

## 6.2.4 Public, Protected und Private Instanzvariablen

Mittwoch, 1. März 2023 10:56

#### 6.2.4. Public, Protected und Private Instanzvariablen

Manchmal macht es Sinn, dass gewisse Instanzvariablen nicht ohne Überprüfung einfach geändert werden können oder sie erst gar nicht gegen Aussen sichtbar sein sollten. Nehmen wir als Beispiel folgende Eingabe:

```
katzeChiChi.weight= -10
```

`katzeChiChi` ist eine Instanz der Klasse `Cat`, wie wir es oben schon definiert haben. Hier wurde nun dem Gewicht der Katze `katzeChiChi` ein negativer Wert zugeordnet, was in der Realität gar nicht vorkommen kann.

Möchte man verhindern, dass die Instanzvariablen einer Klasse von außen ohne weiteres geändert oder gar gelesen werden kann, gibt es zwei Möglichkeiten, die Python einem zur Verfügung stellt.

1. Jedes Attribut, welches mit genau einem Unterstrich beginnt, ist **protected**. In diesem Fall kann das Attribut zwar immer noch gelesen und verändert werden, aber durch den Unterstrich wird kommuniziert, dass dies *verboten* oder *nicht erwünscht* ist.
2. Wenn der Name eines Attributes mit zwei Unterstrichen beginnt, wird der Zugriff von Außen durch Python weiter erschwert. In diesem Fall kann nur noch über `obj._classname_attribut` darauf zugegriffen werden. Ein solches Attribut wird **private** genannt.

Dazu sehen wir uns folgende Beispielklasse an:

```
1 class A():
2     def __init__(self):
3         self.__priv = "Ich bin privat"
4         self._prot = "Ich bin protected"
5         self.pub = "Ich bin öffentlich"
```

Übersicht:

Name	Bezeichnung	Bedeutung
pub	Public	Von außen lesbar und veränderbar
_prot	Protected	Von außen lesbar und änderbar, sollte jedoch nicht gemacht werden.
__priv	Private	Von außen lesbar, wenn Klassenname angefügt wird.