Review python basic

String operations

In [5]:
```
"my name is"
```

Out[5]:'my name is'

In [6]:
```
'1 2 3 '
```

Out[6]:'1 2 3 '

In [7]:
```
'@#_#]&*$%'
```

Out[7]:'@#_#]&*$%'

In [8]:
```
print("good morning!")
```

good morning!

In [8]:
```
name= "Ahmed khaled"
name
```

Out[8]:'Ahmed khaled'

In [9]:
```
print(name[0])
```

A

In [10]:
```
print(name[6])
```

k

In [11]:
```
print(name[-1])
```

d

In [13]:
```
name[0:4]
```

Out[13]:'Ahme'

In [14]:
```
name[8:12]
```

Out[14]:'aled'

In [15]:
```
name[::2]
```

Out[15]:'Amdkae'

In [16]:
```
name[0:5:2]
```

Out[16]:'Amd'

In [17]:
```
S =name + "is the best"
S
```

Out[17]:'Ahmed khaledis the best'

In [18]:
```
3*"Hello"
```

Out[18]:'HelloHelloHello'

In [19]:
```
name="Lamia Ahmed "
name=name + "is the best"
name
```

Out[19]:'Lamia Ahmed is the best'

In [20]:
```
print("Lamia Ahmed \n is the best")
```

Lamia Ahmed
 is the best

In [21]:
```
print("Lamia Ahmed \t is the best")
```

Lamia Ahmed is the best

In [22]:
```
print("Lamia Ahmed \\ is the best")
```

Lamia Ahmed \ is the best

In [23]:
```
print(r"Lamia Ahmed \ is the best")
```

Lamia Ahmed \ is the best

In [24]:
```
a= "Good morning"
print("before upper:",a)
b=a.upper()
print("After upper:",b)
```

before upper: Good morning
After upper: GOOD MORNING

In [25]:
```
a="He is the best"
b=a.replace('He','she')
b
```

Out[25]:'she is the best'

In [26]:
```
name="Good day"
name.find('od')
```

Out[26]:2

In [27]:
```
name.find('day')
```

Out[27]:5
Tuples in python

In [28]:
```
tuple1=("day",20,1.5)
tuple1
```

Out[28]:('day', 20, 1.5)
In [29]:
```
type(tuple1)
```

Out[29]:tuple
In [30]:
```
print(tuple1[0])
print(tuple1[1])
print(tuple1[2])
```

```
day
20
1.5
```
In [31]:
```
print(type(tuple1[0]))
print(type(tuple1[1]))
print(type(tuple1[2]))
```

```
<class 'str'>
<class 'int'>
<class 'float'>
```
In [32]:
```
tuple1[-1]
```

Out[32]:1.5
In [33]:
```
tuple1[-2]
```

Out[33]:20
In [34]:
```
tuple1[-3]
```

Out[34]:'day'
In [35]:
```
tuple2=tuple1 +("time",10)
tuple2
```

Out[35]:('day', 20, 1.5, 'time', 10)
In [36]:
```
tuple2[0:3]
```

Out[36]:('day', 20, 1.5)
In [37]:
```
tuple2[3:5]
```

Out[37]:('time', 10)
In [38]:
```
len(tuple2)
```

Out[38]:5
In [39]:
```
Ratings=(0,14,6,3,10,8,9,4,2)
```

In [40]:
```
RatingsSorted=sorted(Ratings)
RatingsSorted
```

Out[40]:[0, 2, 3, 4, 6, 8, 9, 10, 14]
In [41]:
```
NestedT=(1,2,("p","happy"),(3,3),("day",(5,2)))
```

In [42]:
```
print("Element 0 of Tuple:",NestedT[0])
print("Element 1 of Tuple:",NestedT[1])
print("Element 2 of Tuple:",NestedT[2])
print("Element 3 of Tuple:",NestedT[3])
print("Element 4 of Tuple:",NestedT[4])
```

```
Element 0 of Tuple: 1
Element 1 of Tuple: 2
Element 2 of Tuple: ('p', 'happy')
Element 3 of Tuple: (3, 3)
Element 4 of Tuple: ('day', (5, 2))
```
In [43]:
```
print("Element 2,0 of Tuple:",NestedT[2][0])
print("Element 2,1 of Tuple:",NestedT[2][1])
print("Element 3,0 of Tuple:",NestedT[3][0])
print("Element 3,1 of Tuple:",NestedT[3][1])
print("Element 4,0 of Tuple:",NestedT[4][0])
print("Element 4,1 of Tuple:",NestedT[4][1])
```

```
Element 2,0 of Tuple: p
Element 2,1 of Tuple: happy
Element 3,0 of Tuple: 3
Element 3,1 of Tuple: 3
Element 4,0 of Tuple: day
Element 4,1 of Tuple: (5, 2)
```
In [44]:
```
NestedT[2][1][0]
```

Out[44]:'h'
In [45]:
```
NestedT[2][1][1]
```

Out[45]:'a'

In [46]:
```
NestedT[4][1][0]
```

Out[46]:5

In [47]:
```
NestedT[4][1][1]
```

Out[47]:2

In [ ]:

In [1]:
```
N = ["Ali Kaild", 11.2, 1879]
N
```

Out[1]:['Ali Kaild', 11.2, 1879]

In [2]:
```
print ('the same element using negative and positive indexing: \n postive:', N,[0],
    '\n Negative:', N[-3])
print ('the same element using negative and positive indexing: \n postive:', N,[1],
    '\n Negative:', N[-2])
print ('the same element using negative and positive indexing: \n postive:', N,[2],
    '\n Negative:', N[-1])
```

```
the same element using negative and positive indexing:
 postive: ['Ali Kaild', 11.2, 1879] [0]
 Negative: Ali Kaild
the same element using negative and positive indexing:
 postive: ['Ali Kaild', 11.2, 1879] [1]
 Negative: 11.2
the same element using negative and positive indexing:
 postive: ['Ali Kaild', 11.2, 1879] [2]
 Negative: 1879
```

In [3]:
```
["Ali Kalid", 11.2,1879,[3,4],("B",3)]
```

Out[3]:['Ali Kalid', 11.2, 1879, [3, 4], ('B', 3)]

In [4]:
```
N =["Ali Kalid",11.2,1879,"AK",3]
N
```

Out[4]:['Ali Kalid', 11.2, 1879, 'AK', 3]

In [5]:
```
N[3:5]
```

Out[5]:['AK', 3]

In [6]:
```
N = ["Ali Kalid", 11.3]
N.extend(['pop',11])
N
```

Out[6]:['Ali Kalid', 11.3, 'pop', 11]

In [7]:
```
N =["Ali Kalid",11.3]
N.append(['pop',11])
N
```

Out[7]:['Ali Kalid', 11.3, ['pop', 11]]

In [8]:
```
N = ["Ali Kalid", 11.3]
N.extend(['pop',11])
N
```

Out[8]:['Ali Kalid', 11.3, 'pop', 11]

In [9]:
```
N.append(['a,b'])
N
```

Out[9]:['Ali Kalid', 11.3, 'pop', 11, ['a,b']]

In [10]:
```
A=["diisco", 11,2.1]
print ('Before change:',A)
A[0]= 'hard rock'
print ('After change:',A)
```

```
Before change: ['diisco', 11, 2.1]
After change: ['hard rock', 11, 2.1]
```

In [11]:
```
print ('Before change:',A)
del (A[0])
print('After change :',A)
```

```
Before change: ['hard rock', 11, 2.1]
After change : [11, 2.1]
```

In [12]:
```
'hard rock'.split()
```

Out[12]:['hard', 'rock']

In [13]:
```
'E,Z,R,H'.split(',')
```

Out[13]:['E', 'Z', 'R', 'H']

In [14]:
```
E=["hard rock",11,2.1]
Z=E
print('E:',E)
print('Z:',Z)
```

```
E: ['hard rock', 11, 2.1]
Z: ['hard rock', 11, 2.1]
```

In [15]:
```
print('Z[0]:',Z[0])
E[0]= "Apple"
print ('Z[0]:',Z[0])
```

```
Z[0]: hard rock
Z[0]: Apple
```

In [16]:
```
Z=E[:]
Z
```

Out[16]:['Apple', 11, 2.1]

In [17]:
```
print('Z[0]:',Z[0])
E[0]= "hard rock"
print ('Z[0]:',Z[0])
```

```
Z[0]: Apple
Z[0]: Apple
```

In [18]:
```
Dict = {"key1":1,"key2":"2","key3":[3,3,3],"key4":(4,4,4),('key5'):5,(0,1):6}
Dict
```

Out[18]:
```
{'key1': 1,
 'key2': '2',
 'key3': [3, 3, 3],
 'key4': (4, 4, 4),
 'key5': 5,
 (0, 1): 6}
```

In [19]:
```
Dict["key1"]
```

Out[19]:1

In [20]:
```
Dict[(0, 1)]
```

Out[20]:6

In [21]:
```
release_year_dict ={"Thriller": "1999", "Back in Black": "1970",\
                    "The Dark Side of the Moon": "1977","The Bodyguard": "1993",\
                    "Bat Out of Hell":"1978", "Their Greatest Hits(1972-1976)": "19977",\
                    "Monday Night Fever": "1976","Rumours": "1976"}
release_year_dict
```

Out[21]:
```
{'Thriller': '1999',
 'Back in Black': '1970',
 'The Dark Side of the Moon': '1977',
 'The Bodyguard': '1993',
 'Bat Out of Hell': '1978',
 'Their Greatest Hits(1972-1976)': '19977',
 'Monday Night Fever': '1976',
 'Rumours': '1976'}
```

In [22]:
```
release_year_dict['Thriller']
```

Out[22]:'1999'

In [23]:
```
release_year_dict['The Bodyguard']
```

Out[23]:'1993'

In [24]:
```
release_year_dict.keys()
```

Out[24]:dict_keys(['Thriller', 'Back in Black', 'The Dark Side of the Moon', 'The Bodyguard', 'Bat Out of Hell', 'Their Greatest Hits(1972-1976)', 'Monday Night F
ever', 'Rumours'])

In [25]:
```
release_year_dict.values()
```

Out[25]:dict_values(['1999', '1970', '1977', '1993', '1978', '19977', '1976', '1976'])

In [26]:
```
release_year_dict['Graduation'] = '2010'
release_year_dict
```

Out[26]:
```
{'Thriller': '1999',
 'Back in Black': '1970',
 'The Dark Side of the Moon': '1977',
 'The Bodyguard': '1993',
 'Bat Out of Hell': '1978',
 'Their Greatest Hits(1972-1976)': '19977',
 'Monday Night Fever': '1976',
 'Rumours': '1976',
 'Graduation': '2010'}
```

In [27]:
```
del(release_year_dict['Thriller'])
del(release_year_dict['Graduation'])
release_year_dict
```

Out[27]:
```
{'Back in Black': '1970',
 'The Dark Side of the Moon': '1977',
 'The Bodyguard': '1993',
 'Bat Out of Hell': '1978',
 'Their Greatest Hits(1972-1976)': '19977',
 'Monday Night Fever': '1976',
 'Rumours': '1976'}
```

In [28]:
```
'The Bodyguard' in release_year_dict
```

Out[28]:True

In [29]:
```
soundtrack_dic = {"The Bodyguard":"1993", "Monday Night Fever":"1976"}
soundtrack_dic
```

Out[29]:
```
{'The Bodyguard': '1993', 'Monday Night Fever': '1976'}
```

In [ ]:

Sets

In [8]:
```
A={"name","time","name","L&A","namr"}
A
```

Out[8]:{'L&A', 'name', 'namr', 'time'}

In [13]:
```
A =set (["Group","number","happy"])
A
```

Out[13]:{'Group', 'happy', 'number'}

In [15]:
```
A.add("time")
A
```

Out[15]:{'Group', 'happy', 'number', 'time'}

In [16]:
```
A.remove("time")
A
```

Out[16]:{'Group', 'happy', 'number'}

In [17]:
```
"AD/FG" in A
```

Out[17]:False

In [18]:
```
B=set(["Study",'Ac/ff','clock time'])
C=set(['AC/ff',"clock time",'Good day'"'])
```

In [19]:
```
B,C
```

Out[19]:({'Ac/ff', 'Study', 'clock time'}, {'AC/ff', 'Good day"', 'clock time'})

In [22]:
```
N=B & C
N
```

Out[22]:{'clock time'}

In [23]:
```
B.difference(C)
```

Out[23]:{'Ac/ff', 'Study'}

In [24]:
```
C.difference(B)
```

Out[24]:{'AC/ff', 'Good day"'}

In [25]:
```
B.intersection(C)
```

Out[25]:{'clock time'}

In [26]:
```
B.union(C)
```

Out[26]:{'AC/ff', 'Ac/ff', 'Good day"', 'Study', 'clock time'}

Conditions in python

In [29]:
```
a=4
a==5
```

Out[29]:False

In [30]:
```
i=7
i>5
```

Out[30]:True

In [31]:
```
a=5
a !=5
```

Out[31]:False

In [32]:
```
"AAAA"=="Good moring"
```

Out[32]:False

In [33]:
```
"AAAA"!= "Good morning"
```

Out[33]:True

In [34]:
```
'B'>'A'
```

Out[34]:True

In [35]:
```
'BA'>'AB'
```

Out[35]:True

In [39]:
```
age = 18

if age >18:
    print("You can enter")
elif age == 18:
    print("go see pink floyd")
else:
     print("go see meat Loaf")

print("move on")
```

go see pink floyd
move on

In [6]:
```
Year =1973
Year=1960
```

```
    if Year > 1970:
        print("A year is greater than 1850")
    print('do something..')
```

```
do something..
```

In [8]:
```
Year=1900
if(Year<1890) or (Year>1899):
    print("Year was not made in the 1890")
else:
    print("Year was made in the 1890")
```

```
Year was not made in the 1890
```

In [9]:
```
Year=1900
if not (Year=='1901'):
    print("Year was not made in the 1901")
```

```
Year was not made in the 1901
```

In [ ]:

In [1]:
```
range (10)
```

Out[1]:range(0, 10)

In [2]:
```
dates = [2002 , 1999 , 2000]
N = len(dates)

for i in range(N) :
    print(dates[i])
```

```
2002
1999
2000
```

In [4]:
```
for i in range(0 , 9) :
    print(i)
```

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

In [5]:
```
for year in dates:
    print(year)
```

```
2002
1999
2000
```

In [7]:
```
squares = ['blue' , 'yellow' , 'green' , 'pink' , 'red']

for i in range ( 0, 5):
    print("Before square " , i , 'is' , squares[i])
    squares[i] = 'white'
    print("After square " ,i, 'is' , squares[i])
```

```
Before square  0 is blue
After square  0 is white
Before square  1 is yellow
After square  1 is white
Before square  2 is green
After square  2 is white
Before square  3 is pink
After square  3 is white
Before square  4 is red
After square  4 is white
```

In [10]:
```
squares=['blue' , 'yellow' , ' green' , 'pink ' , 'red']

for i, square in enumerate(squares):
    print(i, square)
```

```
0 blue
1 yellow
2  green
3 pink
4 red
```

In [11]:
```
dates = [ 2002 , 1999 , 2000, 1990]

i= 0
year = dates[0]

while(year != 1990):
    print(year)
    i = i + 1
    year = dates[i]


print("It took ", i , "repetitions to get out of loop.")
```

```
2002
1999
2000
It took  3 repetitions to get out of loop.
```

In [16]:
```
help (add)
```

```
Help on function add in module __main__:

add(a)
    add 1 to 5
```

In [17]:
```
add(8)
```

```
5  if you add one 6
```
Out[17]:6

In [21]:
```
def Mult(a, b):
    c = a * b
    return(c)
    print('This is not printed')

result = Mult(12,2)
print(result)
```

```
24
```
In [22]:
```
Mult(4,8)
```

Out[22]:32

In [23]:
```
Mult(5.2 , 2.14)
```

Out[23]:11.128000000000002

In [24]:
```
Mult(4, "Michael Jackson ")
```

Out[24]:'Michael Jackson Michael Jackson Michael Jackson Michael Jackson '

In [27]:
```
def square(a):

    b = 1
    c = a * a + b
    print(a, "if you square + 1 " , c)
    return(c)
```

In [28]:
```
s = 4
g = square(s)
g
```

```
4 if you square + 1  17
```
Out[28]:17

In [29]:
```
square (8)
```

```
8 if you square + 1  65
```
Out[29]:65

In [30]:
```
def LM() :
    print (' Ghaida ')

def LM () :
    print ('Ghaida ')
    return(None)
```

In [31]:
```
LM()
```

```
Ghaida
```
In [34]:
```
print (LM())
print(LM())
```

```
Ghaida
None
Ghaida
None
```
In [35]:
```
def con(a,b) :
    return(a+b)
```

In [36]:
```
con ("This ", "is")
```

Out[36]:'This is'

In [38]:
```
a1 = 7
b1 = 8
c1 = a1 + b1 + 2 * a1 * b1 - 1
if(c1 < 0 ):
    c1 = 0
else :
    c1 = 5
c1
```

Out[38]:5

In [39]:
```
a2 = 0
b2 = 0
c2 = a2 + b2 + 2 * a2 * b2 -1
if ( c2 < 0 ):
    c2 = 0
else :
    c2 = 5
c2
```

Out[39]:0

In [46]:
```
def Equation (a,b) :
    c = a + b + 2 * a * b - 1
    if(c < 0):
        c = 0
```

```
    else:
        c = 5
    return(c)
```

In [47]:
```
a1 = 8
b1 = 9
c1 = Equation(a1,b1)
c1
```

Out[47]:5

In [49]:
```
a2 = 2
b2 = 2
c2 + Equation(a2,b2)
c2
```

Out[49]:0

In [51]:
```
album_ratings = [11.9,5.9,8.3,2.2,10.0,6.6,7.9,4.44]
print(album_ratings)
```

[11.9, 5.9, 8.3, 2.2, 10.0, 6.6, 7.9, 4.44]

In [52]:
```
sum(album_ratings)
```

Out[52]:57.239999999999995

In [53]:
```
len(album_ratings)
```

Out[53]:8

In [54]:
```
def PrintList(the_list):
    for element in the_list:
        print(element)
```

In [58]:
```
PrintList(['5' , 1 , 'the woman', "wow"])
```

5
1
the woman
wow

In [59]:
```
def isGoodRating(rating=4):
    if(rating < 8):
        print("this album sucks it's rating is" , rating)

    else:
        print("this album is good its rating is", rating)
```

In [60]:
```
isGoodRating()
isGoodRating(6)
```

this album sucks it's rating is 4
this album sucks it's rating is 6

In [67]:
```
artist = " Ghaida"
def printer(artist):
    internal_var = artist
    print(artist, "is an artist")

printer(artist)
```

Ghaida is an artist

In [70]:
```
myFavouriteBand = "GS/NL"

def getBandRating(bandname):
    if bandname == myFavouriteBand:
        return 5.5
    else: return 8,4

print("GS/NL' s rating is :", getBandRating("GS/NL"))
print("Deep Purple's rating is:",getBandRating("Deep Purple"))
print("My Favourite band is" , myFavouriteBand)
```

GS/NL' s rating is : 5.5
Deep Purple's rating is: (8, 4)
My Favourite band is GS/NL

In [71]:
```
def printDictionary(**args):
    for key in args:
        print(key + " : " + args[key])

printDictionary(Country='Canda ',Province='Ontario',City='Toronto')
```

Country : Canda
Province : Ontario
City : Toronto

In [75]:
```
def addItems(list):
    list.append("Three")
    list.append("Four")

myList = ["One" , "Two"]
```

```
addItems(myList)

myList
```

```
Out[75]:['One', 'Two', 'Three', 'Four']
In [ ]:
```