

Redesigning Poor Code

Your task is to reformat a poorly designed code for a Library Management System with the following functionalities

1. Manage books (add, update, delete, list, and search by various attributes like title, author, or ISBN)
2. Manage users (add, update, delete, list, and search by attributes like name, user ID)
3. Check out and check-in books
4. Track book availability
5. Simple logging of operations

Task for the Candidate

1. The application should utilize classes and objects more effectively, with clear relationships and responsibilities among them with Object Oriented Design.
2. Implement or refactor the storage.py for reliable storage and retrieval using file-based storage (JSON, CSV, etc.)
3. Obtain input from the user using CLI to access information from the user in a friendly and intuitive manner.
4. Implement error handling and input validation throughout the application.
5. The application's design doesn't facilitate easy extension or modification.
6. Ensure that the application is modular and scalable, allowing for future expansions such as new types of items to manage or additional features like due dates for books, late fees, etc.
7. Include documentation and comments to explain the design decisions, architecture, and usage of classes and methods.

Code

You can find the code to be worked on : [here](#)

```
.  
├── book.py  
├── check.py  
├── main.py  
├── models.py  
├── storage.py  
└── user.py
```

