Cheatography

Python_Chap_4 Cheat Sheet

by Theo666 via cheatography.com/193738/cs/40331/

Lists

A list is a data structure that contains a series of values. Python allows the construction of a list containing values of different types. It's iterable, indexable, ordered, mutable and not hashable.

combination of lists and dictionaries

[[...], [...], [...], ...]	list of list
list_of_list[...][...]	index elements in a list of list
dict{key : [..., ...]}	values of a dictionary can be a list
[dict1, dict2]	list of dictionary

But the keys of dictionary are hasables, not ordered and not duplicated.
list[:] is a new list duplicating the original, so list[:][0] will return the first element of the list

Operation in the lists

list + list	add lists
list * int	multiply the list

- We cannont substract directly a list from another by using ' - ' . [i for i in list1 if i not in list2]; set(list1) - set(list2)
- To repeat each element of lists, [i for i in list for _ in range(int)]
- list_1 += list_2 equal to list_1 = list_1 + list_2

Examples

```
[i for i in range(10)]
[i for i in range(31) if i % 2 == 0]
[[m.upper(), len(m)] for m in msg_lst]
[seq[i :i+ width] for i in range(0, len(seq), width)]
```

Common Functions

`list[start:stop :step]`	list sciling (tranche)
`enumerate (list)`	return positions and items
`list.index(item)`	return the position of the item
`list.count(item)`	returns the number of times that element appears in the list
`list(string)`	convert a string to a list one by one charcter
`string.split (sep)`	convert a string to a list with a separator
`'sep'.join(list)`	convert a list to a string
`len(list)`	length of lists
`max(list)`	find the maximun
`min(list)`	find the minimun
`sum(list)`	calculate the sum
`list.sort (reverse=)`	sort the elements of a list in-place. reverse False from smallest to largest values; alphabetic order possible
`sorted (list, reverse= )`	create a new sorted list without modifying the original list; alphabetic order possible
`list.reverse()`	reverse the elements of a list in-place
`reversed( list)`	create a new reversed list without modifying the original list
`list.append( item)`	add an element to the end of lists
`list.insert( item, pos)`	insert an element at a position of lists
`list.remove( item)`	remove an item from lists; remove only one first element.
`list.pop()`	remove and return the last element
`del list[pos]`	remove the item by its position idenx
`range( start, stop,step)`	similar to lists, but immutable. stop at n-1
`set(list)`	remove the duplicated elements

- `list[start:stop :step]` step 1 by default; stop at n-1 even if negative index
- `list[:]` create a new list. lst2 = lst1 creates a reference to the original list with the same ID
- `'sep'.join(list)` cannot combine a list containing only number (int & float). [str(i) for i in list]
- `list.remove( item)` If there're duplicated elements, it remove only the first element
- `range( start, stop,step)` stop could be higher than start with a negative step