A Simple Calculator Report

This report provides an overview of my second project(**INTERMEDIATE**) i.e. a simple calculator made with HTML, CSS and Javascript. It performs the basic arithematic calculations.

Contents:

1. Introduction

- **Project Overview**: This report details the development of a basic calculator capable of performing fundamental arithmetic operations (addition, subtraction, multiplication, and division) using HTML, CSS, and JavaScript.
- **Technologies Used**:

  - **HTML**: For structuring the calculator interface.
  - **CSS**: For styling the calculator.
  - **JavaScript**: For implementing the functionality.

2. HTML Structure

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Innovative Calculator</title>
   <link rel="stylesheet" href="calculator.css">
```

```html
</head>
<body>
  <div class="calculator ">
    <input type="text" id="display" disabled />
    <div class="buttons">
      <button onclick="clear_display()">C</button>
      <button onclick="append_to_display('/')">/</button>
      <button onclick="append_to_display(')">')</button>
      <button onclick="append_to_display('-')">-</button>
      <button onclick="append_to_display('7')">7</button>
      <button onclick="append_to_display('8')">8</button>
      <button onclick="append_to_display('9')">9</button>
      <button onclick="append_to_display('+')">+</button>
      <button onclick="append_to_display('4')">4</button>
      <button onclick="append_to_display('5')">5</button>
      <button onclick="append_to_display('6')">6</button>
      <button class="equal" onclick="calculate_result()">=</button>
      <button onclick="append_to_display('1')">1</button>
      <button onclick="append_to_display('2')">2</button>
      <button onclick="append_to_display('3')">3</button>
      <button onclick="append_to_display('0')">0</button>
    </div>
  </div>
  <script src="calculator.js"></script>
</body>
</html>
```

3. CSS Styling

```css
* {
  box-sizing: border-box;
}

body {
  display: flex;
  justify-content: center;
  align-items: center;
```

```css
    height: 700px;
    background-color: #050505;
    font-family: 'Arial', sans-serif;
}

.calculator {
    background: rgb(31, 27, 27);
    border-radius: 10px;
    box-shadow: 0 4px 20px rgba(0, 0, 0, 0.2);
    padding: 20px;
    width: 300px;


}

#display {
    width: 100%;
    height: 40px;

    /* background-color: #e7e7e7; */
    background-color: white;
    border-radius: 5px;
    font-size: 24px;
    text-align: right;
    padding-right: 10px;
    margin-bottom: 20px;
}

.buttons {
    display: grid;
    grid-template-columns: repeat(4, 1fr);
    gap: 10px;
}

button {
    padding: 20px;
    border: 2px solid white;
```

```css
    border-radius: 25px;
    font-size: 18px;
    background-color:   #ca6dab;

    color: black;
    cursor: pointer;
    transition: background-color 0.3s, transform 0.1s;
}

button:hover {
    background-color: white;
}

button:active {
    transform: scale(0.95);
}

.equal {
    background-color: #e67206;
}

.equal:hover {
    background-color: white;
}

.calculator:hover{
    border: 2px solid white;
}
```

4. JavaScript Structure

```javascript
function append_to_display(value) {
    const display = document.getElementById("display");
    display.value += value;
}

function clear_display() {
    const display = document.getElementById("display");
```

```
    display.value = "";
}

function calculate_result() {
    const display = document.getElementById("display");
    try {
        display.value = eval(display.value);
    } catch {
        display.value = "Error";
        setTimeout(clear_display, 1500);
    }
}
```

5. Features

User Interface

- **Display Screen**:
  - Shows the current input and results.
- **Buttons**:
  - Numeric buttons (0-9).
  - Arithmetic operation buttons (+, -, ×, ÷).
  - Functional buttons (C for clear, = for equals).

Functionality

- **Basic Operations**:
  - Addition, subtraction, multiplication, and division.
- **Clear Function**:
  - Resets the display and input.
- **Input Validation**:
  - Handles invalid inputs gracefully.

6. Screenshots

C	/	*	-
7	8	9	+
4	5	6	=
1	2	3	0