Mini Project Report - 07

Master of Computer Application – Generative AI
Semester – I

Sub: Front-End Frameworks and Technologies

Topic: Student Reg Form Using DOM&EventHandlers

By
Name: MADAN KK
Reg no.: PROV/ASAC/MCA/7/25/078

Faculty Name: VEERA RAGHAV K

Faculty Signature: _____

Department of Computer Application
Alliance University
Chandapura - Anekal Main Road, Anekal
Bengaluru - 562 106

September 2025

INDEX

INTRODUCTION

Introduction to HTML

HTML (HyperText Markup Language) is the backbone of all websites. It is used to structure and organize content on the web so that browsers can display text, images, videos, and links in a readable way. Instead of running calculations like a programming language, HTML works by marking up content with different tags to tell the browser *what each part of the page means*.

For example, you can use headings to make titles stand out, paragraphs for text, images for visuals, and links to connect one page to another. HTML works together with **CSS** (for styling and design) and **JavaScript** (for interactivity) to create complete and dynamic websites.

Some of the most important HTML tags are:

- <html> → The container that wraps the entire webpage.
- <head> → Holds meta information, links to CSS/JS, and the page title.
- <title> → Sets the title shown in the browser tab.
- <body> → Holds the visible content like headings, text, images, and links.

Introduction to CSS

CSS (Cascading Style Sheets) is the language that makes websites visually attractive and user-friendly. While HTML provides the structure of a webpage, CSS is responsible for its design—controlling colors, fonts, spacing, layouts, and even animations. With CSS, the same HTML content can be presented in completely different styles,

giving web designers full creative control over how a site looks and feels.

In CSS:

- **Selector** → Targets the HTML element you want to style (e.g., h1).
- **Property** → Specifies the aspect to change (e.g., color, font-size).
- **Value** → Defines the exact style setting (e.g., blue, 20px).

Introduction to JS

JavaScript is a scripting language used to make web pages interactive and dynamic. It works with HTML and CSS to enhance user experience by responding to user actions like clicks or inputs. JS allows developers to update content, control multimedia, and animate elements without reloading the page. It can also handle form validation and interactive menus. Overall, JavaScript adds life and functionality to modern websites.

INPUT CODE :

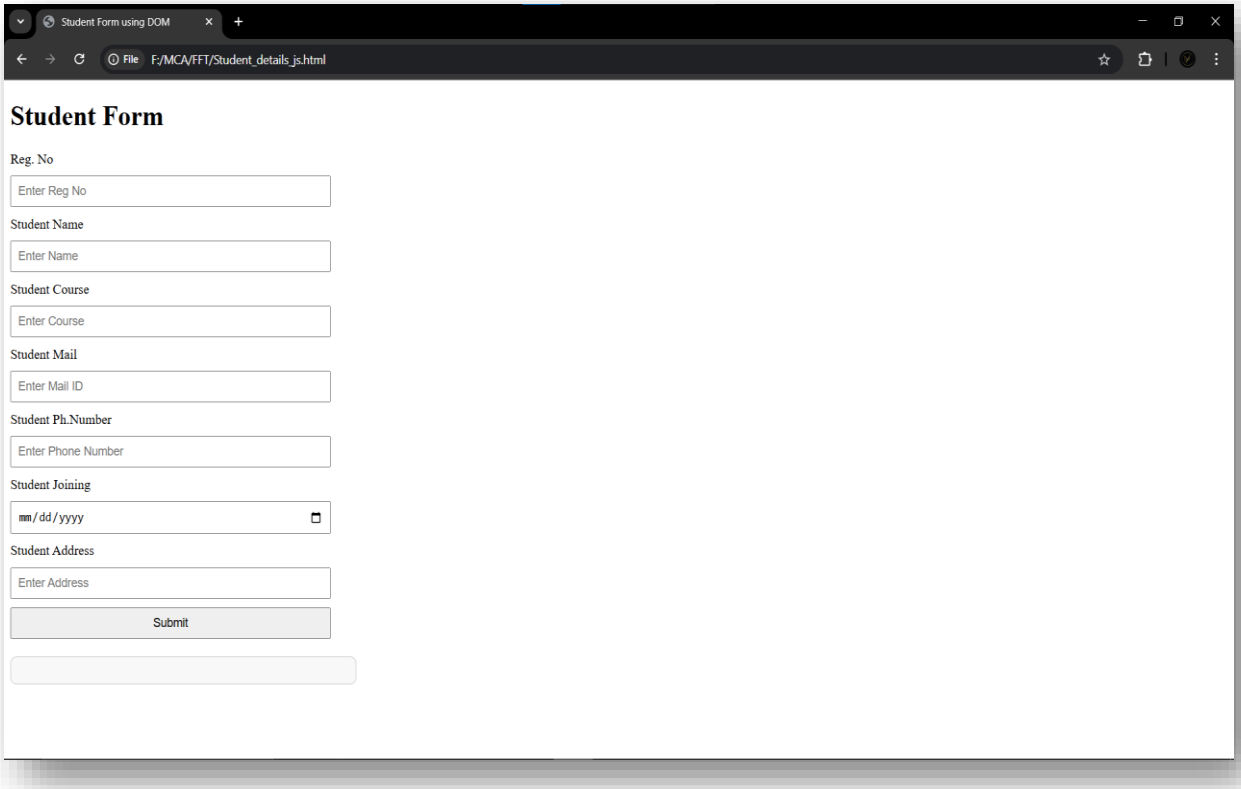
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Student Form using DOM</title>
    <style>
        form {
            display: grid;
            gap: 10px;
            max-width: 400px;
            margin-bottom: 20px;
```

```
        }
        input, button {
            padding: 8px;
            font-size: 14px;
        }
        #output {
            background-color: #f8f8f8;
            border: 1px solid #ccc;
            padding: 15px;
            max-width: 400px;
            border-radius: 8px;
        }
        h2 {
            color: #333;
        }

    </style>
</head>
<body>
    <h1>Student Form</h1>
    <form>
        <label>Reg. No</label>
        <input id="Reg" type="number" placeholder="Enter Reg
No">

        <label>Student Name</label>
        <input id="Name" type="text" placeholder="Enter
Name">

        <label>Student Course</label>
        <input id="course" type="text" placeholder="Enter
Course">

        <label>Student Mail</label>
        <input id="mail" type="text" placeholder="Enter Mail
```

```
ID">

        <label>Student Ph.Number</label>
        <input id="num" type="number" placeholder="Enter
Phone Number">

        <label>Student Joining</label>
        <input id="join" type="date">

        <label>Student Address</label>
        <input id="address" type="text" placeholder="Enter
Address">

        <button id="btn" type="submit" >Submit</button>
    </form>


    <div id="output"></div>

    <script>
        const Reg = document.getElementById("Reg");
        const Name = document.getElementById("Name");
        const course = document.getElementById("course");
        const mail = document.getElementById("mail");
        const num = document.getElementById("num");
        const join = document.getElementById("join");
        const address = document.getElementById("address");
        const form = document.querySelector("form");
        const output = document.getElementById("output");


        form.addEventListener("submit", function(event) {
            event.preventDefault();

            output.innerHTML = `
```

```
            <h2>Submitted Details</h2>
            <p><b>Reg No:</b> ${Reg.value}</p>
            <p><b>Name:</b> ${Name.value}</p>
            <p><b>Course:</b> ${course.value}</p>
            <p><b>Mail ID:</b> ${mail.value}</p>
            <p><b>Phone Number:</b> ${num.value}</p>
            <p><b>Date of Joining:</b> ${join.value}</p>
            <p><b>Address:</b> ${address.value}</p>
        `;

    });
  </script>
</body>
</html>
```

OUTPUT:



CONCLUSION:

This mini project highlights how combining HTML, CSS and Js can transform plain content into a well-structured and visually appealing Calculator. Through this project, I gained hands-on experience in organizing content with HTML and enhancing its design with CSS. While the design is simple, it demonstrates the practical use of web technologies in creating professional and presentable documents, laying a solid foundation for building more advanced web projects in the future.