TITLE: CodTech IT Solutions Internship – Task Documentation: **"SIMPLE CALCULATOR WITH ADVANCE FEATURES"** Using PYTHON.

INTERN INFORMATION:

Name: Bhagya Sri Hani Allu

ID: COD5944

INTRODUCTION

In the Real-Time Simple calculator is a versatile tool designed to make your mathematical calculations quick and effortless. This Simple Calculator with Advanced Features project is an enhanced version of the traditional calculator implemented in Python. While retaining the Simplicity and ease of use of a basic calculator, this calculator incorporates additional functionalities to cater to more complex mathematical needs. By leveraging Python's versatility and power, this calculator offers users a comprehensive tool for performing various calculations efficiently.

Purpose and Objectives:

The Primary purpose of the Simple Calculator with Advanced Features is to provide users with a versatile tool that goes beyond basic arithmetic operations. The objectives of this project include:

1.Enhanced Functionality:

In Addition to standard arithmetic operations such as addition, Subtraction, Multiplication, division, the calculator will support advanced mathematical functions such as exponentiation and quit functions.

2.Memory Functions:

Incorporating memory features such as storing and recalling previous calculations, allowing users to easily reuse or reference results.

3.Customization:

Providing options for customization, including user preferences for display format, precision, and scientific notation.

4. Error Handling and Validation:

Implementing robust error handling mechanisms to detect and handle invalid input, ensuring reliable operation and preventing unexpected errors.

5.Cross-Platform Compatibility:

Ensuring that the calculator is compatible with various operating systems and can be easily deployed on different platforms.

Implementation

- Set Up your Development Environment: Install Python and Visual Studio Code on your system. Open VS code and create a new Python file for your calculator Project.
- Define the Basic Structure: Begin by outlining the basic structure of the calculator application. This includes creating a main function or class to handle user input, perform calculations, and display results.
- Implement Input Handling: Develop functions to handle user input, including numbers, operators, and special commands. Consider using Python's input() function or a graphical user interface(GUI) library for user interaction.
- Support Basic Arithmetic Operations: Implement functions to Perform basic arithmetic operations such as addition, subtraction, multiplication, division and exponentiation. Ensure that these operations handle both integer and floating-point numbers correctly. Utilize Python's ** operator for exponentiation function.
- Testing and Validation: Test the calculator thoroughly to ensure that it produce correct results for various inputs and scenarios.Validate its performance against expected outcomes and verify that all advanced features function as intended.
- Run Your Calculator: Once you are confident in your implementation, run your calculator script in VS Code to test its functionality interactively.

CODE EXPLAINATION

In this code Structure explained in the Following:

- We define five functions (‘**Add’, ‘Subtract’, ‘Multiply’,’ Divide’ and ‘Exponentiate**) to perform basic arithmetic operations. The ‘exponentiate’ function raises the first number to the power of the second number using the ‘**’ operator.
- The ‘**calculator**’ function displays a menu of operations for the user to choose from and prompts the user to input two numbers.
- Each function takes two arguments (‘x’ and ‘y’) representing the operands and returns the result of the operation.
- A ‘**display_menu’** function is defined to print menu options for the user to choose from.
- The ‘**main’** function is the entry point of the program. It welcomes the user and prompts them to enter an expression. It then calls the calculate function to compute the result and prints it to the console. The loop continues until the user enters ‘quit’, to exit the program.
- The user is asked to enter two numbers for each operation, and the result is printed accordingly.
- Error handling is implemented in the division function to prevent division by zero error. Ex (2/0 = division by zero error occurs).
- The program continues to run until the user chooses the “quit” option (‘**choice =’6’**), at which point the loop breaks, and the program exits.

- The **‘if_ _name_ _ = =”_ _main_ _”:** ‘block ensures that the ‘main’ function is only called if the script is executed directly, not when it’s imported as a module into another script.
- The code prompts the user to enter an expression and accepts it as input. If the user enters ‘quit’, the program exits. Otherwise, it passes the expression to calculate function for evaluation.

  This code provides a basic implementation of a calculator that can handle both simple and complex mathematical expressions.

Marking Tasks as Completed:

Utilizes the code by tasks marked as completed is when a task is clicked implemented the functions like addition , subtraction, multiplication, division, exponentiate to perform basic arithmetic operations. Run the task and see the output in the VS code terminal.

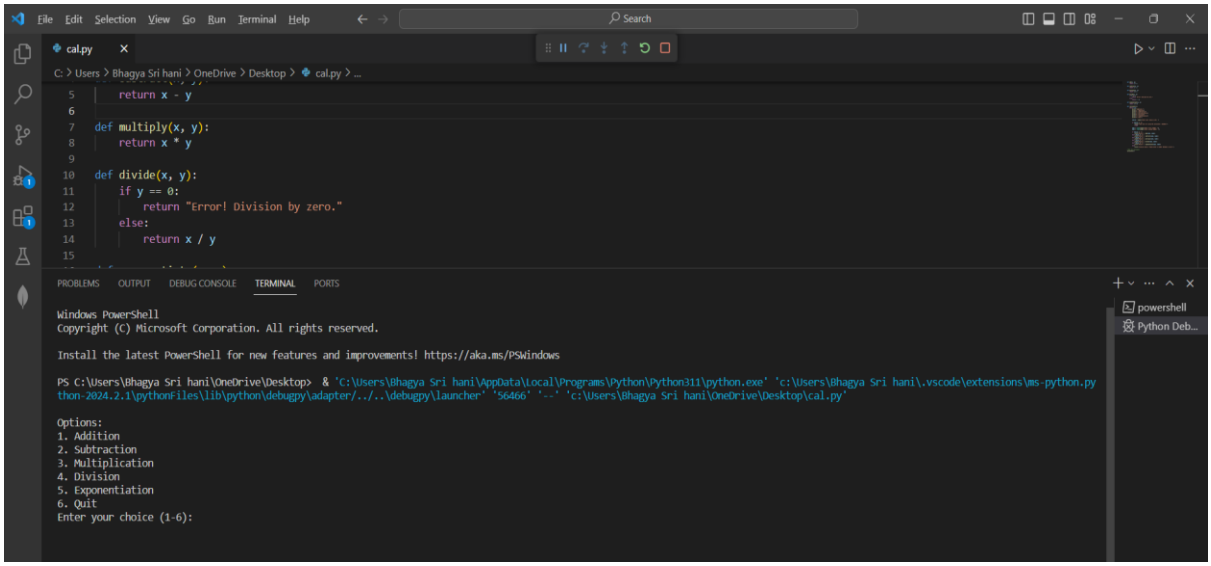
Removing Tasks:

The close button (‘X’) added to each task allows users to remove tasks from the code. This is shows the removing tasks in the project terminal.

CONCLUSION:

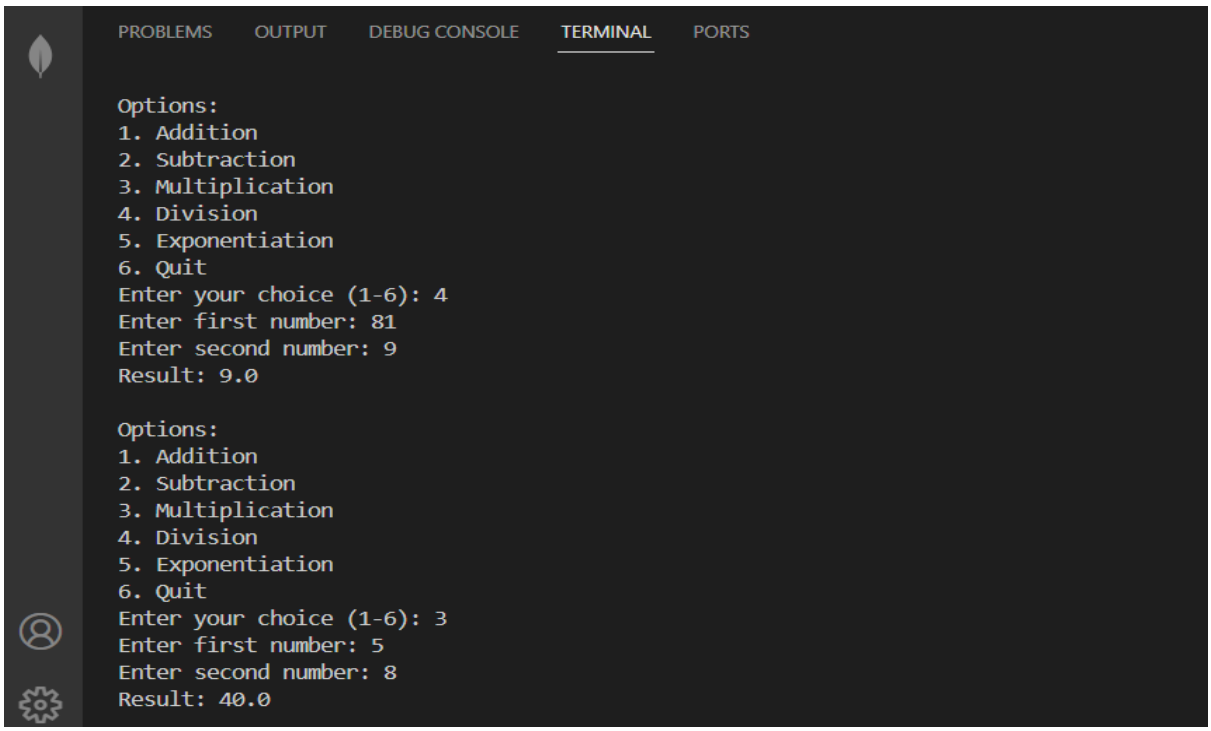
In conclusion, the Simple Calculator with Advanced Features by using Python project has successfully achieved its objectives of providing users with a versatile and user-friendly tool for performing a wide range of mathematical calculations. This project has successfully combined the simplicity of basic calculator with advanced functionalities to cater to a wide range of user needs. Whether performing simple arithmetic or tackling complex mathematical expressions, this calculator is equipped to handle the task efficiently and accurately. Further enhancements and refinements can be made in the future to continue improving the functionality and usability of the calculator.

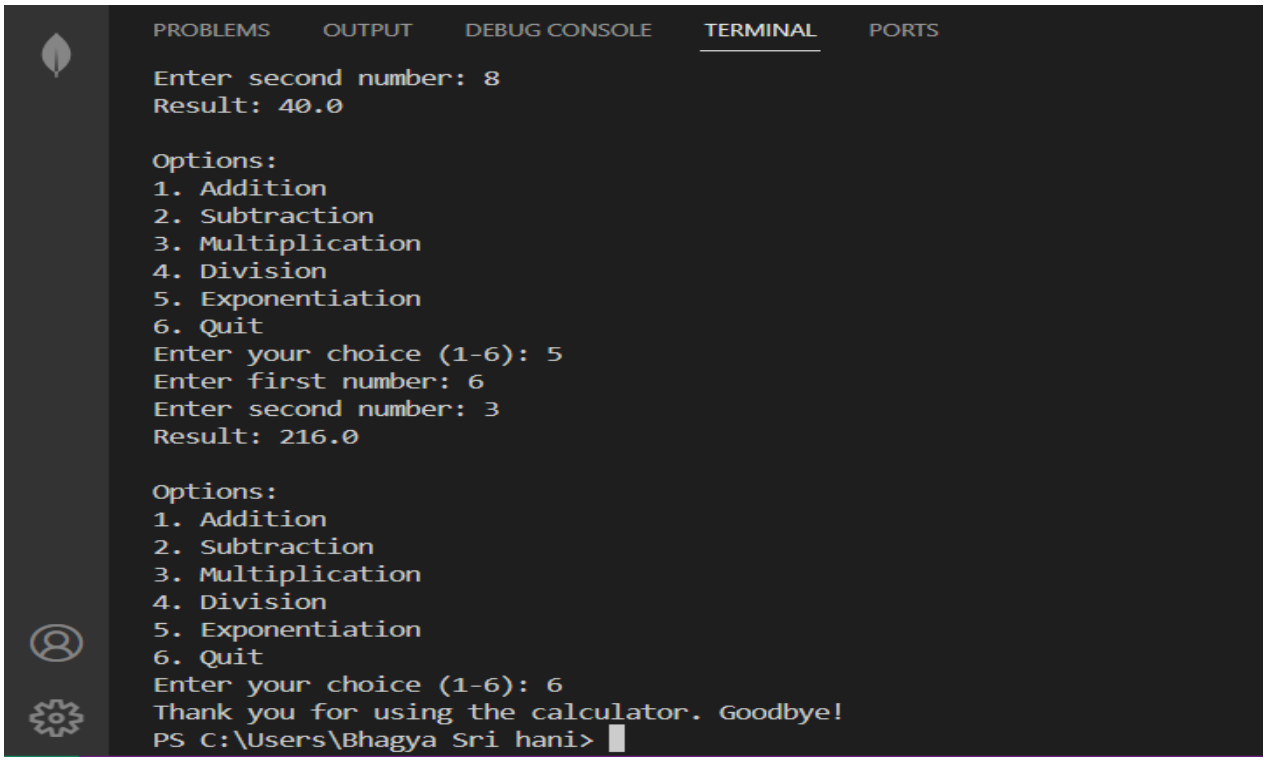
OUTPUT:




```
PS C:\Users\Bhagya Sri hani> & "C:/Users/Bhagya Sri hani/AppData/Local/Programs/Python/Python311/python.exe" "c:/Users/Bhagya Sri hani/OneDrive/Desktop/cal.py

Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 2
Enter first number: 20
Enter second number: 3
Result: 17.0
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 1
Enter first number: 12
Enter second number: 16
Result: 28.0

Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 4
Enter first number: 16
Enter second number: 0
Result: Error! Division by zero.
```


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 4
Enter first number: 81
Enter second number: 9
Result: 9.0
Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 3
Enter first number: 5
Enter second number: 8
Result: 40.0


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Enter second number: 8
Result: 40.0
Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 5
Enter first number: 6
Enter second number: 3
Result: 216.0
Options:
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exponentiation
6. Quit
Enter your choice (1-6): 6
Thank you for using the calculator. Goodbye!
PS C:\Users\Bhagya Sri hani>