Set

Subset

Superset

Disjoint

In [2]:
```
A={1,2,3,4,5,6,7,8,9}
B={3,4,5,6,7,8}
C={10,20,30,40}
print(A)
print(B)
print(C)
```

```
{1, 2, 3, 4, 5, 6, 7, 8, 9}
{3, 4, 5, 6, 7, 8}
{40, 10, 20, 30}
```

In [6]:
```
A.issubset(B) # A is not a Subset of B because, all the elements present in the
```

Out[6]:
```
False
```

In [8]:
```
B.issubset(A) # B is a subset because, all the elements of 'B' are present in 'A
```

Out[8]:
```
True
```

In [12]:
```
B.issuperset(A) # B is not a superset because, all the elements of 'A' are not p
```

Out[12]:
```
False
```

In [14]:
```
A.issuperset(B) # A is superset now. beCause all the elements of B are also pres
```

Out[14]:
```
True
```

In [16]:
```
A.isdisjoint(B) # If there are no common elements then it is said to be disjoint
```

Out[16]:
```
False
```

In [18]:
```
A.isdisjoint(C) # Here we have common elements.so, True
```

Out[18]:
```
True
```

In [ ]:
```
A.tab # It gives you all the functions i.e., add, clear, copy, difference, diffe
     # discard, intersection, intersection_update, isdisjoint, isuperset, issubs
    # remove, union, update, symmetric_difference, symmetric_difference_update
```

In [20]:
```
sum(A)
```

Out[20]:
```
45
```

In [22]:
```
max(A)
```

Out[22]:
```
9
```

In [24]:
```
min(A)
```

Out[24]:
```
1
```

In [27]:
```
len(A)
```

Out[27]:
```
9
```

Dictionary

In [32]:
```
mydict=dict() # Empty Dictionary
mydict
```

Out[32]:
```
{}
```

In [34]:
```
mydict={} # Empty Dictionary
mydict
```

Out[34]:
```
{}
```

In [36]:
```
mydict={1:'one',2:'two',3:'three'} # Dictionary with integer keys
mydict                             # Here 1=Key, One=Value
                                   # Keys cannot be duplicate. Always it should
                                   # Values can be duplicate.
```

Out[36]:
```
{1: 'one', 2: 'two', 3: 'three'}
```

In [40]:
```
mydict
```

Out[40]:
```
{1: 'one', 2: 'two', 3: 'three'}
```

In [45]:
```
mydict=dict({1: 'one', 2: 'two', 3: 'three'}) # Creating dictionary using dict()
mydict
```

Out[45]:
```
{1: 'one', 2: 'two', 3: 'three'}
```

In [47]:
```
mydict={'A':'one','B':'Two','C':'Three'} # Dictionary with character keys.
mydict
```

Out[47]:
```
{'A': 'one', 'B': 'Two', 'C': 'Three'}
```

In [49]:
```
mydict={'A':'one',2:'two',3:'Three'} # Dictionary with Mixed Keys.
mydict
```

Out[49]:
```
{'A': 'one', 2: 'two', 3: 'Three'}
```

In [51]:
```
mydict.keys() # To get the keys as output.
```

Out[51]:
```
dict_keys(['A', 2, 3])
```

In [53]:
```
mydict.values() # To get the values as output.
```

Out[53]:
```
dict_values(['one', 'two', 'Three'])
```

In [55]:
```
mydict.items() # access each key-value pair that present within a dictionary.
```

Out[55]:
```
dict_items([('A', 'one'), (2, 'two'), (3, 'Three')])
```

In [57]:
```
mydict={1:'one',2:'two',3:'three','A':['ravi','teja','beri'],'B':['cat','bat','r
mydict  # Dictionary with nested list.
```

Out[57]:
```
{1: 'one',
 2: 'two',
 3: 'three',
 'A': ['ravi', 'teja', 'beri'],
 'B': ['cat', 'bat', 'rat']}
```

In [59]:
```
keys={'a','b','c','d'} # Creates a dictionary from a sequence of keys without va
d1=dict.fromkeys(keys)
d1
```

Out[59]:
```
{'c': None, 'd': None, 'b': None, 'a': None}
```

In [65]:
```
keys={'a','b','c','d'} # Creates a dictionary from a sequence of keys with value
value=10
d1=dict.fromkeys(keys, value)
d1
```

Out[65]:
```
{'c': 10, 'd': 10, 'b': 10, 'a': 10}
```

In [90]:
```
keys={'a','b','c','d'} # Creates a dictionary from a sequence of keys with value
value=[10,20,30,40,50]
d1=dict.fromkeys(keys, value)
d1
```

Out[90]:
```
{'c': [10, 20, 30, 40, 50],
 'd': [10, 20, 30, 40, 50],
 'b': [10, 20, 30, 40, 50],
 'a': [10, 20, 30, 40, 50]}
```

In [92]:
```
value.append(60) # adding the elements at last.
d1
```

Out[92]:
```
{'c': [10, 20, 30, 40, 50, 60],
 'd': [10, 20, 30, 40, 50, 60],
 'b': [10, 20, 30, 40, 50, 60],
 'a': [10, 20, 30, 40, 50, 60]}
```

Accessing Items

In [94]:
```
d1
```

Out[94]:
```
{'c': [10, 20, 30, 40, 50, 60],
 'd': [10, 20, 30, 40, 50, 60],
 'b': [10, 20, 30, 40, 50, 60],
 'a': [10, 20, 30, 40, 50, 60]}
```

In [96]:
```
d1.pop('b') # It removes/deletes items and pop's out the 'b' with entire 'values
```

```
Out[96]:  [10, 20, 30, 40, 50, 60]
```

```
In [172…  d1.popitem() # It pop's out with both 'keys' and 'value'.But, won't delete as 'p
                       # A random item is removed.
                        # Its the only difference between 'pop' and 'popitem'.
```

```
Out[172…  ('a', [10, 20, 30, 40, 50, 60])
```

```
In [104…  d1.pop('c')
          d1
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[104], line 1
----> 1 d1.pop('c')
      2 d1

KeyError: 'c'
```

```
In [106…  d1.pop('d')
```

```
Out[106…  [10, 20, 30, 40, 50, 60]
```

```
In [108…  d1
```

```
Out[108…  {'a': [10, 20, 30, 40, 50, 60]}
```

```
In [112…  d1['a'] # Accessing item using 'key'
```

```
Out[112…  [10, 20, 30, 40, 50, 60]
```

```
In [118…  d2={1:'one',2:'two',3:'three'}
          d2
```

```
Out[118…  {1: 'one', 2: 'two', 3: 'three'}
```

```
In [120…  d2[1]
```

```
Out[120…  'one'
```

```
In [122…  d2[3]
```

```
Out[122…  'three'
```

```
In [116…  d1.get('a') # Accessing item using get() method.
```

```
Out[116…  [10, 20, 30, 40, 50, 60]
```

```
In [152…  d3={'Name':'Raviteja Beri','Job':'Aspiring AI Engineer','ID':1989406,'Location':
          d3
```

```
Out[152…  {'Name': 'Raviteja Beri',
           'Job': 'Aspiring AI Engineer',
           'ID': 1989406,
           'Location': 'Hyderabad'}
```

```
In [154…  d3['Location']
```

```
Out[154…   'Hyderabad'
```

```
In [156…   d3.get('ID')
```

```
Out[156…   1989406
```

```
In [158…   d3['DOB'] = 2003 # Here i have added DOB in the dict.
```

```
In [160…   d3
```

```
Out[160…   {'Name': 'Raviteja Beri',
            'Job': 'Aspiring AI Engineer',
            'ID': 1989406,
            'Location': 'Hyderabad',
            'DOB': 2003}
```

```
In [162…   d3
```

```
Out[162…   {'Name': 'Raviteja Beri',
            'Job': 'Aspiring AI Engineer',
            'ID': 1989406,
            'Location': 'Hyderabad',
            'DOB': 2003}
```

```
In [164…   d3['DOB']=2000 # Changing several values.
           d3['Location']='India'
           d3
```

```
Out[164…   {'Name': 'Raviteja Beri',
            'Job': 'Aspiring AI Engineer',
            'ID': 1989406,
            'Location': 'India',
            'DOB': 2000}
```

```
In [166…   d2={'DOB': 2001} # Here ypu need to take another dict and assign value.
           d3.update(d2)    # otherwise It removes all values and returns as 'DOB':2001
           d3
```

```
Out[166…   {'Name': 'Raviteja Beri',
            'Job': 'Aspiring AI Engineer',
            'ID': 1989406,
            'Location': 'India',
            'DOB': 2001}
```

```
In [168…   d3
```

```
Out[168…   {'Name': 'Raviteja Beri',
            'Job': 'Aspiring AI Engineer',
            'ID': 1989406,
            'Location': 'India',
            'DOB': 2001}
```

```
In [174…   d3['Designation']='Software Engineer' # Adding items in the dictionary.
           d3
```

```
Out[174…  {'Name': 'Raviteja Beri',
           'Job': 'Aspiring AI Engineer',
           'ID': 1989406,
           'Location': 'India',
           'DOB': 2001,
           'Designation': 'Software Engineer'}
```

```
In [176…  del[d3['ID']] # removing item using del function.
          d3
```

```
Out[176…  {'Name': 'Raviteja Beri',
           'Job': 'Aspiring AI Engineer',
           'Location': 'India',
           'DOB': 2001,
           'Designation': 'Software Engineer'}
```

```
In [178…  d3.clear() # Delete all items of the dictionary using clear method.
          d3
```

```
Out[178…  {}
```

```
In [180…  del d3 # Delete dictionary object. Hence, deleted.
          d3
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[180], line 2
      1 del d3
----> 2 d3

NameError: name 'd3' is not defined
```

Copy dictionary

```
In [201…  d4={'Name':'Raviteja Beri','Job':'Aspiring AI Engineer','ID':1989406,'Location':
          d4
```

```
Out[201…  {'Name': 'Raviteja Beri',
           'Job': 'Aspiring AI Engineer',
           'ID': 1989406,
           'Location': 'Hyderabad'}
```

```
In [203…  d5=d4.copy() # Copying the items from on dictionary to other dictionary.
          d5
```

```
Out[203…  {'Name': 'Raviteja Beri',
           'Job': 'Aspiring AI Engineer',
           'ID': 1989406,
           'Location': 'Hyderabad'}
```

```
In [205…  id(d4), id(d5) # Gives different address.
```

```
Out[205…  (1971729213504, 1971729324736)
```

```
In [207…  d6=d5.copy()
```

```
In [209…  d6
```

Out[209…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'Hyderabad'}
```

In [211…

```python
d6=d5
```

In [213…

```python
id(d6), id(d5) # It consists same address becuase we have declared d6=d5 in the
```

Out[213…

```
(1971729324736, 1971729324736)
```

In [215…

```python
d5
```

Out[215…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'Hyderabad'}
```

In [217…

```python
d5['Location']='India'
d5
```

Out[217…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'India'}
```

In [219…

```python
d6 # Here d6 also changes because we have declared d5 and d6 are eqaul. So autom
```

Out[219…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'India'}
```

In [221…

```python
id(d5), id(d6) # same id's
```

Out[221…

```
(1971729324736, 1971729324736)
```

In [223…

```python
d4 # Copy of list won't be affected, if we made changes in the original.
```

Out[223…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'Hyderabad'}
```

In [225…

```python
id(d4)
```

Out[225…

```
1971729213504
```

Loop Through dictionary

In [232…

```python
d7={'Name':'Raviteja Beri','Job':'Aspiring AI Engineer','ID':1989406,'Location':
d7
```

Out[232…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'Hyderabad'}
```

In [242…

```
for i in d7:
    print(i, ':',d7[i]) # Because we need to Key & value pair
```

```
Name : Raviteja Beri
Job : Aspiring AI Engineer
ID : 1989406
Location : Hyderabad
```

In [244…

```
for i in enumerate(d7):
    print(i)
```

```
(0, 'Name')
(1, 'Job')
(2, 'ID')
(3, 'Location')
```

In [294…

```
for i,key in enumerate(d7):
    value=d7[key]
    print(i, ':',key, ':', value)
```

```
0 : Name : Raviteja Beri
1 : Job : Aspiring AI Engineer
2 : ID : 1989406
3 : Location : Hyderabad
```

In [300…

```
for i in d7: # it gives the values
    print(d7[i])
```

```
Raviteja Beri
Aspiring AI Engineer
1989406
Hyderabad
```

Dictionary membership

In [303…

```
d7
```

Out[303…

```
{'Name': 'Raviteja Beri',
 'Job': 'Aspiring AI Engineer',
 'ID': 1989406,
 'Location': 'Hyderabad'}
```

In [305…

```
'iD' in d7
```

Out[305…

```
False
```

In [307…

```
'ID'in d7
```

Out[307…

```
True
```

All / Any

In [312…

```
all(d7) # It returns false if it consists value '0' in it
```

Out[312…

```
True
```

In [ ]:

```

```