## Data Structures and Algorithms

## Exercise 2:

## E-Commerce Platform Search Function

This project implements a fast and optimized search functionality for an E-Commerce Platform using Java. It compares linear and binary search algorithms to efficiently retrieve products based on user queries. Here's a detailed explanation:

## Step 1: Understanding Asymptotic Notations

- **Big O Notation:** Describes algorithm efficiency based on input size
- Linear Search:
  - **Best Case:** *O(1)* (if target is at beginning)
  - **Average/Worst Case:** *O(n)*
- Binary Search:
  - **Best Case:** *O(1)* (middle element is the target)
  - **Average/Worst Case:** *O(log n)*
- Requires Sorted array**.**

---

## Step 2: Setup

- A *Product* class is created with *productId*, *productName*, and *category*.

---

## Step 3: Implementation

- Two search methods:
  - *linearSearch()* – works on unsorted arrays.
  - *binarySearch()* – works on sorted arrays.
- Products are stored in an array and sorted using *Arrays.sort()* for binary search.

---

## Step 4: Time Complexity Analysis

| Search Method | Time Complexity | Requirement |
|---|---|---|
| Linear Search | O(n) | Unsorted allowed |
| Binary Search | O(log n) | Sorted array only |

- **Binary Search** is preferred for large, sorted datasets due to its logarithmic efficiency.

---

## Output