

# Advanced Programming Assignment

## Library Automation System

Contributors: Swapnil Panigrahi

## What is the Project?

The project is a library automation system, which can be accessed by the librarian and the member alike. The librarian has the freedom to add members, remove them, add books, and remove them. Additionally, he can view the member details, such as the books in their possession and current fine as well. The students have the ability to issue books, return them, and pay their fines.

## 1 Classes and Methods

### 1.1 Book Class

- **Properties:** Title, Author, Book ID, Availability, etc.
- **Methods:** Tracking book details, managing availability, and providing information.

### 1.2 Student Class

- **Attributes:** Student ID, Name, Issued Books, etc.
- **Methods:** Borrowing and returning books, checking account status, updating fine, etc.

### 1.3 Librarian Class

- **Attributes:** Library which he has been assigned to
- **Methods:** Managing book inventory, handling student requests, registering students

### 1.4 Library Class

- **Attributes** List of Students, List of Books
- **Methods:** Oversees all the methods, checks if the inputs are valid, calls getter methods across all classes to utilize their properties

## 2 Relationships

### 2.1 Composition

The Library class is composed of an array of Book objects and an array of Student objects. For example, the issue status of Book object can be updated by the library, and it also manages the number of copies of the Book object.

```
1 for(Book i : this.BookList){
2     if (i.getID()==bookID && i.isIssued()){
3         temp = i;
4         success = true;
5     }
6 }
```

Listing 1: Snippet from the Library Class showing it accessing the Book's ID getter method and Issue Status getter method

```

1  for (Book i : this.BookList){
2      if (Objects.equals(i.getAuthor(), author) && Objects.equals(i.getTitle(), title)){
3          i.increaseCopies();
4          copies++;
5      }
6  }

```

Listing 2: Snippet from the Library Class showing it setting the copies of the Book object

## 2.2 Association

The Librarian class has an association with the Library class, representing that they manage books and students from the library. For example, the Librarian class can only access the Library class properties using only its getter methods.

```

1  public void add_book(){
2      System.out.print("Title: ");
3      String title = input.nextLine();
4      System.out.print("Author: ");
5      String author = input.nextLine();
6
7      this.Lib.add_book(title,author);
8  }

```

Listing 3: Snippet from the Librarian Class showing it accessing the Library Class using the getter methods

## 2.3 Dependency

```

1  public Book return_book(Student student, int bookID){
2      for (Book i : student.getIssued()){
3          if (i.getID()==bookID){
4              System.out.println("The book with ID: "+bookID+" has been returned.");
5              i.returnBook();
6              return i;
7          }
8      }
9      return null;
10 }

```

Listing 4: Snippet from the Librarian Class showing it only passing the Student as parameter without initialising it

## 3 Basic Operations

- **Adding Books:** The Librarian can add books to the Library by taking input of the title and the author and passing it to the Library object method of adding a new Book object.
- **Borrowing and Returning Books:** Student can borrow books by selecting available books from the Library. The Book's availability status is updated. Students return books, which are then marked as available.
- **Searching for Books:** Student can search for books by ID. Librarian can search for books and provide information to students.
- **Managing Library Users:** Librarian can manage Student account and tracking issued Books. Librarian can penalise the Student for returning Book late. The Student can clear their dues by logging into their account.

## How to run the project

1. Unzip the folder in any directory of choice or clone the project from the [Github Repository](#)
2. Open the terminal at `..\AP-Assignment-1`
3. Run the following commands in order:

```
mvn clean
mvn compile
mvn package
cd target
java -jar AP-Assignment-1-1.0-SNAPSHOT
```
4. To re-run it, you can simply enter `java -jar AP-Assignment-1-1.0-SNAPSHOT` if you are in the `..\AP-Assignment-1\target` directory