

Ausgabe

```
print "Hello World"
print " Hel lo", " Joh n"
name = " Joh n"
print " Hel lo", name
print " Hello {}".f or mat -
(name)
print # erzeugt leere Zeile und
Zeilen umbruch
```

Eingabe

```
eingabe = raw_input("Enter
something: ")
# Datentyp ist immer(!) String
# evtl. Casting zu anderem
Datentyp
try:
    eingabe = int(ei ngabe)
except ValueE rror, e:
    print " Error casting
the input."
    print " Mes sage:
{}".f or mat (e.m es sage)
```

Variablen

Kleinschreibung

Trennung mehrerer Wörter durch Unters-
trich

Keine Ziffer am Beginn, nur Buchstabe oder
Unterstrich

Sprechende Namen verwenden

Kommentare

Einzeilige Kommentare werden durch das
Hash-Zeichen (Raute) eingeleitet.
Mehrzeilige Kommentare werden mit drei
Anführungszeichen begonnen und müssen
auch wieder mit drei Anführungszeichen
beendet werden.

Kommentare (cont)

```
# Dies ist ein Kommentar
" " " Dies ist ein
langer Kommentar
" " "
```

Fehlerbehandlung

```
try:
    # Anweis ungen
except Except ionType e:
    # Anweis ungen
    # Zugriff auf Fehler -
mel dung: e.message
```

ExceptionType immer explizit angeben

Operatoren

x+y	Addition
x-y	Subtraktion
x*y	Multiplikation
x/y	(Integer-)Division
x%y	Modulo
x**y	Exponentiation

Bedingungen

<	kleiner als	wert < 10
>	größer als	wert > 5
==	gleich	wert == 23
<=	kleiner gleich	5 <= 7
>=	größer gleich	23 >= 22
!=	ungleich	78 != 93
in	(enthalten) in	b in " Lib ell - e"

Bedingungen (cont)

not in	nicht	y not in " Vog e
	(entha-	l"
	lten) in	

Datentypen

Integer	-25, 34
Float	-2.53, 43.1
String	"Hello", 'World'
Boolean	True, False
List	[value, ...]
Tupel	(value, ...)
Dictionary	{key:value, ...}

Wiederholungen / Schleifen

for-Schleife

```
container = [1,2,3]
for element in container:
    print element
```

while- Schleife

```
x = 5
while x > 0:
    print "x hat den Wert
{}".f or mat(x)
    x-=1
```

for-Schleife hat enumerate-Erweiterung zum
Mitzählen der Indexposition
Gefahr von Endlosschleifen beim Einsatz
von while



By Tobias Siebenlist (t7l)
cheatography.com/t7l/

Published 7th July, 2015.
Last updated 13th May, 2016.
Page 1 of 2.

Sponsored by **Readable.com**
Measure your website readability!
<https://readable.com>

Bedingte Abfragen

```
if bedingung:
    # Anweisungen
elif andere_bedingung:
    # Anweisungen
else:
    # Anweisungen
```

beliebig viele elif-Abschnitte möglich
else erhält keine Bedingung
Abarbeitung von oben nach unten

Funktionen

```
def funktionsname(parameter1,
parameter2):
    ergebnis = parameter1 +
parameter2
    return ergebnis
```

Laden und Verwenden eines Moduls

Importanweisung	Befehlsaufruf
<code>import modulname</code>	<code>modulname.methode()</code>

`dir(modulname)` zeigt alle Methoden eines Moduls an
`help(methode)` zeigt Hilfe für eine Methode an

Slicing bei Sequenzen

0	1	2	3	4	5	6	7	8	9	10	11
M	o	n	t	y		P	y	t	h	o	n
-12	-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1

[6:10] (highlighted)
[-12:-7] (highlighted)

Arbeiten mit Strings

<code>len(mystr)</code>	Anzahl der Zeichen von mystr
<code>mystr.lower()</code>	Umwandlung zu Kleinbuchstaben
<code>mystr.upper()</code>	Umwandlung zu Großbuchstaben
<code>mystr.replace(old, new)</code>	old durch new ersetzen
<code>mystr.split(char)</code>	teilt an char auf, erzeugt Liste
<code>mystr[1:5]</code>	Zeichen 1-5 ausschneiden
<code>mystr[1:5:2]</code>	nur jedes zweite Zeichen

Arbeiten mit Dateien

```
with open(filename, modus) as variable_name:
    # Anweisungen, z.B.
    cvariable_name.read()
    clvariable_name.readlines()
    variable_name.write(some_string_content)

read() liest Inhalt komplett in einen String ein
readlines() liest Inhalt zeilenweise in eine Liste ein
write(some_string_content) schreibt Stringinhalt in Datei
```

Arbeiten mit Listen

<code>len(my_list)</code>	Anzahl der Elemente
<code>mylist[i]</code>	i-tes Element der Liste
<code>mylist[i:j]</code>	Ausschnitt von i bis j
<code>mylist[i:j:2]</code>	nur jedes zweite Element
<code>x in mylist</code>	True, wenn x in mylist ist
<code>mylist.append(x)</code>	x an mylist anhängen
<code>mylist[i] = x</code>	Element an Stelle i ersetzen
<code>"c".join(my_list)</code>	erzeugt String, c verbindet

Arbeiten mit Dictionaries

<code>len(my_dict)</code>	Anzahl der Einträge
<code>del mydict[key]</code>	löscht Schlüssel key
<code>mydict.keys()</code>	Liste aller Schlüssel
<code>mydict.values()</code>	Liste aller Werte
<code>mydict.items()</code>	Liste von Tupeln
<code>key in mydict</code>	True, wenn key vorhanden



By Tobias Siebenlist (t7l)
cheatography.com/t7l/

Published 7th July, 2015.
Last updated 13th May, 2016.
Page 2 of 2.

Sponsored by [Readable.com](https://readable.com)
Measure your website readability!
<https://readable.com>