Palestine Polytechnic University
College Of Information Technology and Computer Engineering

Object-Oriented Programming Project

Project Name : **Library Registration System**

Instructor: Mohammed Jabari

Done by: Deena Seadahmad , (217032)
Mohammed Mummar , (211028)

Dec 17 , 2023

➔ **Description :**

In designing a Library Registration System, we will employ object-oriented design concepts to model the main objects and relationships. The system will encapsulate entities such as Books, Magazines, Journals, Authors, Students, ensuring a flexible and scalable representation of the library's functionality.

➔ **System Overview**

Describe the main components (classes) and their roles in the system.

1. Book Class (implements Display)

Attributes: Title, Author, Number, Genre, Version, Date., onloan , DueDate , LoanDate.

Methods: Getters and setters for attributes.

getInfo(): Returns formatted information about the book.

inLoan(): Indicates whether the book is currently on loan.

2. Magazine Class (inherits from Book)

Additional Attributes: Issue Number, Release Date.

Methods: Getters and setters for attributes.

getInfo(): Returns formatted information about the Magazine.

inLoan(): Indicates whether the Magazine is currently on loan.

3. Journal (inherits from Book):

Additional Attributes: Conference Name, Conference Number.

Methods: Getters and setters for attributes.

getInfo(): Returns formatted information about the Journals.

inLoan(): Indicates whether the Journals is currently on loan.

4. Author Class

Attributes: Id, Name, Address, Birth Date.

Methods: Getters and setters for attributes.

getInfo(): Returns formatted information about the Author.

5. Student Class (inherits from Author):

Additional Attributes:, Major , loanedBooks

Methods: Getters and setters for attributes.

addLoanedBook(Book book): Adds a book to the list of books loaned by the student.

removeLoanedBook(Book book): Removes a book from the list of books loaned by the student.

getInfo(): Overrides the getInfo() method from the Author class to include information about the major.

6. Library Class

Attributes: student , books

Methods:

addBook(Book book): Adds a book to the library.

addStudent(Student student): Adds a student to the library.

searchByTitle(String title): Searches for a book by title.

searchByNumber(Integer number): Searches for a book by number.

searchByAuthor(String authorName): Searches for a book by author.

searchStudentByNumber(Integer number): Searches for a student by number.

checkBookAvailable(): Checks the availability of books in the library.

showBooks(): Displays information about books in the library.

loanBook(int studentNumber, int bookNumber): Facilitates the loaning of a book to a student.

returnBook(int studentNumber, int bookNumber): Facilitates the return of a book by a student.

calculateDueDate(LocalDateTime loanDateTime): Calculates the due date for a loaned book.

checkLoan(Book book): Checks the status of a loan and provides feedback to the user.

7. Date Class

Attributes: Day , Month , Year

Methods: Getters and setters for attributes.

getInfo(): Returns formatted information about the Date.

8. Display Interface

Methods : getInfo() , inLoan()

9. Library Management System - Main Program

The main program of the Library Management System (LibrarySys) acts as the user interface for interacting with the library functionalities. The program provides a menu-driven interface allowing users to perform various actions related to adding books, adding students, searching for books, managing loans, and more.

Features and Functionality

Menu-Driven Interface:

The program displays a menu with various options, such as adding a book, adding a student, searching for books, checking books on loan, and more.

Exception Handling:

The program includes exception handling to address input mismatches. If the user enters invalid input, the program prompts the user to enter a valid option.

Library Operations:

Users can interact with the library by adding books and students, searching for books by title, number, or author, checking books on loan, showing all books, loaning books to students, and returning books.

Continuous Operation:

The program operates in a continuous loop, allowing users to perform multiple operations without restarting the program.

GUI User-Friendly Interface for the System:

To enhance the user experience of the library management system, a user-friendly graphical interface has been implemented. The GUI features intuitive forms for entering and searching for books, providing a seamless interaction for users. The design prioritizes clarity and ease of use, making it accessible for both library staff and patrons.

Add Book ,Journal and Magazine Forms :

The "Add Book" form includes fields for entering essential book details such as title, author, number, genre, version, and date of publication. This form streamlines the process of adding new books to the library catalog.

Add Student Form:

A dedicated form allows staff to input student information, including ID, name, address, birth date, and major. This form facilitates the efficient registration of students within the library system.

Search and Show Books Forms:

The system provides multiple search forms for quick and precise retrieval of information:

Search by Title: Users can enter a book title to find relevant information.

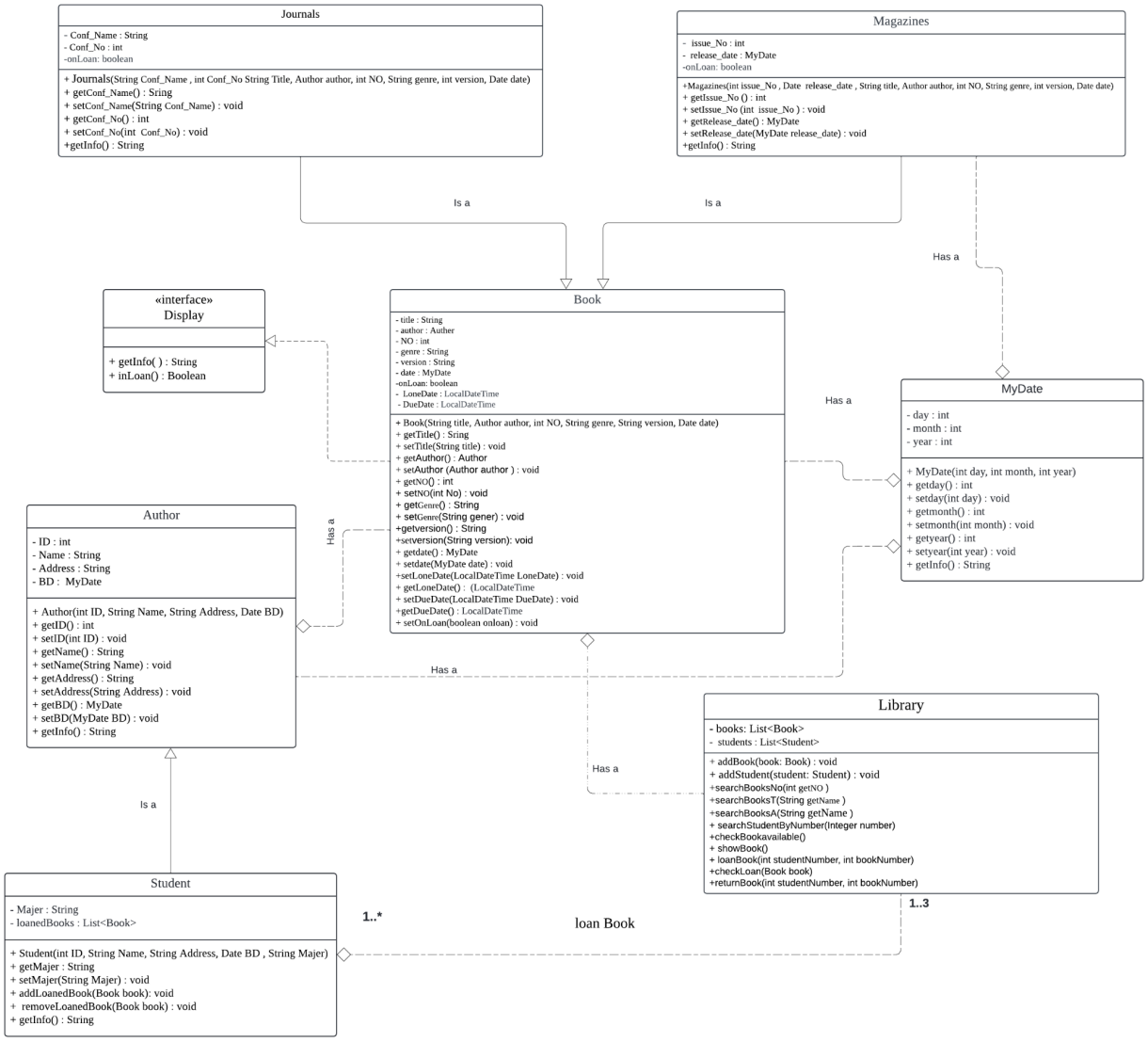
Search by Number: This form enables users to locate a book using its unique identification number.

Search by Author: Users can input an author's name to discover books authored by a specific individual.

Loan and Return Forms :

Streamlining the loan and return processes, the GUI includes forms for both actions. Staff can efficiently manage book loans and returns through these intuitive interfaces.

- UML Diagram


Journals
- Conf_Name : String
- Conf_No : int
-onLoan: boolean
+ Journals(String Conf_Name , int Conf_No String Title, Author author, int NO, String genre, int version, Date date)
+ getConf_Name() : Sring
+ setConf_Name(String Conf_Name) : void
+ getConf_No() : int
+ setConf_No(int Conf_No) : void
+getInfo() : String
Magazines
- issue_No : int
- release_date : MyDate
-onLoan: boolean
+Magazines(int issue_No , Date release_date , String title, Author author, int NO, String genre, int version, Date date)
+ getIssue_No () : int
+ setIssue_No (int issue_No) : void
+ getRelease_date() : MyDate
+ setRelease_date(MyDate release_date) : void
+getInfo() : String
Is a
Is a
Has a
«interface»
Display
+ getInfo( ) : String
+ inLoan() : Boolean
Book
- title : String
- author : Auther
- NO : int
- genre : String
- version : String
- date : MyDate
-onLoan: boolean
- LoneDate : LocalDateTime
- DueDate : LocalDateTime
+ Book(String title, Author author, int NO, String genre, String version, Date date)
+ getTitle() : Sring
+ setTitle(String title) : void
+ getAuthor() : Author
+ setAuthor (Author author) : void
+ getNO() : int
+ setNO(int No) : void
+ getGenre() : String
+ setGenre(String gener) : void
+getversion() : String
+setversion(String version): void
+ getdate() : MyDate
+ setdate(MyDate date) : void
+setLoneDate(LocalDateTime LoneDate) : void
+ getLoneDate() : (LocalDateTime
+ setDueDate(LocalDateTime DueDate) : void
+getDueDate() : LocalDateTime
+ setOnLoan(boolean onloan) : void
Has a
MyDate
- day : int
- month : int
- year : int
+ MyDate(int day, int month, int year)
+ getday() : int
+ setday(int day) : void
+ getmonth() : int
+ setmonth(int month) : void
+ getyear() : int
+ setyear(int year) : void
+ getInfo() : String
Author
- ID : int
- Name : String
- Address : String
- BD : MyDate
+ Author(int ID, String Name, String Address, Date BD)
+ getID() : int
+ setID(int ID) : void
+ getName() : String
+ setName(String Name) : void
+ getAddress() : String
+ setAddress(String Address) : void
+ getBD() : MyDate
+ setBD(MyDate BD) : void
+ getInfo() : String
Has a
Has a
Has a
Library
- books: List<Book>
- students : List<Student>
+ addBook(book: Book) : void
+ addStudent(student: Student) : void
+searchBooksNo(int getNO)
+searchBooksT(String getName)
+searchBooksA(String getName)
+ searchStudentByNumber(Integer number)
+checkBookavailable()
+ showBook()
+ loanBook(int studentNumber, int bookNumber)
+checkLoan(Book book)
+returnBook(int studentNumber, int bookNumber)
1..3
Is a
Student
- Majer : String
- loanedBooks : List<Book>
+ Student(int ID, String Name, String Address, Date BD , String Majer)
+ getMajer : String
+ setMajer(String Majer) : void
+ addLoanedBook(Book book): void
+ removeLoanedBook(Book book) : void
+ getInfo() : String
1..*
loan Book