

Datentypen und Variablen

Variablen kann man sich gewissermaßen als ein Behältnis für veränderliche Daten vorstellen. Wer zuvor noch nie von Variablen gehört hat, dem hilft der Vergleich mit einem Eimer möglicherweise am besten weiter. Der Eimer kann auch unterschiedlichen Inhalt haben, z.B. Sand oder Wasser. So ist es auch mit den Variablen. Sie können z.B. Zahlen oder Zeichenketten beeinhalteten. Da Java eine streng typisierte Sprache ist, muss bei der Deklaration jeder Variable der **Datentyp** angegeben werden. Über den Datentypen wird festgelegt, welche Werte die Variable annehmen kann. Für unseren Eimer würde das heißen, dass auf diesem ein Schild mit der Aufschrift "Sand" bzw. "Wasser" aufgeklebt werden müsste.

Die **Deklaration** einer Variablen erfolgt in Java wie folgt:

```
Datentyp Variablenname; //Variablendeklaration
```

So weisen wir der Variablen einen Wert zu. Die erste Wertzuweisung wird als **Initialisierung** bezeichnet. Diese ist erforderlich bei komplexen Datentypen, damit man mit der Variable arbeiten kann. Sollte auf die Variable zugegriffen werden, ohne dass zuvor eine Initialisierung erfolgt ist, wird eine Fehlermeldung ausgegeben. Einfache Datentypen benötigen nicht zwingend eine Initialisierung, da sie standardmäßig durch den Compiler mit dem Wert 0 initialisiert werden. Jedoch ist es empfehlenswert jede Variable so früh wie möglich zu initialisieren.

```
Variablenname = Wert; //Initialisierung
```

Die Deklaration und die Initialisierung wird häufig auch in einem Schritt erledigt:

```
Datentyp Variablenname = Wert; /*Deklaration und Initialisierung */
```

Man unterscheidet in Java **primitive** (auch als einfache oder elementare bezeichnet) und **komplexe Datentypen**.

In Java gibt es acht **primitive Datentypen**:

- Der einfachste Datentyp ist **boolean**. Ein Attribut vom Datentyp boolean kann genau zwei Werte annehmen: true (wahr) und false (falsch). Der Datentyp boolean wird für bedingte Abzweigungen sehr häufig verwendet, mit denen wir uns in einem späteren Kapitel beschäftigen wollen.
- Ein weiterer einfacher Datentyp ist der sogenannte **Character**, in Java durch den Datentyp **char** dargestellt wird. Ein Attribut von diesem Datentyp kann alle Unicode-Zeichen enthalten. Die maximale Länge umfasst 16 Bit. Der Wertebereich reicht von 0 bis 65535.
- Der Datentyp **byte** ist eine mit Vorzeichen behaftete Ganzzahl. Der Wertebereich (8 Bit) reicht von -128 (-2^7) bis +127 ($(2^7) - 1$). Dieser Datentyp wird häufig im Bereich der Dateneingabeverarbeitung oder der grafischen Datenverarbeitung verwendet. Dazu verwendet man häufig sogenannte Byte-Arrays. Arrays werden in einem späteren Kapitel genau behandelt.
- Ein weiterer Ganzzahl-Datentyp in Java heißt **short**. Der Wertebereich (16 Bit) hier ist doppelt so groß wie der des Datentyps byte. Dezimal ausgedrückt fängt der vorzeichenbehaftete bei -32768 (-2^{15}) an und endet bei 32767 ($(2^{15}) - 1$).
- Als **int** wird ein weiterer vorzeichenbehafteter Ganzzahl-Datentyp bezeichnet. Die Größe eines int-Wertes liegt bei 32 Bit. Er umfasst einen Wertebereich von dezimal -2147483648 (-2^{31}) bis 2147483647 ($(2^{31}) - 1$).
- Der größte vorzeichenbehaftete Ganzzahl-Datentyp nennt sich in Java **long**. Dieser Datentyp ist doppelt so groß wie sein Vorgänger und umfasst 64 Bit. Der Wertebereich liegt somit zwischen -9223372036854775808 (-2^{63}) und 9223372036854775807 ($(2^{63}) - 1$).

- In Java gibt es zwei Datentypen für Fließkommazahlen. Der kleinere Fließkomma-Datentyp heißt **float**. Er hat eine Größe von 32 Bit und der Wertebereich reicht von $-3,4 * 10^{38}$ bis $3,4 * 10^{38}$.
- Der größere Fließkomma-Datentyp wird als **double** bezeichnet. Double besitzt eine Größe von 64 Bit und der Wertebereich erstreckt sich von $-1,7 * 10^{308}$ bis $1,7 * 10^{308}$.

Beispiel: Wenn wir eine Variable mit dem Namen "varZahl" vom Typ int deklarieren und direkt mit der Zahl 23 belegen möchten, sieht dies so aus:

```
int varZahl = 23;           //Deklaration der Variablen "varZahl"
```

Später möchten wir den Wert der Variablen auf 42 ändern:

```
varZahl = 42;           /*Variable "varZahl" bekommt den Wert 42 zugewiesen */
```

Neben den primitiven Datentypen gibt es auch komplexe Datentypen. Jede Klasse, die über mehrere Attribute verfügt, definiert einen komplexen Datentypen. Als komplex werden sie deshalb bezeichnet, weil sie sich im Gegensatz zu den primitiven Datentypen aus verschiedenen Elementen zusammensetzen. Einer der bekanntesten komplexen Datentypen ist der Datentyp String. Der String-Datentyp repräsentiert Zeichenketten.

Quelle: <https://www.java-tutorial.org/datentypenundvariablen.html> abgerufen am 01.10.2020