

Programmieren mit Python

Übungen

Fehlt eine Fragestellung, so ist die Ausgabe bzw. sind die Ausgaben des Python-Programms anzugeben.

Aufgabe 1.1

Wie heisst der Informatiker, der die Programmiersprache Python entwickelt hat?

Aufgabe 1.1

Wie heisst der Informatiker, der die Programmiersprache Python entwickelt hat?

Guido van Rossum

Aufgabe 1.2

Wonach wurde der Name der Programmiersprache „Python“ gewählt?

Aufgabe 1.2

Wonach wurde der Name der Programmiersprache „Python“ gewählt?

Nach der Komikertruppe „Monty Python“

Aufgabe 1.3

Was für eine Art von Programm braucht es, um ein Python-Programm zu schreiben?

Aufgabe 1.3

Was für eine Art von Programm braucht es, um ein Python-Programm zu schreiben?

Man benötigt einen Texteditor. Das ist ein Programm mit dem man Texte schreiben und bearbeiten kann.

Aufgabe 1.4

Was für eine Art von Programm braucht es, um ein Python-Programm auszuführen?

Aufgabe 1.4

Was für eine Art von Programm braucht es, um ein Python-Programm auszuführen?

einen Python-Interpreter (also ein Programm, das wiederum ein Programm ausführt)

Aufgabe 1.5

Zähle die drei grundsätzlichen Typen von Programmierfehlern auf und nenne jeweils ein Beispiel dazu.

Aufgabe 1.5

Zähle die drei grundsätzlichen Typen von Programmierfehlern auf und nenne jeweils ein Beispiel dazu.

- ▶ Syntax-Fehler
 - ▶ eine Variable falsch geschrieben,
 - ▶ Code nicht 4 Zeichen eingerückt,
 - ▶ Doppelpunkt vergessen,
 - ▶ ...
- ▶ Laufzeit-Fehler
 - ▶ Division durch Null,
 - ▶ eine Datei, die eingelesen werden soll, ist nicht vorhanden,
 - ▶ ...
- ▶ Logische Fehler
 - ▶ Fehler in einer Berechnung ,
 - ▶ Bedingungen nicht korrekt formuliert,
 - ▶ falsche Reihenfolge von Code,
 - ▶ ...

Aufgabe 1.6

Welche Dateiendung haben Python-Programme normalerweise?

Aufgabe 1.6

Welche Dateiendung haben Python-Programme normalerweise?

Die Endung `.py`

Aufgabe 2.1

```
print(1 + (2 * 3) - 4)
```

Aufgabe 2.1

```
print(1 + (2 * 3) - 4)
```

3

Aufgabe 2.2

```
print((1 + 2) * (3 - 4))
```

Aufgabe 2.2

```
print((1 + 2) * (3 - 4))
```

-3

Aufgabe 2.3

```
print(1 + 2 * 3 - 4)
```

Aufgabe 2.3

```
print(1 + 2 * 3 - 4)
```

3

Aufgabe 2.4

```
print(17 // 3)
```

Aufgabe 2.4

```
print(17 // 3)
```

5

Aufgabe 2.5

```
print(17 % 3)
```

Aufgabe 2.5

```
print(17 % 3)
```

2

Aufgabe 2.6

```
print(3 // 17)
```

Aufgabe 2.6

```
print(3 // 17)
```

0

Aufgabe 2.7

```
print(3 % 17)
```

Aufgabe 2.7

```
print(3 % 17)
```

3

Aufgabe 2.8

```
print(2**3)
```

Aufgabe 2.8

```
print(2**3)
```

8

Aufgabe 2.9

```
print(4**0.5)
```

Aufgabe 2.9

```
print(4**0.5)
```

2.0

Aufgabe 2.10

```
print(9**0)
```

Aufgabe 2.10

```
print(9**0)
```

1

Aufgabe 2.11

```
print(0**0)
```

Aufgabe 2.11

```
print(0**0)
```

1

Aufgabe 2.12

```
print(-3**2)
```

Aufgabe 2.12

```
print(-3**2)
```

-9

Aufgabe 2.13

```
print((-3)**2)
```

Aufgabe 2.13

```
print((-3)**2)
```

9

Aufgabe 2.14

```
print(10**(-2))
```

Aufgabe 2.14

```
print(10**(-2))
```

0.01

Aufgabe 2.15

```
print(-2**-2)
```

Aufgabe 2.15

```
print(-2**-2)
```

-0.25

Aufgabe 2.16

```
print((2**(3**2)))
```

Aufgabe 2.16

```
print((2**(3**2)))
```

512

Aufgabe 2.17

```
print((2**3)**2)
```

Aufgabe 2.17

```
print((2**3)**2)
```

64

Aufgabe 2.18

```
print(2**3**2)
```

Aufgabe 2.18

```
print(2**3**2)
```

512

Aufgabe 2.19

```
print((2**3)**2-1)
```

Aufgabe 2.19

```
print((2**3)**2-1)
```

63

Aufgabe 2.20

Welcher Datentyp hat das Resultat des Ausdrucks in Python?
(ohne Berechnung)

```
print(type(4 * 9))
```

```
print(type(5.0 + 3))
```

```
print(type(8 / 2))
```

```
print(type(8 // 3))
```

```
print(type(7.0 // 4))
```

```
print(type(9**0.5))
```

Aufgabe 2.20

```
print(type(4 * 9))  
print(type(5.0 + 3))  
print(type(8 / 2))  
print(type(8 // 3))  
print(type(7.0 // 4))  
print(type(9**0.5))
```

```
<class 'int'>  
<class 'float'>  
<class 'float'>  
<class 'int'>  
<class 'float'>  
<class 'float'>
```

Aufgabe 3.1

```
a = 3
b = -a + 7
c = 2*a + b
d = c - 3*a
print(d)
```

Aufgabe 3.1

```
a = 3  
b = -a + 7  
c = 2*a + b  
d = c - 3*a  
print(d)
```

1

Aufgabe 3.2

```
a = 17
b = a % 5
c = b * 4
d = c // 3
print(d)
```

Aufgabe 3.2

```
a = 17
b = a % 5
c = b * 4
d = c // 3
print(d)
```

2

Aufgabe 3.3

```
a = -1
b = 7
a = b
b = a
print(a)
print(b)
```

Aufgabe 3.3

```
a = -1
b = 7
a = b
b = a
print(a)
print(b)
```

7

7

Aufgabe 3.4

```
a = 4
b = 9
c = a
a = b
b = c
print(a)
print(b)
```

Aufgabe 3.4

```
a = 4
b = 9
c = a
a = b
b = c
print(a)
print(b)
```

9

4

Aufgabe 3.5

```
a = 7
a += 3
a *= 9
print(a)
```

Aufgabe 3.5

```
a = 7
a += 3
a *= 9
print(a)
```

90

Aufgabe 3.6

```
a = 16
b = 3
a -= b
a //= 2
print(a)
```

Aufgabe 3.6

```
a = 16
b = 3
a -= b
a //= 2
print(a)
```

6

Aufgabe 3.7

```
a = 7
b = 2
a, b = b, a
print(a)
print(b)
```

Aufgabe 3.7

```
a = 7
b = 2
a, b = b, a
print(a)
print(b)
```

2

7

Aufgabe 4.1

```
print(5 != 5)
```

Aufgabe 4.1

```
print(5 != 5)
```

False

Aufgabe 4.2

```
print(True and False)
```

Aufgabe 4.2

```
print(True and False)
```

```
False
```


Aufgabe 4.3

```
print((5 < 7) or (3 > 8))
```

Aufgabe 4.3

```
print((5 < 7) or (3 > 8))
```

True

Aufgabe 4.4

```
print(not (2 < 1) and (7 > 6))
```

Aufgabe 4.4

```
print(not (2 < 1) and (7 > 6))
```

True

Aufgabe 4.5

```
print((3 < 2) or (7 > 7) or (5 != 8))
```

Aufgabe 4.5

```
print((3 < 2) or (7 > 7) or (5 != 8))
```

True

Aufgabe 4.6

```
print(not (not (3 < 2) and (7 > -1)))
```

Aufgabe 4.6

```
print(not (not (3 < 2) and (7 > -1)))
```

False

Aufgabe 4.7

```
print(True and True and True and True and False)
```

Aufgabe 4.7

```
print(True and True and True and True and False)
```

False

Aufgabe 4.8

```
print(False or False or False or False or False)
```

Aufgabe 4.8

```
print(False or False or False or False or False)
```

False

Aufgabe 4.9

```
x = 3.1
print(2.8 < x < 7.6)
```

Aufgabe 4.9

```
x = 3.1
print(2.8 < x < 7.6)
```

True

Aufgabe 4.10

```
x = 8  
print(2 < x < 7)
```

Aufgabe 4.10

```
x = 8
```

```
print(2 < x < 7)
```

False

Aufgabe 4.11

```
print(True and False or False or True and False)
```

Aufgabe 4.11

```
print(True and False or False or True and False)
```

```
False
```

Aufgabe 5.1

```
a = 3
b = 8
if a > b:
    a = b
print(a)
```

Aufgabe 5.1

```
a = 3
b = 8
if a > b:
    a = b
print(a)
```

3

Aufgabe 5.2

```
a = 3
b = 8
if b % 2 == 0:
    a = b
print(a)
```

Aufgabe 5.2

```
a = 3
b = 8
if b % 2 == 0:
    a = b
print(a)
```

8

Aufgabe 5.3

```
a = 3
b = 8
c = 4
if c < a:
    a = b
else:
    a = c
print(a)
```

Aufgabe 5.3

```
a = 3
b = 8
c = 4
if c < a:
    a = b
else:
    a = c
print(a)
```

4

Aufgabe 5.4

```
a = 3
b = 8
c = 4
if a + c > b:
    b = c - a
else:
    b = b + 2
print(b)
```

Aufgabe 5.4

```
a = 3
b = 8
c = 4
if a + c > b:
    b = c - a
else:
    b = b + 2
print(b)
```

10

Aufgabe 5.5

```
a = 3
b = 8
c = 4
if a <= 2:
    if b > 9:
        c = 8
else:
    if c > 7:
        c = c + 3
    else:
        c = 3 * c
print(c)
```

Aufgabe 5.5

```
a = 3
b = 8
c = 4
if a <= 2:
    if b > 9:
        c = 8
else:
    if c > 7:
        c = c + 3
    else:
        c = 3 * c
print(c)
```

12

Aufgabe 5.6

```
a = 8
if a > 5:
    b = 1
elif a > 7:
    b = 2
elif a > 9:
    b = 3
else:
    b = 4
print(b)
```

Aufgabe 5.6

```
a = 8
if a > 5:
    b = 1
elif a > 7:
    b = 2
elif a > 9:
    b = 3
else:
    b = 4
print(b)
```

1

Aufgabe 5.7

```
a = 8
b = 4
if a < 5:
    b = 1
elif a < 6:
    b = 2
elif a < 7:
    b = 3
print(b)
```

Aufgabe 5.7

```

a = 8
b = 4
if a < 5:
    b = 1
elif a < 6:
    b = 2
elif a < 7:
    b = 3
print(b)

```

4

Aufgabe 6.1

```
for k in range(1, 6):  
    print(k)
```

Aufgabe 6.1

```
for k in range(1, 6):  
    print(k)
```

1

2

3

4

5

Aufgabe 6.2

```
for x in [1, 4, 9, 3]:
    print(x)
```

Aufgabe 6.2

```
for x in [1, 4, 9, 3]:
    print(x)
```

1

4

9

3

Aufgabe 6.3

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

Aufgabe 6.3

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

0

1

2

3

4

Aufgabe 6.4

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

Aufgabe 6.4

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

1
2
3
4
5

Aufgabe 6.5

```
for j in range(0, 5):  
    print(4-j)
```

Aufgabe 6.5

```
for j in range(0, 5):
    print(4-j)
```

4

3

2

1

0

Aufgabe 6.6

```
for i in range(0, 5):
    print(3*i)
```

Aufgabe 6.6

```
for i in range(0, 5):
    print(3*i)
```

0

3

6

9

12

Aufgabe 6.7

```
for x in [8, 1, 7, 4, 6, 9]:  
    if x % 2 == 0:  
        print(x)
```

Aufgabe 6.7

```
for x in [8, 1, 7, 4, 6, 9]:
    if x % 2 == 0:
        print(x)
```

8

4

6

Aufgabe 6.8

```
s = 0
for x in [8, 5, 6, 2, 9]:
    s = s + x
print(s)
```

Aufgabe 6.8

```
s = 0
for x in [8, 5, 6, 2, 9]:
    s = s + x
print(s)
```

30

Aufgabe 6.9

```
s = 0
for x in [6, 1, 8, 7, 5]:
    s = s + x
print(s)
```

Aufgabe 6.9

```
s = 0
for x in [6, 1, 8, 7, 5]:
    s = s + x
    print(s)
```

6

7

15

22

27

Aufgabe 6.10

```
A = [7, 4, 8]
for k in range(0, len(A)):
    print(k, A[k])
```

Aufgabe 6.10

```
A = [7, 4, 8]
for k in range(0, len(A)):
    print(k, A[k])
```

0 7

1 4

2 8

Aufgabe 6.11

```
A = [1, 4, 7]
b = 0
for j in range(0, len(A)):
    b = b+A[j]**2
print(b)
```

Aufgabe 6.11

```
A = [1, 4, 7]
b = 0
for j in range(0, len(A)):
    b = b+A[j]**2
print(b)
```

66

Aufgabe 6.12

```
A = [8, 4, 3]
B = [4, 9, 7]
c = 0
for j in range(0, len(A)):
    c = c+A[j]*B[j]
print(c)
```

Aufgabe 6.12

```
A = [8, 4, 3]
B = [4, 9, 7]
c = 0
for j in range(0, len(A)):
    c = c+A[j]*B[j]
print(c)
```

89

Aufgabe 6.13

```
A = [5, 1, 4, 9]
for k in range(1, len(A)):
    print(A[k-1]*A[k])
```

Aufgabe 6.13

```
A = [5, 1, 4, 9]
for k in range(1, len(A)):
    print(A[k-1]*A[k])
```

5

4

36

Aufgabe 6.14

```
C = [3, 5, 1, 6, 9, 4, 8]
x = C[0]
for j in range(0, len(C)):
    if C[j] > x:
        x = C[j]
print(x)
```

Aufgabe 6.14

```
C = [3, 5, 1, 6, 9, 4, 8]
x = C[0]
for j in range(0, len(C)):
    if C[j] > x:
        x = C[j]
print(x)
```

9

Aufgabe 6.15

```
for a in range(2, 5):
    for b in range(4, 7):
        print(a,b)
```

Aufgabe 6.15

```
for a in range(2, 5):
    for b in range(4, 7):
        print(a,b)
```

2 4

2 5

2 6

3 4

3 5

3 6

4 4

4 5

4 6

Aufgabe 6.16

```
C = []
for x in [4, 7]:
    for y in [7, 3, 8]:
        C.append(x*y)
print(C)
```

Aufgabe 6.16

```
C = []
for x in [4, 7]:
    for y in [7, 3, 8]:
        C.append(x*y)
print(C)
```

[28, 12, 32, 49, 21, 56]

Aufgabe 6.17

```
for z in [-4, 2, 4, 1, 3, -1]:  
    if z > 0:  
        print(z)  
    else:  
        break
```

Aufgabe 6.17

```
for z in [-4, 2, 4, 1, 3, -1]:  
    if z > 0:  
        print(z)  
    else:  
        break
```

Aufgabe 6.18

```
for z in [0, 2, -5, -2, 3, -3]:
    if z > 0:
        print(z)
    else:
        continue
```

Aufgabe 6.18

```
for z in [0, 2, -5, -2, 3, -3]:
    if z > 0:
        print(z)
    else:
        continue
```

2

3

Aufgabe 6.19

```
a = 25
b = 15
while True:
    if b == 0:
        print(a)
        break
    (a, b) = (b, a % b)
```

Aufgabe 6.19

```
a = 25
b = 15
while True:
    if b == 0:
        print(a)
        break
    (a, b) = (b, a % b)
```

5

Aufgabe 6.20

```
A = [1, 1]
for k in range(2, 7):
    A.append(A[k-1]+A[k-2])
print(A)
```

Aufgabe 6.20

```
A = [1, 1]
for k in range(2, 7):
    A.append(A[k-1]+A[k-2])
print(A)
```

[1, 1, 2, 3, 5, 8, 13]

Aufgabe 7.1

```
a = [-5, -7, -1, -6, -2]
print(a[3])
```

Aufgabe 7.1

```
a = [-5, -7, -1, -6, -2]
print(a[3])
```

-6

Aufgabe 7.2

```
a = [6, 3, -3, -6, -2]
print(a[5])
```

Aufgabe 7.2

```
a = [6, 3, -3, -6, -2]  
print(a[5])
```

Traceback (most recent call last):

```
File "python-07-ueb-02.py", line 2, in <module>  
    print(a[5])
```

IndexError: list index out of range

Aufgabe 7.3

```
a = [-9, 3, 4, 9, 2]
print(a[-4])
```

Aufgabe 7.3

```
a = [-9, 3, 4, 9, 2]
print(a[-4])
```

3

Aufgabe 7.4

```
a = [-1, -2, -6, 1, -9]
a[0] = 10
print(a[0])
```

Aufgabe 7.4

```
a = [-1, -2, -6, 1, -9]
```

```
a[0] = 10
```

```
print(a[0])
```

10

Aufgabe 7.5

```
a = [[8, -9, 3], [-3, 6, 2], [9, -1, -6]]
print(a[1][1])
```

Aufgabe 7.5

```
a = [[8, -9, 3], [-3, 6, 2], [9, -1, -6]]
print(a[1][1])
```

6

Aufgabe 7.6

```
a = [8, -2, 2, 9, 1]
print(a[1:2])
```

Aufgabe 7.6

```
a = [8, -2, 2, 9, 1]
```

```
print(a[1:2])
```

```
[-2]
```

Aufgabe 7.7

```
a = [0, -9, 4, -6, -7]
print(a[:2])
```

Aufgabe 7.7

```
a = [0, -9, 4, -6, -7]
print(a[:2])
```

[0, -9]

Aufgabe 7.8

```
a = [-6, 1, 7, 4, -1]
print(a[1:])
```

Aufgabe 7.8

```
a = [-6, 1, 7, 4, -1]
print(a[1:])
```

```
[1, 7, 4, -1]
```


Aufgabe 7.9

```
a = [0, 9, 6, 3, 8]
print(a[:])
```

Aufgabe 7.9

```
a = [0, 9, 6, 3, 8]
print(a[:])
```

[0, 9, 6, 3, 8]

Aufgabe 7.10

```
a = [-8, 6, -2, 3, -3]
print(a[:-1])
```

Aufgabe 7.10

```
a = [-8, 6, -2, 3, -3]
```

```
print(a[:-1])
```

```
[-8, 6, -2, 3]
```

Aufgabe 7.11

```
a = [2, 8, 9, -7, -2]
print(len(a))
```

Aufgabe 7.11

```
a = [2, 8, 9, -7, -2]
print(len(a))
```

5

Aufgabe 7.12

```
a = []  
print(len(a))
```

Aufgabe 7.12

```
a = []  
print(len(a))
```

0

Aufgabe 7.13

```
a = [7, -6]
b = [-4, 7, 3]
print(a + b)
```

Aufgabe 7.13

```
a = [7, -6]
b = [-4, 7, 3]
print(a + b)
```

[7, -6, -4, 7, 3]

Aufgabe 7.14

```
a = [2]
print(4 * [2])
```

Aufgabe 7.14

```
a = [2]
print(4 * [2])
```

[2, 2, 2, 2]

Aufgabe 7.15

```
a = [-9, 6, -8, 0, 2]
a.append(19)
print(a)
```

Aufgabe 7.15

```
a = [-9, 6, -8, 0, 2]
```

```
a.append(19)
```

```
print(a)
```

```
[-9, 6, -8, 0, 2, 19]
```

Aufgabe 7.16

```
a = [9, 3, -4, 7, -1]
a.pop()
print(a)
```

Aufgabe 7.16

```
a = [9, 3, -4, 7, -1]
a.pop()
print(a)
```

```
[9, 3, -4, 7]
```


Aufgabe 7.17

```
a = [-2, -9, -3, 9, 7]
a.pop(3)
print(a)
```

Aufgabe 7.17

```
a = [-2, -9, -3, 9, 7]
```

```
a.pop(3)
```

```
print(a)
```

```
[-2, -9, -3, 7]
```

Aufgabe 7.18

```
a = [2, 5, -1, -9, -3]
a.reverse()
print(a)
```

Aufgabe 7.18

```
a = [2, 5, -1, -9, -3]
a.reverse()
print(a)
```

```
[-3, -9, -1, 5, 2]
```

Aufgabe 7.19

```
a = [7, -5, 9, -3, 6]
b = sorted(a)
print(b)
```

Aufgabe 7.19

```
a = [7, -5, 9, -3, 6]
b = sorted(a)
print(b)
```

```
[-5, -3, 6, 7, 9]
```

Aufgabe 7.20

```
a = [-8, 6, 2, -3, -1]
for i in range(0, len(a)):
    print(a[i])
```

Aufgabe 7.20

```
a = [-8, 6, 2, -3, -1]
for i in range(0, len(a)):
    print(a[i])
```

-8

6

2

-3

-1

Aufgabe 7.21

```
a = [4, 6, 2, 9, -4]
n = len(a)
for i in range(0, n):
    print(a[n-i-1])
```

Aufgabe 7.21

```
a = [4, 6, 2, 9, -4]
n = len(a)
for i in range(0, n):
    print(a[n-i-1])
```

-4

9

2

6

4

Aufgabe 7.22

```
a = [-9, 8, 1, 5, 0]
for i in range(0, len(a)):
    print(a[i]+3)
```

Aufgabe 7.22

```
a = [-9, 8, 1, 5, 0]
for i in range(0, len(a)):
    print(a[i]+3)
```

-6

11

4

8

3

Aufgabe 7.23

```
a = [-9, -2, -4, 7, 4]
for i in range(0, len(a)):
    print(a[i]**2)
```

Aufgabe 7.23

```
a = [-9, -2, -4, 7, 4]
for i in range(0, len(a)):
    print(a[i]**2)
```

81

4

16

49

16

Aufgabe 7.24

```
a = []
for i in range(0, 5):
    a.append(i % 2)
print(a)
```

Aufgabe 7.24

```
a = []
for i in range(0, 5):
    a.append(i % 2)
print(a)
```

[0, 1, 0, 1, 0]

Aufgabe 7.25

```
a = [-7, 8, -1, -8, 4]
b = []
for i in range(0, len(a)):
    b.append(a.pop())
print(b)
```

Aufgabe 7.25

```
a = [-7, 8, -1, -8, 4]
b = []
for i in range(0, len(a)):
    b.append(a.pop())
print(b)
```

```
[4, -8, -1, 8, -7]
```

Aufgabe 7.26

```
a = [-7, 9, -9, 5, 7]
for e in a:
    if e > 1:
        print(e)
```

Aufgabe 7.26

```
a = [-7, 9, -9, 5, 7]
for e in a:
    if e > 1:
        print(e)
```

9

5

7

Aufgabe 7.27

```
x = [-1, -2, 9, -6, 3]
```

```
y = x
```

```
y[4] = 12
```

```
print(x)
```

Aufgabe 7.27

```
x = [-1, -2, 9, -6, 3]
```

```
y = x
```

```
y[4] = 12
```

```
print(x)
```

```
[-1, -2, 9, -6, 12]
```

Aufgabe 8.1

```
def f(x):  
    return 2*x + 1  
  
print(f(5))
```

Aufgabe 8.1

11

Aufgabe 8.2

```
def f(x):  
    return x*(x-1)  
  
print(f(3))
```

Aufgabe 8.2

6

Aufgabe 8.3

```
def f(x):
    y = 6
    return x*y

print(f(10))
```

Aufgabe 8.3

60

Aufgabe 8.4

```
def g(a):
    return a % 2

print(g(9))
```

Aufgabe 8.4

1

Aufgabe 8.5

```
def f(x):
    return 7

print(f(23))
```

Aufgabe 8.5

7

Aufgabe 8.6

```
def f(x):
    return 7

print(f(-5))
```

Aufgabe 8.6

7

Aufgabe 8.7

```
def f():
    return 2*4 + 3

print(f())
```

Aufgabe 8.7

11

Aufgabe 8.8

```
def f(x):
    print(x+1)

f(2.7)
```

Aufgabe 8.8

3.7

Aufgabe 8.9

```
# gemein!
def change(x):
    y = x

y = 3
change(5)
print(y)
```

Aufgabe 8.9

3

Aufgabe 8.10

```
# noch gemeiner!
def change(L):
    L.append(4)

L = [1, 2, 3]
change(L)
print(L)
```

Aufgabe 8.10

[1, 2, 3, 4]

Aufgabe 8.11

```
def u(x):  
    y = 3*x + 2  
    return y  
  
print(u(4)+1)
```

Aufgabe 8.11

15

Aufgabe 8.12

```
def u(x):  
    y = 3*x + 2  
    return y  
  
print(u(4+1))
```

Aufgabe 8.12

17

Aufgabe 8.13

```
def fun(x=42):
    return x + 5

print(fun(7))
```

Aufgabe 8.13

12

Aufgabe 8.14

```
def fun(x=42):
    return x + 5

print(fun())
```

Aufgabe 8.14

47

Aufgabe 8.15

```
def g(x):
    4*x + 1

print(g(5))
```

Aufgabe 8.15

None

Aufgabe 8.16

```
def f(x):  
    x = x + 1  
    return x  
    x = x + 1  
  
print(f(20))
```

Aufgabe 8.16

21

Aufgabe 8.17

```
def f(x, y):  
    return x + y + 1  
  
print(f(3, 4))
```

Aufgabe 8.17

8

Aufgabe 8.18

```
def f(x, y):
    return 3*x - 2*y

print(f(10, 1))
```

Aufgabe 8.18

28

Aufgabe 8.19

```
def f(x, y, z):
    return y + 2*x + 3*z

print(f(1, 2, 3))
```

Aufgabe 8.19

13

Aufgabe 8.20

```
def rechteck(a=1, b=1):
    return a * b

print(rechteck(3, 8))
```

Aufgabe 8.20

24

Aufgabe 8.21

```
def rechteck(a=1, b=1):
    return a * b

print(rechteck(4))
```

Aufgabe 8.21

4

Aufgabe 8.22

```
def rechteck(a=1, b=1):
    return a * b

print(rechteck())
```

Aufgabe 8.22

1

Aufgabe 8.23

```
def F(u, v, w):  
    return u + w  
  
print(F(1, 2, 3))
```

Aufgabe 8.23

4

Aufgabe 8.24

```
def first(L):
    return L[0]

print(first([6, -5, 12, 3]))
```

Aufgabe 8.24

6

Aufgabe 8.25

```
def summe(L):  
    s = 0  
    for x in L:  
        s += x  
    return s  
  
print(summe([3, 5, 2, 0, 8]))
```

Aufgabe 8.25

18

Aufgabe 8.26

```
def f(u, v):  
    return u[0]*v[0] + u[1]*v[1] + u[2]*v[2]  
  
print(f([3, 0, 1], [4, -7, 5]))
```

Aufgabe 8.26

17

Aufgabe 8.27

```
def h(x):  
    return 2*x - 1  
  
print(h(h(2)))
```

Aufgabe 8.27

5

Aufgabe 8.28

```
def c(x):
    return x*(x-1)

print(c(c(c(c(2)))))
```

Aufgabe 8.28

2

Aufgabe 8.29

```
def f(x, y):  
    return 2*x + y  
  
def g(x, y):  
    return x + y + 3  
  
print(f(g(1, 2), f(3, 0)))
```

Aufgabe 8.29

18

Aufgabe 8.30

```
def factorial(n):  
    if n == 1:  
        return 1  
    else:  
        return n*factorial(n-1)  
  
print(factorial(4))
```

Aufgabe 8.30

24

Aufgabe 8.31

```
def fibo(n):  
    if n <= 1:  
        return 1  
    else:  
        return fibo(n-1) + fibo(n-2)  
  
print(fibo(4))
```

Aufgabe 8.31

5

Aufgabe 9.1

```
wort = 'HALLO'  
print(wort[1])
```

Aufgabe 9.1

```
wort = 'HALLO'  
print(wort[1])
```

A

Aufgabe 9.2

```
wort = "HALLO"  
print(wort[-1])
```

Aufgabe 9.2

```
wort = "HALLO"
print(wort[-1])
```

0

Aufgabe 9.3

```
wort = 'hundert'  
print(wort[1:4])
```

Aufgabe 9.3

```
wort = 'hundert'
print(wort[1:4])
```

und

Aufgabe 9.4

```
satz = "Was soll das?"  
print(len(satz))
```

Aufgabe 9.4

```
satz = "Was soll das?"  
print(len(satz))
```

13

Aufgabe 9.5

```
wort = 'Feeler'
wort[2] = 'h'
print(wort)
```

Aufgabe 9.5

```
wort = 'Feeler'  
wort[2] = 'h'  
print(wort)
```

Traceback (most recent call last):

File "python-09-ueb-05.py", line 2, in <module>

wort[2] = 'h'

TypeError: 'str' object does not support item assignment

Aufgabe 9.6

```
a = 'abc'
b = 'xyz'
print(a + b)
```

Aufgabe 9.6

```
a = 'abc'
b = 'xyz'
print(a + b)
```

abcxyz

Aufgabe 9.7

```
print('a' + 2 * 'n' + 'a')
```

Aufgabe 9.7

```
print('a' + 2 * 'n' + 'a')
```

anna

Aufgabe 9.8

```

satz = 'Das\nist\nso.'
print(satz)

```

Aufgabe 9.8

```
satz = 'Das\nist\nso.'  
print(satz)
```

```
Das  
ist  
so.
```

Aufgabe 9.9

```
satz = '''Woher
weisst du das?'''
print(satz)
```

Aufgabe 9.9

```
satz = '''Woher
weisst du das?'''
print(satz)
```

```
Woher
weisst du das?
```

Aufgabe 9.10

```

satz = '''Das \
ist vielleicht \
sinnlos!'''
print(satz)

```

Aufgabe 9.10

```
satz = '''Das \
ist vielleicht \
sinnlos!'''
print(satz)
```

Das ist vielleicht sinnlos!

Aufgabe 9.11

```
text = '15'  
print(int(text))
```

Aufgabe 9.11

```
text = '15'
print(int(text))
```

15

Aufgabe 9.12

```
text = '15'  
print(int(text, 7))
```

Aufgabe 9.12

```
text = '15'
print(int(text, 7))
```

12

Aufgabe 9.13

```
text = '1.2345e3'
print(float(text))
```

Aufgabe 9.13

```
text = '1.2345e3'
print(float(text))
```

1234.5

Aufgabe 9.14

```
satz = "Das ist das Zeichen \\.\"
print(satz)
```

Aufgabe 9.14

```
satz = "Das ist das Zeichen \\.  
print(satz)
```

Das ist das Zeichen \.

Aufgabe 9.15

```
text = "Sag \"Hallo\""
print(text)
```

Aufgabe 9.15

```
text = "Sag \"Hallo\""
print(text)
```

Sag "Hallo"

Aufgabe 9.16

```
print('\N{QUESTION MARK}')
```

Aufgabe 9.16

```
print('\N{QUESTION MARK}')
```

?

Aufgabe 9.17

```
print(ord("A"))
```

Aufgabe 9.17

```
print(ord("A"))
```

65

Aufgabe 9.18

```
print(chr(65))
```

Aufgabe 9.18

```
print(chr(65))
```

A

Aufgabe 9.19

```
a = 'Das'
b = 'ist'
c = 'schlecht'
print(a, b, c, sep='.')
```

Aufgabe 9.19

```
a = 'Das'
b = 'ist'
c = 'schlecht'
print(a, b, c, sep='.')
```

Das.ist.schlecht

Aufgabe 9.20

```
wort = 'Ragusa'  
print('usa' in wort)
```

Aufgabe 9.20

```
wort = 'Ragusa'
print('usa' in wort)
```

True

Aufgabe 9.21

```
text = '{0} + {1} = {2}'.format(3, 4, 7)
print(text)
```

Aufgabe 9.21

```
text = '{0} + {1} = {2}'.format(3, 4, 7)
print(text)
```

3 + 4 = 7

Aufgabe 9.22

```
text = '{1} Meter kosten {0} Fr.'.format(20, 30)
print(text)
```

Aufgabe 9.22

```
text = '{1} Meter kosten {0} Fr.'.format(20, 30)
print(text)
```

30 Meter kosten 20 Fr.

Aufgabe 9.23

```
text = "{0:#b}".format(7)
print(text)
```

Aufgabe 9.23

```
text = "{0:#b}".format(7)
print(text)
```

0b111

Aufgabe 9.24

```
text = 'gut gemacht!'
print(text.capitalize())
```

Aufgabe 9.24

```
text = 'gut gemacht!'
print(text.capitalize())
```

Gut gemacht!

Aufgabe 9.25

```
wort = 'abracadabra'  
print(wort.count('ab'))
```

Aufgabe 9.25

```
wort = 'abracadabra'  
print(wort.count('ab'))
```

2

Aufgabe 9.26

```
wort = 'Mississippi'  
print(wort.find('is'))
```

Aufgabe 9.26

```
wort = 'Mississippi'
print(wort.find('is'))
```

1

Aufgabe 9.27

```
liste = ['25', '2', '2013']
print('.'.join(liste))
```

Aufgabe 9.27

```
liste = ['25', '2', '2013']
print('.'.join(liste))
```

25.2.2013

Aufgabe 9.28

```
wort = "HAMMER"  
wort.lower()  
print(wort)
```

Aufgabe 9.28

```
wort = "HAMMER"  
wort.lower()  
print(wort)
```

HAMMER

Aufgabe 9.29

```
wort = "Hammer"  
wort = wort.replace('m', 'c', 1)  
wort = wort.replace('m', 'k', 1)  
print(wort)
```

Aufgabe 9.29

```
wort = "Hammer"  
wort = wort.replace('m', 'c', 1)  
wort = wort.replace('m', 'k', 1)  
print(wort)
```

Hacker

Aufgabe 9.30

```
satz = 'Das ist gut.'
satz = satz.strip('.')
print(satz)
```

Aufgabe 9.30

```
satz = 'Das ist gut.'
satz = satz.strip('.')
print(satz)
```

Das ist gut

Aufgabe 9.31

```
satz = 'Das ist gut'
abc = satz.split(' ')
print(abc)
```

Aufgabe 9.31

```
satz = 'Das ist gut'
abc = satz.split(' ')
print(abc)
```

```
['Das', 'ist', 'gut']
```

Aufgabe 9.32

```
wort = "ANANAS"
wert = wort.split('N')
print(wert)
```

Aufgabe 9.32

```
wort = "ANANAS"
wert = wort.split('N')
print(wert)
```

```
['A', 'A', 'AS']
```

Aufgabe 9.33

```
text = '7'  
text = text.zfill(3)  
print(text)
```

Aufgabe 9.33

```
text = '7'
text = text.zfill(3)
print(text)
```

007

Aufgabe 9.34

```
a = 'mit'
b = list(a)
b.reverse()
c = ''.join(b)
print(c)
```

Aufgabe 9.34

```
a = 'mit'
b = list(a)
b.reverse()
c = ''.join(b)
print(c)
```

tim

Aufgabe 10.1

```
print('abc', 'def', sep='*', end='\n...')
print('xyz', '123', sep='*')
```

Aufgabe 10.1

```
print('abc', 'def', sep='*', end='\n...')
print('xyz', '123', sep='*')
```

```
abc*def
...xyz*123
```

Aufgabe 10.2

```
print('{0:>3}{1:>3}'.format(123, 45))
print('{0:<3}{1:>3}'.format(67, 8))
```

Aufgabe 10.2

```
print('{0:>3}{1:>3}'.format(123, 45))
print('{0:<3}{1:>3}'.format(67, 8))
```

123 45

67 8

Aufgabe 10.3

```
x = input('Eingabe: ') # ... es wird 7 eingegeben
print(2*x)
```

Aufgabe 10.3

```
x = input('Eingabe: ') # ... es wird 7 eingegeben
print(2*x)
```

77

Aufgabe 10.4

```
x = int(input('Eingabe: ')) # ... es wird 7 eingegeben
print(2*x)
```

Aufgabe 10.4

```
x = int(input('Eingabe: ')) # ... es wird 7 eingegeben
print(2*x)
```

14

Aufgabe 10.5

```
x = float(input('Eingabe: ')) # ... es wird 7 eingegeben
print(2*x)
```

Aufgabe 10.5

```
x = float(input('Eingabe: ')) # ... es wird 7 eingegeben
print(2*x)
```

14.0

Aufgabe 10.6

Was steht nach der Ausführung des folgenden Programms in der Datei `python-10-ueb-06.out`?

```
fd = open('python-10-ueb-06.out', mode='w')
for i in range(1,5):
    if i % 2 == 0:
        fd.write('{0}\n'.format(i))
    else:
        fd.write('{0}'.format(i))
fd.close()
```

Aufgabe 10.6

```
fd = open('python-10-ueb-06.out', mode='w')
for i in range(1,5):
    if i % 2 == 0:
        fd.write('{0}\n'.format(i))
    else:
        fd.write('{0}'.format(i))
fd.close()
```

In der Datei python-10-ueb-06.out steht:

12

34

Aufgabe 10.7

Die Datei `python-10-ueb-07.data` hat den Inhalt:

1.3

2.9

4.7

3.1

Welche Ausgabe macht das folgende Programm?

```
s = 0
n = 0
fd = open('python-10-ueb-07.data')
for zeile in fd:
    s += float(zeile)
    n += 1
fd.close()
print(s/n)
```

Aufgabe 10.7

```
s = 0
n = 0
fd = open('python-10-ueb-07.data')
for zeile in fd:
    s += float(zeile)
    n += 1
fd.close()
print(s/n)
```

3.0

Wenn in der Aufgabe keine Sortierfunktion verwendet wird, kann die Menge der Schlüssel-Wert-Paare in beliebiger Reihenfolge angegeben werden.

Aufgabe 11.1

```
D = {'a': 6, 'b': -1, 'e': -6}
print(D['e'])
```


Aufgabe 11.1

```
D = {'a': 6, 'b': -1, 'e': -6}
print(D['e'])
```

-6

Aufgabe 11.2

```
D = {3: 6, -3: -9, 6: -6}
print(D[1+2])
```

Aufgabe 11.2

```
D = {3: 6, -3: -9, 6: -6}
print(D[1+2])
```

6

Aufgabe 11.3

```
D = {'a':'d', 'e':'g', 'd':'e', 'g':'a'}
print(D[D[D['e']]])
```

Aufgabe 11.3

```
D = {'a':'d', 'e':'g', 'd':'e', 'g':'a'}
print(D[D[D['e']]])
```

d

Aufgabe 11.4

```
D = {'a': 3, 'e': 2, 'g': -3, 'f': 4}
D.pop('a')
print(len(D))
```

Aufgabe 11.4

```
D = {'a': 3, 'e': 2, 'g': -3, 'f': 4}
```

```
D.pop('a')
```

```
print(len(D))
```

3

Aufgabe 11.5

```
D = {'b': -2, 'e': 8, 'd': 2, 'g': -6}
del D['b']
print(len(D))
```


Aufgabe 11.5

```
D = {'b': -2, 'e': 8, 'd': 2, 'g': -6}
del D['b']
print(len(D))
```

3

Aufgabe 11.6

```
D = {'c': -6, 'd': 4, 'g': -9}
D.pop('b')
print(D)
```

Aufgabe 11.6

```
D = {'c': -6, 'd': 4, 'g': -9}
D.pop('b')
print(D)
```

Traceback (most recent call last):

...

D.pop('b')

KeyError: 'b'

Aufgabe 11.7

```
D = {'c': 4, 'g': 1, 'f': -6}
print(D.get('e', 0))
```

Aufgabe 11.7

```
D = {'c': 4, 'g': 1, 'f': -6}
print(D.get('e', 0))
```

0

Aufgabe 11.8

```
D = {'b': 6, 'g': 0, 'f': 8}
E = {'b': -2, 'g': 0, 'f': 8}
D.update(E)
print(D)
```

Aufgabe 11.8

```
D = {'b': 6, 'g': 0, 'f': 8}
E = {'b': -2, 'g': 0, 'f': 8}
D.update(E)
print(D)
```

```
{'b': -2, 'g': 0, 'f': 8}
```

Aufgabe 11.9

```
D = {'d': 2, 'g': 7, 'f': -4}
E = D.copy()
E['f'] = 99
print(D['f'])
```


Aufgabe 11.9

```
D = {'d': 2, 'g': 7, 'f': -4}
E = D.copy()
E['f'] = 99
print(D['f'])
```

-4

Aufgabe 11.10

```
D = {'e': 3, 'd': -6, 'f': -4}
for x in D:
    print(x)
```

Aufgabe 11.10

```
D = {'e': 3, 'd': -6, 'f': -4}
```

```
for x in D:
```

```
    print(x)
```

e

d

f

Aufgabe 11.11

```
D = {'a': 0, 'c': 6, 'd': -8}
for y in sorted(D.keys()):
    print(y)
```

Aufgabe 11.11

```
D = {'a': 0, 'c': 6, 'd': -8}
for y in sorted(D.keys()):
    print(y)
```

a

c

d

Aufgabe 11.12

```
D = {'e': -4, 'g': 7, 'f': 3}
for x in D.keys():
    print(D[x])
```

Aufgabe 11.12

```
D = {'e': -4, 'g': 7, 'f': 3}
for x in D.keys():
    print(D[x])
```

-4

7

3

Aufgabe 11.13

```
D = {'a': 3, 'b': -2, 'e': -3}
for y in sorted(D.values()):
    print(y)
```


Aufgabe 11.13

```
D = {'a': 3, 'b': -2, 'e': -3}
for y in sorted(D.values()):
    print(y)
```

-3

-2

3

Aufgabe 11.14

```
D = {3:2, 2:0, 4:5}
for v, w in D.items():
    print(v-w)
```

Aufgabe 11.14

```
D = {3:2, 2:0, 4:5}
for v, w in D.items():
    print(v-w)
```

1

2

-1

Aufgabe 11.15

```
D = {'e': 3, 'g': -5, 'f': -9}
for t in sorted(D.keys()):
    print(D[t])
```

Aufgabe 11.15

```
D = {'e': 3, 'g': -5, 'f': -9}
for t in sorted(D.keys()):
    print(D[t])
```

3

-9

-5

Aufgabe 12.1

```
A = {3, 9, -1, -7, -6, -2}
```

```
print(len(A))
```

Aufgabe 12.1

```
A = {3, 9, -1, -7, -6, -2}
```

```
print(len(A))
```

6

Aufgabe 12.2

```
A = set()
print(len(A))
```


Aufgabe 12.2

```
A = set()
print(len(A))
```

0

Aufgabe 12.3

```
A = {5, 6, 3, -3, -8, 6}
print(len(A))
```

Aufgabe 12.3

```
A = {5, 6, 3, -3, -8, 6}
print(len(A))
```

5

Aufgabe 12.4

```
a = [4, 1, 2, 4, 3, 1]
A = set(a)
print(A)
```

Aufgabe 12.4

```
a = [4, 1, 2, 4, 3, 1]
```

```
A = set(a)
```

```
print(A)
```

```
{1, 2, 3, 4}
```

Aufgabe 12.5

```
A = {-2, -6, 5, -1}
```

```
B = {5, -2, -9}
```

```
print(A | B)
```

Aufgabe 12.5

$A = \{-2, -6, 5, -1\}$

$B = \{5, -2, -9\}$

`print(A | B)`

$\{5, -1, -9, -6, -2\}$

Aufgabe 12.6

```
A = {-9, 8, 9, 3}
```

```
B = {9, -9, 3, 8}
```

```
print(B | A)
```


Aufgabe 12.6

A = {-9, 8, 9, 3}

B = {9, -9, 3, 8}

`print(B | A)`

{3, 8, 9, -9}

Aufgabe 12.7

```
A = {4, -3, -2, -9}
```

```
B = {8, -3, -9}
```

```
print(A.union(B))
```

Aufgabe 12.7

```
A = {4, -3, -2, -9}
```

```
B = {8, -3, -9}
```

```
print(A.union(B))
```

```
{4, -9, 8, -3, -2}
```

Aufgabe 12.8

```
A = {-2, 2, 9, 7}
```

```
B = {9, -2, 7, 2}
```

```
print(A.union(B))
```

Aufgabe 12.8

```
A = {-2, 2, 9, 7}
```

```
B = {9, -2, 7, 2}
```

```
print(A.union(B))
```

```
{2, 7, 9, -2}
```

Aufgabe 12.9

A = {0, 4, 5, -9}

B = {0, -6, -9}

`print(A & B)`

Aufgabe 12.9

A = {0, 4, 5, -9}

B = {0, -6, -9}

`print(A & B)`

{0, -9}

Aufgabe 12.10

A = {9, 2, 4, -4, 6}

B = {8, 1, 5, 0, -9}

`print(A & B)`

Aufgabe 12.10

A = {9, 2, 4, -4, 6}

B = {8, 1, 5, 0, -9}

`print(A & B)`

`set()`

Aufgabe 12.11

```
A = {7, -4, -2, -9}
```

```
B = {-8, -1, -2, 7}
```

```
print(A.intersection(B))
```

Aufgabe 12.11

$A = \{7, -4, -2, -9\}$

$B = \{-8, -1, -2, 7\}$

```
print(A.intersection(B))
```

$\{-2, 7\}$

Aufgabe 12.12

```
A = {-7, 9, 3, -2, -9}
B = {-8, 4, -4, 5, -1}
print(A.intersection(B))
```

Aufgabe 12.12

```
A = {-7, 9, 3, -2, -9}
B = {-8, 4, -4, 5, -1}
print(A.intersection(B))
```

set()

Aufgabe 12.13

```
A = {9, -1, 5, -2, 7}
```

```
B = {9, 5, -2, 1}
```

```
print(A - B)
```

Aufgabe 12.13

$A = \{9, -1, 5, -2, 7\}$

$B = \{9, 5, -2, 1\}$

`print(A - B)`

$\{-1, 7\}$

Aufgabe 12.14

$A = \{0, 4, -4, -1, -7\}$

$B = \{8, -8, 5, -3, -9\}$

`print(A - B)`

Aufgabe 12.14

$A = \{0, 4, -4, -1, -7\}$

$B = \{8, -8, 5, -3, -9\}$

`print(A - B)`

$\{0, 4, -7, -4, -1\}$

Aufgabe 12.15

```
A = {9, -4, 5}
```

```
B = {9, 2, -4, 5, -3}
```

```
print(A - B)
```

Aufgabe 12.15

```
A = {9, -4, 5}
```

```
B = {9, 2, -4, 5, -3}
```

```
print(A - B)
```

```
set()
```

Aufgabe 12.16

```
A = {8, -6, 4, -3, -2}
```

```
B = {8, -6, -3, 6, 7}
```

```
print(A.difference(B))
```

Aufgabe 12.16

$A = \{8, -6, 4, -3, -2\}$

$B = \{8, -6, -3, 6, 7\}$

```
print(A.difference(B))
```

$\{4, -2\}$

Aufgabe 12.17

```
A = {8, -8, 2, -5, 6}
B = {1, -7, -4, 9, 7}
print(B.difference(A))
```

Aufgabe 12.17

```
A = {8, -8, 2, -5, 6}
```

```
B = {1, -7, -4, 9, 7}
```

```
print(B.difference(A))
```

```
{1, 7, 9, -7, -4}
```

Aufgabe 12.18

```
A = {9, -5, 1}
```

```
B = {9, -5, -3, 1}
```

```
print(A.difference(B))
```


Aufgabe 12.18

```
A = {9, -5, 1}
```

```
B = {9, -5, -3, 1}
```

```
print(A.difference(B))
```

```
set()
```

Aufgabe 12.19

```
A = set()
B = {8, -8, -5}
print(A < B)
```

Aufgabe 12.19

```
A = set()
B = {8, -8, -5}
print(A < B)
```

True

Aufgabe 12.20

```
A = {8, 1, -4, -3}
```

```
B = {8, 1, -4, -3}
```

```
print(A < B)
```

Aufgabe 12.20

```
A = {8, 1, -4, -3}
```

```
B = {8, 1, -4, -3}
```

```
print(A < B)
```

False

Aufgabe 12.21

```
A = {-7, -1, -2, 5}
```

```
B = {8, 4, 5, -1}
```

```
print(A > B)
```

Aufgabe 12.21

```
A = {-7, -1, -2, 5}
```

```
B = {8, 4, 5, -1}
```

```
print(A > B)
```

False

Aufgabe 12.22

```
A = set()
B = {8, 5, -9}
print(A.issubset(B))
```


Aufgabe 12.22

```
A = set()
B = {8, 5, -9}
print(A.issubset(B))
```

True

Aufgabe 12.23

```
A = {1, 3, -4, -1}
B = {1, 3, -4, -1}
print(A.issubset(B))
```

Aufgabe 12.23

```
A = {1, 3, -4, -1}
B = {1, 3, -4, -1}
print(A.issubset(B))
```

True

Aufgabe 12.24

```
A = {-8, 2, -5, 7}
B = {8, 3, -5, -1}
print(A.issuperset(B))
```

Aufgabe 12.24

```
A = {-8, 2, -5, 7}
B = {8, 3, -5, -1}
print(A.issuperset(B))
```

False

Aufgabe 12.25

```
A = {9, 3, -4, -7}
B = {-4, 6, -2, -9}
print(A == B)
```

Aufgabe 12.25

```
A = {9, 3, -4, -7}
B = {-4, 6, -2, -9}
print(A == B)
```

False

Aufgabe 12.26

```
A = {1, 9, 5, 7}
B = {9, 6, -2, 1}
print(A != B)
```


Aufgabe 12.26

```
A = {1, 9, 5, 7}
B = {9, 6, -2, 1}
print(A != B)
```

True

Aufgabe 12.27

```
A = {-7, -5, -1, 7}
```

```
B = {-8, 9, -6, -9}
```

```
print(A.isdisjoint(B))
```

Aufgabe 12.27

```
A = {-7, -5, -1, 7}
```

```
B = {-8, 9, -6, -9}
```

```
print(A.isdisjoint(B))
```

True

Aufgabe 12.28

```
A = {8, 1, 2, 7}
```

```
B = {2, -4, 7}
```

```
print(A.isdisjoint(B))
```

Aufgabe 12.28

```
A = {8, 1, 2, 7}
```

```
B = {2, -4, 7}
```

```
print(A.isdisjoint(B))
```

False

Aufgabe 12.29

```
A = {0, 8, -3, 7}
A.discard(8)
print(A)
```

Aufgabe 12.29

```
A = {0, 8, -3, 7}
```

```
A.discard(8)
```

```
print(A)
```

```
{0, -3, 7}
```

Aufgabe 12.30

```
A = {-7, -6, -4, 5}
A.discard(-1)
print(A)
```


Aufgabe 12.30

```
A = {-7, -6, -4, 5}
A.discard(-1)
print(A)
```

```
{-7, -6, -4, 5}
```

Aufgabe 12.31

```
a = {3, -4, -5, 7}
a.clear()
print(a)
```

Aufgabe 12.31

```
a = {3, -4, -5, 7}
```

```
a.clear()
```

```
print(a)
```

```
set()
```

Aufgabe 12.32

```
A = {4, -3, -1}
```

```
A.add(-3)
```

```
print(A)
```

Aufgabe 12.32

```
A = {4, -3, -1}
```

```
A.add(-3)
```

```
print(A)
```

```
{4, -3, -1}
```

Aufgabe 12.33

```
A = {-8, 3, 6}
A.add(-6)
print(A)
```

Aufgabe 12.33

```
A = {-8, 3, 6}
```

```
A.add(-6)
```

```
print(A)
```

```
{-8, -6, 3, 6}
```

Aufgabe 12.34

```
A = {3, -4, 7}
B = {2, 3, -6, 7}
A.update(B)
print(A)
```


Aufgabe 12.34

```
A = {3, -4, 7}
```

```
B = {2, 3, -6, 7}
```

```
A.update(B)
```

```
print(A)
```

```
{2, 3, 7, -6, -4}
```

Aufgabe 12.35

```
A = {9, -6, 6, -1, -7}
for e in A:
    if e % 2 == 0:
        print(e)
```

Aufgabe 12.35

```
A = {9, -6, 6, -1, -7}
```

```
for e in A:
```

```
    if e % 2 == 0:
```

```
        print(e)
```

6

-6

Aufgabe 13.1

```
import module1
```

```
print(a)
```

Aufgabe 13.1

```
import module1
```

```
print(a)
```

Traceback (most recent call last):

File "python-13-ueb-01.py", line 3, in <module>

print(a)

NameError: name 'a' is not defined

Aufgabe 13.2

```
from module2 import *
```

```
a = 1
```

```
print(a)
```

Aufgabe 13.2

```
from module2 import *
```

```
a = 1
```

```
print(a)
```

```
1
```

Aufgabe 13.3

```
a = 3
```

```
from module1 import *
```

```
print(a)
```


Aufgabe 13.3

```
a = 3
```

```
from module1 import *
```

```
print(a)
```

5

Aufgabe 13.4

```
from module2 import b as a
```

```
print(a)
```

Aufgabe 13.4

```
from module2 import b as a
```

```
print(a)
```

```
1
```

Aufgabe 13.5

```
import module1
import module2

print(module1.fun1(3) + module2.fun1(4))
```

Aufgabe 13.5

```
import module1
import module2

print(module1.fun1(3) + module2.fun1(4))
```

47

Aufgabe 13.6

```
from module1 import fun1 as f
from module2 import a

print(f(a))
```

Aufgabe 13.6

```
from module1 import fun1 as f
from module2 import a

print(f(a))
```

7

Aufgabe 13.7

```
from module2 import *
from module1 import *

print(fun2(3,2))
```


Aufgabe 13.7

```
from module2 import *
from module1 import *

print(fun2(3,2))
```

5

Aufgabe 13.8

```
import math
```

```
print(math.sqrt(9))
```

Aufgabe 13.8

```
import math
```

```
print(math.sqrt(9))
```

```
3.0
```

Aufgabe 13.9

```
from math import pi, sin

print(sin(pi/2))
```

Aufgabe 13.9

```
from math import pi, sin
```

```
print(sin(pi/2))
```

```
1.0
```

Aufgabe 14.1

```
def myfunction(x, y):  
    try:  
        z = 2*x - y  
        return (x+y)/z  
    except:  
        return 0  
  
print(myfunction(3,2))  
print(myfunction(2,1))
```

Aufgabe 14.1

```
def myfunction(x, y):  
    try:  
        z = 2*x - y  
        return (x+y)/z  
    except:  
        return 0
```

```
print(myfunction(3,2))  
print(myfunction(2,1))
```

1.25

1.0

Aufgabe 14.2

```
def myfunction(x):  
    try:  
        return x**0.5  
    except:  
        return None  
  
print(myfunction('a'))  
print(myfunction(4))
```


Aufgabe 14.2

```
def myfunction(x):
    try:
        return x**0.5
    except:
        return None

print(myfunction('a'))
print(myfunction(4))
```

None

2.0

Aufgabe 15.1

```
1  class Example:
2
3      a = 3
4
5      def xyz(s, t):
6          return s * t
7
8      def __init__(self, u, v):
9          self.u = u
10         self.v = v
11
12     def abc(self):
13         return (self.u + self.v)
```

Gib die Zeilennummern an, in denen

- (a) Instanzvariablen,
- (b) Instanzmethoden,
- (c) Klassenvariablen,
- (d) Klassenmethoden

stehen.

Aufgabe 15.1

```

1  class Example:
2
3      a = 3
4
5      def xyz(s, t):
6          return s * t
7
8      def __init__(self, u, v):
9          self.u = u
10         self.v = v
11
12     def abc(self):
13         return (self.u + self.v)

```

- (a) Instanzvariablen: 9, 10
- (b) Instanzmethoden: 12–13
- (c) Klassenvariablen: 3
- (d) Klassenmethoden: 5–6

Aufgabe 15.2

Schreibe das folgende Programm ab und studiere seinen Output.

```
class Exercise:
```

```
    def __init__(self):  
        self.a = -1  
        self.b = 3.14  
        self.c = 'Hello'  
        self.d = False
```

```
x = Exercise()  
print(type(x))  
print(type(x.a))  
print(type(x.b))  
print(type(x.c))  
print(type(x.d))
```

Aufgabe 15.2

```
class Exercise:

    def __init__(self):
        self.a = -1
        self.b = 3.14
        self.c = 'Hello'
        self.d = False

x = Exercise()
print(type(x))
print(type(x.a))
print(type(x.b))
print(type(x.c))
print(type(x.d))

<class '__main__.Exercise'>
<class 'int'>
<class 'float'>
```

Aufgabe 15.3

Welche Ausgabe macht das folgende Codefragment?

```
class Example:
```

```
    c = 3
```

```
    def __init__(self, b):
        self.a = Example.c + b
```

```
x = Example(5)
```

```
print(x.a)
```

Aufgabe 15.3

```
class Example:

    c = 3

    def __init__(self, b):
        self.a = Example.c + b

x = Example(5)
print(x.a)
```

8

Aufgabe 15.4

Welche Ausgabe macht das folgende Programm?

```
class StrangeMath:

    def __init__(self, x):
        self.x = x

    def __str__(self):
        return '{0}'.format(self.x)

    def __add__(self, other):
        return StrangeMath(self.x * other.x)

a = StrangeMath(4)
b = StrangeMath(1)
print(a + b)
```

Aufgabe 15.4

```
class StrangeMath:

    def __init__(self, x):
        self.x = x

    def __str__(self):
        return '{0}'.format(self.x)

    def __add__(self, other):
        return StrangeMath(self.x * other.x)

a = StrangeMath(4)
b = StrangeMath(1)
print(a + b)
```

Aufgabe 15.5

Welche Ausgabe macht das folgende Programm?

```
class Quadrat():

    n=0

    def __init__(self, a):
        self.a = a
        Quadrat.n += 1

    def flaeche(self):
        return self.a * self.a

a = Quadrat(5)
b = Quadrat(4)
print(Quadrat.n)
```

Aufgabe 15.5

```
class Quadrat():

    n=0

    def __init__(self, a):
        self.a = a
        Quadrat.n += 1

    def flaeche(self):
        return self.a * self.a

a = Quadrat(5)
b = Quadrat(4)
print(Quadrat.n)
```

Aufgabe 15.6

```
class Konto:
    def __init__(self, name, vorname, nr, saldo=0):
        self.name = name
        self.vorname = vorname
        self.nr = nr
        self.saldo = saldo

    def einzahlung(self, betrag):
        self.saldo = self.saldo + betrag

    def auszahlung(self, betrag):
        self.saldo = self.saldo - betrag

k = Konto('Muster', 'Tim', '01-234-567', 7350)
k.einzahlung(300)
k.auszahlung(400)
print(k.name, k.saldo)
```

Aufgabe 15.6

```
class Konto:
    def __init__(self, name, vorname, nr, saldo=0):
        self.name = name
        self.vorname = vorname
        self.nr = nr
        self.saldo = saldo

    def einzahlung(self, betrag):
        self.saldo = self.saldo + betrag

    def auszahlung(self, betrag):
        self.saldo = self.saldo - betrag

k = Konto('Muster', 'Tim', '01-234-567', 7350)
k.einzahlung(300)
k.auszahlung(400)
print(k.name, k.saldo)
```

Aufgabe 15.7

Welche Ausgabe macht das folgende Programm?

```
class Parent:
    def __init__(self, a):
        self.a = a
    def methodA(self):
        return (self.a + 1)
    def methodB(self):
        return (self.a + 2)

class Child(Parent):
    def __init__(self, a):
        super().__init__(a)
    def methodA(self):
        return (self.a + 3)

x = Child(1)
print(x.methodA() * x.methodB())
```

Aufgabe 15.7

```
class Parent:
    def __init__(self, a):
        self.a = a
    def methodA(self):
        return (self.a + 1)
    def methodB(self):
        return (self.a + 2)

class Child(Parent):
    def __init__(self, a):
        super().__init__(a)
    def methodA(self):
        return (self.a + 3)

x = Child(1)
print(x.methodA() * x.methodB())
```


Aufgabe 15.8

Implementiere gemäss dem folgenden Klassendiagramm eine Klasse für die Berechnung von Volumen und Oberfläche von Quader-Objekten.

Quader
<code>a: float</code> <code>b: float</code> <code>c: float</code>
<code>Quader(a: float, b: float, c: float)</code> <code>volumen(): float</code> <code>oberflaeche(): float</code>

Aufgabe 15.8

```
class Quader:

    def __init__(self, a, b, c):
        self.a = a
        self.b = b
        self.c = c

    def volumen(self):
        return self.a * self.b * self.c

    def oberflaeche(self):
        return 2*(self.a*self.b + self.b*self.c
                  + self.c*self.a)
```

Aufgabe 15.9

Implementiere aufgrund des folgenden Klassendiagramms eine Klasse `Fach` zum Verwalten von Prüfungsnoten in einem Schulfach. *Hinweise:*

- ▶ Der Konstruktor soll die Variable `noten` mit der leeren Liste initialisieren.
- ▶ Die Methode `neueNote()` soll die Notenliste um eine Note erweitern.
- ▶ Die Methode `mittelwert()` soll den Durchschnitt der darin enthaltenen Noten berechnen und als Wert zurückgeben.

Fach
name: str noten: list
Fach(name: str) neueNote(note: float)

Aufgabe 15.9

```
class Fach:

    def __init__(self, name):
        self.name = name
        self.noten = []

    def neueNote(self, note):
        self.note.append(note)

    def mittelwert(self):
        return sum(self.noten)/len(noten)
```

Aufgabe 15.10

Was ist eine Klasse?

Aufgabe 15.10

Was ist eine Klasse?

Ein Bauplan für Objekte

Aufgabe 15.11

Was ist eine Instanz?

Aufgabe 15.11

Was ist eine Instanz?

Ein Objekt, das zur Laufzeit aus einer Klasse erzeugt wird.

Aufgabe 15.12

Was ist eine Methode?

Aufgabe 15.12

Was ist eine Methode?

Eine Funktion, die zu einer Klasse gehört.

Aufgabe 15.13

Was ist ein Konstruktor?

Aufgabe 15.13

Was ist ein Konstruktor?

Eine spezielle Methode, mit der eine Objekt initialisiert wird.