<1T>

# s.isdisjoint(other)

Return True if the set has no elements common with set other. Sets are disjoint if and only if their intersection is the empty set.

# s.issubset(other) [alias: s <= other]

Test whether every element in set s is in other.

# s < other

Test whether the set s if a proper subset of other, that is, s <= other and s != other

# s.issuperset(other) [alias: s >= other]

Test whether every element in other is in the set.

# s > other

Test whether the set s is a proper subset of other, that is s >= other and s != other

# s.union(\*others) [alias: s | other | …]

Return a new set with elements from the set and all others.

# s.intersection(\*others) [alias: s & other & …]

Return a new set with elements from the set and all others.

# s.difference(\*others) [alias: s – other – …]

Return a new set with elements in the set that are not in the others

# s.symmetric\_difference(other) [alias: s^other]

Return a new set with elements in either the set or the other but not both