Library Management System

Project Title: Library Management System
Developed By: Yash Bajpai

Branch: Computer Science Engineering
Semester: 6th

Objective

To develop a simple Library Management System that enables a librarian to add, borrow, and return books while keeping track of availability and return dates using Python programming.

Tools and Technologies Used

Tool/Technology	**Description**
Python	Programming Language
datetime module	To calculate return dates
File I/O (optional)	For data storage (not used here)
VS Code / PyCharm	IDE for development

Features of the System

- Add Book: Allows the librarian to add new books with name, author, and number of copies.
- View Books: Displays a list of all available books and how many copies are left.
- Borrow Book: Lets a user borrow a book. A return date (14 days from issue) is shown.
- Return Book: Updates the system when a book is returned.
- Validation: Handles invalid input like borrowing when no copies are available or returning unborrowed books.

Test Cases

Test Case No.	Description	Input	Expected Output	Status
TC1	Add a new book	Book: "Python 101", Copies: 5	Book added successfully	☑
TC2	View available books	-	Display list with all book details	☑
TC3	Borrow a book with copies	Select book with available copy	Shows return date (14 days from today)	☑
TC4	Borrow book with 0 copies	Book already borrowed by others	Error: All copies are borrowed	☑
TC5	Return a borrowed book	Return previously borrowed book	Confirmation message	☑
TC6	Return book not borrowed	Try returning un-borrowed book	Error: No copies borrowed	☑
TC7	Exit system	Option 5	Program exits cleanly	☑

Sample Output & Code

Below is the Python code for the Library Management System:

```python
import datetime


library = []

def add_book():
    book_name = input("Enter Book Name: ")
    author = input("Enter Author Name: ")
    total_copies = int(input("Enter Number of Copies: "))
    library.append({
        'book_name': book_name,
        'author': author,
        'copies': total_copies,
        'borrowed': 0
    })
    print(f"Book '{book_name}' added successfully.\n")

def view_books():
    if not library:
        print("No books in the library.\n")
        return
    print("\nAvailable Books:")
    for idx, book in enumerate(library, 1):
        print(f"{idx}. {book['book_name']} by {book['author']} - 
Available: {book['copies'] - book['borrowed']} copies")

def borrow_book():
    view_books()
    choice = int(input("\nEnter the number of the book you want to borrow: 
")) - 1
    if 0 <= choice < len(library):
        book = library[choice]
        if book['borrowed'] < book['copies']:
            book['borrowed'] += 1
            return_date = datetime.date.today() + 
datetime.timedelta(days=14)
            print(f"You borrowed '{book['book_name']}'. Please return by 
{return_date}.\n")
        else:
            print("Sorry, all copies are currently borrowed.\n")
    else:
        print("Invalid choice.\n")

def return_book():
    view_books()
    choice = int(input("\nEnter the number of the book you want to return: 
")) - 1
    if 0 <= choice < len(library):
        book = library[choice]
        if book['borrowed'] > 0:
            book['borrowed'] -= 1
            print(f"Thank you for returning '{book['book_name']}'.\n")
        else:
            print("No copies of this book are borrowed.\n")
    else:
        print("Invalid choice.\n")

def main():
    while True:
        print("==== Library Management System ====")
        print("1. Add Book")
        print("2. View Books")
        print("3. Borrow Book")
        print("4. Return Book")
        print("5. Exit")
        choice = input("Enter your choice: ")
        if choice == '1':
            add_book()
        elif choice == '2':
            view_books()
        elif choice == '3':
            borrow_book()
        elif choice == '4':
            return_book()
        elif choice == '5':
            break
        else:
            print("Invalid choice. Try again.\n")

if __name__ == "__main__":
    main()
```

Sample Outputs (Terminal)

```
==== Library Management System ====

1. Add Book

2. View Books

3. Borrow Book

4. Return Book

5. Exit

Enter your choice: 1

Enter Book Name: DBMS

Enter Author Name: Yashbjp

Enter Number of Copies: 10

Book 'DBMS' added successfully.
```

```
==== Library Management System ====
1. Add Book
2. View Books
3. Borrow Book
4. Return Book
5. Exit
Enter your choice: 2
Available Books:
1. DBMS by Yashbjp - Available: 10 copies
==== Library Management System ====
1. Add Book
2. View Books
3. Borrow Book
4. Return Book
5. Exit
Enter your choice: 3
Available Books:
1. DBMS by Yashbjp - Available: 10 copies
Enter the number of the book you want to borrow: 1
You borrowed 'DBMS'. Please return by 2025-04-06.
==== Library Management System ====
1. Add Book
2. View Books
3. Borrow Book
4. Return Book
5. Exit
```

```
Enter your choice: 4
Available Books:
1. DBMS by Yashbjp - Available: 9 copies
Enter the number of the book you want to return: 1
Thank you for returning 'DBMS'.
==== Library Management System ====
1. Add Book
2. View Books
3. Borrow Book
4. Return Book
5. Exit
Enter your choice: 5
=== Code Execution Successful ===
```

Conclusion

This project successfully demonstrates the implementation of a basic Library Management System using Python. It allows for easy management of books and introduces core programming concepts such as lists, functions, loops, conditionals, and date handling.

Future Enhancements

- Implement GUI using Tkinter or PyQt
- Add user login system (admin/student)
- Database integration (e.g., SQLite or MySQL)
- Email notifications for return reminders