

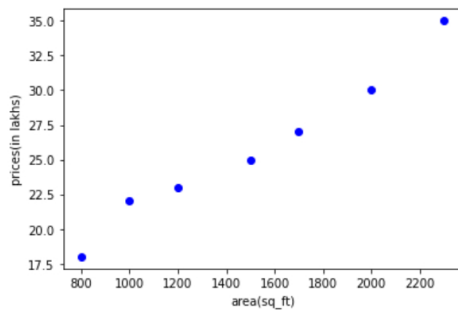
```
In [1]: #import required modules
import pandas as pd
import matplotlib.pyplot as plt
from sklearn import linear_model
```

```
In [2]: #read the csv file
df = pd.read_excel("D:\github repo\PracticeFolder\housing_prices.csv")
df
```

```
Out[2]:
```

	area	price
0	1500	25
1	1200	23
2	1000	22
3	2300	35
4	2000	30
5	800	18
6	1700	27

```
In [3]: #plot the graph
%matplotlib inline
plt.xlabel('area(sq_ft)')
plt.ylabel('prices(in lakhs)')
plt.scatter(df.area, df.price, color='blue')
plt.show()
```



```
In [4]: #use Linear Regression to fit/train the data
reg = linear_model.LinearRegression()
reg.fit(df[['area']], df.price)
```

```
Out[4]:
```

LinearRegression
LinearRegression()

```
In [5]: #predict with a parameter
reg.predict([[1700]])
```

C:\Users\PRATIKBAWANE\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\base.py:450: UserWarning: X does not have valid feature names, but LinearRegression was fitted with feature names
warnings.warn(

```
Out[5]: array([27.74837662])
```

```
In [6]: #Correlation Coefficient
reg.coef_
```

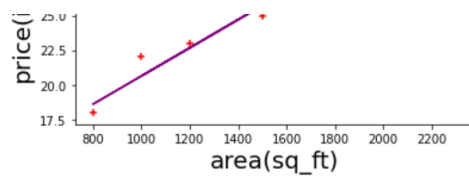
```
Out[6]: array([0.01017045])
```

```
In [7]: #Intercept
reg.intercept_
```

```
Out[7]: 10.4586038961039
```

```
In [8]: %matplotlib inline
plt.xlabel('area(sq_ft)', fontsize=20)
plt.ylabel('price(in_lakhs)', fontsize=20)
plt.scatter(df.area, df.price, color='red', marker='+')
plt.plot(df.area, reg.predict(df[['area']]), color='purple')
```





```
In [9]: #have a new data file for value prediction
newData = pd.read_excel("D:\\github repo\\PracticeFolder\\new_housing_rates.xlsx")
newData
```

```
Out[9]:
```

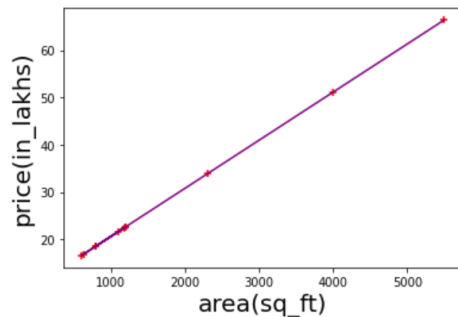
	area
0	1100
1	1180
2	1200
3	625
4	600
5	800
6	1200
7	790
8	2300
9	4000
10	5500

```
In [17]: newPredict = reg.predict(newData)
newPredict
```

```
Out[17]: array([21.6461039 , 22.45974026, 22.66314935, 16.81513799, 16.56087662,
18.59496753, 22.66314935, 18.49326299, 33.85064935, 51.14042208,
66.3961039 ])
```

```
In [18]: #adding a column to this newDataframe
newData['price'] = newPredict
```

```
In [27]: %matplotlib inline
plt.xlabel('area(sq_ft)', fontsize=20)
plt.ylabel('price(in_lakhs)', fontsize=20)
plt.scatter(newData.area, newData.price, color='red', marker='+')
plt.plot(newData.area, reg.predict(newData[['area']]), color='purple')
plt.show()
```



```
In [ ]: # We can save this predicted file likewise
newData.to_csv('D:\\github repo\\PracticeFolder\\newlyPredictedRates.csv', index=False)
```