# interface

### Voorbeeld

### Te gebruiken bij:

-

### Eigenschappen:

-

### Info

#### overzicht

#### aantekening

Een interface in Java definieert een contract of een set van methoden die een klasse moet implementeren. Het specificeert de methoden die een klasse moet bieden zonder aan te geven hoe die methoden geïmplementeerd moeten worden.

Interfaces worden gebruikt om abstractie te bereiken en een manier te bieden voor niet-gerelateerde klassen om met elkaar te communiceren op basis van een gemeenschappelijk gedrag dat wordt gedefinieerd door de interface. Door een interface te implementeren, garandeert een klasse dat het de gespecificeerde methoden zal bieden en kan het worden behandeld als een instantie van die interface.

In dit voorbeeld definieert de interface "Afdrukbaar" de methode "afdrukken". De klassen "Boek" en "Document" implementeren de interface en bieden hun eigen implementatie van de "afdrukken" methode. In de "Hoofd" klasse worden objecten van zowel "Boek" als "Document" gemaakt en kan de "afdrukken" methode op elk van hen worden opgeroepen. Dit stelt je in staat om verschillende objecten met verschillende implementaties te verzamelen en ze op uniforme wijze aan te roepen door de interface te gebruiken.

interface Printable {

void print();

}

class Book implements Printable {

private String title;

public Book(String title) {

this.title = title;

}

@Override

public void print() {

System.out.println("Book: " + title);

}

}

class Document implements Printable {

private String name;

public Document(String name) {

this.name = name;

}

@Override

public void print() {

System.out.println("Document: " + name);

}

}

public class Main {