Note:- Syntax reference:

https://www.geeksforgeeks.org/python-programming-language/

Consider the following Python dictionary data and Python list labels:

data = {'birds': ['Cranes', 'Cranes', 'plovers', 'spoonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills', 'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4], 'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'no', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']}

labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

1. Create a DataFrame birds from this dictionary data which has the index labels.

In [338]:

```
import pandas as pd, numpy as np
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']}, index=labels)
print('Data frame with the index - labels')
data_birds
```

```
Data frame with the index - labels
```

Out[338]:

	birds	age	visits	priority
a	Cranes	3.5	2	yes
b	Cranes	4.0	4	yes
c	plovers	1.5	3	no
d	spoonbills	NaN	4	yes

	birds	age	visits	priority
e	spoonbills	6.0	3	no
f	Cranes	3.0	4	no
g	plovers	5.5	2	no
h	Cranes	NaN	2	yes
i	spoonbills	8.0	3	no
j	spoonbills	4.0	2	no

2. Display a summary of the basic information about birds DataFrame and its data.

In [339]:

```
import pandas as pd, numpy as np
labels = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
print('Summary of the basic information about birds DataFrame and its d
ata.')
print(data_birds.describe())
```

```
Summary of the basic information about birds DataFrame and its data.
            age     visits
count  8.000000  10.000000
mean   4.437500   2.900000
std    2.007797   0.875595
min    1.500000   2.000000
25%    3.375000   2.000000
50%    4.000000   3.000000
75%    5.625000   3.750000
max    8.000000   4.000000
```

3. Print the first 2 rows of the birds dataframe

In [340]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
print('The first 2 rows of the birds dataframe.')
data_birds.iloc[:2]
```

```
The first 2 rows of the birds dataframe.
```

Out[340]:

	birds	age	visits	priority
0	Cranes	3.5	2	yes
1	Cranes	4.0	4	yes

4. Print all the rows with only 'birds' and 'age' columns from the dataframe

In [342]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
print('All the rows with only \'birds\' and \'age\' columns from the da
taframe.')
data_birds[['birds', 'age']]
```

```
All the rows with only 'birds' and 'age' columns from the dataframe.
```

Out[342]:

	birds	age
0	Cranes	3.5
1	Cranes	4.0
2	plovers	1.5
3	spoonbills	NaN
4	spoonbills	6.0

	birds	age
5	Cranes	3.0
6	plovers	5.5
7	Cranes	NaN
8	spoonbills	8.0
9	spoonbills	4.0

5. select [2, 3, 7] rows and in columns ['birds', 'age', 'visits']

In [345]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
a = data_birds[['birds', 'age', 'visits']]
print("Rows 2, 3, 7 and columns \'birds\', \'age\', \'visits\' of the d
ata frame.")
a.iloc[[2,3,7]]
#data_birds.iloc[[2,3,7], [0,1,2]]
```

```
Rows 2, 3, 7 and columns 'birds', 'age', 'visits' of the data frame.
```

Out[345]:

	birds	age	visits
2	plovers	1.5	3
3	spoonbills	NaN	4
7	Cranes	NaN	2

6. select the rows where the number of visits is less than 4

In [346]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
print('The rows where the number of visits is less than 4.')
data_birds[data_birds['visits'] < 4]
```

```
The rows where the number of visits is less than 4.
```

Out[346]:

	birds	age	visits	priority
0	Cranes	3.5	2	yes
2	plovers	1.5	3	no
4	spoonbills	6.0	3	no
6	plovers	5.5	2	no
7	Cranes	NaN	2	yes
8	spoonbills	8.0	3	no
9	spoonbills	4.0	2	no

7. select the rows with columns ['birds', 'visits'] where the age is missing i.e NaN

In [347]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
db=data_birds[data_birds['age'].isnull()]
print('The rows with columns \'birds\', \'visits\' where the age is mis
sing.')
db[['birds','visits']]
```

```
The rows with columns 'birds', 'visits' where the age is missing.
```

Out[347]:

	birds	visits
3	spoonbills	4

	birds	visits
7	Cranes	2

8. Select the rows where the birds is a Cranes and the age is less than 4

In [337]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
db = data_birds[data_birds['age'] < 4]
db1 = db.groupby('birds')
print('Below are the rows where the bird is Crane and the age is less t
han 4.')
db1.get_group('Cranes')
```

```
Below are the rows where the bird is Crane and the age is less than 4.
```

Out[337]:

	birds	age	visits	priority
0	Cranes	3.5	2	yes
5	Cranes	3.0	4	no

9. Select the rows the age is between 2 and 4(inclusive)

In [335]:

```
print('Below are the rows where the age of the birds is between 2 and 4
 (inclusive).')
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
data_birds[(data_birds['age'] >= 2) & (data_birds['age'] <= 4)]
```

```
Below are the rows where the age of the birds is between 2 and 4 (inclu
```

sive).

Out[335]:

	birds	age	visits	priority
0	Cranes	3.5	2	yes
1	Cranes	4.0	4	yes
5	Cranes	3.0	4	no
9	spoonbills	4.0	2	no

10. Find the total number of visits of the bird Cranes

In [333]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'sp
oonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills',
'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4],
'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'n
o', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
print("Total number of visits of the bird Cranes")
data_birds.groupby('birds').get_group('Cranes').sum()['visits']
```

```
Total number of visits of the bird Cranes
```

Out[333]:

```
12
```

11. Calculate the mean age for each different birds in dataframe.

In [325]:

```
g = data_birds.groupby('birds')
print('Mean of each kind of nird in the data frame:')
for b, b_d in g:
    print(b)
    print(b_d.mean()['age'])
```

```
Mean of each kind of nird in the data frame:
plovers
3.5
spoonbills
6.0
```

```
trumpeters
3.5
```

12. Append a new row 'k' to dataframe with your choice of values for each column. Then delete that row to return the original DataFrame.

In [323]:

```
N_d = pd.DataFrame({
  "birds":["Eagle"],
    "age":[2.5],
    "visits":[5],
    "priority":"yes"
}, index = ['k'])
N1_d = pd.concat([data_birds,N_d])
print('Data frame with the new row \'k\' appended to the original data frame.')
print(N1_d)
print('------------------------------------------------------------------')
print('Original data frame after deleting the new row \'k\'.')
N1_d=N1_d.drop(['k'],axis=0)
print(N1_d)
```

```
Data frame with the new row 'k' appended to the original data frame.
        birds  age  visits priority
0  trumpeters  3.5       2      yes
1  trumpeters  4.0       4      yes
2     plovers  1.5       3       no
3  spoonbills  NaN       4      yes
4  spoonbills  6.0       3       no
5  trumpeters  3.0       4       no
6     plovers  5.5       2       no
7  trumpeters  NaN       2      yes
8  spoonbills  8.0       3       no
9  spoonbills  4.0       2       no
k       Eagle  2.5       5      yes
------------------------------------------------------------------
Original data frame after deleting the new row 'k'.
        birds  age  visits priority
```

```
0  trumpeters  3.5       2      yes
1  trumpeters  4.0       4      yes
2     plovers  1.5       3       no
3  spoonbills  NaN       4      yes
4  spoonbills  6.0       3       no
5  trumpeters  3.0       4       no
6     plovers  5.5       2       no
7  trumpeters  NaN       2      yes
8  spoonbills  8.0       3       no
9  spoonbills  4.0       2       no
```

13. Find the number of each type of birds in dataframe (Counts)

In [311]:

```
print('Number of birds in each kind:')
print(data_birds.groupby('birds').size())
```

```
Number of birds in each kind:
birds
plovers       2
spoonbills    4
trumpeters    4
dtype: int64
```

14. Sort dataframe (birds) first by the values in the 'age' in decending order, then by the value in the 'visits' column in ascending order.

In [317]:

```
print('Data frame sorted by \'age\'')
print(data_birds.sort_values('age', ascending = True))
print('--------------------------------------------------')
print('Data frame sorted by \'visits\'')
print(data_birds.sort_values('visits', ascending = False))
```

```
Data frame sorted by 'age'
        birds  age  visits priority
2     plovers  1.5       3       no
5  trumpeters  3.0       4       no
0  trumpeters  3.5       2      yes
```

```
1  trumpeters  4.0       4      yes
9  spoonbills  4.0       2       no
6     plovers  5.5       2       no
4  spoonbills  6.0       3       no
8  spoonbills  8.0       3       no
3  spoonbills  NaN       4      yes
7  trumpeters  NaN       2      yes
--------------------------------------------------
Data frame sorted by 'visits'
        birds  age  visits priority
1  trumpeters  4.0       4      yes
3  spoonbills  NaN       4      yes
5  trumpeters  3.0       4       no
2     plovers  1.5       3       no
4  spoonbills  6.0       3       no
8  spoonbills  8.0       3       no
0  trumpeters  3.5       2      yes
6     plovers  5.5       2       no
7  trumpeters  NaN       2      yes
9  spoonbills  4.0       2       no
```

15. Replace the priority column values with'yes' should be 1 and 'no' should be 0

In [307]:

```
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'spoonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills', 'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4], 'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'no', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
data_birds['priority'] = data_birds['priority'].replace(['yes','no'], [1,0])
print(data_birds)
```

```
        birds  age  visits  priority
0      Cranes  3.5       2         1
1      Cranes  4.0       4         1
2     plovers  1.5       3         0
3  spoonbills  NaN       4         1
4  spoonbills  6.0       3         0
```

```
5      Cranes  3.0       4         0
6     plovers  5.5       2         0
7      Cranes  NaN       2         1
8  spoonbills  8.0       3         0
9  spoonbills  4.0       2         0
```

16. In the 'birds' column, change the 'Cranes' entries to 'trumpeters'.

In [308]:

```python
data_birds = pd.DataFrame({'birds': ['Cranes', 'Cranes', 'plovers', 'spoonbills', 'spoonbills', 'Cranes', 'plovers', 'Cranes', 'spoonbills', 'spoonbills'], 'age': [3.5, 4, 1.5, np.nan, 6, 3, 5.5, np.nan, 8, 4], 'visits': [2, 4, 3, 4, 3, 4, 2, 2, 3, 2], 'priority': ['yes', 'yes', 'no', 'yes', 'no', 'no', 'no', 'yes', 'no', 'no']})
data_birds['birds'] = data_birds['birds'].replace('Cranes', 'trumpeters')
print(data_birds)
```

```
        birds  age  visits priority
0  trumpeters  3.5       2      yes
1  trumpeters  4.0       4      yes
2     plovers  1.5       3       no
3  spoonbills  NaN       4      yes
4  spoonbills  6.0       3       no
5  trumpeters  3.0       4       no
6     plovers  5.5       2       no
7  trumpeters  NaN       2      yes
8  spoonbills  8.0       3       no
9  spoonbills  4.0       2       no
```