

Objektorientierte Programmierung

Eine kleine Einführung in wichtige Konzepte, Part1

Michael Greminger 2021-03-17

Klassen (1)

- Objektorientierte Programmierung (OOP) versucht, reale Dinge abzubilden
 - Adresse
 - Window
 - Konto
- Dafür gibt es in den wesentlichen Programmiersprachen das Konstrukt einer Klasse
- In Python

```
class Window:
```

```
    ...
```

Klassen (2)

- Eine Klasse besteht aus

- Attributen

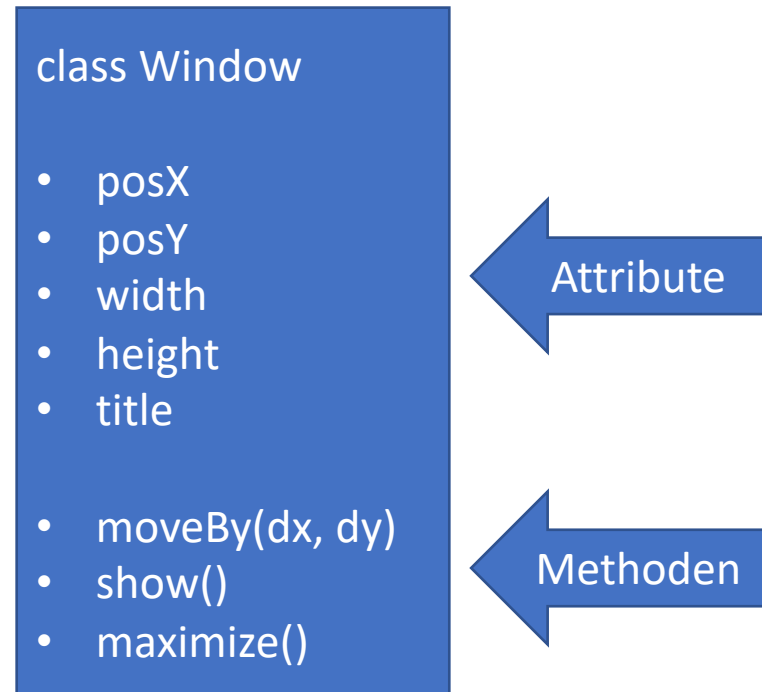
- «Variablen»

- Werden so in Python genannt. In anderen Programmiersprachen werden andere Bezeichnungen benutzt, zb data member in C++

- Methoden

- «Funktionen»

- (in C++ zb. «member functions»)



Instanzen

- Damit man mit einer Klasse “arbeiten” kann, muss man Instanzen von ihr machen

class Window

- posX
- posY
- width
- height
- title

- moveBy(dx, dy)
- show()
- maximize()

instance of class Window

- 0
- 0
- 500
- 300
- “Python”

- moveBy(dx, dy)
- show()
- maximize()

instance of class Window

- 456
- 356
- 200
- 300
- “Microsoft Word”

- moveBy(dx, dy)
- show()
- maximize()

- Eine Klasse braucht “keinen” Speicherplatz, nur die Instanzen!

Konkret in Python

```
1 class Rect:
2     def area(self):
3         return self.width * self.height
4
5     def setWidth(self, w):
6         self.width = w
7
8     def setHeight(self, h):
9         self.height = h
10
11 myRect1 = Rect()
12 myRect1.setWidth(3.4)
13 myRect1.setHeight(2.4)
14
15 myRect2 = Rect()
16 myRect2.setWidth(123.4)
17 myRect2.setHeight(34.4)
18
19 print myRect1.area()
20 print myRect2.area()
```

Ein paar Bemerkungen

- Eine Methode (einer Klasse) kann auf folgende Variablen zugreifen:
 - Attribute (von sich “selbst”!)
 - Argumente
 - (Globale Variablen)
 - ...
- Oft möchte man beim Instanzieren einer Klasse gleich Argumente übergeben (oder die Attribute initialisieren). Dies geschieht wie folgt (in anderen Sprachen “Konstruktor” genannt)

```
11     def __init__(self, w, h):  
12         self.width = w  
13         self.height = h
```