Set in python

- It is an unordered collection of data of different data types.
- Set does not contain duplicate data.
- A set is created using curly brackets.

Create set

```
str_set={"Apple","Orange","Mango"}
int_set={15,25,36,84,59}
float_set={2.3,5.6,1.4,9.6}
mixed_set={"Easy",165,25.3}
```

Access values of set using loop

- We can't access items of set using index value because it is unordered collection of data.
- We cannot be sure in which order the items will appear because sets are unordered.
- For better understanding see the example.

```
fruit_set={"Apple","Orange","Mango"}
print("Fruit set :",fruit_set)
"""
***Output***
Run 1
Fruit set : {'Mango', 'Orange', 'Apple'}
Run 2
Fruit set : {'Apple', 'Mango', 'Orange'}
"""
```

Update item of set

- We can not change the value of set.

```
fruit_set={"Apple","Orange","Mango"}
#this line will generate error
#because we can't change the value of set
fruit_set[1]="Banana"
#TypeError: 'set' object does not support item assignment
```

Length of set

- len() function is used to get length of set.

```
fruit_set={"Apple","Orange","Mango"}
print("Length of set is ",len(fruit_set))
"""
***Output***
Length of set is  3
"""
```

Add items into set

- add() function is used to add new item in a set.

```
fruit_set={"Apple","Orange","Mango"}
print("Fruit Set:",fruit_set)
#this line will add
#Cherry at the end of set
fruit_set.add("Cherry")
print("Fruit Set:",fruit_set)
"""
***Output***
Fruit Set: {'Orange', 'Mango', 'Apple'}
Fruit Set: {'Orange', 'Mango', 'Cherry', 'Apple'}
"""
```

Add more than one item to a set

- update() function is used to add more than one item to a set.
- update() and add() function discard the duplicate elements.

```
fruit_set={"Apple","Orange","Mango"}
print("Fruit Set:",fruit_set)
#Add multiple items to a set
fruit_set.update(["Cherry","Banana","Apple"])
print("Fruit Set:",fruit_set)
"""
***Output***
Fruit Set: {'Mango', 'Orange', 'Apple'}
Fruit Set: {'Orange', 'Banana', 'Mango', 'Cherry', 'Apple'}
"""
```

Delete item from set using remove() function

- remove() function is used to remove specified item from a set.

```
fruit_set={"Apple","Orange","Mango"}
print("Fruit Set:",fruit_set)
#this line will remove Orange from set
fruit_set.remove("Orange")
print("Fruit Set:",fruit_set)
"""
***Output***
Fruit Set: {'Mango', 'Apple', 'Orange'}
Fruit Set: {'Mango', 'Apple'}
"""
```

Delete item from set using discard() function

- discard() function is also used to remove specified item from a set.

```
fruit_set={"Apple","Orange","Mango"}
print("Fruit Set:",fruit_set)
#this line will remove Orange from set
fruit_set.discard("Orange")
print("Fruit Set:",fruit_set)
"""
***Output***
Fruit Set: {'Mango', 'Apple', 'Orange'}
Fruit Set: {'Mango', 'Apple'}
"""
```

The difference between remove() and discard() function is that if the specified element does not exist in the set then remove() function will raise an error but discard() function will not raise an error.

Join two sets

- We can join two set using union() function.

```
set1={"Apple","Orange","Mango"}
set2={"Cherry","Grapes","Melon"}
#this line will join set1 and set2
set3=set1.union(set2)
print("set3 items")
print(set3)
"""
***Output***
set3 items
{'Orange', 'Grapes', 'Cherry', 'Mango', 'Apple', 'Melon'}
"""
```

Program to search particular element in set

```
fruit_set = {"Apple", "Orange", "Mango"}
str=input("Enter any string to search:")
if str in fruit_set:
  print(str," is found")
else:
  print("Not found")
"""
***Output***
Enter any string to search:Apple
Apple  is found
"""
```

Clear set

- clear() function is used to clear or empty the set.

```
fruit_set={"Apple","Orange","Mango"}
print("Before clear")
print(fruit_set)
#this line will empty the list
fruit_set.clear()
print("After clear")
print(fruit_set)
"""
***Output***
Before clear
{'Apple', 'Orange', 'Mango'}
After clear
set()
"""
```

Delete set

- del keyword is also used to delete set completely.

```
fruit_set={"Apple","Orange","Mango"}
print("Set Items")
print(fruit_set)
#this line will delete  set
del fruit_set
print("Deleted successfully")
"""
***Output***
Set Items
{'Apple', 'Orange', 'Mango'}
Deleted successfully
"""
```

Some commom Operation on a set

```
Set_A={4,5,6,9}
Set_B={1,2,5,6,8}
#This operation will return
#common elements in both set
print("AND or Intersection Operation:",Set_A & Set_B)
#This operation will combine
#elements of both set and discard duplicate elements
print("OR or Union Operation:",Set_A | Set_B)
#this operation will return Set_A elements
#that does not exist in the Set_B
print("Set difference:",Set_A-Set_B)
"""
***Output***
AND or Intersection Operation: {5, 6}
OR or Union Operation: {1, 2, 4, 5, 6, 8, 9}
Set difference: {9, 4}
"""
```