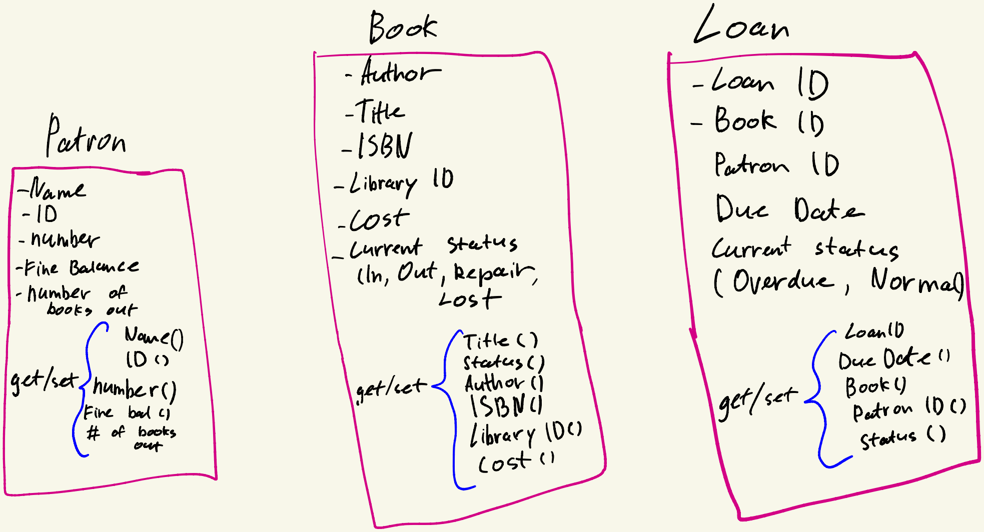
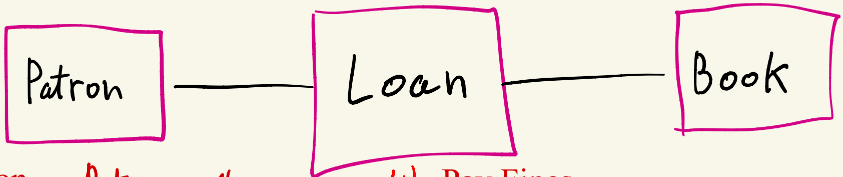
Taras Glushko
Homework 2
CSCE 1040.001
Design Diagram





1) Add Patron

Patron class 1-4

Prompt user for Patron Name
Prompt user for Patron ID
Create a Patron object
Set Patron Name
Set Patron ID
Set Fine Balance to 0
Set Current Number of Books Out to 0
Add Patron object to the Patron collection

2) Edit Patron

Prompt user for Patron ID
Find the Patron object using the Patron ID
Prompt user for new Name (optional)
Update Patron Name (if provided)
Prompt user for new ID (optional)
Update Patron ID (if provided)

3) Delete Patron

Prompt user for Patron ID
Find the Patron object using the Patron ID
Remove the Patron object from the collection

4) Pay Fines

Prompt user for Patron ID
Find the Patron object using the Patron ID
Prompt user for fine payment amount
Subtract the payment from the Patron's Fine Balance

5) Add Book

Book class (5-7)

Prompt user for Book Title
Prompt user for Author
Prompt user for ISBN
Prompt user for Library ID
Prompt user for Cost
Set Book Status to 'In'
Create a Book object
Set Book attributes (Title, Author, ISBN, etc.)
Add Book object to the Book collection

Edit Book

6) Prompt user for Book Library ID
Find the Book object using the Library ID
Prompt user for updated information (Title, Author, etc.)
Update Book attributes (Title, Author, ISBN, etc.)

7) Delete Book

Prompt user for Book Library ID
Find the Book object using the Library ID
Remove the Book object from the collection

Loan class (8- 12)

8) Check Out a Book

Prompt user for Patron ID
Find Patron using Patron ID
Check if Patron has no overdue books and books out <= 6
Prompt user for Book ID
Find Book using Book ID
Check if Book is available (status is 'In')
Create Loan object
Set Loan ID, Book ID, Patron ID, Due Date
Update Book Status to 'Out'
Increase Patron's number of books out by 1
Add Loan to Loan collection

9) Check In a Book

Prompt user for Loan ID
Find Loan object using Loan ID
Check if Book is overdue, calculate fines (if any)
Update Patron's fine balance (if needed)
Update Book status to 'In'
Decrease Patron's number of books out by 1
Remove Loan object from Loan collection

10) Re-Check a Book

Prompt user for Loan ID
Find Loan object using Loan ID
Check if re-check is allowed (1 re-check per loan)
Update Due Date (extend by 10 days)

11) List Overdue Books

```
For each Loan in the Loan collection:
    If Loan is overdue:
        Print Loan details (Book ID, Patron ID, Due Date)
        Find Patron by Patron ID
        Print Patron details (Name, Fine Balance)
```

12) Report a Book as Lost

Prompt user for Loan ID
Find Loan object using Loan ID
Update Book status to 'Lost'
Find Book cost using Book ID
Update Patron's fine balance (add Book cost)
Remove Loan object from Loan collection

Additional Requirements Verification:

-Before borrowing a book, verify the Patron's status:
-No overdue books
-Books out <= 6 including the new borrow
-Fines: Ensure fines are calculated properly upon book return.
-Lost Books: Ensure lost books are reported correctly, with the book cost being added to the Patron's fine balance.

Report:

The design of the library management system was based on establishing relationships between the Patron. The greatest challenge, how to handle loans and borrowing limits, overdue statuses, and fines. I really paid close attention to the details to make sure that, after an action like borrowing or returning was performed, appropriate updates would happen in the status of both the Patron and the Book.

A part of it included the implementation of fine tracking to make sure that all lost books were duly tracked and costs charged to the Patron. The use of the collection for each class made the addition, editing, deletion, and searching for records quite easy.

In general, this design follows clear and flexible structuring while providing further extensibility with ease of data integrity and usability.