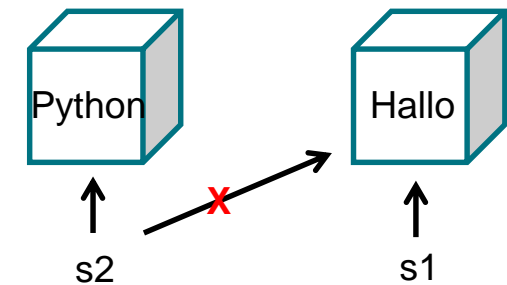


- In einem früheren Kapitel haben wir ein Beispiel ähnlich wie folgendes gesehen:

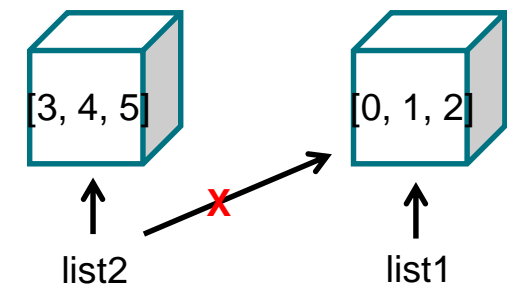
```
s1 = "Hallo"  
s2 = s1  
print(s1, id(s1), s2, id(s2))  
s2 = "Python"  
print(s1, id(s1), s2, id(s2))
```



Die Zeichenreihe "Hallo" wird nicht geändert.

- Ganze Zahlen, Fließkommazahlen, Zeichenketten und Tupel sind unveränderliche Daten.
- Im Prinzip gilt das auch für Listen. Beispiel:

```
list1 = [0, 1, 2]  
list2 = list1  
print(list1, id(list1), list2, id(list2))  
list2 = [3, 4, 5]  
print(list1, id(list1), list2, id(list2))
```

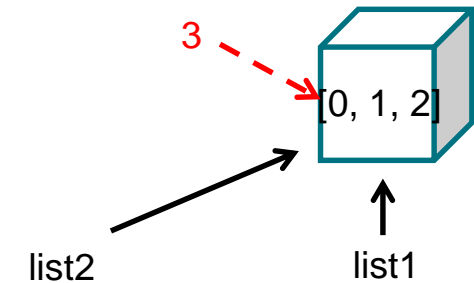


- Allerdings können Listenelemente geändert werden. Beispiel:

```
list1 = [0, 1, 2]
list2 = list1
print(list1, id(list1), list2, id(list2))
list1[0] = 3
print(list1, id(list1), list2, id(list2))
```

Damit wurde auch `list2[0]` geändert, auch wenn das im Programm nicht sofort ersichtlich ist.

- Die Änderung des Inhalts von `list1` hat als Seiteneffekt auch den Inhalt von `list2` geändert. Das kann zu schwerwiegenden Fehlern führen.




- Um dieses Problem zu vermeiden, kann die Liste kopiert werden, d.h. anstelle von

```
list1 = [0, 1, 2]
list2 = list1
```

kann

```
list1 = [0, 1, 2]
list2 = list1.copy()
```

verwendet werden. Statt der copy-Funktion kann man übrigens auch Slicing verwenden.

- Dieses Kopieren nennt man auch **Shallow Copy**.
- Das funktioniert aber nicht, wenn eine Liste eine Liste enthält. Dazu benötigt man **Deep Copy**, das rekursiv auch alle geschachtelten Listen kopiert. Mehr dazu in (Klein 2018), S. 111
-  Kapitel 13.1 – 13.4 in (Klein 2018)



- Welche der folgenden Anweisungen sind fehlerhaft (Klein 2018, S. 41)?

```
t = (4,7,9)
s = "Ich bin ein String"
l = [45,98,"787",[3,4]]
t2 = (4,8,[45,98])
print(t[0])
print(t[3])
print(t(3))
print(s[4])
s[4] = "x"
l[2][0] = "g"
l[3][0] = "g"
print(l)
t2[2][0] = 42
list2 = list1
```

- Welche Konsolausgabe hat das folgende Skript (Klein 2018, S. 42)?

```
x= [4, 8, 9]
title = "The Name of the Rose"
address = ["Frank Fitmann", "Feldstr. 19", "89000 München"]
nested = [(4,8,9)]
book = [title, "Umberto Eco"]
print(len(x), len(title), len(address), len(nested), len(book))
```