**EMAIL SPAM DETECTION WITH**

**MACHINE LEARNING**

**DATASET: spam.csv**

**CODE:**

import numpy as np

import pandas as pd

import nltk

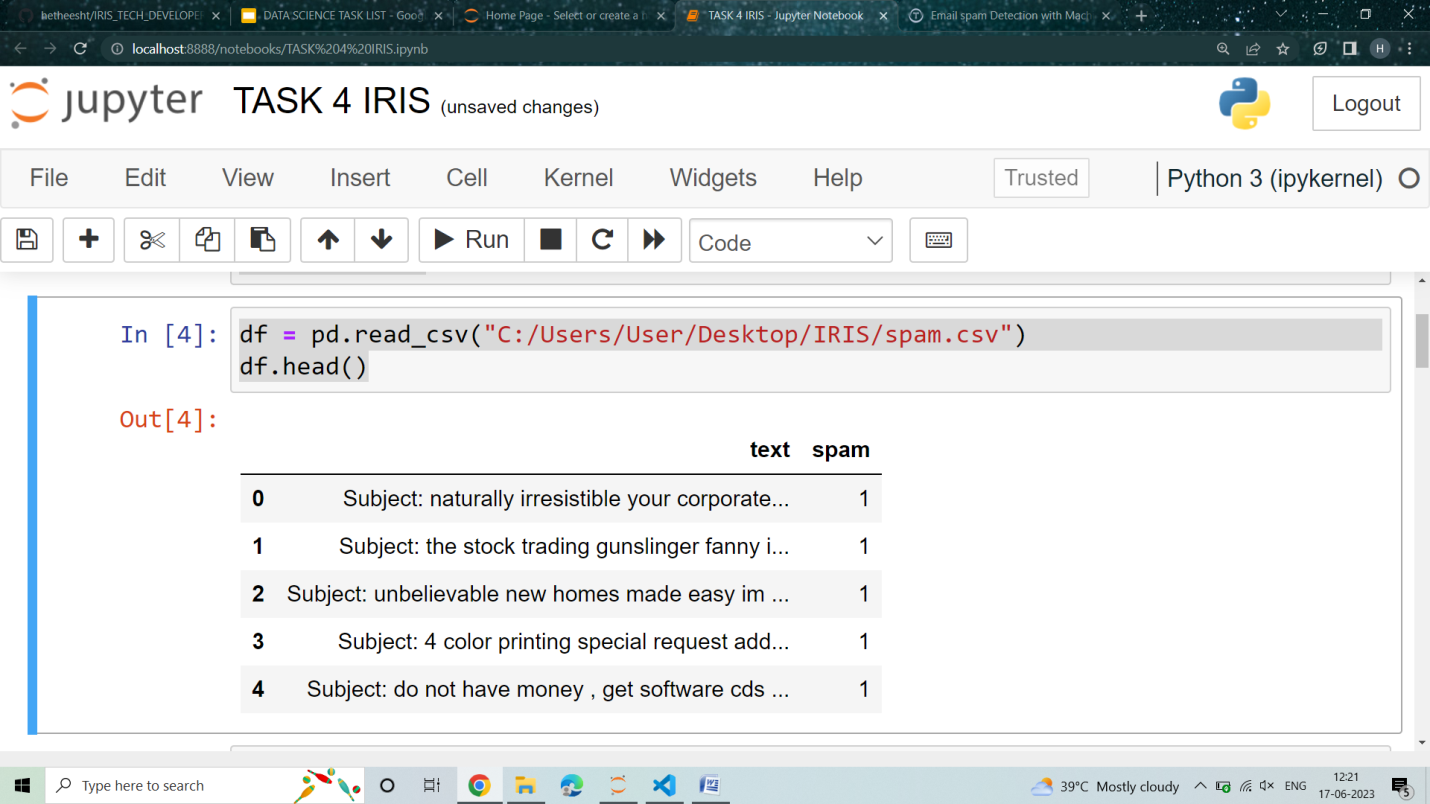
from nltk.corpus import stopwords

import string

df = pd.read\_csv("C:/Users/User/Desktop/IRIS/spam.csv")

df.head()

**Result:**



#continue

df.shape

**Result:**

**(5728, 2)**

#continue

df.columns

nltk.download("stopwords")

**Result:**

**Index(['text', 'spam'], dtype='object')**

**True**

#continue

def process(text):

    nopunc = [char for char in text if char not in string.punctuation]

    nopunc = ''.join(nopunc)

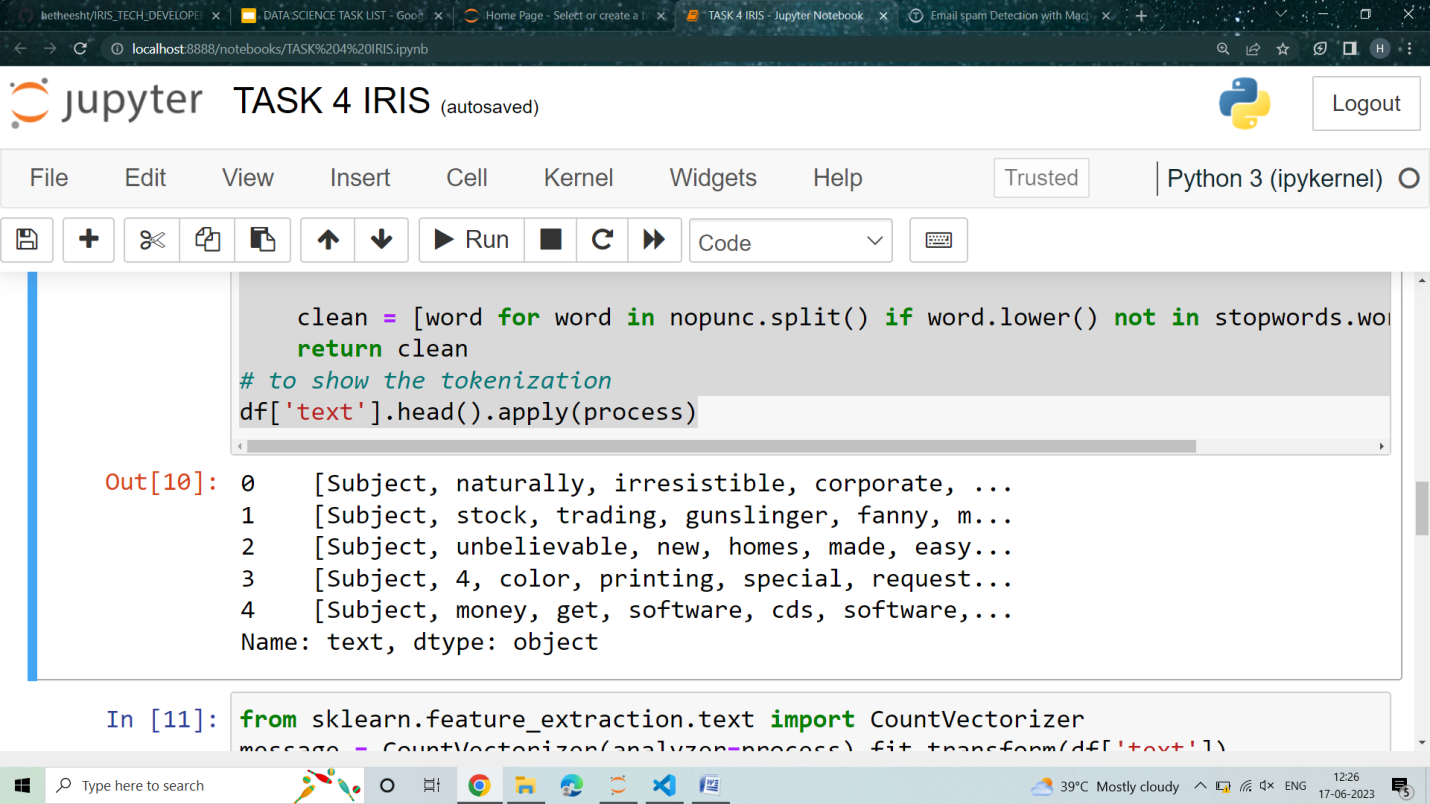
    clean = [word for word in nopunc.split() if word.lower() not in stopwords.words('english')]

    return clean

# to show the tokenization

df['text'].head().apply(process)

**Result:**

****

**.**

#continue

from sklearn.feature\_extraction.text import CountVectorizer

message = CountVectorizer(analyzer=process).fit\_transform(df['text'])

#split the data into 80% training and 20% testing

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

xtrain, xtest, ytrain, ytest = train\_test\_split(message, df['spam'], test\_size=0.20, random\_state=0)

# To see the shape of the data

print(message.shape)

**Result:**

**(5695, 37229)**

**.**

from sklearn.metrics import classification\_report, confusion\_matrix, accuracy\_score

pred = classifier.predict(xtrain)

print(classification\_report(ytrain, pred))

print()

print("Confusion Matrix: \n", confusion\_matrix(ytrain, pred))

print("Accuracy: \n", accuracy\_score(ytrain, pred))

**OUTPUT:**

precision recall f1-score support

0 1.00 1.00 1.00 3457

1 0.99 1.00 0.99 1099

accuracy 1.00 4556

macro avg 0.99 1.00 1.00 4556

weighted avg 1.00 1.00 1.00 4556

Confusion Matrix:

[[3445 12]

[ 1 1098]]

Accuracy: 0.9920983318700615