruitList" size="4">
      <option value="Apple">Apple</option>
      <option value="Banana">Banana</option>
      <option value="Mango">Mango</option>
      <option value="Orange">Orange</option>
    </select>
    <div id="output" class="output">Your selection will appear here.</div>
  </div>
  <script src="12.js"></script>
</body>
</html>
```

12.js

```
window.onload = function () {
  const colorSelect = document.getElementById("colorSelect");
  const fruitList = document.getElementById("fruitList");
  const output = document.getElementById("output");
  // Using Event Handler for dropdown (color)
  colorSelect.onchange = function () {
    displaySelection();
  };
  // Using Event Listener for list box (fruit)
  fruitList.addEventListener("change", function () {
    displaySelection();
  });
  function displaySelection() {
    const color = colorSelect.value;
    const fruit = fruitList.value;
    if (color && fruit) {
      output.textContent = `Selected Color: ${color}, Selected Fruit: ${fruit}`;
    } else if (color) {
      output.textContent = `Selected Color: ${color}`;
    } else if (fruit) {
      output.textContent = `Selected Fruit: ${fruit}`;
    } else {
      output.textContent = "Please select both color and fruit.";
    }
  }
};
```