In [1]:
```
s={}
s
```

Out[1]: {}

In [2]:
```
type(s)
```

Out[2]: dict

In [3]:
```
s=set()
```

In [4]:
```
type(s)
```

Out[4]: set

In [5]:
```
a={1,2,3,4}
b={3,4,5,6}
c={6,7,8,9}
```

In [6]:
```
a.add(5)
```

In [7]:
```
a
```

Out[7]: {1, 2, 3, 4, 5}

In [8]:
```
a.clear()
```

In [9]:
```
a={1,2,3,4}
```

In [10]:
```
a
```

Out[10]: {1, 2, 3, 4}

In [11]:
```
a.pop() #pop will reomve the random element we cannot choose which element to re
a
```

Out[11]: {2, 3, 4}

In [12]:
```
a.add(1)
a
```

Out[12]: {1, 2, 3, 4}

In [13]:
```
print(a)
print(b)
print(c)
```

```
{1, 2, 3, 4}
{3, 4, 5, 6}
{8, 9, 6, 7}
```

In [14]:
```
a.remove(4) # removing the fourth index element
```

In [15]:
```
a
```

Out[15]:
```
{1, 2, 3}
```

In [16]:
```
a1=a.copy()
a1
```

Out[16]:
```
{1, 2, 3}
```

In [17]:
```
a.union(b)  # union is adding of both sets by removing the duplicate
```

Out[17]:
```
{1, 2, 3, 4, 5, 6}
```

In [18]:
```
a.difference(b) # it prints all the elements in the first set by removing the co
```

Out[18]:
```
{1, 2}
```

In [19]:
```
a.symmetric_difference(b) #it prints the all elements in set a and b except the
```

Out[19]:
```
{1, 2, 4, 5, 6}
```

In [20]:
```
a.union(b,c)
```

Out[20]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9}
```

In [21]:
```
a.update(c) #adds and updates the elements without duplicate
```

In [22]:
```
a
```

Out[22]:
```
{1, 2, 3, 6, 7, 8, 9}
```

In [23]:
```
c
```

Out[23]:
```
{6, 7, 8, 9}
```

In [24]:
```
a.intersection(c)
```

Out[24]:
```
{6, 7, 8, 9}
```

In [25]:
```
a&b #intersection
```

Out[25]:
```
{3, 6}
```

In [26]:
```
a|b|c #union
```

Out[26]:
```
{1, 2, 3, 4, 5, 6, 7, 8, 9}
```

In [27]:
```
a-b #difference
```

Out[27]:
```
{1, 2, 7, 8, 9}
```

In [47]:
```
a2={1,2,3,4,5,6,7,8,9}
b2={5,6,7,8,9}
c2={20,30,40,50}
```

In [49]:
```
a2.issuperset(b2) # here every element present in first set is in second set wit
                  #thats y a2 is superset to b2
```

Out[49]: True

In [51]:
```
b2.issubset(a2) # here all elements present in b2 are in a2 but there are more e
                # thats y b2 is sub set to a2
```

Out[51]: True

In [55]:
```
c2.isdisjoint(a2) # there are no common elements between c2 and a2 thats why the
```

Out[55]: True

In [104…
```
type(s)
```

Out[104… set

In [106…
```
s={4,8,2,9,3,7} #set automatically print in sorted
s
```

Out[106… {2, 3, 4, 7, 8, 9}

In [114…
```
s
```

Out[114… {2, 3, 4, 7, 8, 9}

In [116…
```
s[5]=6 # sets are immutable
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[116], line 1
----> 1 s[5]=6

TypeError: 'set' object does not support item assignment
```

dict

In [60]:
```
d={} # dict is all about keys and values where keys are give more importance
type(d)
```

Out[60]: dict

In [62]:
```
d1={'name':'akhil','dob':2002,'address':'hyderabad'}
d1
```

Out[62]: {'name': 'akhil', 'dob': 2002, 'address': 'hyderabad'}

In [64]:
```
d1.clear()
```

In [66]:
```
len(d1)
```

Out[66]: 0

In [68]:
```
d1={'name':'akhil','dob':2002,'address':'hyderabad'}
d1
```

Out[68]: {'name': 'akhil', 'dob': 2002, 'address': 'hyderabad'}

In [70]:
```
d2=d1.copy()
```

In [72]:
```
d2
```

Out[72]: {'name': 'akhil', 'dob': 2002, 'address': 'hyderabad'}

In [76]:
```
d2.items()
```

Out[76]: dict_items([('name', 'akhil'), ('dob', 2002), ('address', 'hyderabad')])

In [78]:
```
d2.keys()
```

Out[78]: dict_keys(['name', 'dob', 'address'])

In [82]:
```
d2.values()
```

Out[82]: dict_values(['akhil', 2002, 'hyderabad'])

In [84]:
```
d2.pop('dob') # by using pop we can remove the selected key and value
```

Out[84]: 2002

In [86]:
```
d2
```

Out[86]: {'name': 'akhil', 'address': 'hyderabad'}

In [88]:
```
d2.update(d1) #update will add both dicts without duplicate
```

In [90]:
```
d2
```

Out[90]: {'name': 'akhil', 'address': 'hyderabad', 'dob': 2002}

In [92]:
```
d2['dob']=2001 # dict is mutable
d2
```

Out[92]: {'name': 'akhil', 'address': 'hyderabad', 'dob': 2001}

In [94]:
```
d2.get('name')
```

Out[94]: 'akhil'

In [110…
```
d2
```

Out[110… {'name': 'akhil', 'address': 'hyderabad', 'dob': 2001}

In [112…
```
d2.popitem() #pop item will remove the random element from the dict
d2
```

Out[112… {'name': 'akhil', 'address': 'hyderabad'}

In [118…
```
for i in d2: # for loop wit
    print(i)
```

```
name
address
```

In [122…

```
for i in d2:
    print (i,':',d2[i])
```

```
name : akhil
address : hyderabad
```

In [ ]: