Calculator Application

Features

1. **Dual Modes of Operation**
   - **Terminal Calculator:**
     - Interact via the command line.
     - Prompts users for two numbers and a choice of arithmetic operation.
     - Supports error handling for invalid inputs and division by zero.
   - **Dialog Box Calculator:**
     - Provides a GUI using the `tkinter` library.
     - Features a fully functional calculator interface with buttons for digits, operators, clear, and equals.
2. **Terminal Calculator**
   - **User Interaction:**
     - Asks for two numerical inputs from the user.
     - Offers a menu to select operations: addition, subtraction, multiplication, and division.
   - **Error Handling:**
     - Displays an error message for invalid inputs.
     - Handles division by zero gracefully by notifying the user.
   - **Operation Results:**
     - Computes and displays the result of the chosen operation.
3. **GUI Calculator**
   - **Modern Interface:**
     - Uses `tkinter` to build a calculator with buttons for digits (`0-9`), operators (`+`, `-`, `*`, `/`), clear (`C`), and equals (`=`).
   - **Dynamic Input Handling:**
     - Displays the current input or result dynamically in an entry widget.
   - **Error Notifications:**
     - Invalid expressions trigger a pop-up error message using `messagebox`.
   - **Responsive Layout:**
     - Dynamically adjusts the layout of buttons and entry widget to fit the window.
4. **Main Program Flow**
   - The user selects between terminal and GUI mode.
   - Depending on the selection, either the terminal or dialog box calculator is launched.

Important points

- **Error Handling:**
    - Handles invalid numerical inputs in the terminal calculator (**bold feature**).
    - Prevents division by zero and notifies the user.
    - Detects invalid expressions in the GUI calculator and displays an error dialog box (**bold feature**).
- **GUI Features:**
    - Fully functional calculator with an intuitive layout.
    - Responsive design ensures buttons and widgets adapt to the window size (**bold feature**).
- **Interactive User Experience:**
    - Offers a choice of operating mode at startup (**bold feature**).
    - Provides clear feedback and user-friendly error messages.

Description of the Code: Terminal and GUI Calculator Application

This repository contains a Python program that provides a **calculator application** with two modes: a **terminal-based calculator** and a **graphical user interface (GUI) calculator** using the `tkinter` library. The program allows users to perform basic arithmetic operations such as addition, subtraction, multiplication, and division.

Features

1. **Dual Modes of Operation**
    - **Terminal Calculator:**
        - Interact via the command line.
        - Prompts users for two numbers and a choice of arithmetic operation.
        - Supports error handling for invalid inputs and division by zero.
    - **Dialog Box Calculator:**
        - Provides a GUI using the `tkinter` library.
        - Features a fully functional calculator interface with buttons for digits, operators, clear, and equals.
2. **Terminal Calculator**
    - **User Interaction:**
        - Asks for two numerical inputs from the user.
        - Offers a menu to select operations: addition, subtraction, multiplication, and division.
    - **Error Handling:**
        - Displays an error message for invalid inputs.
        - Handles division by zero gracefully by notifying the user.
    - **Operation Results:**
        - Computes and displays the result of the chosen operation.
3. **GUI Calculator**
    - **Modern Interface:**

        - Uses `tkinter` to build a calculator with buttons for digits (`0-9`), operators (`+`, `-`, `*`, `/`), clear (`C`), and equals (`=`).
    - **Dynamic Input Handling:**
        - Displays the current input or result dynamically in an entry widget.
    - **Error Notifications:**
        - Invalid expressions trigger a pop-up error message using `messagebox`.
    - **Responsive Layout:**
        - Dynamically adjusts the layout of buttons and entry widget to fit the window.
4. **Main Program Flow**
    - The user selects between terminal and GUI mode.
    - Depending on the selection, either the terminal or dialog box calculator is launched.

Important Highlights

- **Error Handling:**
    - Handles invalid numerical inputs in the terminal calculator (**bold feature**).
    - Prevents division by zero and notifies the user.
    - Detects invalid expressions in the GUI calculator and displays an error dialog box (**bold feature**).
- **GUI Features:**
    - Fully functional calculator with an intuitive layout.
    - Responsive design ensures buttons and widgets adapt to the window size (**bold feature**).
- **Interactive User Experience:**
    - Offers a choice of operating mode at startup (**bold feature**).
    - Provides clear feedback and user-friendly error messages.

How to Run the Program

1. **Install Python** if not already installed.
2. Save the script to a file, e.g., `calculator.py`.
3. Run the script in a terminal:

```
bash
CopyEdit
python calculator.py
```

4. Choose between the terminal and GUI mode:
    - Enter `1` for terminal mode.
    - Enter `2` for GUI mode.

Prerequisites

- Python 3.x
- Tkinter (comes pre-installed with Python)

Future Improvements

- Add support for advanced mathematical operations (e.g., square root, exponentiation).
- Include memory functionality (e.g., store and recall values).
- Enhance the GUI with themes or additional styling.