

Merkblatt - Vergleichsoperatoren

May 7, 2020

0.0.1 Vergleichsoperatoren

Um die Bedingungen in den if-else-Strukturen besser zu verstehen, schauen wir uns **Vergleichsoperatoren** an. Das wird uns helfen, eine Vielfalt an Bedingungen formulieren zu können.

0.0.2 Ungleichheitsoperatoren und Bools

```
[1]: if 6 < 5:  
      print("JA")
```

Warum kommt es zu keiner Ausgabe mit print()? Schauen wir uns doch Ausdrücke mit Ungleichheitszeichen (<,>) im Detail an:

```
[5]: print(6 < 5)  
      print(5 < 6)
```

False
True

True und **False** (Großschreibung beachten!) sind weitere *feststehende Ausdrücke* in Python. Neben Strings (Zeichenketten), Ganzzahlen (Integer) und Fließkommazahlen (Floats) bilden sie einen weiteren Datentyp - den **Bool**.

```
[7]: b = False  
      print(b)
```

False

Genauer wird eine if-Bedingung also nur dann ausgeführt, wenn nach dem if ein Bool mit dem Wert True steht:

```
[8]: result = 5 < 6  
      if result:  
          print("5 ist kleiner als 6")
```

5 ist kleiner als 6

```
[10]: print(5 < 6)
```

True

0.0.3 Der Gleichheitsoperator

Neben Ungleichheiten können wir natürlich auch Gleichheiten abfragen, und zwar mit `==`

```
[11]: print(5 == 5)
      print(5 == 4)
```

True
False

```
[19]: if 5 == 5:
      print("5 ist 5")
```

5 ist 5

Mittels des Gleichheitsoperators können wir auch die Zustände *größer gleich* (`>=`) und *kleiner gleich* (`<=`) abfragen:

```
[18]: print(5 < 5)
      print(5 <= 5)
      print(5 >= 5)
```

False
True
True

0.0.4 Strings vergleichen

Wir können nicht nur Zahlen miteinander vergleichen, sondern auch Strings:

```
[15]: word = "Hallo"
      print(word == "Hallo")
      print(word == "Welt")
```

True
False

0.0.5 Der Ungleichheitsoperator

Auf Ungleichheit checken wir mit dem Zeichen `!=`

```
[16]: word = "Hallo"
      print(word != "Hallo")
      print(word != "Welt")
```

False
True

Auch Zahlen kann man auf Ungleichheit hin miteinander vergleichen:

```
[3]: zahl = 4
      print(zahl != 4)
      print(zahl != 5.5)
```

False
True

0.0.6 Spiel doch jetzt ein wenig mit dem, was du gelernt hast, herum:

- Formuliere eigene Vergleichsoperationen mit den Operatoren aus dieser Lektion! :-)

```
[ ]:
```