**MODULE 2: ALGORITHMS\_DATA STRUCTURES**

**Task 1 : E-commerce Platform Search Function**

**Code :**

**ProductSearch.java**

import java.util.Arrays;

import java.util.Comparator;

class Item {

int id;

String name;

String type;

public Item(int id, String name, String type) {

this.id = id;

this.name = name.trim();

this.type = type.trim();

}

@Override

public String toString() {

return "ID: " + id + " | Name: " + name + " | Category: " + type;

}

}

public class ProductSearch {

public static Item findUsingLinearSearch(Item[] items, String targetName) {

targetName = targetName.trim();

for (Item item : items) {

if (item.name.equalsIgnoreCase(targetName)) {

return item;

}

}

return null;

}

public static Item findUsingBinarySearch(Item[] items, String targetName) {

targetName = targetName.trim();

Arrays.sort(items, Comparator.comparing(i -> i.name.toLowerCase()));

int start = 0, end = items.length - 1;

while (start <= end) {

int mid = (start + end) / 2;

int result = targetName.compareToIgnoreCase(items[mid].name);

if (result == 0) {

return items[mid];

} else if (result < 0) {

end = mid - 1;

} else {

start = mid + 1;

}

}

return null;

}

public static void main(String[] args) {

Item[] inventory = {

new Item(201, "Gaming Keyboard", "Electronics"),

new Item(202, "Leather Wallet", "Accessories"),

new Item(203, "Bluetooth Speaker", "Electronics"),

new Item(204, "Office Desk", "Furniture"),

new Item(205, "Sports Watch", "Wearables")

};

String keyword = "Bluetooth Speaker";

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

Item linearResult = findUsingLinearSearch(inventory, keyword);

System.out.println(linearResult != null ? "Result: " + linearResult : "Item not found in linear search.");

System.out.println("\nSearch result using Binary Search:");

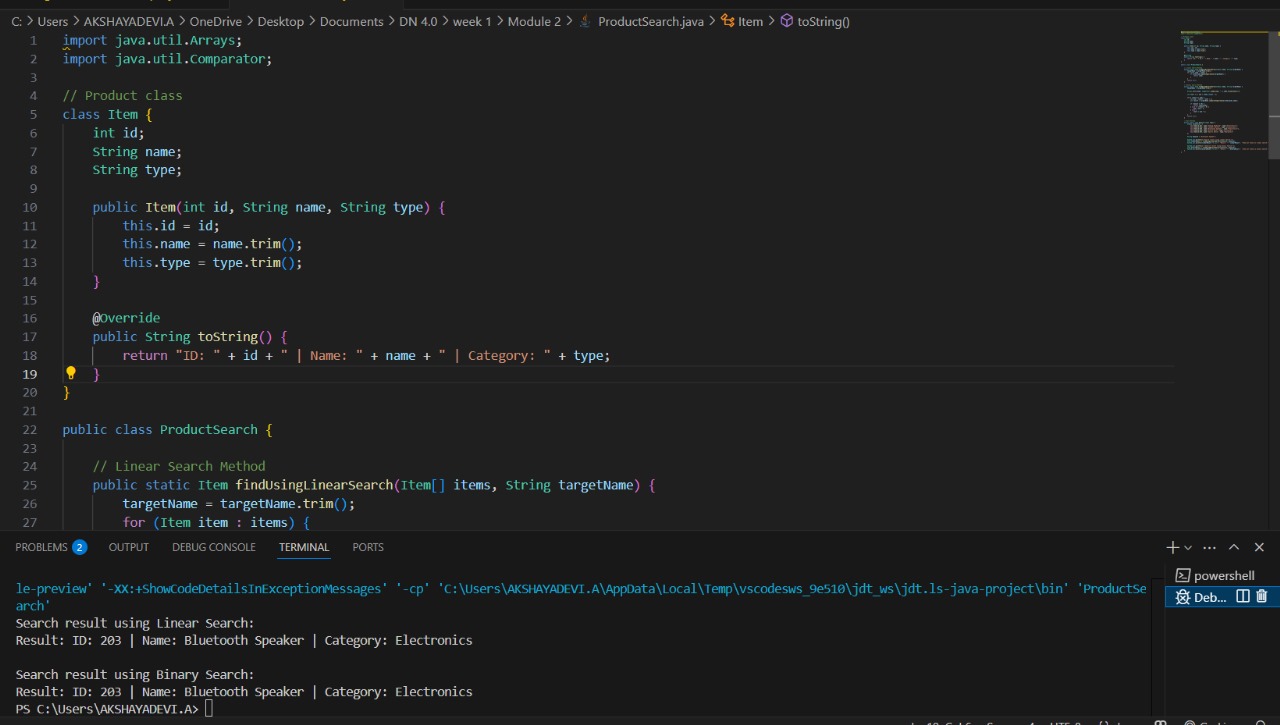
Item binaryResult = findUsingBinarySearch(inventory, keyword);

System.out.println(binaryResult != null ? "Result: " + binaryResult : "Item not found in binary search.");

}

}

**Output :**

****

**Task 2 : Financial Forecasting**

**Code:**

**FinancialForecast.java**

public class FinancialForecast {

public static double calculateFutureValue(double principal, double annualRate, int timePeriod) {

if (timePeriod == 0) {

return principal;

}

return calculateFutureValue(principal, annualRate, timePeriod - 1) \* (1 + annualRate);

}

public static void main(String[] args) {

double principalAmount = 10000.0;

double growthRate = 0.05;

int durationYears = 5;

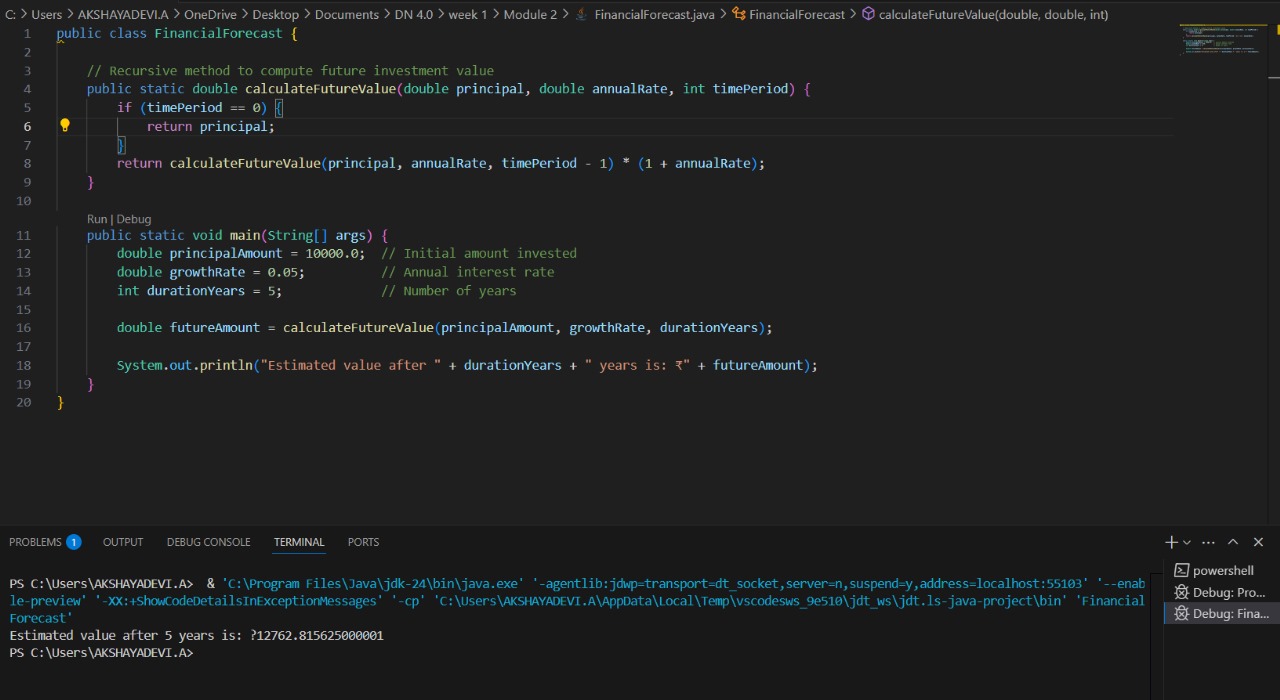
double futureAmount = calculateFutureValue(principalAmount, growthRate, durationYears);

System.out.println("Estimated value after " + durationYears + " years is: ₹" + futureAmount);

}

}

**Output:**

****