```java
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;

public class VehicleCatalogue {

    static class Vehicle {
        String type;
        String model;
        String colour;
        int horsePower;

        public Vehicle(String type, String model, String colour, int hourseHours) {
            this.type = type;
            this.model = model;
            this.colour = colour;
            this.horsePower = hourseHours;
        }

        public String getType() {
            return type;

        }

        public String getModel() {
            return model;
        }

        public String getColour() {
            return colour;
        }

        public int getHorsePower() {
            return horsePower;
        }

        @Override
        public String toString() {
            return String.format("Type: %s\nModel: %s\nColor: %s\nHorsepower: %d",
                    getType().toUpperCase().charAt(0) + getType().substring(1), this.model, this.colour,
```

```
41 this.horsePower);
42         }
43     }
44
45     public static void main(String[] args) {
46         Scanner scanner = new Scanner(System.in);
47
48         List<Vehicle> vehicleList = new ArrayList
   <>();
49         String input = scanner.nextLine();
50         while (!"End".equals(input)) {
51             String[] specification = input.split(" "
   );
52             String type = specification[0];
53             String model = specification[1];
54             String colour = specification[2];
55             int horsePower = Integer.parseInt(
   specification[3]);
56
57             Vehicle vehicle = new Vehicle(type, model
   , colour, horsePower);
58             vehicleList.add(vehicle);
59
60
61             input = scanner.nextLine();
62         }
63
64         input = scanner.nextLine();
65         while (!"Close the Catalogue".equals(input
   )) {
66             String model1 = input;
67
68             vehicleList.stream()
69                     .filter(vehicle -> vehicle.
   getModel().equals(model1))
70                     .forEach(vehicle -> System.out.
   println(vehicle.toString()));
71
72
73             input = scanner.nextLine();
74         }
75         List<Vehicle> cars = vehicleList.stream()
76                 .filter(vehicle -> vehicle.getType()
77                         .equals("car")).collect(
```

```
77 Collectors.toList());
78
79         List<Vehicle> trucks = vehicleList.stream()
80                 .filter(vehicle -> vehicle.getType()
81                         .equals("truck")).collect(
   Collectors.toList());
82
83         double carsAvgHp = avgHp(cars);
84         double trucksAvgHp = avgHp(trucks);
85
86         System.out.printf("Cars have average
   horsepower of: %.2f.\n", carsAvgHp);
87         System.out.printf("Trucks have average
   horsepower of: %.2f.", trucksAvgHp);
88     }
89
90     public  static double avgHp(List<Vehicle>
   vehicles) {
91         if(vehicles.size() == 0) {
92             return 0.0;
93         }
94         return vehicles.stream().mapToDouble(Vehicle
   ::getHorsePower).sum() / vehicles.size();
95     }
96 }
97
```