

HO GENT

H8 Testen - Werkcollege

Table of Contents

1. Oefening Zichtrekening	1
2. Oefening Rechthoek	1
2.1. Unit testen voor de constructor	2
2.2. Unit testen voor geefOmtrek()	2
2.3. Unit testen voor geefOppervlakte()	3
2.4. Unit testen voor schaal()	3

1. Oefening Zichtrekening

De domeinklasse Zichtrekening

Zichtrekening
-saldo : BigDecimal
+getSaldo() : BigDecimal +storten(bedrag : BigDecimal) : void

Testscenario's

- Jan heeft een nieuwe zichtrekening.
 - Het saldo is gelijk aan 0.
- Jan heeft een nieuwe zichtrekening.
 - Hij stort 200 euro op zijn rekening. Het saldo wordt dan 200 euro.
- Testen op exceptions
 - Storten van een negatief getal → `IllegalArgumentException` met als boodschap: "Negatief bedrag kan niet gestort worden"
 - Storten van ongeldig object (= null referentie) → `IllegalArgumentException` met als boodschap: "Ongeldig bedrag"

2. Oefening Rechthoek

De domeinklasse Rechthoek en de bijhorende domeinregels

Rechthoek
<<Property>> -lengte : double <<Property>> -breedte : double
+Rechthoek(lengte : double, breedte : double) +geefOmtrek() : double +geefOppervlakte() : double +schaal(percentages : int) : void -setLengte(lengte : double) : void -setBreedte(breedte : double) : void

Domeinregels:

- De lengte en de breedte van een rechthoek moeten groter zijn dan 0
- Het schalingspercentage moet tussen 1 en 200 liggen

Voor elke publieke methode uit Rechthoek maken we 1 of meerdere unit testen aan.

Overzicht te testen methodes:

- constructor
- geefOmtrek
- geefOppervlakte
- schaal

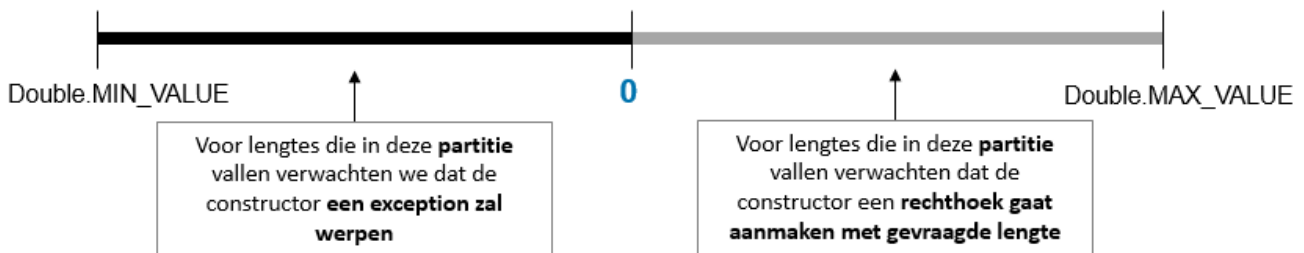
Wordt niet getest:

- getLengte en getBreedte (gegenereerde methodes)
- private set-methodes:
 - private methodes kunnen niet rechtstreeks getest worden, maar worden onrechtstreeks aan testen onderworpen doordat we de methodes testen die deze private methodes gebruiken

2.1. Unit testen voor de constructor

`Rechthoek(lengte: double, breedte: double)`

- lengte



- breedte: identieke equivalentiepartities en grenswaarden

Volgens de domeinregel moeten én lengte én breedte in de rechtse partitie liggen om succesvol een rechthoek te kunnen aanmaken



Merk op dat de grenswaarde 0 geen deel uitmaakt van deze partitie

Stel de nodige testcases op, op basis van de equivalentiepartities en de grenswaarde.

Implementeer de nodige testen.

2.2. Unit testen voor geefOmtrek()

`public double geefOmtrek()`

We vertrekken van een geldige rechthoek, ongeldige rechthoeken kunnen niet aangemaakt worden

Voor elke geldige rechthoek moet deze methode de omtrek juist berekenen

Stel een testcase op om deze methode te testen en implementeer ze.

2.3. Unit testen voor geefOppervlakte()

```
public double geefOppervlakte()
```

We vertrekken opnieuw van een geldige rechthoek, ongeldige rechthoeken kunnen niet aangemaakt worden.

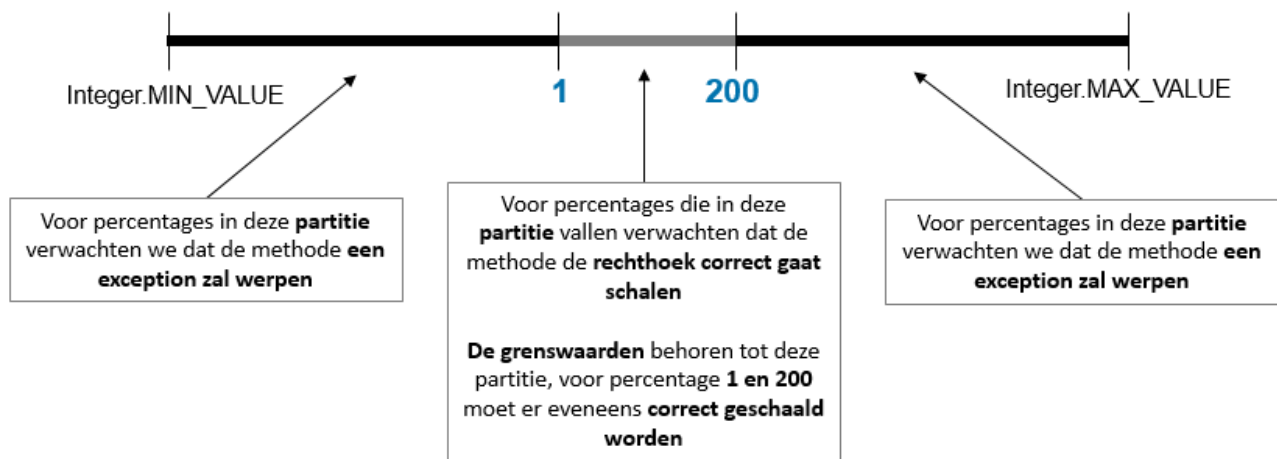
Voor elke geldige rechthoek moet deze methode de oppervlakte juist berekenen.

Zorg ervoor dat je code kan hergebruiken uit de testcase voor de methode geefOmtrek(), en dit voor het instantiëren van een geldige rechthoek.

2.4. Unit testen voor schaal()

```
public void schaal(int percentage)
```

Partities en grenswaarden:



Stel de nodige testcases op, op basis van de equivalentiepartities en de grenswaarde.

Implementeer de nodige testen en zorg dat je code kan hergebruiken door gebruik te maken van geparametriseerde testen.