SHIFA KHAN
21BBS0255

JAVA PROGRAMMING
LA - 3

Q1 Hyundai car showroom sells the cars on a weekly basis. It has a sales module to perform the sale of a car. The functionality of the sales modules are:

Write a Java program to implement the concept of constructors for the below-mentioned problem.

a) The new sale order function performs the input activity like getting the details of the car (Model, Color, Engine Type, Fuel Type, Price).

b) The Authorize function checks the payment (Should be greater than 50 percent of the unit price) made by the user approves the sale order and calls the other function called order processing.

c) The order processing function checks the stock for the availability of the car and prints the delivery order(format given below) otherwise it will print the production order (format given below)

Delivery Order:

Owner Name	Car Model	Color	Fuel Type	Delivery date

Production Order:

Production branch Name	Showroom address	Car Model	Color	Engine type	Date of delivery

CODE:

(For Delivery order)

```
import java.util.Date;


class Car {
   String model;
   String color;
   String engineType;
   String fuelType;
   double price;


   public Car(String model, String color, String engineType, String fuelType, double price) {
      this.model = model;
      this.color = color;
      this.engineType = engineType;
      this.fuelType = fuelType;
      this.price = price;
```

```
    }
}

class SaleOrder {
    Car car;
    String ownerName;

    public SaleOrder(String ownerName, Car car) {
        this.ownerName = ownerName;
        this.car = car;
    }

    public void authorize(double payment) {
        if (payment >= 0.5 * car.price) {
            System.out.println("Payment authorized. Processing order...");
            orderProcessing();
        } else {
            System.out.println("Payment not sufficient. Order authorization failed.");
        }
    }

    private void orderProcessing() {
        boolean isStockAvailable = checkStockAvailability();

        if (isStockAvailable) {
            printDeliveryOrder();
        } else {
            printProductionOrder();
        }
    }

    private boolean checkStockAvailability() {
        return true;
    }

    private void printDeliveryOrder() {
```

```
        System.out.println("Delivery Order:");

System.out.println("_____
_____");
        System.out.printf(" %-15s | %-15s | %-10s | %-10s | %-15s \t\t     |\n",
                "Owner Name", "Car Model", "Color", "Fuel Type", "\tDelivery Date");

System.out.println("_____|_____|_____|_____|_____
_____|");
        System.out.printf(" %-16s| %-16s| %-11s|%-12s| %-17s  |\n",
                ownerName, car.model, car.color, car.fuelType, new Date());


    }


    private void printProductionOrder() {
        System.out.println("Production Order:");

System.out.println("_____
_____");
        System.out.printf("| %-25s | %-20s | %-14s | %-10s | %-15s | %-19s |\n",
                "Production Branch Name", "Showroom Address", "Car Model", "Color", "Engine Type", "Date of Delivery");

System.out.println("|_____|_____|_____|_____|_____
_____|");
        System.out.printf("| %-27s | %-22s | %-16s | %-11s | %-14s | %-20s |\n",
                "XYZ Branch", "ABC Address", car.model, car.color, car.engineType, new Date());


    }
}


public class CarSalesDemo {
    public static void main(String[] args) {
        Car car = new Car("Hyundai Sonata", "Blue", "V6", " Petrol", 25000.0);
        SaleOrder saleOrder = new SaleOrder("John Doe", car);
        saleOrder.authorize(0.6 * car.price);


         System.out.println("\n\nShifa Khan 21BBS0255....");
    }
}
```

(For Production order)

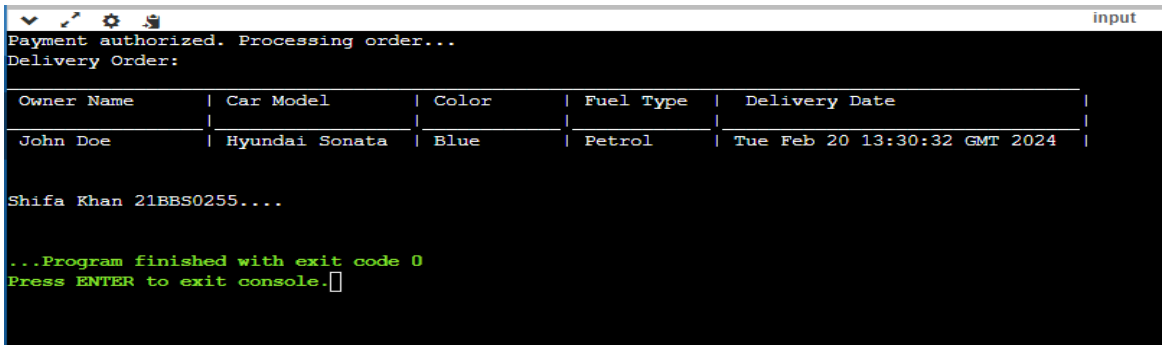
```
import java.util.Date;

class Car {
    String model;
    String color;
    String engineType;
    String fuelType;
    double price;

    public Car(String model, String color, String engineType, String fuelType, double price) {
        this.model = model;
        this.color = color;
        this.engineType = engineType;
        this.fuelType = fuelType;
        this.price = price;
    }
}

class SaleOrder {
    Car car;
    String ownerName;

    public SaleOrder(String ownerName, Car car) {
        this.ownerName = ownerName;
        this.car = car;
    }

    public void authorize(double payment) {
        if (payment >= 0.5 * car.price) {
            System.out.println("Payment authorized. Processing order...");
            orderProcessing();
        } else {
            System.out.println("Payment not sufficient. Order authorization failed.");
        }
```

```
    }

    private void orderProcessing() {
        boolean isStockAvailable = checkStockAvailability();

        if (isStockAvailable) {
            printDeliveryOrder();
        } else {
            printProductionOrder();
        }
    }

    private boolean checkStockAvailability() {
        return false;
    }

    private void printDeliveryOrder() {
        System.out.println("Delivery Order:");

System.out.println("_____
_____");
        System.out.printf(" %-15s | %-15s | %-10s | %-10s | %-15s \t\t     |\n",
                "Owner Name", "Car Model", "Color", "Fuel Type", "\tDelivery Date");

System.out.println("____________________|____________________|____________|____________|_____________________
__________|");
        System.out.printf(" %-16s| %-16s| %-11s|%-12s| %-17s  |\n",
                ownerName, car.model, car.color, car.fuelType, new Date());

    }

    private void printProductionOrder() {
        System.out.println("Production Order:");

System.out.println("_____
_____");
        System.out.printf("%-25s | %-20s | %-14s | %-10s | %-15s | %-19s |\n",
                "Production Branch Name", "Showroom Address", "Car Model", "Color", "Engine Type", "Date of Delivery");
```

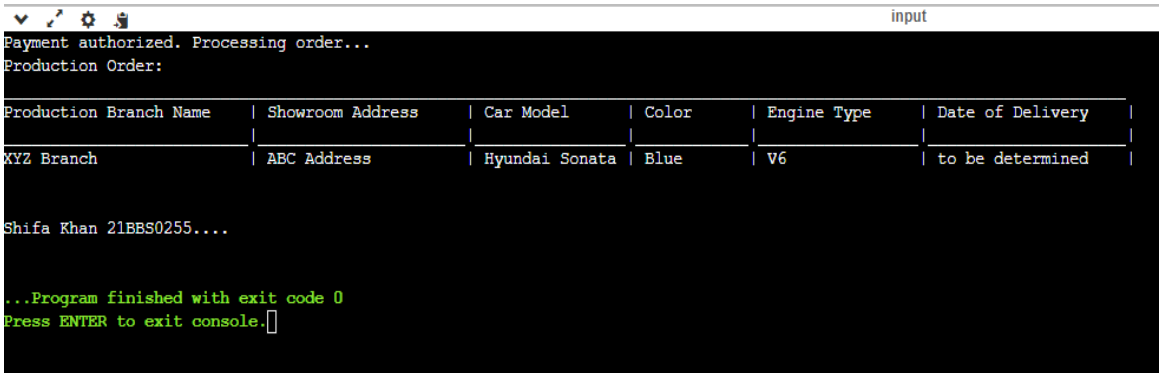
```
System.out.println("_____|_____|_____|_____|_____|_____|");
        System.out.printf("%-26s| %-21s| %-15s| %-11s| %-16s| %-20s|\n",
                "XYZ Branch", "ABC Address", car.model, car.color, car.engineType, "to be determined");


    }
}

public class CarSalesDemo {
    public static void main(String[] args) {
        Car car = new Car("Hyundai Sonata", "Blue", "V6", "Petrol", 25000.0);
        SaleOrder saleOrder = new SaleOrder("John Doe", car);
        saleOrder.authorize(0.6 * car.price);


         System.out.println("\n\nShifa Khan 21BBS0255....");
    }
}
```

OUTPUT:

(For Delivery order)

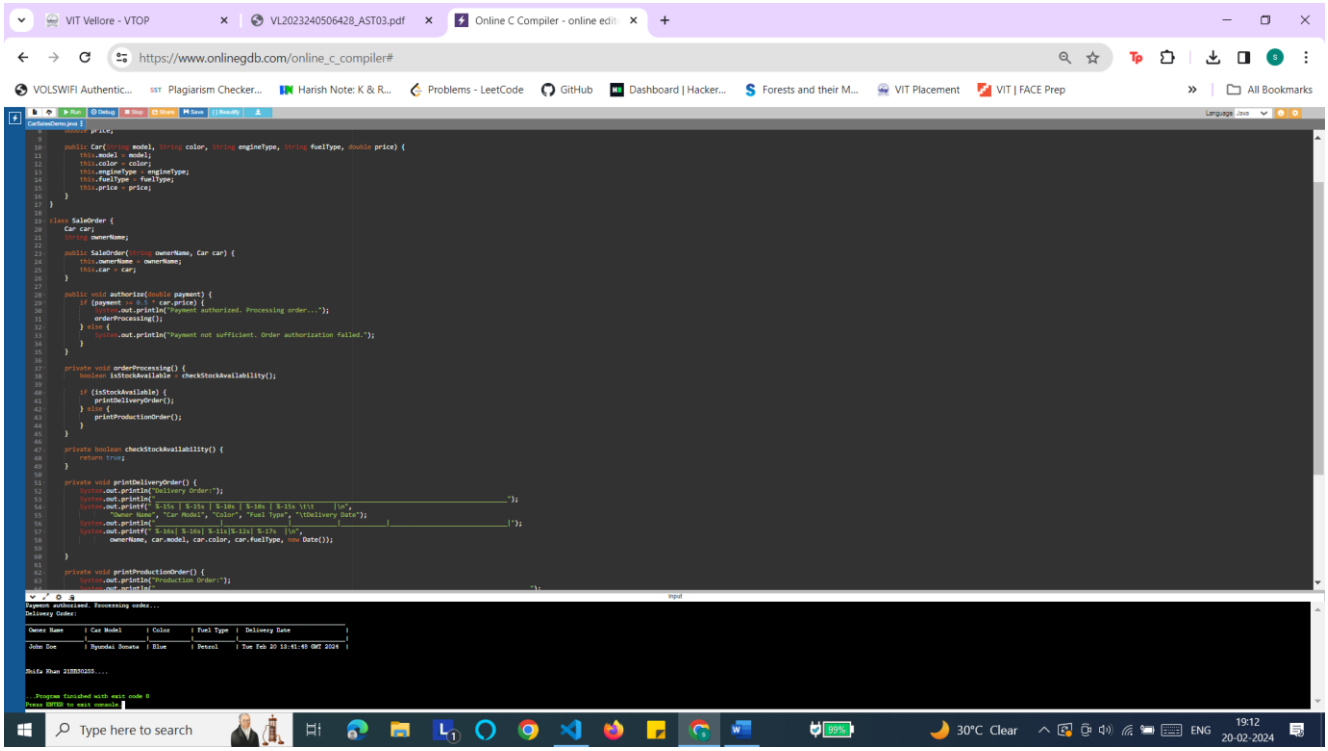


(For Production order)



SCREENSHOT:

(For Delivery order)



(For Production order)