

## Methoden

*Methoden* in Java sind im Prinzip die objektorientierte Variante der Funktionen in Python. Ein wesentlicher Unterschied ist der Aufbau der *Signatur* bzw. des *Methodenkopfs*. Funktionssignaturen in Python folgen dem Schema `def <Bezeichner>(<Parameter>)`, wobei keine Datentypen angegeben werden müssen. In Java lautet der grundsätzliche Aufbau der Signatur:

`<Zugriffsmodifikatoren> <Rückgabedatentyp> <Bezeichner>(<Parameter>)`

Dabei müssen auch für die Parameter jeweils die Datentypen vor den Parametern angegeben werden, einzelne Parameter werden mit Kommas voneinander getrennt. Wir werden zunächst Methoden betrachten, die keinen Rückgabewert haben. Hier muss dann als Rückgabedatentyp `void` angegeben werden. Die Zugriffsmodifikatoren regeln, unter welchen Bedingungen die Methode ausgeführt werden darf. Zwei Zugriffsmodifikatoren kennen wir bereits:

*public*: Dieser Modifikator bestimmt, aus welchen anderen Klassen die Methode ausgeführt werden kann: In diesem Fall ist die Methode *öffentlich*, d. h. die Methode könnte aus jeder anderen Klasse heraus aufgerufen werden. Alternative Zugriffsmodifizierer: *private*, *protected*.

*static*: Dieser Modifikator bewirkt, dass aus der Klasse kein Objekt erzeugt werden muss, um die Methode auszuführen. Statische Methoden und Variablen sind also ein direkter Bestandteil der Klasse.

### Aufgabe 1: Methoden ohne Rückgabewert

Bearbeite die folgenden Aufgaben zum Lernvideo *Methoden* (Nr. 16 der Playlist):

- (a) Beschreibe die Funktionsweise einer Methode ohne Rückgabewert und Parameter.
- (b) Erläutere, wie das folgende Java-Programm funktioniert und was beim Ausführen der Main-Methode ausgegeben wird.

```
1 public class HalloMethode {
2     public static void hello(String name) {
3         System.out.println("Hallo, " + name + "!");
4     }
5     public static void main(String[] args) {
6         String name = new String("Elon Musk");
7         hello(name);
8     }
9 }
```

- (c) Erkläre den Methodenkopf der Main-Methode und begründe, warum alle bisher betrachteten Methoden statisch waren.

## Aufgabe 2: Methoden mit Rückgabewert

Bearbeite die folgenden Aufgaben zum Lernvideo *Methoden mit Rückgabewert* (Nr. 17 der Playlist).

- (a) Beschreibe, wie sich Methoden mit Rückgabewert von Methoden ohne Rückgabewert unterscheiden.
- (b) Erkläre die Funktionsweise des folgenden Programms:

```
1 public class Teilbarkeit {  
2     public static boolean istTeilbarDurch(int zahl, int teiler) {  
3         if (zahl % teiler == 0) {  
4             return true;  
5         } else {  
6             return false;  
7         }  
8     }  
9     public static void main(String[] args) {  
10        int z = 84;  
11        int t = 7;  
12        if (istTeilbarDurch(z, t)) {  
13            System.out.println(z + " ist teilbar durch " + t);  
14        } else {  
15            System.out.println(z + " ist nicht teilbar durch " + t);  
16        }  
17    }  
18 }
```

- (c) Implementiere basierend auf dem im vorigen Aufgabenteil eine Methode *boolean istPrimzahl(int num)*, die testet, ob eine Zahl eine Primzahl ist oder nicht. Überlege dir vorher, welche mathematischen Eigenschaften eine Primzahl besitzt.