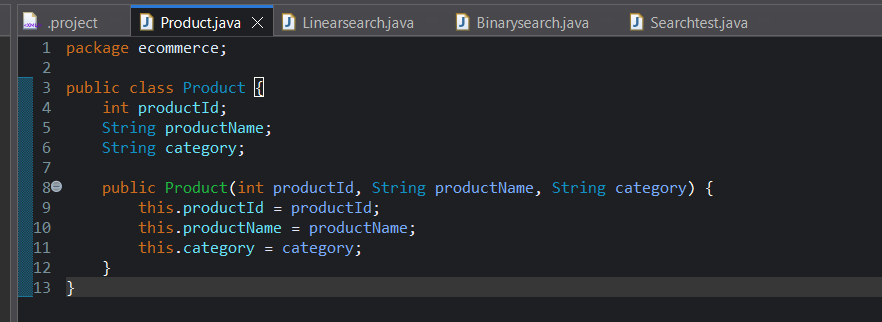
**Exercise 2: E-commerce Platform Search Function**

**Scenario:**

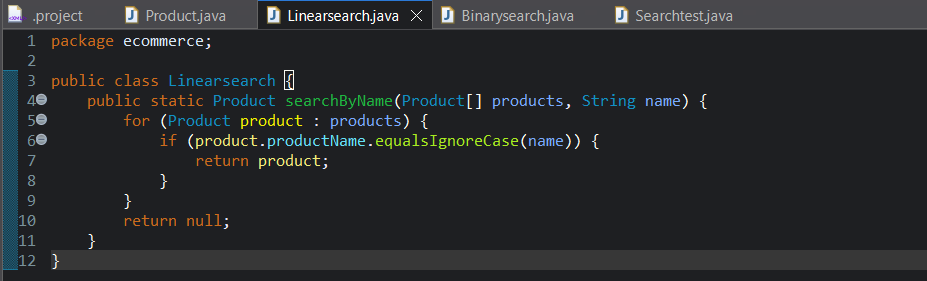
You are working on the search functionality of an e-commerce platform. The search needs to be optimized for fast performance.

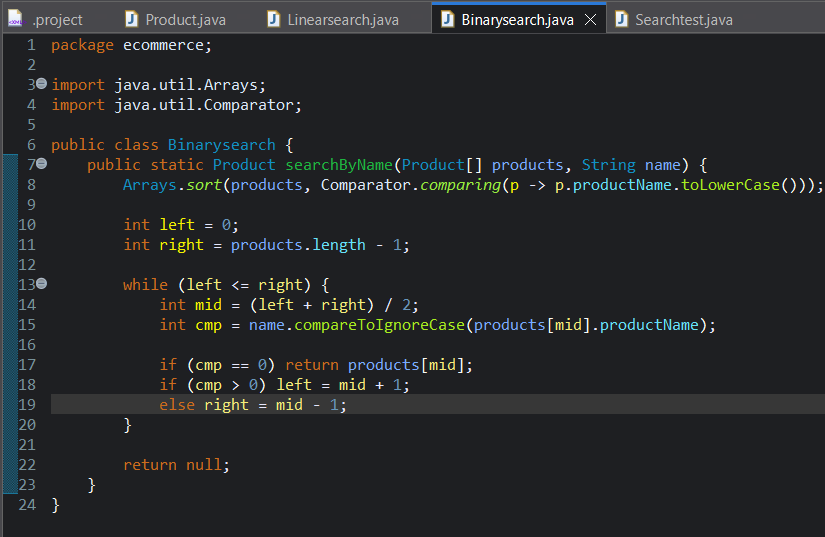
**Steps:**

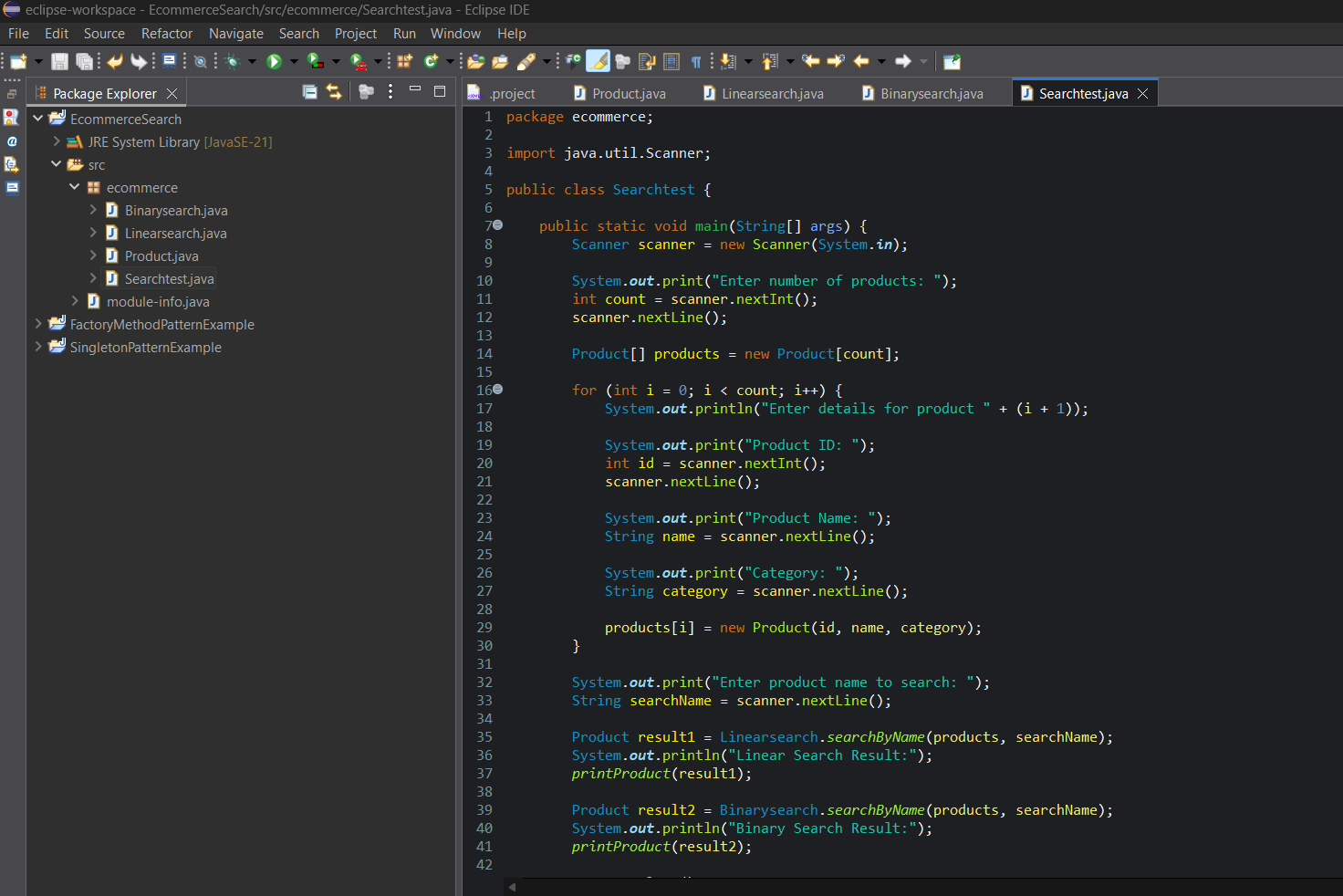
1. **Understand Asymptotic Notation:**
   * Explain Big O notation and how it helps in analyzing algorithms.
   * Describe the best, average, and worst-case scenarios for search operations.
2. **Setup:**
   * Create a class **Product** with attributes for searching, such as **productId, productName**, and **category**.



1. **Implementation:**
   * Implement linear search and binary search algorithms.
   * Store products in an array for linear search and a sorted array for binary search.







1. **Analysis:**
   * Compare the time complexity of linear and binary search algorithms.
   * Discuss which algorithm is more suitable for your platform and why.

**PROGRAM :**

package ecommerce;

public 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;

}

}

package ecommerce;

public class Linearsearch {

public static Product searchByName(Product[] products, String name) {

for (Product product : products) {

if (product.productName.equalsIgnoreCase(name)) {

return product;

}

}

return null;

}

}

package ecommerce;

import java.util.Arrays;

import java.util.Comparator;

public class Binarysearch {

public static Product searchByName(Product[] products, String name) {

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

int left = 0;

int right = products.length - 1;

while (left <= right) {

int mid = (left + right) / 2;

int cmp = name.compareToIgnoreCase(products[mid].productName);

if (cmp == 0) return products[mid];

if (cmp > 0) left = mid + 1;

else right = mid - 1;

}

return null;

}

}

package ecommerce;

import java.util.Scanner;

public class Searchtest {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.***in***);

System.***out***.print("Enter number of products: ");

int count = scanner.nextInt();

scanner.nextLine();

Product[] products = new Product[count];

for (int i = 0; i < count; i++) {

System.***out***.println("Enter details for product " + (i + 1));

System.***out***.print("Product ID: ");

int id = scanner.nextInt();

scanner.nextLine();

System.***out***.print("Product Name: ");

String name = scanner.nextLine();

System.***out***.print("Category: ");

String category = scanner.nextLine();

products[i] = new Product(id, name, category);

}

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

String searchName = scanner.nextLine();

Product result1 = Linearsearch.*searchByName*(products, searchName);

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

*printProduct*(result1);

Product result2 = Binarysearch.*searchByName*(products, searchName);

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

*printProduct*(result2);

scanner.close();

}

public static void printProduct(Product product) {

if (product != null) {

System.***out***.println("Product Found:");

System.***out***.println("ID: " + product.productId);

System.***out***.println("Name: " + product.productName);

System.***out***.println("Category: " + product.category);

} else {

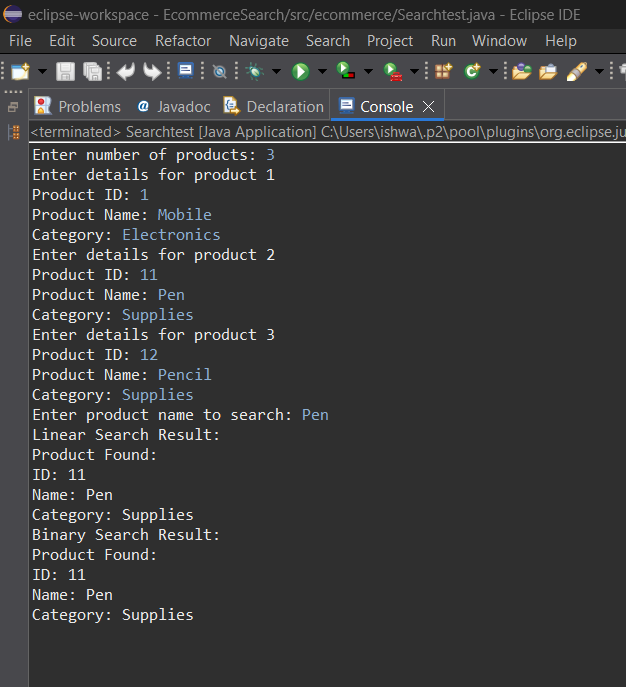
System.***out***.println("Product not found.");

}

}

}

**OUTPUT :**

****