Bereich: Vererbung (1)

Fahrzeuge, Wettrennen	**Musterlösung**
Package: `de.dhbwka.java.exercise.classes.vehicles`	**Klasse:** `Race`

```
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class Vehicle {

	private int wheels;
	private double speed;
	private double maxSpeed;
	private double position;

	public Vehicle() {
		this(0.0);
	}

	public Vehicle(double speed) {
		this(0, speed, 0.0, 0.0);
	}

	protected Vehicle(int wheels, double speed, double maxSpeed,
	    double position) {
		setWheels(wheels);
		setMaxSpeed(maxSpeed);
		setSpeed(speed); // after setMaxSpeed!
		setPosition(position);
	}

	public void drive(double minutes) {
		this.position += speed / 60.0 * minutes;
	}

	@Override
	public String toString() {
		return this.getClass().getSimpleName() + " at " + position +
			" km with " + wheels + " wheels at speed " + speed +
			" km/h of max. " + maxSpeed + " km/h.";
	}

	public int getWheels() {
		return wheels;
	}
```

```
	public void setWheels(int wheels) {
		this.wheels = wheels;
	}

	public double getSpeed() {
		return speed;
	}

	public void setSpeed(double speed) {
		this.speed = Math.min(this.maxSpeed, speed);
	}

	public double getMaxSpeed() {
		return maxSpeed;
	}

	public void setMaxSpeed(double maxSpeed) {
		this.maxSpeed = maxSpeed;
	}

	public double getPosition() {
		return position;
	}

	public void setPosition(double position) {
		this.position = position;
	}

}
```

```
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class Bicycle extends Vehicle {

	public Bicycle() {
		this(0.0);
	}

	public Bicycle(double speed) {
		super(2, speed, 30.0, 0.0);
	}

}
```

```java
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class Car extends Vehicle {

	public Car() {
		this(0.0);
	}

	public Car(double speed) {
		this(speed, 140.0);
	}

	protected Car(double speed, double maxSpeed) {
		super(4, speed, maxSpeed, 0.0);
	}

}
```

```java
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class RacingCar extends Car {

	public RacingCar() {
		this(0.0);
	}

	public RacingCar(double speed) {
		this(speed, 220.0);
	}

	protected RacingCar(double speed, double maxSpeed) {
		super(speed, maxSpeed);
	}

}
```

```java
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class Ambulance extends Car {

	private boolean signal;

	public Ambulance() {
		this(0.0, false);
	}

	public Ambulance(double speed) {
		this(speed, false);
	}

	public Ambulance(double speed, boolean signal) {
		super(speed);
		setSignal(signal);
	}

	@Override
	public String toString() {
		return super.toString()+
			" Signal " + (signal ? "on":"off") + ".";
	}

	public boolean hasSignal() {
		return signal;
	}

	public void setSignal(boolean signal) {
		this.signal = signal;
	}

}
```

```java
package de.dhbwka.java.exercise.classes.vehicles;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'. Baden-Wuerttemberg
 * Cooperative State University.
 *
 * (C) 2016 by J. Sidler, T. Schlachter, C. Schmitt, W. Suess
 */
public class Race {

	public static void main(String[] args) {
		Vehicle[] vehicles = new Vehicle[4];
		vehicles[0] = new Bicycle(20.0);
		vehicles[1] = new Car(150.0);
		vehicles[2] = new RacingCar(200.0);
		vehicles[3] = new Ambulance(80.0, true);
		// 4 hours lead for the bike
		vehicles[0].drive(240.0);
		// 1 hour of driving for everyone
		for (int i = 0; i < vehicles.length; i++) {
			vehicles[i].drive(60);
		}
		// Output Race
		for (int i = 0; i < vehicles.length; i++) {
			System.out.println(vehicles[i].toString());
		}
	}

}
```