27-10-2025 practice pdf

SETS Set Creation

In [1]:
```
myset = {1,2,3,4,5}
myset
```

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

In [2]:
```
len(myset)
```

Out[2]: 5

In [3]:
```
my_set = {1,1,2,2,3,4,5,5}
my_set
```

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

In [4]:
```
myset1 = {1.79,2.08,3.99,4.56,5.45}
myset1
```

Out[4]: {1.79, 2.08, 3.99, 4.56, 5.45}

In [5]:
```
myset2 = {'Asif', 'John', 'Tyrion'}
myset2
```

Out[5]: {'Asif', 'John', 'Tyrion'}

In [6]:
```
myset3 = {10,20, "Hola", (11, 22, 32)}
myset3
```

Out[6]: {(11, 22, 32), 10, 20, 'Hola'}

In [7]:
```
myset3 = {10,20,"Hola",[11,22,32]}
myset3
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[7], line 1
----> 1 myset3 = {10,20,"Hola",[11,22,32]}
      2 myset3

TypeError: unhashable type: 'list'
```

In [8]:
```
myset4 = set()
print(type(myset4))
```

```
<class 'set'>
```

In [9]:
```
my_set1 = set(('one' , 'two' , 'three' , 'four'))
my_set1
```

Out[9]:
```
{'four', 'one', 'three', 'two'}
```

Loop through a Set

In [10]:
```
myset = {'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight'}

for i in myset:
    print(i)
```

```
two
one
four
eight
seven
six
five
three
```

In [14]:
```
for i in enumerate(myset):
    print(i)
```

```
(0, 'two')
(1, 'one')
(2, 'four')
(3, 'eight')
(4, 'seven')
(5, 'six')
(6, 'five')
(7, 'three')
```

Set Membership

In [12]:
```
myset
```

Out[12]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [13]:
```
'one' in myset
```

Out[13]:
```
True
```

In [15]:
```
'ten' in myset
```

Out[15]:
```
False
```

In [16]:
```
if 'three' in myset:
    print('Three is present in the set')
else:
    print('Three is not present in the set')
```

```
Three is present in the set
```

In [17]:
```
if 'eleven' in myset:
```

```
    print('eleven is present in the sety')
else:
    print('eleven is not present in the set')
```

```
eleven is not present in the set
```

Add & Remove Items

In [18]:
```
myset
```

Out[18]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [19]:
```
myset.add('NINE')
myset
```

Out[19]:
```
{'NINE', 'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [20]:
```
myset.update(['TEN' , 'ELEVEN' , 'TWELVE'])
myset
```

Out[20]:
```
{'ELEVEN',
 'NINE',
 'TEN',
 'TWELVE',
 'eight',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'three',
 'two'}
```

In [21]:
```
myset.remove('NINE')
myset
```

Out[21]:
```
{'ELEVEN',
 'TEN',
 'TWELVE',
 'eight',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'three',
 'two'}
```

In [22]:
```
myset.discard('TEN')
myset
```

Out[22]:
```
{'ELEVEN',
 'TWELVE',
 'eight',
 'five',
 'four',
 'one',
 'seven',
 'six',
 'three',
 'two'}
```

In [23]:
```
myset.clear()
myset
```

Out[23]:
```
set()
```

In [24]:
```
del myset
myset
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[24], line 2
      1 del myset
----> 2 myset

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

Copy Set

In [25]:
```
myset = {'one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight'}
myset
```

Out[25]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [26]:
```
myset1 = myset
myset1
```

Out[26]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [27]:
```
id(myset) , id(myset1)
```

Out[27]:
```
(1889108922144, 1889108922144)
```

In [28]:
```
my_set = myset.copy()
my_set
```

Out[28]:
```
{'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}
```

In [29]:
```
id(my_set)
```

Out[29]:
```
1889108921024
```

In [31]:
```
myset.add('nine')
myset
```

Out[31]: {'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}

In [32]:
```
myset1
```

Out[32]: {'eight', 'five', 'four', 'nine', 'one', 'seven', 'six', 'three', 'two'}

In [33]:
```
my_set
```

Out[33]: {'eight', 'five', 'four', 'one', 'seven', 'six', 'three', 'two'}

Set Operation

Union

In [34]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
C = {8,9,10}
```

In [35]:
```
A | B
```

Out[35]: {1, 2, 3, 4, 5, 6, 7, 8}

In [36]:
```
A.union(B)
```

Out[36]: {1, 2, 3, 4, 5, 6, 7, 8}

In [37]:
```
A.union(B, C)
```

Out[37]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

In [38]:
```
A.update(B,C)
A
```

Out[38]: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}

Intersection

In [39]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [40]:
```
A & B
```

Out[40]: {4, 5}

In [41]:
```
A.intersection(B) Intersection of A and B
```

```
  Cell In[41], line 1
    A.intersection(B) Intersection of A and B
                      ^
SyntaxError: invalid syntax
```

In [42]:
```
A.intersection_update(B)
A
```

Out[42]: {4, 5}

Difference

In [43]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [44]:
```
A - B
```

Out[44]: {1, 2, 3}

In [45]:
```
A.difference(B)
```

Out[45]: {1, 2, 3}

In [46]:
```
B - A
```

Out[46]: {6, 7, 8}

In [47]:
```
B.difference(A)
```

Out[47]: {6, 7, 8}

In [48]:
```
B.difference_update(A)
B
```

Out[48]: {6, 7, 8}

Symmetric Difference

In [49]:
```
A = {1,2,3,4,5}
B = {4,5,6,7,8}
```

In [50]:
```
A ^ B
```

Out[50]: {1, 2, 3, 6, 7, 8}

In [51]:
```
A.symmetric_difference(B)
```

Out[51]: {1, 2, 3, 6, 7, 8}

In [ ]:

In [52]:
```
A.symmetric_difference_update(B)
A
```

Out[52]: {1, 2, 3, 6, 7, 8}

Subset , Superset & Disjoint

In [53]:
```
A = {1,2,3,4,5,6,7,8,9}
B = {3,4,5,6,7,8}
C = {10,20,30,40}
```

In [54]:
```
B.issubset(A)
```

Out[54]: True

In [55]:
```
A.issuperset(B)
```

Out[55]: True

In [56]:
```
C.isdisjoint(A)
```

Out[56]: True

In [57]:
```
B.isdisjoint(A)
```

Out[57]: False

Other Builtin functions

In [58]:
```
A
```

Out[58]: {1, 2, 3, 4, 5, 6, 7, 8, 9}

In [59]:
```
sum(A)
```

Out[59]: 45

In [60]:
```
max(A)
```

Out[60]: 9

In [61]:
```
min(A)
```

Out[61]: 1

In [62]:
```
len(A)
```

Out[62]: 9

In [63]:
```
list(enumerate(A))
```

Out[63]: [(0, 1), (1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8), (8, 9)]

In [64]:
```
D= sorted(A,reverse=True)
D
```

Out[64]: [9, 8, 7, 6, 5, 4, 3, 2, 1]

In [65]:
```
sorted(D)
```

Out[65]: [1, 2, 3, 4, 5, 6, 7, 8, 9]

Dictionary

Create Dictionary

In [66]:
```
mydict = dict()
mydict
```

Out[66]: {}

In [67]:
```
mydict = {}
mydict
```

Out[67]: {}

In [68]:
```
mydict = {1:'one' , 2:'two' , 3:'three'}
mydict
```

Out[68]: {1: 'one', 2: 'two', 3: 'three'}

In [69]:
```
mydict = dict({1:'one' , 2:'two' , 3:'three'})
mydict
```

Out[69]: {1: 'one', 2: 'two', 3: 'three'}

In [70]:
```
mydict = {'A':'one' , 'B':'two' , 'C':'three'}
mydict
```

Out[70]: {'A': 'one', 'B': 'two', 'C': 'three'}

In [71]:
```
mydict = {1:'one' , 'A':'two' , 3:'three'}
mydict
```

Out[71]: {1: 'one', 'A': 'two', 3: 'three'}

In [72]:
```
mydict.keys()
```

Out[72]: dict_keys([1, 'A', 3])

In [73]:
```
mydict.values()
```

Out[73]: dict_values(['one', 'two', 'three'])

In [74]:
```
mydict.items()
```

Out[74]: dict_items([(1, 'one'), ('A', 'two'), (3, 'three')])

In [75]:
```
mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria']}
mydict
```

Out[75]: {1: 'one', 2: 'two', 'A': ['asif', 'john', 'Maria']}

In [78]:
```
mydict = {1:'one' , 2:'two' , 'A':['asif' , 'john' , 'Maria'], 'B':('Bat' , 'cat',
mydict
```

Out[78]:
```
{1: 'one',
 2: 'two',
 'A': ['asif', 'john', 'Maria'],
 'B': ('Bat', 'cat', 'hat')}
```

In [79]:
```
mydict = {1:'one' , 2:'two' , 'A':{'Name':'asif' , 'Age' :20}, 'B':('Bat' , 'cat',
mydict
```

Out[79]:
```
{1: 'one',
 2: 'two',
 'A': {'Name': 'asif', 'Age': 20},
 'B': ('Bat', 'cat', 'hat')}
```

In [80]:
```
keys = {'a' , 'b' , 'c' , 'd'}
mydict3 = dict.fromkeys(keys)
mydict3
```

Out[80]:
```
{'c': None, 'a': None, 'd': None, 'b': None}
```

In [81]:
```
keys = {'a' , 'b' , 'c' , 'd'}
value = 10
mydict3 = dict.fromkeys(keys , value)
mydict3
```

Out[81]:
```
{'c': 10, 'a': 10, 'd': 10, 'b': 10}
```

In [82]:
```
keys = {'a' , 'b' , 'c' , 'd'}
value = [10,20,30]
mydict3 = dict.fromkeys(keys , value)
mydict3
```

Out[82]:
```
{'c': [10, 20, 30], 'a': [10, 20, 30], 'd': [10, 20, 30], 'b': [10, 20, 30]}
```

In [83]:
```
value.append(40)
mydict3
```

Out[83]:
```
{'c': [10, 20, 30, 40],
 'a': [10, 20, 30, 40],
 'd': [10, 20, 30, 40],
 'b': [10, 20, 30, 40]}
```

Accessing Items

In [84]:
```
mydict = {1:'one' , 2:'two' , 3:'three' , 4:'four'}
mydict
```

Out[84]:
```
{1: 'one', 2: 'two', 3: 'three', 4: 'four'}
```

In [85]:
```
mydict[1]
```

Out[85]:
```
'one'
```

In [86]:
```
mydict.get(1)
```

Out[86]:
```
'one'
```

In [87]:
```
mydict1 = {'Name':'Asif' , 'ID':74123 , 'DOB': 1991 , 'job' :'Analyst'}
mydict1
```

Out[87]:
```
{'Name': 'Asif', 'ID': 74123, 'DOB': 1991, 'job': 'Analyst'}
```

In [88]:
```
mydict1['Name']
```

Out[88]:
```
'Asif'
```

In [90]:
```
mydict1.get('job')
```

Out[90]:
```
'Analyst'
```

Add, Remove & Change Items

In [91]:
```
mydict1 = {'Name':'Asif' , 'ID':12345 , 'DOB':1991 , 'Address' : 'Hilsiniki'}
mydict1
```

Out[91]:
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsiniki'}
```

In [92]:
```
mydict1['DOB'] = 1992
mydict1['Address'] = 'Delhi'
mydict1
```

Out[92]:
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1992, 'Address': 'Delhi'}
```

In [93]:
```
dict1 = {'DOB':1995}
mydict1.update(dict1)
mydict1
```

Out[93]:
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1995, 'Address': 'Delhi'}
```

In [95]:
```
mydict1['Job'] = 'Analyst'
mydict1
```

Out[95]:
```
{'Name': 'Asif',
 'ID': 12345,
 'DOB': 1995,
 'Address': 'Delhi',
 'Job': 'Analyst'}
```

In [96]:
```
mydict1.pop('Job')
mydict1
```

Out[96]:
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1995, 'Address': 'Delhi'}
```

In [97]:
```
mydict1.popitem()
```

Out[97]:
```
('Address', 'Delhi')
```

In [98]:
```
mydict1
```

Out[98]:
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1995}
```

In [99]:
```
del[mydict1['ID']]
mydict1
```

Out[99]:
```
{'Name': 'Asif', 'DOB': 1995}
```

In [100…
```
mydict1.clear()
mydict1
```

Out[100…
```
{}
```

In [101…
```
del mydict1
mydict1
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[101], line 2
      1 del mydict1
----> 2 mydict1

NameError: name 'mydict1' is not defined
```

Copy Dictionary

In [102…
```
mydict = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Hilsinki'}
mydict
```

Out[102…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsinki'}
```

In [103…
```
mydict1 = mydict
```

In [104…
```
id(mydict) , id(mydict1)
```

Out[104…
```
(1889109138240, 1889109138240)
```

In [105…
```
mydict2 = mydict.copy()
```

In [106…
```
id(mydict2)
```

Out[106…
```
1889109110656
```

In [107…
```
mydict['Address'] = 'Mumbai'
```

In [108…
```
mydict
```

Out[108…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Mumbai'}
```

In [109…
```
mydict1
```

Out[109…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Mumbai'}
```

In [110…
```
mydict2
```

Out[110…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Address': 'Hilsinki'}
```

Loop through a Dictionary

In [115…
```
mydict1 = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Address' : 'Hilsinki' ,'Job
mydict1
```

Out[115…
```
{'Name': 'Asif',
 'ID': 12345,
 'DOB': 1991,
 'Address': 'Hilsinki',
 'Job': 'Analyst'}
```

In [116…
```
for i in mydict1:
    print(i , ':' , mydict1[i])
```

```
Name : Asif
ID : 12345
DOB : 1991
Address : Hilsinki
Job : Analyst
```

In [117…
```
for i in mydict1:
    print(mydict1[i])
```

```
Asif
12345
1991
Hilsinki
Analyst
```

Dictionary Membership

In [118…
```
mydict1 = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Job': 'Analyst'}
mydict1
```

Out[118…
```
{'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Job': 'Analyst'}
```

In [119…
```
'Name' in mydict1
```

Out[119…
```
True
```

In [120…
```
'Asif' in mydict1
```

Out[120…
```
False
```

In [121…
```
'ID' in mydict1
```

Out[121…
```
True
```

In [122…
```
'Address' in mydict1
```

Out[122… False

All / Any

In [123…
```
mydict1 = {'Name':'Asif' , 'ID': 12345 , 'DOB': 1991 , 'Job': 'Analyst'}
mydict1
```

Out[123… {'Name': 'Asif', 'ID': 12345, 'DOB': 1991, 'Job': 'Analyst'}

In [124…
```
all(mydict1)
```

Out[124… True

In [ ]: