**MACHINE LEARNING PROJECT**

In this project, I have used Logistic Regression Machine Learning model for the iris dataset from Kaggle website

**PROBLEM STATEMENT:**

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper The use of multiple measurements in taxonomic problems as an example of linear discriminant analysis. It is sometimes called Anderson's Iris data set because Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species. Two of the three species were collected in the Gaspé Peninsula "all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus".

**ML METHODOLOGY:**

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression).

**DATASET DESCRIPTION:**

Some relevant columns in the dataset

* Sepal length(cm)
* Sepal width(cm)
* petal length(cm)
* petal width(cm)
* Flower\_type

**PRE\_PROCESSING:**

Pre\_processing refers to the transformations applied to our data before feeding it to the algorithm.

1)%matplotlib inline

import matplotlib.pyplot as plt

import numpy as np

import pandas as pd

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

from sklearn import datasets

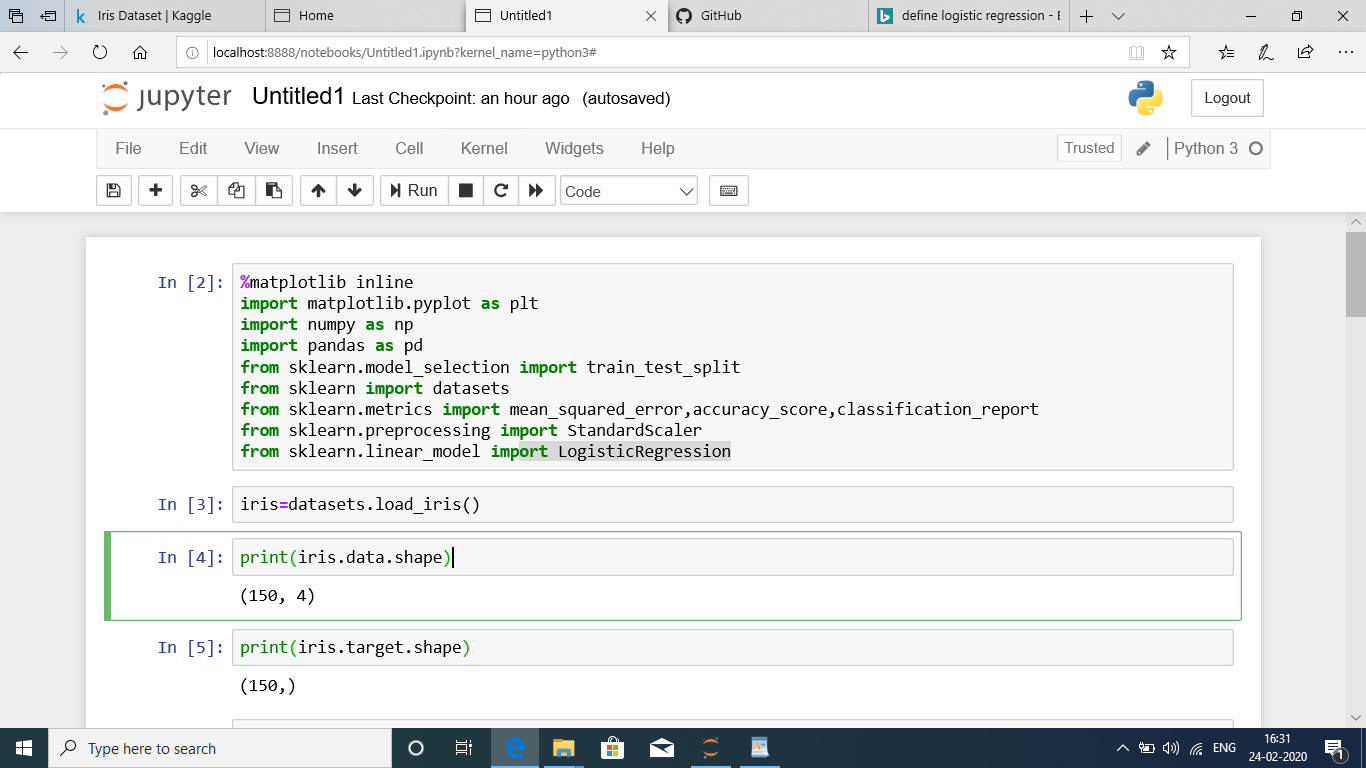
from sklearn.metrics import mean\_squared\_error,accuracy\_score,classification\_report

from sklearn.preprocessing import StandardScaler

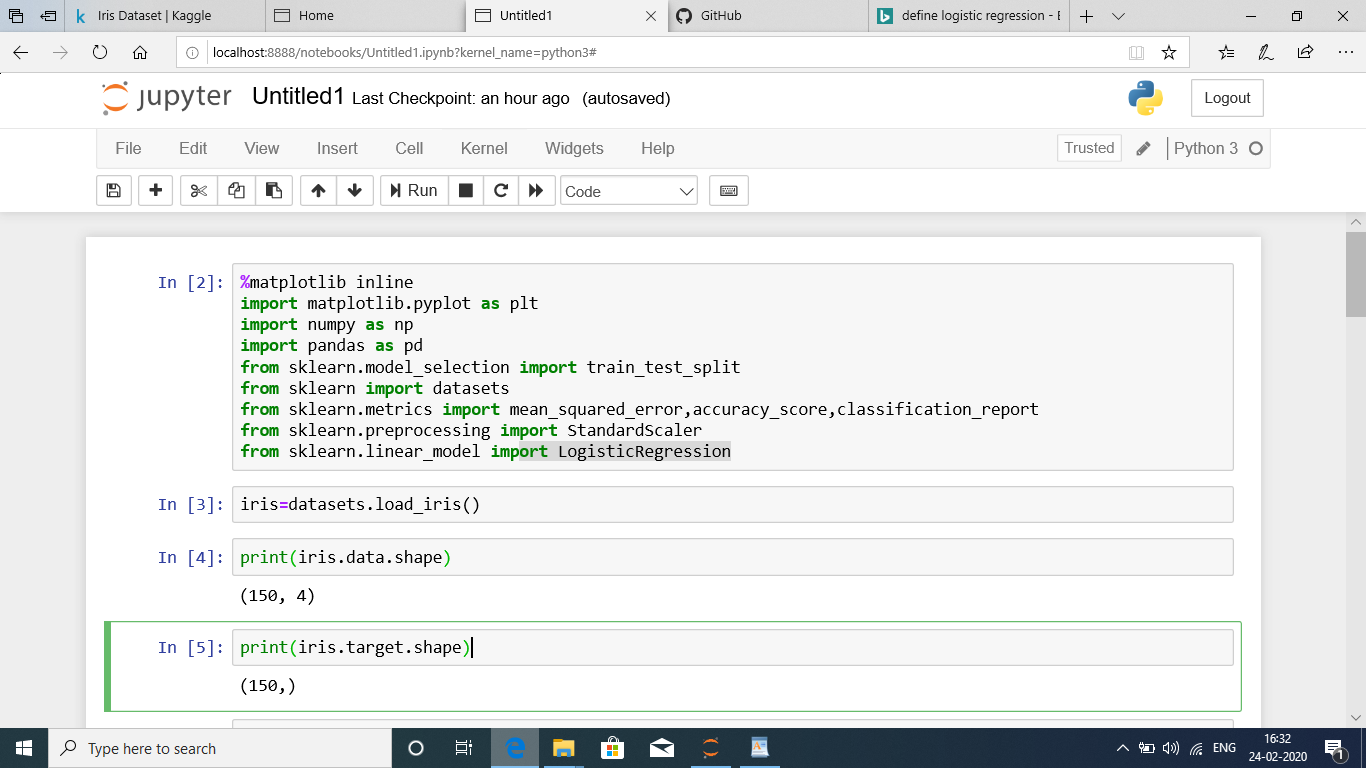
from sklearn.linear\_model import LogisticRegression

2)iris=datasets.load\_iris()

3)print(iris.data.shape)

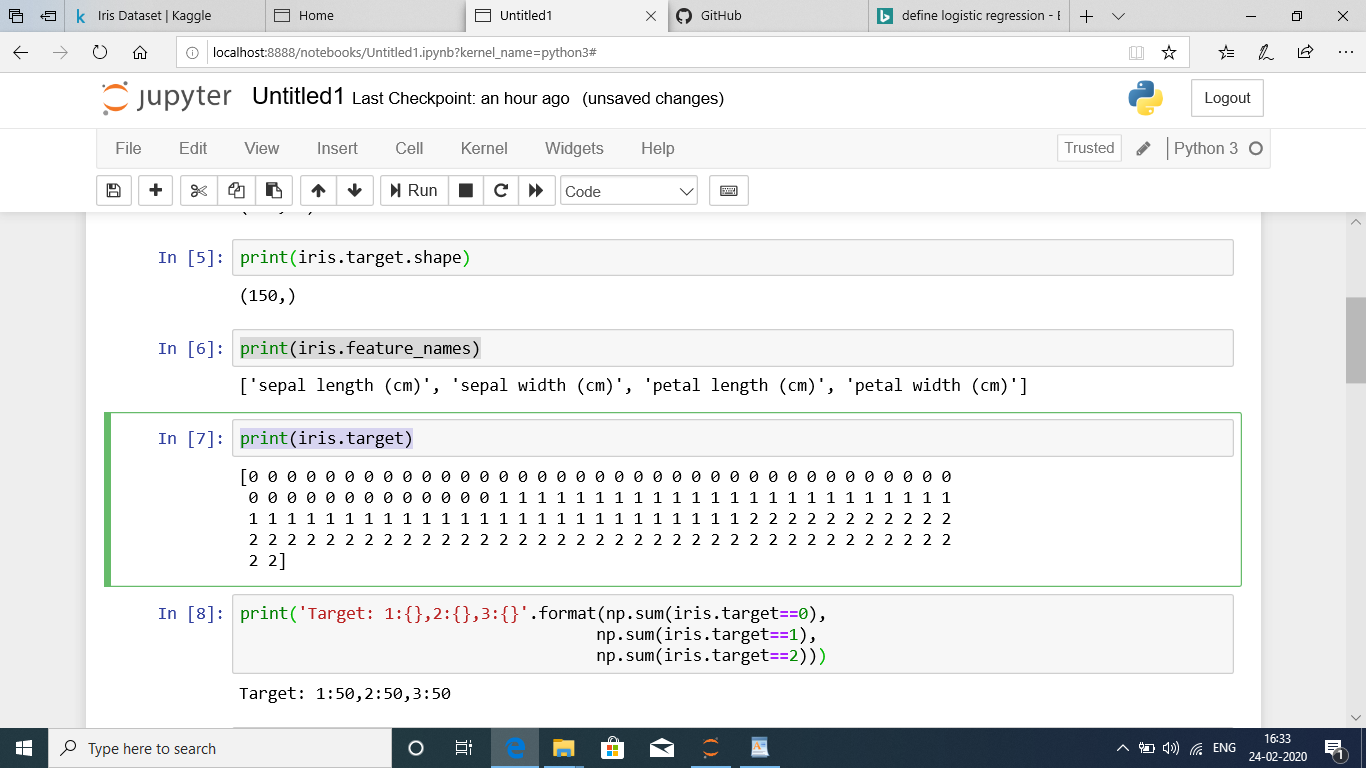


print(iris.target.shape)



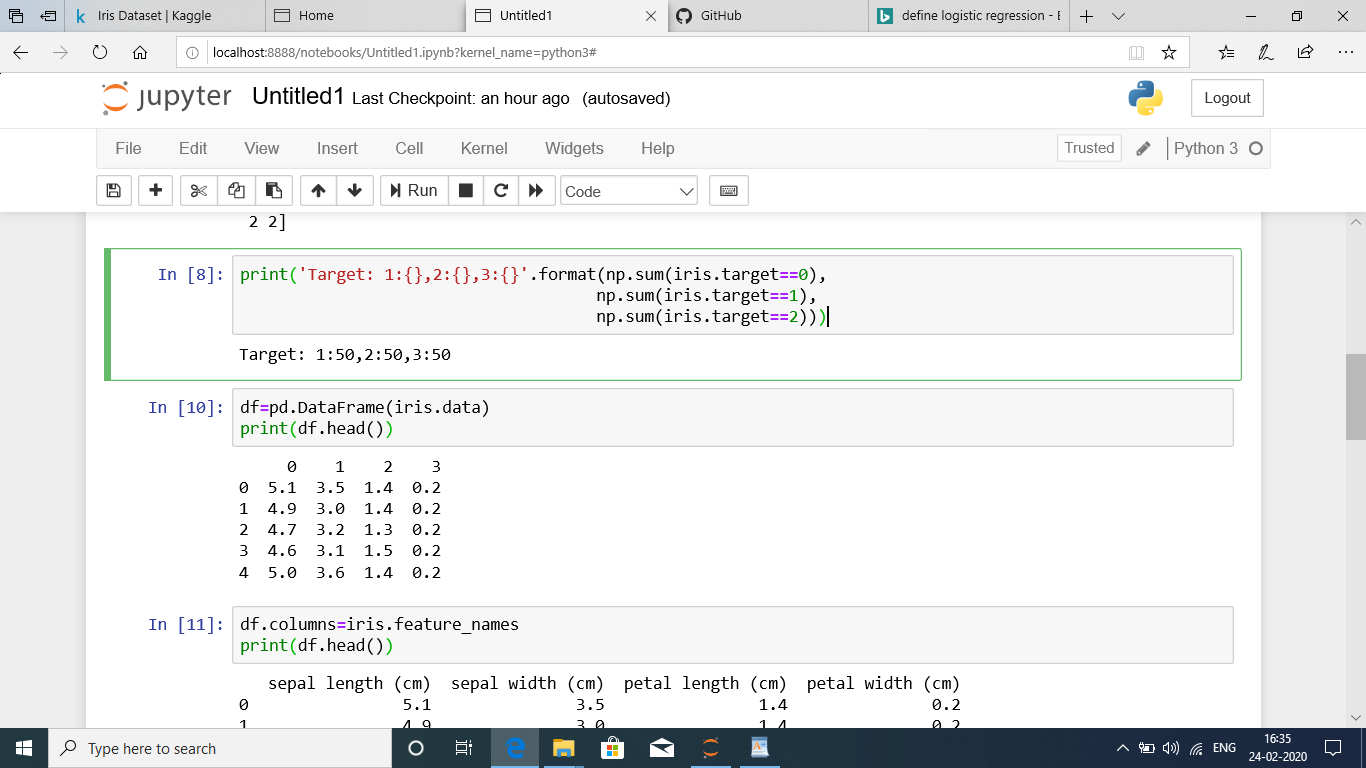
print(iris.feature\_names)

print(iris.target)

print('Target: 1:{},2:{},3:{}'.format(np.sum(iris.target==0),

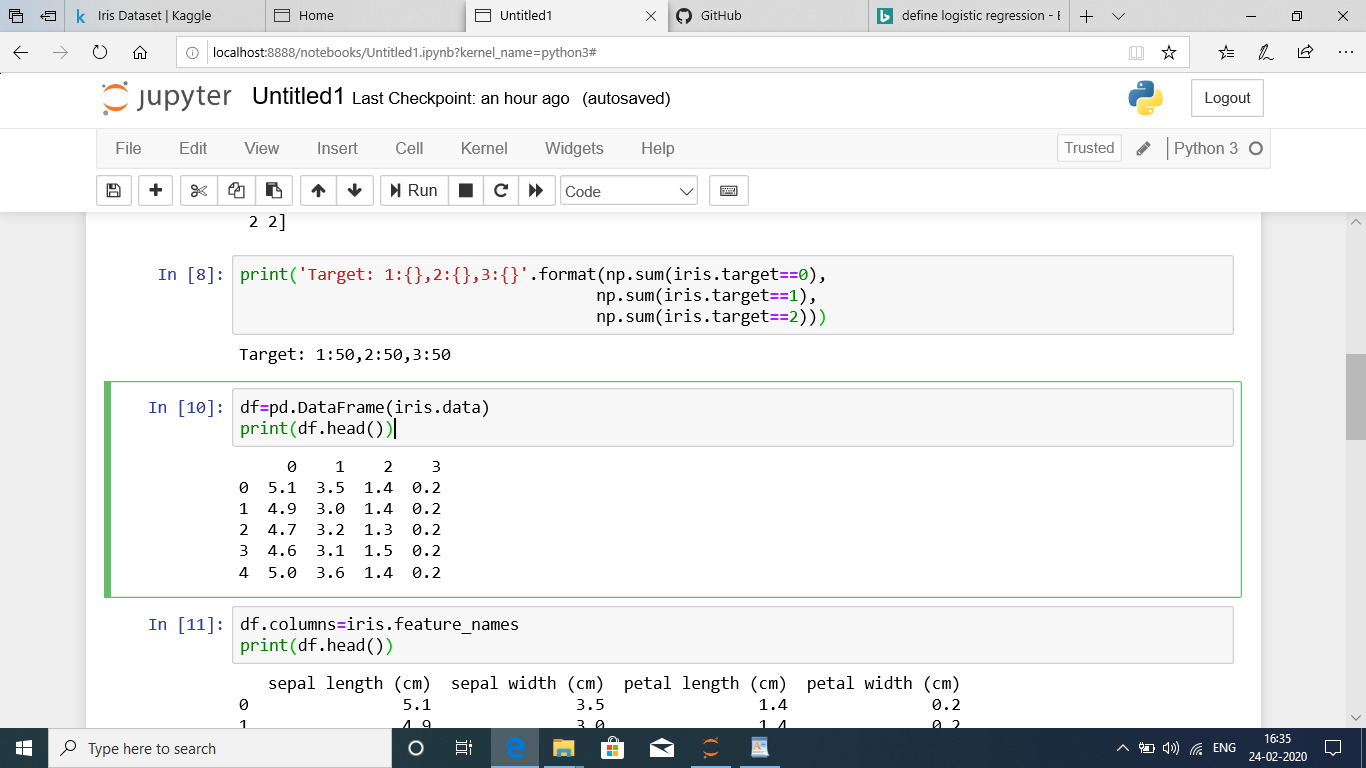
np.sum(iris.target==1),

np.sum(iris.target==2)))



df=pd.DataFrame(iris.data)

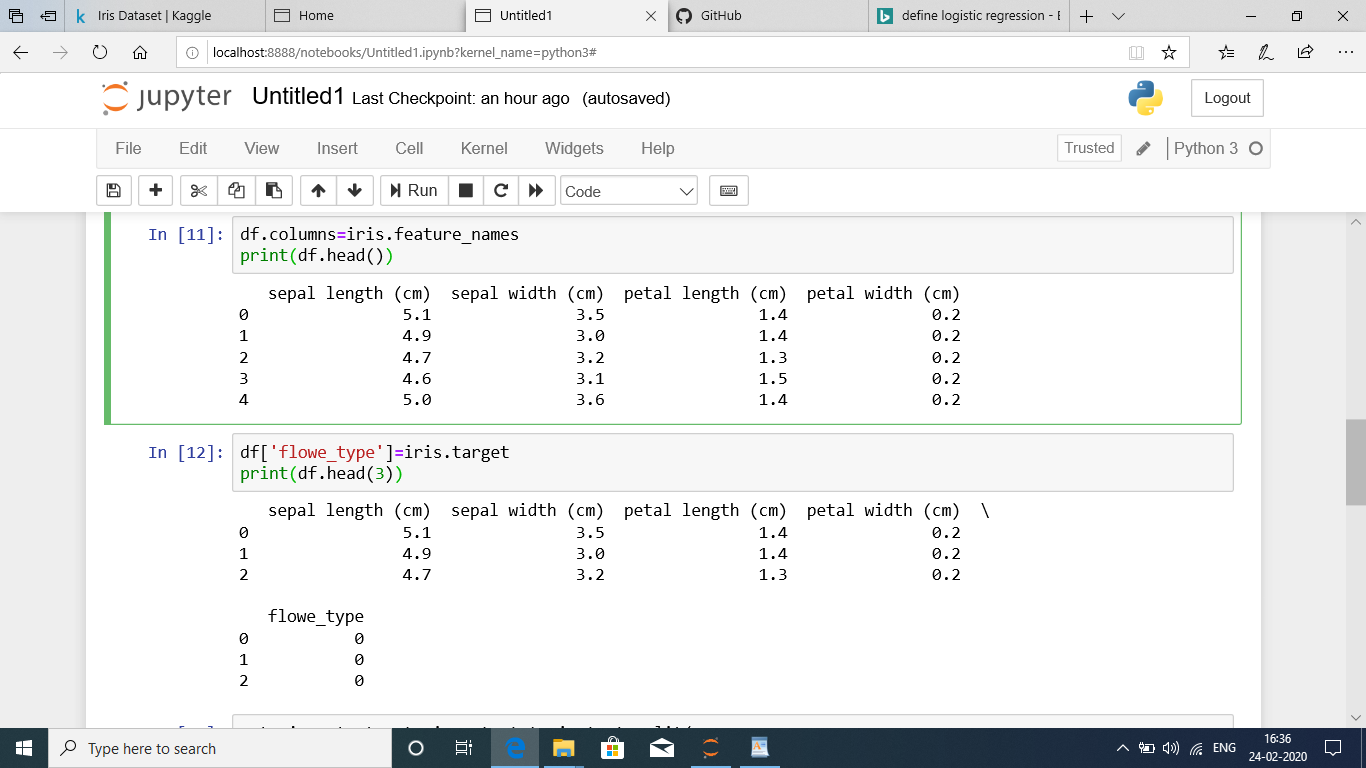
print(df.head())

df.columns=iris.feature\_names

print(df.head())

df['flowe\_type']=iris.target

print(df.head(3))



**Building,training and evaluation of all ML models**

x\_train,x\_test,y\_train,y\_test=train\_test\_split(

iris.data,iris.target,test\_size=0.25,random\_state=42)

scaler=StandardScaler()

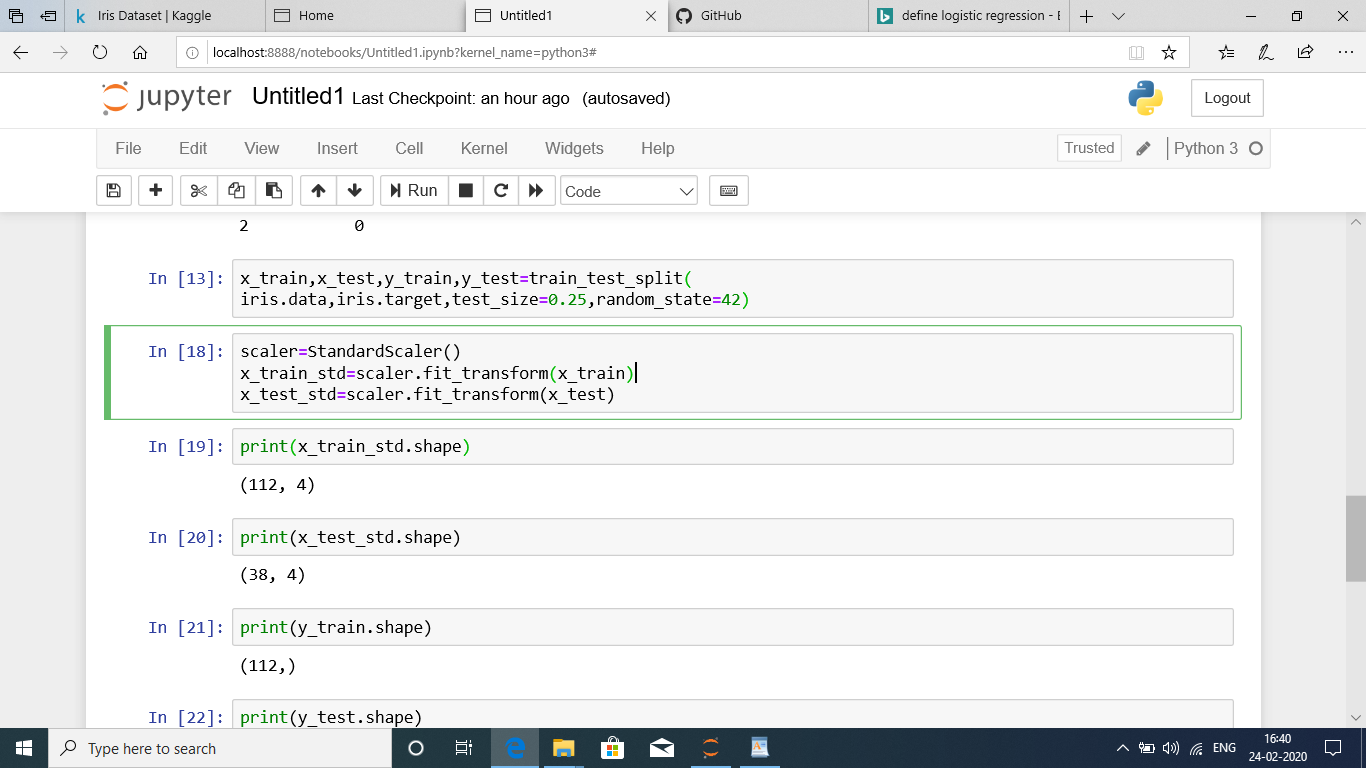
x\_train\_std=scaler.fit\_transform(x\_train)

x\_test\_std=scaler.fit\_transform(x\_test)

print(x\_train\_std.shape)

print(x\_test\_std.shape)

print(y\_train.shape)

print(y\_test.shape)

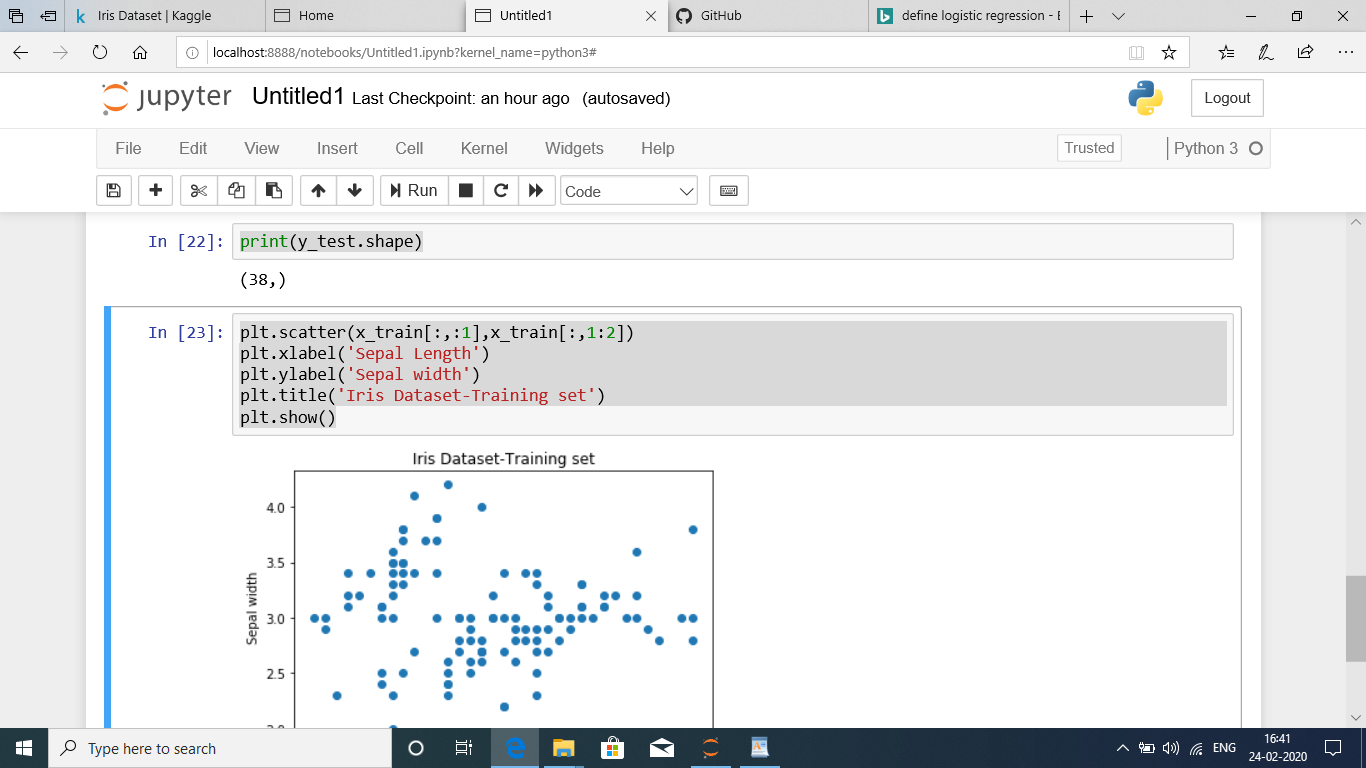
plt.scatter(x\_train[:,:1],x\_train[:,1:2])

plt.xlabel('Sepal Length')

plt.ylabel('Sepal width')

plt.title('Iris Dataset-Training set')

plt.show()



**CONCLUSION:**

Model weights of datasets are calculated by using LogisticRegression and calculate the training and testing dataset.