

Week Nine: Designing Input Forms with Email and Phone Number Validation using HTML, CSS, JavaScript, and Validator.js

Ritika Taphasvi G | CSE FC | 230701266

Aim

To design an input form that validates email and phone number entries and displays appropriate error messages using HTML, CSS, JavaScript, and Validator.js.

Procedure

1. **Create the HTML Form:**
 - Add input fields for email and phone number.
 - Include `` elements to display error messages.
 - Link external CSS (`style.css`) and JavaScript (`script.js`) files.
 2. **Style the Form using CSS:**
 - Centre the form on the page.
 - Style the input fields, button, and error messages for a clean look.
 3. **Add Validation using JavaScript:**
 - Use the **Validator.js** library to validate the email and phone number.
 - Show error messages if the inputs are invalid.
 - If inputs are valid, log the values to the console.
-

Program Code

1. HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Form Validation</title>
  <link rel="stylesheet" href="style.css">
</head>
```

```

<body>
  <div class="container">
    <form id="myForm">
      <label for="email">Email:</label>
      <input type="email" id="email" name="email" required>
      <span id="emailError" class="error"></span>

      <label for="phone">Phone Number:</label>
      <input type="text" id="phone" name="phone" required>
      <span id="phoneError" class="error"></span>

      <button type="submit">Submit</button>
    </form>
  </div>

  <script
src="https://cdnjs.cloudflare.com/ajax/libs/validator/13.6.0/validator.min.js"></script>
  <script src="script.js"></script>
</body>
</html>

```

2. CSS

```

body {
  font-family: Arial, sans-serif;
  background-color: #f4f4f4;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
  margin: 0;
}

.container {
  background-color: white;
  padding: 20px;
  border-radius: 5px;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

form {
  display: flex;
  flex-direction: column;
}

label {
  margin-bottom: 5px;
}

```

```

input {
  margin-bottom: 10px;
  padding: 10px;
  border: 1px solid #ccc;
  border-radius: 3px;
}

button {
  padding: 10px;
  background-color: #28a745;
  color: white;
  border: none;
  border-radius: 3px;
  cursor: pointer;
}

button:hover {
  background-color: #218838;
}

.error {
  color: red;
  font-size: 0.875em;
}

```

3. JavaScript

```

/* script.js */
document.getElementById('myForm').addEventListener('submit', function (e) {
  e.preventDefault();

  let email = document.getElementById('email').value;
  let phone = document.getElementById('phone').value;
  let emailError = document.getElementById('emailError');
  let phoneError = document.getElementById('phoneError');

  // Clear previous error messages
  emailError.textContent = "";
  phoneError.textContent = "";

  // Validate email
  if (!validator.isEmail(email)) {
    emailError.textContent = 'Please enter a valid email address.';
  }

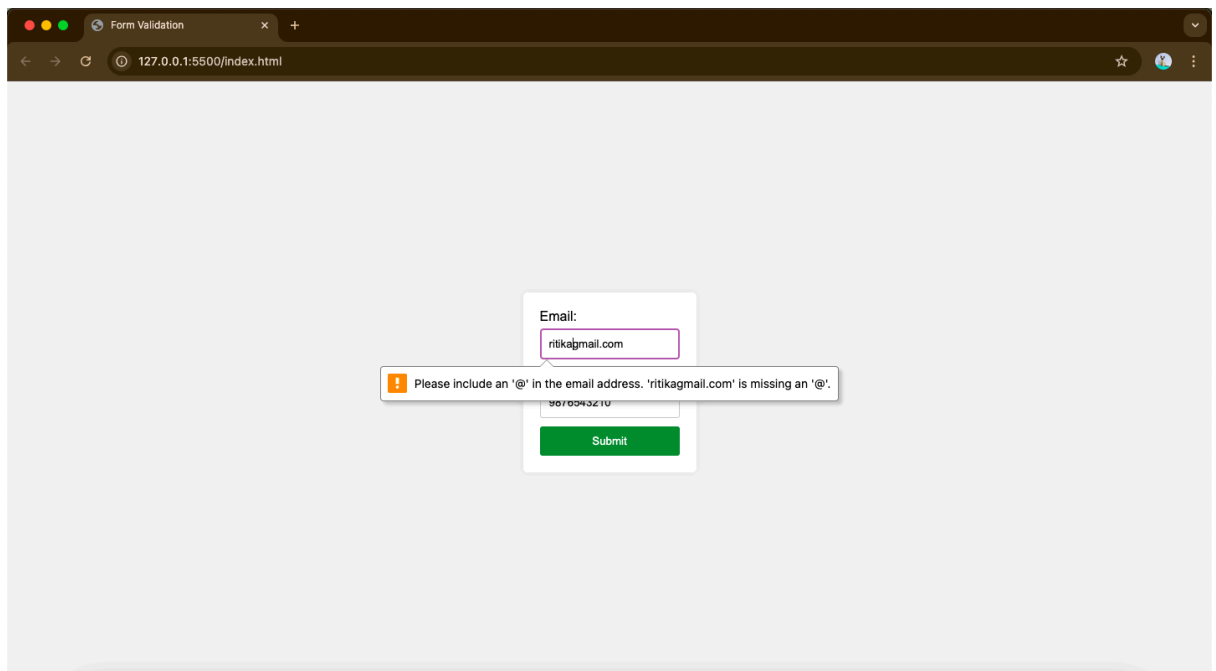
  // Validate phone number
  if (!validator.isMobilePhone(phone, 'any')) {
    phoneError.textContent = 'Please enter a valid phone number.';
  }
}

```

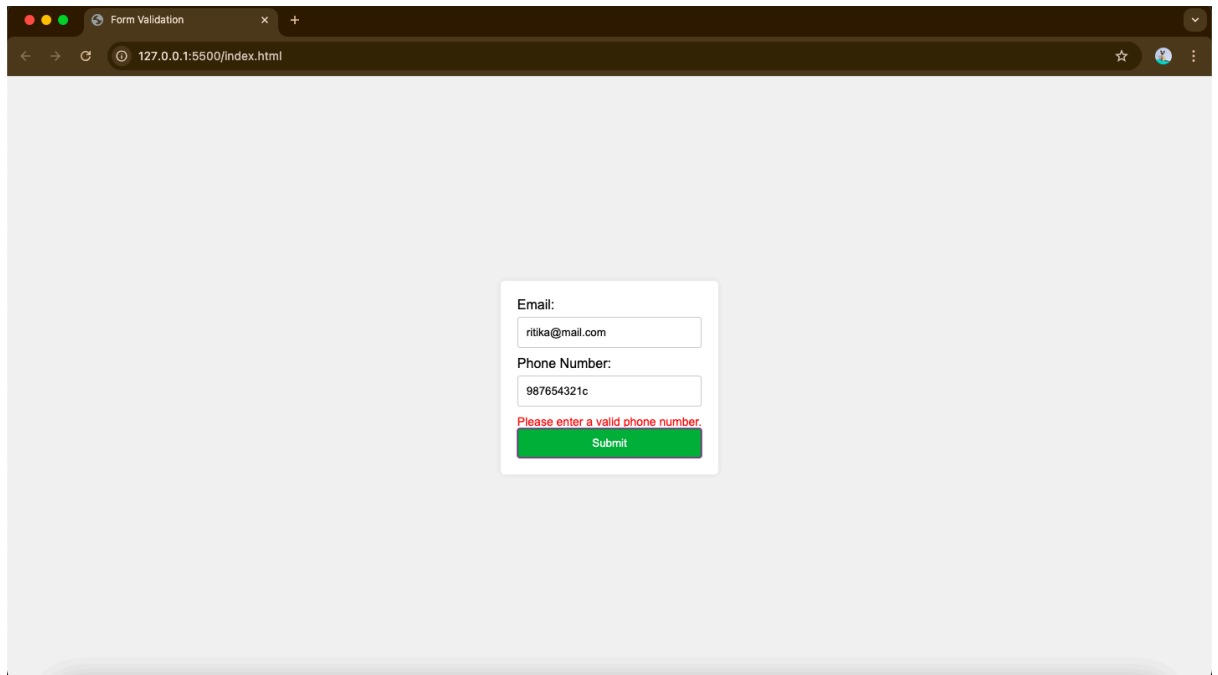
```
// If no errors, submit the form (for demonstration purposes, we'll just log the values)
if (validator.isEmail(email) && validator.isMobilePhone(phone, 'any')) {
  console.log('Email:', email);
  console.log('Phone:', phone);
}
});
```

Output

1. When the wrong email is entered:

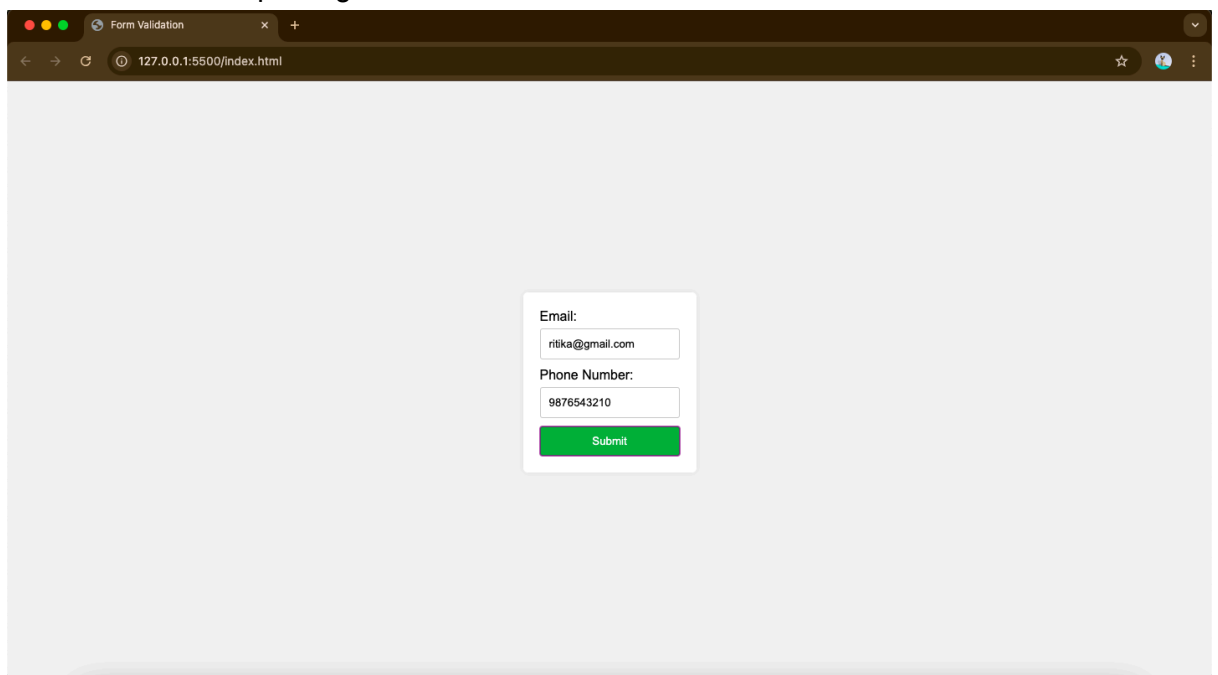


2. When an incorrect mobile number is entered:



The screenshot shows a web browser window titled "Form Validation" with the address bar displaying "127.0.0.1:5500/index.html". The form contains two input fields: "Email:" with the value "ritika@mail.com" and "Phone Number:" with the value "987654321c". Below the phone number field, a red error message reads "Please enter a valid phone number." A green "Submit" button is located at the bottom of the form.

3. When the correct input is given:



The screenshot shows the same web browser window as before, but the "Phone Number:" field now contains the correct value "9876543210". The red error message is no longer present, and the green "Submit" button remains visible at the bottom of the form.

Result

A fully functional and styled form that successfully validates email and phone number inputs, displaying clear error messages when invalid data is entered.
