Cheatography

Python 3 Cheat Sheet

by aragow via cheatography.com/24054/cs/11425/

Grundlagen

python -V**	Verison Python 2
python3 -V	Version Python 3
print('Hello World')	Ausgabe Kommandozeile
print(1+2)	Ausgabe Zahlenoperationen
input("Eingeben:")	Eingabe
quit()	Pythonkonsole verlassen
exit()	Pythonkonsole verlassen
"String" **'String'**	Hochkommata doppelt oder einfach (bevorzugt)
\n	New Line; Escape Character
\"	Backslash für verbotene Buchstaben (z.B. Hochkomma)
""" Drei """	Drei Hochkommata = Automatischer Zeilenumbruch
"String" + 'cool'	String Konkatenation
print("spam" * 3)	Strin Multiplikation
int("4")	Typkonversation (selbiges mit **float()** und **str()**)

Nach der Installation Zugriff über Kommandozeile, der Python GUI oder IDLE (Integrierte Entwicklungsumgebung).
Die Python Konsole is eine **REPL** (Read-eval-print loop)

Rechenoperationen

2 [+/-/*/:] 2	Rechenoperationen
2* (3+4)	Klammern
10.56	Float: Fließkommazal
4 / 2 = 2.0	Division von Integer ergibt Float
2 ** 5	Exponentation 2 hoch 5
9 ** (1/2)	Wurzel
20 // 6 = 3	Ganzzahldivision
1.25 & 0.5	Modulo

Rechenoperationen können im Normalfall entsprechend der matemathischen Notation angegeben werden. Division mit der 0 produziert einen fehler

[B eispiel]Typenkonversation

```
>>> float(input("Enter a number: ")) + float(input("Enter another number: "))
Enter a number: 40
Enter another number: 2
42.0
```

Eingabenumwandlung in Fließkommazahlen mit anschließender Rechnung.

Variablen

x = 7	Variablendeklaration
del variable	Variable löschen
x += 3	Werterhöhung, ebenso bei Strings, mit vielen Operationen möglich

Variablen haben keinen spezifischen Datentyp. Erlaubte Zeichen sind Buchstaben, Nummern und Unterstriche. Variablen dürfen nicht mit Zahlen beginnen.

Kontrollstrukturen

b = **True|False**	Boolean, groß geschrieben
==	Gleich
!=	Ungleich
< | >	Größer / kleiner
<= | >=	Kleiner gleich, größer gleich
and	Logisches UND
or	Logisches ODER
not 1==1	Logisches NICHT
if Bedingung: statements	If-Abfrage
else: statements	Else Statement
elif Bedingung: statements	else-if Abfrage
while Bedingung: Statement	While-Schleife
while 1==1: Statement	Endlosschleife
break	Beenden Schleife
**continue	Schleife stoppen, Sprung nächste Iteration

Python benutzt Lerrzeichen zu Beginn einer Zeile, um die Zugehörigkeit zu einem Codesegment zu markieren (Beispiel if-Abfrage).

Cheatography

Python 3 Cheat Sheet

by aragow via cheatography.com/24054/cs/11425/

Listen / Arrays

arr = ["a","b","c"]	Array/Liste
print(**arr[0]**)	Ausgabe Listenelement
arr = ["a",5,[3,3.0]]	Mehrere Datentypen und Listen
st = "Hello" print(**st[2]**)	String als Array, Ausgabe entsprechend möglich
nums[2] = 5	Neuzuordnung Feld 2
nums = [1, 2, 3] print(**nums + [4, 5, 6]**)	Addition von Arrays, Multiplikation möglich
print("egg" **in** words)	Prüft Existenz vom String im Array *words*
print(**not 4 in** nums) print(**4 not in** nums)	Prüfung Element nicht in Liste

Beim Array werden die Elemente durch Kommas separiert. Es ist möglich, bei der Definition ein Komma an letzter Stelle einzufügen, was kein zusätzliches Element einfügt, aber syntaktisch valide ist. Einzelne Felder können neu zugeordnet werden.

Boolean Operatoren

Operator	Description
**	Exponentiation (raise to the power)
~ + -	Complement, unary plus and minus (method names for the last two are +@ and -@)
* / % //	Multiply, divide, modulo and floor division
+ -	Addition and subtraction
>> <<	Right and left bitwise shift
&	Bitwise 'AND'
^ |	Bitwise exclusive `OR' and regular `OR'
<= < > >=	Comparison operators
<> == !=	Equality operators
= %= /= //= -= += *= **=	Assignment operators
is is not	Identity operators
in not in	Membership operators
not or and	Logical operators

[Beispiel] If & While

```
i = 0
while True:
    i = i +1
    if i == 2:
        print("Skipping 2")
        continue
    if i == 5:
        print("Breaking")
        break
    print(i)
print("Finished")
```