

package com.hoffrogge.lehreinheiten.uebungen.auto;

Package muss immer als erstes angegeben werden

Import von anderen Klassen

import com.hoffrogge.lehreinheiten.uebungen.Autobahn;

Definition der Klasse mit Namen

public class Auto {
 Innerhalb der {} stehen Variablen und Methoden der Klasse

private int geschwindigkeit;
private int maxTankInhalt;
private String marke;

Hier stehen Variablen außerhalb von Methoden, es sind Klassenvariablen

public Auto() {
 Standardkonstruktor, der Klassenvariablen nicht ändert
}

public Auto(int geschwindigkeit, int maxTankInhalt, String marke) {

 this.geschwindigkeit = geschwindigkeit;
 this.maxTankInhalt = maxTankInhalt;
 this.marke = marke;

Konstruktor, der Klassenvariablen ändert

 .this zur Unterscheidung von Klassenvariablen und lokalen Variablen/Parametern

Getter für die Geschwindigkeit. Ein anderes Objekt kann damit die Geschwindigkeit abfragen.

public int getGeschwindigkeit() {
 return geschwindigkeit;
}

Setter für die Geschwindigkeit. Ein anderes Objekt kann damit die Geschwindigkeit **ändern**.

public void setGeschwindigkeit(int geschwindigkeit) {
 this.geschwindigkeit = geschwindigkeit;
}

Getter für die Marke. Ein anderes Objekt kann damit die Marke abfragen.

public String getMarke() {
 return marke;
}

Setter für die Marke. Ein anderes Objekt kann damit die Marke **ändern**.

public void setMarke(String marke) {
 this.marke = marke;
}

Methode ohne Rückgabe, die einen Text auf die Konsole schreibt

public void fahre() {
 System.out.println("Mein " + marke + " fährt " + geschwindigkeit + " km/h");
}

Methode, die einen Wert berechnet und zurückgibt

public double berechneVerbrauch(double gefahreneKilometer) {
 double verbrauchProLiter = maxTankInhalt / gefahreneKilometer;
 double verbrauchProHundertLiter = verbrauchProLiter * 100;
 return verbrauchProHundertLiter;
}

Lokale Variablen, die nur in der Methode existieren

Methode, die true oder false als Wert zurückgibt

public boolean isAutobahntauglich() {
 return this.geschwindigkeit >= Autobahn.AUTOBAHN_MINDESTGESCHWINDIGKEIT;
}

Diese } markiert das Ende der Klasse

Klassen, Konstruktoren und Methoden haben immer einen „Rumpf“ der durch { und } umschlossen wird