**Week 1\_AlgorithmsDataStructures \_HandsOn**

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

**Product.java**

package model;

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;

}

}

**BinarySearch.java**

package search;

import model.Product;

import java.util.Arrays;

import java.util.Comparator;

public class BinarySearch {

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

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

int low = 0, 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;

}

}

**LinearSearch.java**

package search;

import model.Product;

public class LinearSearch {

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

for (Product product : products) {

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

return product;

}

}

return null;

}

}

**SearchTest.java**

package test;

import model.Product;

import search.LinearSearch;

import search.BinarySearch;

public class SearchTest {

public static void main(String[] args) {

Product[] products = {

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

new Product(102, "Camera", "Electronics"),

new Product(103, "Shoes", "Fashion"),

new Product(104, "Watch", "Accessories"),

new Product(105, "Phone", "Electronics")

};

Product result1 = LinearSearch.searchByName(products, "Camera");

System.out.println("Linear Search: " + (result1 != null ? result1.getProductName() : "Not found"));

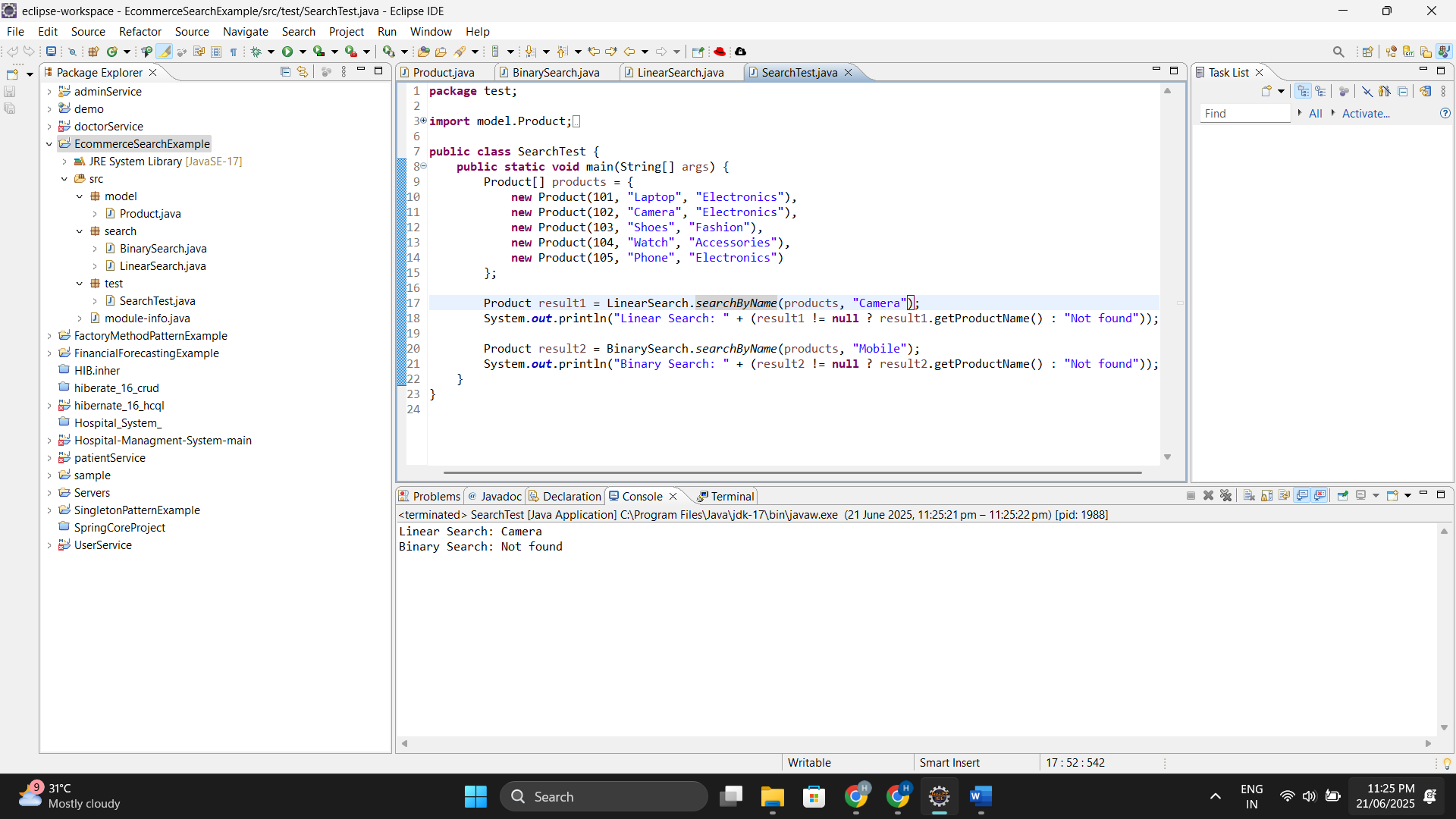
Product result2 = BinarySearch.searchByName(products, "Mobile");

System.out.println("Binary Search: " + (result2 != null ? result2.getProductName() : "Not found"));

}

}

**OUTPUT:**



**Exercise 7: Financial Forecasting**

**Financialforecasting.java**

package financialforecasting;

public class FinancialForecast {

public static double futureValue(double pv, double[] rates, int n) {

if (n == 0) {

return pv;

}

return (1 + rates[n - 1]) \* *futureValue*(pv, rates, n - 1);

}

public static void main(String[] args) {

double pv = 1000;

double[] rates = {0.05, 0.04, 0.06, 0.03, 0.05, 0.02, 0.04, 0.03, 0.05, 0.04};

int n = rates.length;

double fv = *futureValue*(pv, rates, n);

System.*out*.println("Future Value: " + fv);

}

}

**OUTPUT:**

