

# Python 3 Beginner's Reference Cheat Sheet

Alvaro Sebastian  
<http://www.sixthresearcher.com>

## Main data types

**boolean** = *True / False*  
**integer** = 10  
**float** = 10.01  
**string** = "123abc"  
**list** = [ value1, value2, ... ]  
**dictionary** = { key1:value1, key2:value2, ... }

## Numeric operators

+	addition
-	subtraction
*	multiplication
/	division
**	exponent
%	modulus
//	floor division

## Comparison operators

==	equal
!=	different
>	higher
<	lower
>=	higher or equal
<=	lower or equal

## Boolean operators

<b>and</b>	logical AND
<b>or</b>	logical OR
<b>not</b>	logical NOT

## Special characters

<b>#</b>	comment
<b>\n</b>	new line
<b>\&lt;char&gt;</b>	escape char

## String operations

<b>string[i]</b>	retrieves character at position i
<b>string[-1]</b>	retrieves last character
<b>string[i:j]</b>	retrieves characters in range i to j

## List operations

<b>list = []</b>	defines an empty list
<b>list[i] = x</b>	stores x with index i
<b>list[i]</b>	retrieves the item with index i
<b>list[-1]</b>	retrieves last item
<b>list[i:j]</b>	retrieves items in the range i to j
<b>del list[i]</b>	removes the item with index i

## Dictionary operations

<b>dict = {}</b>	defines an empty dictionary
<b>dict[k] = x</b>	stores x associated to key k
<b>dict[k]</b>	retrieves the item with key k
<b>del dict[k]</b>	removes the item with key k

## String methods

<b>string.upper()</b>	converts to uppercase
<b>string.lower()</b>	converts to lowercase
<b>string.count(x)</b>	counts how many times x appears
<b>string.find(x)</b>	position of the x first occurrence
<b>string.replace(x,y)</b>	replaces x for y
<b>string.strip(x)</b>	returns a list of values delimited by x
<b>string.join(L)</b>	returns a string with L values joined by string
<b>string.format(x)</b>	returns a string that includes formatted x

## List methods

<b>list.append(x)</b>	adds x to the end of the list
<b>list.extend(L)</b>	appends L to the end of the list
<b>list.insert(i,x)</b>	inserts x at i position
<b>list.remove(x)</b>	removes the first list item whose value is x
<b>list.pop(i)</b>	removes the item at position i and returns its value
<b>list.clear()</b>	removes all items from the list
<b>list.index(x)</b>	returns a list of values delimited by x
<b>list.count(x)</b>	returns a string with list values joined by S
<b>list.sort()</b>	sorts list items
<b>list.reverse()</b>	reverses list elements
<b>list.copy()</b>	returns a copy of the list

## Dictionary methods

<b>dict.keys()</b>	returns a list of keys
<b>dict.values()</b>	returns a list of values
<b>dict.items()</b>	returns a list of pairs (key,value)
<b>dict.get(k)</b>	returns the value associated to the key k
<b>dict.pop()</b>	removes the item associated to the key and returns its value
<b>dict.update(D)</b>	adds keys-values (D) to dictionary
<b>dict.clear()</b>	removes all keys-values from the dictionary
<b>dict.copy()</b>	returns a copy of the dictionary

**Legend:** x,y stand for any kind of data values, s for a string, n for a number, L for a list where i,j are list indexes, D stands for a dictionary and k is a dictionary key.

# Python 3 Beginner's Reference Cheat Sheet

Alvaro Sebastian  
<http://www.sixthresearcher.com>

## Built-in functions

<b>print(x, sep='y')</b>	prints x objects separated by y
<b>input(s)</b>	prints s and waits for an input that will be returned
<b>len(x)</b>	returns the length of x (s, L or D)
<b>min(L)</b>	returns the minimum value in L
<b>max(L)</b>	returns the maximum value in L
<b>sum(L)</b>	returns the sum of the values in L
<b>range(n1,n2,n)</b>	returns a sequence of numbers from n1 to n2 in steps of n
<b>abs(n)</b>	returns the absolute value of n
<b>round(n1,n)</b>	returns the n1 number rounded to n digits
<b>type(x)</b>	returns the type of x (string, float, list, dict ...)
<b>str(x)</b>	converts x to string
<b>list(x)</b>	converts x to a list
<b>int(x)</b>	converts x to a integer number
<b>float(x)</b>	converts x to a float number
<b>help(s)</b>	prints help about x
<b>map(function, L)</b>	Applies function to values in L

## Conditional statements

```
if <condition> :  
    <code>  
else if <condition> :  
    <code>  
...  
else:  
    <code>  
  
if <value> in <list>:
```

## Data validation

```
try:  
    <code>  
except <error>:  
    <code>  
else:  
    <code>
```

## Working with files and folders

```
import os  
os.getcwd()  
os.makedirs(<path>)  
os.chdir(<path>)  
os.listdir(<path>)
```

## Loops

```
while <condition>:  
    <code>  
  
for <variable> in <list>:  
    <code>  
  
for <variable> in  
range(start,stop,step):  
    <code>  
  
for key, value in  
dict.items():  
    <code>
```

## Loop control statements

<b>break</b>	finishes loop execution
<b>continue</b>	jumps to next iteration
<b>pass</b>	does nothing

## Running external programs

```
import os  
os.system(<command>)
```

## Functions

```
def function(<params>):  
    <code>  
    return <data>
```

## Modules

```
import module  
module.function()  
  
from module import *  
function()
```

## Reading and writing files

```
f = open(<path>,'r')  
f.read(<size>)  
f.readline(<size>)  
f.close()  
  
f = open(<path>,'r')  
for line in f:  
    <code>  
f.close()  
  
f = open(<path>,'w')  
f.write(<str>)  
f.close()
```

**Legend:** x,y stand for any kind of data values, s for a string, n for a number, L for a list where i,j are list indexes, D stands for a dictionary and k is a dictionary key.