Working Calculator in Python

Coder : Harshit Soni

Registration Number: 25BCE11030

□ Project Overview

This project is a **console-based working calculator built in Python**, capable of performing fundamental arithmetic operations.
It is designed with clear user interaction, structured logic, and strong error-handling to ensure smooth operation.
The project demonstrates essential Python skills and serves as an excellent beginner-level program to understand computational logic.

□ Objective

The primary objectives of this project are:

- To implement the core arithmetic operations using Python.
- To understand how user input, decision-making, and output formatting work.
- To practice function-based programming and modular code structure.
- To build a reliable, user-friendly calculator tool.

□□ Features

✔□ Addition
✔□ Subtraction
✔□ Multiplication
✔□ Division (with zero-division protection)
✔□ Modulus
✔□ Power calculation
✔□ Floor division
✔□ Input validation and error handling
✔□ Clean and interactive user prompts

□ Project Structure

```
□ Working-Calculator
  ├ □ calculator.py
  ├ □ README.md
```

□ How the Calculator Works

- The user is prompted to enter two numbers.
- The user selects an operation such as addition, subtraction, multiplication, division, etc.
- The calculator processes these inputs using conditional logic.
- If the input is invalid (wrong operator or non-numeric value), the program handles it gracefully.
- A final result is displayed in a clear and readable format.

□ Concepts Demonstrated

- **Variables and Data Types**
- **Conditional Statements** (`if-elif-else`)
- **Arithmetic Operators**
- **Functions in Python**
- **Exception Handling** (`try-except`)
- **Basic input/output handling**

□□ Requirements

- Python 3.x
- Any IDE or text editor (VS Code, PyCharm, Sublime Text, etc.)
- Command-line / Terminal to run the script

To execute the project:

```
python calculator.py
```

□ Applications

This calculator can be used for:

- Basic mathematical calculations
- Learning how computation works inside programming
- Understanding console-based UI
- Extending into GUI applications in the future

□ Future Enhancements

Possible improvements include:

- Graphical User Interface (GUI) using **Tkinter or PyQt**
- Scientific functions (trigonometry, logarithms, etc.)
- Calculation history tracking
- A more advanced multi-operation calculator
- Keyboard shortcuts for faster input

□ Academic Declaration

I, **Harshit Soni (Registration No. 25BCE11030)**, hereby declare that this Python calculator project is created by me solely for learning and academic purposes.
All logic, structure, and design decisions are original and implemented by me.