

HO GENT

H7 Pijlers van OO - Oefeningen

Table of Contents

1. Doelstellingen	1
2. Oefeningen.....	1
2.1. Spel met voorwerpen.....	1
2.2. Stageverplaatsingen.....	3
2.3. Spel met voorwerpen uitgebreid	5

1. Doelstellingen

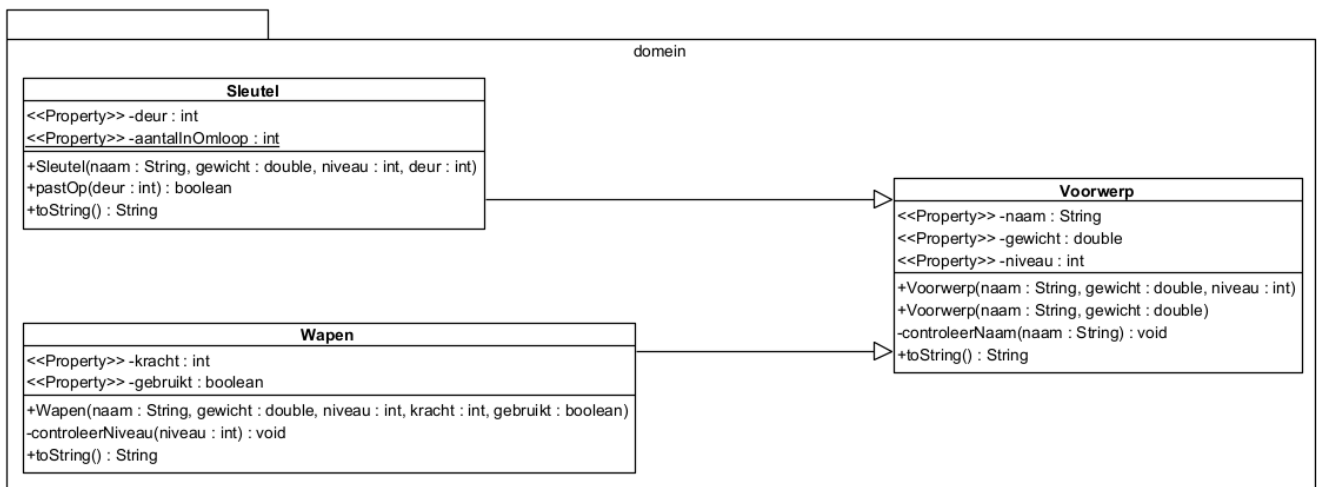
- Oefening 2.1 en 2.2: implementatie overerving inoefenen
- Oefening 2.3: implementatie polymorfisme inoefenen

2. Oefeningen

2.1. Spel met voorwerpen

In een spel worden wapens en sleutels gebruikt.

package domein:



Maak de domeinklasse **Voorwerp**:

- 3 attributen: naam (verplicht, mag niet gewijzigd worden), gewicht (in kg) en niveau (int)
- constructor met 3 parameters (controles zie beschrijving setter)
- constructor met 2 parameters. Niveau wordt hier standaard op 1 ingesteld.
- getter voor elk attribuut
- setter voor het attribuut:
 - gewicht: moet een positief getal zijn, kleiner dan 1000 kg
 - niveau: er zijn 10 niveaus in het spel, genummerd van 1 t.e.m. 10
 - geen setter voor naam want naam mag niet gewijzigd worden. Naam mag niet null of leeg zijn.
- toString **Voorwerp** ... met gewicht ... (3 cijfers na komma) kg uit niveau ...

Maak 2 subklassen:

De klasse **Wapen**:

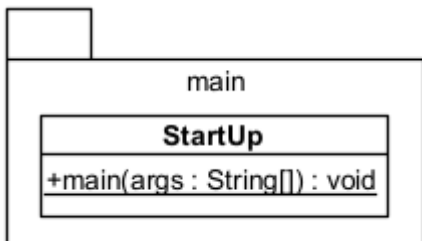
- 2 extra attributen: kracht en gebruikt

- constructor
- getters voor de extra attributen
- setter voor elk extra attribuut:
 - kracht: is een positief getal
 - gebruikt: geen controle nodig
- wapens zijn enkel beschikbaar in niveaus 1 t.e.m. 5: voorzie hiervoor een controleerNiveau-methode
- toString (zie verder voor de gevraagde output)

De klasse **Sleutel**:

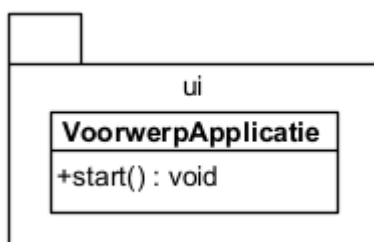
- 2 extra attributen: deur (int), nummer van de deur waarop de sleutel past en aantalInOmloop: stelt het aantal sleutels voor die aanwezig zijn in het spel
- constructor
- getters voor de extra attributen
- setter voor deur: moet een positief getal zijn
- toString (zie verder voor de gevraagde output)
- methode pastOp: geeft true/false terug naargelang de huidige sleutel op de als parameter meegegeven deur past of niet

package main:



In de klasse **StartUp** wordt de main-methode uitgewerkt. In deze main-methode starten we de applicatie door de methode start aan te roepen uit de ui-klasse.

package ui:



Schrijf een applicatie waarin een array van Wapens en een array van Sleutels worden aangemaakt. Laat de applicatie de gecreëerde objecten weergeven.

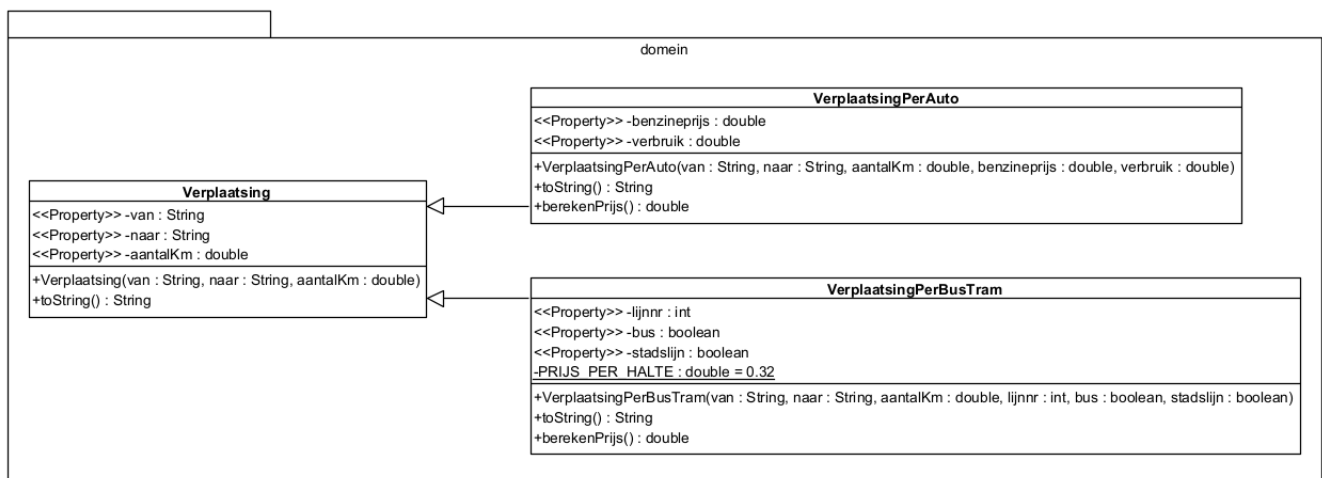
Mogelijke uitvoer:

Wapen colt met gewicht 1,500 kg uit niveau 3 en met kracht 6 nog niet gebruikt.
 Wapen brown met gewicht 0,500 kg uit niveau 1 en met kracht 23 al gebruikt.
 Sleutel voordeur met gewicht 0,500 kg uit niveau 3 past op deur 1.
 Er zijn 2 sleutel(s) in omloop.
 Sleutel achterdeur met gewicht 0,500 kg uit niveau 1 past op deur 2.
 Er zijn 2 sleutel(s) in omloop.

2.2. Stageverplaatsingen

Een hogeschool vraagt aan de lectoren om de stageverplaatsingen bij te houden. Er bestaan 2 soorten verplaatsingen: per auto of per bus/tram. In beide gevallen moeten enkele gegevens worden bijgehouden. De bedoeling is dat de kostprijs kan worden berekend van elke verplaatsing en zo ook het totaal van alle verplaatsingskosten van een lector.

package domein:



superklasse Verplaatsing (zie ook UML en testklasse):

- 3 attributen: van (String), naar (String) en aantalKm (double).
- constructor met 3 parameters
- getter voor elk attribuut
- setter voor elk attribuut:
 - van en naar moeten ingevuld zijn, anders exception gooien met duidelijke boodschap
 - aantalKm moet een positief getal zijn, anders exception met passende boodschap gooien
 - toString: plaats in 1 string: "verplaatsing van X naar Y" (met X en Y ingevuld met de gegeven waarden)

subklasse VerplaatsingPerAuto (zie ook UML en testklasse):

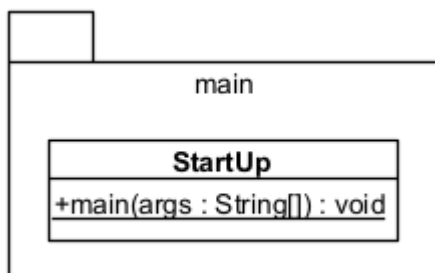
- 2 extra attributen: benzineprijs (double, in euro/liter) en verbruik (double, in liter/km)
- constructor met 5 parameters.
- getter voor elk extra attribuut
- setter voor elk extra attribuut :

- benzineprijs moet tussen 0.50 en 2.00 euro liggen (grenzen inbegrepen), anders exception gooien
- verbruik moet tussen 0.02 en 0.07 liter liggen (grenzen inbegrepen), anders exception gooien
- extra methode berekenPrijs: de prijs van een verplaatsing per auto (heen en terug) wordt berekend adhv de formule: $\text{verbruik} * \text{benzineprijs} * \text{aantalKm} * 2$
- toString: zie output

subklasse VerplaatsingPerBusTram (zie ook UML en testklasse):

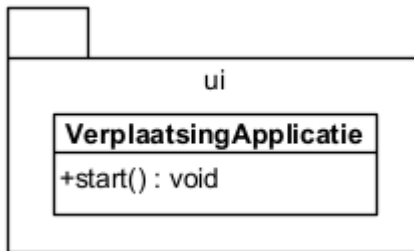
- 4 extra attributen: lijnnr (int), bus (boolean), stadslijn (boolean), PRIJS_PER_HALTE (double). Het laatste attribuut is een vaste waarde voor alle verplaatsingen per bus of tram en zal in de loop van het programma niet gewijzigd kunnen worden.
- constructor met 6 parameters (alles behalve PRIJS_PER_HALTE).
- getter voor elk extra attribuut behalve PRIJS_PER_HALTE
- setter voor elk extra attribuut behalve PRIJS_PER_HALTE:
 - lijnnr moet een strikt positief geheel getal zijn, anders exception gooien
- extra methode berekenPrijs: de prijs van de verplaatsing (heen en terug) bereken je als volgt:
 - er is 1 halte per km, dus als je de afstand (het aantalKm) van de verplaatsing afrondt naar boven (en naar een geheel getal) dan ken je het aantal haltes
 - de prijs van de verplaatsing is dan $2 * \text{aantalHaltes} * \text{PRIJS_PER_HALTE}$
 - voor een verplaatsing met een stadslijn krijg je 20% korting op deze prijs.
- toString: zie output

package main:



In de klasse StartUp wordt de main-methode geschreven die de defaultconstructor uit de ui-klasse oproept.

package ui:



In de applicatie maak je in de methode `start` een array van `VerplaatsingPerAuto`-objecten en een tweede array van `VerplaatsingPerBusTram`-objecten aan. Geef eerst alle informatie over de verplaatsingen per auto weer (zie uitvoer) en vervolgens dezelfde informatie over de verplaatsingen per bus of tram (zie eveneens uitvoer). Bereken de totale kosten voor alle verplaatsingen samen en toon ook deze.

Mogelijke uitvoer:

```
verplaatsing van HoGent campus Schoonmeersen Gent naar EFFIX Waregem per auto
Kosten voor deze verplaatsing: € 4,58
```

```
verplaatsing van HoGent campus Aalst naar Brenso NV Affligem per auto
Kosten voor deze verplaatsing: € 1,89
```

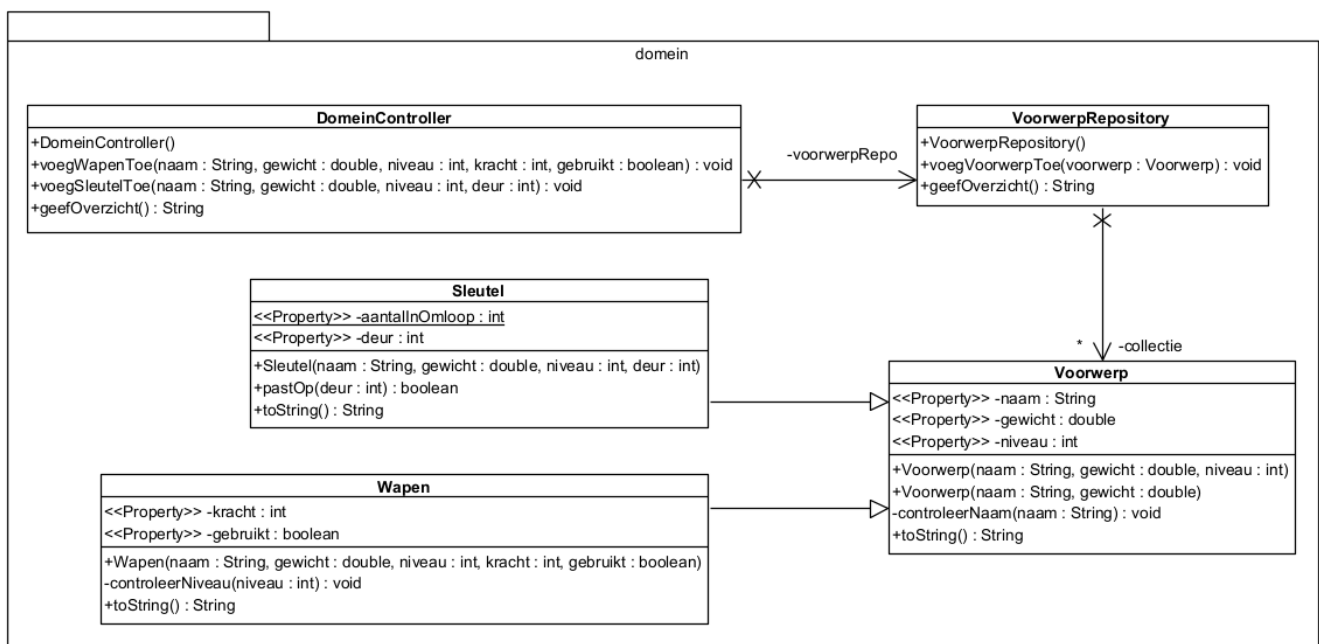
```
verplaatsing van HoGent campus Schoonmeersen Gent naar Technologiepark Zwijnaarde met stadsbus 70
Kosten voor deze verplaatsing: € 2,56
```

```
verplaatsing van Sint-Pietersstation Gent naar Vijfwindgatenstraat Gent met stadstram 22
Kosten voor deze verplaatsing: € 1,54
```

```
Totale kosten voor alle verplaatsingen samen: € 10,57
```

2.3. Spel met voorwerpen uitgebreid

We voegen nu aan de domeinlaag 2 klassen toe:



De klasse **VoorwerpRepository** (zie testklasse):

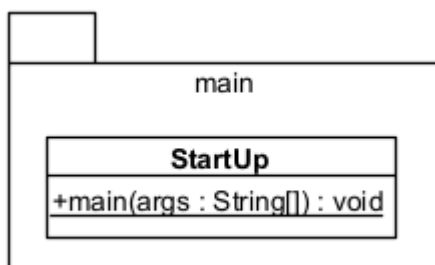
- 1 attribuut: collectie (List <Voorwerp>); de collectie voorwerpen wordt opgeslagen in een arraylist in de domeinklasse VoorwerpRepository
- constructor: maakt een lege arraylist
- voegVoorwerpToe met 1 parameter: een voorwerp
- geefOverzicht: maakt een overzicht van alle voorwerpen in de collectie en geeft dit in Stringvorm terug; geeft de melding "collectie is leeg" terug als er geen voorwerpen in de collectie zitten (zie uitvoer)

De klasse **Domeincontroller**:

- 1 attribuut: object van de domeinklasse VoorwerpRepository
- constructor: maakt de repository
- voegWapenToe met 5 parameters: roept de voegVoorwerpToe-methode aan in VoorwerpRepository
- voegSleutelToe met 4 parameters: roept de voegVoorwerpToe-methode aan in VoorwerpRepository
- geefOverzicht: roept de overeenkomstige methode aan in VoorwerpRepository

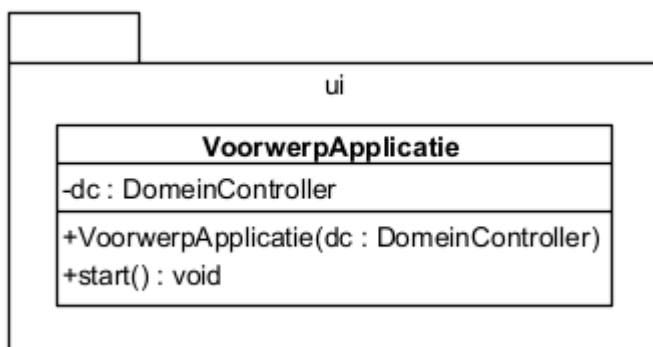
Verder passen we ook de packages main en ui aan:

package main:



In de klasse StartUp wordt de main-methode geschreven die de methode start uit de ui-klasse oproept. Let op: de start-methode heeft nu een DomeinController-object als parameter nodig!

package ui:



Schrijf een applicatie die 2 sleutels en 2 wapens in de repository plaatst. Toon de collectie van voorwerpen die in de repository aanwezig is VOOR en NA het toevoegen van deze objecten.

Mogelijke uitvoer:

Beginsituatie:
collectie is leeg

Na het toevoegen van enkele wapens en sleutels:

Wapen Colt met gewicht 1,500 kg uit niveau 3 en met kracht 6 nog niet gebruikt.

Sleutel Voordeur met gewicht 0,500 kg uit niveau 3 past op deur 1.

Er zijn 2 sleutel(s) in omloop.

Wapen Brown met gewicht 0,500 kg uit niveau 1 en met kracht 23 al gebruikt.

Sleutel Achterdeur met gewicht 0,500 kg uit niveau 1 past op deur 2.

Er zijn 2 sleutel(s) in omloop.