

```
# Zuweisungen
k = 8, x = 3.2
x += 1, k*=2      # Kurzformen

# Variablennamen
entfernungZurWand = 3.2      # camel case
entfernung_zur_wand = 3.2    # snake_case

# print
print(k), print(k, x)

# print mit f-Strings
print(f'Das Doppelte von {x} ist {2*x}.')
print(f"Pi auf zwei Nachkommastellen: {pi:.2f}")

# Einzeiliger Kommentar
''' mehrzeiliger
Kommentar'''

# Arithmetische Operationen
+, -, *, /, ** Exponentiation
% modulo, // ganzzahlige Division

# Vergleichsoperatoren
i > j, i >= j, i < j, i <= j, i == j, i != j

# Boolesche Operatoren
and, or, not

# Input
s = input('Name: ')
k = int(input('Ganze Zahl: '))
x = float(input('Zahl: '))

# Bedingungen
if x > 10:
    print('A')
elif x > 0:
    print('B')
else:
    print('C')

# Bedingte Zuweisung
s = 'gerade' if x % 2 == 0 else 'ungerade'
```

```
# Strings
s = ''            # leerer String
s = 'Hallo' , s = "Hallo"
k = len(s)        # Länge des Strings
s = s1 + s2       # Konkatenieren
s1 = 3*s          # Vervielfachen

s1 = s.upper(), s.lower() # alles in Klein-/Großbuchstaben
k = s.count('abi')      # Vorkommen zählen
s1 = s.replace('abi','oha') # Teilstrings ersetzen

b = 'abi' in s          # Teilstring vorhanden?
b = 'abi' not in s      # Teilstring nicht vorhanden?

# Indexing bei Strings
s[0], s[1]              # erstes, zweites Zeichen
s[len(s)-1], s[-1]     # letztes Zeichen
s[-2]                  # vorletztes Zeichen

# Slicing bei Strings
s[:3], s[-3:]          # erste 3, letzte 3 Zeichen
s[2:], s[:-2]          # außer ersten 2, letzten 2 Zeichen
s[1:5]                 # Teilstring von Index 1 bis (ausschließlich) 5

# Schleifen durch Strings
for i in range(len(s)): # Variable i läuft durch Index
    print(s[i])

for c in s:             # Variable c läuft durch Zeichen
    print(c)

# Funktionen
def calc(x, y): # Kopf mit Name und Parametern x und y
    z = x + y
    z = 3*z + 1
    return z
a = calc(2,4) # Aufruf mit den Argumenten 2 und 4

# eingebaute Funktionen
k = ord('a')           # Unicode-Zahl
c = chr(65)            # Zeichen zur Unicode-Zahl
max(a), min(a), sum(a) # bei Listen oder Tuples von Zahlen
```

```

# Listen
a = []          # leere Liste
a = [1,2,3], a = ['Montag', 'Dienstag']
k = len(a)      # Länge der Liste
a1 = a * 5      # Liste vervielfachen
b = 5 in a      # Ist 5 in Liste?

# Liste ändern
a[1] = 5        # Zuweisung
a.append(5)      # hinten was dranhängen
x = a.pop()      # hinten was herausnehmen

# Liste sortieren
b = sorted(a), b = sorted(a, reverse=True)

# Indexing und Slicing wie bei Strings
# Scheifen durch Listen wie bei Strings

# String s in Wortliste aufteilen
a = s.split()

# Unpacking
x, y = [5,2]    # gleichzeitige Zuweisung mehrerer Variablen

# List-Comprehensions
a = [x*x for x in range(1,10)]
b = [2*x for x in a if x % 3 == 0]
a = [int(x) for x in eingabe.split()] # Eingabe mehrerer Zahlen

# Dictionaries
m = {}, m = dict() # leeres dict
m = {'Thorben':2, 'Soeren':3, 'Maike':2}
len(m)             # Anzahl key-value Paare

v = m['Soeren']    # v ist value an der Stelle Soeren
m['Soeren'] = 4     # hinzufügen bzw. ändern des values
x = m.pop('Soeren') # Eintrag löschen, x ist gelöschter value
'Lena' in m        # ist Lena key in m

list(m.keys()), list(m.values()) # Liste aller keys / values

# Schleife durch ein dict:
for k in m:         # alle keys durchlaufen
    print(k, m[k])

# Tuple
t = (3,1,6)
t = (), t = (1,)    # leeres Tuple, Tuple mit einem Element
len(t)             # Länge des Tuples

# Indexing, Slicing, Schleifen, Unpacking wie bei Listen

a = list(tup)       # Tuple in Liste umwandeln
tup = tuple(a)      # Liste in Tuple umwandeln

m = {(1,2): 10, (3,4): 20} # Tuples können keys von dicts sein

# Zufall
import random
random.randint(0,20) # int Zufallszahl ∈ [0,20]
random.random()      # zufällige float ∈ [0,1]

# Dateien lesen (open/close nur im ersten Beispiel)
f = open('input.txt', encoding='utf-8')
zeile = f.readline().strip() # Eine Zeile lesen
f.close()

n = int(f.readline()) # erste Zeile ist Anzahl Folgezeilen
data = []
for i in range(n):
    data.append(int(f.readline())) # ganze Zahlen in Folgezeilen

for i in range(n):
    k, name = f.readline().split() # Folgezeile: 10 Lena
    data.append((int(k),name))

data = [int(x) for x in f.readline().split()] # mehrere Zahlen in Zeile

# Dateien schreiben
f = open('output.txt', encoding='utf-8', mode='w')
print(f'Die Summe von {x} und {y} ist {x+y}.', file=f)
f.close()

```