**Interactive Calculator**

**Introduction**

This report outlines the development of an interactive calculator application. The calculator project comprises three main files: HTML, CSS, and JavaScript. The goal was to create a simple, user-friendly calculator that performs basic arithmetic operations with a visually appealing interface.

**Project Structure**

The project consists of the following components:

1. HTML (calculator.html): Defines the structure and layout of the calculator.

2. CSS (calculator.css): Provides the styling for the calculator, including layout, colours, and button styles.

3. JavaScript (calculator.js): Implements the functionality of the calculator, handling user inputs and performing calculations.

**HTML (calculator.html)**

The HTML file defines the basic structure of the calculator. It includes an input field for displaying results and buttons for digits and arithmetic operations. The buttons are arranged in a grid layout for easy access.

Key Elements:

- Input field to display the result (`<input type="text" id="result" disabled>`).

- Buttons for digits (0-9) and basic operations (+, -, \*, /).

- Clear button (`C`) to reset the display.

- Equals button (`=`) to calculate the result.

**CSS (calculator.css)**

The CSS file provides styling to make the calculator visually appealing. It includes styles for the calculator container, input field, and buttons. The layout is designed to be responsive and user-friendly.

Key Styles:

- Flexbox for centering the calculator on the page.

- Grid layout for buttons.

- Button styles for visual feedback on hover and active states.

**JavaScript (calculator.js)**

The JavaScript file implements the functionality of the calculator. It includes functions to handle button clicks, perform arithmetic operations, and update the display.

Key Functions:

- `appendNumber(number)`: Adds a digit to the current input.

- `appendOperator(operator)`: Adds an operator to the current input.

- `clearResult()`: Clears the input field.

- `calculateResult()`: Evaluates the expression and updates the display.

**Conclusion**

The interactive calculator project successfully combines HTML, CSS, and JavaScript to create a functional and visually appealing calculator. The clear structure and responsive design ensure a user-friendly experience, while the JavaScript functions provide the necessary operations for basic arithmetic calculations. This project demonstrates effective use of web technologies to build a simple yet useful application.