

Week 4: Decorator

Minh-Triet Diep - Lars Jaeqx - T61

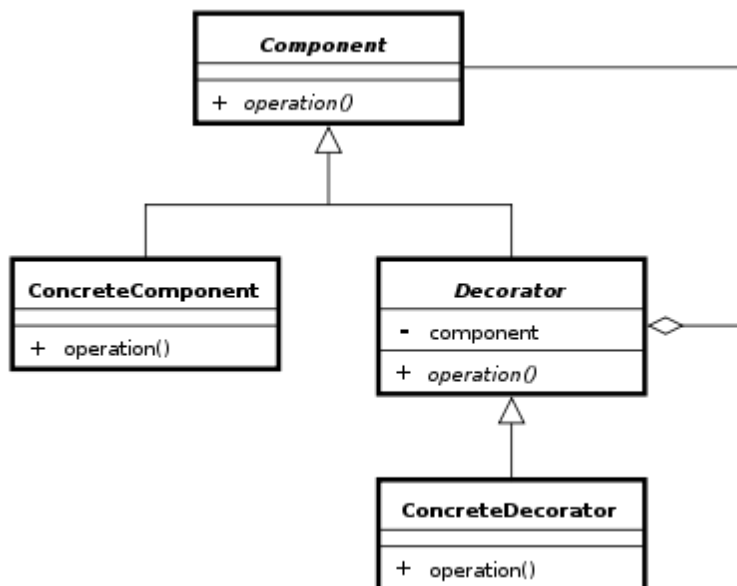
Het patroon

Het decorator patroon zorgt ervoor dat je verschillende klassen kunt maken met verschillende opties zonder dat deze opties voor elke klasse opnieuw moeten worden aangemaakt. Je zorgt ervoor dat de klassen die de opties bevatten de hoofdklasse uitbreidt en hierdoor extra functionaliteit toevoegt.

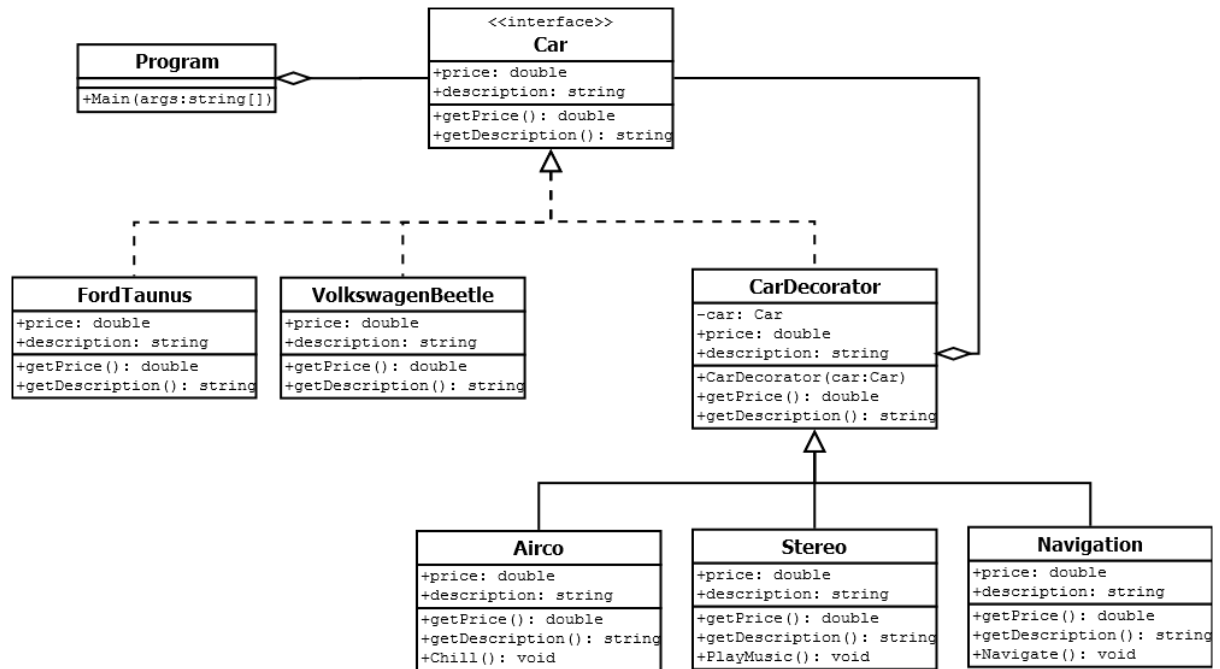
Om dit te bereiken dien je een decorator klasse aan te maken die de parent meekrijgt. Deze decorator klasse wordt weer uitgebreid door de onderliggende klassen. Hierdoor krijgt het een flexibele en uitgebreide functionaliteit zonder dit voor elke hoofdklasse opnieuw aangemaakt hoeft te worden.

UML

Hieronder de UML-diagrammen van het decorator patroon:



Originele UML-diagram van het decorator pattern



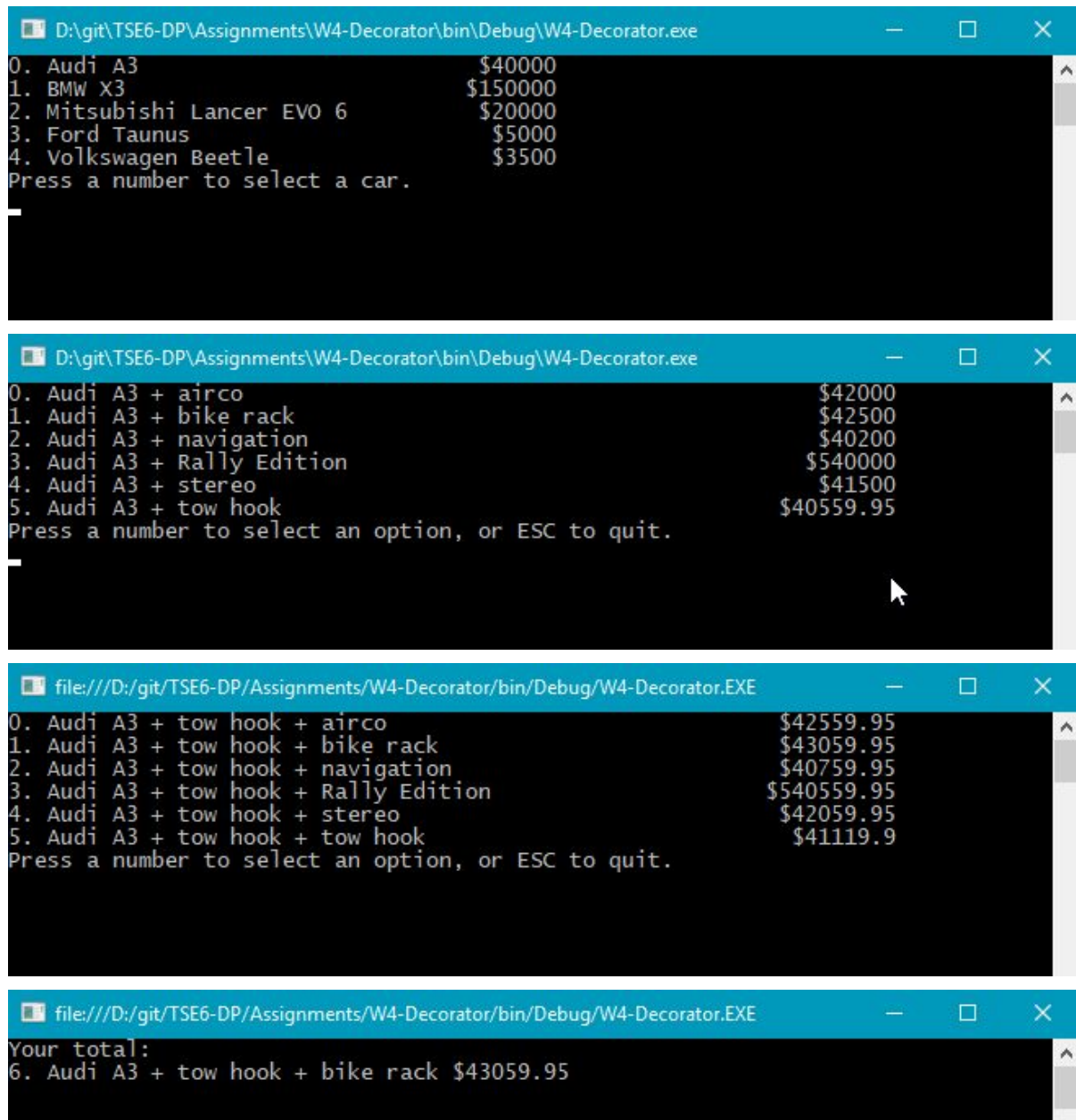
UML-diagram van de implementatie voor de opdracht.

Vergeleken met de echte implementatie zijn klassen weggelaten. Dit is voor het overzicht, want de rest van de klassen zijn identiek aan de afgebeelde klassen.

Test run results

Om het patroon te demonstreren hebben we een simpele autodealer geïmplementeerd. De gebruiker kan hier een baseline auto selecteren, om er daarna opties aan toe te voegen. Bij het afrekenen wordt de totaalkeuze afgebeeld.

Om het simpel te houden hebben we geen duplicatiecheck ingebouwd.



```
D:\git\TSE6-DP\Assignments\W4-Decorator\bin\Debug\W4-Decorator.exe
0. Audi A3 $40000
1. BMW X3 $150000
2. Mitsubishi Lancer EVO 6 $20000
3. Ford Taunus $5000
4. Volkswagen Beetle $3500
Press a number to select a car.

```

```
D:\git\TSE6-DP\Assignments\W4-Decorator\bin\Debug\W4-Decorator.exe
0. Audi A3 + airco $42000
1. Audi A3 + bike rack $42500
2. Audi A3 + navigation $40200
3. Audi A3 + Rally Edition $540000
4. Audi A3 + stereo $41500
5. Audi A3 + tow hook $40559.95
Press a number to select an option, or ESC to quit.

```

```
file:///D:/git/TSE6-DP/Assignments/W4-Decorator/bin/Debug/W4-Decorator.EXE
0. Audi A3 + tow hook + airco $42559.95
1. Audi A3 + tow hook + bike rack $43059.95
2. Audi A3 + tow hook + navigation $40759.95
3. Audi A3 + tow hook + Rally Edition $540559.95
4. Audi A3 + tow hook + stereo $42059.95
5. Audi A3 + tow hook + tow hook $41119.9
Press a number to select an option, or ESC to quit.

```

```
file:///D:/git/TSE6-DP/Assignments/W4-Decorator/bin/Debug/W4-Decorator.EXE
Your total:
6. Audi A3 + tow hook + bike rack $43059.95

```

Voor en nadelen

Reusability

- + De decorator klassen kunnen voor elke concrete klasse worden gebruikt.

Maintainability

- Er zijn heel veel kleine klassen, hierdoor kan het zoeken naar de juiste wat langer duren.

Extensibility

- + Je kunt onbeperkt nieuwe decorators toevoegen.
- + Er kunnen nieuwe concrete klassen worden gemaakt die alle decorators kan gebruiken..
- Om een kleine functionaliteit aan te maken dient er een nieuwe klasse aangemaakt te worden.

Reflectie

Dit patroon was eenvoudig om te implementeren, maar door de C#-taal is er tijd verloren gegaan aan het de keywords. Het keyword “virtual” bij de abstracte klasse CarDecorator ontbrak, en bij de concrete decorator klassen ontbrak het woord “override”, waardoor niet de goede methode werd aangeroepen. Nadat dit bekend was, werd het implementeren van het patroon geen probleem.

Wat opviel is dat het toevoegen van beide een nieuwe auto of een nieuwe decorator, en het aanroepen hiervan makkelijk was. Ook is het makkelijk om over lijsten van deze items te gaan, zoals wordt gedemonstreerd in de applicatie.