Loops

Common loops

1. For
2. While
3. Do While

1. Syntax for For Loop

for in (how many times the loop run): print(variable name)

In [1]:
```
for i in range(0,5):        # "i" is a variable and 0 to 5 specifies the range for the loop , the value of i ch
    print(i)                # this line defines the action of the loop
```
```
0
1
2
3
4
```

In [2]:
```
for i in range(6,12):        # "i" is a variable and 0 to 5 specifies the range for the loop ,
                             #the value of i changes for every loop
    print(i)                 # this line defines the action of the loop
```
```
6
7
8
9
10
11
```

In [3]:
```
for i in range(0,5):        # "i" is a variable and 0 to 5 specifies the range for the loop , the value of i ch
    print("Hello")              # this line defines the action of the loop
```
```
Hello
Hello
Hello
Hello
Hello
```

In [4]:
```
mylist = ["machine", " learning", "is", "fun"]
type(mylist)
```
Out[4]: list

In [5]:
```
len(mylist)
```
Out[5]: 4

In [6]:
```
for i in range(0,len(mylist)):  #automate the count
    print(i)
    print(mylist[i])  #print individual item in the list
```
```
0
machine
1
 learning
2
is
3
fun
```

In [7]:
```
for item in mylist:
    print(item)
```
```
machine
 learning
is
fun
```

In [8]:
```
a=[12,24,54,87,57,68]
for i in range(0,len(a)):
    print(a[i])
```
```
12
24
54
87
57
68
```

In [9]:
```
a=[12,24,54,87,57,68]
for i in range(0,len(a)):
    print(a)
```
```
[12, 24, 54, 87, 57, 68]
[12, 24, 54, 87, 57, 68]
[12, 24, 54, 87, 57, 68]
[12, 24, 54, 87, 57, 68]
[12, 24, 54, 87, 57, 68]
[12, 24, 54, 87, 57, 68]
```

In [10]:
```
for i in range(5,0,-2):
    print(i)
```
```
5
3
1
```

In [11]:
```
a=[12,24,54,87,57,68]
for i in range(0,len(a)):
    if(a[i]%2==0):
        print("the number",a[i],"is even")
    else:
        print("the number",a[i],"is odd hence converting it in even",a[i]+1) # append
```
```
the number 12 is even
the number 24 is even
the number 54 is even
the number 87 is odd hence converting it in even 88
the number 57 is odd hence converting it in even 58
the number 68 is even
```

In [12]:
```
a=[12,24,54,87,57,68]
for i in range(0,len(a)):
    if(a[i]%2==0):
        print("the number",a[i],"is even")
```
```
the number 12 is even
the number 24 is even
the number 54 is even
the number 68 is even
```

I have a list and i want to print all the elements

In [13]:
```
a=[12,24,54,87,57,68]
for i in range(0,len(a)):
    print(a[i])
```
```
12
24
54
87
57
68
```

In [14]:
```
#applying loop in reverse

for i in range(0,9,2):   #"2" indicates the step take by the loop
    print(i)
```
```
0
2
4
6
8
```

In [15]:
```
a=[1,2,3,1,5,3,1,2,2,10,2,1,0,2,4,0]
for i in range(0,9,2):    #"2" indicates the step take by the loop
    sum=0
    sum=sum+a[i]
    print(sum)
print("the sum is",sum)
```
```
1
3
5
1
2
the sum is 2
```

Brackets

() function/making a tuples

[] indexing/accesing a particular object/define a list

{} Dictionaries/defining Sets

In [16]:
```
websites_lst = ["analytics.com","towardsdatascience.com","medium.com"]
```

In [17]:
```
print(websites_lst)
```
```
['analytics.com', 'towardsdatascience.com', 'medium.com']
```

In [18]:
```
for i in range(0,len(websites_lst)):
    print(websites_lst[i])
```
```
analytics.com
towardsdatascience.com
medium.com
```

In [19]:
```
for i in range(0,len(websites_lst)):
    print(websites_lst[i][0])      #[0] indicates the indexing of the element
```
```
a
t
m
```

In [20]:
```
for i in range(0,len(websites_lst)):
    print(websites_lst[i][3])
```
```
l
a
i
```

In [21]:
```
websites_lst[0]="analyticsvidya.com"
```

In [22]:
```
websites_lst
```
Out[22]: ['analyticsvidya.com', 'towardsdatascience.com', 'medium.com']

In [23]:
```
for i in range(0,len(websites_lst)):
    if(i%2==0):
        print(websites_lst[i])
```
```
analyticsvidya.com
medium.com
```

In [24]:
```
websites_lst.append("these","are","good","websites")     #Doubt : how to add multiple elements in a single appe
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-24-3d8499b64daa> in <module>
----> 1 websites_lst.append("these","are","good","websites")     #Doubt : how to add multiple elements in a sin
gle append command ?

TypeError: append() takes exactly one argument (4 given)
```

In [25]:
```
websites_lst.append("these")
```

In [26]:
```
websites_lst
```
Out[26]: ['analyticsvidya.com', 'towardsdatascience.com', 'medium.com', 'these']

In [27]:
```
apped_lst = ["these","are","good","websites"]
apped_lst
```
Out[27]: ['these', 'are', 'good', 'websites']

In [28]:
```
websites_lst.append(apped_lst)
```

In [29]:
```
websites_lst
```
```
['analyticsvidya.com',
 'towardsdatascience.com',
 'medium.com',
 'these',
 ['these', 'are', 'good', 'websites']]
```

In [30]:
```
for new_element in apped_lst:
    websites_lst.append(new_element)
```

In [31]:
```
websites_lst
```
```
['analyticsvidya.com',
 'towardsdatascience.com',
 'medium.com',
 'these',
 ['these', 'are', 'good', 'websites'],
 'these',
 'are',
 'good',
 'websites']
```

In [32]:
```
apped_lst
```
Out[32]: ['these', 'are', 'good', 'websites']

Extend

To add elements directly to the list

In [33]:
```
websites_lst.extend(apped_lst)
```

In [34]:
```
websites_lst
```
```
['analyticsvidya.com',
 'towardsdatascience.com',
 'medium.com',
 'these',
 ['these', 'are', 'good', 'websites'],
 'these',
 'are',
 'good',
 'websites',
 'these',
 'are',
 'good',
 'websites']
```

Check if element is in the list

In [35]:
```
'websites' in websites_lst
```
Out[35]: True

In [36]:
```
#Print the value if exist
websites_lst['websites' in websites_lst] # it is not reliable to check location, as it will give only 1 for Tru
```
Out[36]: 'towardsdatascience.com'

In [37]:
```
websites_lst['hello' in websites_lst]
```
Out[37]: 'analyticsvidya.com'

In [38]:
```
#in this list print all the ODD nos.
a=[1,4,99,65,69,55,47,74,2549]
for i in range(0,len(a)):
    if(a[i]%2!=0):      # != denotes "not equal"
        print("the number ",a[i],"is odd")
```
```
the number  65 is odd
the number  69 is odd
the number  55 is odd
the number  47 is odd
the number  2549 is odd
```

List operations

In [39]:
```
a=[12,13,24,324,12,12,12,12]
```

In [40]:
```
b=[232,24423,5452,52]
```

In [41]:
```
a+b
```
Out[41]: [12, 13, 24, 324, 12, 12, 12, 12, 232, 24423, 5452, 52]

In [42]:
```
c=a+b
```

In [43]:
```
c
```
Out[43]: [12, 13, 24, 324, 12, 12, 12, 12, 232, 24423, 5452, 52]

In [44]:
```
del c
```

In [45]:
```
c
```
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-45-2b66fd261ee5> in <module>
----> 1 c

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

In [46]:
```
c=a+b
```

In [47]:
```
a.count(12)       # count function is used to count the occurence of a particular entry within a list
```
Out[47]: 4

In [48]:
```
c
```
Out[48]: [12, 13, 24, 324, 12, 12, 12, 232, 24423, 5452, 52]

In [50]:
```
min(c)          # print the minimum value within a list
```
Out[50]: 12

In [51]:
```
max(c)        # print the max value within a list
```
Out[51]: 24423

In [52]:
```
a.clear()     #clear the complete list
```

In [53]:
```
a
```
Out[53]: []

In [54]:
```
a=[1,95,87,45]
```

In [55]:
```
a[1]=35
```

In [56]:
```
a
```
Out[56]: [1, 35, 87, 45]

In [57]:
```
a[3] = 0
a
```
Out[57]: [1, 35, 87, 0]

In [58]:
```
a.insert(3,200)  # insert elemen to any desided location
```

In [59]:
```
a
```
Out[59]: [1, 35, 87, 200, 0]

In [60]:
```
a.index(200)
```
Out[60]: 3

In [61]:
```
a.remove(200)
```

In [62]:
```
a
```
Out[62]: [1, 35, 87, 0]

In [63]:
```
a.append(21)
```

In [64]:
```
a
```
Out[64]: [1, 35, 87, 0, 21]

In [65]:
```
a.pop(1)
```
Out[65]: 35

In [66]:
```
a
```
Out[66]: [1, 87, 0, 21]

In [67]:
```
a.pop(3)
```
Out[67]: 21

In [68]:
```
a
```
Out[68]: [1, 87, 0]

In [79]:
```
z=[1,1,23,23,44,565,87685,97]
```

In [80]:
```
z.drop(1)
```
```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-80-adf4dc428784> in <module>
----> 1 z.drop(1)

AttributeError: 'list' object has no attribute 'drop'
```

In [81]:
```
z
```
Out[81]: [1, 1, 23, 23, 44, 565, 87685, 97]

In [82]:
```
z.remove(23)
```

In [83]:
```
z
```
Out[83]: [1, 1, 23, 44, 565, 87685, 97]

In [84]:
```
z.remove(1)
```

In [85]:
```
z
```
Out[85]: [1, 23, 44, 565, 87685, 97]

In [86]:
```
z.pop(1)
```
Out[86]: 23

In [87]:
```
z.sort()
```

In [88]:
```
z
```
Out[88]: [1, 44, 97, 565, 87685]

In [89]:
```
z.sort(reverse=True)       #for sorting in descending order, as by default reverse is equal to False
```

In [90]:
```
z
```
Out[90]: [87685, 565, 97, 44, 1]

Tuples

They cannot be modified once created

In [91]:
```
lst = [10,20,30,40,50]
```

In [92]:
```
tup = tuple(lst)        #convert the list into tuple
```

In [93]:
```
type(tup)
```
Out[93]: tuple

In [94]:
```
tup
```
Out[94]: (10, 20, 30, 40, 50)

In [95]:
```
tup[0] =20
```
```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-95-2fe6b57b313b> in <module>
----> 1 tup[0] =20

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

In [96]:
```
for i in range(0,len(tup)):
    print(tup[i]**2)
```
```
100
400
900
1600
2500
```

In [97]:
```
for i in range(0,len(tup)):
    print(tup)
```
```
(10, 20, 30, 40, 50)
(10, 20, 30, 40, 50)
(10, 20, 30, 40, 50)
(10, 20, 30, 40, 50)
(10, 20, 30, 40, 50)
```

In [98]:
```
empty_lst=[]       #blank list
```

In [100]:
```
for i in range(0,len(tup)):        #conversion of tuples values into another list
    empty_lst.append(tup[i])
print(empty_lst)
```
```
[10, 20, 30, 40, 50]
```

In [101]:
```
empty_lst.pop(2)
```
Out[101]: 30

In [102]:
```
print(empty_lst)
```
```
[10, 20, 40, 50]
```

In [103]:
```
empty_lst[:3]
```
Out[103]: [10, 20, 40]

In [104]:
```
lst=[["SBI",454],["ONGC",549],["BOB",110],["RS",130]]
lst
```
Out[104]: [['SBI', 454], ['ONGC', 549], ['BOB', 110], ['RS', 130]]

In [105]:
```
type(lst)
```
Out[105]: list

In [107]:
```
import pandas as pd                       # importing panda library
```

In [108]:
```
stocks_df = pd.DataFrame(lst,columns=["Company","Price"])
```

In [109]:
```
type(stocks_df)
```
Out[109]: pandas.core.frame.DataFrame

In [110]:
```
stocks_df
```
Out[110]:

	Company	Price
0	SBI	454
1	ONGC	549
2	BOB	110
3	RS	130

In [111]:
```
stocks_df.head()
```
Out[111]:

	Company	Price
0	SBI	454
1	ONGC	549
2	BOB	110
3	RS	130

In [112]:
```
stocks_df.tail()
```
Out[112]:

	Company	Price
0	SBI	454
1	ONGC	549
2	BOB	110
3	RS	130

In [113]:
```
stocks_df.columns
```
Out[113]: Index(['Company', 'Price'], dtype='object')

In [114]:
```
stocks_df["Price"]
```
```
0    454
1    549
2    110
3    130
Name: Price, dtype: int64
```

In [115]:
```
stocks_df["Price"].sum()
```
Out[115]: 1243

Question

Print the price whose values are more that the median in the dataframe.

In [117]:
```
m=list(stocks_df["Price"])
m
```
Out[117]: [454, 549, 110, 130]

In [118]:
```
g=stocks_df["Price"].median()
for i in range(0,len(m)):
    if(m[i]>g):
        print(m[i])
```
```
454
549
```

In [119]:
```
# Using if else loop

prices = stocks_df['Price']
price_med = stocks_df['Price'].median()

for i in range(0,len(prices)):
    if prices[i] > price_med:
        print(prices[i])

    else:
        continue
```
```
454
549
```

In [120]:
```
stocks_df['Price'] > stocks_df['Price'].median()
```
```
0     True
1     True
2    False
3    False
Name: Price, dtype: bool
```

In [121]:
```
# In single line
stocks_df[stocks_df['Price'] > stocks_df['Price'].median()]
```
Out[121]:

	Company	Price
0	SBI	454
1	ONGC	549

In [122]:
```
blanks=[]
for i in range(0,len(stocks_df)):
    if(stocks_df.Price[i]>stocks_df.Price.median()):
        blanks.append(stocks_df.Price[i])
        blanks.append(stocks_df.Company[i])
```

In [124]:
```
print(blanks)
```
```
[454, 'SBI', 549, 'ONGC']
```

In [125]:
```
pd.DataFrame(blanks)
```
Out[125]:

	0
0	454
1	SBI
2	549
3	ONGC

Dictionary

In [126]:
```
j={"name":"Shobhit","Marks":89}
```

In [127]:
```
j
```
Out[127]: {'name': 'Shobhit', 'Marks': 89}

In [128]:
```
pd.DataFrame([j])
```
Out[128]:

	name	Marks
0	Shobhit	89

In [129]:
```
type(j)
```
Out[129]: dict

In [130]:
```
len(j)
```
Out[130]: 2

In [131]:
```
k={"name":"Future_Minds","Student":15,"mode":"classroom"}
```

In [132]:
```
pd.DataFrame([k])
```
Out[132]:

	name	Student	mode
0	Future_Minds	15	classroom

In [133]:
```
k={"name":["Future","Master", "Minds"],"car":9095,"mobile":"nokia"}
```

In [134]:
```
pd.DataFrame([k])
```
Out[134]:

	name	car	mobile
0	[Future, Master, Minds]	9095	nokia

In [135]:
```
f=k.copy()
```

In [136]:
```
k.clear()
```

In [137]:
```
f
```
Out[137]: {'name': ['Future', 'Master', 'Minds'], 'car': 9095, 'mobile': 'nokia'}

In [138]:
```
pd.DataFrame([f])
```
Out[138]:

	name	car	mobile
0	[Future, Master, Minds]	9095	nokia

In [140]:
```
f.keys()
```
Out[140]: dict_keys(['name', 'car', 'mobile'])

In [141]:
```
f.values()
```
Out[141]: dict_values([['Future', 'Master', 'Minds'], 9095, 'nokia'])

In [142]:
```
#dictionary.get() extracts the value from     #solve it
print("Name is :",f.get("name"))
```
```
Name is : ['Future', 'Master', 'Minds']
```

In [148…

```
# Imputation
newd = {"phone. no.":98123}
f.update(newd)
```

In [149…

```
f
```

Out[149…

```
{'name': ['Future', 'Master', 'Minds'],
 'car': 9095,
 'mobile': 'nokia',
 'phone. no.': 98123}
```

In [150…

```
f.keys()
```

Out[150…

```
dict_keys(['name', 'car', 'mobile', 'phone. no.'])
```

In [151…

```
f.values()
```

Out[151…

```
dict_values([['Future', 'Master', 'Minds'], 9095, 'nokia', 98123])
```

** HAPPY LEARNING**

In [ ]:

```
```