Library management
231901062
231901061
231901017

```
import java.util.ArrayList;
import java.util.Scanner;

// Book class to represent a book
class Book {
    private int id;
    private String title;
    private String author;
    private boolean isAvailable;

    // Constructor
    public Book(int id, String title, String author) {
        this.id = id;
        this.title = title;
        this.author = author;
        this.isAvailable = true;  // Book is available by default
    }

    // Getters
    public int getId() {
        return id;
    }

    public String getTitle() {
        return title;
    }

    public String getAuthor() {
        return author;
    }

    public boolean isAvailable() {
        return isAvailable;
    }

    // Setters
    public void setAvailable(boolean available) {
        isAvailable = available;
    }
```

```
    // Method to display book information
    public void displayInfo() {
        System.out.println("ID: " + id + " | Title: " + title + " | Author: " + author +
            " | Available: " + (isAvailable ? "Yes" : "No"));
    }
}
// Library class to manage the books
public class LibraryManagementSystem {
    private ArrayList<Book> books;
    private Scanner scanner;

    // Constructor
    public LibraryManagementSystem() {
        books = new ArrayList<>();
        scanner = new Scanner(System.in);
    }

    // Add new book to the library
    public void addBook() {
        System.out.print("Enter book ID: ");
        int id = scanner.nextInt();
        scanner.nextLine();  // Consume newline

        System.out.print("Enter book title: ");
        String title = scanner.nextLine();

        System.out.print("Enter book author: ");
        String author = scanner.nextLine();

        Book newBook = new Book(id, title, author);
        books.add(newBook);
        System.out.println("Book added successfully!");
    }

    // View all books in the library
    public void viewAllBooks() {
        if (books.isEmpty()) {
            System.out.println("No books available in the library.");
        } else {
            System.out.println("\nBooks in the Library:");
            for (Book book : books) {
                book.displayInfo();
            }
```

```
    }
}

// Borrow a book
public void borrowBook() {
    System.out.print("Enter the ID of the book you want to borrow: ");
    int id = scanner.nextInt();

    Book book = findBookById(id);

    if (book != null) {
        if (book.isAvailable()) {
            book.setAvailable(false);
            System.out.println("You have successfully borrowed the book: " + book.getTitle());
        } else {
            System.out.println("Sorry, the book is currently unavailable.");
        }
    } else {
        System.out.println("Book not found.");
    }
}
// Return a borrowed book
public void returnBook() {
    System.out.print("Enter the ID of the book you want to return: ");
    int id = scanner.nextInt();

    Book book = findBookById(id);

    if (book != null) {
        if (!book.isAvailable()) {
            book.setAvailable(true);
            System.out.println("You have successfully returned the book: " + book.getTitle());
        } else {
            System.out.println("This book was not borrowed.");
        }
    } else {
        System.out.println("Book not found.");
    }
}

// Search for a book by title
public void searchBookByTitle() {
    scanner.nextLine();  // Consume newline
    System.out.print("Enter book title to search: ");
```

```
        String title = scanner.nextLine();

        boolean found = false;
        for (Book book : books) {
            if (book.getTitle().toLowerCase().contains(title.toLowerCase())) {
                book.displayInfo();
                found = true;
            }
        }

        if (!found) {
            System.out.println("No books found with that title.");
        }
    }

    // Find a book by ID
    private Book findBookById(int id) {
        for (Book book : books) {
            if (book.getId() == id) {
                return book;
            }
        }
        return null;
    }

    // Display the menu
    public void displayMenu() {
        System.out.println("\n*** Library Management System ***");
        System.out.println("1. Add Book");
        System.out.println("2. View All Books");
        System.out.println("3. Borrow Book");
        System.out.println("4. Return Book");
        System.out.println("5. Search Book by Title");
        System.out.println("6. Exit");
    }

    // Main method
    public static void main(String[] args) {
        LibraryManagementSystem library = new LibraryManagementSystem();
        int choice;

        // Main loop to keep the system running until exit
        do {
            library.displayMenu();
```

```
            System.out.print("Enter your choice: ");
            choice = library.scanner.nextInt();

            switch (choice) {
                case 1:
                    library.addBook();
                    break;
                case 2:
                    library.viewAllBooks();
                    break;
                case 3:
                    library.borrowBook();
                    break;
                case 4:
                    library.returnBook();
                    break;
                case 5:
                    library.searchBookByTitle();
                    break;
                case 6:
                    System.out.println("Exiting the system...");
                    break;
                default:
                    System.out.println("Invalid choice. Please try again.");
            }
        } while (choice != 6);
    }
}
```