In [13]:
```
basket={'apple','mango','orange','grapes','apple','mango'} #duplicates removed
print(basket)
```

```
{'orange', 'mango', 'grapes', 'apple'}
```

In [11]:
```
'orange'in basket  #membership testing
```

Out[11]: True

In [5]:
```
'mango' in basket
```

Out[5]: True

In [7]:
```
'banana' in basket
```

Out[7]: False

In [21]:
```
#Demonstrate set operations on unique leeters from two words

a= set('MISSISSIPII')  #unique letters in a
b= set('ROSSUMGUIDO')
a
```

Out[21]: {'I', 'M', 'P', 'S'}

In [51]:
```
b
```

Out[51]: {'e', 'h', 'l', 'o'}

In [49]:
```
a-b
```

Out[49]: {'i', 'm', 'p', 's'}

In [35]:
```
a =set('mississippi')
b =set('hellolleh')
a
```

Out[35]: {'i', 'm', 'p', 's'}

In [37]:
```
b
```

Out[37]: {'e', 'h', 'l', 'o'}

In [39]:
```
a-b        #letterrs in a but not in b
```

Out[39]: {'i', 'm', 'p', 's'}

In [53]:
```
b-a  #letters in b but not in a
```

Out[53]: {'e', 'h', 'l', 'o'}

In [55]:
```
a|b   #letters in both a or b -----but removin the duplicates
```

Out[55]:
```
{'e', 'h', 'i', 'l', 'm', 'o', 'p', 's'}
```

In [57]:
```
a & b
```

Out[57]:
```
set()
```

In [61]:
```
a ^ b      #letters in a or b but not both
```

Out[61]:
```
{'e', 'h', 'i', 'l', 'm', 'o', 'p', 's'}
```

In [ ]:
```

```

Dictionaries

In [66]:
```
a ={'rossum':4799,'guido':3265,'john':6448}
a['sape']=67590
a
```

Out[66]:
```
{'rossum': 4799, 'guido': 3265, 'john': 6448, 'sape': 67590}
```

In [68]:
```
del a['sape']
a['irv']=5748
a
```

Out[68]:
```
{'rossum': 4799, 'guido': 3265, 'john': 6448, 'irv': 5748}
```

In [70]:
```
del a['rossum']
a
```

Out[70]:
```
{'guido': 3265, 'john': 6448, 'irv': 5748}
```

In [72]:
```
del a['guido']
a['rossum']=1111
a
```

Out[72]:
```
{'john': 6448, 'irv': 5748, 'rossum': 1111}
```

In [74]:
```
'rossum' in a
```

Out[74]:
```
True
```

In [76]:
```
'john' in a
```

Out[76]:
```
True
```

In [80]:
```
'sape'in a
```

Out[80]:
```
False
```

{x:x**2 for x in (2,4,6)}

In [84]:
```
{y:y**5 for y in (5,10,20)}
```

Out[84]:
```
{5: 3125, 10: 100000, 20: 3200000}
```

In [86]:
```
dict(sape=4689,guido=6576, rossum=36533)
```

Out[86]:
```
{'sape': 4689, 'guido': 6576, 'rossum': 36533}
```

In [ ]:
```

```

Using Python as a calculator

In [90]:
```
2+2
```

Out[90]:
```
4
```

In [92]:
```
50-5*6
```

Out[92]:
```
20
```

In [94]:
```
(60-6*7)/4
```

Out[94]:
```
4.5
```

In [96]:
```
14/4
```

Out[96]:
```
3.5
```

In [100…
```
15//5
```

Out[100…
```
3
```

In [102…
```
17 % 3
```

Out[102…
```
2
```

List

In [107…
```
squares = [1,3,15,30,45]
squares
```

Out[107…
```
[1, 3, 15, 30, 45]
```

In [111…
```
squares[1]
```

Out[111… 3

In [113…
```
squares[3]
```

Out[113… 30

In [115…
```
squares[9]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[115], line 1
----> 1 squares[9]

IndexError: list index out of range
```

In [117…
```
squares[-3]
```

Out[117… 15

In [119…
```
squares[-1:]
```

Out[119… [45]

In [121…
```
squares[:-1]
```

Out[121… [1, 3, 15, 30]

List are mutable

In [128…
```
l=[10,20,30,40,50]
l
```

Out[128… [10, 20, 30, 40, 50]

In [130…
```
l.append(60)
l
```

Out[130… [10, 20, 30, 40, 50, 60]

In [132…
```
l.remove(60)
l
```

Out[132… [10, 20, 30, 40, 50]

In [135…
```
rgb = ["Green","Black","Blue"]
rgba=rgb
id(rgb)==id(rgba)
```

Out[135… True

In [ ]:

Range() Function

In [138…
```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

In [140…
```
list(range(10,20))
```

Out[140…
```
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

In [142…
```
list(range(0,10,4))
```

Out[142…
```
[0, 4, 8]
```

In [144…
```
list(range(-10,-100,-30))
```

Out[144…
```
[-10, -40, -70]
```

In [148…
```
a = ['Mary','had','a','little','lamb']
for i in range(len(a)):
    print(i,a[i])
```

```
0 Mary
1 had
2 a
3 little
4 lamb
```

In [ ]:

Tuple and sequences

In [151…
```
t= 6789,1234,'python!'
t[0]
```

Out[151…
```
6789
```

In [153…
```
t
```

Out[153…
```
(6789, 1234, 'python!')
```

In [155…
```
u =t,(1,2,3,4,5)  #Tuples may be nested
u
```

Out[155…
```
((6789, 1234, 'python!'), (1, 2, 3, 4, 5))
```

In [157…
```
t[0]=5432    #Tuples are imm
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[157], line 1
----> 1 t[0]=5432

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

In [ ]: