

## Variablen

Sobald während der Laufzeit eines Programmes irgendwelche Daten gespeichert werden müssen – und das kommt in fast jedem Programm vor – kommen Variablen ins Spiel. Bildlich gesprochen ist eine Variable eine Schachtel, in der etwas aufbewahrt werden kann.

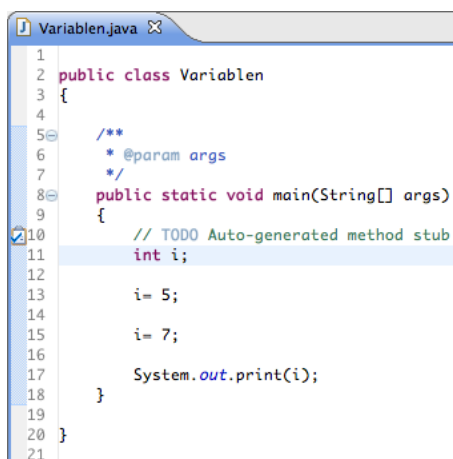
Erläuterungen	Informatische Sprechweise	Bildliche Sprechweise
Variablen können nicht einfach verwendet werden; sie müssen vor der ersten Verwendung eingerichtet worden sein.	Eine Variable muss deklariert werden.	Eine Schachtel muss vorhanden sein.
Bei der Deklaration erhält die Variable einen Namen.	Namen für Variablen sind z.B. i oder j oder zaehler oder obergrenze ...	Auf die Außenseite der Schachtel wird mit Filzstift eine Bezeichnung geschrieben.
Bei der Deklaration wird der Typ der Variablen festgelegt.	Gängige Typen für Variablen sind z.B. int für ganze Zahlen, double für Gleitkommazahlen, char für Zeichen, usw.	Es gibt keine Universal-schachteln für alles, was man vielleicht je aufbewahren möchte; jeder Schachteltyp kann nur einen bestimmten Inhaltstyp aufnehmen.
Was passiert bei der Deklaration?	<code>int i;</code> legt eine Integer-Variable mit dem Namen i an. Zur Laufzeit des Programmes wird im Hauptspeicher für diese Variable ausreichend Platz reserviert und das Programm merkt sich, wo das ist.	Es wird eine Schachtel, die den Typ int aufnehmen kann mit i beschriftet und ins Regal gestellt.

Mit Variablen kann man im Wesentlichen zwei Dinge tun:

- Man kann einen Wert in der Variable speichern.
- Man kann nachschauen, welcher Wert in einer Variablen gespeichert ist.

Dabei ist immer sauber zu trennen zwischen dem Namen der Variablen, dem Typ der Variablen und dem Wert, der in der Variable gespeichert ist.

Ein erstes einfaches Programm zum Umgang mit Variablen kann wie folgt aussehen:



```
1
2 public class Variablen
3 {
4
5     /**
6      * @param args
7      */
8     public static void main(String[] args)
9     {
10        // TODO Auto-generated method stub
11        int i;
12
13        i = 5;
14
15        i = 7;
16
17        System.out.print(i);
18    }
19 }
20 }
21
```

- In Zeile 11 wird die Integer-Variablen `i` deklariert.
- In Zeile 13 wird dieser Variablen der Wert 5 zugewiesen. Für eine Zuweisung wird das Gleichheitszeichen verwendet – es handelt sich dabei aber *nicht* um eine Gleichung, sondern das was rechts vom Gleichheitszeichen steht wird der links vom Gleichheitszeichen stehenden Variablen als Wert zugewiesen.
- In Zeile 15 wird derselben Variablen der Wert 7 zugewiesen. Was ist nun in der Variablen `i` gespeichert?
  - Immer noch die 5, weil die schon drin stand und somit kein Platz mehr für die 7 war?
  - Die 7, weil die 5 einfach weggeworfen wurde?
  - 5 und 7, weil zur schon gespeicherten 5 die 7 auch noch in der Variablen abgelegt wurde?
  - 12, weil der Computer die Werte einfach addiert hat?
- In Zeile 17 wird der Wert dieser Variablen ausgegeben – damit sieht man auch, welche der genannten Varianten zutrifft. Bei der Ausgabe des Wertes einer Variablen muss in der print-Anweisung einfach die Variablen mit ihrem Namen genannt werden. Die print-Anweisung gibt dann den Wert der Variablen aus.

Halte hier deine Beobachtung fest:

Eine Frage ist noch zu klären: Was enthält eine Variable, wenn sie zwar deklariert wurde, ihr aber noch kein Wert zugewiesen wurde?

Kommentiere dazu testweise die beiden Zuweisungen aus:

```
int i;  
//i= 5;  
//i= 7;  
System.out.print(i);
```

Was passiert jetzt? Halte deine Beobachtungen hier fest:

Häufig benötigt wird das Vertauschen des Wertes zweier Variablen miteinander. Gegeben sei das folgende Programmfragment:

```
int a, b;  
  
a= 10;  
b= 5;
```

Durch Ergänzen von Print-Anweisungen kann leicht überprüft werden, welcher Wert in welcher Variable steht.

Die Aufgabe ist nun, die Werte der beiden Variablen miteinander zu vertauschen, so dass eine Print-Anweisung dann für die Variable a den Wert 5 und die Variable b den Wert 10 ausgibt. Notiere hier deine Lösung:

Werden Variablen gleichen Typs mit Hilfe eines Containers zusammengefasst, dann spricht man von einem Array. Ein Array wird wie folgt angelegt:

```
int[] zahlen= new int[10];
```

Für das oben gezeigte Array gilt:

- Es trägt den Namen `zahlen`.
- Es ist vom Typ `int`, also jedes Arrayelement kann eine ganze Zahl speichern.
- Es hat 10 Elemente, kann also 10 Zahlen speichern – in jedem Arrayelement eine. Die laufenden Nummern der Arrayelemente gehen von 0 bis 9.

Die einzelnen Arrayelemente werden über ihre Nummer – angegeben in eckigen Klammern – angesprochen:

```
zahlen[0]= 3;  
zahlen[1]= 7;  
zahlen[2]= 10;  
zahlen[3]= 1;  
zahlen[4]= 6;  
zahlen[5]= 4;  
zahlen[6]= 8;  
zahlen[7]= 2;  
zahlen[8]= 9;  
zahlen[9]= 5;
```

Im obigen Programmcode wird jedem Arrayelement des Beispielarrays ein Wert zugewiesen. Die Nummer eines Arrayelementes kann selbst wiederum der Wert einer Variable sein, wie das folgende Beispiel zeigt:

```
System.out.print(zahlen[i]);
```

In diesem Beispiel wird der Wert des *i*-ten Arrayelementes ausgegeben – dazu muss natürlich erst der Wert der Variablen *i* ermittelt werden.

Häufig benötigte Typen für Variablen sind:

- `int` für ganze Zahlen
- `double` für Dezimalzahlen