

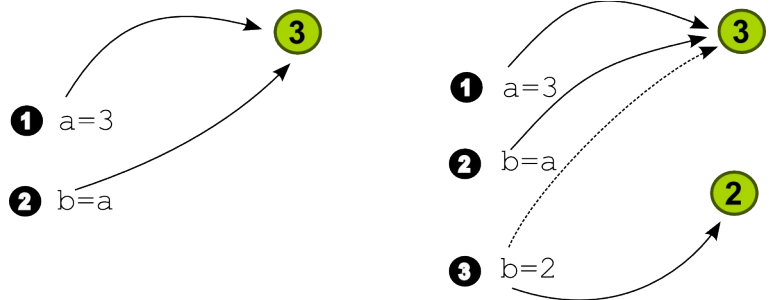
# Arbeitsblatt Referenzen: Lösungen

## Zuweisungen in Python

### Aufgabe 1

Schaue dir die folgende Folge von Anweisungen an, welche die Namen `a` und `b` vergeben. Im Diagramm siehst du, was hinter den Kulissen geschieht. Kläre zu deiner Zufriedenheit, weshalb die Zuweisung `b = 2` in Schritt ③ keinen Einfluss auf `a` hat.

```
>>> a = 3 ①
>>> b = a ②
>>> b = 2 ③
>>> b
2
>>> a
3
```

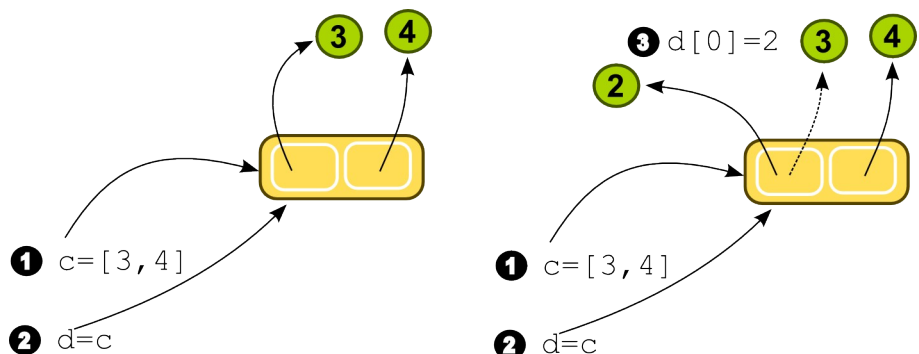


**Erklärung:** Die Zuweisung in Schritt ③ hat keinen Einfluss auf `a`, weil nur der Pfeil (d.h. die Referenz) von `b`, der nach Schritt ② auf die Zahl 3 verweist, zur Zahl 2 „umgebogen“ wird. Die Referenz von `a` auf die Zahl 3 wird dabei nicht angetastet.

### Aufgabe 2

Unten ist eine fast identische Anweisungsfolge wie in Aufgabe 1 zu sehen, aber diesmal wird mit Listen gearbeitet statt mit einfachen Zahlen. Erkläre, weshalb das Ergebnis sich von Aufgabe 1 unterscheidet (die Zuweisung an `d[0]` betrifft auch die mit `c` bezeichnete Liste) und zeichne dazu ein Diagramm analog zum Diagramm von Aufgabe 1:

```
>>> c = [3, 4] ①
>>> d = c      ②
>>> d[0] = 2    ③
>>> d
[2, 4]
>>> c
[2, 4]
```



**Erklärung:** Sowohl `c` wie auch `d` sind Namen für dieselbe Liste. Die Zuweisung `d[0]` ändert nichts am Pfeil von `d` zur Liste (d.h. `d` referenziert immer noch dieselbe Liste wie `c`). Die Zuweisung ändert vielmehr die Referenz, die vom ersten Element in der Liste ausgeht, und zwar wird diese Referenz von der Zahl 3 auf die Zahl 2 geändert.

Da `c` und `d` Synonyme für die gleiche Liste darstellen, sind die Änderungen an der Liste natürlich auch über den Namen `c` festzustellen.