```java
package ObjectsAndClassesExercise.CarSalesman;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.util.LinkedList;
import java.util.List;

public class Main {
    public static class Car {
        private String model;
        private Engine engine;
        private String weight;
        private String color;

        public Car(String model, Engine engine, String weight, String color) {
            this.model = model;
            this.engine = engine;
            this.weight = weight;
            this.color = color;
        }

        public String getModel() {
            return model;
        }

        public Engine getEngine() {
            return engine;
        }

        public String getWeight() {
            return weight;
        }

        public String getColor() {
            return color;
        }

        @Override
        public String toString() {
            Engine engine = this.getEngine();
            return String.format("%s:%n%s%n  Weight: %s%n  Color: %s",
                    model, engine.toString(), weight, color);
        }
    }

    public static class Engine {
        private String model;
        private int power;
        private String displacement;
        private String efficiency;

        public Engine(String model, int  power,String displacement, String efficency) {
            this.model = model;
            this.power = power;
            this.displacement = displacement;
            this.efficiency = efficiency;
        }
```

```java


        public String getModel() {
            return model;
        }

        public int getPower() {
            return power;
        }

        public String getDisplacement() {
            return displacement;
        }

        public String getEfficiency() {
            return efficiency;
        }

        @Override
        public String toString() {
            return String.format("  %s:%n    Power: %d%n    Displacement: %s%n    Efficiency: %s",
                    model,power,displacement,efficiency);
        }
    }
    public static void main(String[] args) throws IOException {
        BufferedReader reader = new BufferedReader(new InputStreamReader(System.in));

        int n = Integer.parseInt(reader.readLine());

        List<Car> carList = new LinkedList<>();
        List<Engine> engineList = new LinkedList<>();

        while (n-- > 0){
            String[] tokens = reader.readLine().split(" ");
            String model = tokens[0];
            int power = Integer.parseInt(tokens[1]);
            String displacement = "n/a";
            String efficency = "n/a";

            if (tokens.length == 4){
                displacement = tokens[2];
                efficency = tokens[3];
            } else if (tokens.length == 3){
                try {
                    int disp = Integer.parseInt(tokens[2]);
                    displacement = tokens[2];
                } catch (Exception e){
                    efficency = tokens[2];
                }
            }
            Engine engine = new Engine(model,power,displacement,efficency);
            engineList.add(engine);

        }
        int m = Integer.parseInt(reader.readLine());

        while (m-- > 0){
            String[] tokens = reader.readLine().split(" ");
```

```
            String model = tokens[0];
            String engineName = tokens[1];
            String weight = "n/a";
            String color = "n/a";


            if (tokens.length == 4){
                weight = tokens[2];
                color = tokens[3];
            } else if (tokens.length == 3){
                try {
                    int test = Integer.parseInt(tokens[2]);
                    weight = tokens[2];
                } catch (Exception e){
                    color = tokens[2];
                }
            }
            Engine currentEngine = engineList.stream().filter(x -> x.getModel().equals(engineName))
                    .findFirst().get();
            Car car = new Car(model,currentEngine,weight,color);
            carList.add(car);


        }

        for (Car car : carList){
            System.out.println(car.toString());
        }
    }

}
```