Pharmacy Management System — Final Project Proposal

Project Choice:
I'm planning to build a Pharmacy Management System in Python. The system will help manage medicine inventory, handle sales, and calculate bills for customers at a pharmacy.

How my project meets the requirements:

1. **Data structures and data representation:**

- I'll use Python dictionaries and lists to store information about medicines and sales. Each medicine will have details like name, quantity, price, and expiry date stored in a dictionary.

2. **Object-Oriented Programming:**

- The program will use classes like Medicine, Inventory, and Sales with constructors and methods to manage the pharmacy operations.

- I'll also include some inheritance, such as a base class for products and subclasses if needed.

3. **Loops, dictionaries, lists, and recursion:**

- Loops will help with going through the lists and dictionaries, like when showing inventory or calculating bills.

- Dictionaries and lists will be the main data structures.

- I plan to use recursion for a small task, such as searching through the medicine list or generating reports.

4. **Input validation:**

- The program will check user input to make sure values like quantity and price are positive numbers and dates are formatted correctly.

5. **Conditional statements:**

- If-else logic will be used throughout to check things like stock availability, validate input, and control how the program flows.

Extra features I might add:

- **File read/write:** I want to add the ability to save and load inventory data from files so the data isn't lost when the program closes.

- **GUI with Tkinter:** If I have enough time, I'll try to build a simple graphical interface to make it easier to use.

Plan to complete the project:

1. Create the classes and data structures for the medicines and inventory.

2. Build functions to add, update, and remove medicines from the inventory.

3. Develop the sales module to handle customer purchases and calculate bills.

4. Add input validation to catch invalid entries.

5. Implement file saving and loading for the inventory.

6. Use recursion for a feature like searching or reporting.

7. Test everything to make sure it works smoothly.

8. Prepare a flowchart and documentation.

Concerns:

- Figuring out a good way to use recursion might take some extra thought.

- Managing expiry dates and inventory with simple files could be a bit limited but should work for this project.

- Building a GUI depends on how much time I have left.

How this could be used in a real business:

- This system would be helpful for small pharmacies that want an easy way to track their stock and sales without paying for expensive software.

- It could also be improved over time to work for bigger businesses.