

Einiges zu Datentypen in Java

| Typ | Länge | Wertebereich |
|---------|-------|---|
| boolean | 2 | true oder false |
| char | 2 | Unicode-Zeichen |
| byte | 1 | -2^7 – $2^7 - 1$ |
| short | 2 | -2^{15} – $2^{15} - 1$ |
| int | 4 | -2^{31} – $2^{31} - 1$ |
| long | 8 | -2^{63} – $2^{63} - 1$ |
| float | 4 | $\approx 1,4\text{E}-45$ – $3,4\text{E}+38$ |
| double | 8 | $\approx 4,9\text{E}-324$ – $1,8\text{E}+308$ |

- **Nur Deklaration**
 - `int counter;`
- **Deklaration mit Wertzuweisung**
 - `int counter = 0;`
 - `boolean allesKlar = true;`
- **Gleich mehrerer Variablen deklarieren**
 - `double a = 10, b, c, d = 5;`

- Membervariablen von Objekten, statische Variablen und Felder werden standardmäßig mit `null` oder `false` belegt.
- Für Lokale Variablen gilt dies nicht
 - Beispiel:

```
public static void main( String[] args ) {  
    int nanu, willi;  
    System.out.println( nanu ); // Fehler  
    nanu = 0;  
    if ( nanu == 1 )  
        willi = 2;  
    System.out.println( willi ); // Fehler  
}
```

- Lokale Variablen sind nur im jeweiligen Block (Bereich zwischen zwei "{}") gültig.

– Beispiel:

```
public static void main( String args[] )
{
    int i;
    {
        int k; // k gilt nur in dem Block
        int i; // das würde hier nicht gehen
        k = 1;
    }
    k = 2; // k gibt es hier nicht mehr!
}
```

- Ganzzahlige Literale sind in Java von Natur aus immer `int`

- Beispiel:

```
long l = 123456789098
```

Führt zu einem Compilerfehler (The literal 123456789098 of type int is out of range)

- Richtig

```
long l = 123456789098L
```

- Daten eines kleineren Datentyps werden automatisch (implizit) dem größeren angepasst.

– Beispiel:

```
int z = 12345;  
//implizite Anpassung  
float d = z;
```

- Ein größerer Typ kann einem kleineren Typ (explizit) mit möglichem Verlust von Informationen zugewiesen werden.

– Beispiel:

```
short s = ( short ) 510.2;    // 01 1111 1110
byte b = ( byte ) s;         //      1111 1110
System.out.println(s);
System.out.println(b);
```

Ausgabe: 510

-2

- Wertzuweisungen an `char` immer mit einfachen Anführungszeichen (Apostroph)

- Beispiel: `char c = 'a';`

- Bei `Strings` immer doppelte Anführungszeichen (Zollzeichen)

- Beispiel:

```
String s = "Heute ";  
s = s + "schon ";  
s += "gelacht?";  
System.out.println(s);
```

Ausgabe: Heute schon gelacht?