Summer Internship 2023

Designing and Implementation of a Scientific Calculator

Github repo link- https://github.com/Kashish-varshney123/Dabotics-Task2

Linkedin profile - www.linkedin.com/in/kashish-varshney44

Submitting to - Dabotics India Pvt. Ltd.

Submitted by - Kashish Varshney

Introduction:

This documentation provides a detailed overview of a scientific calculator designed and implemented using HTML, CSS, and JavaScript (JS). The calculator is built to perform a wide range of mathematical operations, including basic arithmetic, trigonometry, logarithms, and more. The user interface is designed with a modern and intuitive look, making it easy to use for both beginners and advanced users.

1. HTML Structure:

The calculator's user interface is built using HTML to create the layout and structure. The HTML file contains various elements like buttons, display screen, and other components required for the calculator's functionality. Additionally, the layout is designed to be responsive and work on different screen sizes and devices.

2. CSS Styling:

The CSS file is responsible for styling the calculator's user interface. It defines the color schemes, font styles, button shapes, and overall appearance of the calculator. The CSS ensures a visually appealing and user-friendly design.

3. Basic Arithmetic Operations:

The calculator allows users to perform basic arithmetic operations like addition, subtraction, multiplication, and division. The user can input numbers using the numeric buttons and perform calculations with the help of the operator buttons (+, -, *, /).

4. Advanced Mathematical Functions:

The scientific calculator includes various advanced mathematical functions such as trigonometric functions (sine, cosine, tangent), logarithmic functions (log, ln), square root, power, and more. These functions are accessible through specialized buttons, expanding the calculator's utility for more complex calculations.

5. Memory Functionality:

The calculator also offers memory functionality, allowing users to store and retrieve values from memory. This feature is particularly useful when performing multiple calculations that require the same value repeatedly.

6. Clear and Delete Functionality:

To improve user experience, the calculator includes clear and delete buttons. The "Clear" button clears the display, while the "Delete" button removes the last inputted character, helping users to correct any mistakes made during input.

7. Error Handling:

To enhance usability, the calculator implements error handling. If the user attempts an invalid operation, such as dividing by zero, an appropriate error message is displayed, preventing unexpected behavior.

8. User Interaction:

The calculator responds to user interactions, such as mouse clicks and keyboard inputs. Button clicks are registered and translated into actions by the JavaScript code, providing immediate feedback to users.

9. Modular and Organized JavaScript:

The JavaScript code is designed to be modular and organized, ensuring easy maintenance and scalability. It handles user inputs, performs calculations, and updates the display screen with the appropriate results.

Conclusion:

In conclusion, the scientific calculator implemented using HTML, CSS, and JavaScript offers a comprehensive set of features, making it a powerful tool for performing various mathematical calculations. The well-designed user interface, along with error handling and memory functionality, ensures a smooth and enjoyable user experience. The calculator's modular JavaScript code ensures easy maintenance and extensibility for future enhancements.