Exercise 15:

Design input forms that validate data (e.g., email, phone number) and display error messages using HTML/CSS, JavaScript (with Validator.js)

AIM:

The aim is to design input forms that validate data, such as email and phone number, and display error messages using HTML/CSS and JavaScript with Validator.js.

PROCEDURE:

Step 1: Setting Up the HTML Form

Start by creating an HTML form with input fields for the email and phone number.

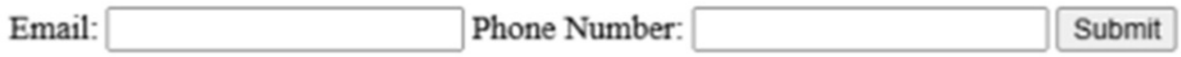
html

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta ch□   arset="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>
```

OUTPUT:



Step 2: Styling the Form with CSS

Next, add some basic styling to make the form look nice.

```
css
            style.cs
□           s */
       /*
       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;
}
```

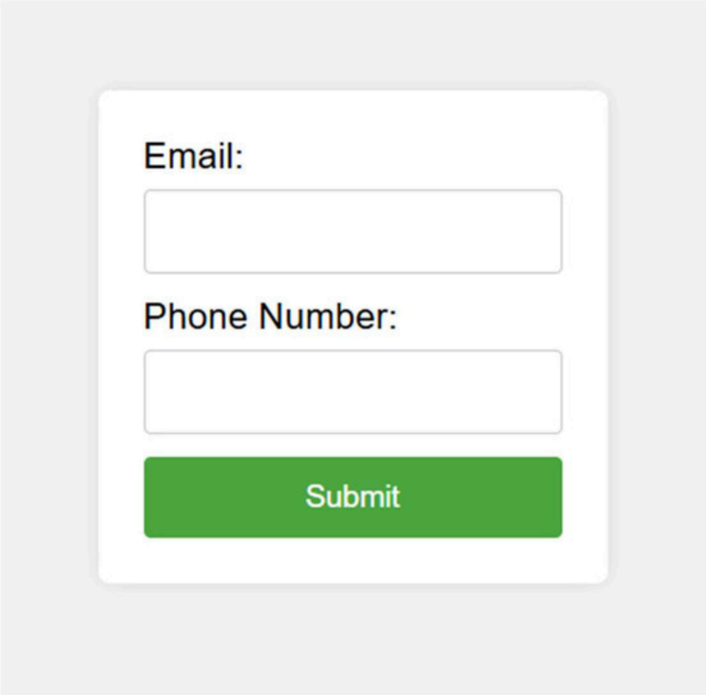
Step 3: Adding JavaScript for Validation

Finally, add JavaScript to validate the input fields using Validator.js and display error messages.

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:



RESULT:
The output was verified successfully.