

Autoren: Marius Birk
Pieter Vogt
Tutor: Florian Brandt

Abgabe: 26.05.2020, 12:00 Uhr

Smileys:

A1	A2	A3	Σ

Objektorientierte Modellierung und Programmierung

Abgabe Übungsblatt Nr.05

(Alle allgemeinen Definitionen aus der Vorlesung haben in diesem Dokument Bestand, es sei denn sie erhalten eine explizit andere Definition.)

Aufgabe 1

```
1 %%hier dein code
```

Aufgabe 2

Bill.java

```
1 import java.util.ArrayList;
2
3 public class Bill {
4
5     //fields
6
7     String name;
8     double billPrice = 0;
9     ArrayList<BillItem> items = new ArrayList<>();
10
11     //methods
12
13     public void add(CarPart part) {
14         items.add(new BillItem(part));
15     }
16
17     //getter - setter
18
19     public double getTotalPrice() {
20         return billPrice;
21     }
22
23     public String toString() {
24         StringBuffer tempString = new StringBuffer("Receipt for
25             Bill: ");
26         double receiptTotal = 0;
27         tempString.append(this.name);
28         tempString.append("\n");
```

```
28     for (int i = 0; i < items.size(); i++) {
29         tempString.append(items.get(i).item.getName()); //add
           ItemName
30         tempString.append("\t");
31         tempString.append(items.get(i).item.getPrice()); //
           add ItemPrice
32         tempString.append("\n");
33         receipTotal = receipTotal + items.get(i).item.
           getPrice();
34     }
35     tempString.append("\n");
36     Math.nextUp(receipTotal); //doesn't work for some reason
       .
37     tempString.append("In Total this receipt is: " +
           receipTotal);
38     String output = tempString.toString();
39     return output;
40 }
41
42 //constructors
43
44 public Bill(String name) {
45     this.name = name;
46 }
47
48 //nested classes
49
50 private class BillItem {
51
52     //fields
53
54     CarPart item;
55
56     //methods
57
58     //getter - setter
59
60     public CarPart getItem() {
61         return item;
62     }
63
64     public void setItem(CarPart item) {
65         this.item = item;
66     }
67
68     public BillItem(CarPart item) {
69         this.item = item;
70     }
71 }
```

```
72
73 }
```

Car.java

```
1 import java.util.ArrayList;
2
3 public class Car {
4     ArrayList<CarPart> parts = new ArrayList<>();
5 }
```

CarComponent.java

```
1 public interface CarComponent {
2     public String getName();
3 }
```

CarPart.java

```
1 public class CarPart implements CarComponent {
2     String name;
3     double price;
4
5     @Override
6     public String getName() {
7         return null;
8     }
9
10    public double getPrice() {
11        return price;
12    }
13
14    public static class Seat extends CarPart {
15        String name = new String("Seat");
16        double price = 2000.0;
17
18        @Override
19        public String getName() {
20            return name;
21        }
22
23        public double getPrice() {
24            return price;
25        }
26    }
27
```

```
28     public static class Wheel extends CarPart {
29         String name = new String("Wheel");
30
31         double price = 1000.0;
32
33         @Override
34         public String getName() {
35             return name;
36         }
37
38         public double getPrice() {
39             return price;
40         }
41     }
42
43     public static class Motor extends CarPart {
44         String name = new String("Motor");
45
46         double price = 100000;
47
48         @Override
49         public String getName() {
50             return name;
51         }
52
53         public double getPrice() {
54             return price;
55         }
56     }
57 }
```

Main.java

```
1 public class Main {
2     public static void main(String[] args) {
3         Bill bill = new Bill("Rolls Royce");
4         bill.add(new CarPart.Motor());
5         bill.add(new CarPart.Seat());
6         bill.add(new CarPart.Wheel());
7         bill.add(new CarPart.Wheel());
8         bill.add(new CarPart.Wheel());
9         bill.add(new CarPart.Wheel());
10        System.out.println(bill.toString());
11    }
12 }
```

Aufgabe 3

```
1 %%hier dein code
```