**set**

In Python, sets are unordered collections of unique elements. They are mutable, meaning you can add or remove elements from them. No duplicate element or value accept. Set is mutable but sets element are immutable.set don’t have any indexing concept

Once set is created you cannot change its items but you can add new itemsand remove items

a.Here set can have tuple but no list and set inside set

>>>s={"sun","mon","tus",(12,3)}

>>> print(s)

{'tus', 'sun', 'mon', (12, 3)}

b. set can created by using constructor and range function

s=set(‘abcd’) s1=set(range(1,5))

s s

{‘a’,’b’,’c’,’d’} {1,2,3,4}

**Set traversing :**

For vaiable in set:

Print (variable)

e.g. s={1,2,3,45,6,7,8,9}

for no in s:

print(no)

**Set Operations:**

1. Union (|): Returns a set containing all unique elements from both sets.
2. Intersection (&): Returns a set containing elements that are present in both sets.
3. Difference (-): Returns a set containing elements that are present in the first set but not in the second.
4. Symmetric Difference (^): Returns a set containing elements that are present in either of the sets, but not in both.
5. Subset (<=): Returns True if all elements of one set are present in the other set.
6. Superset (>=): Returns True if all elements of one set are present in the other set.
7. Disjoint (isdisjoint()): Returns True if two sets have no common elements.

**Set Functions:**

1. len(set): Returns the number of elements in the set.
2. max(set): Returns the maximum element in the set.
3. min(set): Returns the minimum element in the set.
4. sum(set): Returns the sum of all elements in the set (only if all elements are numeric).
5. sorted(set): Returns a new sorted list from the elements of the set.
6. any(set): Returns True if any element of the set is true. If the set is empty, returns False.
7. all(set): Returns True if all elements of the set are true (or if the set is empty).

**Set Methods**:

1. add(element): Adds a single element to the set.
2. update(iterable): Adds multiple elements to the set.
3. remove(element): Removes a specific element from the set. Raises KeyError if the element is not present.
4. discard(element): Removes a specific element from the set if it is present.
5. pop(): Removes and returns an arbitrary element from the set.
6. clear(): Removes all elements from the set.
7. copy(): Returns a shallow copy of the set.
8. difference(): returns a new set containing the difference between two or more sets .
9. difference\_update(iterable): Removes all elements of another set from this set.
10. Intersection(): Returns a new set which is the intersection of two or more sets.
11. intersection\_update(iterable): Updates the set with the intersection of itself and another.
12. symmetric\_difference():
13. symmetric\_difference\_update(iterable): Updates the set with the symmetric difference of itself and another.
14. union\_update(iterable): Updates the set with the union of itself and another.
15. isdisjoint(other): Returns True if the set has no elements in common with other.
16. issubset(other): Returns True if all elements of the set are in other.
17. issuperset(other): Returns True if the set contains all elements of other.