Sets

Sets are like dictionaries without values.

Creating sets

* {1, 2, 3, 4}	Set of elements	fruit = {'apple', 'pear'}
* set([1, 2, 3])	Set from a list (Or other iterable object)	unique_numbers = set([3, 1, 3]) letters = set('chocolate')
* set()	Empty set – {} is dictionary	
* frozenset(...)	Set that you can't change	

Basic operations

len(x)	Number of elements
k in x	Is k element in the set?
* x.copy()	Copy of the set
x.clear()	Empty the set

Elements

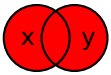
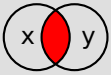
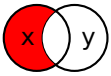
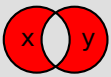
x.add(k)	Add element k
x.remove(k)	Remove element (KeyError if k is not in the set)
x.discard(k)	Remove k element only if it's in the set
x.pop()	It will remove and show any element

Subset

x.isdisjoint(y)	True if sets are disjoint? (tj. they don't have any element in common)
x.issubset(y) x <= y	Is x subset of y?
x < y	Only subset (x!=y)
x.issuperset(y) x >= y	Is x superset of y?
x > y	Only superset (x!=y)

Operations on sets

Only commands with update will change x, others return result

* x.union(y) * x | y x.update(y)	 Union
* x.intersection(y) * x & y x.intersection_update(y)	 Intersection
* x.difference(y) * x - y x.difference_update(y)	 Difference
* x.symmetric_difference(y) * x ^ y x.symmetric_difference_update(y)	 Symetric difference

More info: https://docs.python.org/3/library/stdtypes.html#set-types-set-frozenset

* Function with this sign will create new set