

## 6.1.3 Klassen und Instanzvariablen

Dienstag, 22. November 2022 09:44

### 4.1.3. Klassen und Instanzvariablen

Python unterscheidet zwei Arten von Variablen, die innerhalb von Klassen verwendet werden können:

#### **Instanzvariablen:**

Alle Variablen, denen `self` vorangestellt wird, sind Instanzvariablen. Sie werden also einer bestimmten Instanz (einem konkreten Objekt) zugeordnet. Von außen (also von Code, der sich außerhalb der Klasse befindet) erfolgt der Zugriff auf Instanzvariablen in der Form `obj.varname`.

#### **Klassenvariablen:**

Variablen, die auf Klassenebene definiert werden, gehören zur Klasse, nicht zu einzelnen Objekten. Egal wie viele Objekte es zu einer Klasse gibt – der Wert der Klassenobjekte gilt für alle Objekte gemeinsam. Möchten wir zum Beispiel einen Zähler realisieren, der mit jeder Objektgenerierung hochgezählt wird, benötigen wir eine solche Klassenvariable. Hier ist synonym auch die Rede von einer statischen Variable. Aus Instanzen ist der Zugriff auf Klassenvariablen möglich. Von außen erfolgt der Zugriff auf Klassenvariablen in der Form `Klassenname.varname`.

#### **Lokale Variablen:**

Zusätzlich gibt es wie bei Funktionen lokale Variablen: Wenn also eine Variable ohne vorangestelltes `self` in einer Methode verwendet wird, dann beschränken sich die Gültigkeit und Lebensdauer dieser Variablen auf die Methode.



Entscheiden Sie im untenstehenden Code, um welche Variable es sich handelt. Notieren Sie Ihr Ergebnis als Kommentar im Code.

```
Beispiel:
class MyClass():
    magicNumber=42      #klassenvariable
    def __init__(self, zahl_1, zahl_2):
        somevar=123     #lokale variable
        self.zahl_1=zahl_1    #instanzvariable
        self.zahl_2=zahl_2    #instanzvariable
```