Q1. Create a page with email field, phone field and password field with below checks.

● email check

● phone number check

● password check (std validations like min 8chars, 1lowercase, 1uppercase, 1special char)

Code

Html code

<!DOCTYPE html>

<html>

<head>

    <title>Registration Form</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <h1>Registration Form</h1>

    <form method="post" action="">

        <label for="email">Email:</label>

        <input type="text" id="email" name="email" placeholder="kumari1234@gmail.com" required><br>

        <span id="email-error"></span><br>

        <label for="phone">Phone:</label>

        <input type="text" id="phone" name="phone" placeholder="(123) 456-7890" required><br>

        <span id="phone-error"></span><br>

        <label for="password">Password:</label>

        <input type="password" id="password" name="password" required><br>

        <span id="password-error"></span><br>

        <input type="submit" value="Submit">

    </form>

    <script>

        const form = document.querySelector("form");

        const emailInput = document.getElementById("email");

        const emailError = document.getElementById("email-error");

        const phoneInput = document.getElementById("phone");

        const phoneError = document.getElementById("phone-error");

        const passwordInput = document.getElementById("password");

        const passwordError = document.getElementById("password-error");

        form.addEventListener("submit", function(event) {

            if (!validateEmail()) {

                event.preventDefault();

            }

            if (!validatePhone()) {

                event.preventDefault();

            }

            if (!validatePassword()) {

                event.preventDefault();

            }

        });

        function validateEmail() {

            const emailPattern = /^\w+@[a-zA-Z\_]+?\.[a-zA-Z]{2,3}$/;

            if (emailInput.value.match(emailPattern)) {

                emailError.textContent = "";

                return true;

            } else {

                emailError.textContent = "Please enter a valid email address.";

                return false;

            }

        }

        function validatePhone() {

            const phonePattern = /^\(\d{3}\)\s\*\d{3}-\d{4}$/;

            if (phoneInput.value.match(phonePattern)) {

                phoneError.textContent = "";

                return true;

            } else {

                phoneError.textContent = "Please enter a valid phone number.";

                return false;

            }

        }

        function validatePassword() {

            const passwordPattern = /^(?=.\d)(?=.[a-z])(?=.[A-Z])(?=.[!@#$%^&\*()\_+]).{8,}$/;

            if (passwordInput.value.match(passwordPattern)) {

                passwordError.textContent = "";

                return true;

            } else {

                passwordError.textContent = "Password must be at least 8 characters long and include at least one number, one uppercase letter, one lowercase letter, and one special character (!@#$%^&\*()\_+).";

                return false;

            }

        }

    </script>

</body>

</html>

CSS code

\*{

    margin: 0;

    padding: 0;

    font-family: 'Roboto', sans-serif;

}

form {

    width: 400px;

    margin: 0 auto;

    padding: 20px;

    background-color: #f1f1f1;

    border: 1px solid #ccc;

    border-radius: 5px;

  }

  h1 {

    text-align: center;

    margin-top: 50px;

    margin-bottom: 30px;

  }

  label {

    display: block;

    margin-bottom: 5px;

  }

  input[type="text"],

  input[type="password"] {

    width: 100%;

    padding: 10px;

    margin-bottom: 10px;

    border: 1px solid #ccc;

    border-radius: 3px;

    box-sizing: border-box;

  }

  input[type="submit"] {

    background-color: #4CAF50;

    color: white;

    border: none;

    padding: 10px 20px;

    border-radius: 3px;

    cursor: pointer;

    font-size: 16px;

  }

  input[type="submit"]:hover {

    background-color: purple;

  }

  span {

    color: red;

    font-size: 0.8em;

    font-style: italic;

  }

Q2. Create a page to

● Show some data on table

● use data table and add features: sorting, search functionality

Code

Html code

<!DOCTYPE html>

<html>

<head>

  <title>My Data Table</title>

  <link rel="stylesheet" href="style.css">

  <meta charset="utf-8">

  <link rel="stylesheet" href="https://cdn.datatables.net/1.11.3/css/jquery.dataTables.min.css">

</head>

<body>

  <h1>My Data Table</h1>

  <table id="myTable" class="display">

    <thead>

      <tr>

        <th>Name</th>

        <th>Age</th>

        <th>City</th>

        <th>Country</th>

      </tr>

    </thead>

    <tbody>

      <tr>

        <td>Ram</td>

        <td>30</td>

        <td>New York</td>

        <td>USA</td>

      </tr>

      <tr>

        <td>Shyam</td>

        <td>25</td>

        <td>London</td>

        <td>UK</td>

      </tr>

      <tr>

        <td>Prince</td>

        <td>40</td>

        <td>Paris</td>

        <td>France</td>

      </tr>

      <tr>

        <td>Satyam</td>

        <td>35</td>

        <td>Tokyo</td>

        <td>Japan</td>

      </tr>

    </tbody>

  </table>

  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

  <script src="https://cdn.datatables.net/1.11.3/js/jquery.dataTables.min.js"></script>

  <script>

    $(document).ready(function() {

       // Initialize the DataTable plugin with sorting and search functionalit

      $('#myTable').DataTable({

        // Enable sorting on all columns by default

        "order": [],

        // Add search functionality with placeholder text

        "language": {

          "searchPlaceholder": "Search..."

        }

      });

      // Add event listeners to allow sorting on all columns

      $('#myTable th').on('click', function() {

        var columnIndex = $(this).index();

        $('#myTable').DataTable().order([columnIndex, 'asc']).draw();

        $(this).removeClass('sorting\_desc').addClass('sorting');

      });

      // Add event listeners to allow reverse sorting on all columns

      $('#myTable th.sorting').on('click', function() {

        var columnIndex = $(this).index();

        $('#myTable').DataTable().order([columnIndex, 'desc']).draw();

        $(this).removeClass('sorting').addClass('sorting\_desc');

      });

    });

  </script>

</body>

</html>

CSS code

/\* Styling for the search input \*/

.dataTables\_filter input {

  border: 1px solid #ccc;

  padding: 5px;

  border-radius: 4px;

}

/\* Styling for the table header \*/

#myTable th {

  cursor: pointer;

}

/\* Styling for the sorted column header \*/

.sorting:after {

  content: "\25b2";

}

/\* Styling for the reverse sorted column header \*/

.sorting\_desc:after {

  content: "\25bc";

}

Q3. WRITE JS code with below logic in place

● sort a unsorted array of numbers

● a function that takes in an array of numbers and returns the second highest and second lowest numbers in the array

● Create a function that takes in two arrays of numbers and returns a new array that contains only the elements that are common to both array

Code

JS code

// sort an unsorted array of numbers

const unsortedArray = [4, 1, 9, 2, 7];

const sortedArray = unsortedArray.sort((a, b) => a - b);

console.log(sortedArray); // output: [1, 2, 4, 7, 9]

// function to find second highest and second lowest numbers in an array

function findSecondHighestAndLowest(numbers) {

const sortedNumbers = numbers.sort((a, b) => a - b);

const secondHighest = sortedNumbers[sortedNumbers.length - 2];

const secondLowest = sortedNumbers[1];

return [secondHighest, secondLowest];

}

const arrayToFind = [1, 4, 7, 2, 9];

console.log(findSecondHighestAndLowest(arrayToFind)); // output: [7, 2]

// function to find common elements in two arrays

function findCommonElements(array1, array2) {

const commonElements = [];

for (let i = 0; i < array1.length; i++) {

if (array2.includes(array1[i])) {

commonElements.push(array1[i]);

}

}

return commonElements;

}

const firstArray = [1, 2, 3, 4, 5];

const secondArray = [4, 5, 6, 7, 8];

console.log(findCommonElements(firstArray, secondArray)); // output: [4, 5]