# Mini Project Report - 06

Master of Computer Application – Generative AI
Semester – I

**Sub: Front-End Frameworks and Technologies**

**Topic: Calculator**

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


| S.No | CONTENTS | PAGE NO |
|---|---|---|
| 1 | Introduction to HTML and CSS | 1 |
| 2 | Source Code | 2 |
| 3 | Output | 4 |
| 4 | Conclusion | 5 |

# 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).

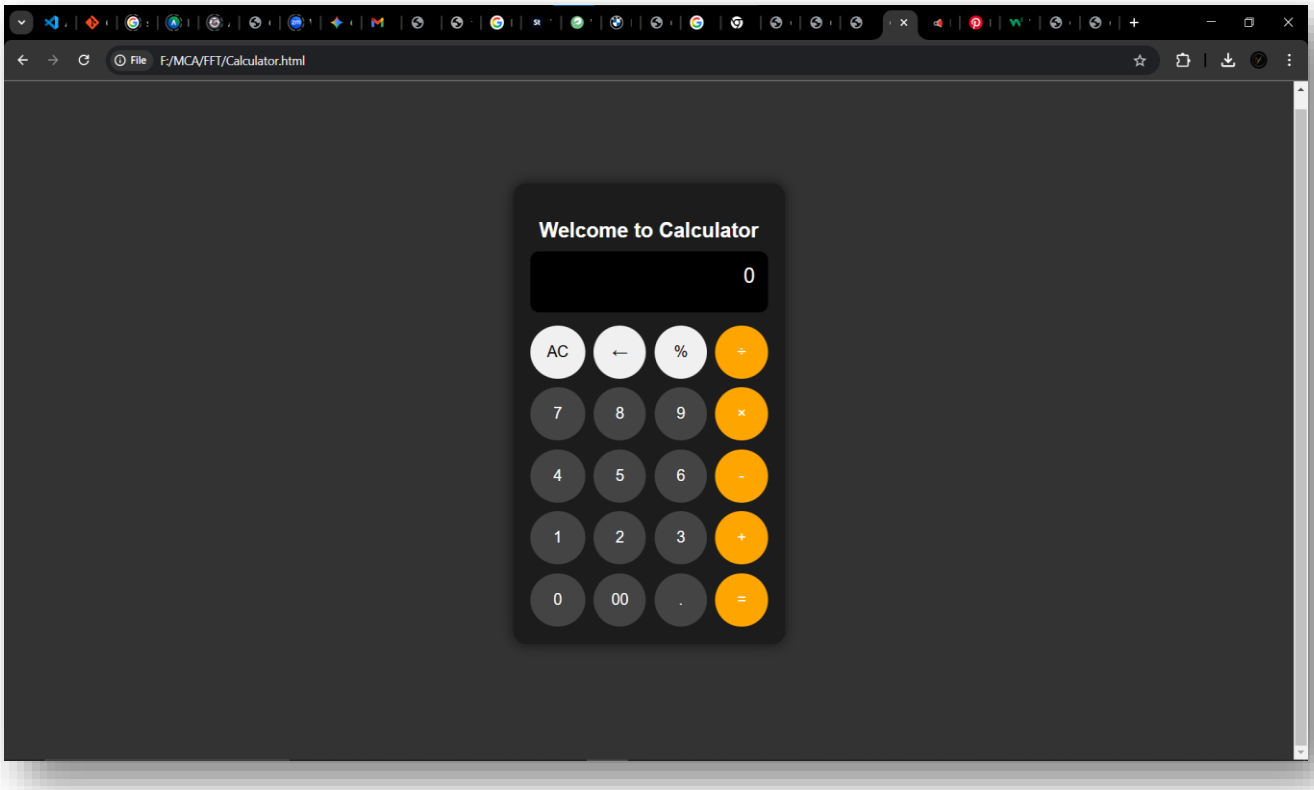
## INPUT CODE :

```
<head>
 <meta charset="UTF-8">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Calculator</title>
 <style>
     body {
     display: flex;
     justify-content: center;
     align-items: center;
     height: 100vh;
     background: #333;
     font-family: Arial, sans-serif;
     }
     .calculator{
     background: #1c1c1c;
     padding: 20px;
     border-radius: 15px;
     box-shadow: 0px 0px 15px rgb(0,0,0,0.6);
     width: 280px;
     }
     .calculator h2{
     text-align: center;
     color: white;
     margin-bottom: 10px;
     }
     .display{
```

```
background: #000;
color: white;
text-align: right;
padding: 15px;
font-size: 24px;
border-radius: 10px;
margin-bottom: 15px;
height: 40px;
}
.buttons{
display: grid;
grid-template-columns: repeat(4, 1fr);
gap: 10px;
}
button{
padding: 20px;
font-size: 18px;
border: none;
border-radius: 50%;
cursor: pointer;
transition: 0.2s;
}

button:active{
transform: scale(0.95);
}
.btn {
background: #444;
color: white;
}
.operator{
background: orange;
color: white;
}
.special{
background:
color: black;
}
.equal{
border-radius: 40px;
grid-column: span 1;
```

```
      }
  </style>
</head>
<body>
   <div class="calculator">
      <h2>Welcome to Calculator</h2>
      <div class="display">0</div>
      <div class="buttons">
         <button class="special">AC</button>
         <button class="special">←</button>
         <button class="special">%</button>
         <button class="operator">÷</button>

         <button class="btn">7</button>
         <button class="btn">8</button>
         <button class="btn">9</button>
         <button class="operator">×</button>

         <button class="btn">4</button>
         <button class="btn">5</button>
         <button class="btn">6</button>
         <button class="operator">-</button>

         <button class="btn">1</button>
         <button class="btn">2</button>
         <button class="btn">3</button>
         <button class="operator">+</button>

         <button class="btn">0</button>
         <button class="btn">00</button>
         <button class="btn">.</button>
         <button class="operator equal">=</button>
      </div>
   </div>
</body>
</html>
```

# OUTPUT:


File
F:/MCA/FFT/Calculator.html
Welcome to Calculator
0
AC
←
%
÷
7
8
9
×
4
5
6
-
1
2
3
+
0
00
.
=

# CONCLUSION:

This mini project highlights how combining HTML and CSS 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.