TITLE: CodTech IT Solutions Internship - Task Documentation:"**simple calculator with advance features**," Using PYTHON PROGRAMMING

INTERN INFORMATION:

Name: Akash Siliveru

ID: C0D5431

INTRODUCTION

This document provides a detailed explanation of the Python code for a simple calculator with advanced features. The calculator allows users to perform basic arithmetic operations (addition, subtraction, multiplication, and division) along with exponentiation and Quit option.

Functionality Overview

The program offers the following functionalities:

User Interface: Presents a menu-driven interface for selecting the desired operation.

Input Validation: Ensures users enter valid numerical values.

Error Handling: Catches division by zero errors and provides an informative message.

Calculations: Performs the chosen mathematical operation on the provided numbers.

Output: Displays the results of the calculations.

Implementation

➢ **Define functions for basic arithmetic operations:** Create functions for addition, subtraction, multiplication, and division. These functions will take two numbers as input and return the result of the operation

➢ **Take input from the user:** Prompt the user to enter the operation they want to perform and the numbers on which they want to perform the operation.

> ➤ **Perform the requested operation:** Based on the user's input, call the appropriate function to perform the operation on the given numbers.

> ➤ **Display the result:** Print the result of the operation to the user.

CODE EXPLAINATION

Code Structure

The code is well-organized and utilizes several functions for better readability and maintainability.

Function Name Description

- add(x, y) Performs addition of two numbers.

- subtract(x, y) Performs subtraction of two numbers.

- multiply(x, y) Performs multiplication of two numbers.

- divide(x, y) Performs division of two numbers with error handling for division by zero.

- exponentiate(x, y) Calculates x raised to the power of y.

- get_user_input() Handles user input, ensuring valid numbers are entered.

- display_menu() Displays the available menu options for the calculator.

- main() The entry point of the program, controlling the overall program flow.

Function Descriptions:

I. **add(x, y)**: This function takes two numbers (x and y) as input and returns their sum.

II. **subtract(x, y):** This function subtracts the second number (y) from the first number (x) and returns the difference.

III. **multiply(x, y):** This function multiplies two numbers (x and y) and returns the product.

IV. **divide(x, y):** This function divides the first number (x) by the second number (y).

It checks for division by zero. If encountered, it prints an error message and returns None.

Otherwise, it returns the result of the division.

V. **exponentiate(x, y):** This function raises the first number (x) to the power of the second number (y) and returns the result.

VI. **get_user_input():** This function prompts the user to enter two numbers.

It uses a loop to continuously ask for input until valid numerical values are provided.

It utilizes a try-except block to handle potential ValueError exceptions that might occur during conversion from string to float.

VII. **display_menu():** This function displays the menu with a list of available options for the calculator.

VIII. **main():** This function serves as the entry point of the program.

It runs in a continuous loop:

➢ Displays the menu using the display_menu function.

➢ Prompts the user for a choice (1-6) and stores it in the choice variable.

Validates the user's choice.

➢ If the choice is valid (1-5), it calls the get_user_input function to retrieve the numbers.

➢ Based on the chosen operation (indicated by the choice), it calls the appropriate function (add, subtract, multiply, exponentiate) to perform the calculation.

➢ The calculated result is then displayed.

➢ If the user chooses option 6 (Quit), the loop terminates, and the program exits.

➢ For any invalid user input (choice outside the range 1-6), an error message Is displayed.

Error Handling

1. The code incorporates error handling to gracefully address potential issues:

2. The divide function checks for division by zero and prevents the program from crashing. It displays an error message and returns None to indicate an error.

3. The get_user_input function uses a try-except block to handle situations where the user enters non-numerical values. It prompts the user to re-enter valid numbers.

CONCLUSION

This well-structured code demonstrates the implementation of a simple calculator with error handling and user interaction in Python. It provides a foundation for understanding basic mathematical operations, function usage, and error handling mechanisms in the language.

OUTPUT

```
Python 3.7.8 (tags/v3.7.8:4b47a5b6ba, Jun 28 2020, 08:53:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============================= RESTART: C:/Users/ramak/AppData/Local/Programs/Python/Python37/cal.py =============================

*** Simple Calculator with Advanced Features ***
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exponentiate
6. Quit
Enter your choice (1-6):
|
```

```
Python 3.7.8 (tags/v3.7.8:4b47a5b6ba, Jun 28 2020, 08:53:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============================= RESTART: C:/Users/ramak/AppData/Local/Programs/Python/Python37/cal.py =============================

*** Simple Calculator with Advanced Features ***
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exponentiate
6. Quit
Enter your choice (1-6):
1
1
Enter the first number: 22
22.0
Enter the second number: 33
33.0
22.0 + 33.0 = 55.0

Invalid choice. Please enter a number between 1 and 6.

*** Simple Calculator with Advanced Features ***
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exponentiate
6. Quit
Enter your choice (1-6):
|
```

```
Python 3.7.8 (tags/v3.7.8:4b47a5b6ba, Jun 28 2020, 08:53:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============================= RESTART: C:/Users/ramak/AppData/Local/Programs/Python/Python37/cal.py =============================

*** Simple Calculator with Advanced Features ***
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exponentiate
6. Quit
Enter your choice (1-6):
1
1
Enter the first number: 22
22.0
Enter the second number: 33
33.0
22.0 + 33.0 = 55.0

Invalid choice. Please enter a number between 1 and 6.

*** Simple Calculator with Advanced Features ***
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exponentiate
6. Quit
Enter your choice (1-6):
6
6
Exiting calculator...
>>> exited!!
```