In [2]:

```
import numpy as np
import pandas as pd
```

In [12]:

```
class LinearRegression:
    def __init__(self,df, n, sum_x, sum_y, sum_xy,sum_x2, sum_X_h2):
        self.df=df
        self.n=n
        self.sum_x=sum_x
        self.sum_y=sum_y
        self.sum_xy=sum_xy
        self.sum_x2=sum_x2
        self.sum_X_h2=sum_X_h2
    def m_value(self):
        numerator_m = (n*(sum_xy)) - (sum_x*sum_y)
        denominator_m = (n*(sum_x2)) - (sum_X_h2)
        m = numerator_m / denominator_m
        return m

    def b_value(self):
        numerator_b = sum_y - (obj.m_value() * sum_x)
        denominator_b = n
        b = numerator_b/denominator_b
        return b

    def y_outputs(self):
        outputs = [obj.m_value() * X + obj.b_value() for X in df['X']]
        df['outputs']=outputs
        return outputs

    def futureyval(self):
        y_futureval=[(obj.m_value()* X) + obj.b_value() for X in [i for i in range(8,15)]]
        return y_futureval

dataset={'X':[i for i in range(1,8)],
         'Y':[1.5,3.8,6.7,9.0,11.2,13.6,16.0]}
df=pd.DataFrame(dataset)
df['XY'] = df['X'] * df['Y']
df['X2'] = df['X'] ** 2
n = len(df)
sum_x = df['X'].sum()
sum_y = df['Y'].sum()
sum_xy = df['XY'].sum()
sum_x2 = df['X2'].sum()
sum_X_h2 = sum_x ** 2
obj=LinearRegression(df, n, sum_x, sum_y, sum_xy,sum_x2, sum_X_h2)
print("m value is :",obj.m_value())
print("b value is :",obj.b_value())
print("y value is :",obj.y_outputs())
print("Y future values for x in range(8,15) :",obj.futureyval())
print(df)
```

```
m value is : 2.4142857142857133
b value is : -0.8285714285714231
y value is : [1.5857142857142903, 4.0000000000000036, 6.414285714285717, 8.82857142857143,
11.242857142857142, 13.657142857142857, 16.07142857142857]
Y future values for x in range(8,15) : [18.485714285714284, 20.9, 23.31428571428571, 25.72
8571428571424, 28.14285714285714, 30.55714285714285, 32.97142857142856]
   X     Y     XY  X2    outputs
0  1   1.5    1.5   1   1.585714
1  2   3.8    7.6   4   4.000000
2  3   6.7   20.1   9   6.414286
```

```
3  4   9.0   36.0  16   8.828571
4  5  11.2   56.0  25  11.242857
5  6  13.6   81.6  36  13.657143
6  7  16.0  112.0  49  16.071429
```

In [ ]:

In [ ]: