

## 2. Gleichheit und Identität

In der objektorientierten Programmierung muss klar unterschieden werden, ob Objektreferenzen

- gleich oder
- identisch

sind, wenn sie auf Objekte des gleichen Typs (der gleichen Klasse) verweisen.

### Gleichheit

Objektreferenzen werden dann als gleich betrachtet, wenn sie auf Objekte vom gleichen Typ (der gleichen Klasse) verweisen.

Die Gleichheit wird mit der Funktion `isinstance` überprüft

Unter der Annahme, dass eine Klasse `Point` existiert, wird folgender Code ausgeführt:

```
if __name__ == "main":
    obj1 = Point(10,10)
    obj2 = Point(5,5)
    obj3 = obj1
    #
    if isinstance(obj1, Point):
        # hier kann nun Code folgen, der das Objekt nutzt
    else:
        # hier könnte eine Fehlermeldung folgen
```

Da `obj1` vom Typ `Point` ist, wird der Code nach dem `if`-Befehl ausgeführt. In diesem Beispiel liefert `isinstance` für alle 3 Objekte (`obj1`, `obj2` und `obj3`) den Wert `True`.

### Identität

Objektreferenzen werden dann als identisch betrachtet, wenn sie auf ein und das selbe Objekt verweisen.

Die Identität einer Referenz wird über den Operator `is` ermittelt.

```
if obj1 is obj2:
    # bei Identität von obj1 und obj2 wird dieser Code ausgeführt
else:
    print('obj1 und obj2 sind nicht identisch')
# - -
if obj1 is obj3:
    print('obj1 und obj3 sind identisch')
else:
    # sind obj1 und obj3 nicht identisch, wird dieser Code ausgeführt
```

Der erste Vergleich liefert `False`, da die Referenzen `obj1` und `obj2` auf unterschiedliche Objekte zeigen. Beim zweiten Vergleich wird `True` geliefert, da `obj1` und `obj3` auf ein und das selbe Objekt

zeigen.

---



© René Probst

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

[https://wiki.bzz.ch/modul/m320/learningunits/lu02/theorie/lu2-kapitel\\_2](https://wiki.bzz.ch/modul/m320/learningunits/lu02/theorie/lu2-kapitel_2)

Last update: **2024/03/28 14:07**

