In [ ]:

```
Create a list of 8 elemenrs in which two are duplicates and convert that list into a set
Remove the element 3 from the set and check whetehr the element 3 and 4 is present or no
now Iterate over eac element of the set reate a setch that each element is equal to the
create set using above two sets
get the length of that set , call it l1
now Iterate over eac element of the set and create a list such that each element is equa
now create set using the list and set above
get the length of this set  and call it l2
check whether if l1=l2
Create another set and find the len, min, max, mean and count
Try to sort that set
clear all the elements from the set
create two sets of random numbers and find the union, intersection and difference of the
compare those two sets (commonaltiy)
Create a list and convert it into a set, now check whether an element is in the set
read about frozen set and create it
create an empty set and list=[1,2,3] and load these elements into that set in reveres or
venn diagrams
```

In [3]:

```
#Create a list of 8 elemenrs in which two are duplicates and convert that list into
a_list=[1,2,3,4,5,6,7,8]
a_set=set([1,2,3,4,5,6,7,8])
print(a_set)
a_set.add(100)
print(a_set)

```

```
{1, 2, 3, 4, 5, 6, 7, 8}
{1, 2, 3, 4, 5, 6, 7, 8, 100}
```

In [17]:

```
#Remove the element 3 from the set and check whetehr the element 3 and 4 is present
a_set.discard(3)
print(a_set)
print(3 in a_set)
print(4 in a_set)



```

```
{2, 4, 5, 6, 7, 8, 100}
False
True
```

In [29]:

```
#Iterate over each element of the set create a set such that each element is equal t
def my_function(a_set):
    my_set=[]
    for i in a_set:
        my_set.append(i*i)
    return my_set

result=set(my_function({2,3,4,5}))
print(result)

```

```
{16, 9, 4, 25}
```

In [37]:

```
#create set using above two sets
a_set.update(result)
print(a_set)
l1=len(a_set)
print(l1)

```

```
{2, 4, 5, 6, 7, 8, 100, 9, 16, 25}
10
```

In [42]:

```
#Iterate over eac element of the set and create a list such that each element is equ
#now create set using the list and set above
#get the length of this set  and call it l2
#check whether if l1=l2
def my_function1(a_list):
    my_list=[]
    for i in a_list:
        my_list.append(i*i)
    return my_list

my_function1({6,7,8,9})



```

Out[42]:

```
[64, 81, 36, 49]
```

In [46]:

```
#now create set using the list and set above
my_list=[64, 81, 36, 49]
a_set.update(my_list)
print(a_set)
l2=len(a_set)
print(l2)

```

```
{64, 2, 4, 5, 6, 7, 8, 100, 9, 36, 16, 81, 49, 25}
14
```

In [49]:

```
#check whether if l1=l2
if l2 in range(l1):
    print("true")
else:
    print("false")

```

```
false
```

In [75]:

```
print(l1==l2)
```

```
False
```

In [58]:

```
#Create another set and find the len, min, max, mean and count
b_set={5,6,11,1,0}
print(len(b_set))
print(min(b_set))
print(max(b_set))
c=b_set.count(11)
print(c)
#print
```

```
5
0
11

---------------------------------------------------------------------------
-
AttributeError                            Traceback (most recent call las
t)
Cell In [58], line 6
      4 print(min(b_set))
      5 print(max(b_set))
----> 6 c=b_set.count(11)
      7 print(c)

AttributeError: 'set' object has no attribute 'count'
```

In [60]:

```
#Try to sort that set
print(sorted(b_set))
```

```
[0, 1, 5, 6, 11]
```

In [62]:

```
#clear all the elements from the set
b_set.clear()
print(b_set)
```

```
set()
```

In [87]:

```
#create two sets of random numbers and find the union, intersection and difference o



set1={11,39,52,32}
set2={10,20,15,32}
set3=set1.union(set2)
print(set3)
set4=set1.intersection(set2)
print(set4)
set5=set1.symmetric_difference(set2)
print(set5)
set6=set2.symmetric_difference(set1)
print(set6)
```

```
{32, 39, 10, 11, 15, 52, 20}
{32}
{39, 10, 11, 15, 20, 52}
{39, 10, 11, 15, 52, 20}
```

In [88]:

```
#compare those two sets (commonaltiy)
print(set5==set6)
```

```
True
```

In [72]:

```
#Create a list and convert it into a set, now check whether an element is in the set
list1=[21,35,49,72,91]
set1=set(list1)
print(set1)
print(49 in set1)

```

```
{35, 72, 49, 21, 91}
True
```

In [73]:

```
#read about frozen set and create it
A={10,20,21,21}
c=frozenset(A)
print(c)
```

```
frozenset({10, 20, 21})
```

In [86]:

```
#create an empty set and list=[1,2,3] and load these elements into that set in rever
#venn diagrams

list4=[1,2,3]
set2=set(list4)
set3=sorted(list4, reverse=True)
print(set3)
```

```
{1, 2, 3}
```

In [ ]:

```

```