K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)

Batch: E-2 **Roll No.: 16010123325**

Experiment No. 04

TITLE: Develop and demonstrate JavaScript with POP-UP boxes and functions

AIM: To demonstrate the functionalities of JavaScript using HTML and CSS

Expected Outcome of Experiment: Design static web pages using various HTML tags.

Books/ Journals/ Websites referred:

1. https://www.w3schools.com/js/

Describe and utilize Javascript programming concepts such as variables, arrays, conditionals, and loops.
Write and deploy Javascript code to solve practical web design problems.

Problem Statement: Description of the application implemented with output:

a) Input: Click on Display Date button using onclick( ) function
Output: Display date in the textbox
b) Input: A number n obtained using prompt
Output: Factorial of n number using alert
c) Input: A number n obtained using prompt
Output: A multiplication table of numbers from 1 to 10 of n using
d**) Write JavaScript to validate the following fields for the registration page**.
Name (Name should contain alphabets and the length should not be less than 6 characters).
Password (Password should not be less than 6 characters length).
E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
Phone number (Phone number should contain 10 digits only).

JavaScript Basic Concepts Learned With Syntax

- **Event Handling:** You used onclick and onsubmit to trigger specific functions when an event occurs.

  Syntax: <button onclick="displayDate ()">Display Date</button>

- **DOM Manipulation:** You dynamically updated or accessed elements using their IDs

  Syntax**:** document.getElementById('dateBox').value = today.toLocaleString();

- **User Input via Prompt:** You used prompt() to gather user input and perform calculations.
  Syntax: const n = parseInt(prompt("Enter a number to find its factorial:", ""));

- **Alerts for Feedback :** Alerts were used to inform users of results or errors.
  Syntax: alert(`The factorial of ${n} is ${factorial}.`);

- **Loops:**

  Syntax:
  ```
  for (let i = 1; i <= n; i++) {
      factorial *= i;
  }
  ```

- **Form Validation:**

  Syntax :
  ```
  if (!/^[a-zA-Z]{6,}$/.test(name)) {
          alert("Name should contain only alphabets");
  }
  if (!emailPattern.test(email)) {
          alert("Invalid email.");
  }
  ```

- **Function Design:** Separating code logic into clear, reusable functions

  displayDate(), calculateFactorial(), displayMultiplicationTable(), and validateForm().

Description of the application implemented with output:

1. **Display Date**

   Input: Click "Display Date" button.

   Output: Shows the current date in the textbox using the displayDate() function.

2. **Factorial Calculation**

   Input: Enter a number via prompt().

   Output: Displays factorial of the number using alert().

3. **Multiplication Table**

   Input: Enter a number via prompt().

   Output: Shows the multiplication table (1 to 10) via alert().

4. **Registration Form Validation**

   Name: Must be alphabetic, at least 6 characters.

   Password: Minimum length of 6 characters.

   Email: Valid format (e.g., name@domain.com).

   Phone: Must be 10 digits long.

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Combined Exercises</title>
</head>
<body>
    <header>
        <h1>Shreyans Tatiya</h1>
        <p>Roll Number: 16010123325</p>
```

```
    </header>

    <section id="registrationSection">
        <h2>Registration Form</h2>
        <form id="registrationForm">
            <label for="nameInput">Name:</label>
            <input type="text" id="nameInput" required>

            <label for="passwordInput">Password:</label>
            <input type="password" id="passwordInput" minlength="6"
required>

            <label for="emailInput">Email:</label>
            <input type="email" id="emailInput" required>

            <label for="phoneInput">Phone:</label>
            <input type="tel" id="phoneInput" pattern="[0-9]{10}"
required>

            <button type="submit">Register</button>
        </form>
    </section>

    <section id="multiplicationSection">
        <h2>Multiplication Table</h2>
        <div>
            <label for="numberInput">Enter a number:</label>
            <input type="number" id="numberInput">
            <button onclick="generateTable()">Generate Table</button>
            <div id="multiplicationTable"></div>
        </div>
    </section>

    <section id="dateSection">
        <h2>Display Date</h2>
        <div>
            <input type="text" id="dateInput">
            <button onclick="displayDate()">Display Date</button>
        </div>
    </section>

    <section id="factorialSection">
        <h2>Factorial Calculator</h2>
        <div>
            <button onclick="calculateFactorial()">Calculate
Factorial</button>
            <p id="factorialResult"></p>
        </div>
    </section>
```

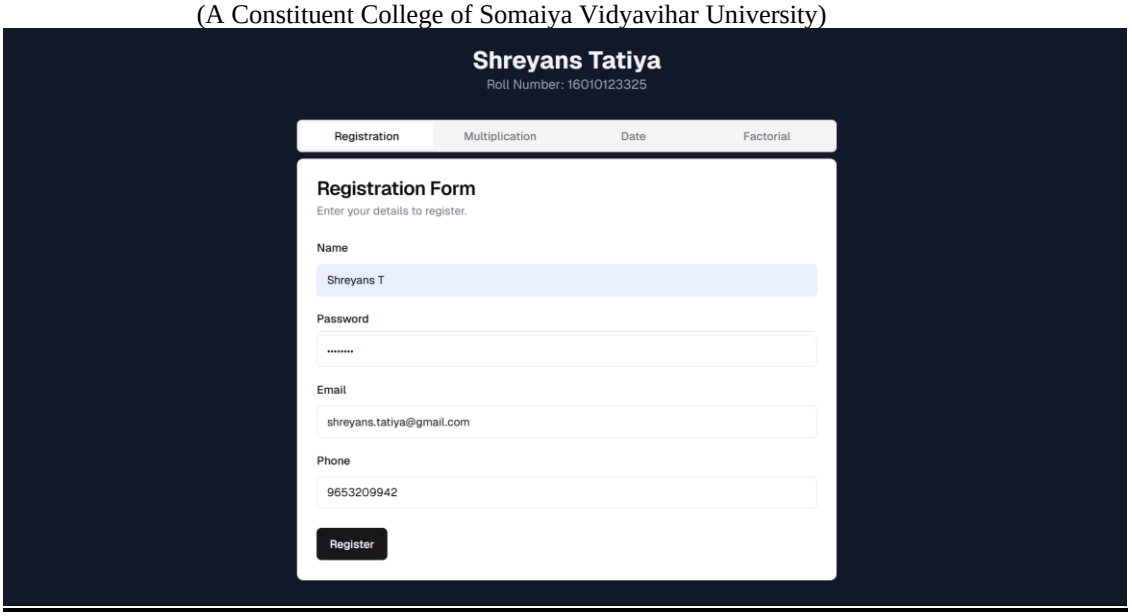
K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)

```
    <script>
        // Registration Form Validation
        const nameInput = document.getElementById("nameInput");
        const passwordInput = document.getElementById("passwordInput");
        const emailInput = document.getElementById("emailInput");
        const phoneInput = document.getElementById("phoneInput");
        const registrationForm =
document.getElementById("registrationForm");

        function validateName() {
            const name = nameInput.value.trim();
            if (/^[a-zA-Z]{6,}$/.test(name)) {
                nameInput.setCustomValidity("");
            } else {
                nameInput.setCustomValidity(
                    "Name should contain alphabets and the length should
not be less than 6 characters.");
            }
        }

        function validatePassword() {
            const password = passwordInput.value;
            if (password.length >= 6) {
                passwordInput.setCustomValidity("");
            } else {
                passwordInput.setCustomValidity("Password should not be
less than 6 characters length.");
            }
        }

        function validateEmail() {
            const email = emailInput.value.trim();
            if (/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {
                emailInput.setCustomValidity("");
            } else {
                emailInput.setCustomValidity("Invalid email format.");
            }
        }

        function validatePhone() {
            const phone = phoneInput.value.trim();
            if (/^[0-9]{10}$/.test(phone)) {
                phoneInput.setCustomValidity("");
            } else {
                phoneInput.setCustomValidity("Phone number should
contain 10 digits only.");
            }
        }
```

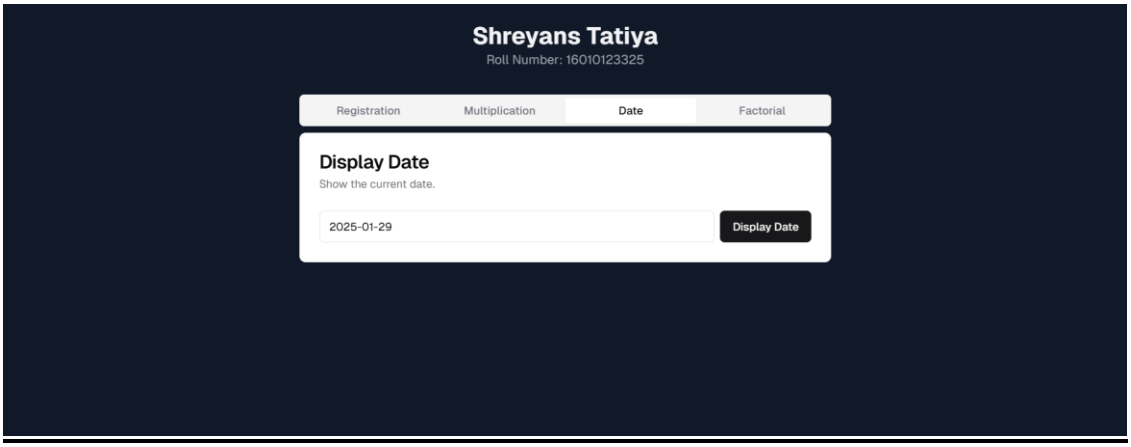
```
        nameInput.addEventListener("input", validateName);
        passwordInput.addEventListener("input", validatePassword);
        emailInput.addEventListener("input", validateEmail);
        phoneInput.addEventListener("input", validatePhone);

        registrationForm.addEventListener("submit", function(event) {
            event.preventDefault();
            if (registrationForm.checkValidity()) {
                alert("Registration successful!");
                registrationForm.reset();
            }
        });

        // Multiplication Table Generation
        function generateTable() {
            const numberInput = document.getElementById("numberInput");
            const tableContainer =
document.getElementById("multiplicationTable");
            const n = parseInt(numberInput.value);

            if (!isNaN(n)) {
                let table =
"<table><tr><th>Number</th><th>Multiplication</th></tr>";
                for (let i = 1; i <= 10; i++) {
                    table += `<tr><td>${n}</td><td>${n} x ${i} = ${n *
i}</td></tr>`;
                }
                table += "</table>";
                tableContainer.innerHTML = table;

                // Log multiplication table to console
                for (let i = 1; i <= 10; i++) {
                    console.log(`${n} x ${i} = ${n * i}`);
                }
            } else {
                tableContainer.innerHTML = "<p>Please enter a valid
number.</p>";
            }
        }

        // Display Date
        function displayDate() {
            const dateInput = document.getElementById("dateInput");
            const today = new Date();
            const date = today.getDate();
            const month = today.getMonth() + 1;
            const year = today.getFullYear();
            const formattedDate = `${year}-${month}-${date}`;
```

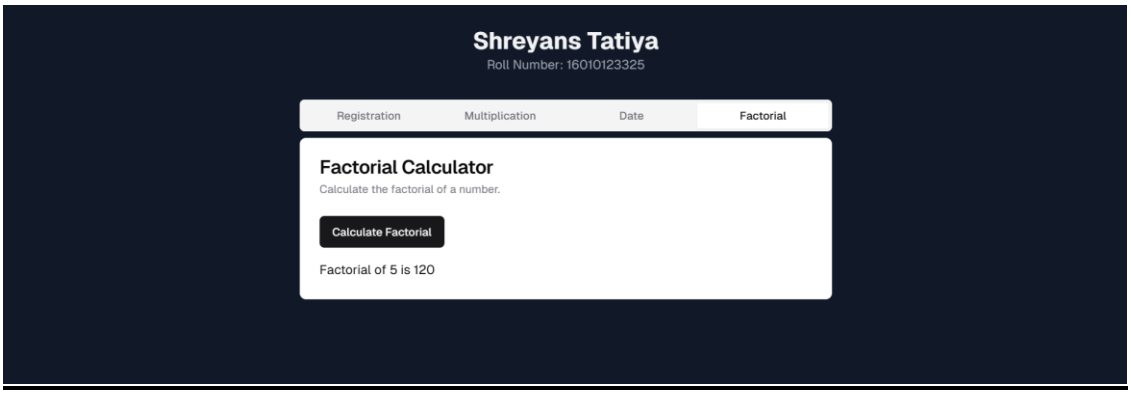
```
            dateInput.value = formattedDate;
        }

        // Factorial Calculator
        function calculateFactorial() {
            const n = parseInt(prompt("Enter a number: "));
            if (!isNaN(n)) {
                let factorial = 1;
                for (let i = 1; i <= n; i++) {
                    factorial *= i;
                }
                document.getElementById("factorialResult").textContent =
`Factorial of ${n} is ${factorial}`;
                alert(`Factorial of ${n} is ${factorial}`);
            } else {
                alert("Please enter a valid number.");
            }
        }

        // Initial multiplication table (as per the original script)
        const a = parseInt(prompt("Enter a number for initial
multiplication table: "));
        if (!isNaN(a)) {
            for (let i = 1; i <= 10; i++) {
                const result = a * i;
                console.log(`${a} x ${i} = ${result}`);
            }
        }
    </script>
</body>
</html>
```

Output-


Shreyans Tatiya
Roll Number: 16010123325
Registration
Multiplication
Date
Factorial
Registration Form
Enter your details to register.
Name
Shreyans T
Password
Email
shreyans.tatiya@gmail.com
Phone
9653209942
Register


Shreyans Tatiya
Roll Number: 16010123325
Registration
Multiplication
Date
Factorial
Multiplication Table
Generate a multiplication table.
Generate Table
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100


Shreyans Tatiya
Roll Number: 16010123325
Registration
Multiplication
Date
Factorial
Display Date
Show the current date.
2025-01-29
Display Date


Shreyans Tatiya
Roll Number: 16010123325
Registration
Multiplication
Date
Factorial
Factorial Calculator
Calculate the factorial of a number.
Calculate Factorial
Factorial of 5 is 120

Post Lab Objective with Ans :

What are the possible ways to create objects in JavaScript?

1. **Object Literals**:

   ```
   const obj = { key: "value" };
   ```

2. **Using the Object Constructor**:

   ```
   const obj = new Object();
   obj.key = "value";
   ```

3. **Using a Constructor Function**:

   ```
   function Person(name) {
     this.name = name;
   }
   const person = new Person("John");
   ```

4. **Using the class Keyword**:

   ```
   class Person {
     constructor(name) {
       this.name = name;
     }
   }
   const person = new Person("Jane");
   ```

5. **Using object. create()**:

   ```
   const p= { greet: () => "Hello" };
   const obj = Object.create(p);
   ```

What is the Difference between == and === operators

1. **== (Equality Operator)**:
   - Checks **value** equality after type coercion.
   - Example:

     ```
     5 == "5"; // true          null == undefined; // true
     ```

2. **=== (Strict Equality Operator)**:
   - Checks both **value** and **type** equality.
   - Example:

     ```
     5 === "5"; // false        null === undefined; // false
     ```

What is the difference between let and var

var

- The var is a keyword that is used to declare a variable
- Syntax -
  var name = value;
- The variables that are defined with var statement have function scope
- We can declare a variable again even if it has been defined previously in the same scope

let

- The let is also a keyword that is used to declare a variable
- Syntax -
  let name = value;
- The variables that are defined with let statement have block scope
- We cannot declare a variable more than once if we defined that previously in the same scope