

Antwalk Software Engineering Foundations

Inheritance, Overloading, Overriding

Assignment 1

Name : Arijit Saha , College : Future Institute of Engineering and Management ,
Employee Id : T23010330 , Mail Id : arijits@trainee.nrifintech.com

Q1. Implement a class Calculator with the method mentioned below.

Method Description

findAverage()

Calculate the average of three numbers

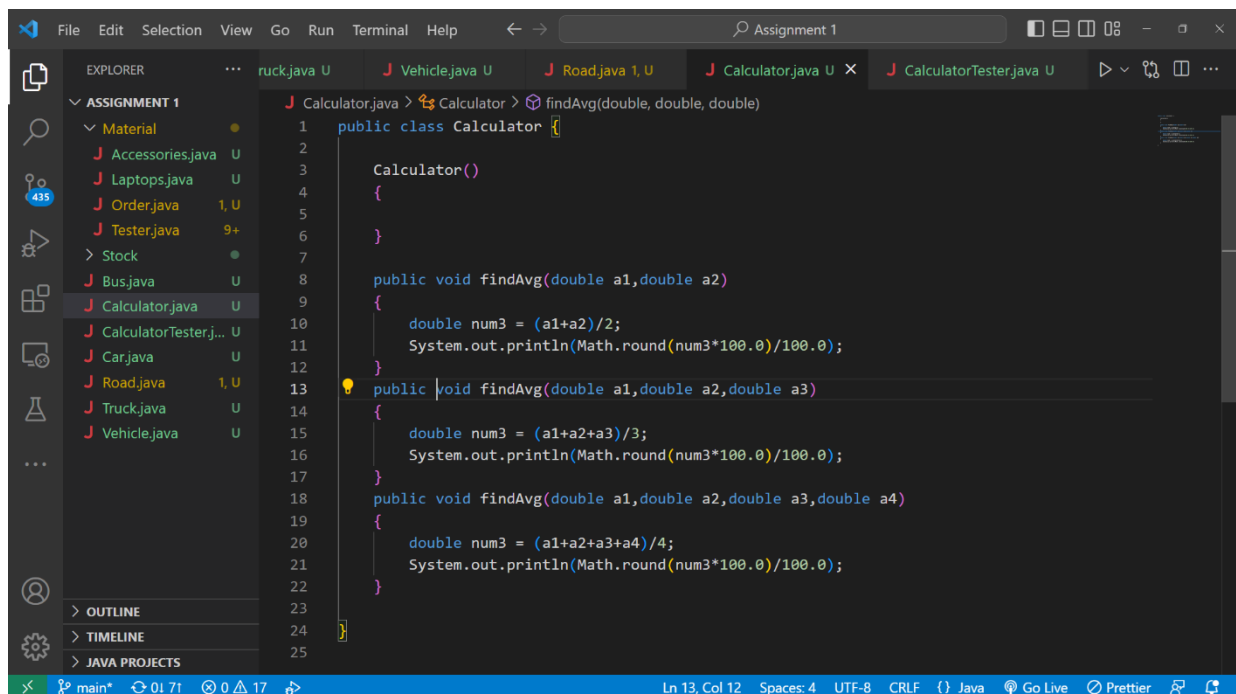
Calculate the average of four numbers

Calculate the average of five numbers

Return the average rounded off to two decimal digits

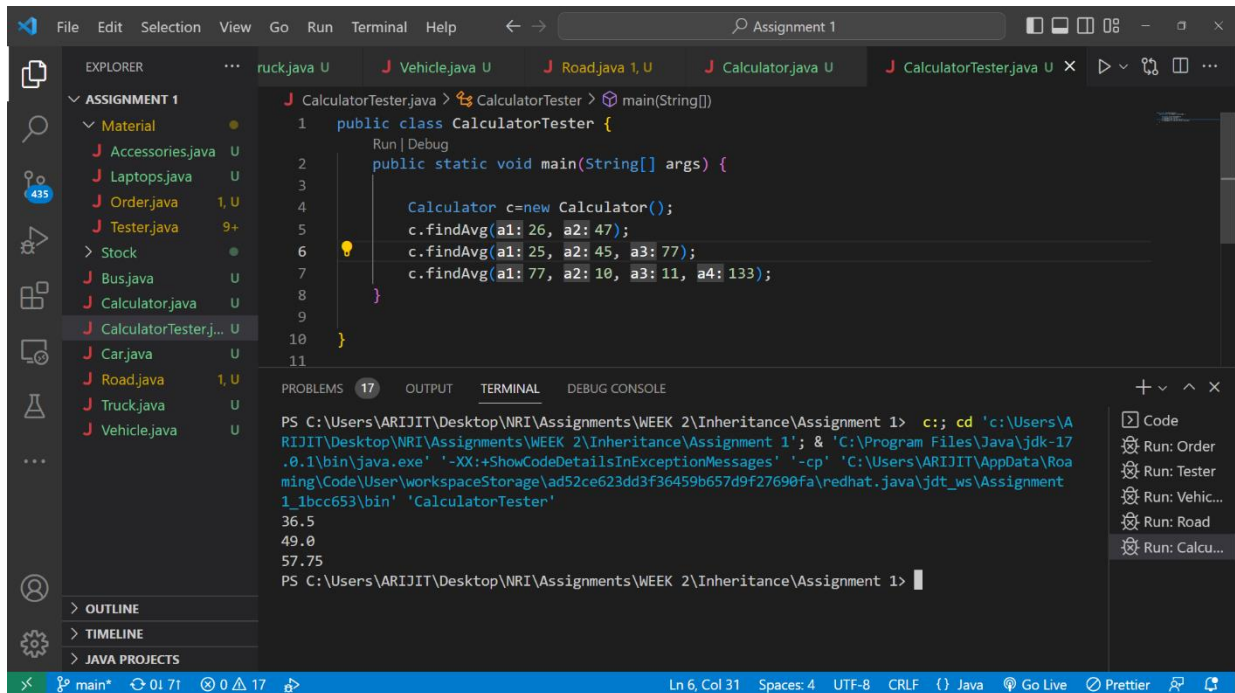
Test the functionalities using the provided Tester class.

Calculator.java



```
1 public class Calculator {
2
3     Calculator()
4     {
5
6     }
7
8     public void findAvg(double a1,double a2)
9     {
10         double num3 = (a1+a2)/2;
11         System.out.println(Math.round(num3*100.0)/100.0);
12     }
13     public void findAvg(double a1,double a2,double a3)
14     {
15         double num3 = (a1+a2+a3)/3;
16         System.out.println(Math.round(num3*100.0)/100.0);
17     }
18     public void findAvg(double a1,double a2,double a3,double a4)
19     {
20         double num3 = (a1+a2+a3+a4)/4;
21         System.out.println(Math.round(num3*100.0)/100.0);
22     }
23
24 }
25
```

CalculatorTester.java Along with Output



The screenshot shows an IDE with the following components:

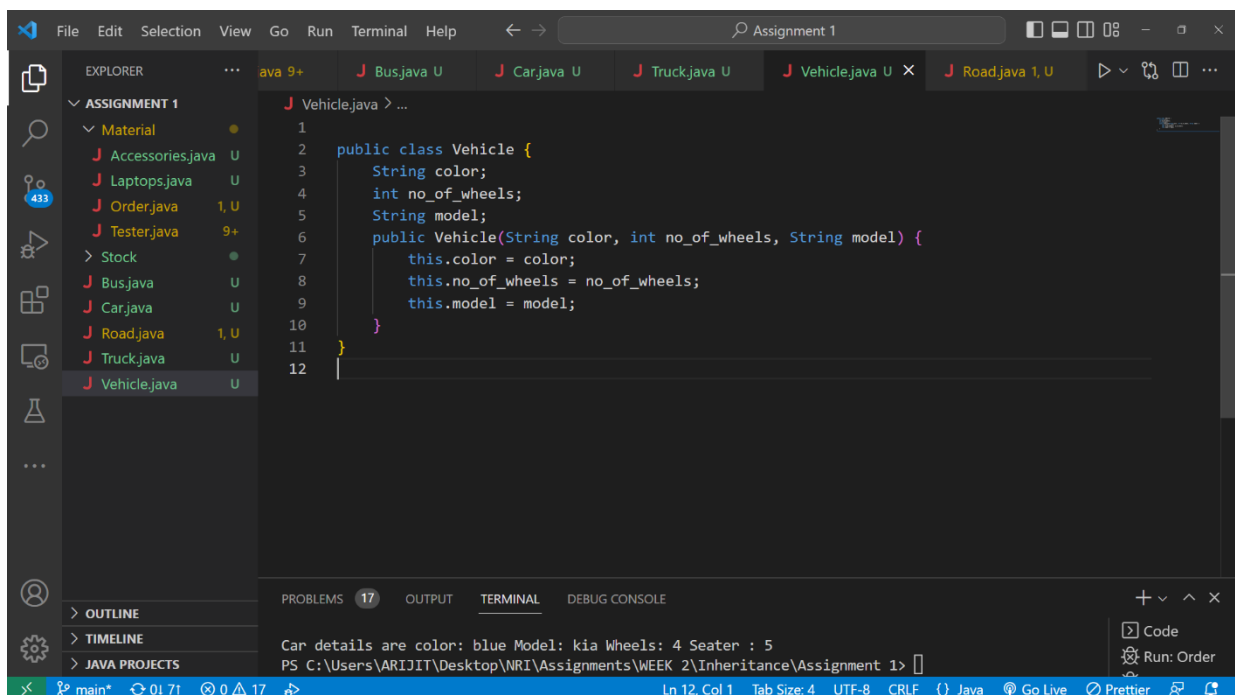
- EXPLORER:** A project named 'ASSIGNMENT 1' containing a 'Material' folder with files like Accessories.java, Laptops.java, Order.java, Tester.java, Stock, Bus.java, Calculator.java, CalculatorTester.java, Car.java, Road.java, Truck.java, and Vehicle.java.
- CalculatorTester.java:** The code defines a `CalculatorTester` class with a `main` method. It creates a `Calculator` object and calls `findAvg` with various arrays of numbers.
- TERMINAL:** Shows the command to run the program and the output of the `findAvg` method calls: 36.5, 49.0, and 57.75.

```
1 public class CalculatorTester {
2     public static void main(String[] args) {
3
4         Calculator c=new Calculator();
5         c.findAvg(a1: 26, a2: 47);
6         c.findAvg(a1: 25, a2: 45, a3: 77);
7         c.findAvg(a1: 77, a2: 10, a3: 11, a4: 133);
8     }
9 }
10
11
```

```
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1> c:: cd 'c:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1'; & 'C:\Program Files\Java\jdk-17.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\ARIJIT\AppData\Roaming\Code\User\workspaceStorage\ad52ce623dd3f36459b657d9f27690fa\redhat.java\jdt_ws\Assignment 1_1bcc653\bin' 'CalculatorTester'
36.5
49.0
57.75
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
```

Q2. Create a class called Vehicle. Create subclasses Truck, Bus and Car. Add common methods in the base class and specific methods in the corresponding subclasses. Create a class called Road and create objects for Truck, Bus and Car and display the appropriate messages. Also, in the Vehicle class constructor, initialise the variables colour, no of wheels and model. Give appropriate values for these variables from the invoking subclass.

Vehicle.java



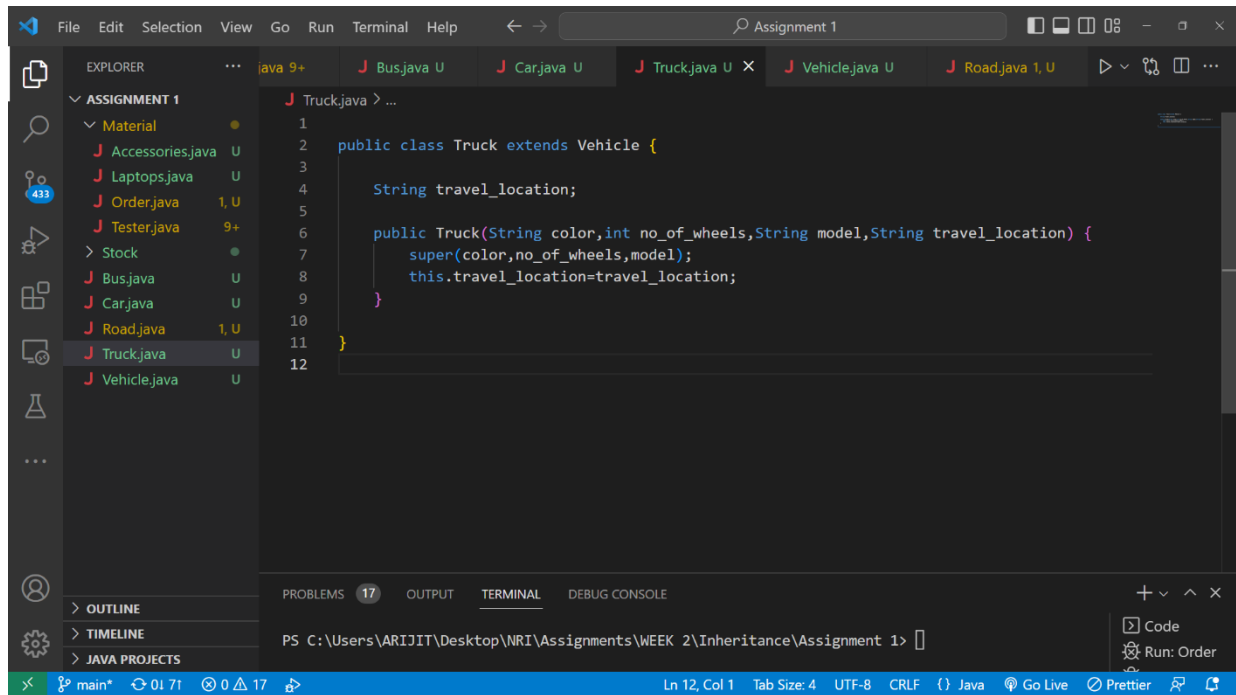
The screenshot shows an IDE with the following components:

- EXPLORER:** The same project structure as the first screenshot, with 'Vehicle.java' selected.
- Vehicle.java:** The code defines a `Vehicle` class with attributes `color`, `no_of_wheels`, and `model`. It includes a constructor to initialize these attributes.
- TERMINAL:** Shows the output of the program, displaying details for a Car object: 'Car details are color: blue Model: kia Wheels: 4 Seater : 5'.

```
1
2 public class Vehicle {
3     String color;
4     int no_of_wheels;
5     String model;
6     public Vehicle(String color, int no_of_wheels, String model) {
7         this.color = color;
8         this.no_of_wheels = no_of_wheels;
9         this.model = model;
10    }
11 }
12
```

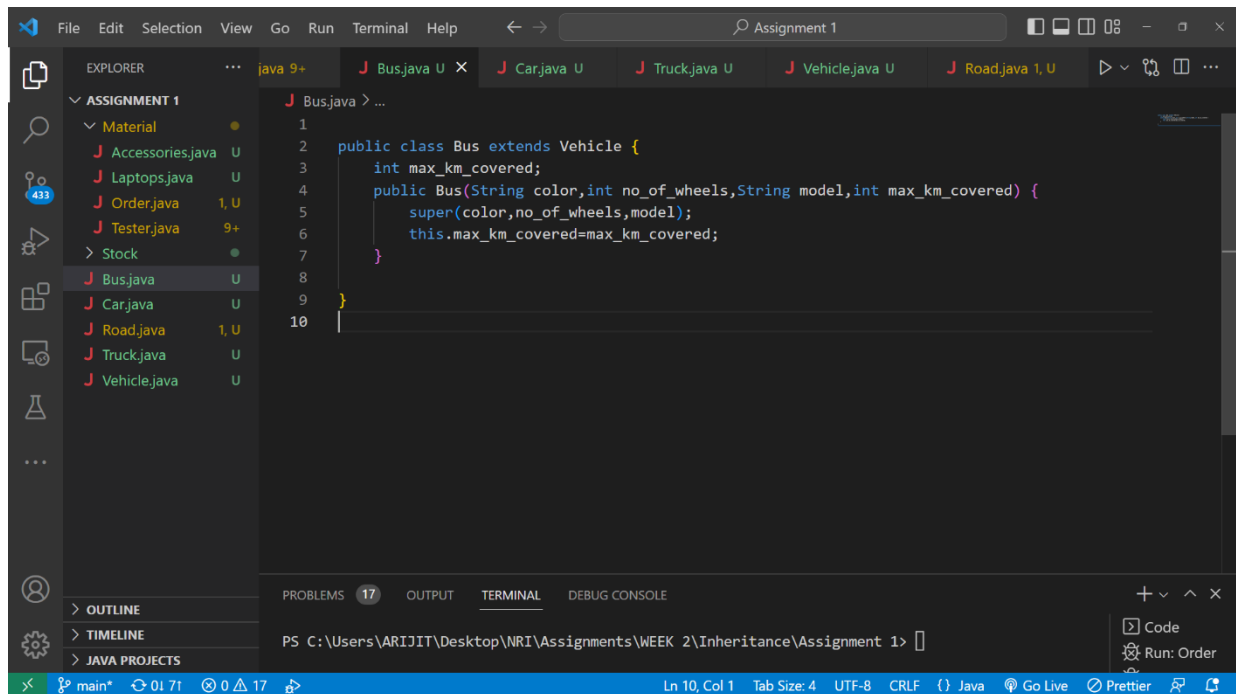
```
Car details are color: blue Model: kia Wheels: 4 Seater : 5
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
```

Truck.java



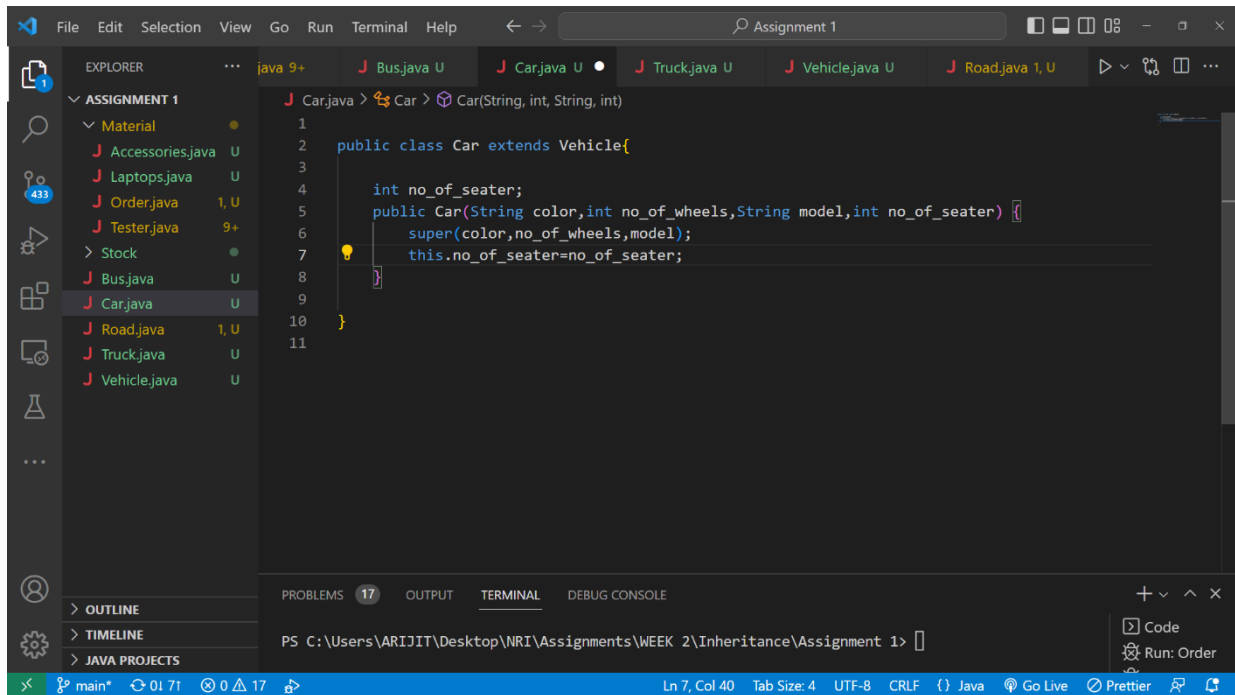
```
1 public class Truck extends Vehicle {
2
3     String travel_location;
4
5     public Truck(String color,int no_of_wheels,String model,String travel_location) {
6         super(color,no_of_wheels,model);
7         this.travel_location=travel_location;
8     }
9 }
10
11
12
```

Bus.java



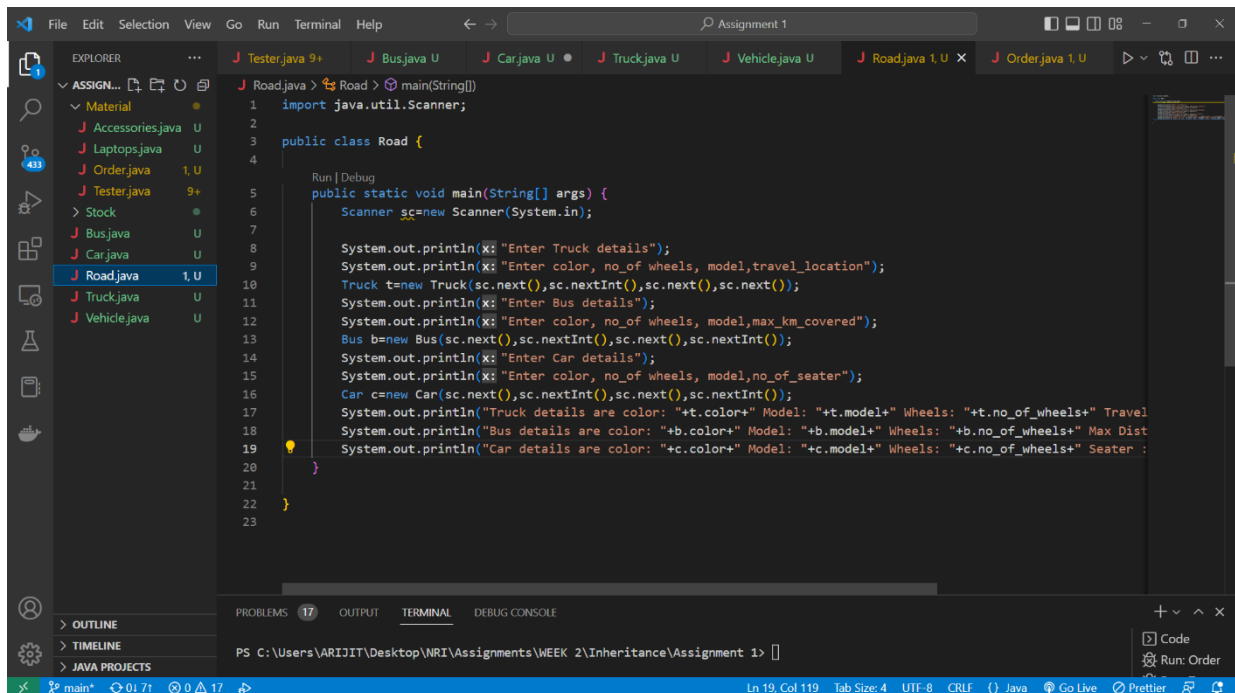
```
1 public class Bus extends Vehicle {
2     int max_km_covered;
3     public Bus(String color,int no_of_wheels,String model,int max_km_covered) {
4         super(color,no_of_wheels,model);
5         this.max_km_covered=max_km_covered;
6     }
7 }
8
9
10
```

Car.java



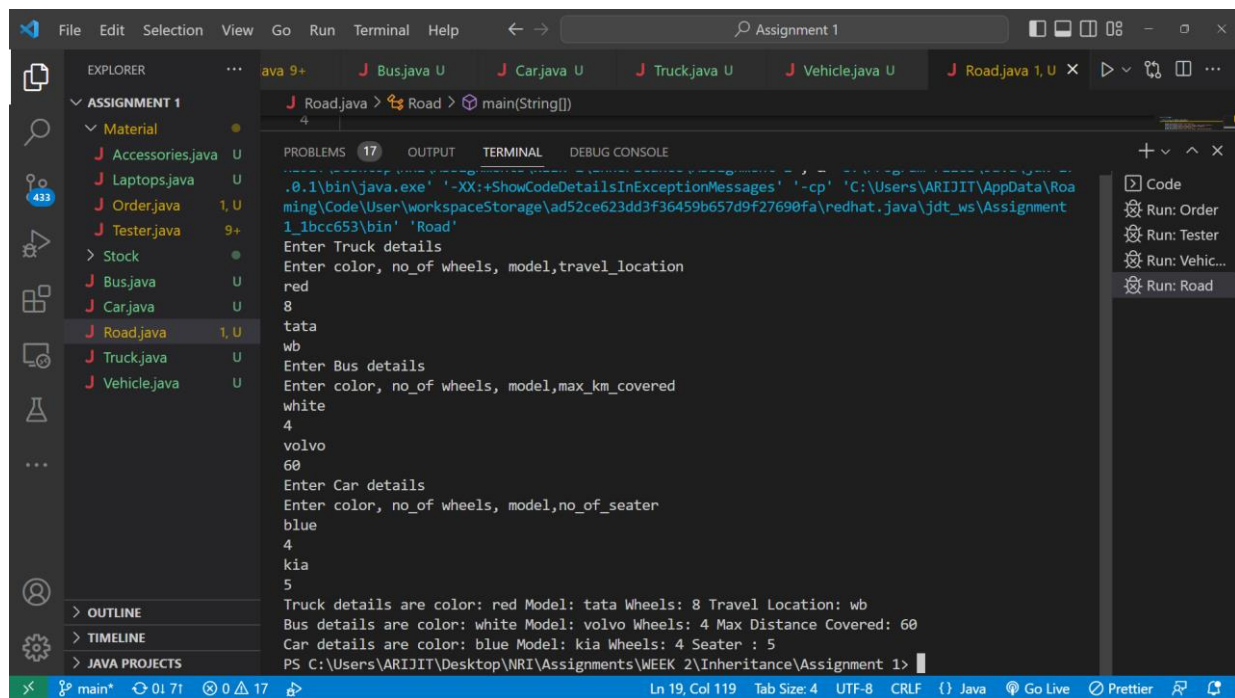
```
1 public class Car extends Vehicle{
2
3
4     int no_of_seater;
5     public Car(String color,int no_of_wheels,String model,int no_of_seater) {
6         super(color,no_of_wheels,model);
7         this.no_of_seater=no_of_seater;
8     }
9
10 }
11
```

Road.java



```
1 import java.util.Scanner;
2
3 public class Road {
4
5     public static void main(String[] args) {
6         Scanner sc=new Scanner(System.in);
7
8         System.out.println("Enter Truck details");
9         System.out.println("Enter color, no_of wheels, model,travel_location");
10        Truck t=new Truck(sc.next(),sc.nextInt(),sc.next(),sc.next());
11        System.out.println("Enter Bus details");
12        System.out.println("Enter color, no_of wheels, model,max_km_covered");
13        Bus b=new Bus(sc.next(),sc.nextInt(),sc.next(),sc.nextInt());
14        System.out.println("Enter Car details");
15        System.out.println("Enter color, no_of wheels, model,no_of_seater");
16        Car c=new Car(sc.next(),sc.nextInt(),sc.next(),sc.nextInt());
17        System.out.println("Truck details are color: "+t.color+" Model: "+t.model+" Wheels: "+t.no_of_wheels+" Travel
18        System.out.println("Bus details are color: "+b.color+" Model: "+b.model+" Wheels: "+b.no_of_wheels+" Max Dist
19        System.out.println("Car details are color: "+c.color+" Model: "+c.model+" Wheels: "+c.no_of_wheels+" Seater :
20
21    }
22 }
23
```

Output :



The screenshot shows an IDE window titled 'Assignment 1'. The Explorer panel on the left shows a project structure with 'ASSIGNMENT 1' containing a 'Material' package with 'Accessories.java', 'Laptops.java', 'Order.java', and 'Tester.java', and a 'Stock' package with 'Bus.java', 'Car.java', 'Road.java', 'Truck.java', and 'Vehicle.java'. The main editor shows the 'main(String[] args)' method in 'Road.java'. The terminal output shows the execution of the program, which prompts for truck, bus, and car details and then prints the details for each.

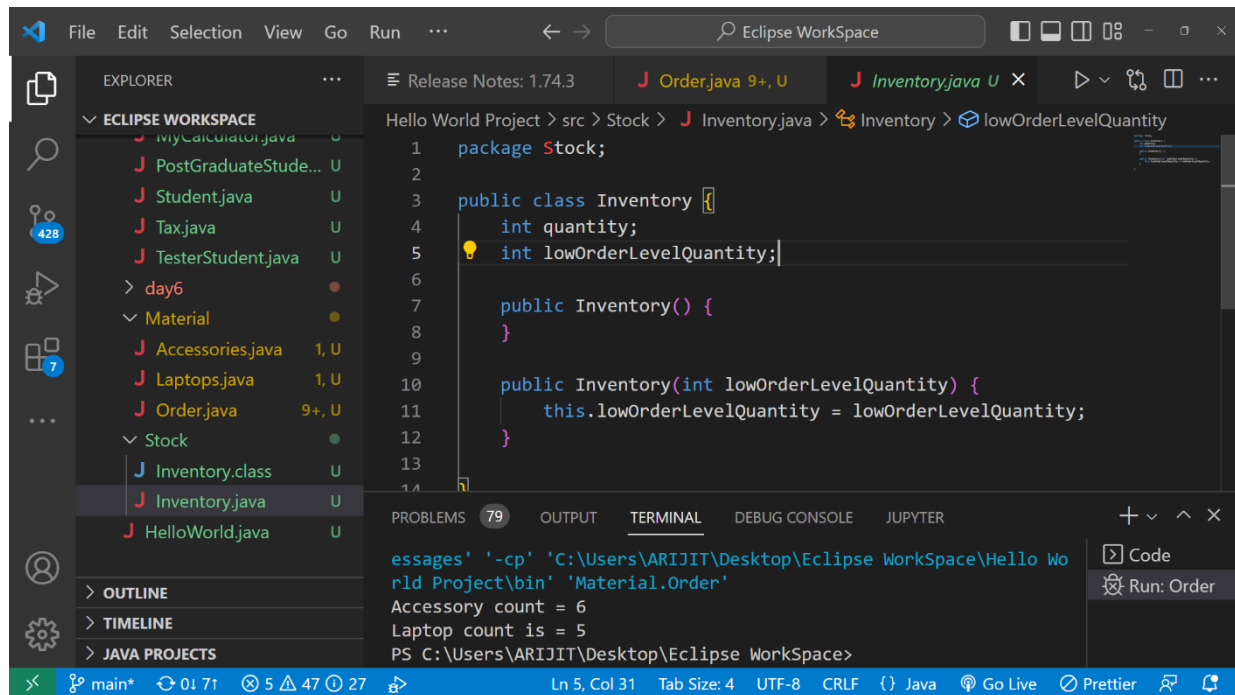
```
.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\ARIJIT\AppData\Roaming\Code\User\workspaceStorage\ad52ce623dd3f36459b657d9f27690fa\redhat.java\jdt_ws\Assignment 1_ibcc653\bin' 'Road'
Enter Truck details
Enter color, no_of wheels, model,travel_location
red
8
tata
wb
Enter Bus details
Enter color, no_of wheels, model,max_km_covered
white
4
volvo
60
Enter Car details
Enter color, no_of wheels, model,no_of_seater
blue
4
kia
5
Truck details are color: red Model: tata Wheels: 8 Travel Location: wb
Bus details are color: white Model: volvo Wheels: 4 Max Distance Covered: 60
Car details are color: blue Model: kia Wheels: 4 Seater : 5
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
```

Q3. Create a class called Inventory in a package stock. This class has data members called quantity and lowOrderLevelQuantity. Two classes that inherit from this class -Accessories and Laptops, are in package called material. The lowOrderLevelQuantity for laptops is 3, while lowOrderLevelQuantity for Accessories is 5. Apart from these members, Accessories and Laptops also have members describing them and a unique id.

Create 5 laptops and 10 Accessories objects. The quantity member must add up accordingly in the individual classes.

Create an Order class and have customers place orders. If the ordered quantity is available then Invoice should be generated. If the quantity is below lowOrderLevelQuantity then a RequestForMaterial (RFM) must be generated.

Inventory.java



```
package Stock;

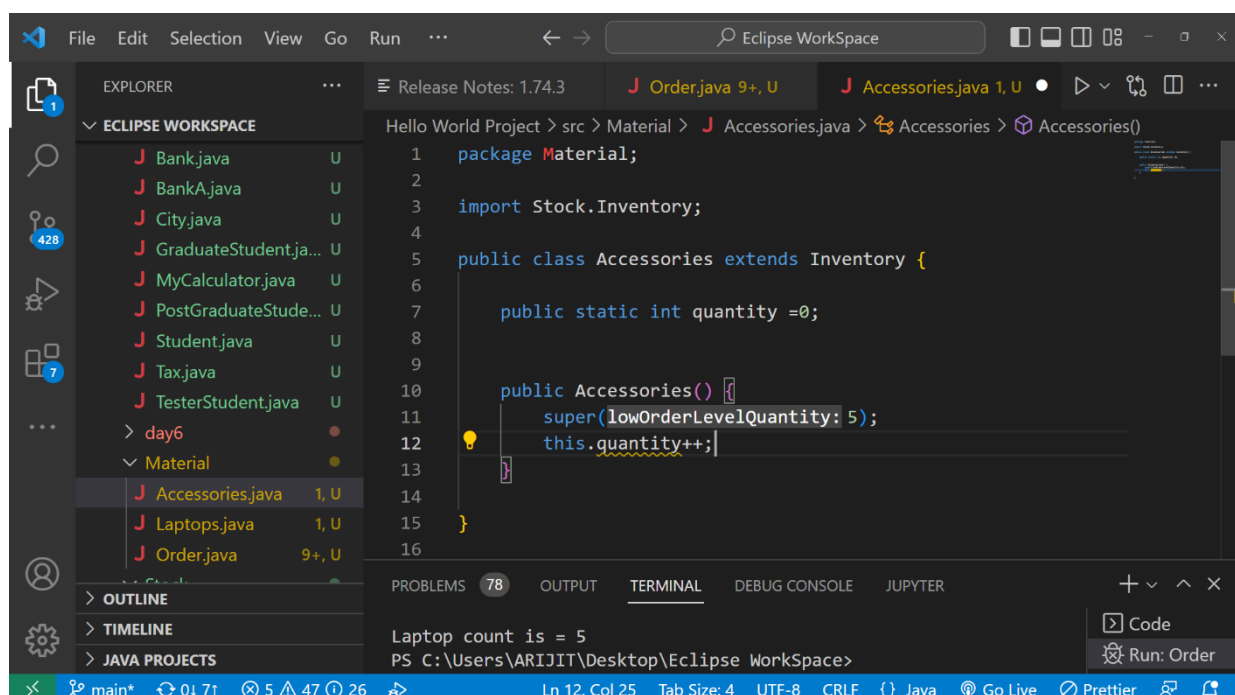
public class Inventory {
    int quantity;
    int lowOrderLevelQuantity;

    public Inventory() {
    }

    public Inventory(int lowOrderLevelQuantity) {
        this.lowOrderLevelQuantity = lowOrderLevelQuantity;
    }
}
```

main* 01:71 5 47 27 Ln 5, Col 31 Tab Size: 4 UTF-8 CRLF {} Java Go Live Prettier

Accessories.java



```
package Material;

import Stock.Inventory;

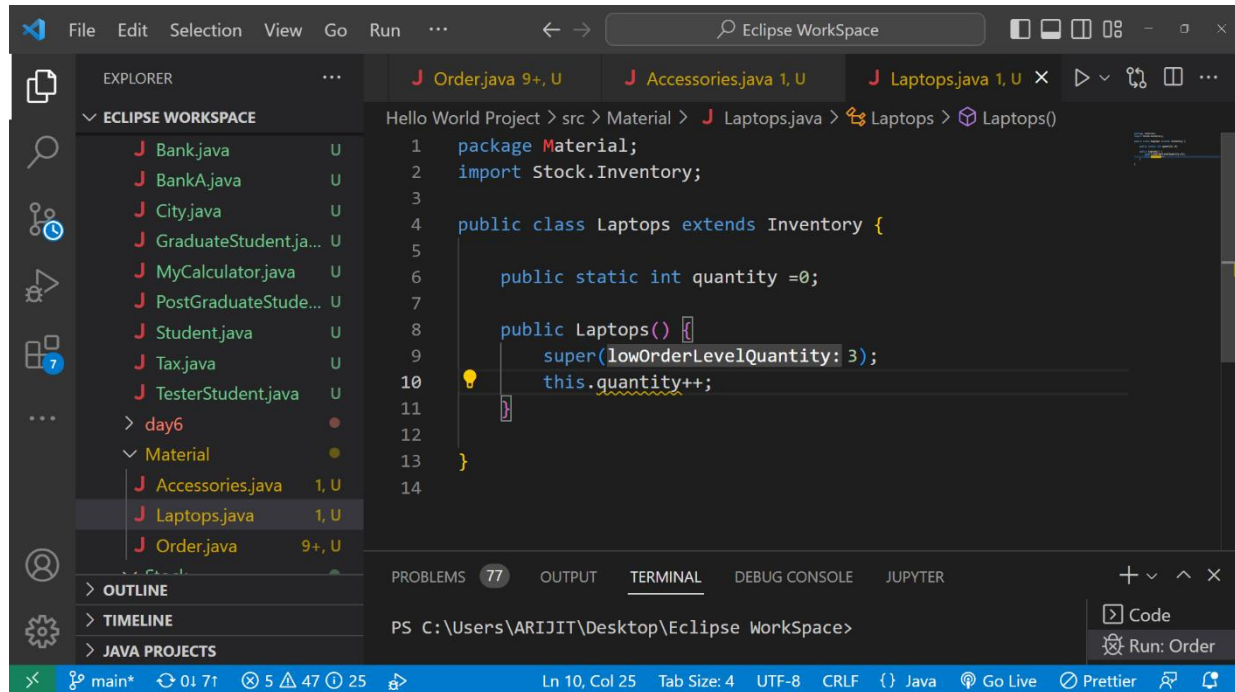
public class Accessories extends Inventory {

    public static int quantity = 0;

    public Accessories() {
        super(lowOrderLevelQuantity: 5);
        this.quantity++;
    }
}
```

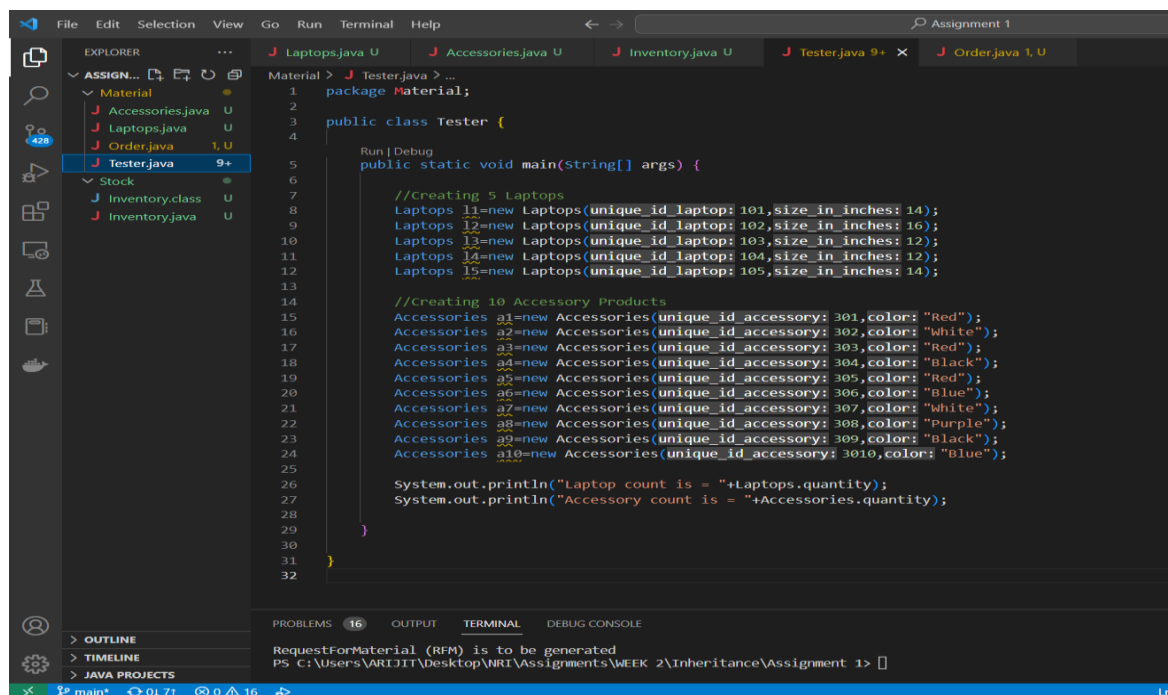
main* 01:71 5 47 26 Ln 12, Col 25 Tab Size: 4 UTF-8 CRLF {} Java Go Live Prettier

Laptop.java



```
1 package Material;
2 import Stock.Inventory;
3
4 public class Laptops extends Inventory {
5
6     public static int quantity =0;
7
8     public Laptops() {
9         super(lowOrderLevelQuantity: 3);
10        this.quantity++;
11    }
12
13 }
14
```

Tester.java (For showing count of 5 laptops & 10 accessories)



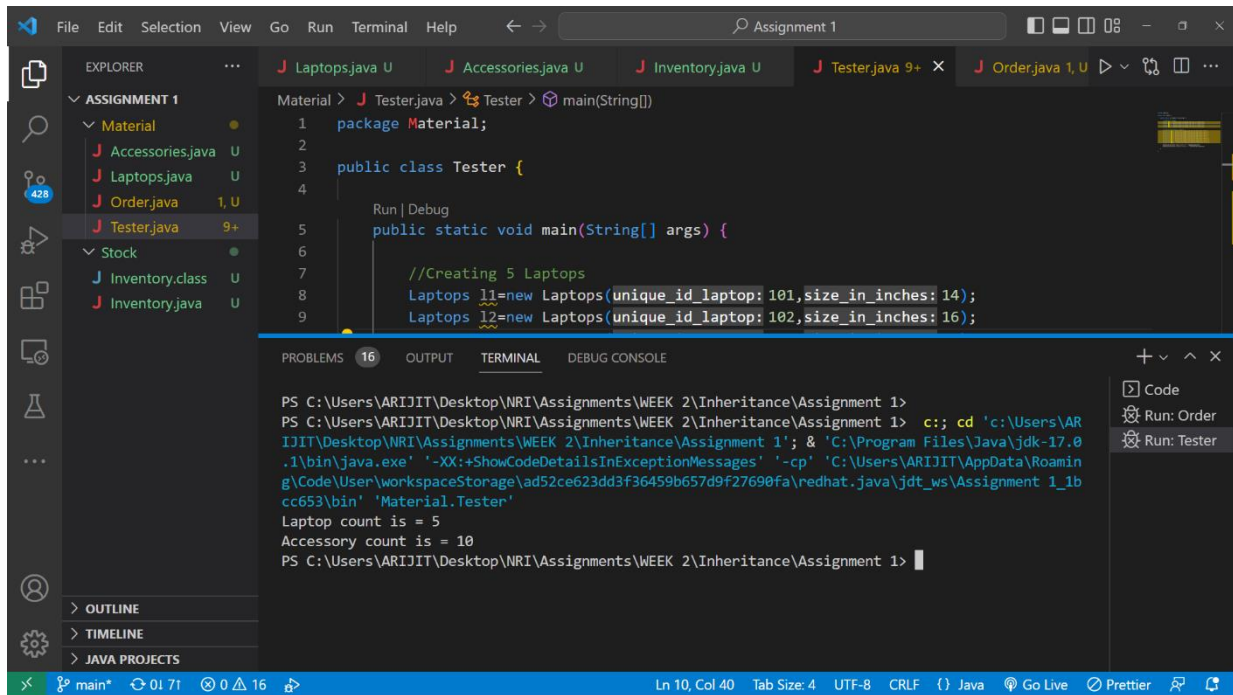
```
1 package Material;
2
3 public class Tester {
4
5     public static void main(String[] args) {
6
7         //Creating 5 Laptops
8         Laptops l1=new Laptops(unique_id_laptop: 101,size in inches: 14);
9         Laptops l2=new Laptops(unique_id_laptop: 102,size in inches: 16);
10        Laptops l3=new Laptops(unique_id_laptop: 103,size in inches: 12);
11        Laptops l4=new Laptops(unique_id_laptop: 104,size in inches: 12);
12        Laptops l5=new Laptops(unique_id_laptop: 105,size in inches: 14);
13
14        //Creating 10 Accessory Products
15        Accessories a1=new Accessories(unique_id_accessory: 301,color: "Red");
16        Accessories a2=new Accessories(unique_id_accessory: 302,color: "White");
17        Accessories a3=new Accessories(unique_id_accessory: 303,color: "Red");
18        Accessories a4=new Accessories(unique_id_accessory: 304,color: "Black");
19        Accessories a5=new Accessories(unique_id_accessory: 305,color: "Red");
20        Accessories a6=new Accessories(unique_id_accessory: 306,color: "Blue");
21        Accessories a7=new Accessories(unique_id_accessory: 307,color: "White");
22        Accessories a8=new Accessories(unique_id_accessory: 308,color: "Purple");
23        Accessories a9=new Accessories(unique_id_accessory: 309,color: "Black");
24        Accessories a10=new Accessories(unique_id_accessory: 3010,color: "Blue");
25
26        System.out.println("Laptop count is = "+Laptops.quantity);
27        System.out.println("Accessory count is = "+Accessories.quantity);
28
29    }
30
31 }
32
```


Order.java

```
1 package Material;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class Order {
7     public static void main(String[] args) {
8
9         Scanner sc=new Scanner(System.in);
10        ArrayList<Laptops> lap=new ArrayList<>();
11        ArrayList<Accessories> accs=new ArrayList<>();
12        System.out.println("Start Ordering for Laptops OR Accessories");
13
14        do{
15            System.out.println("What do you want to order ?");
16            System.out.println("Enter 1 for Laptop");
17            System.out.println("Enter 2 for Accessories");
18            System.out.println("Enter anything else ");
19            int ch=sc.nextInt();
20            switch(ch)
21            {
22                case 1:
23                    System.out.println("Enter the unique id ");
24                    int id_lap=sc.nextInt();
25                    System.out.println("Enter the Size of Laptop in Inches");
26                    int size=sc.nextInt();
27                    Laptops l1=new Laptops(id_lap,size);
28                    lap.add(l1);
29                    break;
30                case 2:
31                    System.out.println("Enter the unique id ");
32                    int id_acc=sc.nextInt();
33                    System.out.println("Enter the Color of Accessory");
34                    String color=sc.next();
35
36                    Accessories a1=new Accessories(id_acc,color);
37                    accs.add(a1);
38                    break;
39                case 0:
40                    System.exit(status: 0);
41                default:
42                    System.out.println("Wrong Input");
43            }
44            System.out.println("Do You Want to Continue ?");
45            System.out.println("If Yes I then type 'Yes' else type 'No' ");
46            while(sc.next().toString().equals(anObject: "Yes"));
47
48            System.out.println("Laptop count is = "+Laptops.quantity);
49            System.out.println("Accessory count is = "+Accessories.quantity);
50
51            if(Laptops.quantity> 3 || Accessories.quantity> 5)
52                System.out.println("RequestForMaterial (RFM) is to be generated");
53
54        }
55    }
56 }
57 }
```


Output:

Tester Output



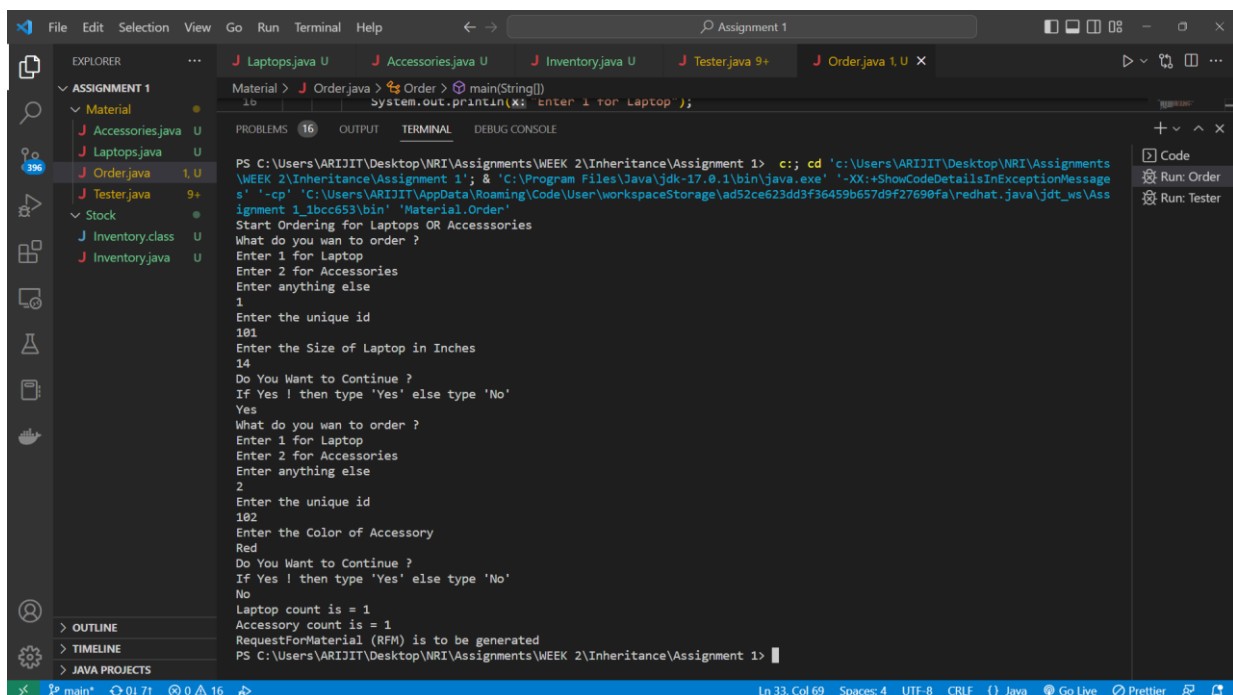
The screenshot shows the Visual Studio Code editor with the 'Tester.java' file open. The file is located in the 'Material' package and contains the following code:

```
1 package Material;
2
3 public class Tester {
4
5     Run | Debug
6     public static void main(String[] args) {
7
8         //Creating 5 Laptops
9         Laptops l1=new Laptops(unique_id_laptop:101,size_in_inches:14);
10        Laptops l2=new Laptops(unique_id_laptop:102,size_in_inches:16);
```

The terminal output shows the execution of the program:

```
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1> c;; cd 'c:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1'; & 'C:\Program Files\Java\jdk-17.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\ARIJIT\AppData\Roaming\Code\User\workspaceStorage\ad52ce623dd3f36459b657d9f27690fa\redhat.java\jdt_ws\Assignment 1_1bcc653\bin' 'Material.Tester'
Laptop count is = 5
Accessory count is = 10
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
```

Order output



The screenshot shows the Visual Studio Code editor with the 'Order.java' file open. The file is located in the 'Material' package and contains the following code:

```
1 package Material;
2
3 public class Order {
4     Run | Debug
5     public static void main(String[] args) {
6
7         System.out.println("Enter 1 for Laptop ");
```

The terminal output shows the execution of the program, which prompts the user for input:

```
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1> c;; cd 'c:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1'; & 'C:\Program Files\Java\jdk-17.0.1\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\ARIJIT\AppData\Roaming\Code\User\workspaceStorage\ad52ce623dd3f36459b657d9f27690fa\redhat.java\jdt_ws\Assignment 1_1bcc653\bin' 'Material.Order'
Start Ordering for Laptops OR Accessories
What do you want to order ?
Enter 1 for Laptop
Enter 2 for Accessories
Enter anything else
1
Enter the unique id
101
Enter the Size of Laptop in Inches
14
Do You Want to Continue ?
If Yes ! then type 'Yes' else type 'No'
Yes
What do you want to order ?
Enter 1 for Laptop
Enter 2 for Accessories
Enter anything else
2
Enter the unique id
102
Enter the Color of Accessory
Red
Do You Want to Continue ?
If Yes ! then type 'Yes' else type 'No'
No
Laptop count is = 1
Accessory count is = 1
RequestForMaterial (RFM) is to be generated
PS C:\Users\ARIJIT\Desktop\NRI\Assignments\WEEK 2\Inheritance\Assignment 1>
```