



## Java – Methoden

Aufgaben, die ein Programm immer wiederkehrend vornehmen soll, können in Methoden ausgelagert werden. Diese sind *innerhalb* der Klasse, aber *außerhalb* der Main-Methode zu definieren. Die Methoden können dabei...

- ...ohne Parameter für Ein- und Ausgabe...  
`public void methode( ) ...`
- ...mit einem oder mehreren Parametern zur Eingabe...  
`public void methode(Datentyp1 attribut1, Datentyp2 attribut2, ...)`
- ...mit einem Parameter zur Ausgabe...  
`public DatentypRückgabe methode( ) ...`
- ...mit einem Parameter zur Ausgabe und einem oder mehreren Parametern zur Eingabe...  
`public DatentypRückgabe methode(Datentyp1 attribut1, ...)`

...vorkommen. Der Wert, bzw. das Attribut, das bei einer Methode mit Rückgabeparameter zurückgegeben werden soll wird mit `return attribut;` übergeben.

Eine Methode wird dabei auch bzgl. ihrer Sichtbarkeit eingeschränkt (`public` und `private`). Eine einfache Methode ohne Ein- und Ausgabe könnte daher z.B. so deklariert werden (Innerhalb der Klasse, außerhalb der Main-Methode ☺):

```
public class Methoden{  
    public static void main(String args[ ]){  
        ...  
    }  
  
    public void vor(){  
        ...  
    }  
}
```

Zu beachten ist, Variablen/Attribute, die in einer Methode deklariert werden, stehen den anderen Methoden nicht automatisch zur Verfügung. Sie müssen ggf. als Parameter mit übergeben werden. Attribute, die außerhalb jeder Methode (aber innerhalb der Klasse) deklariert werden stehen allen Methoden zur Verfügung. Dies sollte aber die Ausnahme sein, z.B. bei Konstanten, da ansonsten jede Methode beliebig Änderungen am Wert des Attributs vornehmen kann und somit die Fehleranfälligkeit erhöht wird.

### Aufgabe 1

Schreibe ein Java-Programm `Kreisflaeche`, das den Radius eines Kreises als Eingabe entgegennimmt und die zugehörige Kreisfläche ( $A = \pi \cdot r^2$ ) als Ergebnis ausgibt. Dabei soll folgendes beachtet werden:

- $\pi$  ist als Konstante für alle Methoden verfügbar.
- Die Fläche wird durch eine eigene Methode `flaeche` berechnet.

Programmiere das Programm zweimal (`Kreisflaeche` und `Kreisflaeche2`, einmal soll `flaeche` das Ergebnis direkt ausgeben, das andere Mal nur berechnen, die Ausgabe übernimmt die `main`-Methode).

### Aufgabe 2

Erweitere dein Programm aus Aufgabe 1 (die Variante, bei der die `main`-Methode für die Ausgabe zuständig ist) um folgende Funktion:



Nach Eingabe des Radius soll der User eine Auswahl treffen können, ob er die Fläche, den Umfang ( $U = d \cdot \pi$ , bzw.  $U = 2 \cdot r \cdot \pi$ ) oder das Kugelvolumen ( $V = \frac{4}{3} \cdot \pi \cdot r^3$ ) berechnet haben möchte. Es soll jeweils eine eigene Methode zur Berechnung erzeugt werden, die Ausgabe übernimmt weiterhin die main-Methode.

### Aufgabe 3

Das Programm Multiplikation soll zwei Ganzzahlen als Eingabe entgegennehmen und durch eine geeignete Methode die beiden Zahlen multiplizieren. Beachte dabei, dass du die Multiplikation nicht einfach durch \* vornehmen sollst, sondern ein geeignetes Iterations-Verfahren in deine Methode implementierst.

### Arrays als Übergabe- und Rückgabeparameter

Auch ein Array kann als Eingabe, bzw. Ausgabe einer Methode dienen. Wichtig dabei ist, dass klar ist, an welchen Stellen eckige Klammern gesetzt werden müssen und wo sie nicht gesetzt werden dürfen:

- Bei der Rückgabe kommen die eckigen Klammern an den Datentyp  
`public int[] methode() ...`
- Bei der Eingabe wie gehabt an den Namen des Attributs  
`public void methode(int lottoZahlen[], ...) ...`
- Beim return wird nur der Name des Attributs angegeben  
`return lottozahlen;`

### Aufgabe 4

Ein Array mit Integerwerten soll sortiert werden und nach dem Sortiervorgang wieder ausgegeben werden. Löse die Aufgabe mit geeigneten Methoden.

- a) Die Ausgabe des sortierten Arrays erfolgt in einer Methode.
- b) Der Sortiervorgang erfolgt in einer Methode.