

5 Boolesche Ausdrücke

Sowohl die if Verzweigung als auch die while Schleife benötigen eine Bedingung. Im folgenden Kapitel wollen wir etwas genauer analysieren, wie solche Bedingungen formuliert werden können. Ein boolescher Ausdruck ist ein Ausdruck, der entweder wahr oder falsch ist, in Python True und False genannt. Dies sind auch gleich die einfachsten booleschen Ausdrücke.

True Dieser Ausdruck ist immer wahr.

False Dieser Ausdruck ist immer falsch.

5.1 Vergleiche

Wir können zwei Variablen var_1 und var_2 miteinander vergleichen, falls in beiden derselbe Datentyp gespeichert ist. Wenn verschiedene Datentypen gespeichert sind, werden die Werte immer als ungleich betrachtet. Es existieren die folgenden Vergleiche:

Vergleich	Bedeutung
var_1 == var_2	var_1 gleich var_2
var_1 > var_2	var_1 grösser als var_2
var_1 >= var_2	var_1 grösser gleich var_2
var_1 < var_2	var_1 kleiner als var_2
var_1 <= var_2	var_1 kleiner gleich var_2
var_1 != var_2	var_1 ungleich var_2

5.2 Boolesche Operatoren

Mit booleschen Operatoren können verschiedene boolesche Ausdrücke verkettet oder verneint werden. So können komplizierte Bedingungen wie „Ist A gleich B oder B gleich D“ formuliert werden. Es gibt die folgenden Operatoren:

not:

Kehrt den Wahrheitswert eines Ausdrucks um. Der Operator macht dasselbe wie das deutsche Wort nicht.

and:

Ist wahr, wenn die Ausdrücke links und rechts des Operators wahr sind. Es handelt sich um eine und Verknüpfung.

or:

Ist wahr, wenn der Ausdruck links oder der Ausdruck rechts des Operators wahr ist. Es ist auch wahr, wenn beide wahr sind. Es handelt sich um eine oder Verknüpfung.

Beispiel:

```
>>> 4 == 4 or 4 == 5
True
```

Dieses Beispiel ist wahr, da 4 entweder gleich 4 oder gleich 5 ist.

Hingegen ist das Ergebnis falsch, wenn man dieselbe Frage mit einem und stellt:

```
>>> 4 == 4 and 4 == 5
False
```

Denn vier kann nicht gleich vier und gleich fünf sein. Da vier nicht gleich fünf ist, ist aber die Frage danach, ob vier nicht gleich fünf sei wahr:

```
>>> not 4 == 5
True
```

5.2.1 Aufgaben

1. Entscheide für jede Codezeile, ob der Boolesche Ausdruck wahr oder falsch ist, ohne das Beispiel in die Python-Konsole einzugeben. Prüfe deine Antwort anschliessend mit der Konsole.

```
>>> 3 > 4
>>> "Hallo Welt" > "Hallo"
>>> 4 != 5
>>> "Hallo" == "Hallo Welt"
>>> "Hallo Welt" > "Hallo" and 3 > 4
>>> "Hallo Welt" > "Hallo" or 3 > 4
>>> not "Hallo Welt" >= "Hallo Welt"
>>> not not 5 == 5
>>> not 3 >= 4 and not 4 >= 5
```

2. Das folgende Programm benutzt zwei geschachtelte if Verzweigungen. Schreibe das Programm um, so dass es mit einer einzelnen Verzweigung auskommt:

```
1 zahl = int(input("Gib eine Zahl ein: "))
2
3 if zahl % 2 == 0:
4     if zahl % 3 == 0:
5         print("Die eingegebene Zahl ist durch drei und zwei teilbar.")
6     else:
7         print("Die eingegebene Zahl ist nicht durch drei und zwei teilbar.")
8 else:
9     print("Die eingegebene Zahl ist nicht durch drei und zwei teilbar.")
```

3. Schreibe ein Programm, welches prüft, ob ein Jahr ein Schaltjahr ist oder nicht. Verwende dabei im Gegensatz zum letzten solchen Programm nur genau eine if Verzweigung.