Exercise 2: E-commerce Platform Search Function

1. Understanding Asymptotic Notation

Big O Notation is used to analyze the performance of an algorithm in terms of time and space. It describes how the runtime grows with input size. It helps developers to analyze the efficiency of algorithms allowing them to select the best algorithm for large-scale applications.

- **Linear Search** :
    - **Best Case**: O(1) (first element is the match)
    - **Average**: O(n)
    - **Worst Case**: O(n)
- **Binary Search (only on sorted data)**:
    - **Best Case**: O(1) (Middle element is the match)
    - **Average** : O(n)
    - **Worst Case**: O(log n)

This helps in choosing the most efficient search approach based on data size and structure.

2. Setup :

We created a Product class with the following attributes:

- productId (Product ID number starts from 1000)
- productName
- category
- price

To **maintain clean and flexible object construction**, I used the **Builder Design Pattern** in the Product class.
The user is allowed to input search criteria such as ID, name, category, or price range.

3. Implementation

Two search techniques are implemented:

- **Linear Search:**
    - Traverses the array from start to end.
    - No sorting required.
    - Suitable for unsorted or small data.
- **Binary Search:**
    - Requires data to be pre-sorted based on search attribute.
    - Uses divide-and-conquer by checking the middle element recursively or iteratively.

The user can choose between Linear or Binary Search, and then select:

- Search by ID
- Search by Name

- Search by Category
- Search by Price Range

All inputs are collected via Scanner, making it interactive and testable in real time.

4. Analysis

Search Type	Time Complexity	Sorting Needed	Space
Linear Search	O(n)	No	O(1)
Binary Search	O(log n)	Yes	O(1)

- Linear search is simpler and doesn't need sorting, while binary search is much faster for large, sorted datasets.
- In real-world e-commerce systems, a combination is often used based on query type and data indexing.

Conclusion

This solution combines:

- Good object-oriented design (via Builder Pattern)
- Real-time user input handling
- Dual search algorithm implementation

It simulates the core functionality of a real-world e-commerce platform's search feature, demonstrating both conceptual understanding and practical Java skills.