Set Methods

|  |  |  |  |
| --- | --- | --- | --- |
| SNO | Method Name | Usage | Operator |
| [add()](https://www.w3schools.com/python/ref_set_add.asp) | Adds an element to the set | setobject.add(element) |  |
| [clear()](https://www.w3schools.com/python/ref_set_clear.asp) | Removes all the elements from the set | setobject.clear() |  |
| [copy()](https://www.w3schools.com/python/ref_set_copy.asp) | Returns a copy of the set | Setobject2=setobject.copy() |  |
| [difference()](https://www.w3schools.com/python/ref_set_difference.asp) | Returns a set containing the difference between two or more sets | Result\_Set=setObject1.difference(setObject2) | R=s1-s2 |
| [difference\_update()](https://www.w3schools.com/python/ref_set_difference_update.asp) | Removes the items in this set that are also included in another, specified set | setObject1.difference\_update(setObject2) | s1-s2 |
| [discard()](https://www.w3schools.com/python/ref_set_discard.asp) | Remove the specified item |  |  |
| Union() |  |  | | |
| [intersection()](https://www.w3schools.com/python/ref_set_intersection.asp) | Returns a set, that is the intersection of two other sets | Result\_Set=setObject1.intersection(setObject2) | R=s1&s2 |
| [intersection\_update()](https://www.w3schools.com/python/ref_set_intersection_update.asp) | Removes the items in this set that are not present in other, specified set(s) | setObject1.intersection\_update(setObject2) |  |
| [symmetric\_difference()](https://www.w3schools.com/python/ref_set_symmetric_difference.asp) | Returns a set with the symmetric differences of two sets | z = x.symmetric\_difference(y) | R=S1^S2 |
| [isdisjoint()](https://www.w3schools.com/python/ref_set_isdisjoint.asp) | Returns whether two sets have a intersection or not | Result= setObject1.isdisjoint(setObject2)  Returns True or False |  |
| [issubset()](https://www.w3schools.com/python/ref_set_issubset.asp) | Returns whether another set contains this set or not | Result= setObject1.issubset(setObject2)  Returns True or False | R=s1<s2 |
| [issuperset()](https://www.w3schools.com/python/ref_set_issuperset.asp) | Returns whether this set contains another set or not | Result= setObject1.issuperset(setObject2)  Returns True or False | R=S1>S2 |
|  |  |  |  |
| [pop()](https://www.w3schools.com/python/ref_set_pop.asp) | Removes an element from the set | Setobject.pop() |  |
| [remove()](https://www.w3schools.com/python/ref_set_remove.asp) | Removes the specified element | setObject.remove(element) |  |
| [symmetric\_difference()](https://www.w3schools.com/python/ref_set_symmetric_difference.asp) | Returns a set that contains all items from both sets, but not the items that are present in both sets. |  |  |
| [symmetric\_difference\_update()](https://www.w3schools.com/python/ref_set_symmetric_difference_update.asp) | inserts the symmetric differences from this set and another |  |  |
| [union()](https://www.w3schools.com/python/ref_set_union.asp) | Return a set containing the union of sets  Setobject3 =Setobject1.union(setobject2) |  |  |
| [update()](https://www.w3schools.com/python/ref_set_update.asp) | Update the set with another set, or any other iterable |  |  |