In [1]:

**import pandas as pd**

**from sklearn.model\_selection import** train\_test\_split

In [2]:

**from sklearn.linear\_model import** LinearRegression **import matplotlib.pyplot as plt**

**from sklearn.metrics import** r2\_score

In [3]:

filename='C:/Users/Sneha/Desktop/iris.csv'

ds=pd.read\_csv(filename)

X=ds['petal\_length']

Y=ds['petal\_width']

In [4]:

len(X)

Out[4]:

150

In [5]:

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X,Y,test\_size=0.3)

In [6]:

len(X\_train)

Out[6]:

105

In [7]:

len(X\_test)

Out[7]:

45

In [8]:

model=LinearRegression()

In [9]:

model.fit(X\_train.values.reshape(-1,1),Y\_train.values.reshape(-1,1))

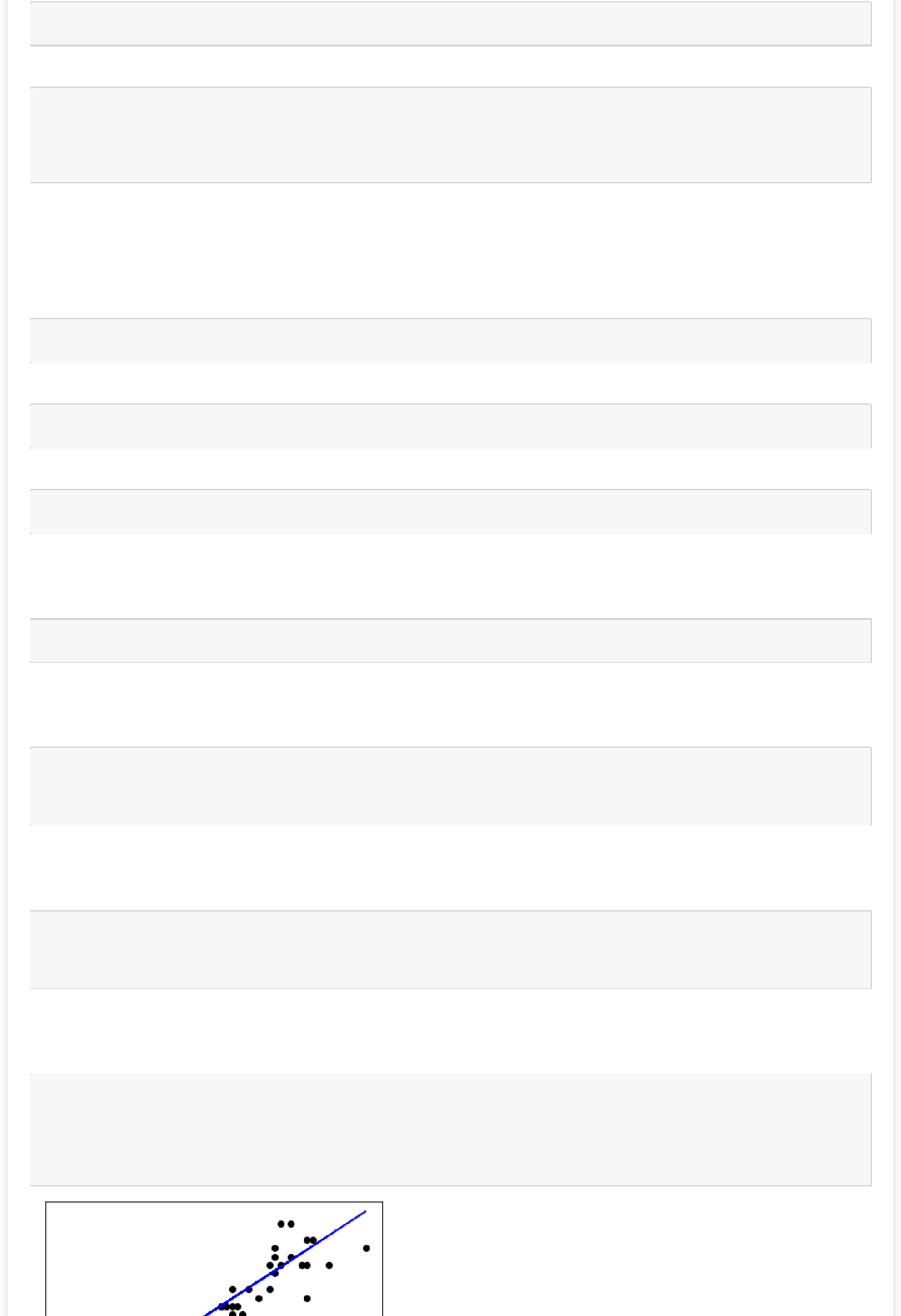
Out[9]:

LinearRegression(copy\_X=True, fit\_intercept=True, n\_jobs=1, normalize=False)

In [10]:

* = model.intercept\_[0]

In [11]:

* = model.coef\_[0,0]

In [12]:

print('Intercept is')

print(c)

print('Slope is')

print(m)

Intercept is

-0.38089127699654446

Slope is

0.421757807883999

In [13]:

y\_pred=model.predict(X\_test.values.reshape(-1,1))

In [14]:

**from sklearn.metrics import** mean\_squared\_error

In [15]:

print("Mean\_Squared\_Error: **%.2f**" % mean\_squared\_error(Y\_test,y\_pred))

Mean\_Squared\_Error: 0.04

In [16]:

print("R2\_Score: **%.2f**" %**r2\_score**(Y\_test,y\_pred))

R2\_Score: 0.92

In [18]:

**from scipy.stats import** pearsonr

r,p = pearsonr(X\_test,Y\_test)

print (r)

0.9628654686561594

In [19]:

**from scipy.stats import** pearsonr

r,p = pearsonr(X,Y)

print (r)

0.9627570970509662

In [20]:

plt.scatter(X\_test,Y\_test,color='black')

plt.plot(X\_test,y\_pred,color='blue')

plt.xticks(())

plt.yticks(())

plt.show()

