Your name

[Email address]

**Assignment 1**

* 1. **Part 1: Design**

**Class: LibraryItem (Abstract Base Class)**

**• Attributes:**

Id (string): Unique id for each item.

Title (string): Item name.

IsAvailable (bool): Indicates if the item is available for borrowing.

BorrowedDate (DateTime?): Date when the item was borrowed.

BorrowedBy (string): ID of user who borrowed the item.

**• Methods:**

GetItemType() (abstract): Returns the type of item (e.g., Book, Article).

CalculatePenalty(int daysOverdue) (virtual): Calculates the penalty for overdue items.

ToString() (override): Returns a string representation of the item.

**• Reasoning:** This abstract class serves as the foundation for all loanable materials (books, articles, digital media). It establishes common properties and behavior, allowing for inheritance and polymorphism.

**Class: Book (extends LibraryItem)**

**• Attributes:**

Author (string): Author of the book.

ISBN (string): International Standard Book Number.

**• Methods:**

GetItemType() (override): Returns "Book".

CalculatePenalty(int daysOverdue) (override): Special penaltycalculation ($1/day in your code, changeable to $5/day).

ToString() (override): Includes book-specific details.

**• Reasoning:** Books are a special type of library item, requiring special attributes like author and ISBN, and a custom penalty rate as per requirements.

**Class: User (Abstract Base Class)**

* **Attributes:**

Id (string): Unique identifier for the user.

Name (string): User’s name.

BorrowedItems (List<LibraryItem>): List of items currently borrowed.

* **Methods:**

GetMaxBorrowDays() (abstract): Returns maximum borrowing period.

GetUserType() (abstract): Returns user type (e.g., Student, Staff).

CalculateTotalPenalty(): Calculates total penalty for overdue items.

ToString() (override): Provides user details.

* **Reasoning:** This base class handles common user functionality, allowing differentiation between students and staff via inheritance.

**Class: Student (Derived from User)**

* **Attributes:**

Department (string): Student’s department.

* **Methods:**

GetMaxBorrowDays() (override): Returns 14 days (needs adjustment to trimester length).

GetUserType() (override): Returns "Student".

ToString() (override): Includes department**.**

* **Reasoning:** Students have specific borrowing limits (5 items per trimester, 1 renewal) and a $5/day penalty, necessitating a distinct class.

**Class: Staff (Derived from User)**

* **Attributes:**

Position (string): Staff member’s job title.

* **Methods:**

GetMaxBorrowDays() (override): Returns 30 days (needs adjustment to 365 days).

GetUserType() (override): Returns "Staff".

ToString() (override): Includes position.

* **Reasoning:** Staff have unlimited borrowings and renewals for a calendar year, requiring a separate class.

**Class: Library**

* **Attributes:**

\_items (List<LibraryItem>): Collection of all library items.

\_users (List<User>): Collection of all users.

* **Methods:**

AddItem(LibraryItem): Adds an item to the library.

BorrowItem(string userId, string itemId): Manages borrowing.

ReturnItem(string userId, string itemId): Manages returns.

CalculateUserPenalty(string userId): Calculates penalties.

GetUserBorrowings(string userId): Retrieves borrowed items.

* **Reasoning**: This class manages the entire system, coordinating interactions between users and items

|  |  |
| --- | --- |
| **LibraryItem** |  |
| Responsibilities | Store item details (ID, title, availability). |
|  | Track borrowing status. |
|  | Calculate penalties (base rate). |
|  | Process borrowing, return, and renewal activities. |
| Collaborators | User (borrows/returns items) |
|  | Derived classes (Book, Article, DigitalMedia) |

|  |  |
| --- | --- |
| **Book** |  |
| Responsibilities | Store book-specific details (author, ISBN). |
|  | Provide item type as "Book". |
|  | Calculate book-specific penalties. |
| Collaborators | LibraryItem (parent) |
|  | Library (managed by) |

|  |  |
| --- | --- |
| **Student** |  |
| Responsibilities | Store department info. |
|  | Limit borrowing to 5 items and 1 renewal. |
|  | Enforce $5/day penalty. |
| Collaborators | User (parent) |
|  | Library (interacts with) |

|  |  |
| --- | --- |
| **User** |  |
| Responsibilities | Store user details (ID, name). |
|  | Manage list of borrowed items. |
|  | Calculate total penalties. |
| Collaborators | LibraryItem (borrows) |
|  | Derived classes (Student, Staff) |

|  |  |
| --- | --- |
| **Staff** |  |
| Responsibilities | Borrow any number of items. |
|  | Renew items any number of times. |
|  | Enforce $5/day penalty. |
| Collaborators | LibraryItem (To borrow/return items) |
|  | BorrowingRecord (To keep borrowing history) |

|  |  |
| --- | --- |
| **Library** |  |
| Responsibilities | Manage collections of items and users. |
|  | Handle borrowing/returning operations. |
|  | Handle borrowing/returning operations. |
| Collaborators | LibraryItem (manages) |
|  | User (manages) |

