PROJECT TITLE : LIBRARY MANGEMENT SYSTEM .

Introduction :

The Library Management System is designed to simulate a library's operations, helping users keep track of books, manage borrowing activities, and view the inventory effectively. It uses the object-oriented programming approach and provides a graphical user interface for interaction.

Problem Domain :

The problem domain revolves around managing the inventory and usage of library resources efficiently. Libraries often face challenges like:

1. **Tracking Inventory**: Keeping accurate records of book availability.
2. **Borrowing and Returns**: Ensuring smooth operations for users borrowing or returning books.
3. **Author-Specific Queries**: Listing books written by specific authors.
4. **User Notifications**: Informing users about the availability or status of books.
5. **System Scalability**: Handling an increasing number of books or users without significant complexity.

This application addresses these challenges by digitizing core operations, reducing manual effort, and enhancing user experience.

Expected Outcomes :

1. Efficient management of book inventory.
2. User-friendly interface to interact with library resources.
3. Accurate tracking of borrowed books and their borrowers.
4. Clear visualization of books in a tabular format.
5. Simplified workflows for adding books, borrowing, and returning

Requirements :

Functional Requirements:

1. Add new books to the library.
2. Display all available books with details such as title, author, pages, price, and borrower status.
3. Filter books by author name.
4. Borrow books if available.
5. Return borrowed books.
6. Count the total number of books.

Methodology

The application employs an **object-oriented programming (OOP)** approach for modularity and scalability:

1. **Class Design**:
    - Library: Represents individual books with attributes like book_name, author, pages, price, and borrower.
    - LibraryApp: Manages the graphical interface and connects user interactions to underlying functionality.
2. **Graphical User Interface (GUI)**:
    - Tkinter is used to build the GUI, providing buttons, dialog boxes, and tables.
    - Features include:
        - Buttons for adding books, displaying books, and borrowing/returning books.
        - Dialog boxes for user inputs like book name or borrower name.
        - Treeview tables for visualizing book records.
3. **Data Storage**:
    - Books are stored as objects in a list, ensuring fast access and updates.
4. **User Interaction**:
    - Dialog boxes simplify input collection.
    - Message boxes provide feedback on actions.
5. **Event Handling**:
    - Button commands trigger specific functions like adding a book or displaying the book list.

Data structures used :

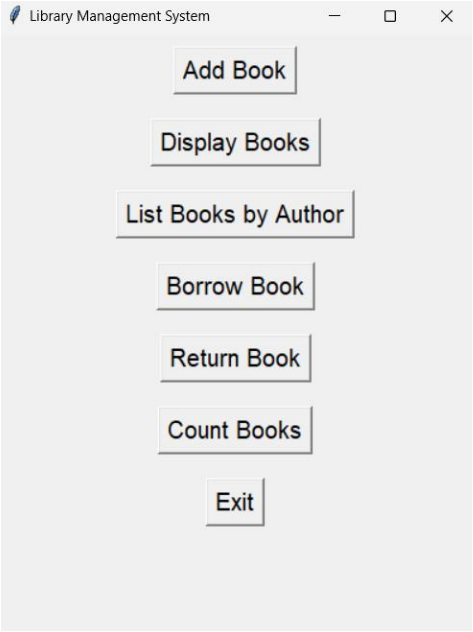
1. List
2. Searching
3. Self.lib

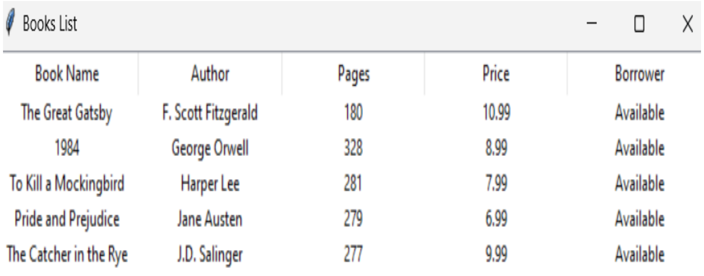
Conclusion :

This Library Management System successfully addresses the problem domain, offering a reliable and interactive solution for small-scale library management. It demonstrates the practical application of object-oriented programming and GUI development, making it a valuable tool for library staff and a learning project for developers. Future enhancements could include:

- Persistent data storage using databases or files.
- Multi-user access with different roles (e.g., librarian, borrower).
- Advanced search filters based on price, pages, or other attributes.
- Integration with online libraries for digital book access.

Output :





Book Name	Author	Pages	Price	Borrower
The Great Gatsby	F. Scott Fitzgerald	180	10.99	Available
1984	George Orwell	328	8.99	Available
To Kill a Mockingbird	Harper Lee	281	7.99	Available
Pride and Prejudice	Jane Austen	279	6.99	Available
The Catcher in the Rye	J.D. Salinger	277	9.99	Available