

Die **range**-Anweisung definiert einen Bereich ganzer Zahlen.

range(10) definiert den Bereich 0, 1, . . . , 9
range(4,21) definiert den Bereich 4, 5, . . . , 20
range(4,21,3) definiert den Bereich 4, 7, 10, . . . , 16, 19
range(-4,3) definiert den Bereich -4, -3, -2, -1, 0, 1, 2

Allgemein gilt:

range(start, stop)
definiert den Bereich **start, , stop-1** ganzer Zahlen,

range(start, stop, step)
definiert den Bereich **start, . . .** mit der Schrittweite **step**, wobei die Zahl **stop** nicht mehr enthalten ist.

Erstellen einer Liste ganzer Zahlen

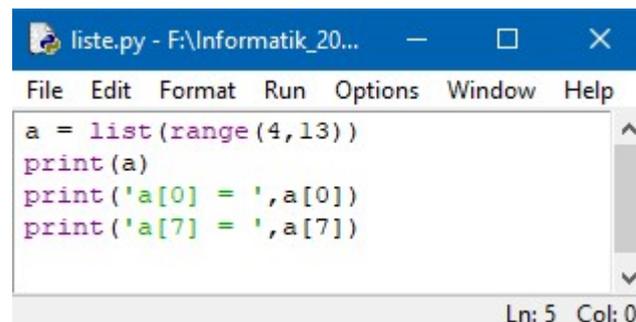
a = list(range(4,13)) erzeugt die Liste

[4, 5, 6, 7, 8, 9, 10, 11, 12];

die (in diesem Fall 9) Elemente dieser Liste heißen auch Komponenten, auf die man mit **a[0], a[1], . . . , a[8]** zugreifen kann.

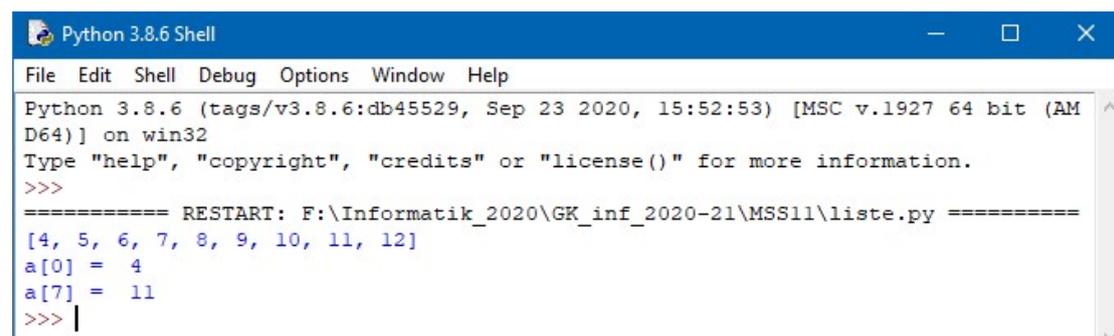
*Bemerkung: Unter einem **Feld** oder einem **array** verstehen wir eine Folge von Variablen gleichen Typs; mit vorstehendem Beispiel haben wir also ein array **a** ganzer Zahlen erzeugt mit den Komponenten **a[0], a[1], . . . , a[8]**.*

Python-Quelltext:



```
liste.py - F:\Informatik_20...
File Edit Format Run Options Window Help
a = list(range(4,13))
print(a)
print('a[0] = ',a[0])
print('a[7] = ',a[7])
Ln: 5 Col: 0
```

Ausgabe:



```
Python 3.8.6 Shell
File Edit Shell Debug Options Window Help
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: F:\Informatik_2020\GK_inf_2020-21\MSS11\liste.py =====
[4, 5, 6, 7, 8, 9, 10, 11, 12]
a[0] = 4
a[7] = 11
>>> |
```

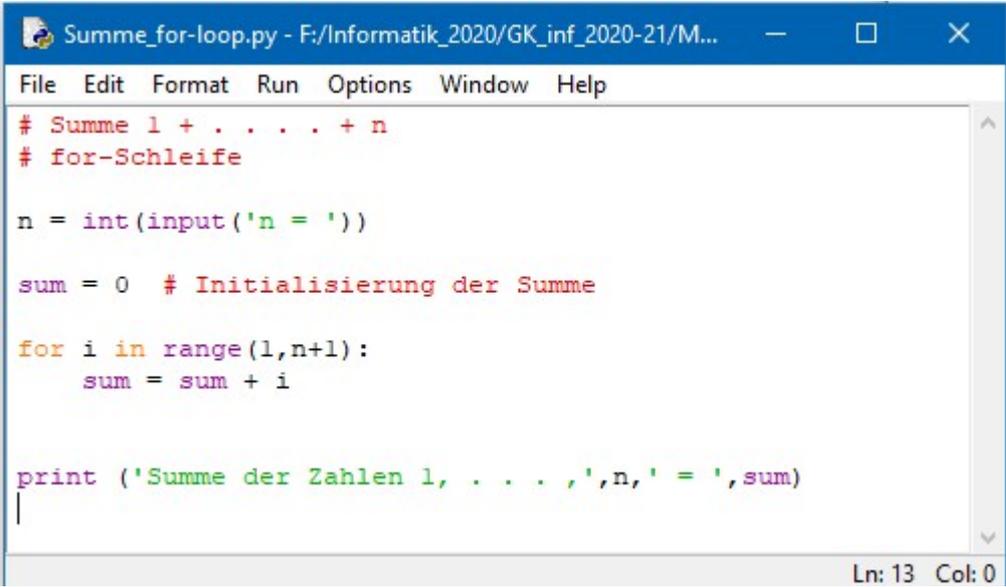
For-Schleife

Das Python-Programm

```
n = int(input('n = '))  
  
for i in range(1,n):  
    print(i)
```

gibt nach Eingabe der natürlichen Zahl n die Zahlen $1, 2, \dots, n-1$ aus; probiert es aus!

Algorithmus zur Bestimmung der Summe der Zahlen $1, \dots, n$ mit Verwendung einer for-Schleife:



```
Summe_for-loop.py - F:/Informatik_2020/GK_inf_2020-21/M...  
File Edit Format Run Options Window Help  
# Summe 1 + . . . . + n  
# for-Schleife  
  
n = int(input('n = '))  
  
sum = 0 # Initialisierung der Summe  
  
for i in range(1,n+1):  
    sum = sum + i  
  
print ('Summe der Zahlen 1, . . . ,',n,' = ',sum)  
|  
Ln: 13 Col: 0
```

Arbeitsaufträge:

1. Schreibe und teste ein Python-Programm, um die ersten 20 Zahlen der Siebener-Reihe auszugeben (also die Zahlen $7, 14, 21, \dots$).
2. Erstelle gemäß dem vorstehenden screenshot den Python-Programmtext zur Berechnung der Summe $1 + \dots + n$ und teste das Programm mit unterschiedlichen Eingaben.
3. Formuliere und teste ein Python-Programm, welches nach Eingabe der natürlichen Zahl k die Summe der ersten k ungeraden natürlichen Zahlen bestimmt, und zwar mit Verwendung einer for-Schleife.
4. Formuliere und teste ein Python-Programm, welches nach Eingabe der natürlichen Zahl n das Produkt der Zahlen $1, \dots, n$ zu berechnet (for-Schleife!).