JAVA MINI PROJECT

LIBRARY MANAGEMENT SYSTEM

NAME: A.SHREYASRI

ROLLNO:231401103

LIBRARY MANAGEMENT SYSTEM

```
import java.util.*;

class Book {
    private String title;
    private String author;
    private boolean isAvailable;

    public Book(String title, String author) {
        this.title = title;
        this.author = author;
        this.isAvailable = true;
    }

    public String getTitle() {
        return title;
    }

    public String getAuthor() {
        return author;
    }

    public boolean isAvailable() {
        return isAvailable;
    }
```

```
    public void borrowBook() {
        if (isAvailable) {
            isAvailable = false;
            System.out.println("You have borrowed: " + title);
        } else {
            System.out.println("Sorry, the book is currently unavailable.");
        }
    }

    public void returnBook() {
        if (!isAvailable) {
            isAvailable = true;
            System.out.println("You have returned: " + title);
        } else {
            System.out.println("This book wasn't borrowed.");
        }
    }
}

class Library {
    private List<Book> books;

    public Library() {
        books = new ArrayList<>();
    }

    public void addBook(String title, String author) {
        Book newBook = new Book(title, author);
        books.add(newBook);
        System.out.println("Book added: " + title);
```

```
    }

    public void listBooks() {
        if (books.isEmpty()) {
            System.out.println("No books in the library.");
            return;
        }
        System.out.println("Books available in the library:");
        for (Book book : books) {
            System.out.println(book.getTitle() + " by " + book.getAuthor() + " - " + (book.isAvailable() ?
"Available" : "Borrowed"));
        }
    }

    public void borrowBook(String title) {
        for (Book book : books) {
            if (book.getTitle().equalsIgnoreCase(title)) {
                book.borrowBook();
                return;
            }
        }
        System.out.println("Book not found.");
    }

    public void returnBook(String title) {
        for (Book book : books) {
            if (book.getTitle().equalsIgnoreCase(title)) {
                book.returnBook();
                return;
            }
        }
```

```
        System.out.println("Book not found.");
    }

    public void searchBook(String searchQuery) {
        boolean found = false;
        for (Book book : books) {
            if (book.getTitle().toLowerCase().contains(searchQuery.toLowerCase()) ||
                book.getAuthor().toLowerCase().contains(searchQuery.toLowerCase())) {
                found = true;
                System.out.println("Found: " + book.getTitle() + " by " + book.getAuthor() + " - " +
(book.isAvailable() ? "Available" : "Borrowed"));
            }
        }
        if (!found) {
            System.out.println("No books found matching the search query.");
        }
    }
}

public class LibraryManagementSystem {
    public static void main(String[] args) {
        Library library = new Library();
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println("\nLibrary Management System");
            System.out.println("1. Add Book");
            System.out.println("2. List Books");
            System.out.println("3. Borrow Book");
            System.out.println("4. Return Book");
            System.out.println("5. Search Book");
```

```
System.out.println("6. Exit");
System.out.print("Enter choice: ");
int choice = scanner.nextInt();
scanner.nextLine(); // Consume newline

switch (choice) {
    case 1:
        System.out.print("Enter book title: ");
        String title = scanner.nextLine();
        System.out.print("Enter book author: ");
        String author = scanner.nextLine();
        library.addBook(title, author);
        break;
    case 2:
        library.listBooks();
        break;
    case 3:
        System.out.print("Enter book title to borrow: ");
        String borrowTitle = scanner.nextLine();
        library.borrowBook(borrowTitle);
        break;
    case 4:
        System.out.print("Enter book title to return: ");
        String returnTitle = scanner.nextLine();
        library.returnBook(returnTitle);
        break;
    case 5:
        System.out.print("Enter search query (title/author): ");
        String query = scanner.nextLine();
        library.searchBook(query);
        break;
```

```
            case 6:
                System.out.println("Exiting... Thank you for using the Library Management System.");
                scanner.close();
                return;
            default:
                System.out.println("Invalid choice, please try again.");
        }
      }
    }
}
```