In [10]: # Required Packages

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

from sklearn import datasets, linear\_model

In [11]: # Function to get data

def get\_data(file\_name):

data = pd.read\_csv(file\_name)

x\_parameter = []

y\_parameter = []

for single\_square\_feet ,single\_price\_value in zip(data['squa

re\_feet'],data['price']):

x\_parameter.append([float(single\_square\_feet)])

y\_parameter.append(float(single\_price\_value))

return x\_parameter,y\_parameter

In [12]: #Printing data

x,y = get\_data("D:/As a Trainer/OGMA/Python/Study Material Folde

r/Linear Regression/House.csv")

print (x)

print (y)

[[50.0], [75.0], [100.0], [125.0], [150.0], [175.0], [200.0],[225.0], [250.0], [275.0], [300.0], [325.0], [350.0], [375.0], [400.0], [425.0], [450.0], [475.0], [500.0], [525.0], [550.0], [575.0], [600.0], [625.0], [650.0], [675.0], [700.0],[800.0], [900.0], [1000.0]][2000.0, 2500.0, 3000.0, 3500.0, 4000.0, 4500.0, 5000.0, 5500.0, 6000.0, 6500.0, 7000.0, 7500.0, 8000.0, 8500.0, 9000.0, 9500.0, 10000.0, 10500.0, 11000.0, 11500.0, 12000.0, 12500.0, 13000.0, 13500.0, 14000.0, 14500.0, 15000.0, 15500.0, 16000.0, 16500.0]

In [13]: # Function for Fitting data to Linear model

def linear\_model\_main(X\_parameters,Y\_parameters,predict\_value):

# Create linear regression object

regr = linear\_model.LinearRegression()

regr.fit(X\_parameters, Y\_parameters)

predict\_outcome = regr.predict(predict\_value)

predictions = {}

predictions['intercept'] = regr.intercept\_

predictions['coefficient'] = regr.coef\_

predictions['predicted\_value'] = predict\_outcome

print("Outcome",predict\_outcome)

#fig=plt.figure()

#plt.plot(predict\_outcome)

return predictions

In [14]: x,y = get\_data("D:/As a Trainer/OGMA/Python/Study Material Folder/Linear egression/House.csv")

predict\_value=int(input("Enter the Area(Square Feet) for which you want to know Price: "))

result = linear\_model\_main(x,y,predict\_value)

print ("Intercept value " , result['intercept'])

print ("coefficient" , result['coefficient'])

print ("Predicted value: ",result['predicted\_value'])

# SELF

#pred=result['predicted\_value']

#print(pred.dtype)

#pred1=pred.tolist()

#print(pred1)

#print(type(pred1))

#y.append(pred1[0])

#print(y)

Enter the Area(Square Feet) for which you want to know Price:

2400

Outcome [43677.56143601]

Intercept value 1788.513300940327

coefficient [17.45377006]

Predicted value: [43677.56143601]

In [15]: # SELF

#predval=[]

#predval.append(predict\_value)

#print(type(predval))

#x.append(predval)

#print(x)

In [18]: # Function to show the resutls of linear fit model

def show\_linear\_line(X\_parameters,Y\_parameters):

# Create linear regression object

regr = linear\_model.LinearRegression()

regr.fit(X\_parameters, Y\_parameters)

plt.scatter(X\_parameters,Y\_parameters,color='blue')

plt.plot(X\_parameters,regr.predict(X\_parameters),color='red'

,linewidth=4)

plt.xticks()

plt.yticks()

plt.show()

print("Plottting Linear regression line")

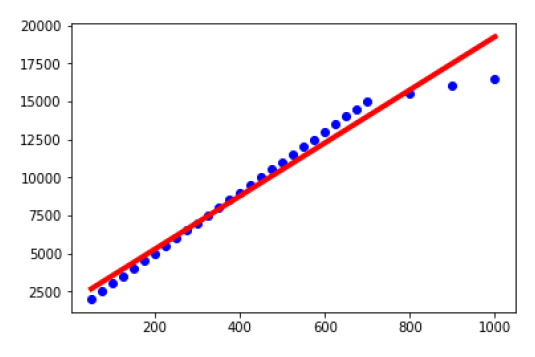
show\_linear\_line(x,y)

print("Plottting the Values")

fig=plt.figure()

plt.plot(x,y,'+m')

Plottting Linear regression line



Plottting the Values

Out[18] : Plottting Linear regression line Plottting the Values [<matplotlib.lines.Line2D at 0x3f1f63df28>]

