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

BinarySearch.java

import java.util.Arrays;

import java.util.Comparator;

public class BinarySearch {

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

        // Sort array by name

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

        int low = 0, high = products.length - 1;

        while (low <= high) {

            int mid = (low + high) / 2;

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

            if (comparison == 0) {

                return products[mid];

            } else if (comparison < 0) {

                high = mid - 1;

            } else {

                low = mid + 1;

            }

        }

        return null;

    }

}

LinearSearch.java

public class LinearSearch {

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

        for (Product product : products) {

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

                return product;

            }

        }

        return null;

    }

}

Product.java

public class Product {

    public int productId;

    public String productName;

    public String category;

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

        this.productId = id;

        this.productName = name;

        this.category = category;

    }

    public String toString() {

        return productId + " - " + productName + " (" + category + ")";

    }

}

TestProduct.java

public class TestSearch {

    public static void main(String[] args) {

        Product[] products = {

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

            new Product(2, "Shirt", "Clothing"),

            new Product(3, "Book", "Education"),

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

            new Product(5, "Pen", "Stationery")

        };

        // Test Linear Search

        System.out.println("🔎 Linear Search:");

        Product result1 = LinearSearch.searchByName(products, "Phone");

        System.out.println(result1 != null ? result1 : "Product not found");

        // Test Binary Search

        System.out.println("\n🔎 Binary Search:");

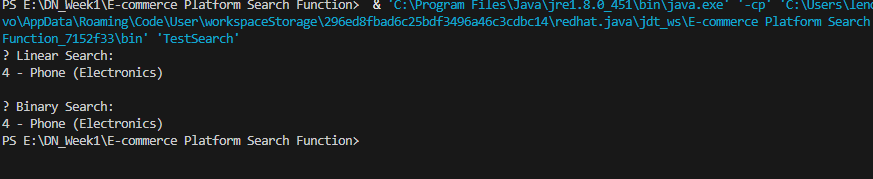
        Product result2 = BinarySearch.searchByName(products, "Phone");

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

    }

}

OUTPUT:



**ANALYSIS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Search Method** | **Time Complexity** | **Use Case** | **Notes** |
| Linear Search | Best: O(1)  Avg/Worst: O(n) | Small product datasets | Simple to implement, inefficient for large lists |
| Binary Search | Best: O(1)  Avg/Worst: O(log n) | Sorted product arrays (by name or ID) | Fast, but requires data to be sorted before searching |