



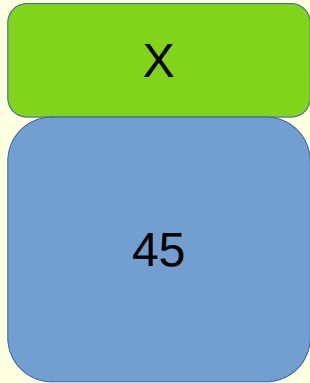
Seiteneffekte

Seiteneffekte



Unerwartete Veränderungen in Variablen

- insbesondere ohne sichtbaren Zugriff auf Variable

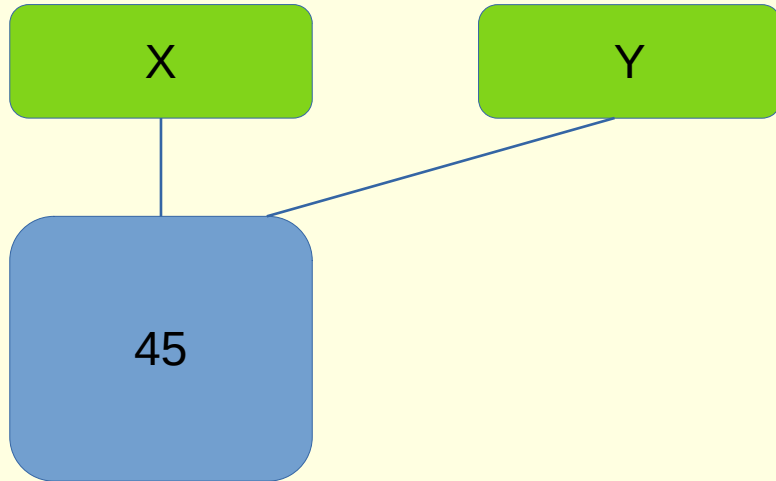


Seiteneffekte



Unerwartete Veränderungen in Variablen

- insbesondere ohne sichtbaren Zugriff auf Variable

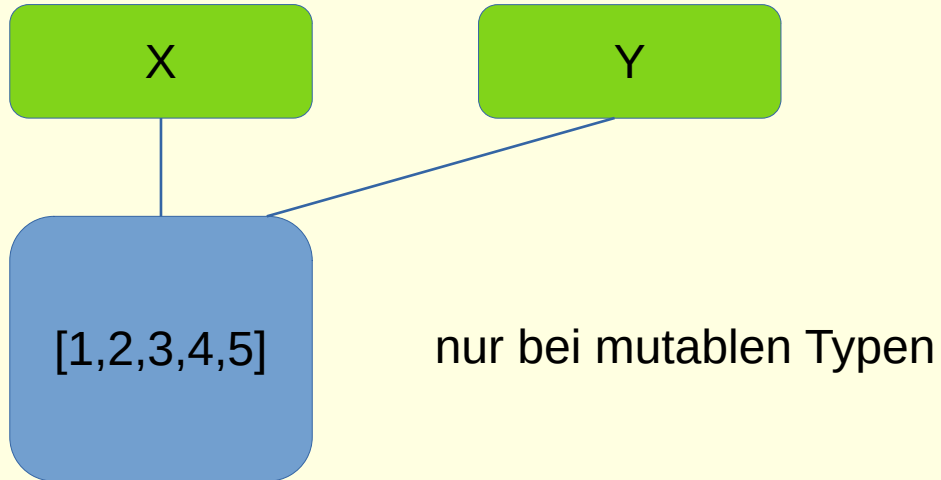


Seiteneffekte



Unerwartete Veränderungen in Variablen

- insbesondere ohne sichtbaren Zugriff auf Variable



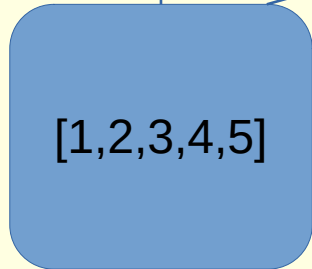
Seiteneffekte



"Call by Reference"

Hauptprogramm

$f(y=x)$

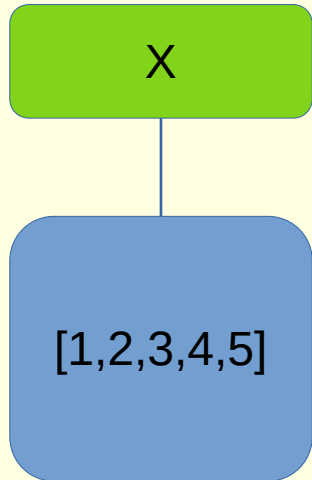


Seiteneffekte

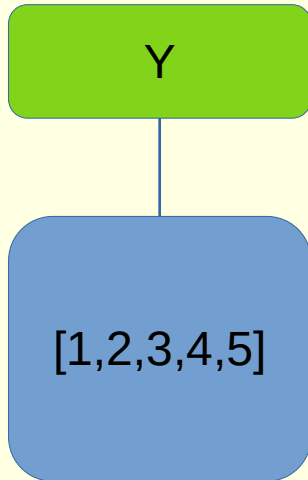


"Call by Value"

Hauptprogramm



$f(y=x)$



- erstellt Kopie
- keine Seiteneffekte
- kostet Ressourcen

Seiteneffekte



Die Funktion `id()`

- jedes Objekt hat eindeutige ID

```
>>> a = 5

>>> b = 5

>>> id(a)
109059988395752

>>> id(b)
109059988395752
```

Seiteneffekte



```
>>> def f(a, b):  
...     print(id(a))  
...     print(id(b))
```

```
>>> p = 1  
>>> q = [1,2,3]
```

```
>>> id(p)  
134537016
```

```
>>> id(q)  
134537004
```

```
>>> f(p, q)  
134537016  
134537004
```


Seiteneffekte



Beispiel für Seiteneffekt

```
>>> def f(liste):  
...     liste[0] = 42  
...     liste += [5,6,7,8,9]  
  
>>> zahlen = [1,2,3,4]  
  
>>> f(zahlen)  
  
>>> zahlen  
[42, 2, 3, 4, 5, 6, 7, 8, 9]
```

Seiteneffekte



Kopieren kann erzwungen werden

```
>>> zahlen = [1,2,3,4]
>>> f(zahlen[:])
>>> zahlen
[1, 2, 3, 4]
```

Seiteneffekte



Seiteneffekte bei Default-Parametern

```
>>> def f(a=[1,2,3]):  
...     a += [4,5]  
...     print(a)  
  
>>> f()  
[1, 2, 3, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5, 4, 5, 4, 5]
```

Seiteneffekte



Seiteneffekte bei Default-Parametern

```
>>> def f(a=[1,2,3]):  
...     a += [4,5]  
...     print(a)  
  
>>> f()  
[1, 2, 3, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5, 4, 5]  
  
>>> f()  
[1, 2, 3, 4, 5, 4, 5, 4, 5, 4, 5]
```

- Instanz für Default-Wert wird nur einmal erstellt
- nicht automatisch zurückgesetzt

Seiteneffekte



Lehren:

- für Parameter vorzugsweise immutable
- Zuweisungen an Parameter vermeiden
- evtl. Kopieren erzwingen