Java Sprint 1 (SD11)

Library Management System Documentation

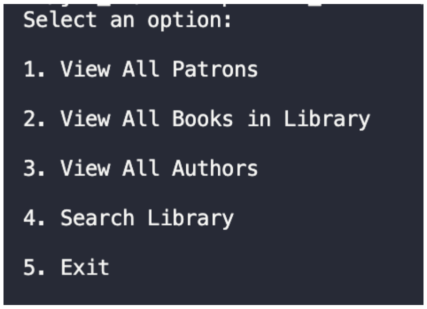
Student: Chelsea Slade (Mayne)
Due: October 31, 2024

Table of Contents

User Documentation

1.1 Library Management System Explanation

The Library Management System program was designed for use by a librarian to handle basic library tasks such as item inventory details (books and periodicals), author and patron details, and a borrowing/return system for patrons. The code is menu-based through running the LibManagementDemo.java file, and a user simply selects which option they would like to select and follows input instructions as displayed in the terminal.



```
Select an option:

1. View All Patrons

2. View All Books in Library

3. View All Authors

4. Search Library

5. Exit
```

Menu Options:

1. *View All Patrons:* This option immediately displays all patrons in the system with their basic details and an index number. The user is prompted with a further menu with patron-based options as follows:

    i. View Books Borrowed
    ii. Add a New Patron
    iii. Edit a Patron
    iv. Delete a Patron
    v. Return to Main Menu

    To view books borrowed, edit, or delete, the user will type the index number of the patron they wish to handle as shown in the list displayed.

2. *View All Books in Library*: This option displays all books currently in the system with a summary of details, and an index number. The user is given the option to add a new book, add new periodical, or return to the main menu. On adding a new item, the user will enter all relevant information for that type of item.

3. *View All Authors*: This option displays all authors currently in the system with a summary of their details, and an index number. The user selects from a menu related to author handling as follows:

    i. View books written by author

ii. Add a New Author
iii. Edit an Author
iv. Delete an Author
v. Return to Main Menu

To view books written, edit, or delete an existing author, the user will be prompted to enter the index number associated with that author as shown in the summary list. When adding a new author, the user will be prompted to add all relevant details.

4. *Search Library:* The user will be prompted to enter a search criterion. This can be any attribute related to the item they are interested in, such as title, author, ID, etc. The search is not case-sensitive. Upon returning the item(s), the user can select one of the options by index (if multiple results), and will be given the option to borrow, edit or delete the item. If the user chooses to borrow the item, they will be prompted to select a patron who is borrowing the item.

1.2 Program Classes

LibraryItem (Class Hierarchy)
The LibraryItem class handles the creation and handling of library item objects and is broken down into the subclasses “Book” and “Periodical”. All library items have attributes for ID, title, author, ISBN, publisher and number of copies available. The periodical subclass has additional attributes for availability in print or electronic form, and book has all the above plus availability in audio form.

Methods are available for library items to add a new item, edit an existing item, convert the item details to a summary form, and a get method for each of the item’s attributes.

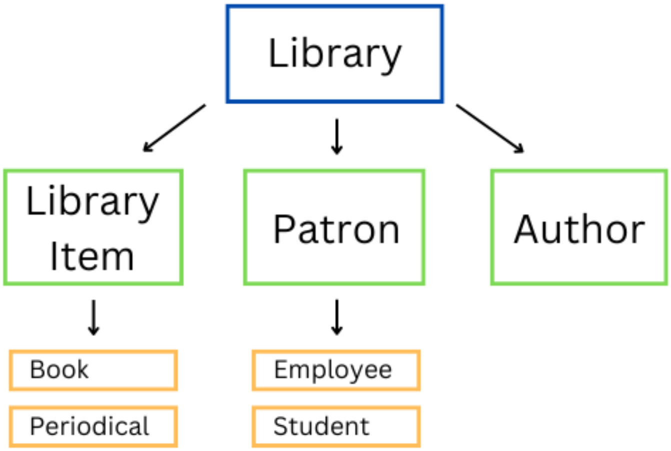
Library
The library class is mainly for the handling of arrays (storage) of other objects, and the display or editing of those arrays. The attributes for Library are separate array lists for library items, authors, and patrons. This allows the program to add or remove an individual object (library item, author, or patron) from the system. It also provides the “search” method so a user can browse items in the library based on their chosen criteria.

Author
The author class handles creating and editing of authors, as well as storage of the books each has written in an array. Methods for the author class include adding a written book, getting books written by a given author, displaying an author in string summary form, adding a new author, and editing details of an existing author.

Patron (Class Hierarchy)

Patron is the class that handles employees and students of the library. Subclasses are available for student and employee to specify their patron status. Attributes of patrons include name, address, phone number, and an array of books that they currently have checked out. Methods for the patron class include a get method for each attribute, adding a new patron, editing an existing patron, borrowing an item, returning an item, and displaying a list of borrowed items by a given patron.

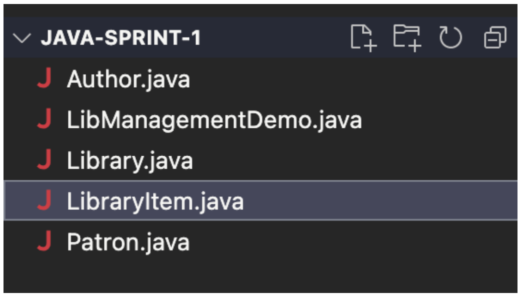
1.3 Class Diagram



Development Documentation

2.1 Source Code Description

The source code for the Library Management System is separated into files for each class (LibraryItem, Library, Patron and Author) and a file called LibManagementDemo for the user to run the library menu.



2.2 Development Standards

- Version control through GitHub with detailed commit messages and branching.
- Classes named according to PascalCase.
- Methods and variables named according to camelCase.
- Code clearly commented throughout for readability.

2.3 Database Theory

In theory, a database would be more useful for storage of information that is presently being stored in arrays. This would allow data to be persistent between uses of the program, rather than reset each time it is run. A table would be created in the database for authors, patrons, library items, books written (by author) and borrowed items.

2.4 Entity Relationships

Main Tables:

PK = Primary Key
FK = Foreign Key

Authors
authorID (PK)
authorLastName
authorFirstName
authorDOB

Patron
patronID (PK)
patronLastName
patronFirstName
patronAddress
patronPhoneNum
patronType

Library Items
itemID (PK)
ISBN
itemTitle
itemPublisher
authorID (FK)
numCopies
itemType
isPrinted
isElectronic
isAudio

Borrowed Items
itemID (FK)
patronID (FK)
borrowDate
returnDate

Books Written
itemID (FK)
patronID (FK)

Deployment Documentation

3.1 Installation Details

Source code for the program can be found at: https://github.com/chelseaslade/Java-Sprint-1/tree/main

To download the code, select the green “code” button, and then copy the link under HTTPS (the repository URL). Enter your chosen IDE (ex. Visual Studio Code), open the terminal, and enter:

```
<git clone *insert URL here*>
```

This will clone the repository to your local system and allow you to run and edit the code.

Detailed instructions to clone from GitHub located here: https://docs.github.com/en/repositories/creating-and-managing-repositories/cloning-a-repository

Once the files are accessible in the IDE, navigate to the LibManagementDemo.java file and run it. This will display the main menu of the program in the IDE terminal and prompt the user to make inputs to choose various options.