LAB 9

Sanjana. R CSE C

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

PROCEDURE:

I have created a folder and opened that in VSCode. In that folder, I have created 3 files

- index.html
- style.css
- script.js

1. index.html

This HTML file is for a **basic form validation webpage**.

- It sets up a **form** with two input fields: **Email** and **Phone Number**, and both fields are **required**.
- There are <span> elements next to each input to display **validation error messages**.
- It links to a **CSS file- style.css** for styling
- Uses javascript file (script.js) and a library Validator.js to perform validation and
  - The **Validator.js** library (from CDN) – used for validating inputs like email and phone formats.
  - A **script.js** file – likely contains custom validation logic for the form.
- When the form is submitted, it will use JavaScript to validate the inputs and display errors if needed.

2. style.css

The style.css file is used to **style the form and its components**, making the page visually appealing and user-friendly.

3. script.js

Handles form submission with the following logic:

- **Prevents default form submission** using e.preventDefault().
- **Fetches input values** from the form.
- **Clears any previous error messages**.
- Uses **Validator.js** to check:

  -> If the **email** is valid using validator.isEmail().

->If the **phone number** is valid using validator.isMobilePhone()

- If both inputs are valid:

  -> Prints the user details in terminal.

- If invalid:

  ->Displays appropriate error messages next to the corresponding field.

Index.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>
```

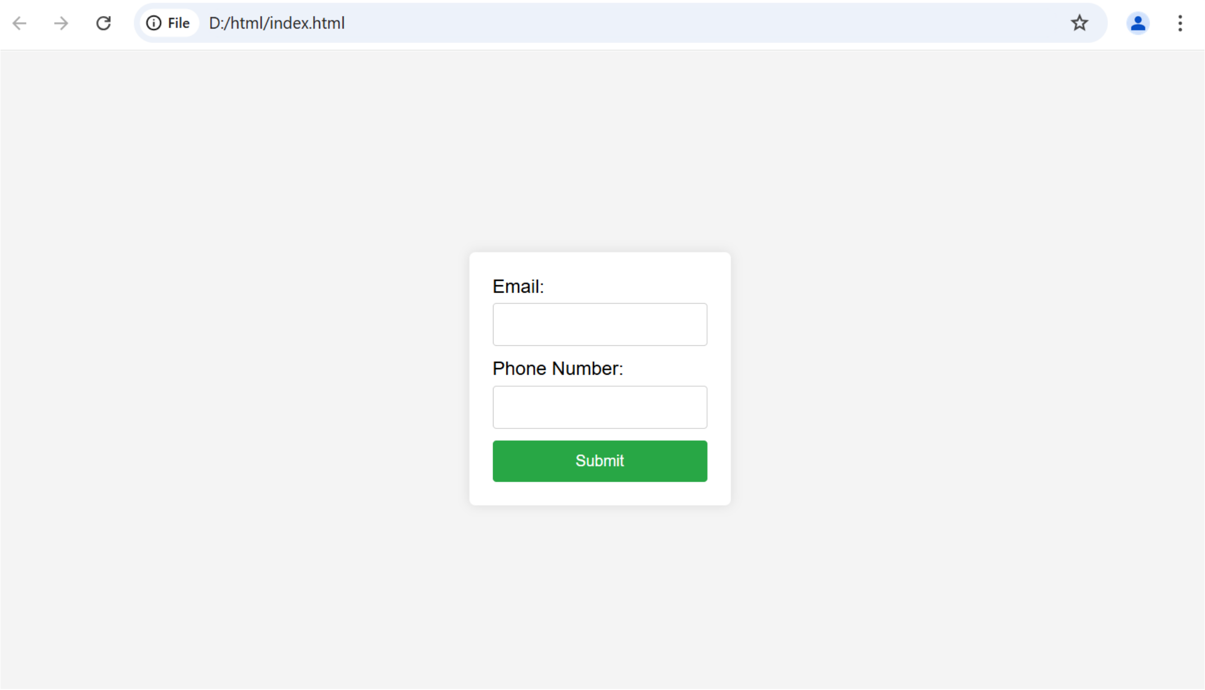
Style.css

```
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;
    }
```

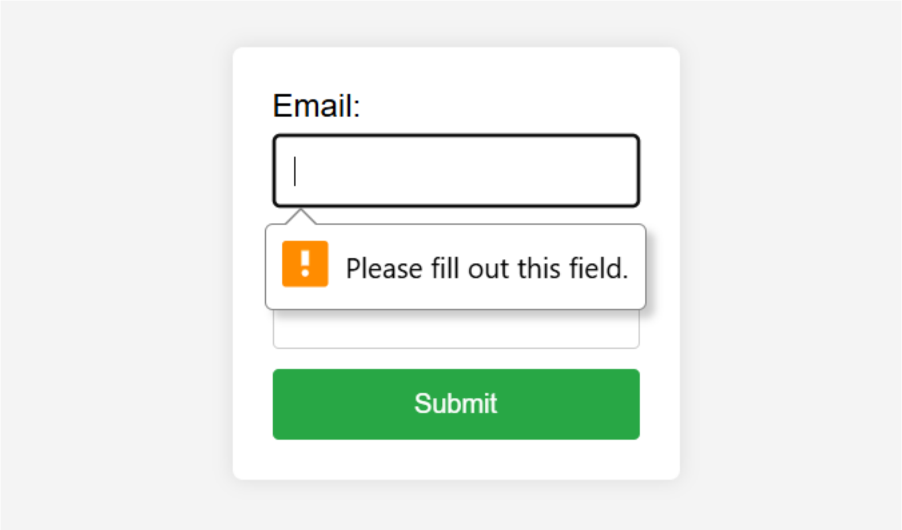
Script.js

```
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);
}
})
```

After running the code, The UI appears as:

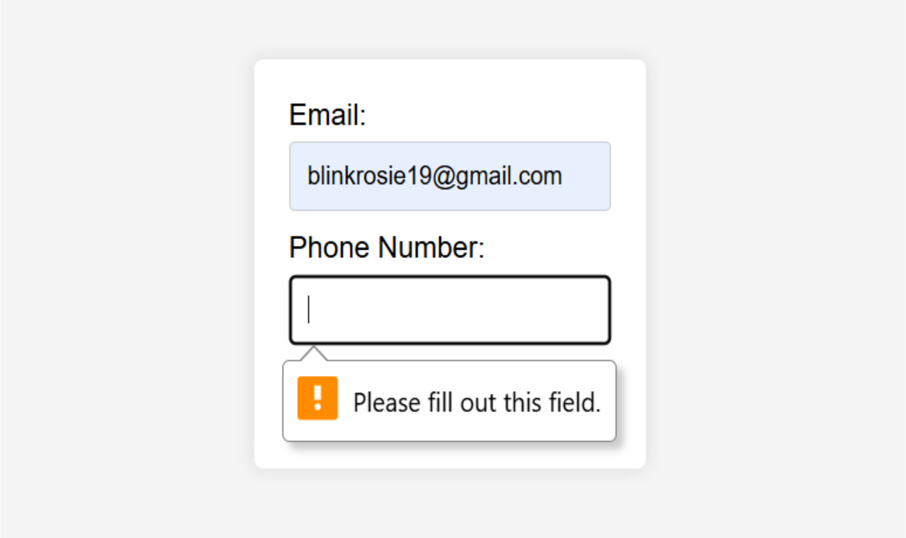


Form validation – The fields cannot be submitted empty and appropriate error messages is displayed.



Email:

Please fill out this field.



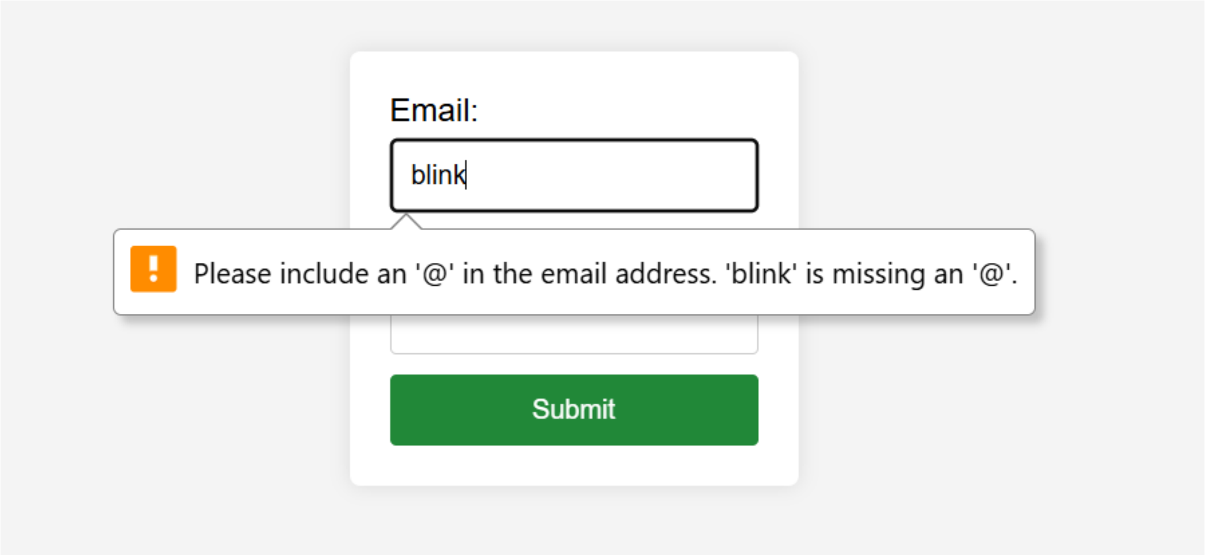
Email:

blinkrosie19@gmail.com

Phone Number:

Please fill out this field.

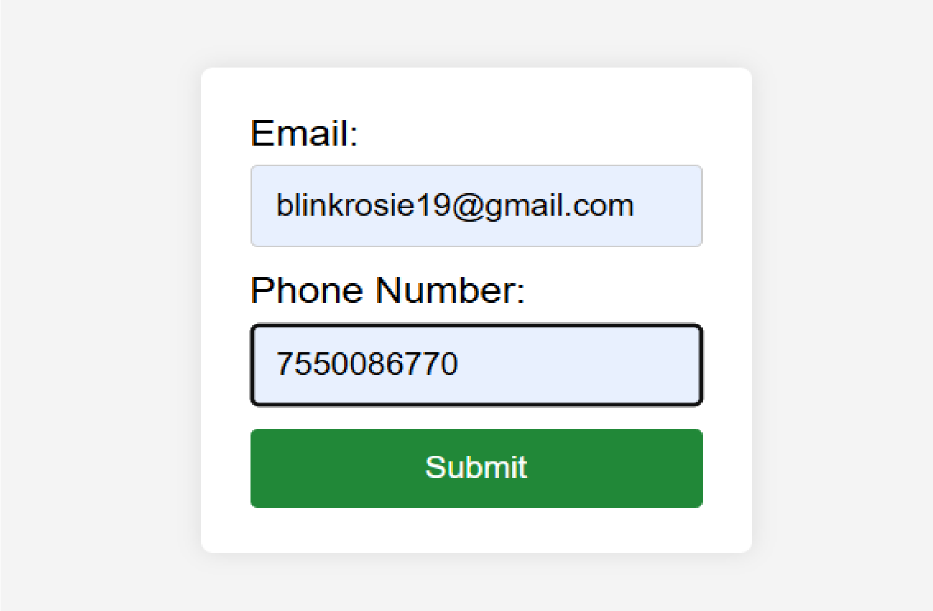
Email Validation – Checks if the user has typed the correct email address and validates it. Displays appropriate error message



blink

Please include an '@' in the email address. 'blink' is missing an '@'.

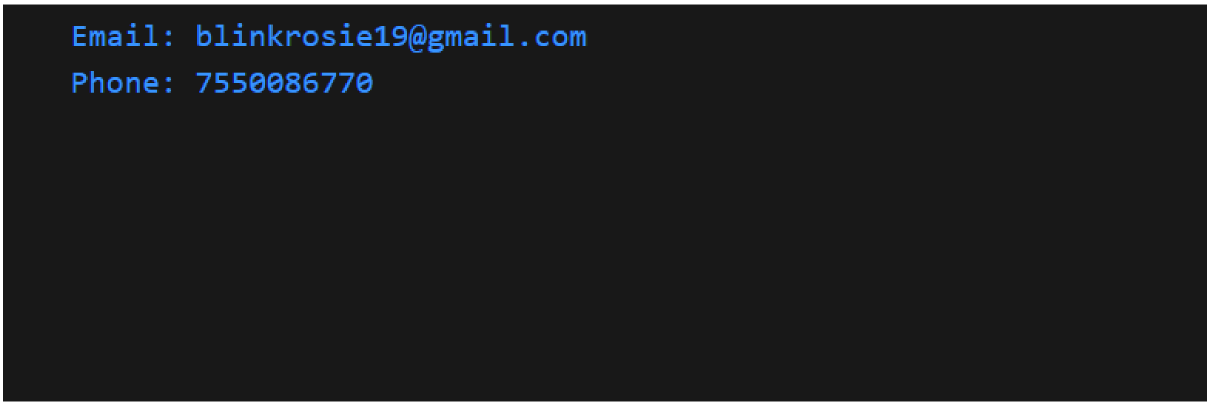
After everything is right and validated, the user can submit the details.



Email: blinkrosie19@gmail.com

Phone Number: 7550086770

As the code is simple, it does not navigate to any other page, instead just prints user details in terminal after validation.



```
Email: blinkrosie19@gmail.com
Phone: 7550086770
```