

Variablentypen

In Java müssen bekanntlich (siehe oben: Kapitel Methoden) alle Komponenten *innerhalb* einer Klasse deklariert werden. Außerhalb einer Klasse ist grundsätzlich keinerlei Deklaration erlaubt. Daraus geht hervor, daß es keine externen oder globalen Variablen geben kann wie in C / C++. Wenn so etwas wie eine globale Variable in einem Javaprogramm benötigt wird, dann muß sie als statische Variable in einer Klasse deklariert werden. Sie kann dann über ihren Klassennamen angesprochen werden (siehe oben Schlüsselwort `static`).

Der genaue Ort der Variablendeklaration innerhalb einer Klasse ist nicht gleichgültig, sondern entscheidet über die grundsätzliche Bedeutung dieser Variablen (*Deklarationsprinzip*).

Objektdaten

Daten, die *außerhalb von Methoden* definiert werden, sind grundsätzlich Objektvariablen (Attribute, Instanzvariablen, Felder¹) bzw. Klassenvariablen. Sie gehören immer zu einem Objekt bzw. zur Klasse selbst.

```
class Test {  
  
    int var1;           // Objektvariablen  
    double var2;  
    ...  
}
```

Anmerkung zur Position: Es gibt unterschiedliche Auffassung darüber, ob die Attribute *vor* oder *nach* den Methoden deklariert werden sollten. In den [Java Code Conventions](#) wird die Deklaration am Beginn der Klasse vorgeschlagen. Dies hat beim Analysieren (Durchlesen) der Klasse den Vorteil, daß die in den Methoden benutzten Variablen bereits bekannt sind und nicht erst am Ende der Klasse nachgeschaut werden. Nach diesem Prinzip wird hier verfahren.

Objektvariablen existieren nur in Verbindung mit einem Objekt und werden über dessen Namen aufgerufen. Gibt es kein Objekt, dann gibt es auch keine Variable. Jedes Objekt besitzt eine Kopie, die einen eigenen Inhalt / Wert hat (siehe Kapitel Klassen - Objekte). Objektvariablen müssen nicht ausdrücklich initialisiert werden, sondern sind automatisch mit dem zum Datentyp passenden Null-Wert vorbelegt (`false`, `0`, `0.0` bzw. `'`).

¹ Der Begriff `Feld` für ein Attribut ist mißverständlich, da er auch für einen Vektor gebraucht wird. Er wird deshalb hier nicht verwendet

Klassenvariablen gehören nicht zu einem einzelnen Objekt, sondern zur Klasse. Sie existieren unabhängig von allen eventuell vorhandenen Objekten und haben für alle Objekte ein und denselben Wert. Klassenvariablen werden durch das Schlüsselwort `static` vor dem Datentyp gekennzeichnet und durch den Klassennamen aufgerufen (siehe Kapitel Zugriffsarten).

```
class Test {  
  
    static int var1;           // Klassenvariablen  
    static double var2;  
  
}
```

Lokale Daten

Daten, die innerhalb einer Methode oder in einem Anweisungsblock zwischen geschweiften Klammern `{ }` deklariert werden, sind **lokale Variablen**. Sie haben mit Objekten nichts zu tun und existieren nur in dem jeweiligen Anweisungsblock. Lokale Variablen sind nur innerhalb des Anweisungsblocks, in dem sie deklariert wurden, gültig. Natürlich gilt dies auch für Konstanten.

```
class Test {  
  
    void methode() {  
        int var1;           // Lokale Variablen  
        double var2;  
    }  
}
```

Für lokale Daten gelten folgende Regeln:

- Lokale Daten sind in anderen Methoden derselben Klasse nicht sichtbar.
- Lokale Daten können nicht über ein Objekt angesprochen werden.
- Lokale Daten sind nicht automatisch mit Null-Werten initialisiert und dürfen nicht verwendet werden, bevor sie initialisiert wurden.
- Namen von lokalen Daten dürfen in einem inneren Block dieser Methode NICHT nochmals deklariert werden.