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.

**Steps:**

1. **Understand Asymptotic Notation:**
   * Explain Big O notation and how it helps in analyzing algorithms.
   * Describe the best, average, and worst-case scenarios for search operations.
2. **Setup:**
   * Create a class **Product** with attributes for searching, such as **productId, productName**, and **category**.
3. **Implementation:**
   * Implement linear search and binary search algorithms.
   * Store products in an array for linear search and a sorted array for binary search.
4. **Analysis:**
   * Compare the time complexity of linear and binary search algorithms.
   * Discuss which algorithm is more suitable for your platform and why.

Code :

import java.util.\*;

class Product {

    int productId;

    String productName;

    String category;

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

        this.productId = productId;

        this.productName = productName;

        this.category = category;

    }

    public String toString() {

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

    }

}

public class ProductSearchSystem {

    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) / 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(101, "Laptop", "Electronics"),

            new Product(102, "Phone", "Electronics"),

            new Product(103, "Shoes", "Fashion"),

            new Product(104, "Watch", "Accessories")

        };

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter product name to search: ");

        String searchName = scanner.nextLine();

        Product linearResult = linearSearch(products, searchName);

        System.out.println("\nLinear Search Result:");

        System.out.println(linearResult != null ? linearResult : "Product Not Found");

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

        Product binaryResult = binarySearch(products, searchName);

        System.out.println("\nBinary Search Result:");

        System.out.println(binaryResult != null ? binaryResult : "Product Not Found");

        scanner.close();

    }

}

Output :

