DataStructures and Algorithms  
  
Exercise 2: E-commerce Platform Search Function  
  
code  
import java.util.Arrays;

import java.util.Comparator;

class Product {

int productId;

String productName;

String category;

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

this.productId = id;

this.productName = name;

this.category = cat;

}

}

class ProductSearch {

// Linear search by product name

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

for (Product product : products) {

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

return product;

}

}

return null;

}

// Binary search by product name (assuming sorted)

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

int left = 0;

int right = products.length - 1;

while (left <= right) {

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

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

if (comparison == 0) {

return products[mid];

} else if (comparison < 0) {

left = mid + 1;

} else {

right = mid - 1;

}

}

return null;

}

}

public class Main {

public static void main(String[] args) {

// Sample products

Product[] products = {

new Product(101, "Laptop", "Electronics"),

new Product(102, "Shoes", "Footwear"),

new Product(103, "Watch", "Accessories"),

new Product(104, "Mobile", "Electronics"),

new Product(105, "Bag", "Travel")

};

// Linear Search

System.out.println("=== Linear Search ===");

Product foundLinear = ProductSearch.linearSearch(products, "Mobile");

if (foundLinear != null) {

System.out.println("Product found: " + foundLinear.productName + " in " + foundLinear.category);

} else {

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

}

// Sort products for binary search

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

// Binary Search

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

Product foundBinary = ProductSearch.binarySearch(products, "Mobile");

if (foundBinary != null) {

System.out.println("Product found: " + foundBinary.productName + " in " + foundBinary.category);

} else {

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

}

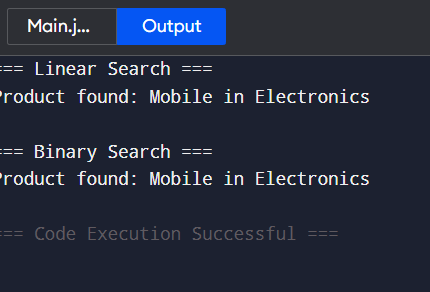
}

}

**✅ Time Complexity Analysis:**

| **Algorithm** | **Time Complexity** | **Best Use Case Scenario** |
| --- | --- | --- |
| Linear Search | O(n) | Unsorted product data |
| Binary Search | O(log n) | Sorted product data and frequent queries |

**OUTPUT**

****