Assignment #1
JavaScript Calculator Implementation

Overview

This assignment focuses on strengthening your <u>JavaScript skills</u> by implementing the core functionality of a basic calculator. <u>The provided HTML and CSS files already define the user interface (UI),</u> and **your primary task is to develop the JavaScript logic that will make the calculator fully operational**. This assignment will enhance your ability to manipulate the Document Object Model (DOM), handle events, and implement essential programming concepts such as functions, operators, and event-driven programming. By the end of this assignment, you will gain hands-on experience with interactive web development and practical problem-solving techniques in JavaScript.

Aim

The primary goal of this assignment is to develop a fully functional JavaScript calculator that can perform standard arithmetic operations. Through this assignment, you will enhance your understanding of JavaScript's interaction with HTML elements, event handling, and basic computation logic. This assignment is designed to reinforce key JavaScript programming concepts and make it a valuable step in your learning journey toward becoming proficient in web development.

Learning Objectives

By completing this assignment, you will:

- Gain experience in manipulating the DOM using JavaScript.
- Develop JavaScript code that dynamically interacts with an existing HTML and CSS structure.
- Use **event listeners** to detect and respond to user interactions (e.g., button clicks).
- Perform arithmetic operations based on user input.
- Implement clear and delete functionalities for a calculator.

Prior Knowledge Required

To successfully complete this assignment, you should have familiarity with the following topics:

- **HTML & CSS Basics:** Understanding of how elements are structured and styled on a webpage.

- **JavaScript Fundamentals:** Variables, functions, and operators.

- **DOM Manipulation:** Selecting and modifying elements using JavaScript.

- **Event Handling:** Adding event listeners and responding to user actions.

- **Basic Debugging Skills:** Using the browser console to test and debug JavaScript code.

Instructions

1. Download the assignment files (HTML and CSS) that have been given to you. **You are only required to create a new JavaScript file (name the file as "calculator.js" and write code in this file**.

2. Implement the following functionalities in JavaScript file:

   - Clicking on a number should display the corresponding digit in the calculator's input field.
   - Clicking on an operator (+, -, *, /) should store the current number and the operation.
   - Clicking the equals (=) button should evaluate the expression and display the result.
   - The clear button (C) should reset the input field and any stored values.
   - The delete button (⊠) should remove the last entered digit from the input field.

   Note: Some of the names of functions you need to implement are as follows:

   calculate_percentage(), clear_entry(), button_clear(), backspace_remove(), division_one(), power_of(), square_root(), plus_minus(), button_number(button), showSelectedOperator()

3. Ensure your JavaScript code is well-structured and commented to explain your approach.
4. Test your implementation with different inputs to confirm that all functionalities work correctly.

Expected Output
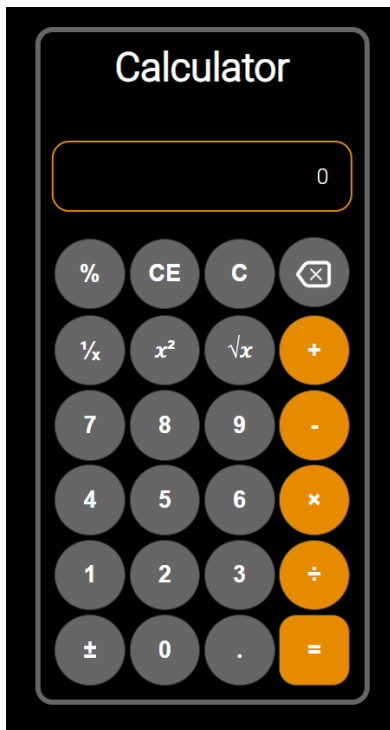
What to Turn into Blackboard

- A JavaScript file (calculator.js) containing your implementation.
- A short README file (optional) explaining how your code works (optional but recommended).

- ▪ Submit all the required files in a single ZIP folder named **JS_Calculator_YourName_R#.zip** (for example, JS_Calculator_Maaz_Amjad_R#12345678.zip**)**

Concepts Used in the Assignment

This assignment will require you to apply the following programming concepts:

- ▪ **Event Listeners:** Detecting and responding to button clicks.

- ▪ **DOM Manipulation:** Updating the input field and displaying results dynamically.

- ▪ **Arithmetic Operations:** Implementing addition, subtraction, multiplication, and division.

- ▪ **String Manipulation:** Handling user inputs and converting them into numerical values.

- ▪ **Conditional Statements:** Ensuring valid operations and handling errors.

- ▪ **Functions:** Writing reusable functions such as clear_entry(), calculate_percentage(), button_number(), etc.

- ▪ **Data Handling:** Storing and updating values dynamically as users interact with the calculator.

Helpful Resources
- • DOM Manipulation Guide
- • JavaScript Event Listeners
- • JavaScript Arithmetic Operators