**Exercise 6: Library Management System**

**1. Understand Search Algorithms**

**Linear Search:**

* **How it works: Checks each element one by one until a match is found.**
* **Best Case: O(1) — if the match is first.**
* **Worst Case: O(n) — if the match is last or not found at all.**
* **Use when:**
  + **Data is unsorted.**
  + **Small data sets.**

**Binary Search:**

* **How it works: Repeatedly divides a sorted list in half to locate the element.**
* **Best Case: O(1)**
* **Average/Worst Case: O(log n)**
* **Use when:**
  + **Data is sorted.**
  + **Large data sets.**

**2. Setup**

Create a class Book with:

* bookId (int)
* title (String)
* author (String)

**3. Implementation**

* Book.java

public class Book {

int bookId;

String title;

String author;

public Book(int bookId, String title, String author) {

this.bookId = bookId;

this.title = title;

this.author = author;

}

@Override

public String toString() {

return bookId + " - " + title + " by " + author;

}

}

* Library.java

import java.util.Arrays;

import java.util.Comparator;

public class Library {

Book[] books;

int count;

public Library(int size) {

books = new Book[size];

count = 0;

}

public void addBook(Book book) {

if (count < books.length) {

books[count++] = book;

}

}

public Book linearSearchByTitle(String title) {

for (int i = 0; i < count; i++) {

if (books[i].title.equalsIgnoreCase(title)) {

return books[i];

}

}

return null;

}

public void sortBooksByTitle() {

Arrays.sort(books, 0, count, Comparator.comparing(b -> b.title.toLowerCase()));

}

public Book binarySearchByTitle(String title) {

int left = 0, right = count - 1;

while (left <= right) {

int mid = (left + right) / 2;

int cmp = books[mid].title.compareToIgnoreCase(title);

if (cmp == 0) return books[mid];

else if (cmp < 0) left = mid + 1;

else right = mid - 1;

}

return null;

}

public void displayBooks() {

for (int i = 0; i < count; i++) {

System.out.println(books[i]);

}

}

}

* LibraryTest.java:

public class LibraryTest {

public static void main(String[] args) {

Library lib = new Library(5);

lib.addBook(new Book(1, "The Alchemist", "Paulo Coelho"));

lib.addBook(new Book(2, "Clean Code", "Robert C. Martin"));

lib.addBook(new Book(3, "1984", "George Orwell"));

System.out.println("Books in Library:");

lib.displayBooks();

System.out.println("\nLinear Search for '1984':");

Book result = lib.linearSearchByTitle("1984");

System.out.println(result != null ? result : "Book not found");

System.out.println("\nSorting books for binary search...");

lib.sortBooksByTitle();

System.out.println("\nBinary Search for 'Clean Code':");

Book result2 = lib.binarySearchByTitle("Clean Code");

System.out.println(result2 != null ? result2 : "Book not found");

}

}  
4. Analysis

| **Search Type** | **Time Complexity** | **Best For** |
| --- | --- | --- |
| **Linear Search** | **O(n)** | **Small, unsorted datasets** |
| **Binary Search** | **O(log n)** | **Large, sorted datasets** |

**Output :**

