

In [8]:

```
import pandas as pd
dictionary={'XII':25,'XI':30,'X':50}
series = pd.Series(dictionary)
#sorted using keys
dic={}
for i in sorted(dictionary):
    dic[i]=dictionary[i]
print(dic)
#sorted using values
dic2=dict(sorted(dictionary.items(),key=lambda x:x[1]))
print(dic2)
```

```
{'X': 50, 'XI': 30, 'XII': 25}
{'XII': 25, 'XI': 30, 'X': 50}
```

In [21]:

```
import numpy as np
import math
b=[]
for i in np.random.rand(10,1):
    b.append(int(math.ceil(i[0]*10)))
print(b)
print(max(b),b.index(max(b)))
print(min(b),b.index(min(b)))
```

```
[4, 8, 6, 8, 7, 6, 1, 6, 6, 10]
10 9
1 6
```

In [31]:

```
import pandas as pd
dict2=[25,40,18,27,32,39,28]
dict3=['Monday','Tuesday','Wednesday','Thursday','Friday','saturday','sunday']
series = pd.Series(dict2,index=dict3)
print(series[3])
print(series['Tuesday'])
```

```
27
40
```

In [50]:

```
import numpy as np
import pandas as pd
df = pd.DataFrame(np.random.randint(1,10, size=(30,6)))
length = len(df)
num = int(0.2*length)
idx_replace = np.random.randint(0, length-1, num)
idx_replace1= np.random.randint(0, 5, num)
df.loc[idx_replace, idx_replace1] = np.nan
print(df)
print(df.isnull().sum())
df2=df.dropna(thresh=4)
print(df2)
normalized_df=(df2-df2.min())/(df2.max()-df2.min())
print(normalized_df)
```

	0	1	2	3	4	5
0	9.0	3.0	2	6.0	8.0	6
1	7.0	9.0	7	2.0	7.0	7
2	5.0	8.0	3	1.0	1.0	6
3	5.0	8.0	8	2.0	8.0	4
4	8.0	7.0	3	2.0	9.0	9
5	NaN	NaN	5	NaN	NaN	1
6	8.0	4.0	9	8.0	7.0	5
7	9.0	6.0	9	2.0	2.0	9
8	3.0	8.0	8	5.0	1.0	8
9	NaN	NaN	8	NaN	NaN	5
10	2.0	8.0	5	4.0	9.0	6
11	9.0	8.0	4	1.0	6.0	5
12	9.0	8.0	3	5.0	8.0	7
13	NaN	NaN	9	NaN	NaN	8
14	3.0	5.0	9	4.0	6.0	7
15	2.0	1.0	9	2.0	7.0	4
16	3.0	1.0	9	3.0	3.0	6
17	1.0	2.0	7	7.0	4.0	1
18	9.0	2.0	9	9.0	1.0	3
19	5.0	3.0	5	5.0	1.0	1
20	6.0	9.0	1	6.0	9.0	2
21	2.0	5.0	2	7.0	9.0	8
22	3.0	8.0	1	5.0	2.0	9
23	2.0	7.0	9	9.0	3.0	7
24	NaN	NaN	3	NaN	NaN	3
25	8.0	9.0	3	3.0	4.0	4
26	8.0	4.0	6	1.0	5.0	4
27	1.0	1.0	5	9.0	8.0	1
28	8.0	4.0	9	1.0	4.0	8
29	6.0	4.0	8	7.0	6.0	3

0 4

1 4

2 0

3 4

4 4

5 0

dtype: int64

	0	1	2	3	4	5
0	9.0	3.0	2	6.0	8.0	6
1	7.0	9.0	7	2.0	7.0	7
2	5.0	8.0	3	1.0	1.0	6
3	5.0	8.0	8	2.0	8.0	4
4	8.0	7.0	3	2.0	9.0	9
6	8.0	4.0	9	8.0	7.0	5
7	9.0	6.0	9	2.0	2.0	9
8	3.0	8.0	8	5.0	1.0	8
10	2.0	8.0	5	4.0	9.0	6
11	9.0	8.0	4	1.0	6.0	5
12	9.0	8.0	3	5.0	8.0	7
14	3.0	5.0	9	4.0	6.0	7
15	2.0	1.0	9	2.0	7.0	4
16	3.0	1.0	9	3.0	3.0	6
17	1.0	2.0	7	7.0	4.0	1
18	9.0	2.0	9	9.0	1.0	3
19	5.0	3.0	5	5.0	1.0	1
20	6.0	9.0	1	6.0	9.0	2
21	2.0	5.0	2	7.0	9.0	8
22	3.0	8.0	1	5.0	2.0	9
23	2.0	7.0	9	9.0	3.0	7
25	8.0	9.0	3	3.0	4.0	4

26	8.0	4.0	6	1.0	5.0	4
27	1.0	1.0	5	9.0	8.0	1
28	8.0	4.0	9	1.0	4.0	8
29	6.0	4.0	8	7.0	6.0	3
	0	1	2	3	4	5
0	1.000	0.250	0.125	0.625	0.875	0.625
1	0.750	1.000	0.750	0.125	0.750	0.750
2	0.500	0.875	0.250	0.000	0.000	0.625
3	0.500	0.875	0.875	0.125	0.875	0.375
4	0.875	0.750	0.250	0.125	1.000	1.000
6	0.875	0.375	1.000	0.875	0.750	0.500
7	1.000	0.625	1.000	0.125	0.125	1.000
8	0.250	0.875	0.875	0.500	0.000	0.875
10	0.125	0.875	0.500	0.375	1.000	0.625
11	1.000	0.875	0.375	0.000	0.625	0.500
12	1.000	0.875	0.250	0.500	0.875	0.750
14	0.250	0.500	1.000	0.375	0.625	0.750
15	0.125	0.000	1.000	0.125	0.750	0.375
16	0.250	0.000	1.000	0.250	0.250	0.625
17	0.000	0.125	0.750	0.750	0.375	0.000
18	1.000	0.125	1.000	1.000	0.000	0.250
19	0.500	0.250	0.500	0.500	0.000	0.000
20	0.625	1.000	0.000	0.625	1.000	0.125
21	0.125	0.500	0.125	0.750	1.000	0.875
22	0.250	0.875	0.000	0.500	0.125	1.000
23	0.125	0.750	1.000	1.000	0.250	0.750
25	0.875	1.000	0.250	0.250	0.375	0.375
26	0.875	0.375	0.625	0.000	0.500	0.375
27	0.000	0.000	0.500	1.000	0.875	0.000
28	0.875	0.375	1.000	0.000	0.375	0.875
29	0.625	0.375	0.875	0.750	0.625	0.250

In [52]:

```
import numpy as np
import pandas as pd
df_A = pd.DataFrame({'Name': ['Aroma', 'Kiran', 'Rayan', 'Rohan', 'Amit', 'Yash', 'Mona', 'Kartik', 'Kavita', 'Pooja'],
                     'Percentage': [79.5, 29.0, 90.5, np.nan, 32.0, 65.0, 56.0, np.nan, 29.0, 89.0],
                     'Qualify': ['yes', 'no', 'yes', 'no', 'no', 'yes', 'yes', np.nan, 'no', np.nan]})
df_B = pd.DataFrame({'Name': ['Parveen', 'Ahil', 'Shaila', 'Shruti', 'Mark'],
                     'Percentage': [89.5, 92.0, 90.5, 91.5, 90],
                     'Qualify': ['yes', 'yes', 'yes', 'yes', 'yes']})
print(df_A)
print(df_B)
```

	Name	Percentage	Qualify
0	Aroma	79.5	yes
1	Kiran	29.0	no
2	Rayan	90.5	yes
3	Rohan	NaN	no
4	Amit	32.0	no
5	Yash	65.0	yes
6	Mona	56.0	yes
7	Kartik	NaN	NaN
8	Kavita	29.0	no
9	Pooja	89.0	NaN

	Name	Percentage	Qualify
0	Parveen	89.5	yes
1	Ahil	92.0	yes
2	Shaila	90.5	yes
3	Shruti	91.5	yes
4	Mark	90.0	yes

In [69]:

```
import numpy as np
import pandas as pd
df5 = pd.read_csv("C:\\Users\\SU20181941\\Downloads\\penguins_lter.csv")
print(df5)
print(df5.count())
print(df5.index,df5.dtypes)
print(df5.head(5))
print(df5.tail(5))
bk=df5["Sample Number"]
print(bk[2])
print(bk[3])
print(df5.describe())
print(df5.corr(min_periods=3))
```

Out[69]:

	Sample Number	Culmen Length (mm)	Culmen Depth (mm)	Flipper Length (mm)	Body Mass (g)	Delta 15 N (o/oo)	Delta 13 C (o/oo)
Sample Number	1.000000	-0.236356	-0.022352	0.040849	-0.007042	0.006952	-0.488690
Culmen Length (mm)	-0.236356	1.000000	-0.235053	0.656181	0.595110	-0.059759	0.189025
Culmen Depth (mm)	-0.022352	-0.235053	1.000000	-0.583851	-0.471916	0.605874	0.429933
Flipper Length (mm)	0.040849	0.656181	-0.583851	1.000000	0.871202	-0.507787	-0.376223
Body Mass (g)	-0.007042	0.595110	-0.471916	0.871202	1.000000	-0.537888	-0.374638
Delta 15 N (o/oo)	0.006952	-0.059759	0.605874	-0.507787	-0.537888	1.000000	0.570615
Delta 13 C (o/oo)	-0.488690	0.189025	0.429933	-0.376223	-0.374638	0.570615	1.000000

In []: