Cheatography

Projektseminare: Python 3 Cheat Sheet

by Tobias Siebenlist (t7l) via cheatography.com/22337/cs/7477/

Ausgabe

```
print("Hello World")
print("Hello", "John")
name = "John"
print("Hello", name)
print("Hello
{}".format(name))
print() # erzeugt leere
Zeile und Zeilenumbruch
```

Eingabe

```
eingabe = input("Eingabe: ")
# Datentyp ist immer(!)
String
# evtl. Casting zu anderem
Datentyp
try:
    eingabe = int(eingabe)
except ValueError as err:
    print("Fehler!")
    print("Meldung:
{}".format(err))
```

Variablen

Kleinschreibung

Trennung mehrerer Wörter durch Unterstrich

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.

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

Fehlerbehandlung

```
try:
    # Anweisungen
except ExceptionType as err:
    # Anweisungen
    # Zugriff auf Fehlermeldung: err
```

ExceptionType immer explizit angeben

Operatoren

`x+y`	Addition
`x-y`	Subtraktion
`x*y`	Multiplikation
`x/y`	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 "Libelle"`
`not in`	nicht (enthalten) in	`y not in "Vogel"`

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: {}".format(x))
    x-=1
```

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

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 fname(param1, param2):
    ergebnis = param1 + param2
    return ergebnis
```

Laden und Verwenden eines Moduls

Importanweisung	Befehlsaufruf
`import modulname`	`modulname.methode()`

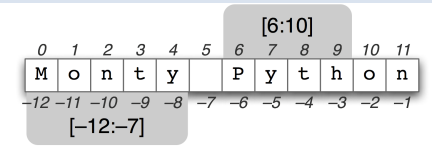
`dir(modulname)` zeigt alle Methoden eines Moduls an
`help(methode)` zeigt Hilfe für eine Methode an

Cheatography

Projektseminare: Python 3 Cheat Sheet

by Tobias Siebenlist (t7l) via cheatography.com/22337/cs/7477/

Slicing bei Sequenzen



Arbeiten mit Strings

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

Arbeiten mit Dateien

```
with open(fname, mod) as var:
    # Anweisungen, z.B.
    c = var.read()
    cl = var.readlines()
    var.write(string_content)
```

`read()` liest Inhalt komplett in einen String ein

`readlines()` liest Inhalt zeilenweise in eine Liste ein

`write(string_content)` schreibt Stringinhalt in Datei

Arbeiten mit Listen

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

Arbeiten mit Dictionaries

`len(mydict)`	Anzahl der Einträge
`del mydict[key]`	löscht Schlüssel key
`list(mydict.keys())`	Liste aller Schlüssel
`list(mydict.values())`	Liste aller Werte
`list(mydict.items())`	Liste von Tupeln
`key in mydict`	True, wenn key vorhanden

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

Published 14th April, 2016.
Last updated 14th April, 2016.