

**Übungsblatt zu Get-/Set-Methoden, Konstruktoren,  
Bedingungen und Ausgaben**  
**Klasse mit Studenten**

SPENGERGASSE 

---

**Aufgabe 1** *Implementierung der Klasse Auto*

Die Klasse **Auto** modelliert ein Automobil mit diversen Eigenschaften. Die Eigenschaften und Methoden sind durch folgendes UML-Diagramm gegeben:

<b>Auto</b>
- marke: String - typ: String - automatik: boolean - ps: int - verbrauch: double
+ Auto(neueMarke: String, neuerTyp: String, neuAutomatik: boolean, neuePs: int, neuerVerbrauch: double) + setMarke(neueMarke: String): void + getMarke(): String + setTyp(neuerTyp: String): void + getTyp(): String + setAutomatik(neuAutomatik: boolean): void + hasAutomatik(): boolean + setPs(neuePs: int): void + getPs(): int + setVerbrauch(neuerVerbrauch: double): void + getVerbrauch(): double

Implementieren Sie zunächst die Get- und Set-Methoden, und anschließend die Konstruktoren.

---

**Aufgabe 2** *Parameterprüfung*

Fügen Sie nun in den Set-Methoden Parameterprüfungen hinzu. Die PS sollen im Bereich 20 bis 900 liegen (jeweils inklusive), der Verbrauch zwischen 0.5 und 50.0 Liter pro 100 km (ebenfalls jeweils inklusive). Prüfen Sie bei den **String**-Parametern, dass diese nicht **null** sind. Zum Beispiel mittels: **neueMarke != null**. Geben Sie entsprechende Fehlermeldungen auf der Kommandozeile aus, falls Parameterprüfungen fehlschlagen.

---

**Aufgabe 3** *Ausgabe*

Fügen Sie weiters eine **print()**-Methode hinzu, welche das Objekt auf die Konsole ausgibt. Die Ausgabe soll wie folgt aussehen:

Mercedes 190 (Automatik), 105 PS, Verbrauch: 9.7 1/100 km

bzw.

BMW 325i (Handschaltung), 170 PS, Verbrauch: 12.3 1/100 km

---