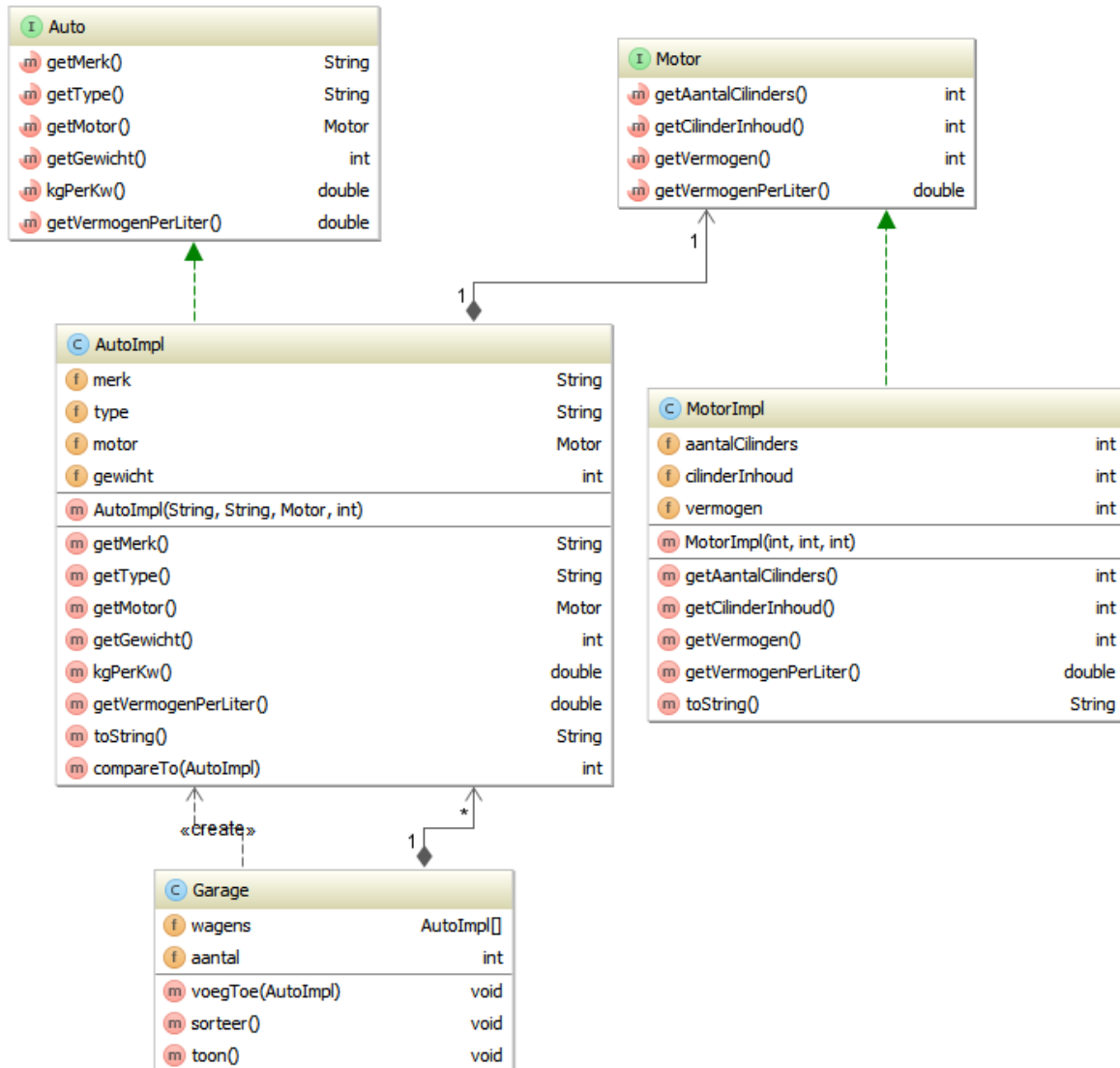


Opdracht Voertuig [EXTRA]

UML klassendiagram (via IntelliJ)



De interface Auto

De interface ziet er zo uit:

```
public interface Auto {
    String getMerk();
    String getType();
    Motor getMotor();
    int getGewicht();
    double kgPerKw();
    double getVermogenPerLiter();
}
```

De interface Motor

De interface ziet er zo uit:

```
public interface Motor {
    int getAantalCilinders();
    int getCilinderInhoud();
    int getVermogen();
    double getVermogenPerLiter();
}
```

Aan beide interfaces mag je niets wijzigen!

De klasse AutoImpl

De klasse **AutoImpl** implementeert zowel de interface **Auto** en de interface **Comparable** (standaard Java interface). Vul de onderstaande klasse aan.

- Voorzie getters voor alle attributen.
- Voorzie de methode **kgPerKw** (gewicht / vermogen in kW)
- Voorzie de methode **getVermogenPerLiter** (vraag dit aan het **MotorImpl**-object).
- Voorzie een aangepaste **toString**-methode (zie uitvoer)

```
public class AutoImpl implements Auto, Comparable<AutoImpl> {
    private String merk;
    private String type;
    private Motor motor;
    private int gewicht; // in kg

    // vul hier aan

    /**
     * Vergelijkt volgens het vermogen per liter.
     *
     * @param auto Een Auto object
     * @return int
     */
    public int compareTo(AutoImpl auto) {
        double mijn = getVermogenPerLiter();
        double ander = auto.getVermogenPerLiter();
        int difference = Double.compare(mijn, ander);
        if (difference == 0) return 0;
        return (difference < 0) ? -1 : 1;
    }
}
```

De klasse MotorImpl

De klasse **MotorImpl** implementeert de interface **Motor**.

De attributen zijn:

- **int** `aantalCilinders`;
- **int** `cilinderInhoud`; // in cc
- **int** `vermogen`; // in kW
 - Voorzie een *constructor* om alle attributen een beginwaarde te geven.
 - Voorzie *getters* voor alle attributen.
 - Voorzie een methode **getVermogenPerLiter**.
 - Voorzie een aangepaste **toString**-methode (zie uitvoer)

De klasse Garage

De klasse **Garage** bevat slechts 2 attributen:

- Een tabel voor VIER **AutoImpl** objecten met de naam **wagens**
- Een **int** met de naam **aantal**.
- Voorzie een methode **voegToe** om een **AutoImpl** object aan **wagens** toe te voegen.
- Voorzie een methode **sorteer** om de auto's volgens oplopend vermogen per liter te sorteren.
- Voorzie een *override* van de methode **toString** om alle gegevens als een string terug te geven. Maak gebruik van de **toString** methoden van de andere klassen.

De klasse TestVoertuig

Deze klasse dient uitsluitend om de goede werking van de overige klassen te testen.

Je mag er niets aan veranderen. Zie verder voor de verwachte afdruk.

```
public class TestVoertuig {
    public static void main(String[] args) {
        Garage garage = new Garage();

        garage.voegToe(new AutoImpl("Porsche", "Boxter S",
                                    new MotorImpl(6, 3387, 217), 1355));
        garage.voegToe(new AutoImpl("Porsche", "Cayman",
                                    new MotorImpl(6, 2687, 155), 1305));
        garage.voegToe(new AutoImpl("Porsche", "GT3",
                                    new MotorImpl(6, 3600, 305), 1395));
        garage.voegToe(new AutoImpl("Porsche", "Carrera S",
                                    new MotorImpl(6, 3824, 261), 1420));

        garage.sorteer();
        System.out.println(garage);
    }
}
```

Gewenste uitvoer:

Merk en type: Porsche Cayman
Motor: 6 cilinders 2687cc 155kW
Gewicht: 1305kg
Vermogen per liter; 57,69
kg per kW: 8,42

Merk en type: Porsche Boxter S
Motor: 6 cilinders 3387cc 217kW
Gewicht: 1355kg
Vermogen per liter; 64,07
kg per kW: 6,24

Merk en type: Porsche Carrera S
Motor: 6 cilinders 3824cc 261kW
Gewicht: 1420kg
Vermogen per liter; 68,25
kg per kW: 5,44

Merk en type: Porsche GT3
Motor: 6 cilinders 3600cc 305kW
Gewicht: 1395kg
Vermogen per liter; 84,72
kg per kW: 4,57