NAME: MAYURI SHRIDATTA YERANDE
ROLL NO: 70
CLASS: D15B
IP EXPERIMENT

EXPERIMENT - 6

AIM: 1. Design user interface for accepting Rollno,Name and marks of subjects. WAP to determine grades depending on the marks scored by students. (make use of different events)
2. WAP to implement a calculator. (make use of different DOM methods for accessing HTML elements)
3. **Experiment to implement the Registration form and apply the following data validation:**
-Phno((country code)10 digit phone number)
-email validation
- should consist of @ and .
- domain name should be min 2 characters and max 3 characters.
- username can consist of alphanumeric characters , _ and .
- Password Validation(password should consist of at least 1 uppercase character, at least 1 digit and at least 1 special character.)

THEORY:

HTML DOM methods are actions you can perform (on HTML Elements). HTML DOM properties are values (of HTML Elements) that you can set or change.

The DOM Programming Interface

The HTML DOM can be accessed with JavaScript (and with other programming languages). In the DOM, all HTML elements are defined as objects. The programming interface is the properties and methods of each object. A property is a value that you can get or set (like changing the content of an HTML element). A method is an action you can do (like add or deleting an HTML element).

Property / Method	Description
accessKey	Sets or returns the accesskey attribute of an element
addEventListener()	Attaches an event handler to an element
appendChild()	Adds (appends) a new child node to an element
attributes	Returns a NamedNodeMap of an element's attributes
blur()	Removes focus from an element
childElementCount	Returns an elements's number of child elements
childNodes	Returns a NodeList of an element's child nodes
children	Returns an HTMLCollection of an element's child elements
classList	Returns the class name(s) of an element
className	Sets or returns the value of the class attribute of an element

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

- An HTML web page has finished loading
- An HTML input field was changed
- An HTML button was clicked

Often, when events happen, you may want to do something.

JavaScript lets you execute code when events are detected.

HTML allows event handler attributes, with JavaScript code, to be added to HTML elements.

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

Regular Expression: A Regular Expression is an object that describes a pattern of characters. Regular Expressions are used to perform pattern-matching and search-and-replace functions on text. So, in this area JavaScript plays a major role in validating the values.

Form validation (validating an email)

Assume a registration form that contains the basic details of the end-users like Name, Phone number, Email id, and Address. When the user enters the email id without the domain name and "@" symbol the form turns out an error that says "domain name not included". Ever wonder how this happens? This happens due to the Regular Expressions in JavaScript. Regular Expression can be defined as a stopper(pattern -match) to the values which are not correct i.e., "indicating an error while the end-user is entering the wrong details instead of the given regular expression ". Some of the characters used are "[abc],[^abc],\w,\W,\S". Thus validating the email address entered by the end-user is done by JavaScript.

Form Validation (validating phone number)

Suppose assume the same registration form. Ever wonder why the number should be started with only from 6, 7, 8, 9 and not the remaining numbers. Here, also the picture is played by Regular Expression which helps in validating one's correct mobile number. Restricting the users to enter only 10 digits where the first digit should of numbers "6,7,8,9" and rest all digits can be of any number from 0-9 is done purely by regular expressions "[^6-9][,0-9]" which help in validating the forms whether the information entered is relevant to the pattern specified.

IMPLEMENTATION:

1. Design user interface for accepting Rollno,Name and marks of subjects. WAP to determine grades depending on the marks scored by students. (make use of different events)

Program:

```
<!DOCTYPE html>
<html>
<head>
        <title>student grade calculater</title>

</head>
<body>
<style>
* {
margin: 0;
padding: 0;
box-sizing: border-box;
}
body {
background: #FFB6C1          ;
font-size: 12px;
}

.container {
flex: 0 1 700px;
margin: auto;
padding: 10px;
}

.screen-body-item {
flex: 1;
padding: 50px;
}
input {
margin: 10px 10px 10px;
}
.showdata {
color: black;
```

```
font-size: 1.2rem;
padding-top: 10px;
padding-bottom: 10px;
}
</style>
        <!-- main html -->
        <div class="container">
        <h1>Student Grade Calculator</h1>
        <div class="screen-body-item">
                <div class="app">
<div class="form-group">
                        <input
                        type="text"
                        class="form-control"
                        placeholder=" Enter Your Roll Number"
                        id="roll"
                        />
                </div>

<div class="form-group">
                        <input
                        type="text"
                        class="form-control"
                        placeholder=" Enter your name"
                        id="name"
                        />
                </div>

                <div class="form-group">
                        <input
                        type="text"
                        class="form-control"
                        placeholder=" Enter Chemistry Marks "
                        id="chemistry"
                        />
                </div>
                <div class="form-group">
                        <input
                        type="text"
                        class="form-control"
```

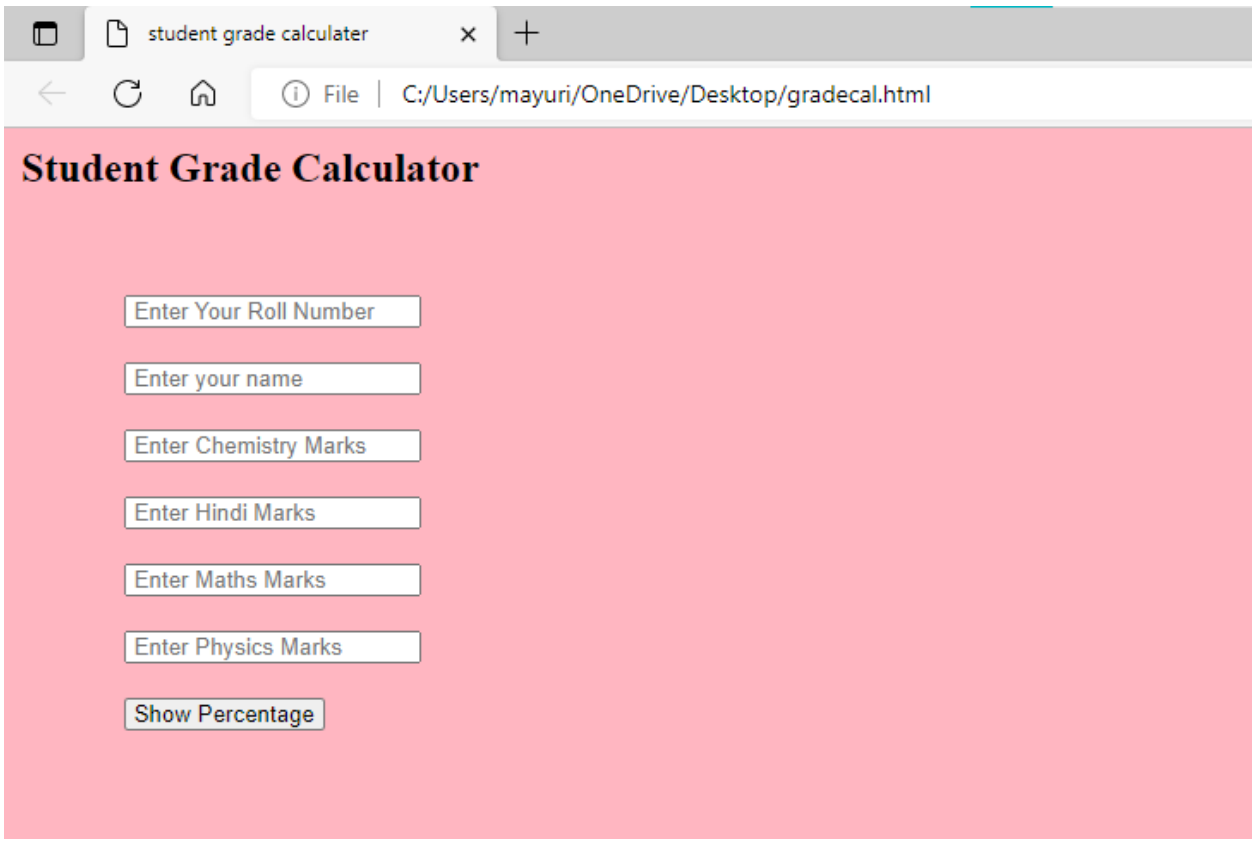
```
                placeholder=" Enter Hindi Marks "
                id="hindi"
                />
            </div>
            <div class="form-group">
                <input
                type="text"
                class="form-control"
                placeholder=" Enter Maths Marks "
                id="maths"
                />
            </div>
            <div class="form-group">
                <input
                type="text"
                class="form-control"
                placeholder=" Enter Physics Marks "
                id="phy"
                />
            </div>
            <div>
                <input
                type="button"
                value=" Show Percentage "
                class="form-button"
                onclick="calculate()"
                />
            </div>
            </div>
        </div>
        <!-- for showing the result-->
        <div class="form-group showdata">
            <p id="showdata"></p>
        </div>
        </div>
        <script>
// Function for calculating grades
const calculate = () => {

let roll = document.querySelector("#roll").value;
```

```
let name = document.querySelector("#name").value;
let chemistry = document.querySelector("#chemistry").value;
let hindi = document.querySelector("#hindi").value;
let maths = document.querySelector("#maths").value;
let phy = document.querySelector("#phy").value;
let grades = "";

// Input is string so typecasting is necessary. */
let totalgrades =
        parseFloat(chemistry) +
        parseFloat(hindi) +
        parseFloat(maths) +
        parseFloat(phy);

// Checking the condition for the providing the
// grade to student based on percentage
let percentage = (totalgrades / 400) * 100;
if (percentage <= 100 && percentage >= 80) {
        grades = "A";
} else if (percentage <= 79 && percentage >= 60) {
        grades = "B";
} else if (percentage <= 59 && percentage >= 40) {
        grades = "C";
} else {
        grades = "F";
}
// Checking the values are empty if empty than
// show please fill them
if (chemistry == "" || hindi == ""
                        || maths == "" || phy == "") {
        document.querySelector("#showdata").innerHTML
                = "Please enter all the fields";
} else {

        // Checking the condition for the fail and pass
        if (percentage >= 39.5) {
        document.querySelector(
                "#showdata"
        ).innerHTML =
                ` Out of 400 your total is ${totalgrades}
```

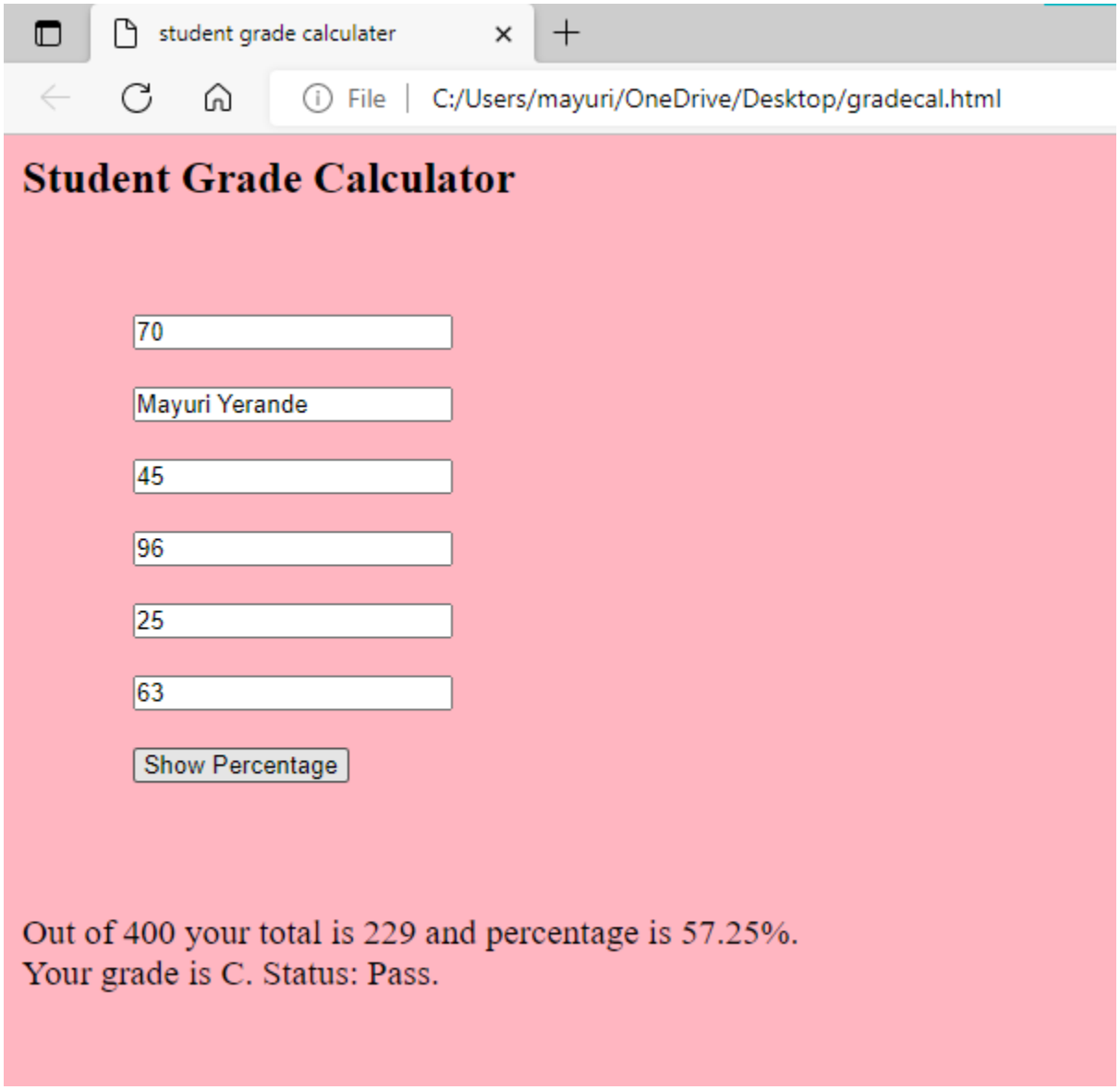
```
                and percentage is ${percentage}%. <br>
                Your grade is ${grades}. Status: Pass. `;
        } else {
        document.querySelector(
                "#showdata"
        ).innerHTML =
                ` Out of 400 your total is ${totalgrades}
                and percentage is ${percentage}%. <br>
                Your grade is ${grades}. Status: Fail. `;
        }
}
};

</script>
</body>
</html>
```

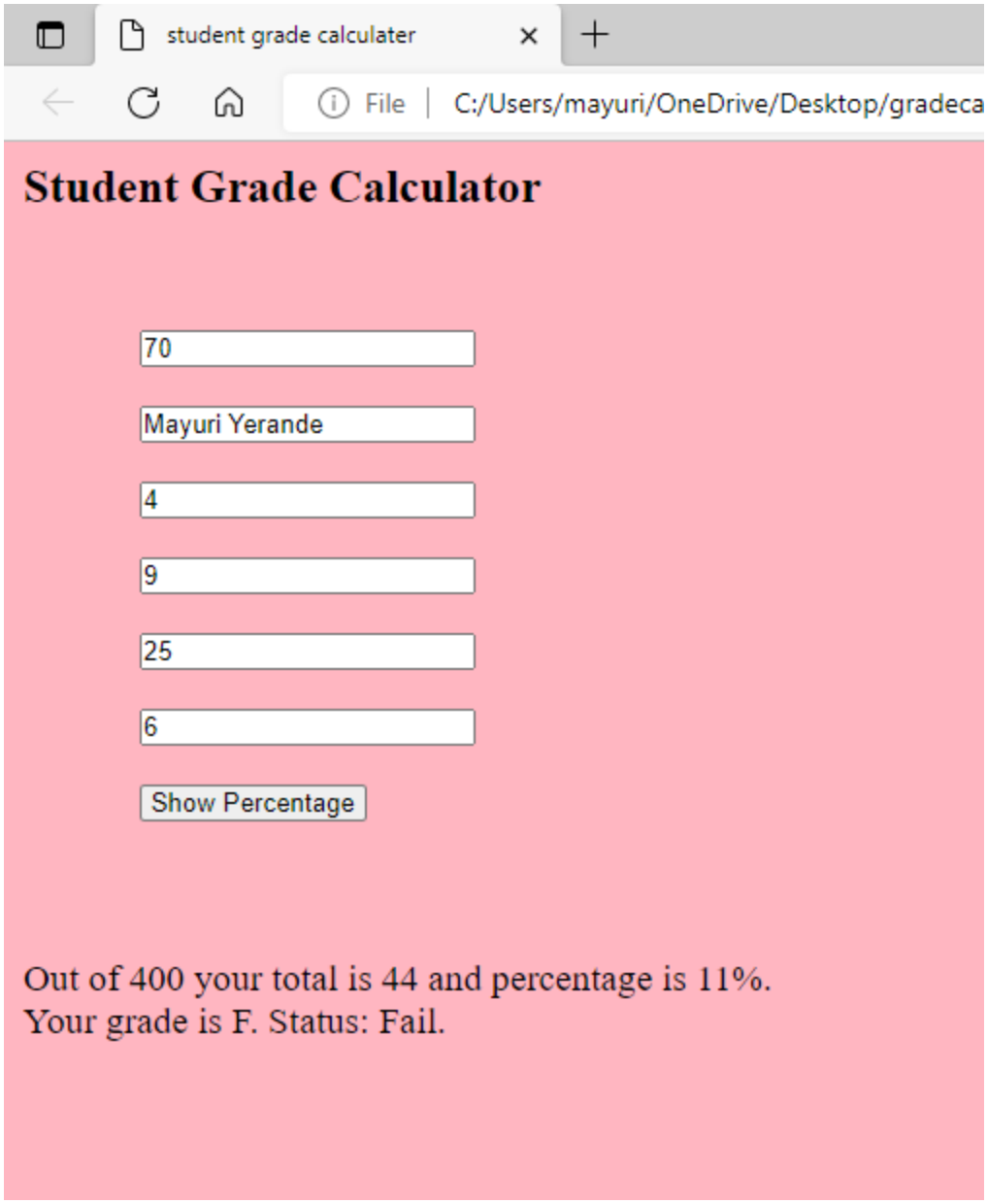
Output:

Without Input:-



With Input:-


student grade calculater
File | C:/Users/mayuri/OneDrive/Desktop/gradecal.html
Student Grade Calculator
70
Mayuri Yerande
45
96
25
63
Show Percentage
Out of 400 your total is 229 and percentage is 57.25%.
Your grade is C. Status: Pass.



2. WAP to implement a calculator. (make use of different DOM methods for accessing HTML elements)

Program:

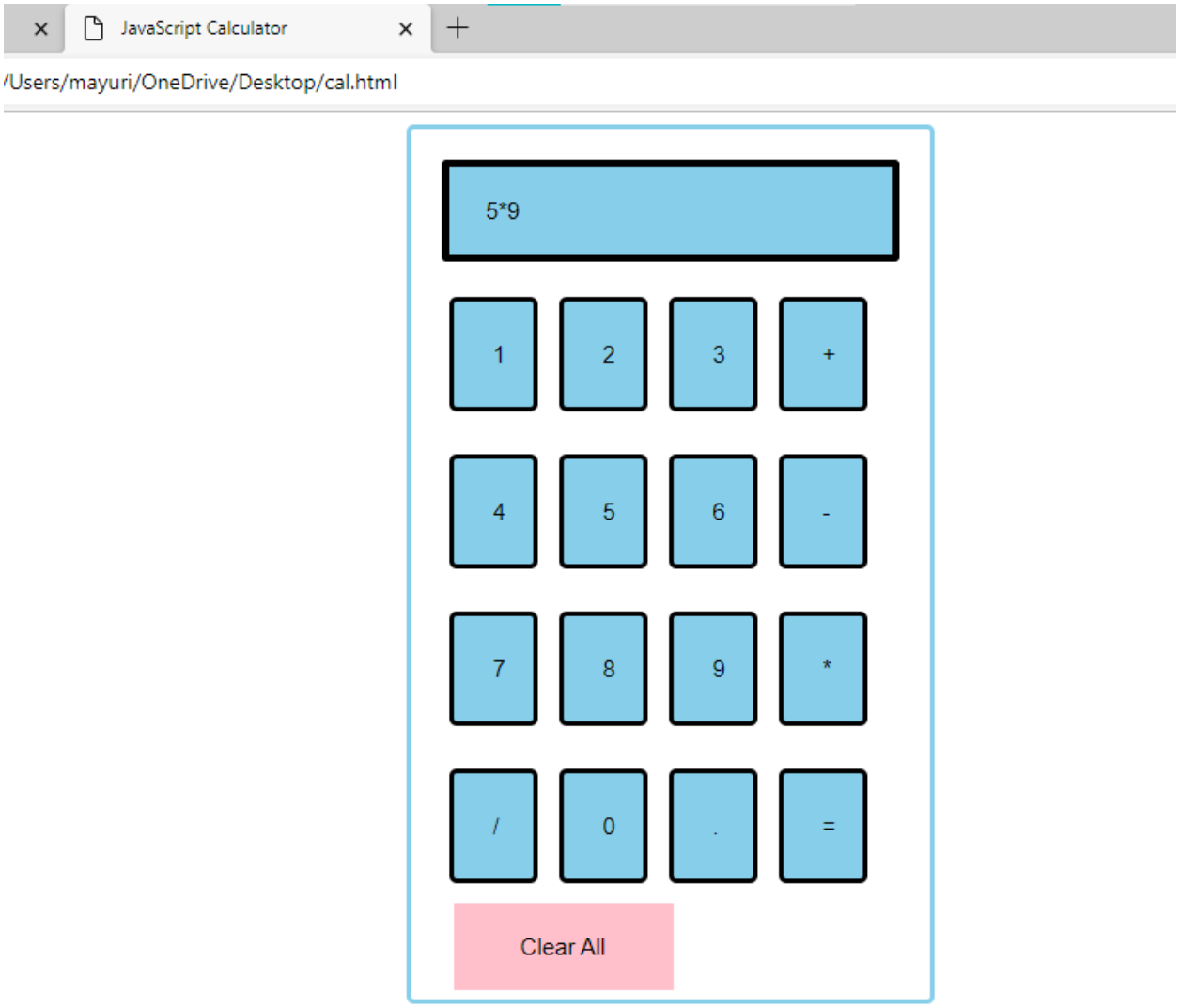
```
<!DOCTYPE html>
<html lang = "en">
<head>
<title> JavaScript Calculator </title>
```

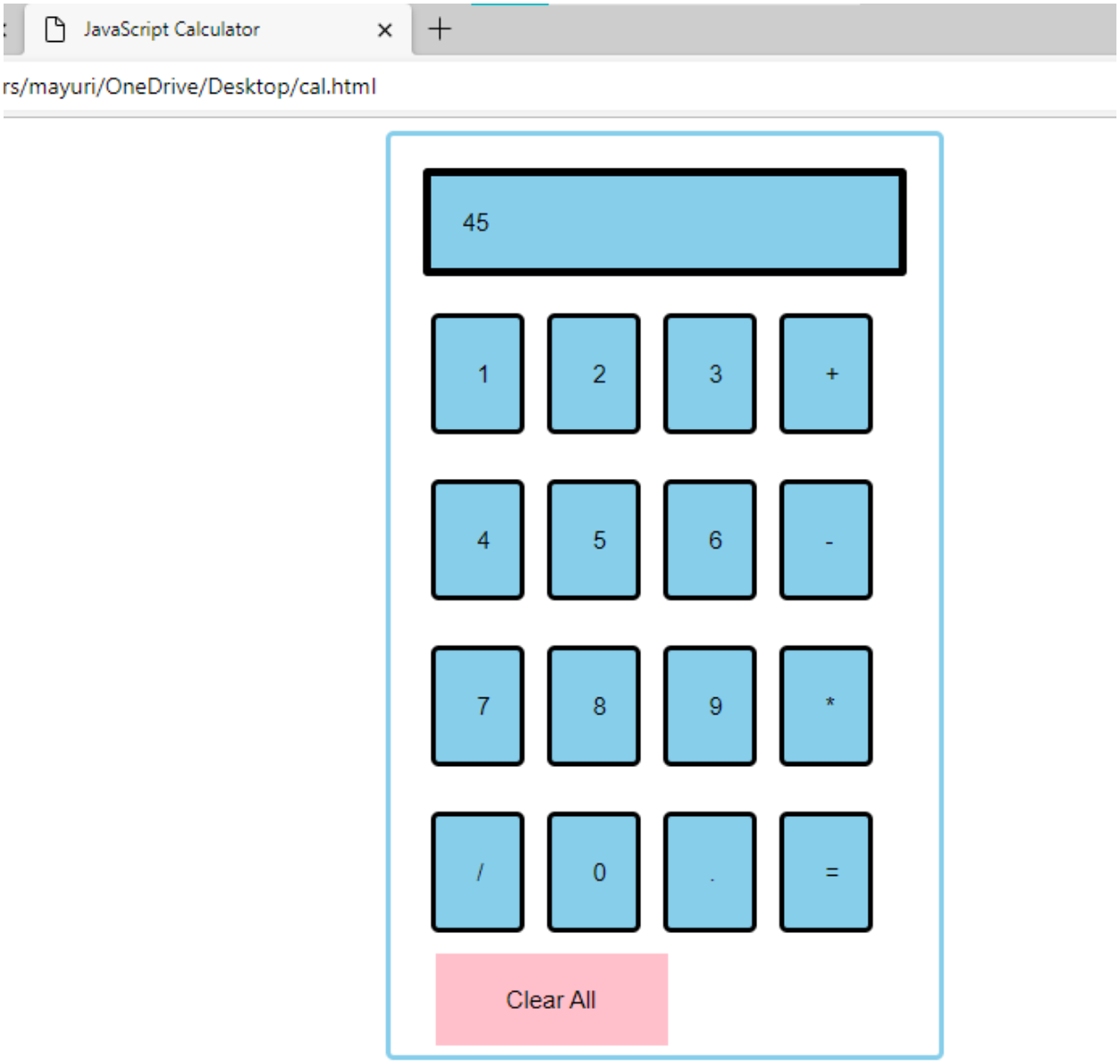
```
<style>
h1 {
    text-align: center;
    padding: 23px;
    background-color: skyblue;
    color: white;
    }

#clear{
width: 150px;
border: 3px solid white;
    border-radius: 3px;
    padding: 20px;
    background-color: pink;
}

.formstyle
{
width: 300px;
height: 530px;
margin: auto;
border: 3px solid skyblue;
border-radius: 5px;
padding: 20px;
}



input
{
width: 20px;
background-color: skyblue;
color: black;
border: 3px solid black;
    border-radius: 5px;
    padding: 26px;
    margin: 5px;
    font-size: 15px;
}
```

```
#calc{
width: 250px;
border: 5px solid black;
    border-radius: 3px;
    padding: 20px;
    margin: auto;
}

</style>

</head>
<body>
<div class= "formstyle">
<form name = "form1">

    <input id = "calc" type ="text" name = "answer"> <br> <br>
  <input type = "button" value = "1" onclick = "form1.answer.value += '1' ">
  <input type = "button" value = "2" onclick = "form1.answer.value += '2' ">
  <input type = "button" value = "3" onclick = "form1.answer.value += '3' ">
   <input type = "button" value = "+" onclick = "form1.answer.value += '+' ">
  <br> <br>

  <input type = "button" value = "4" onclick = "form1.answer.value += '4' ">
  <input type = "button" value = "5" onclick = "form1.answer.value += '5' ">
  <input type = "button" value = "6" onclick = "form1.answer.value += '6' ">
  <input type = "button" value = "-" onclick = "form1.answer.value += '-' ">
  <br> <br>

  <input type = "button" value = "7" onclick = "form1.answer.value += '7' ">
  <input type = "button" value = "8" onclick = "form1.answer.value += '8' ">
  <input type = "button" value = "9" onclick = "form1.answer.value += '9' ">
  <input type = "button" value = "*" onclick = "form1.answer.value += '*' ">
  <br> <br>


  <input type = "button" value = "/" onclick = "form1.answer.value += '/' ">
  <input type = "button" value = "0" onclick = "form1.answer.value += '0' ">
    <input type = "button" value = "." onclick = "form1.answer.value += '.' ">
```

```
   <!-- When we click on the '=' button, the onclick() shows the sum results on the calculator
screen. -->
  <input type = "button" value = "=" onclick = "form1.answer.value = eval(form1.answer.value)
">
  <br>
  <!-- Display the Cancel button and erase all data entered by the user. -->
  <input type = "button" value = "Clear All" onclick = "form1.answer.value = ' ' " id= "clear" >
  <br>

</form>
</div>
</body>
</html>
```

Output:


JavaScript Calculator
/Users/mayuri/OneDrive/Desktop/cal.html
5*9
1
2
3
+
4
5
6
-
7
8
9
*
/
0
.
=
Clear All



3. **Experiment to implement the Registration form and apply the following data validation:**

-Phno((country code)10 digit phone number)
-email validation
- should consist of @ and .
- domain name should be min 2 characters and max 3 characters.
- username can consist of alphanumeric characters , _ and .
- Password Validation(password should consist of at least 1 uppercase character, at least 1 digit and at least 1 special character.)

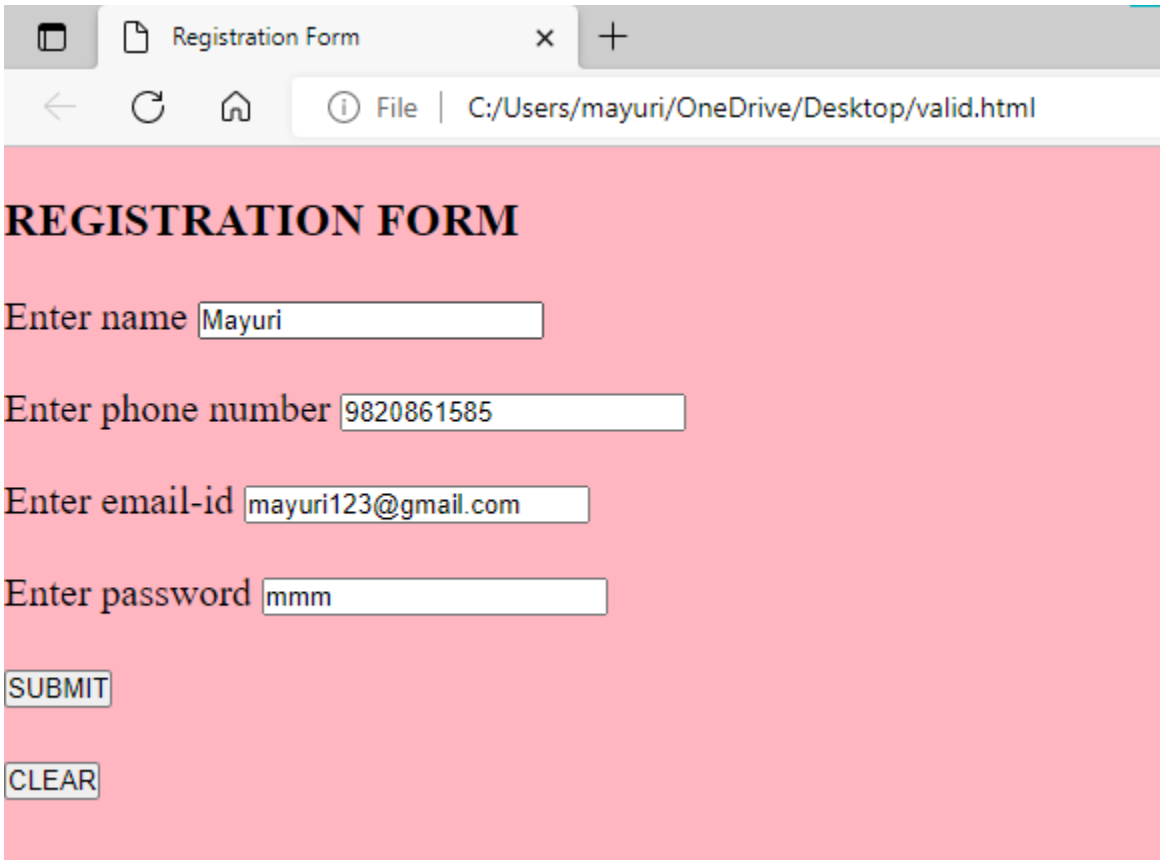
Program:

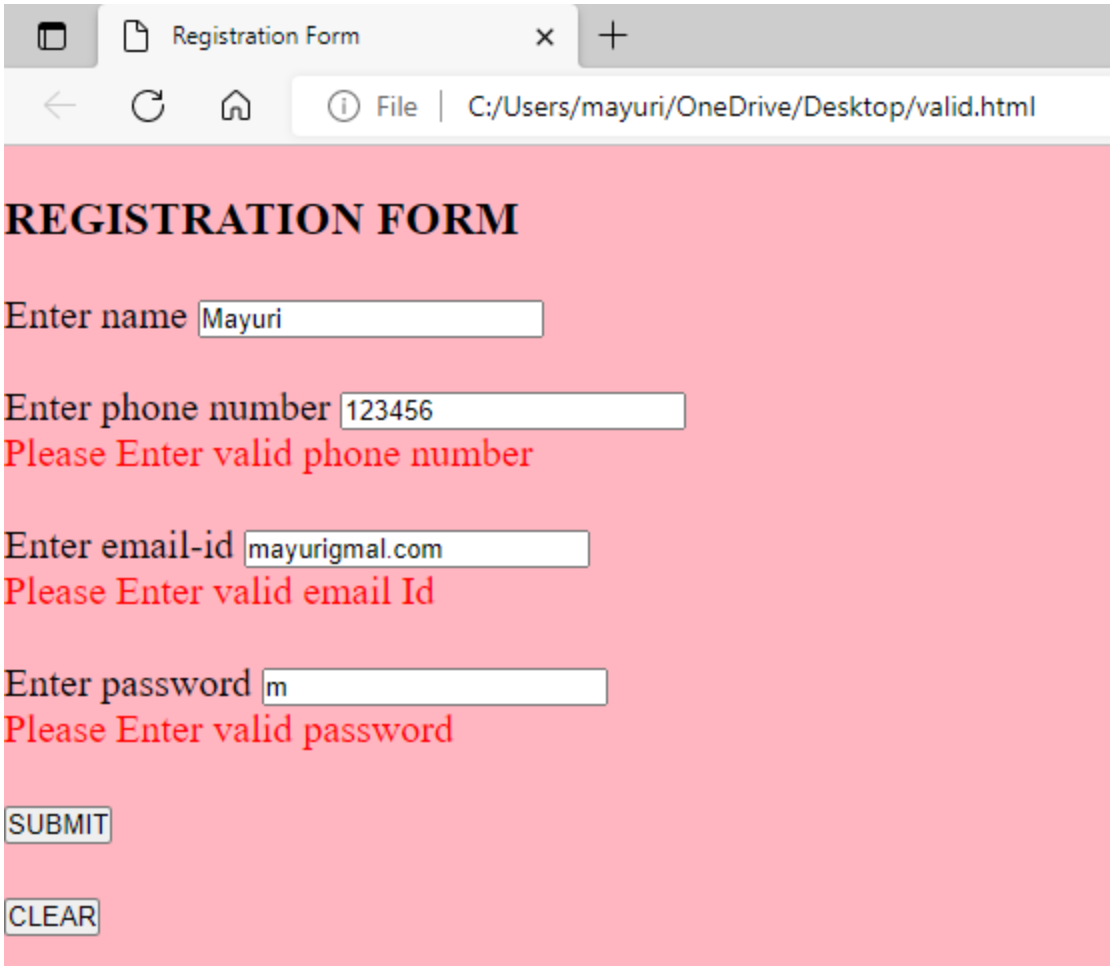
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Registration Form</title>
    <br>
  </head>
  <style>
    * {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    }
    body {
    background: #FFB6C1       ;
    font-size: 20px;
    }

    </style>

  <body>
    <h3>REGISTRATION FORM</h3>
    <br>
    <label for="box1">Enter name</label>
    <input type="text" id="box1" /><br />
    <br>
    <label for="box2">Enter phone number </label>
    <input type="text" id="box2" /><br />
    <p id="error1"></p>
    <br>
    <label for="box3">Enter email-id</label>
    <input type="text" id="box3" /><br />
    <p id="error2"></p>
    <br>
    <label for="box4">Enter password</label>
    <input type="text" id="box4" /><br />
```

```
<p id="error3"></p>
<br>
<input type="button" value="SUBMIT" onclick="CheckPhoneNo() " /><br />
<br>
<input type="button" value="CLEAR" onclick="clearVal()" /><br />

<script>
  function CheckPassword() {
    var password = document.getElementById("box4").value;
    var pw_validate = /^(?=.*\d)(?=.*[a-z])(?=.*[!@#$%^&*])(?=.*[A-Z])/;
    var err_mssg = "Please Enter valid password";
    //atleast 1 number,1 special character,1 uppercase
    if (!pw_validate.test(password)) {
      document.getElementById("error3").innerHTML = err_mssg;
      document.getElementById("error3").style.color = "#ff0000";
    }
  }
  function CheckEmailId() {
    var emailId = document.getElementById("box3").value;
    var ei_validate = /^[a-z0-9._-]+@[a-z]+\.[a-z]{2,3}$/;
    var err_mssg = "Please Enter valid email Id";
    if (!ei_validate.test(emailId)) {
      document.getElementById("error2").innerHTML = err_mssg;
      document.getElementById("error2").style.color = "#ff0000";
    }
    CheckPassword();
  }
  function CheckPhoneNo() {
    var pho = document.getElementById("box2").value;
    var ph_validate = /^(\(?\+?\d{2}\)?)?\d{10}$/;
    var err_mssg = "Please Enter valid phone number";
    if (!ph_validate.test(pho)) {
      document.getElementById("error1").innerHTML = err_mssg;
      document.getElementById("error1").style.color = "#ff0000";
    }
    CheckEmailId();
  }
  function clearVal() {
    document.getElementById("box1").value = "  ";
    document.getElementById("box2").value = "  ";
```

```
        document.getElementById("box3").value = "  ";
        document.getElementById("box4").value = "  ";
        document.getElementById("error1").innerHTML = " ";
        document.getElementById("error2").innerHTML = " ";
        document.getElementById("error3").innerHTML = " ";
      }
    </script>
  </body>
</html>
```

Output:





CONCLUSION: We have studied the advanced concepts of Java Script and successfully implemented programs on JavaScript. We even learnt about validation in javascript.