

Goal: Create a program that converts Markdown files into HTML format. Allow users to input a .md file and get a downloadable .html file as output.

Structure:

- User Interface: Design a console-based user interface for input and output, allowing users to interact with the calculator using simple commands or GUI using libraries like Tkinter (optional).
- Functionality: Implement features to perform basic arithmetic operations, error handling for invalid inputs, and support for decimal or integer inputs.
- Reports: Generate reports or logs of performed operations (optional) and save them into a text file or a database.

Technologies:

- Python: For developing the application's core logic and features.
- Tkinter (Optional): For building a GUI-based calculator with buttons, input fields, and result displays.
- SQLite (Optional): For saving the history of operations performed by the user.



Optional:

- SQLite: For persistent data storage to save the user's history of calculations.
- Pandas (Optional): For storing the history in a table format and generating reports for operations performed.
- Logging: For generating logs of calculations performed and storing them for later reference.

Deliverables:

- Complete Python project files (including source code and necessary configuration files).
- A Python executable or script for running the application.
- Documentation of the application's functionality, usage, and setup.

Checklist:

- Design the system for arithmetic operations with error handling for invalid inputs (e.g., dividing by zero).
- Implement functionality for performing basic operations (addition, subtraction, multiplication, division).
- Optionally implement GUI using Tkinter for a better user experience.
- Optionally implement a history feature using SQLite or another method to store past operations.

- Test the application for various edge cases (e.g., non-numeric input, division by zero).
- Submit the completed project files and documentation.

