EXPERIMENT-9

ROLL NO: 230701304 NAME: SHALINI R K

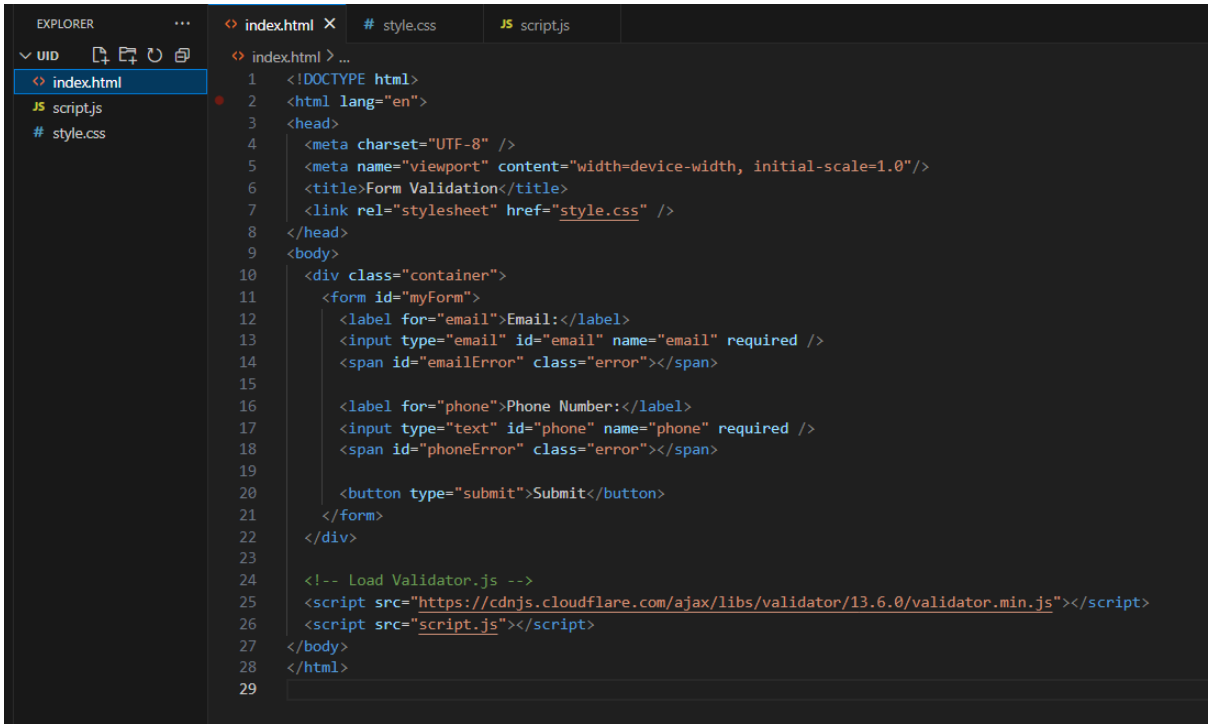
Design input forms that validate data (e.g., email, phone number) and display error messages using HTML/CSS, JavaScript

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.

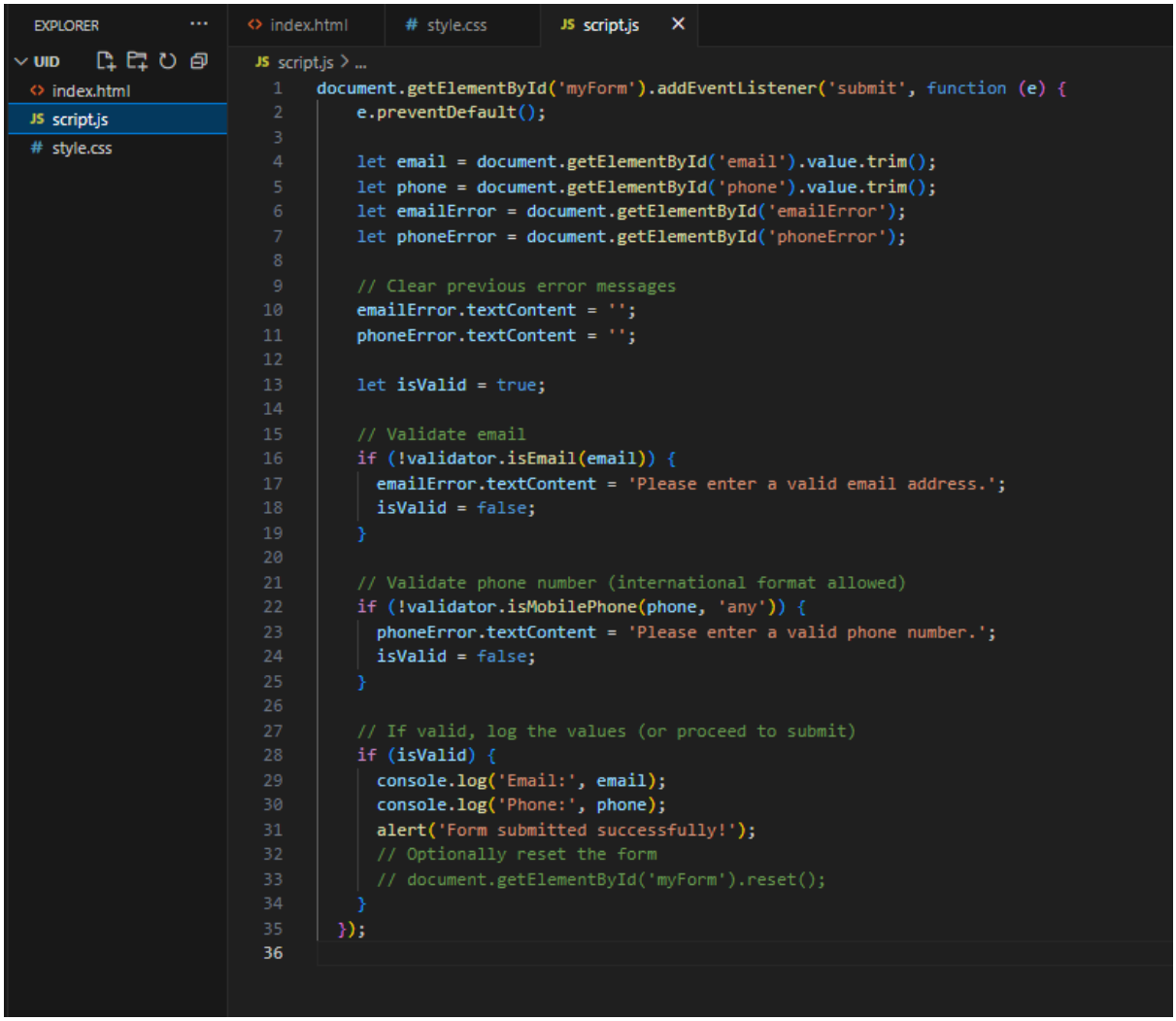
PROCEDURE:

Step 1: Setting Up the HTML Form Start by creating an HTML form with input fields for the email and phone number

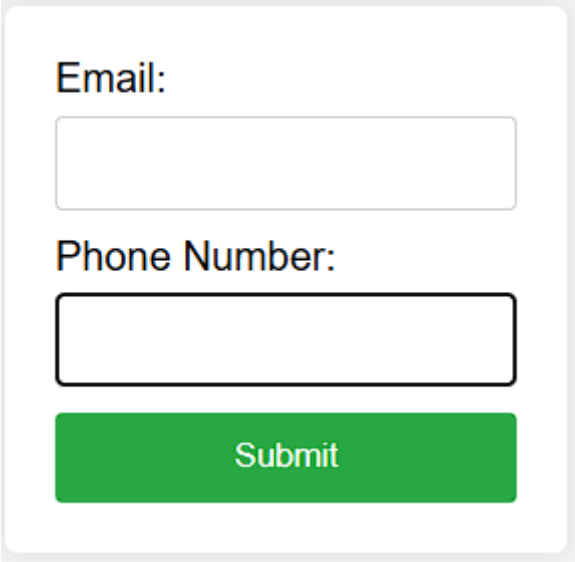


Step 2: Styling the Form with CSS Next, add some basic styling to make the form look nice.

Step 3: Adding JavaScript for Validation Finally, add JavaScript to validate the input fields using Validator.js and display error messages.



OUTPUT1:



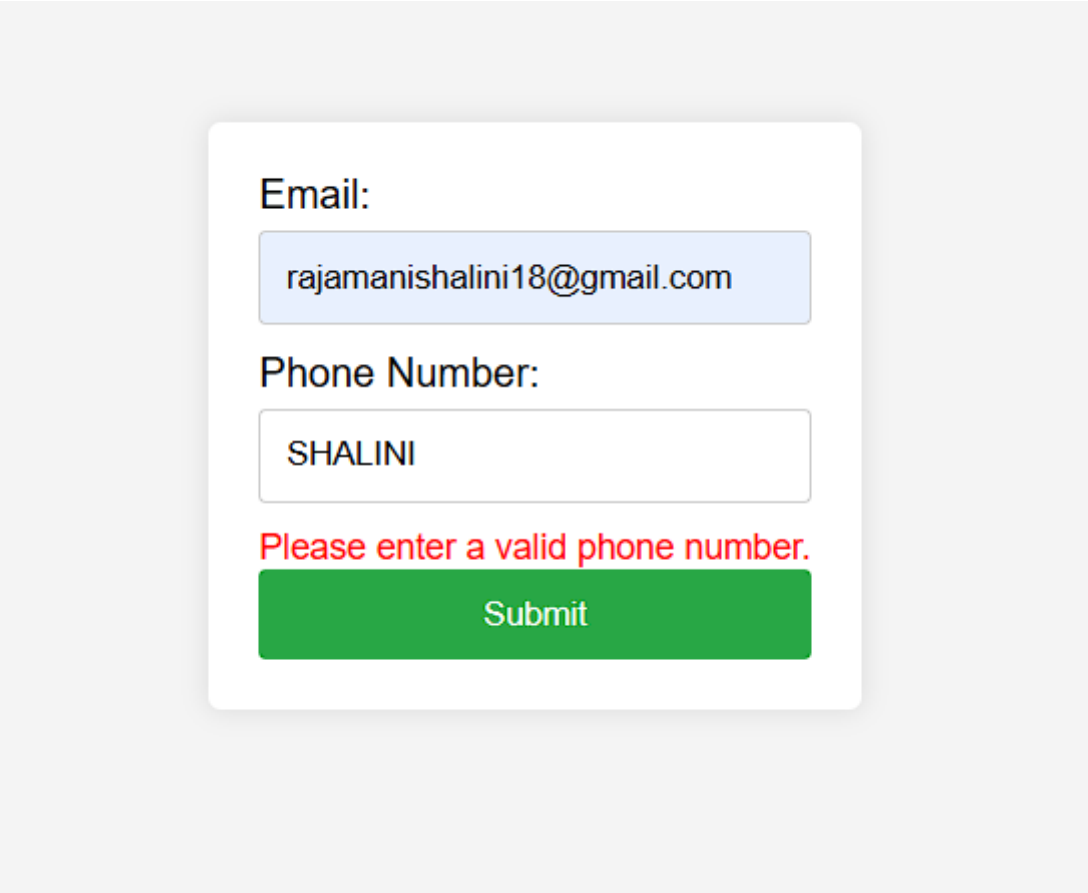
The image you provided shows a **simple form UI** containing:

- A label and input field for **Email**
- A label and input field for **Phone Number**
- A **Submit** button styled with a green background

When the HTML, CSS, and JavaScript code for this UI is executed:

- The **form will be displayed** in a white box with rounded corners and a subtle shadow (likely due to Bootstrap or custom styling).
- Users can type into the email and phone number fields.
- Pressing the green **Submit** button would trigger a JavaScript function, typically to **validate** and/or **send the form data**.

OUTPUT 2:



The image shows a form validation **error message**:

🔴 **Error Message:**

"Please enter a valid phone number." (displayed in red)

💬 **Explanation:**

This message is likely triggered by **JavaScript form validation**. Here's what happened based on the image:

- **Email field** is correctly filled: rajamanishalini18@gmail.com
- **Phone Number field** contains invalid input: "SHALINI" (a name, not a phone number)

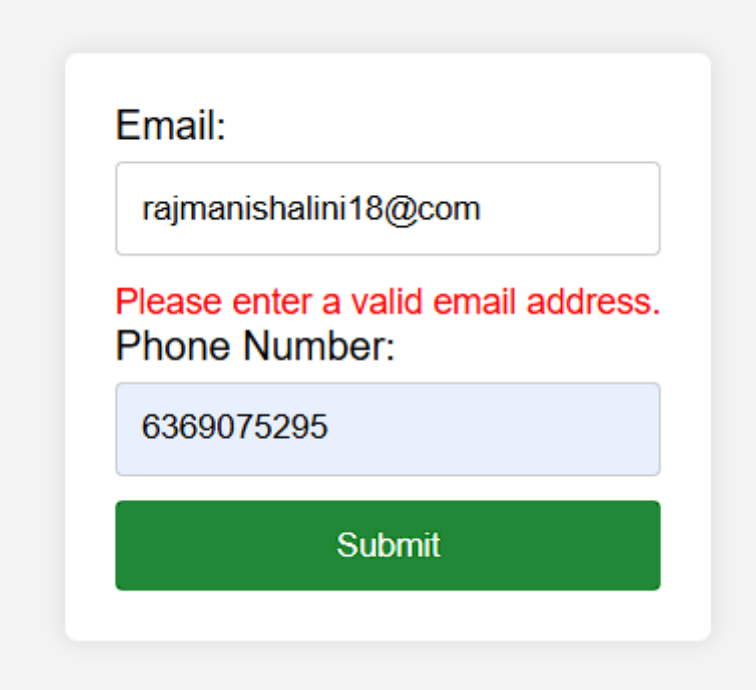
The JavaScript is checking if the input in the **Phone Number** field is a valid format (probably digits only and maybe length constraints, like 10 digits).

✅ **Expected Input Format for Phone Number:**

Likely something like:

- 9876543210 (only digits)
- Matches a regex pattern like: ^[0-9]{10}$

OUTPUT 3:



🔴 **Error Message:**

"Please enter a valid email address." (in red text)

💬 **Explanation:**

- **Email Field Input:** rajamanishalini18@com
  This input is **invalid** because it does not follow the standard format for email addresses.

✅ **Valid Email Format:**

A proper email address should follow this pattern:

pgsql

CopyEdit

username@domain.extension

- Examples of valid formats:
  - rajamanishalini18@gmail.com
  - user@example.in
  - john.doe@outlook.co.uk
- In your case, @com is not a valid domain. A domain like @gmail.com or @yahoo.com would be valid.

✅ **Phone Number Field:**

- **Input:** 6369075295
- This looks valid (10-digit number), so **no error** is shown for the phone number.

OUTPUT 4:

✅ Overview

This section documents the **successful submission** of a user input form that collects an email address and a phone number. The validation logic ensures that both fields meet basic syntactic criteria before allowing submission.

📝 Input Details

- **Email Entered:** rajamanishalini18@gmail.cor
- **Phone Number Entered:** 6369075295

🧪 Validation Checks

📧 Email Validation

- The system checks if the email:
  - Contains the @ symbol.
  - Has characters before and after the @.
  - Ends with a domain extension of **2–3 alphabetic characters** (e.g., .com, .net, .cor).
- In this case, rajamanishalini18@gmail.cor passes the format check, even though .cor is not a standard domain. The validation is based on syntax, not real-world domain existence.

📞 Phone Number Validation

- The system ensures that the phone number:
  - Contains exactly **10 digits**.
  - Consists of only numeric characters (0–9).
- The input 6369075295 satisfies these conditions.

✅ Result

Since both fields pass their respective validations:

- No error messages are displayed.
- The form is submitted successfully.
- A browser alert appears with the message:
  "Form submitted successfully!"