KING SAUD UNIVERSITY COLLEGE OF COMPUTER AND INFORMATION SCIENCES Computer Science Department	
CSC 111 **Computer Programming - I**	**Project** Phase#2 (**Due date: 20-Nov 2025**)

Library Simulation

Description:

In the previous phase, you may have seen that your code contains many variables, some of which are interconnected. To elevate the quality of your programs in this phase, you'll be leveraging object-oriented concepts. A crucial move in this direction is the introduction of a new class called "Member". This class is tailored to encapsulate various aspects of personal library account within your library and is governed by the following UML specification:

Member
- id : int - name: String - borrowedCount: int - numViewBorrowed: int - numBorrows: int - numReturns: int - sessionFees: double + TotalRevenue: double + TotalViewBorrowed: int + TotalBorrows: int + TotalReturns: int
+ Member(id:int, name:String, borrowedCount:int) - canBorrow(): boolean - canReturn(): boolean + viewBorrowedCount(): void + borrowOne(): boolean + returnOne(): boolean + displayStatistics(): void + reset(): void + Setters & getters (as needed)

KING SAUD UNIVERSITY COLLEGE OF COMPUTER AND INFORMATION SCIENCES Computer Science Department	
CSC 111 **Computer Programming - I**	**Project** Phase#2 (**Due date: 20-Nov 2025**)

Attributes:

- **id**: Is the member unique identification number.

- **name**: Is the account holder name of the item.

- **borrowedCount**: Is the number of books currently borrowed.

- **numViewBorrowed**: Is the number of times the operation view borrowed books count was used.

- **numBorrows**: Is the number of times the operation borrow book was used.

- **numReturns**: Is the number of times the operation return book was used.

- **sessionFees**: Is the total amount of fees that incurred during the session.

+ **TotalRevenue**: Accumulates the total library profit generated from charging all accounts for using borrow operations.

+ **TotalViewBorrowed**: Accumulates the number of times the view borrowed books count was used.

+ **TotalBorrows**: Accumulates the number of times the borrow book operation was used.

+ **TotalReturns**: Accumulates the number of times the return book operation was used.

Methods:

+ **Member(id:int, name:String, borrowedCount:int)**: A constructor.

- canBorrow(): Checks if the member can borrow more books (not exceeding 5).

- canReturn(): Checks if the member can return a book (has at least 1 borrowed book).

+ viewBorrowedCount(): Prints the member's current borrowed book count.

+ borrowOne(): Simulates borrowing a book. A fee of 0.50 credit is charged and added to TotalRevenue.

+ returnOne(): Simulates returning a book. No fee is charged.

+ displayStatistics(): Prints the member's session statistics.

+ reset(): Resets the statistics to zero.

Update your program to leverage this class effectively.

KING SAUD UNIVERSITY COLLEGE OF COMPUTER AND INFORMATION SCIENCES Computer Science Department	
CSC 111 **Computer Programming - I**	**Project** Phase#2 (**Due date: 20-Nov 2025**)

Submission Rules:

- Submit your solution as `Member.java & LibrarySimulator.java` on LMS
- The submitted Java files must include the names and ID numbers of all group members at the top as a comment
- Only one student of each group needs to submit the project on behalf of the whole group
- Code must compile and run without errors
- Code style:

  - Use clear variable names

  - Indent properly

  - Add short comments for key parts

Bonus [1 point]
A bonus of 1 point (limited to the 15 points allocated for this project) will be awarded for the use of **GitHub** during project development.

To qualify:

- The group must create and maintain a GitHub repository for the project
- Each member must show active participation through commits, branches, or pull requests
- Effective use of Git tools will be considered in the evaluation

Add the link to your repository at the top of submitted Java files as a comment.