

Python

...für Anfänger

Unit 2

by

Dr. Günter Kolousek

Konvertierung

```
>>> float(2)
2.0
>>> int(1.0)
1
>>> int(1.5)
1
>>> int("2")
2
>>> float("-2.5")
-2.5
```

Konvertierung - 2

```
>>> print(1.5); print("Orangen und");\  
... print(2); print("Bananen")
```

1.5

Orangen und

2

Bananen

```
>>> print(str(1.5) + " Orangen und " + str(2) +\  
... " Bananen")
```

1.5 Orangen und 2 Bananen

print - 3

```
>>> print(1.5, end=" ");\n... print("Orangen und ", end="");\n... print(2, end=" "); print("Bananen")\n1.5 Orangen und 2 Bananen\n>>> print("1.5 Orangen und\\n2 Bananen")\n1.5 Orangen und\n2 Bananen
```

Modul - 2

- ▶ abgeschlossene Einheit einer Software,
- ▶ bestehend aus Code und Daten.
- ▶ In Python ist jede Datei mit der Endung `.py` ein Modul.
- ▶ `math` ist vordefiniertes Modul

```
>>> import math
>>> type(math)
<class 'module'>
```

Importieren

```
>>> import math
```

```
>>> math.pi
```

```
3.141592653589793
```

```
>>> pi
```

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

```
  File "<stdin>", line 1, in <module>
```

```
NameError: name 'pi' is not defined
```

Importieren - 2

```
>>> from math import pi
>>> pi
3.141592653589793
>>> from math import *
>>> sqrt(4)
2.0
```

Strings

- ▶ Folge von Zeichen
- ▶ doppelte Anführungszeichen (engl. single quotes):
`"Heute ist's kalt!"`
- ▶ einfache Anführungszeichen (engl. double quotes): `'Er sagt: "Heute ist es kalt!"'`
- ▶ Escape character (dt. Fluchtzeichen): `'Er sagt: "Heute ist\'s kalt!"'`
 - ▶ `\\` ... ein backslash
 - ▶ `\n` ... newline
 - ▶ `\'` ... ein einfaches Hochkomma
 - ▶ `\"` ... ein doppeltes Hochkomma

Strings - 2

- ▶ jeweils 3 einfache oder doppelte Anführungszeichen (engl. triple-quote strings)

```
>>> print("""Maxi: "Heute ist's kalt!" """)
```

```
Maxi: "Heute ist's kalt"
```

```
>>> """Maxi: "Heute ist's kalt!" """
```

```
'Maxi: "Heute ist\'s kalt!" '
```

```
>>> '''Maxi: "Heute ist's kalt!"'''
```

```
'Maxi: "Heute ist\'s kalt!"'
```

Kommentare und Doc-Strings

- ▶ Kommentar: alle Zeichen einer Zeile nach # werden ignoriert

1 + 1 # *ergibt 2*

- ▶ Docstrings
 - ▶ triple-quotes
 - ▶ außerhalb jeder Anweisung
 - ▶ in einem Programm
 - ▶ → wird ignoriert (wie Kommentar)

"""

*das ist ein Doc-String
und wird ignoriert*

"""

Funktion help

- ▶ help bekommt normalerweise ein Argument
- ▶ gibt Hilfetext zurück

```
>>> from turtle import *
```

```
>>> help(forward)
```

```
Help on function forward in module turtle:
```

```
forward(distance)
```

Bewegt die Turtle um distance nach vorne.

Aliases: forward | fd

Argument:

distance -- a number (integer or float)

Sequenz von Anweisungen

- ▶ Je Zeile eine (einfache) Anweisung!
- ▶ Mehrere einfache Anweisung je Zeile möglich (→ Programmierrichtlinien)

```
>>> print("a"); print("b"); print("c")
```

a

b

c