

# Basic Analysis using numpy and pandas

## Drug Dataset

To import library

In [1]:

```
import numpy as np
```

In [2]:

```
import pandas as pd
```

To import dataset

In [3]:

```
d=pd.read_csv(r"C:\Users\user\Downloads\drug.csv")
d
```

Out[3]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

To get Top 10 record

In [4]:

```
d.head(10)
```

Out[4]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
5	22	F	NORMAL	HIGH	8.607	drugX
6	49	F	NORMAL	HIGH	16.275	drugY
7	41	M	LOW	HIGH	11.037	drugC
8	60	M	NORMAL	HIGH	15.171	drugY
9	43	M	LOW	NORMAL	19.368	drugY

To get last record

In [5]:

```
d.tail(20)
```

Out[5]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
180	22	F	HIGH	NORMAL	22.818	drugY
181	59	F	NORMAL	HIGH	13.884	drugX
182	20	F	LOW	NORMAL	11.686	drugX
183	36	F	HIGH	NORMAL	15.490	drugY
184	18	F	HIGH	HIGH	37.188	drugY
185	57	F	NORMAL	NORMAL	25.893	drugY
186	70	M	HIGH	HIGH	9.849	drugB
187	47	M	HIGH	HIGH	10.403	drugA
188	65	M	HIGH	NORMAL	34.997	drugY
189	64	M	HIGH	NORMAL	20.932	drugY
190	58	M	HIGH	HIGH	18.991	drugY
191	23	M	HIGH	HIGH	8.011	drugA
192	72	M	LOW	HIGH	16.310	drugY
193	72	M	LOW	HIGH	6.769	drugC
194	46	F	HIGH	HIGH	34.686	drugY
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

Statistics Analysis

In [6]:

```
d.describe()
```

Out[6]:

	Age	Na_to_K
<b>count</b>	200.000000	200.000000
<b>mean</b>	44.315000	16.084485
<b>std</b>	16.544315	7.223956
<b>min</b>	15.000000	6.269000
<b>25%</b>	31.000000	10.445500
<b>50%</b>	45.000000	13.936500
<b>75%</b>	58.000000	19.380000
<b>max</b>	74.000000	38.247000

To get row and column

In [7]:

```
np.shape(d)
```

Out[7]:

(200, 6)

Find Number of Elements

In [8]:

```
np.size(d)
```

Out[8]:

1200

Find Missing Value

In [9]:

```
d.isna()
```

Out[9]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...	...	...	...	...	...	...
195	False	False	False	False	False	False
196	False	False	False	False	False	False
197	False	False	False	False	False	False
198	False	False	False	False	False	False
199	False	False	False	False	False	False

200 rows × 6 columns

To drop the missing value

In [10]:

```
d.dropna(axis=1,how="any")
```

Out[10]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

In [11]:

```
d["Age"]
```

Out[11]:

0 23

1 47

2 47

3 28

4 61

..

195 56

196 16

197 52

198 23

199 40

Name: Age, Length: 200, dtype: int64

In [12]:

```
data=d[['Age','Na_to_K']]  
data
```

Out[12]:

	Age	Na_to_K
0	23	25.355
1	47	13.093
2	47	10.114
3	28	7.798
4	61	18.043
...	...	...
195	56	11.567
196	16	12.006
197	52	9.894
198	23	14.020
199	40	11.349

200 rows × 2 columns

In [13]:

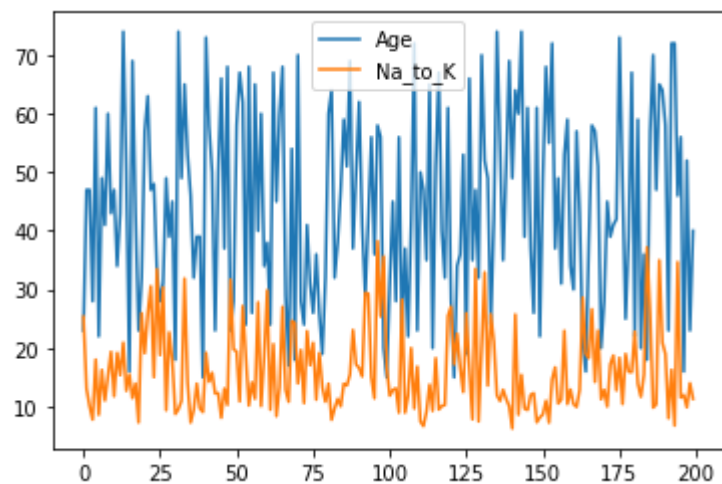
```
import matplotlib.pyplot as pp
```

In [14]:

```
data.plot.line()
```

Out[14]:

<AxesSubplot:>

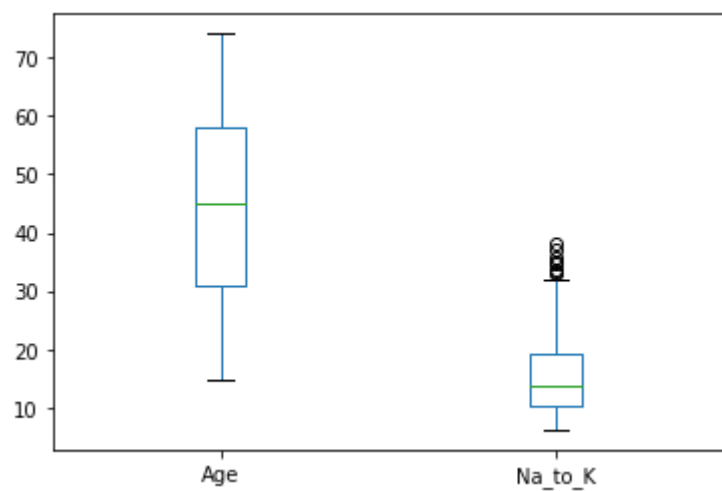


In [15]:

```
data.plot.box()
```

Out[15]:

<AxesSubplot:>

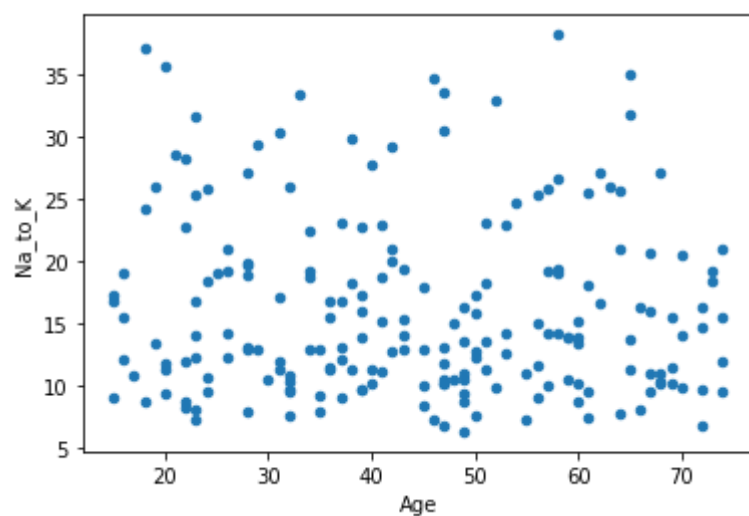


In [16]:

```
data.plot.scatter(x='Age',y='Na_to_K')
```

Out[16]:

&lt;AxesSubplot:xlabel='Age', ylabel='Na\_to\_K'&gt;

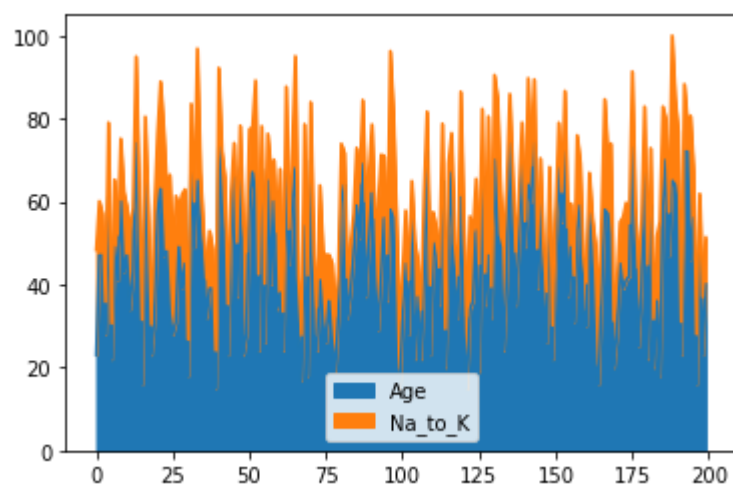


In [17]:

```
data.plot.area()
```

Out[17]:

&lt;AxesSubplot:&gt;



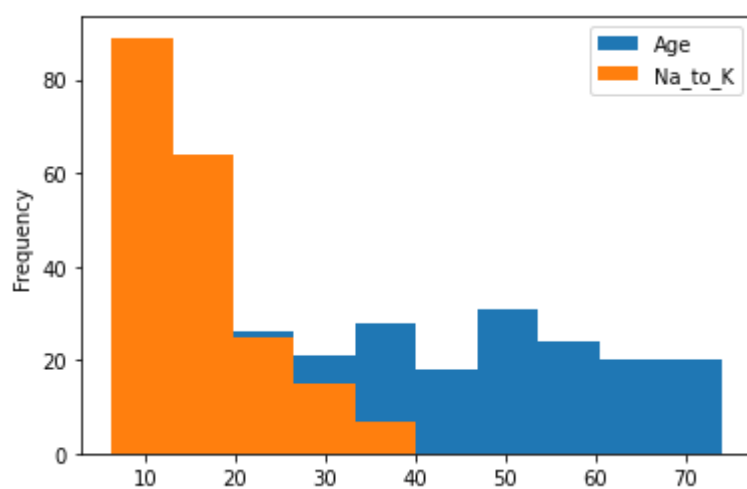


In [18]:

```
data.plot.hist()
```

Out[18]:

&lt;AxesSubplot:ylabel='Frequency'&gt;



In [19]:

```
d.plot.pie(y="Na_to_K")
```

Out[19]:

&lt;AxesSubplot:ylabel='Na\_to\_K'&gt;

