Sets

- A Set is unordered, unindexed, with no duplicate elements that can be of different types
- A Set is created with curly braces

```
set1 = {5, 3, 9}
```

- A Set can be created with a constructor

```
newset = set(("antonio", "luisa“))
```

- Number of elements in the Set

```
len(set1)
```

- Delete a Set

```
del set1
```

- Check if an element exists in a Set

```
if 9 in set1 :
    print(“9 is in the set“)
```

- The elements of a Set can be accessed with a for loop

```
for value in set1:
    print(value)
```

Sets

- Set built-in methods:

set1.add(value)	Adds an element to the set
set1.clear()	Clears all the elements from the set
set1.copy()	Returns a copy of the set
set1.difference(set2)	Returns a set with the elements that exist in set1 but not in set2, …
set1.difference_update(set2)	Removes from set1 the elements that exist in set2, …
set1.discard(value)	Removes the value from the set
set1.intersection(set2)	Returns a set with the elements that exist in set2, …
set1.intersection_update(set2)	Removes the elements in set1 that are not present in set2, …
set1.isdisjoint(set2)	Returns whether the two sets have common elements
set1.issubset(set2)	Returns whether set2 contains set1
set1.issuperset(set2)	Returns whether set1 contains set2
set1.pop()	Removes and returns a random element from the set
set1.remove(value)	Removes the value from the set
set1.union(set2)	Returns a set containing the elements from both sets
set1.update(set2)	Updates set1 with the elements from set2