Exercise 2: E-commerce Platform Search Function

public class ecommerces {

static class Product {

private int productId;

private String productName;

private String category;

public Product(int productId, String productName, String category) {

this.productId = productId;

this.productName = productName;

this.category = category;

}

public String getProductName() {

return productName;

}

@Override

public String toString() {

return "Product[ID=" + productId + ", Name=" + productName + ", Category=" + category + "]";

}

}

// SearchFunctions class

static class SearchFunctions {

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

for (Product product : products) {

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

return product;

}

}

return null;

}

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

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

while (left <= right) {

int mid = (left + right) / 2;

String midName = products[mid].getProductName().toLowerCase();

String target = targetName.toLowerCase();

if (midName.equals(target)) {

return products[mid];

} else if (midName.compareTo(target) < 0) {

left = mid + 1;

} else {

right = mid - 1;

}

}

return null;

}

}

//Main method ---

public static void main(String[] args) {

Product[] products = {

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

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

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

new Product(4, "Backpack", "Accessories")

};

// Linear search

Product foundLinear = SearchFunctions.linearSearch(products, "Smartphone");

System.out.println("Linear Search Result: " + foundLinear);

// Sort array for binary search

java.util.Arrays.sort(products, java.util.Comparator.comparing(p -> p.getProductName().toLowerCase()));

// Binary search

Product foundBinary = SearchFunctions.binarySearch(products, "Smartphone");

System.out.println("Binary Search Result: " + foundBinary);

}

}

