Les 11 lesprogramma

# Opgave Casten met ABC 2

De klassen A, B en C zijn uitgebreid zoals hieronder is weergegeven.

public class A {  
  
 @Override  
 public String toString() {  
 return "A";  
 }  
   
 public void doA() {  
 System.out.println("AA");  
 }  
}

public class B extends A {  
   
 @Override  
 public String toString() {  
 return "B";  
 }  
   
 public void doB() {  
 System.out.println("BB");  
 }  
}

public class C extends B {  
   
 @Override  
 public String toString() {  
 return "C";  
 }  
   
 public void doC() {  
 System.out.println("CC");  
 }  
}

## Opgave A

Onderzoek in het programm ABC2 de of de onderstaande casts mogelijk zijn zonder een error op te leveren.

* a3 casten naar C
* a2 casten naar C
* a1 casten naar C
* b2 casten naar A
* c3 casten naar B
* c3 casten naar A

Ga als volgt te werk:

Implementeer de cast in het hoofdprogramma en controleer welke van de drie methoden doA, doB en doC eclipse je nu toestaat aan te roepen.

Run vervolgens het programma en controleer of je nu een runtime error krijgt.

## Opgave B

Welke algemene conclusies kun je trekken op basis van dit experiment?

# Opgave Figuren 2

*Vervolg van Opgave figuren uit les 10. Nu voegen we een abstracte klasse toe zoals ook in de screencast te zien is.*

## Opgave A

In de opgave Figuren 1 uit les 4-1 heb je de superklasse Figuur gemaakt om alle gedupliceerde code uit rechthoek en cirkel te verhelpen. Voeg aan deze klasse de abstracte methode teken toe en maak de klasse zelf ook abstract.

## Opgave B

Vervang in Cirkel en Rechthoek de methoden tekenCirkel en tekenRechthoek door de methode teken en zorg dat dit een Override is van de abstracte methode uit de klasse Figuur.

## Opgave C

Niet elk figuur hoeft te kunnen bewegen en daarom zou je kunnen overwegen om alle code die te maken heeft met snelheid en versnelling uit Figuur te halen en in een nieuwe klasse BeweegbaarFiguur te stoppen. Wanneer je dit doet, dan zou je ook een BeweegbareCirkel en een BeweegbareRechthoek moeten maken.

Teken een klassendiagram met Figuur, BeweegbaarFiguur, Cirkel, Rechthoek, BeweegbareCirkel, BeweegbareRechthoek. Van alle velden en methoden hoef je alleen de methoden public void setKleur(int kleur), public void doeStap(), public void setSnelheid() en public void teken(PApplet p) op te nemen. Geef wel duidelijk aan welke methoden abstract zijn en welke niet.

## Opgave D

De indeling die je in opgave C hebt getekend zorgt voor gedupliceerde code. Welke code ben je aan het dupliceren?

## Opgave E

Zou je op basis van opgave C en D adviseren een onderscheid te maken in Figuur en BeweegbaarFiguur, of zou je alleen de klasse Figuur die ook alle code bevat om figuren te laten bewegen? Beargumenteer het antwoord.