E-commerce Platform Search Function – Java

Exercise Title: Search Optimization in E-commerce
Module: Data Structures and Algorithms
Track: DN 4.0 DotNet FSE Deep Skilling Program

Objective
To implement and analyze different search algorithms—Linear Search and Binary Search—in an e-commerce platform, improving the performance of product searches based on Big O analysis.

Concepts Applied
* Algorithm Analysis: Asymptotic Notation (Big O)
* Time Complexity Evaluation
* Linear and Binary Search Algorithms
* Object-Oriented Programming (OOP) with Java
* Data Preparation and Sorting

Problem Summary
Build an efficient search system where:
* Each product is represented by a Product class with attributes like productId, productName, and category.
* You implement both linear search (for unsorted data) and binary search (for sorted data).
* Analyze and compare both search methods based on best, average, and worst-case performance using Big O notation.
* Conclude which method is better suited for the e-commerce use case.

Deliverables
* A Product class with relevant search attributes
* Linear search implementation using an unsorted array
* Binary search implementation using a sorted array
* Analysis of search performance in comments or separate documentation
* Organized week-wise folder structure for submission

Tools & Technologies
* Java
* IDE (IntelliJ, Eclipse, or VS Code)

Evaluation Criteria
* Accurate implementation of both search algorithms
* Clear understanding of algorithm complexity through comments or print statements
* Proper use of Java OOP to define Product objects
* Clear documentation of results and conclusions
* Organized folder structure (e.g., Week1/SearchFunctionExample)
* Code successfully compiled, tested, and uploaded to GitHub

Self-Evaluation Checklist
* Product class created with necessary attributes
* Linear Search implemented for unsorted product list

* Binary Search implemented for sorted product list
* Time complexity (Big O) of both algorithms is well understood and documented
* Code structured and commented clearly
* Folder structure follows week-wise submission format
* Code compiled and tested successfully
* Uploaded to public GitHub repository