

Kontrollstrukturen

Python kennt die folgenden
Kontrollstrukturen:

if – elif – else

for

while

Kontrollstrukturen if

Die **if-Anweisung** erwartet ein Boolean. Wenn dieser True ist wird der Block hinter if ausgeführt, sonst wird er übersprungen.

Die **if-Anweisung** muss mit einem **:** abgeschlossen werden.

Der **if-Block** wird eingerückt.

```
>>> if True:
    print('if Block wird
    ausgeführt')

if Block wird ausgeführt

>>> if False:
    print('if Block wird
    ausgeführt')

    print('if Block wird nicht
    ausgeführt')

if Block wird nicht ausgeführt
```

Kontrollstrukturen - else

Wenn 2 verschiedene Blöcke ausgeführt werden sollen, so geht das mit der **else-Anweisung**.

Der **if-Block** wird ausgeführt wenn die Bedingung **True** ist, die **else-Anweisung** wenn sie **False** ist.

```
>>> if True:
    print('if-Block wird
ausgeführt')
    else:
        print('else-Block wird
ausgeführt')
```

if-Block wird ausgeführt

```
>>> if False:
    print('if-Block wird
ausgeführt')
    else:
        print('else-Block wird
ausgeführt')
```

else-Block wird ausgeführt

Kontrollstrukturen - elif

Wenn auf mehr als eine
Bedingung geprüft werden soll
geht das mit elif.

```
>>> x = 9
>>> if x < 10:
    print('x < 10')
elif x > 10:
    print('x > 10')
else:
    print('x = 10')
```

x < 10

```
>>> x = 12
>>> if x < 10:
    print('x < 10')
elif x > 10:
    print('x > 10')
else:
    print('x = 10')
```

x > 10

Kontrollstrukturen - for

Die **for**-Schleife ist dafür vorgesehen alle Elemente einer Sequenz zu bearbeiten.

```
>>> l = [1, 2, 3, 4, 5]
>>> for x in l:
    print(x)

1
2
3
4
5
>>>
```

Kontrollstrukturen – for - range

Auch die klassische for-Schleife aus C lässt sich in Python realisieren. Dazu wird die Funktion **range()** benutzt die eine Zahlenfolge liefert.

```
>>> for x in range(5):  
        print(x)  
  
0  
1  
2  
3  
4
```

Kontrollstrukturen - while

Die **while** Schleife erwartet ein Boolean. Ist es **True**, so wird der Schleifenblock durchlaufen. Bei **False** wird die Schleife verlassen.

```
>>> x = 0
>>> while x < 5:
        print(x)
        x += 1

0
1
2
3
4
```