Ecommerce Platform Search Function

Product.java—

package com.search;

public class Product {

private int productId;

private String productName;

private String category;

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

this.productId = productId;

this.productName = productName;

this.category = category;

}

public int getProductId() {

return productId;

}

public String getProductName() {

return productName;

}

public String getCategory() {

return category;

}

*@Override*

public String toString() {

return "[" + productId + ", " + productName + ", " + category + "]";

}

}

SearchDemo.java

package com.search;

import java.util.Arrays;

import java.util.Comparator;

public class SearchDemo {

public static void main(String[] args) {

Product[] products = {

new Product(101, "Phone", "Electronics"),

new Product(102, "Shirt", "Fashion"),

new Product(103, "Laptop", "Electronics"),

new Product(104, "Shoes", "Footwear"),

new Product(105, "Book", "Education")

};

// Linear Search

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

Product result1 = linearSearch(products, "Laptop");

System.out.println(result1 != null ? result1 : "Product not found");

// Sort for Binary Search (by name)

Arrays.sort(products, Comparator.comparing(Product::getProductName));

// Binary Search

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

Product result2 = binarySearch(products, "Laptop");

System.out.println(result2 != null ? result2 : "Product not found");

}

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

for (Product p : products) {

if (p.getProductName().equalsIgnoreCase(name)) {

return p;

}

}

return null;

}

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

int low = 0;

int high = products.length - 1;

while (low <= high) {

int mid = (low + high) / 2;

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

if (cmp == 0) {

return products[mid];

} else if (cmp < 0) {

low = mid + 1;

} else {

high = mid - 1;

}

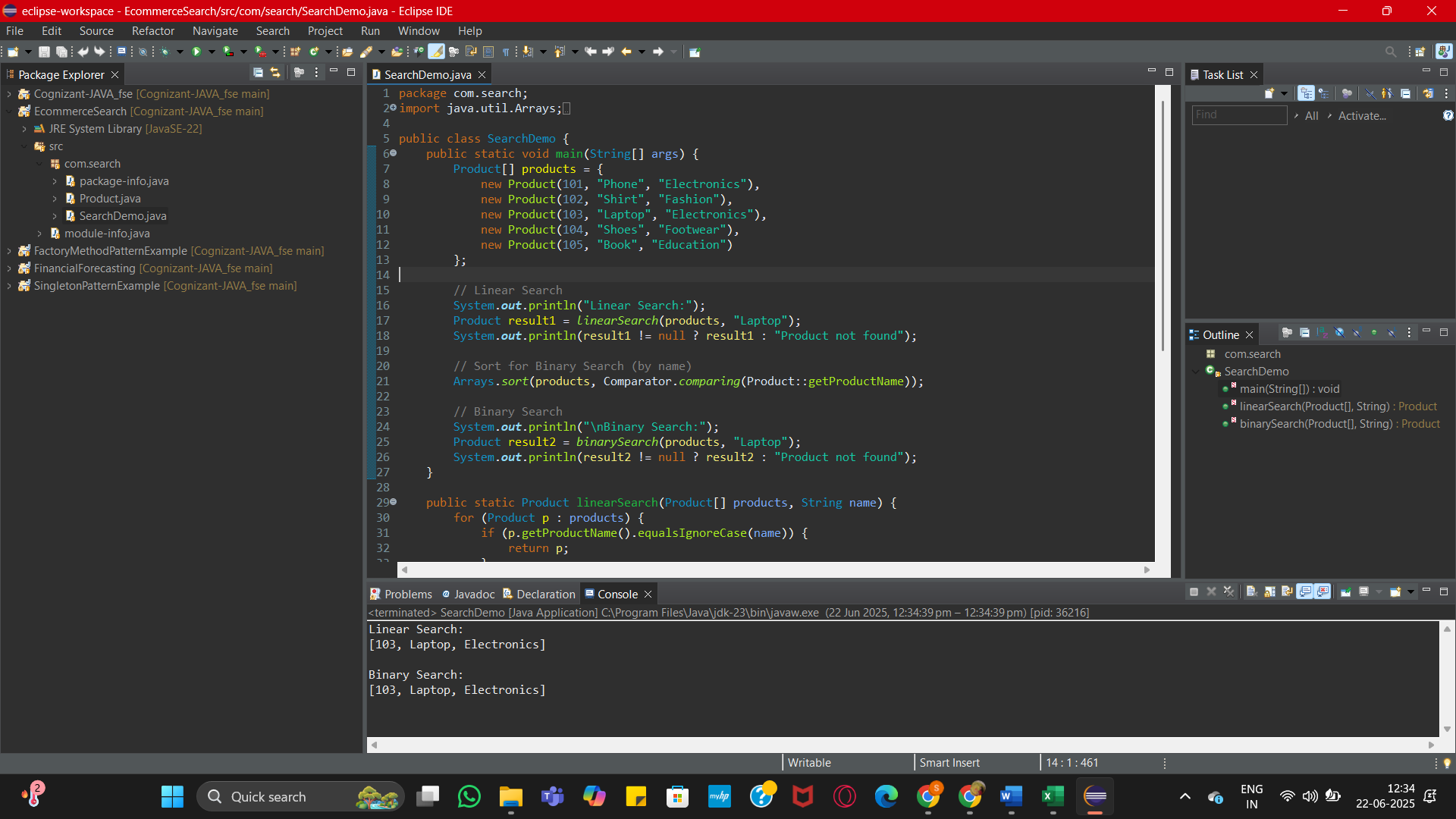
}

return null;

}

}

Output--



FinancialForecasting

FinancialForecast.java-

package com.forecast;

public class FinancialForecast {

// Recursive method to calculate future value

public static double calculateFutureValue(double presentValue, double rate, int years) {

if (years == 0) {

return presentValue; // base case

}

return (1 + rate) \* *calculateFutureValue*(presentValue, rate, years - 1);

}

public static void main(String[] args) {

double presentValue = 15000;

double rate = 0.10; // 10% annual growth

int years = 5;

double futureValue = *calculateFutureValue*(presentValue, rate, years);

System.***out***.printf("Future Value after %d years = ₹%.2f%n", years, futureValue);

}

}

