Python 1 Cheatsheet

Operators

Symbol	What it does	Example
+	addition	5 + 5
-	subtraction	5 - 5
*	multiplication	5 * 5
/	division	5 / 5
**	exponent	5**5 (5 to the fifth power)
%	modulus (remainder)	5 % 2 (results in 3)
==, is	Equals	`5 == 5` results in True `9 is 9` results in True `5 is 7` results in False
!=, is not	Not equals	`5 != 5` results in False `5 is not 7` results in True
>	Greater than	`5 > 6` results in False `11 > 6.23` results in True
>=	Greater than or equal to	`5 >= 4` results in True
<	Less than	`4 < 5` results in True
<=	Less than or equal to	`4 <= 5` results in True

Variable types

Variable Type	Description	Casting	Examples
integer	whole number	int()	5, -11, 0
float	decimal number	float()	9.57, -0.2, 110.24, 4.
string	ordered, immutable character container	str()	"word", "lots of words", "words and numbers 582 29 in quotes"
list	ordered, mutable container	list()	[1,2,3,4] ; ["hi", "bye", 9, -22.1]
dictionary	unordered, mutable container (associative array)	dict()	{"key1":"value1", 9: "three-squared"}
tuple	ordered, immutable container	tuple()	(4, 9) ; ("word1", "word2", "word3")

Useful string methods

Method	Description	Example
`.upper()`	converts to upper case	hi = "my string" `hi.upper()` returns "MY STRING"
`.lower()`	converts to lower case	hi = "My String" `hi.lower()` returns "my string"
`.split()`	split a string on a value into a list	hi = "comma,separated,values,in,the,string" `hi.split(",")` returns ['comma', 'separated', 'values', 'in', 'the', 'string']
`.strip()`	removes leading/trailing whitespace/value	hi = "my string" `hi.strip()` returns "my string" (there was no leading/trailing whitespace!) `hi.strip("g")` returns "my strin"
`.count()`	count instances of a character in a string	hi = "my letterful string" `hi.count("t")` returns 3
`.replace()`	replace all instances of a value	hi = "silliness" `hi.replace("s", "5")` returns "5illine55"

Python 1 Cheatsheet

Useful list methods

Method	Description	Example
`.append()`	Add value to the end of a list	`my_list.append(5)`
`.insert()`	Add value to a specific index in a list	`my_list.insert(5, "index #5 will be this string")`
`.remove()`	Remove all occurrences of a particular value from a list	`my_list.remove(5)`
`.index()`	Determine the index of a particular list value	`my_list.index(5)`

Useful dictionary methods

Method	Description	Example
`.keys()`	Return a list of all keys in a dictionary	`my_dict.keys()`
`.values()`	Return a list of all values in a dictionary	`my_dict.values()`
`.items()`	Return a list of (key,value) tuples from a dictionary	`my_dict.items()`