# Deena 20104016

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as pp
```

# Problem Statement

# LINEAR REGRESSION

```
In [2]: a = pd.read_csv("drug.csv")
```

Out[2]:

|  | 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

# HEAD

In [3]:

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 |

# Data Cleaning and Preprocessing

In [4]:

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 |

In [5]:

Out[5]:

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

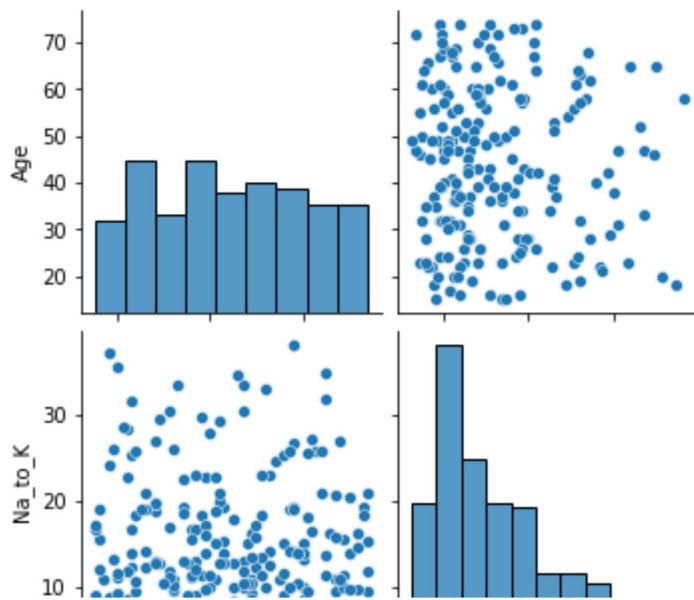# To display heading

In [6]:

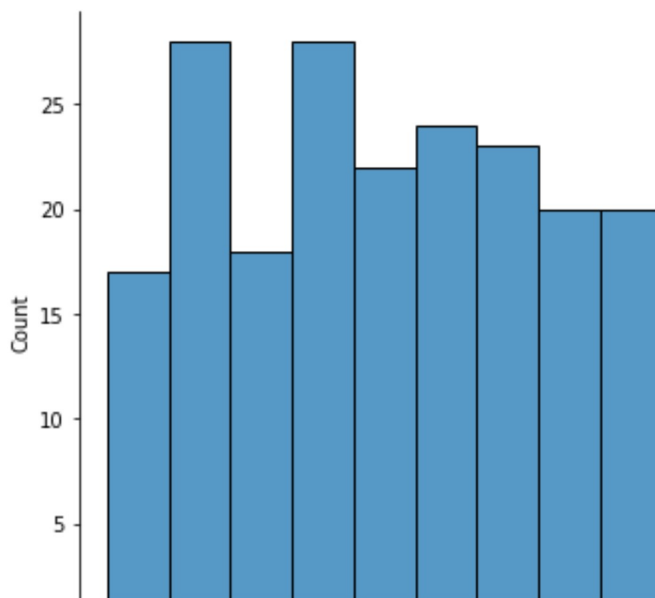Out[6]: Index(['Age', 'Sex', 'BP', 'Cholesterol', 'Na_to_K', 'Drug'], dtype='object')
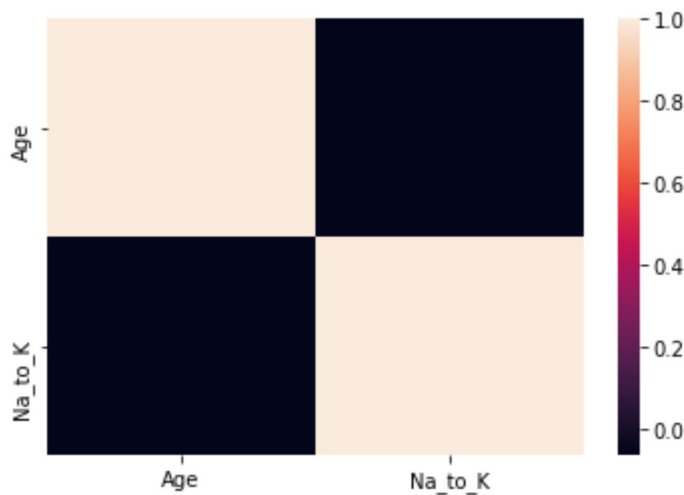
In [7]:

Out[7]: &lt;seaborn.axisgrid.PairGrid at 0x192b5696970&gt;



In [8]:

Out[8]: &lt;seaborn.axisgrid.FacetGrid at 0x192b6fc7130&gt;

In [9]:

Out[9]: `<AxesSubplot:>`



# TO TRAIN THE MODEL - MODEL BUILDING

In [10]:
```python
x = a[['Age']]
```

In [11]:
```python
# to split my dataset into training and test data
from sklearn.model_selection import train_test_split
```

In [12]:
```python
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
```
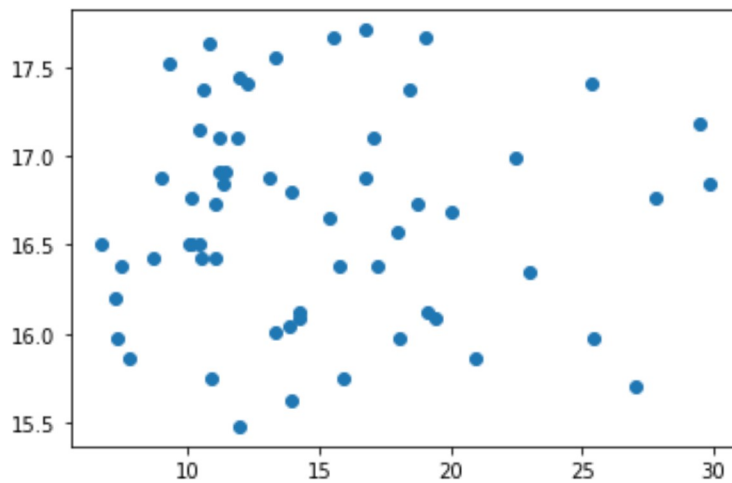
Out[12]: `LinearRegression()`

In [13]:
```python
coeff = pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
```

Out[13]:

|  | Co-efficient |
|---|---|
| **Age** | -0.037787 |

```
In [14]: prediction= lr.predict(x_test)
```

Out[14]: <matplotlib.collections.PathCollection at 0x192b7900970>



```
In [15]:
```

Out[15]: -0.09664460424969046

## LASSO & RIDGE

```
In [16]: from sklearn.linear_model import Ridge,Lasso
         rr=Ridge(alpha=10)
```

Out[16]: Ridge(alpha=10)

```
In [17]:
```

Out[17]: -0.09663881138644626

```
In [18]: la=Lasso(alpha=10)
```

Out[18]: Lasso(alpha=10)

```
In [19]:
```

Out[19]: -0.08419813862708425