


# Interfaces

Hoofdstuk 13

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

Let's start with a  
joke, that you will  
get in the end.



# Abstracte klasse

- ❏ Shape is een abstracte klasse
  - Geen concrete implementaties
  - abstracte methode
  - kan ook niet abstracte methode bevatten.
- ❏ Interface is een soort abstracte klasse met enkel abstracte methoden
  - Sinds Java 8 zijn er ook default en static methoden mogelijk in een interface. (zien we later uitgebreider)

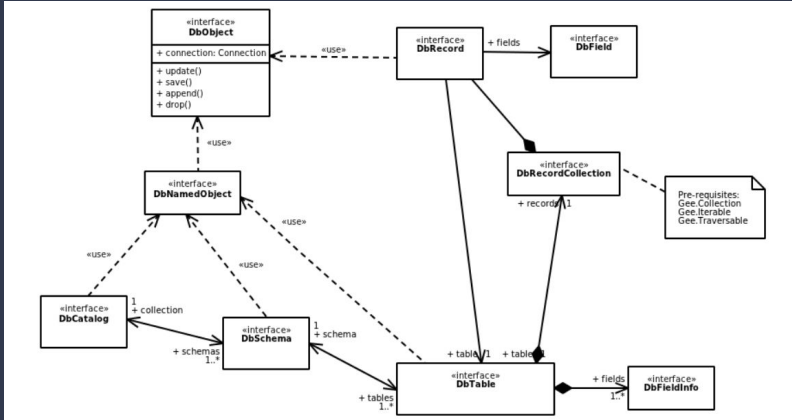
# Interface

- ❑ Interface definieert welke methodes voorhanden zijn.
- ❑ Klassen kunnen een interface **implementeren**.
- ❑ Een interface is een soort contract met de buitenwereld.
  - Klassen die de interface implementeren zullen een implementatie van de opgelegde methoden hebben.
- ❑ In Java kan een klasse meerdere interfaces implementeren (klassen **kunnen enkel van 1 Superklasse extenden**)

# Wat is nu hier het nut van?

- ❑ Polymorfisme.
  - Totaal verschillende klassen toch hetzelfde gedrag geven, zonder dat ze zijn afgeleid van dezelfde superklasse.
- ❑ Methoden opleggen aan klassen.
- ❑ Objecten van een bepaalde interface ter beschikking stellen zonder dat men de klasse of het object moet prijsgeven.
- ❑ Late binding
  - De **concrete koppeling** aan de uit te voeren code gebeurt niet tijdens de compilatie maar **tijdens de uitvoering** van het programma.

# Why tough?



- ❑ Verplichten van een database verbinding op de juiste manier op te zetten

➔ Security bevorderen

- ❑ Verplichten de database Structuur op de juiste manier aanspreken.

# Definitie interface

Definitie interface bestaat uit twee delen

1. declaratie van de interface.
  2. Beschrijving van de interface.
- `public interface InterfaceName implements SuperInt1, SuperInt2{  
 //body  
}`
  - `public interface Scalable {  
 int DOUBLE = 200;  
 int HALF = 50;  
 int QUARTER = 25;  
  
 void scale(int factor);  
}`

# default methoden

- ❏ Sinds Java 8 kunnen interfaces ook default methoden bevatten.
  - Dit zijn methoden die een implementatie kunnen hebben.
  - Worden voorafgegaan door het woord **default**.
- ❏ Implementerende klassen kunnen drie dingen doen.
  1. Niets.
  2. Opnieuw abstract declareren.
  3. Opnieuw implementeren.



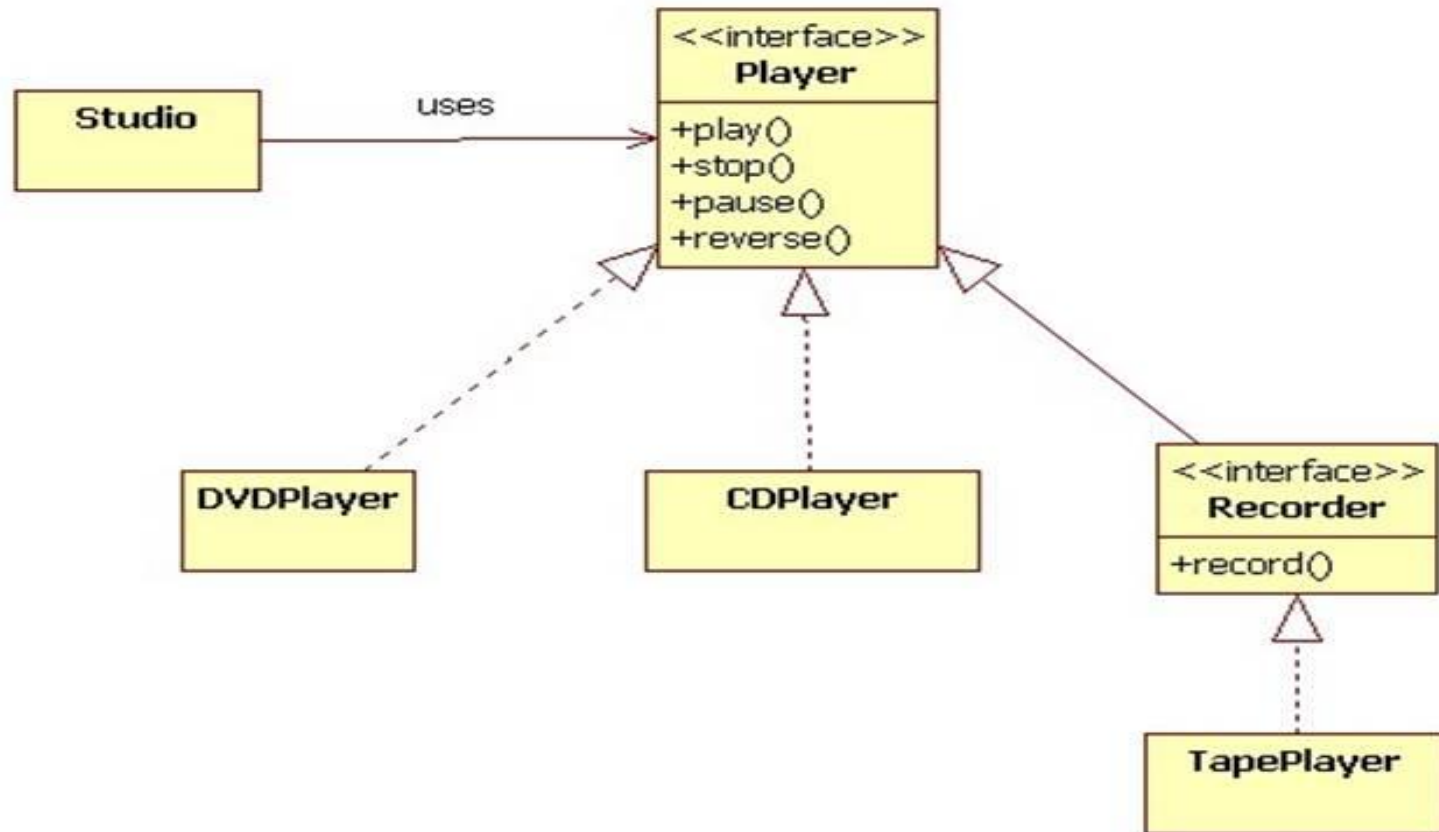
# Statische methoden

- ❑ Sinds Java 8 zijn er ook statische methoden, deze hebben wel een implementatie.
- ❑ Het is niet mogelijk statische methoden op te roepen aan de hand van de implementerende klassen.
  - ➔ De constanten van een interface kunnen zo wel aangeroepen worden.

# De interface als datatype.

- ❑ Door een interface te declareren, declareren we feitelijk een nieuw datatype (referentietype).
- ❑ Tot nu toe hebben we referentie variabelen gedeclareerd als referentie naar een object van die klassen of een subklasse.
- ❑ Men kan echter ook een referentie declareren naar een object dat een interface implementeert.
- ❑ Om na te gaan of een object een bepaalde interface implementeert
  - gebruik maken van instanceof (operator)

Practical



aker

ende