Cheatography

Python3 Cheat Sheet

by sale_f via cheatography.com/82287/cs/19620/

Objekttypen

Integer (int)	Ganzzahl
Float	Gleitkommazahl
String (str)	Zeichenkette
Bool	TRUE or FALSE
Liste	[a,b,c,d]
Tupel	'a','b','c'
Menge (set)	{'Java,'Perl','Python'}
Dictionary	{'Birnen':2, 'Bier':1}

Arithmetische Operatoren

+	Addition
-	Subtraktion
*	Multiplikation
/	Division
%	Modulus (Rest nach Division)
**	Exponent

Verketten von Zeichenkette

+	Verkettung beider Zeichenketten
,	Verkettung gemischter Objekte

Vergleichsoperatoren

A == b	Prüft, ob Werte A und b gleich sind
A != b	Prüft, ob Werte a und b ungleich sind
A > b	Prüft, ob Wert A größer als Wert b ist
A < b	Prüft, ob Wert A kleiner als Wert b ist
A >= b	Prüft, ob Wert a größer oder gleich Wert b ist
A <= b	Prüft, ob Wert a kleiner oder gleich Wert b ist
A is b	Prüft, ob a und b auf das selbe Objekt verweisen
Aussage wahr	TRUE

Logische Verknüpfungen

and	und
or	oder
xor	entweder-oder
not	nicht

Slicing

[a:b:c]	Zugriff auf bestimmten Teil einer Zeichenkette
a	untere Grenze
b	obere Grenze
c	Schrittweite

c nur angeben, wenn bestimmte Schritte gegangen werden sollen

Listen

Element = list[2]	Element mit Index 2 aus list
list.append(-Objekt)	fügt neues Objekt in Liste hinzu
del(list[Ind-exnr.])	löscht Objekt an [Indexnr]
len(list)	Länge der Liste
list.reverse()	Liste wird umgedreht
list.sort()	Liste wird sortiert
"Zeichen".j-oin(list)	Listenelemente als Zeichenkette verbunden über Zeichen-Teilstring

Tupel

Listen die nicht veränderbar sind!

Dictionaries

dic =	(Key: Value, Key: Value)
element = dic[key]	Ausgabe Value vom Key
dic[Key] = string	Aktualisiert dic am Position Key
dic[Key] = string	Fügt neue Werte ein
dic.update(dic2)	Verbindet Dictionaries miteinander
len(dic)	Anzahl der vorhandenen Key im Dic.
dic.clear()	Löscht Inhalt eines Dictionary
list(dic.keys())	Liste mit allen Schlüsseln aus dem Dic.
list(dic.values())	Liste mit allen Values aus den Dic.
list(dic.items())	Liste aller Key-Value-Paare aus den Dic.
var = dic[key]	Zugriff auf Wert von key in dic

Mengen

set(list or string)	Umwandlung von Listen oder Strings in Mengen
	set-Objekt --> einmalige & unveränderbare Elemente aus ursprünglichen Liste
menge.add(TEXT)	Fügt Element zur Menge hinzu
menge.clear()	Leert die Menge
menge.copy()	flache Kopie einer Menge
len(menge)	Anzahl der Elemente in einer Menge
menge.differenc-e(string)	Ausgabe der Differenz einer Menge
menge.differenc-e_update(string)	Aktualisiert ursprüngliche Menge direkt
menge.discard(e-lement)	Entfernt Element aus einer Menge
menge.union(-menge2)	Bestimmung der Vereinigungsmenge zweier Mengen
menge.intersect-ion(menge2)	Bestimmung Schnittmenge zweier Mengen
menge.isdisjoin-t(menge2)	Ermittelt ob 2 Mengen disjunkt sind (Schni-tmenge = leer)
menge.issubset(-menge2)	Falls Menge eine Teilmenge einer anderen Menge ist
menge.issuperse-t(menge2)	Falls Menge, Obermenge einer anderen Menge ist
menge.pop()	Beliebiges Element einer nicht-leeren Menge zurückgeliefert und entfernt

Zeichenketten auftrennen

split(was eingesetzt)	Zeichenkette an einem Substring aufgeteilt

Zufallszahlen

random.random()	Float r mit 0<=r<1
random.randrange(start,s-top,step))	Integer r mit start<=r<stop
random.randint(start,stop)	Integer r mit start <=r<stop
random.sample (Anzahl Zeichen)	Liste mit 4 zufällig gewählten Zeichen
random.uniform(start, stop)	Float r mit start <=r<=stop

import random

Ausgabe in Groß-und Kleinbuchstaben

string.upper()	Alles in Großbuchstaben
string.lower()	Alles in Kleinbuchstaben
string.capitalize()	1. Buchstabe groß geschrieben

Länge einer Zeichenkette

len(string)	Länger einer Zeichenkette als int-Objekt

Suche nach einem Substring

string.find(STR-ING,START,END)	Index des 1. Auftretens STRING in string (von links ausgehend)
string.rfind(STRIN-G,START,END)	Index vom letzten Auftreten von STRING in string (von rechts ausgehend)

Index = -1 --> Falls STRING nicht in string

Teilstring-Anzahl

string.count(STRING,START,END)	Anzahl von STRING in string

Index END nicht enthalten

Ersetzung in einer Zeichenkette

string.replace(REP,NE-W,MAX)	Teilstring REP durch Teilstring NEW ersetzt

Cheatography

Python3 Cheat Sheet

by sale_f via cheatography.com/82287/cs/19620/

Übersetzen von Zeichen in einer Tabelle

string.maketrans(IN,OUT)	Erstellt Übersetzungstablle
translate(string)	Übersetzt erstellte Tabelle

Abfrage String bzgl. bestimmter Kriterien

Isalnum(string)	mind. 1 Zeichen, alle alphanummerisch
Isalpha(string)	mind. 1 Zeichen, alle alphabetisch
Isdigit(string)	Enthält nur Zahlen
Isnumeric(string)	Unicode-String enthält nur numerische Zahlen
Isdecimal(string)	Unicode-String enthält nur Dezimalzahl

Ausgabe: TRUE or FALSE

Aussagenlogik (Regeln log. Verknüpfungen)

Assoziativgesetze	(a and b) and c = a and (b and c)
	(a or b) or c = a or (b or c)
Kommutativgesetze:	A and b = b and a
	A or b = b or a
Distributivgesetze:	A or (b and c) = (a or b) and (a or c)
	A and (b or c) = (a and b) or (a and c)
Absorptionsgesetze:	A or (a and b) = a
	A and (a or b) = a

Reguläre Ausdrücke

re.search(MUSTER,STRING)	Sucht MUSTER im STRING
var.group()	Zugriff auf einzelne Teile bei einem Match
re.finditer(MUSTER, STRING)	Sucht MUSTER im STRING, falls es mehrmals vorkommt
re.sub(MUSTER,ERSETZUNG,STRING))	Ersetzt MUSTER durch ERSETZUNG im STRING

import re

Metazeichen

[abcdef]	Ein Zeichen, aus angegebener Menge
[^abcdef]	Ein Zeichen, nicht aus angegebener Menge
aa | bb	Es muss aa oder bb vorkommen
$	Ende der Zeichenkette
^	Anfang der Zeichenkette

Zeichenklassen

[0-9] (Zeichenmenge) \d (Kurzform) \D (Gegenteil) --> Ziffer
[a-zA-Z0-9] \w \W --> Wortzeichen
[\t\n\r\f] \s \S --> Whitespace

Quantoren

*	Beliebig oft, darf auch fehlen
+	Mindestens einmal
?	Einmal oder keinmal
{ZAHL}	Muss genau ZAHL-mal vorkommen
{ZAHL,}	Mindestens ZAHL-mal
{,ZAHL}	Maximal ZAHL-mal
{ZAHL1,ZAHL2}	Mind. Zahl1-mal, Max. ZAHL2-mal

Modifizierer

re.DEBUG	DEBUG-Info ausgeben
re.I	Groß-& Kleinschreibung ignorieren
re.M	Multi-Zeilen-Treffer
re.S	. trifft auch Zeilenumbruch, ansonsten nicht

bei re.search (MUSTER,STRING,MOD) als MOD eingeben

Datein einlesen

with open ("datei.txt", "r") as fin:
--> for zeile in datei.txt:
--> ---> zeile = zeile.rstrip()

Dateien zeilenweise einlesen und Zeilenumbrüche entfernen

Dateien schreiben

with open ("datei.txt", "w") as fout:
--> print ("Hallo Welt", file = fout)
-->Hallo Welt