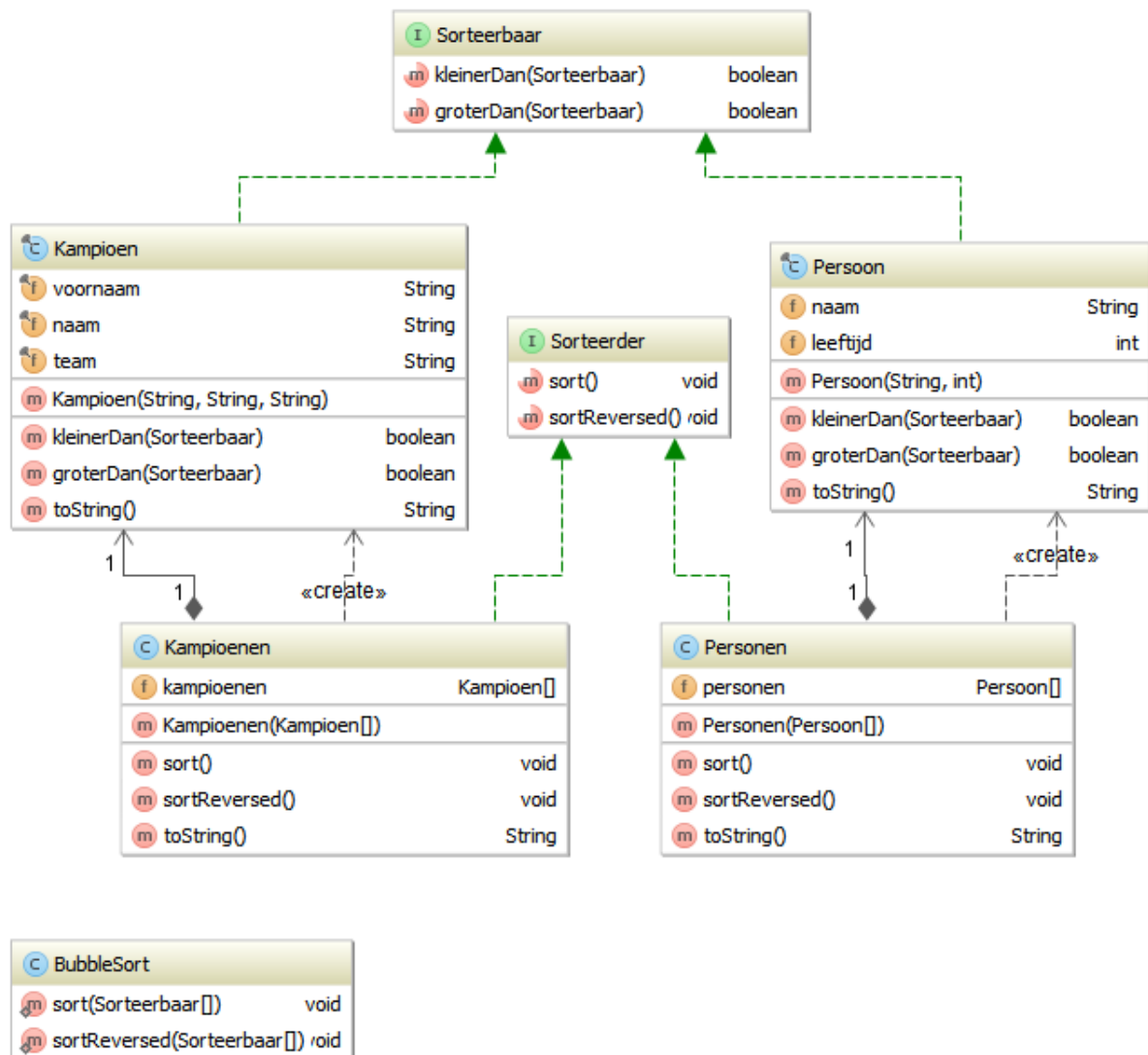


P2W3 Opdracht Sorteerbaar versie2

UML klassendiagram (via IntelliJ)



De interface Sorteertaar

```
public interface Sorteertaar {  
    boolean kleinerDan(Sorteertaar s);  
  
    boolean groterDan(Sorteertaar s);  
}
```

De interface Sorteerder

```
public interface Sorteerder {  
    void sort();  
  
    void sortReversed();  
}
```

De klasse BubbleSort

```
/**  
 * In deze klasse krijg je de sort methode cadeau.  
 * Werk de andere methode zelf uit.  
 */  
public class BubbleSort {  
    public static void sort(Sorteertaar[] tabel) {  
        int n = tabel.length;  
        for (int i = 0; i < n - 1; i++) {  
            for (int j = 0; j < n - i - 1; j++) {  
                if (tabel[j + 1].kleinerDan(tabel[j])) {  
                    Sorteertaar temp = tabel[j + 1];  
                    tabel[j + 1] = tabel[j];  
                    tabel[j] = temp;  
                }  
            }  
        }  
    }  
  
    public static void sortReversed(Sorteertaar[] tabel) {  
    }  
}
```

De klasse Persoon

```
/**
 * Pas aan waar nodig.
 *
 * Maak gebruik van de Sorteerbaar interface.
 */
public final class Persoon {
    private String naam;
    private int leeftijd;

    public Persoon(String naam, int leeftijd) {
        this.naam = naam;
        this.leeftijd = leeftijd;
    }

    /* Deze methode geeft true terug als het eerste getal
     * kleiner is dan het tweede.
     */
    public boolean kleinerdan(Sorteerbaar sorteerbaar) {
        return false;
    }

    /* Deze methode doet net het omgekeerde */
    public boolean groterDan(Sorteerbaar sorteerbaar) {
        return false;
    }

    /* Zie gewenste uitvoer voor de formatering van de string. */
    public String toString() {
        return null;
    }
}
```

De klasse Kampioen

```
/**
 * Pas aan waar nodig.
 *
 * Maak gebruik van de Sorteerbaar interface.
 */
public final class Kampioen {
    private final String voornaam;
    private final String naam;
    private final String team;

    public Kampioen(String voornaam, String naam, String team) {
        this.voornaam = voornaam;
        this.naam = naam;
        this.team = team;
    }

    /* Deze methode geeft true terug als het eerste woord alfabetisch
     * voor het tweede komt.
     */
    public boolean kleinerdan(Sorteerbaar sorteerbaar) {
        return false;
    }

    /* Deze methode doet het omgekeerde van de voorgaande */
    public boolean groterDan(Sorteerbaar sorteerbaar) {
        return false;
    }

    /* Zie gewenste uitvoer voor de formatering van de string. */
    public String toString() {
        return null;
    }
}
```

De klasse Personen

```
/**
 * Vul/pas aan waar nodig. Maak gebruik van de Sorteerder interface.
 */
public class Personen {
    private Persoon personen[];

    /**
     * Maak hier de nieuwe tabel. Kopieer dan de tabel van Persoon
     * objecten naar de nieuwe tabel van het type Persoon.
     */
    public Personen(Persoon[] personen) {

    }

    /** Sorteert met de sort methode (zie klasse BubbleSort) */
    public void sort() {

    }

    /** Sorteert met de sortReversed methode (zie klasse BubbleSort) */
    public void sortReversed() {

    }

    /** Zorg ervoor dat de gewenste afdruk op het scherm komt */
    public String toString() {
        return "nonsens";
    }
}
```

De klasse Kampioenen

```
/**
 * Vul/pas aan waar nodig. Maak gebruik van de Sorteerder interface.
 */
public class Kampioenen {
    private Kampioen kampioenen[];

    /**
     * Maak hier de nieuwe tabel. Kopieer dan de tabel van Kampioen
     * objecten naar de nieuwe tabel van het type Kampioen.
     */
    public Kampioenen(Kampioen[] kampioenen) {

    }

    /** Sorteert met de sort methode (zie klasse BubbleSort) */
    public void sort() {

    }

    /** Sorteert met de sortReversed methode (zie klasse BubbleSort) */
    public void sortReversed() {

    }

    /** Zorg ervoor dat de gewenste afdruk op het scherm komt */
    public String toString() {
        return "nonsens";
    }
}
```

De klasse TestSorteerbaar

```
public class TestSorteerbaar {
    public static void main(String[] args) {
        Persoon[] runaways = {
            new Persoon("Joan Jett", 58),
            new Persoon("Cherie Currie", 53),
            new Persoon("Lita Ford", 55),
            new Persoon("Sandy West", 54),
            new Persoon("Jackie Fox", 53),
        };

        Kampioen[] champions = {
            new Kampioen("Sebastian", "Vettel", "Red Bull"),
            new Kampioen("Jenson", "Button", "Brawn GP"),
            new Kampioen("Kimi", "Räikkönen", "Ferrari"),
            new Kampioen("Fernando", "Alonso", "Renault"),
            new Kampioen("Michael", "Schumacher", "Ferrari"),
            new Kampioen("Mika", "Häkkinen", "McLaren"),
            new Kampioen("Lewis", "Hamilton", "McLaren")
        };

        Sorteerder personen = new Personen(runaways);
        System.out.println("Personen origineel:\n" + personen);
        personen.sort();
        System.out.println("Personen van jong naar oud:\n" + personen);
        personen.sortReversed();
        System.out.println("Personen oud naar jong:\n" + personen);

        Sorteerder kampioenen = new Kampioenen(champions);
        System.out.println("Kampioenen origineel:\n" + kampioenen);
        kampioenen.sort();
        System.out.println("Kampioenen alfabetisch volgens familienaam:\n"
            + kampioenen);
        kampioenen.sortReversed();
        System.out.println("Kampioenen omgekeerd alfabetisch volgens " +
            familienaam:\n" + kampioenen);
    }
}
```

Verwachte uitvoer:

Personen origineel:

Joan Jett	58
Cherie Currie	53
Lita Ford	55
Sandy West	54
Jackie Fox	53

Personen van jong naar oud:

Cherie Currie	53
Jackie Fox	53
Sandy West	54
Lita Ford	55
Joan Jett	58

Personen oud naar jong:

Joan Jett	58
Lita Ford	55
Sandy West	54
Cherie Currie	53
Jackie Fox	53

Kampioenen origineel:

Sebastian Vettel	Red Bull
Jenson Button	Brawn GP
Kimi Räikkönen	Ferrari
Fernando Alonso	Renault
Michael Schumacher	Ferrari
Mika Häkkinen	McLaren
Lewis Hamilton	McLaren

Kampioenen alfabetisch volgens familienaam:

Fernando Alonso	Renault
Jenson Button	Brawn GP
Lewis Hamilton	McLaren
Mika Häkkinen	McLaren
Kimi Räikkönen	Ferrari
Michael Schumacher	Ferrari
Sebastian Vettel	Red Bull

Kampioenen omgekeerd alfabetisch volgend familienaam:

Sebastian Vettel	Red Bull
Michael Schumacher	Ferrari
Kimi Räikkönen	Ferrari
Mika Häkkinen	McLaren
Lewis Hamilton	McLaren
Jenson Button	Brawn GP
Fernando Alonso	Renault