**Exercise 2: E-commerce Platform Search Function**

**Scenario :**

You are working on the search functionality of an e-commerce platform. The search needs to be optimized for fast performance.

### **1. Asymptotic Notation (Big O)**

Big O notation describes the performance (time or space) of an algorithm as the input size grows

| **Algorithm** | **Best Case** | **Average Case** | **Worst Case** |
| --- | --- | --- | --- |
| **Linear Search** | O(1) | O(n) | O(n) |
| **Binary Search** | O(1) | O(log n) | O(log n) |

**2. Java Code :**

import java.util.Arrays;

import java.util.Comparator;

public class ECommerceSearch {

static class Product {

int productId;

String productName;

String category;

Product(int id, String name, String category) {

this.productId = id;

this.productName = name;

this.category = category;

}

public String toString() {

return "Product ID: " + productId + ", Name: " + productName + ", Category: " + category;

}

}

public static Product linearSearch(Product[] products, String name) {

for (Product product : products) {

if (product.productName.equalsIgnoreCase(name)) {

return product;

}

}

return null;

}

public static Product binarySearch(Product[] products, String name) {

int left = 0, right = products.length - 1;

while (left <= right) {

int mid = left + (right - left) / 2;

int cmp = products[mid].productName.compareToIgnoreCase(name);

if (cmp == 0)

return products[mid];

else if (cmp < 0)

left = mid + 1;

else

right = mid - 1;

}

return null;

}

public static void main(String[] args) {

Product[] products = {

new Product(1, "Laptop", "Electronics"),

new Product(2, "Shoes", "Footwear"),

new Product(3, "Camera", "Electronics"),

new Product(4, "Phone", "Electronics"),

new Product(5, "Watch", "Accessories")

};

// Linear Search Test

Product result1 = linearSearch(products, "Camera");

System.out.println("Linear Search Result: " + (result1 != null ? result1 : "Not Found"));

// Binary Search requires sorted array

Arrays.sort(products, Comparator.comparing(p -> p.productName.toLowerCase()));

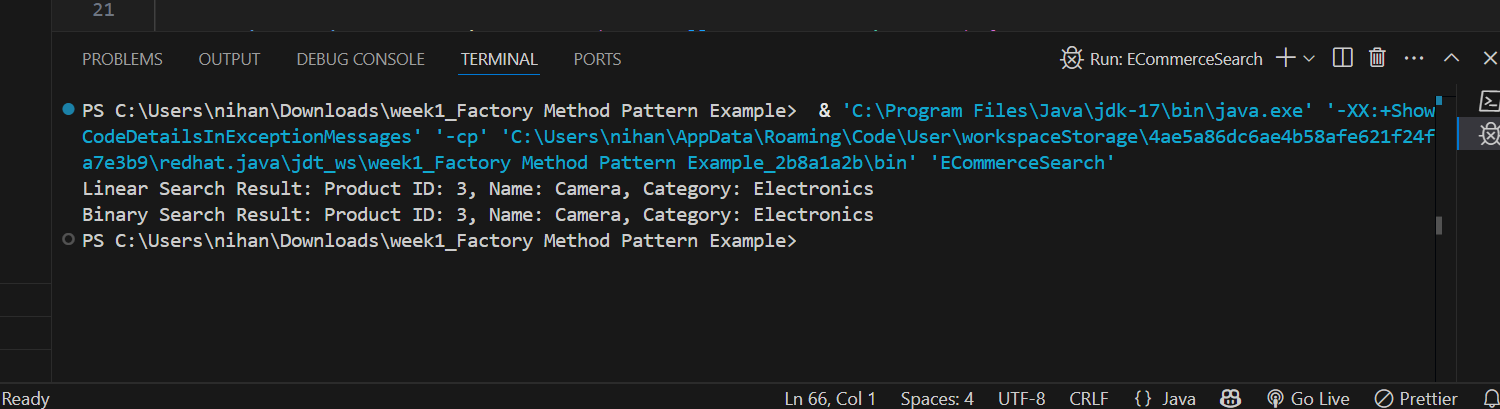
Product result2 = binarySearch(products, "Camera");

System.out.println("Binary Search Result: " + (result2 != null ? result2 : "Not Found"));

}

}

**Output :**

****