

```
In [1]: import pandas as pd
import numpy as np

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']

df = pd.DataFrame(data, index = labels)

print(df)
```

	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
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

```
In [2]: birds_df = pd.DataFrame(data['birds'], index = labels)
print(birds_df)
```

	0
a	Cranes
b	Cranes
c	plovers
d	spoonbills
e	spoonbills

```
f    Cranes
g    plovers
h    Cranes
i    spoonbills
j    spoonbills
```

```
In [3]: print(birds_df.describe())
```

```
count      0
unique      3
top    spoonbills
freq        4
```

```
In [4]: print(birds_df[:2])
```

```
0
a  Cranes
b  Cranes
```

```
In [5]: df[['birds', 'age']]
```

Out[5]:

	birds	age
a	Cranes	3.5
b	Cranes	4.0
c	plovers	1.5
d	spoonbills	NaN
e	spoonbills	6.0
f	Cranes	3.0
g	plovers	5.5
h	Cranes	NaN
i	spoonbills	8.0

	birds	age
j	spoonbills	4.0

```
In [6]: bird_age_visits_df = df[['birds', 'age', 'visits']]
        #print(bird_age_visits_df)
        bird_age_visits_df.iloc[['2', '3', '7']]
```

Out[6]:

	birds	age	visits
c	plovers	1.5	3
d	spoonbills	NaN	4
h	Cranes	NaN	2

```
In [7]: select_visits = df.loc[df.visits < 4]
        print(select_visits)
```

	birds	age	visits	priority
a	Cranes	3.5	2	yes
c	plovers	1.5	3	no
e	spoonbills	6.0	3	no
g	plovers	5.5	2	no
h	Cranes	NaN	2	yes
i	spoonbills	8.0	3	no
j	spoonbills	4.0	2	no

```
In [8]: df.loc[(df.birds=='Cranes') & (df.age < 4)]
```

Out[8]:

	birds	age	visits	priority
a	Cranes	3.5	2	yes
f	Cranes	3.0	4	no

```
In [9]: df.loc[(df.age >= 2) & (df.age <= 4)]
```

Out[9]:

	birds	age	visits	priority
a	Cranes	3.5	2	yes
b	Cranes	4.0	4	yes
f	Cranes	3.0	4	no
j	spoonbills	4.0	2	no

```
In [10]: group_birds = df.groupby('birds')
cranes = group_birds.get_group('Cranes')
#print(cranes)
print(cranes.visits.sum())
```

12

```
In [11]: print(group_birds.age.mean())
```

```
birds
Cranes      3.5
plovers     3.5
spoonbills  6.0
Name: age, dtype: float64
```

```
In [12]: df.loc['k'] = ['plovers', '3.5', '5', 'yes']
print("New DF with row K \n\n", df)
df.drop('k', inplace = True)
print("\n Without row K \n", df)
```

New DF with row K

	birds	age	visits	priority
(a,)	Cranes	3.5	2	yes
(b,)	Cranes	4	4	yes
(c,)	plovers	1.5	3	no
(d,)	spoonbills	NaN	4	yes
(e,)	spoonbills	6	3	no
(f,)	Cranes	3	4	no
(g,)	plovers	5.5	2	no

(h,)	Cranes	NaN	2	yes
(i,)	spoonbills	8	3	no
(j,)	spoonbills	4	2	no
k	plovers	3.5	5	yes

Without row K

	birds	age	visits	priority
(a,)	Cranes	3.5	2	yes
(b,)	Cranes	4	4	yes
(c,)	plovers	1.5	3	no
(d,)	spoonbills	NaN	4	yes
(e,)	spoonbills	6	3	no
(f,)	Cranes	3	4	no
(g,)	plovers	5.5	2	no
(h,)	Cranes	NaN	2	yes
(i,)	spoonbills	8	3	no
(j,)	spoonbills	4	2	no

```
In [13]: Cranes_obj = df.apply(lambda x: True if x['birds'] == 'Cranes' else False, axis=1)
count1 = len(Cranes_obj[Cranes_obj == True].index)
print("No of Cranes = ",count1)

plovers_obj = df.apply(lambda x: True if x['birds'] == 'plovers' else False, axis=1)
count2 = len(plovers_obj[plovers_obj == True].index)
print("No of plovers = ",count2)

spoonbills_obj = df.apply(lambda x: True if x['birds'] == 'spoonbills' else False, axis=1)
count3 = len(spoonbills_obj[spoonbills_obj == True].index)
print("No of spoonbills = ",count3)

No of Cranes = 4
No of plovers = 2
No of spoonbills = 4
```

```
In [14]: sort_by_birds = df.sort_values("age",ascending = False )
print("Sort by Birds Column ::\n\n", sort_by_birds)
```

```
sort_by_visits = df.sort_values("visits")
print("\n Sort by Visits Column ::\n\n",sort_by_visits)
```

Sort by Birds Column ::

	birds	age	visits	priority
(i,)	spoonbills	8	3	no
(e,)	spoonbills	6	3	no
(g,)	plovers	5.5	2	no
(b,)	Cranes	4	4	yes
(j,)	spoonbills	4	2	no
(a,)	Cranes	3.5	2	yes
(f,)	Cranes	3	4	no
(c,)	plovers	1.5	3	no
(d,)	spoonbills	NaN	4	yes
(h,)	Cranes	NaN	2	yes

Sort by Visits Column ::

	birds	age	visits	priority
(a,)	Cranes	3.5	2	yes
(g,)	plovers	5.5	2	no
(h,)	Cranes	NaN	2	yes
(j,)	spoonbills	4	2	no
(c,)	plovers	1.5	3	no
(e,)	spoonbills	6	3	no
(i,)	spoonbills	8	3	no
(b,)	Cranes	4	4	yes
(d,)	spoonbills	NaN	4	yes
(f,)	Cranes	3	4	no

```
In [15]: df.replace(to_replace = ["yes","no"], value = [1,0])
```

Out[15]:

	birds	age	visits	priority
(a,)	Cranes	3.5	2	1
(b,)	Cranes	4.0	4	1
(c,)	plovers	1.5	3	0

	birds	age	visits	priority
(d,)	spoonbills	NaN	4	1
(e,)	spoonbills	6.0	3	0
(f,)	Cranes	3.0	4	0
(g,)	plovers	5.5	2	0
(h,)	Cranes	NaN	2	1
(i,)	spoonbills	8.0	3	0
(j,)	spoonbills	4.0	2	0

In [16]: `df.replace(to_replace = "Cranes",value = 'trumpeters')`

Out[16]:

	birds	age	visits	priority
(a,)	trumpeters	3.5	2	yes
(b,)	trumpeters	4.0	4	yes
(c,)	plovers	1.5	3	no
(d,)	spoonbills	NaN	4	yes
(e,)	spoonbills	6.0	3	no
(f,)	trumpeters	3.0	4	no
(g,)	plovers	5.5	2	no
(h,)	trumpeters	NaN	2	yes
(i,)	spoonbills	8.0	3	no
(j,)	spoonbills	4.0	2	no

In []: