Contributors

- Izzat Ikram
- Ayman Sohail
- Gehad Mohamed

Library Management System - Readme File

Overview:

The Library Management System is a Python application for managing library book borrowing and return operations. The system allows users to view and borrow and return available books. It also includes the ability to calculate fines for overdue books. Additionally, it allows the librarian to add and remove new books from the system. Data is stored in CSV files for easy access and management.

To run the program, you can execute the script using the python Library Management System.py command. After running the program, a menu will appear, allowing the user to choose from several options for managing books.

Requirements:

- Python version 3.6 or later.

- Required libraries:
  - csv: For handling CSV files.
  - datetime: For managing time and dates.

How to use

Once the user launches the program, you will be presented with an interactive list of options. Here's what the system offers:

1. View Available Books: View all available books in the library along with the status of each book (available or borrowed).
2. Borrow a Book: Select an available book to borrow.
3. Return a Book: Return a borrowed book and calculate late fees.

4. View Borrowed Books: View the books you've borrowed.
5. Add Book (for Librarian): Add a new book to the system.
6. Remove Book (for Librarian): Remove a book from the system.
7. Exit: Exit the program.

The program allows the user to choose the service by entering the number associated with the service.

Code Structure

Basic Classes

1. **UserAccount:**

For managing accounts and book-borrowing operations. It contains:

- borrow_book: Borrow a book.
- return_book: Return a book.

2. **Librarian**:

A class inherited from UserAccount, responsible for adding and removing books.

- add_book: Adds a new book to the system.
- remove_book: Removes a book from the system.

3. **ManageBookLending**:

For managing book borrowing operations.

- load_books: Loads books from a CSV file.
- save_books: Saves the book status to a CSV file.
- lend_book: Borrows a book to a user.

4. **ReturnsAndOverduePenalties:**

Calculates late fines for returning books.

- calculate_penalty: Calculates the penalty based on the number of days overdue.
- return_book: Returns the book and calculates the penalty, if any.

Example of use

1. Start the system: Run the program by executing the script file.
2. Display the main menu:

The system will display a list of available options as follows:

```
** Library Management System**
    1. View Available Books
    2. Borrow a Book
    3. Return a Book
    4. View Borrowed Books
    5. Add Book (for Librarian)
    6. Remove Book (for Librarian)
    7. Exit
Enter your choice:
```

3. Book Borrowing Steps :

- Press 2 to select "Borrow a Book."
    - The system will prompt the user to enter the title of the book they wish to borrow by displaying the following message: “Enter the book title to borrow” for example, "War and Peace."

    - Then, the program will display a message saying whether the book is available or not
        - ➔ If the book is not available , the program will display the message "Error: Book is not available." and Display the main menu
        - ➔ If the book is available, the system will confirm the user’s borrowing of the book and display a confirmation message: "You borrowed 'War and Peace.' "'

4. Book Return Steps:

- Press 3 to select Return Book.
    - The system will prompt the user to enter the title of the book they wish to return by displaying the following message: “Enter the book title to return.”
    - Enter the title of the book, for example: "War and Peace."

  - Then, the system will confirm the user's returning of the book and display a confirmation message: "Book 'War and Peace' returned successfully. '"

- Return Confirmation and Calculation of penalties :
  - After entering, the system will process the return and calculate any late fines if the loan period has been exceeded.
  - The system will display a message regarding the fine, if applicable, or the return confirmation: "'War and Peace' successfully returned."