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("Fitness.csv")
```

Out[2]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

HEAD

In [3]:

Out[3]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179

Data Cleaning and Preprocessing

In [4]:

Out[4]:

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179

In [5]:

Out[5]:

	Sum of Total Sales
count	9.000000
mean	255.555556
std	337.332963
min	75.000000
25%	127.000000
50%	167.000000
75%	171.000000
max	1150.000000

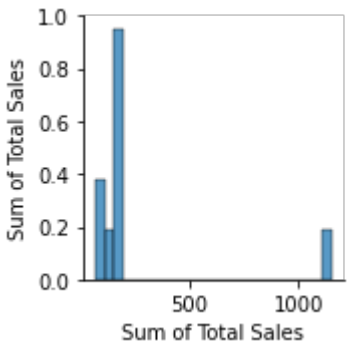
To display heading

In [6]:

Out[6]:
```
Index(['Row Labels', 'Sum of Jan', 'Sum of Feb', 'Sum of Mar',
       'Sum of Total Sales'],
      dtype='object')
```

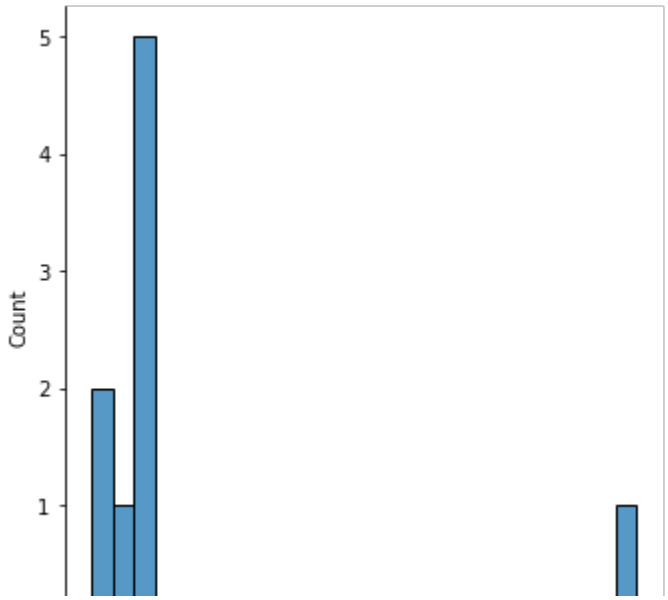
In [7]:

Out[7]:
```
<seaborn.axisgrid.PairGrid at 0x19d59d29df0>
```



In [8]:

Out[8]: <seaborn.axisgrid.FacetGrid at 0x19d59d29280>



In [9]:

Out[9]: <AxesSubplot:>

TO TRAIN THE MODEL - MODEL BUILDING

In [10]:
```
x = a[['Sum of Total Sales']]
```

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

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

Out[12]:
```
LinearRegression()
```

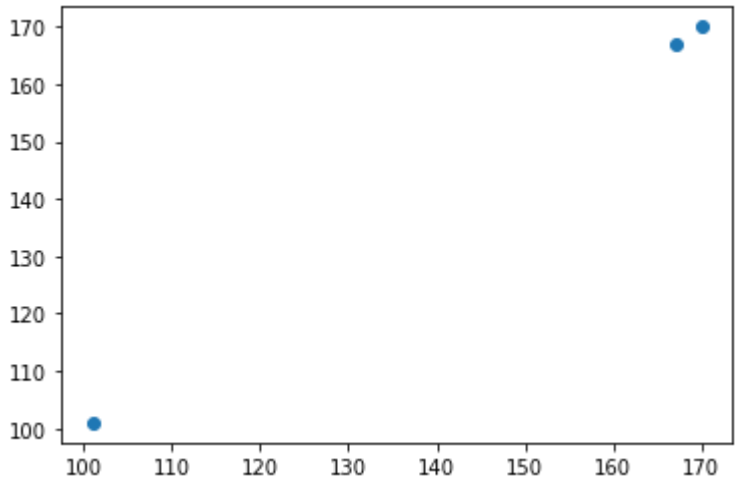
In [13]:
```
coeff = pd.DataFrame(lr.coef_,x.columns,columns=['Co-efficient'])
```

Out[13]:

	Co-efficient
Sum of Total Sales	1.0

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

Out[14]:
```
<matplotlib.collections.PathCollection at 0x19d5bee3be0>
```



In [15]:

Out[15]:
```
1.0
```

RIDGE & LASSO

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

Out[16]:
```
Ridge(alpha=10)
```

In [17]:

Out[17]:
```
0.9999999962049925
```

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

Out[18]:
```
Lasso(alpha=10)
```

In [19]:

Out[19]:
```
0.9999998633765258
```

In [20]:
```
from sklearn.linear_model import ElasticNet
a=ElasticNet()
```

Out[20]:
```
ElasticNet()
```

In [21]:
```
print(a.coef_)
print(a.intercept_)
print(a.score(x_test,y_test))
```

```
[0.99999297]
0.002182120751115235
0.9999999986337749
[167.00100785 101.00147194 170.00098676]
```

In [22]:
```
from sklearn import metrics
print(" Mean Absolute Error :",metrics.mean_absolute_error(y_test,prediction))
print(" Mean Squared Error :",metrics.mean_squared_error(y_test,prediction))
```

```
 Mean Absolute Error : 3.315866100213801e-14
 Mean Squared Error : 1.1443742198406145e-27
 Root Mean Absolute Error : 1.8209519763612112e-07
```

In [ ]:
```

```