**Exercise 6: Library Management System**

Scenario: Developing a library management system where users can search for books by title or author.

1. Understand Search Algorithms:

* **Linear Search:** Goes through each element in the list one by one. Time Complexity: O(n).
* **Binary Search:** Only works on sorted data. It divides the list and searches in halves. Time Complexity: O(log n).

Use linear search for unsorted or small data sets. Binary search is efficient for large, sorted data sets.

1. Setup: Create a class Book with the following attributes:

* String bookId
* String title
* String author

1. Implementation:

Java Code:

import java.util.\*;

class Book {

String bookId;

String title;

String author;

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

this.bookId = bookId;

this.title = title;

this.author = author;

}

public String toString() {

return bookId + " | " + title + " | " + author;

}

}

class LibrarySystem {

Book[] books;

int count;

public LibrarySystem(int size) {

books = new Book[size];

count = 0;

}

void addBook(Book book) {

books[count++] = book;

}

void linearSearch(String title) {

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

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

System.out.println("Found: " + books[i]);

return;

}

}

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

}

void binarySearch(String title) {

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

int low = 0, high = count - 1;

while (low <= high) {

int mid = (low + high) / 2;

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

if (cmp == 0) {

System.out.println("Found: " + books[mid]);

return;

} else if (cmp < 0) {

low = mid + 1;

} else {

high = mid - 1;

}

}

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

}

void displayBooks() {

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

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

}

}

public static void main(String[] args) {

LibrarySystem lib = new LibrarySystem(10);

lib.addBook(new Book("B001", "Java Basics", "John"));

lib.addBook(new Book("B002", "Data Structures", "Alice"));

lib.addBook(new Book("B003", "Algorithms", "Bob"));

System.out.println("All Books:");

lib.displayBooks();

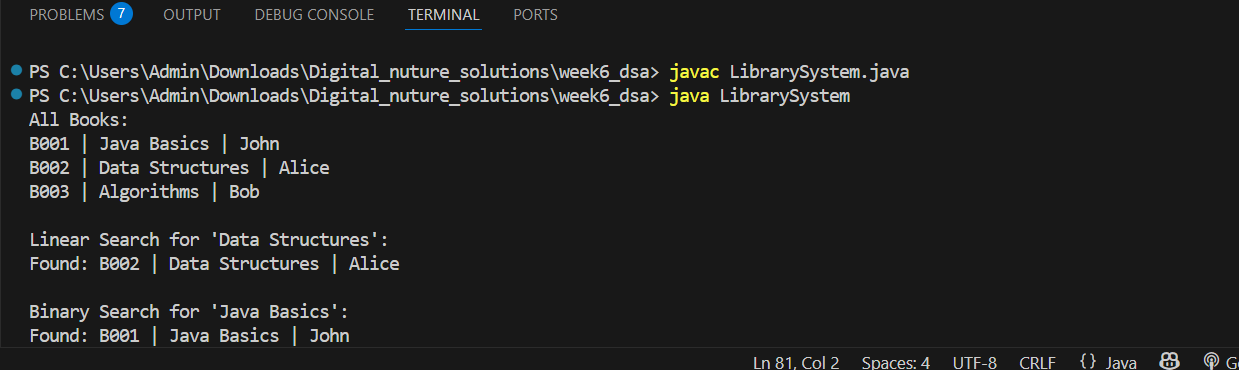
System.out.println("\nLinear Search for 'Data Structures':");

lib.linearSearch("Data Structures");

System.out.println("\nBinary Search for 'Java Basics':");

lib.binarySearch("Java Basics");

}

}  
  
OUTPUT:  
  


1. Analysis:

* **Linear Search:** O(n). Simple and works on any list.
* **Binary Search:** O(log n). Fast but requires a sorted list.

Choose search type based on data order and size:

* Use Linear Search for unsorted or small lists.
* Use Binary Search for sorted and large dataset.