

Python

Übungen

16. September 2025

Aufgabe 1.1

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

Aufgabe 1.1

einen Texteditor (ein Programm mit dem man Texte schreiben und bearbeiten kann.)

Aufgabe 1.2

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

Aufgabe 1.2

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

Aufgabe 1.3

Zähle die drei Fehlerarten beim Programmieren auf und nenne jeweils ein Beispiel dazu.

Aufgabe 1.3

- ▶ 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 1.4

Welches Zeichen leitet in Python einen Kommentar ein.

Aufgabe 1.4

Das Rautenzeichen #

Aufgabe 1.5

Welche Dateiendung haben Python-Programme normalerweise?

Aufgabe 1.5

.py

Aufgabe 2.1

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

Aufgabe 2.1

3

Aufgabe 2.2

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

Aufgabe 2.2

-3

Aufgabe 2.3

```
print(35 / 7)
```


Aufgabe 2.3

5.0

Aufgabe 2.4

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

Aufgabe 2.4

5

Aufgabe 2.5

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

Aufgabe 2.5

2

Aufgabe 2.6

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

Aufgabe 2.6

0

Aufgabe 2.7

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


Aufgabe 2.7

3

Aufgabe 2.8

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

Aufgabe 2.8

8

Aufgabe 2.9

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

Aufgabe 2.9

2.0

Aufgabe 2.10

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

Aufgabe 2.10

1

Aufgabe 2.11

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


Aufgabe 2.11

-8

Aufgabe 2.12

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

Aufgabe 2.12

-9

Aufgabe 2.13

```
print((-1)**4)
```

Aufgabe 2.13

1

Aufgabe 2.14

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

Aufgabe 2.14

0.01

Aufgabe 2.15

```
print(abs(-17))
```


Aufgabe 2.15

17

Aufgabe 2.16

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

Aufgabe 2.16

512

Aufgabe 2.17

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

Aufgabe 2.17

64

Aufgabe 2.18

```
print(bin(7))
```

Aufgabe 2.18

0b111

Aufgabe 2.19

```
print(hex(31))
```


Aufgabe 2.19

0x1f

Aufgabe 2.21

```
print(not False)
```

Aufgabe 2.21

True

Aufgabe 2.22

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

Aufgabe 2.22

False

Aufgabe 2.23

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

Aufgabe 2.23

True

Aufgabe 2.24

```
print(not False and True)
```


Aufgabe 2.24

True

Aufgabe 2.25

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

Aufgabe 2.25

True

Aufgabe 2.26

```
print(False or True and not False or False)
```

Aufgabe 2.26

True

Aufgabe 2.27

```
print(7 + 3 != 11 - 1)
```

Aufgabe 2.27

False

Aufgabe 2.28

```
print(4 + 3 <= 8 - 2)
```


Aufgabe 2.28

False

Aufgabe 2.29

```
print(7 <= 5 or 9 > 8)
```

Aufgabe 2.29

True

Aufgabe 2.30

```
print(int(True))
```

Aufgabe 2.30

1

Aufgabe 2.31

```
print(int(7.9))
```

Aufgabe 2.31

7

Aufgabe 2.32

```
print(int(False))
```


Aufgabe 2.32

0

Aufgabe 2.33

```
print(bool(2.3))
```

Aufgabe 2.33

True

Aufgabe 2.34

```
print(bool('', ''))
```

Aufgabe 2.34

False

Aufgabe 2.35

```
print(bool('abc'))
```

Aufgabe 2.35

True

Aufgabe 2.36

```
print(bool(0))
```


Aufgabe 2.36

False

Aufgabe 2.37

```
print(float(3))
```

Aufgabe 2.37

3.0

Aufgabe 2.38

```
print(True/(True + True))
```

Aufgabe 2.38

0.5

Aufgabe 3.1

Welche der folgenden Bezeichner (Variablennamen) sind syntaktisch korrekt?

(a) 4Musketiere

(b) tom&jerry

(c) peter_pan

(d) U2

(e) _4_

(f) kollegi-stans

(g) if

(h) true

Aufgabe 3.1

(a) 4Musketiere

Aufgabe 3.1

(a) 4Musketiere nein (beginnt mit Ziffer)

(b) tom&jerry

Aufgabe 3.1

- (a) `4Musketiere` nein (beginnt mit Ziffer)
- (b) `tom&jerry` nein (enthält Sonderzeichen)
- (c) `peter_pan`

Aufgabe 3.1

- (a) 4Musketiere nein (beginnt mit Ziffer)
- (b) tom&jerry nein (enthält Sonderzeichen)
- (c) peter_pan ja
- (d) U2

Aufgabe 3.1

- (a) 4Musketiere nein (beginnt mit Ziffer)
- (b) tom&jerry nein (enthält Sonderzeichen)
- (c) peter_pan ja
- (d) U2 ja
- (e) _4_

Aufgabe 3.1

- (a) 4Musketiere nein (beginnt mit Ziffer)
- (b) tom&jerry nein (enthält Sonderzeichen)
- (c) peter_pan ja
- (d) U2 ja
- (e) _4_ ja (Underscore erlaubt)
- (f) kollegi-stans

Aufgabe 3.1

- (a) `4Musketiere` nein (beginnt mit Ziffer)
- (b) `tom&jerry` nein (enthält Sonderzeichen)
- (c) `peter_pan` ja
- (d) `U2` ja
- (e) `_4_` ja (Underscore erlaubt)
- (f) `kollegi-stans` nein (enthält Sonderzeichen)
- (g) `if`

Aufgabe 3.1

- (a) `4Musketiere` nein (beginnt mit Ziffer)
- (b) `tom&jerry` nein (enthält Sonderzeichen)
- (c) `peter_pan` ja
- (d) `U2` ja
- (e) `_4_` ja (Underscore erlaubt)
- (f) `kollegi-stans` nein (enthält Sonderzeichen)
- (g) `if` nein (Python-Schlüsselwort)
- (h) `true`

Aufgabe 3.1

- (a) `4Musketiere` nein (beginnt mit Ziffer)
- (b) `tom&jerry` nein (enthält Sonderzeichen)
- (c) `peter_pan` ja
- (d) `U2` ja
- (e) `_4_` ja (Underscore erlaubt)
- (f) `kollegi-stans` nein (enthält Sonderzeichen)
- (g) `if` nein (Python-Schlüsselwort)
- (h) `true` ja (nicht empfohlen)

Aufgabe 3.2

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


Aufgabe 3.2

1

Aufgabe 3.3

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

Aufgabe 3.3

2

Aufgabe 3.4

```
a = 1
```

```
b = 2
```

```
a = b
```

```
b = a
```

```
print(a+b)
```

Aufgabe 3.4

4

Aufgabe 3.5

```
a = 4
```

```
b = 9
```

```
c = a
```

```
b = c
```

```
a = b
```

```
print(a)
```

Aufgabe 3.5

4

Aufgabe 3.6

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


Aufgabe 3.6

90

Aufgabe 3.7

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

Aufgabe 3.7

6

Aufgabe 3.8

```
a = 7
```

```
b = 2
```

```
a, b = b, a
```

```
print(a)
```

Aufgabe 3.8

2

Aufgabe 3.9

```
a, b, c = 5, 3, 8
```

```
x, y, z = c, a, b
```

```
print(z)
```

Aufgabe 3.9

3

Aufgabe 4.1

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


Aufgabe 4.1

3

Aufgabe 4.2

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

Aufgabe 4.2

8

Aufgabe 4.3

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

Aufgabe 4.3

4

Aufgabe 4.4

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

Aufgabe 4.4

10

Aufgabe 4.5

```
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 4.5

12

Aufgabe 4.6

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

Aufgabe 4.6

1

Aufgabe 4.7

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

Aufgabe 4.7

4

Aufgabe 5.1

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

Aufgabe 5.1

1

2

3

4

5

Aufgabe 5.2

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


Aufgabe 5.2

1

4

9

3

Aufgabe 5.3

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

Aufgabe 5.3

0

1

2

3

4

Aufgabe 5.4

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

Aufgabe 5.4

1

2

3

4

5

Aufgabe 5.5

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

Aufgabe 5.5

4

3

2

1

0

Aufgabe 5.6

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


Aufgabe 5.6

0

3

6

9

12

Aufgabe 5.7

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

Aufgabe 5.7

8

4

6

Aufgabe 5.8

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

Aufgabe 5.8

30

Aufgabe 5.9

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

Aufgabe 5.9

6

7

15

22

27

Aufgabe 5.10

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


Aufgabe 5.10

0 7

1 4

2 8

Aufgabe 5.11

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

Aufgabe 5.11

66

Aufgabe 5.12

```
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 5.12

89

Aufgabe 5.13

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

Aufgabe 5.13

5

4

36

Aufgabe 5.14

```
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 5.14

9

Aufgabe 5.15

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

Aufgabe 5.15

2 4

2 5

2 6

3 4

3 5

3 6

4 4

4 5

4 6

Aufgabe 5.16

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

Aufgabe 5.16

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

Aufgabe 5.17

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

Aufgabe 5.17

Aufgabe 5.18

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


Aufgabe 5.18

2

3

Aufgabe 5.19

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

Aufgabe 5.19

5

Aufgabe 5.20

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

Aufgabe 5.20

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

Aufgabe 6.1

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

Aufgabe 6.1

-6

Aufgabe 6.2

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


Aufgabe 6.2

Traceback (most recent call last):

File

"/home/ge/stans/inf/python-rep/ueb/src/python-06-ueb-02.py

line 2, in <module>

print(a[5])

~~~~

IndexError: list index out of range

## Aufgabe 6.3

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

# Aufgabe 6.3

3

## Aufgabe 6.4

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

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

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

# Aufgabe 6.4

10

## Aufgabe 6.5

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

# Aufgabe 6.5

6

## Aufgabe 6.6

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



## Aufgabe 6.6

[-2]

## Aufgabe 6.7

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

## Aufgabe 6.7

$[0, -9]$

## Aufgabe 6.8

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

## Aufgabe 6.8

[1, 7, 4, -1]

## Aufgabe 6.9

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

## Aufgabe 6.9

[0, 9, 6, 3, 8]

## Aufgabe 6.10

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



## Aufgabe 6.10

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

## Aufgabe 6.11

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

# Aufgabe 6.11

5

## Aufgabe 6.12

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

# Aufgabe 6.12

0

## Aufgabe 6.13

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

## Aufgabe 6.13

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

## Aufgabe 6.14

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



## Aufgabe 6.14

[2, 2, 2, 2]

## Aufgabe 6.15

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

## Aufgabe 6.15

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

## Aufgabe 6.16

```
a = [9, 3, -4, 7, -1]
```

```
a.pop()
```

```
print(a)
```

## Aufgabe 6.16

[9, 3, -4, 7]

## Aufgabe 6.17

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

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

```
print(a)
```

## Aufgabe 6.17

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

## Aufgabe 6.18

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



## Aufgabe 6.18

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

## Aufgabe 6.19

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

## Aufgabe 6.19

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

## Aufgabe 6.20

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

## Aufgabe 6.20

-8

6

2

-3

-1

## Aufgabe 6.21

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

## Aufgabe 6.21

-4

9

2

6

4

## Aufgabe 6.22

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



## Aufgabe 6.22

-6

11

4

8

3

## Aufgabe 6.23

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

## Aufgabe 6.23

81

4

16

49

16

## Aufgabe 6.24

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

## Aufgabe 6.24

[0, 1, 0, 1, 0]

## Aufgabe 6.25

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

## Aufgabe 6.25

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

## Aufgabe 6.26

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



## Aufgabe 6.26

9

5

7

## Aufgabe 7.1

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

# Aufgabe 7.1

11

## Aufgabe 7.2

```
def f(x):  
    y = 6  
    return x*y  
  
print(f(10))
```

# Aufgabe 7.2

60

## Aufgabe 7.3

```
def f(x):  
    return 7  
  
print(f(23))
```

# Aufgabe 7.3

7

## Aufgabe 7.4

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



# Aufgabe 7.4

15

## Aufgabe 7.5

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

# Aufgabe 7.5

8

## Aufgabe 7.6

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

# Aufgabe 7.6

13

## Aufgabe 7.8

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

```
f(2.7)
```

# Aufgabe 7.8

3.7

## Aufgabe 7.9

```
def change(x):  
    y = x
```

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



# Aufgabe 7.9

3

## Aufgabe 7.10

```
def change(L):  
    L.append(4)
```

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

# Aufgabe 7.10

[1, 2, 3, 4]

## Aufgabe 7.11

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

# Aufgabe 7.11

15

## Aufgabe 7.12

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

# Aufgabe 7.12

17

## Aufgabe 7.13

```
def fun(x=42):  
    return x + 5  
  
print(fun(7))
```



# Aufgabe 7.13

12

## Aufgabe 7.14

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

# Aufgabe 7.14

47

## Aufgabe 7.15

```
def g(x):  
    4*x + 1  
  
print(g(5))
```

# Aufgabe 7.15

None

## Aufgabe 7.16

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

# Aufgabe 7.16

21

## Aufgabe 7.17

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



# Aufgabe 7.17

8

## Aufgabe 7.18

```
def f(x, y):  
    return 3*x - 2*y  
  
print(f(10, 1))
```

# Aufgabe 7.18

28

## Aufgabe 7.19

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

# Aufgabe 7.19

13

## Aufgabe 7.20

```
def rechteck(a=1, b=1):  
    return a * b  
  
print(rechteck(3, 8))
```

# Aufgabe 7.20

24

## Aufgabe 7.21

```
def rechteck(a=1, b=1):  
    return a * b  
  
print(rechteck(4))
```



# Aufgabe 7.21

4

## Aufgabe 7.22

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

# Aufgabe 7.22

1

## Aufgabe 7.23

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

# Aufgabe 7.23

4

## Aufgabe 7.24

```
def first(L):  
    return L[0]  
  
print(first([6, -5, 12, 3]))
```

# Aufgabe 7.24

6

## Aufgabe 7.25

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



# Aufgabe 7.25

18

## Aufgabe 7.26

```
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 7.26

17

## Aufgabe 7.27

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

# Aufgabe 7.27

5

## Aufgabe 7.28

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

# Aufgabe 7.28

2

## Aufgabe 7.29

```
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 7.29

18

## Aufgabe 7.30

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

# Aufgabe 7.30

24

## Aufgabe 7.31

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

# Aufgabe 7.31

5

## Aufgabe 7.32

```
def f(n):  
    if n == 1:  
        return 2  
    else:  
        return f(n-1)+3  
  
print(f(4))
```

# Aufgabe 7.32

11

## Aufgabe 8.1

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



# Aufgabe 8.1

abcxyz

## Aufgabe 8.2

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

# Aufgabe 8.2

anna

## Aufgabe 8.3

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

## Aufgabe 8.3

Das  
ist  
so.

## Aufgabe 8.4

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

## Aufgabe 8.4

$$3 + 4 = 7$$

## Aufgabe 8.5

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



## Aufgabe 8.5

30 Meter kosten 20 Fr.

## Aufgabe 8.6

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

# Aufgabe 8.6

Das.ist.schlecht

## Aufgabe 8.8

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

## Aufgabe 8.8

Das  
ist  
so.

## Aufgabe 8.9

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

## Aufgabe 8.9

Woher  
weisst du das?

## Aufgabe 8.10

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



## Aufgabe 8.10

Das ist vielleicht sinnlos!

## Aufgabe 8.11

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

# Aufgabe 8.11

15

## Aufgabe 8.12

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

# Aufgabe 8.12

12

## Aufgabe 8.13

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

# Aufgabe 8.13

1234.5

## Aufgabe 8.14

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



## Aufgabe 8.14

Das ist das Zeichen \.

## Aufgabe 8.15

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

# Aufgabe 8.15

```
Sag "Hallo"
```

## Aufgabe 8.16

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

## Aufgabe 8.16

?

## Aufgabe 8.17

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

# Aufgabe 8.17

65

## Aufgabe 8.18

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



# Aufgabe 8.18

A

## Aufgabe 8.19

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

# Aufgabe 8.19

Das.ist.schlecht

## Aufgabe 8.20

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

## Aufgabe 8.20

True

## Aufgabe 8.21

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

## Aufgabe 8.21

$$3 + 4 = 7$$

## Aufgabe 8.22

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



## Aufgabe 8.22

30 Meter kosten 20 Fr.

## Aufgabe 8.23

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

# Aufgabe 8.23

0b111

## Aufgabe 8.24

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

# Aufgabe 8.24

Gut gemacht!

## Aufgabe 8.25

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

# Aufgabe 8.25

2

## Aufgabe 8.26

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



# Aufgabe 8.26

1

## Aufgabe 8.27

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

# Aufgabe 8.27

25.2.2013

## Aufgabe 8.28

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

# Aufgabe 8.28

HAMMER

## Aufgabe 8.29

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

# Aufgabe 8.29

Hacker

## Aufgabe 8.30

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



# Aufgabe 8.30

Das ist gut

## Aufgabe 8.31

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

## Aufgabe 8.31

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

## Aufgabe 8.32

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

## Aufgabe 8.32

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

## Aufgabe 8.33

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

# Aufgabe 8.33

007

## Aufgabe 8.34

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



# Aufgabe 8.34

tim

## Aufgabe 9.1

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

# Aufgabe 9.1

abc\*def

...xyz\*123

## Aufgabe 9.2

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

## Aufgabe 9.2

123 45

67 8

## Aufgabe 9.3

Die Eingabe des Benutzers ist 7.

```
x = input('Eingabe: ')\nprint(2*x)
```

## Aufgabe 9.3

Eingabe:

## Aufgabe 9.4

Die Eingabe des Benutzers ist 7.

```
x = int(input('Eingabe: '))  
print(2*x)
```



## Aufgabe 9.4

Eingabe: 14

## Aufgabe 9.5

Die Eingabe des Benutzers ist 7.

```
x = float(input('Eingabe: '))  
print(2*x)
```

# Aufgabe 9.5

Eingabe: 14.0

## Aufgabe 9.6

Was steht nach der Ausführung des folgenden Programms in der Datei `python-09-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 9.6

In der Datei `python-09-ueb-06.out` steht:

12

34

## Aufgabe 9.7

Die Datei `python-09-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-09-ueb-07.data')
for zeile in fd:
    s += float(zeile)
    n += 1
fd.close()
print(s/n)
```

# Aufgabe 9.7

3.0

## Aufgabe 10.1

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



## Aufgabe 10.1

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

-6

## Aufgabe 10.2

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

## Aufgabe 10.2

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

6

## Aufgabe 10.3

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

## Aufgabe 10.3

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

d

## Aufgabe 10.4

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

## Aufgabe 10.4

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

3

## Aufgabe 10.5

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



## Aufgabe 10.5

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

3

## Aufgabe 10.6

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

## Aufgabe 10.6

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

```
Traceback (most recent call last):
```

```
...
```

```
    D.pop('b')
```

```
KeyError: 'b'
```

## Aufgabe 10.8

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

## Aufgabe 10.8

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

0

## Aufgabe 10.9

```
D = {'b': 6, 'g': 0, 'f': 8}
```

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

```
D.update(E)
```

```
print(D)
```

## Aufgabe 10.9

```
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 10.10

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



## Aufgabe 10.10

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

-4

## Aufgabe 10.11

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

## Aufgabe 10.11

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

e

d

f

## Aufgabe 10.12

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

## Aufgabe 10.12

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

a

c

d

## Aufgabe 10.13

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

## Aufgabe 10.13

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

-4

7

3

## Aufgabe 10.14

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



## Aufgabe 10.14

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

-3

-2

3

## Aufgabe 10.15

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

## Aufgabe 10.15

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

1

2

-1

## Aufgabe 10.16

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

## Aufgabe 10.16

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

3

-9

-5

## Aufgabe 10.17

```
A = [3, 7, 9]
B = [8, 5, 2]
D = dict(zip(A, B))
for x, y in D.items():
    print(x+y)
```

## Aufgabe 10.17

```
A = [3, 7, 9]
B = [8, 5, 2]
D = dict(zip(A, B))
for x, y in D.items():
    print(x+y)
```

11

12

11

## Aufgabe 11.1

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 11.1

8

## Aufgabe 11.2

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 11.2

4

## Aufgabe 11.3

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 11.3

2



## Aufgabe 11.4

Was ist ein Konstruktor?

## Aufgabe 11.4

Eine spezielle Methode, mit der eine Objekt initialisiert wird.



## Aufgabe 11.1

```
L = [5, 3, 7, 2, 6]
```

```
x = L[0]
```

```
for i in range(1, len(L)):
```

```
    if L[i] > x:
```

```
        x = L[i]
```

```
print(x)
```

# Aufgabe 11.1

7

## Aufgabe 11.2

```
s = 0
```

```
for a in range(1, 3):  
    for b in range(1, 3):  
        s += a*b
```

```
print(s)
```

# Aufgabe 11.2

9

## Aufgabe 11.3

```
L = [1, 1]
```

```
for i in range(2, 7):  
    L.append(L[i-1]+L[i-2])
```

```
print(L)
```

## Aufgabe 11.3

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

## Aufgabe 11.4

```
n = 5
```

```
while n != 1:  
    print(n)  
    if n % 2 == 0:  
        n = n//2  
    else:  
        n = 3*n+1
```

## Aufgabe 11.4

5

16

8

4

2