DESCRIPTION

Comcast is an American global telecommunication company. The firm has been providing terrible customer service. They continue to fall short despite repeated promises to improve. Only last month (October 2016) the authority fined them a $2.3 million, after receiving over 1000 consumer complaints.  
The existing database will serve as a repository of public customer complaints filed against Comcast.  
It will help to pin down what is wrong with Comcast's customer service.

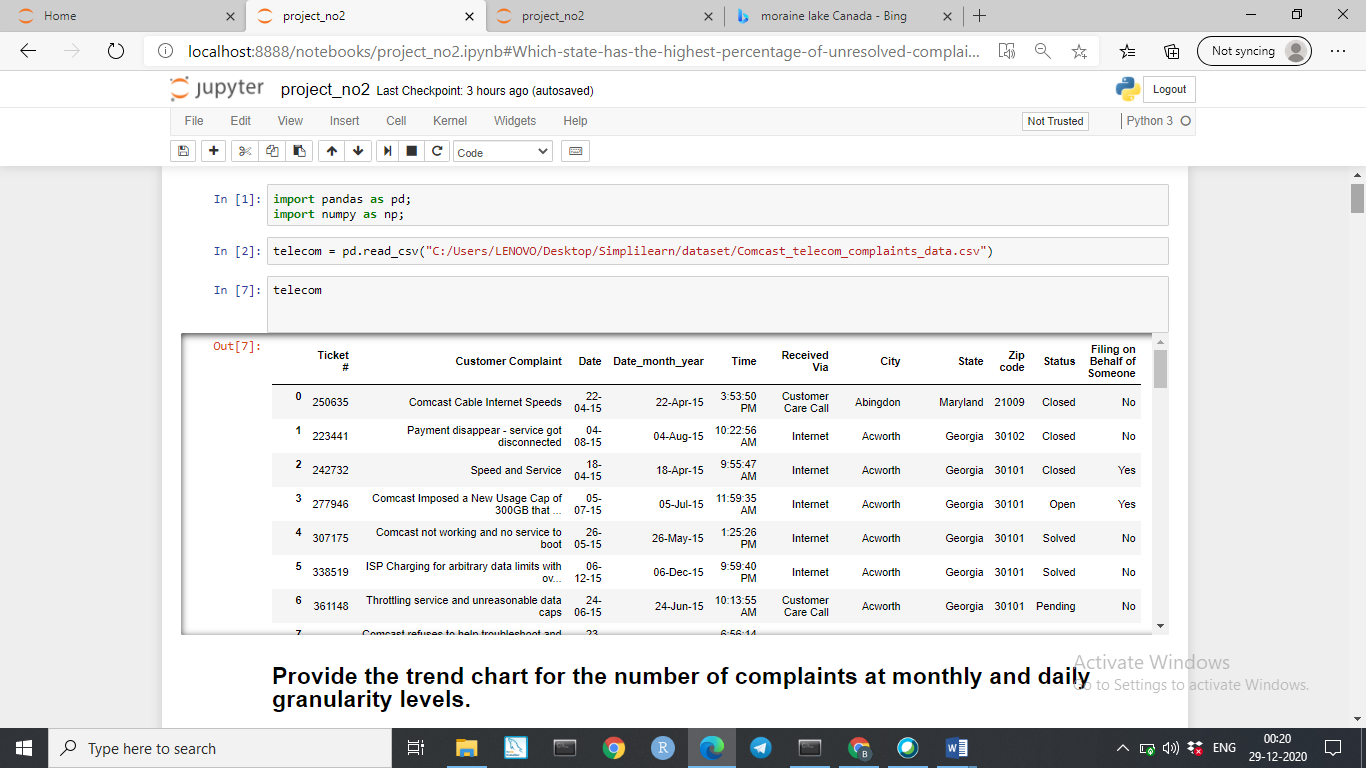
Q 1. - Import data into Python environment.

import pandas as pd;

import numpy as np;

telecom = pd.read\_csv("C:/Users/LENOVO/Desktop/Simplilearn/dataset/Comcast\_telecom\_complaints\_data.csv")

telecom



=======================================================================================================

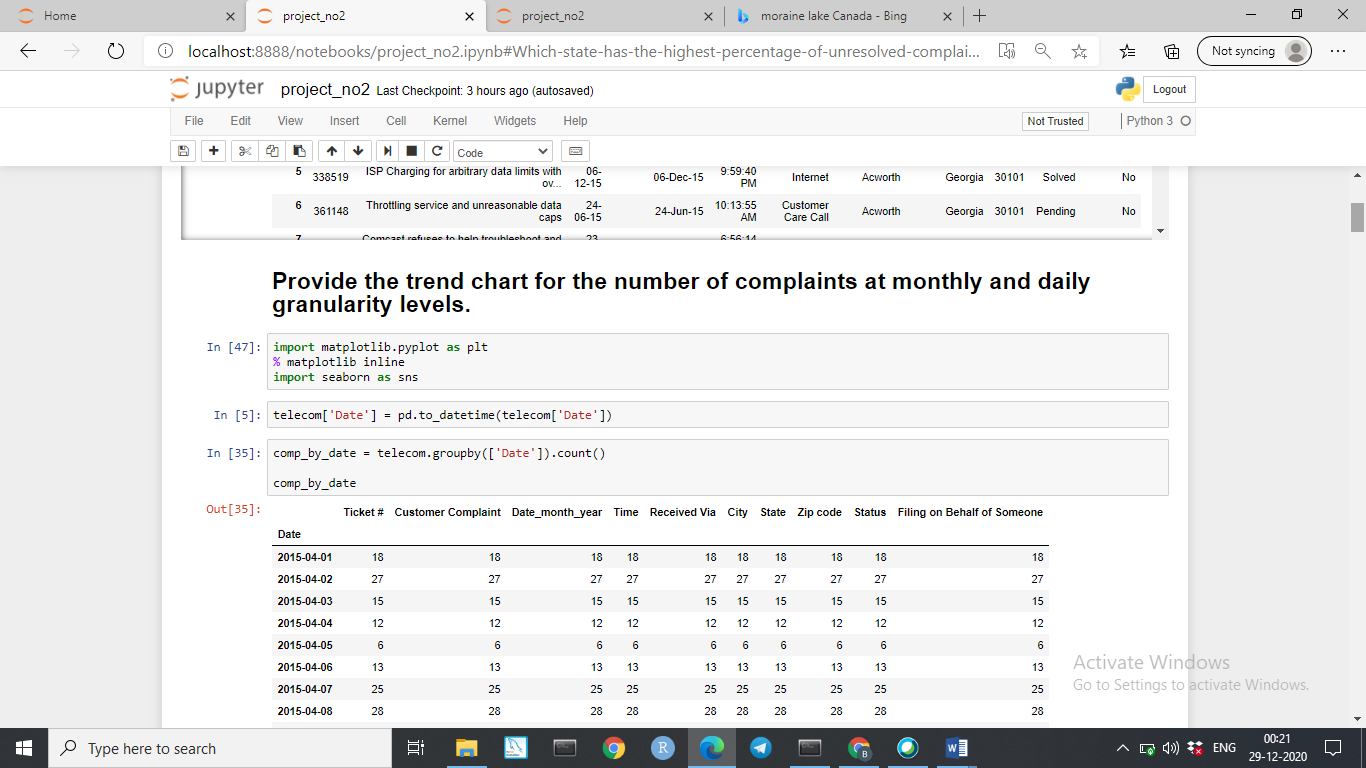
2**. Provide the trend chart for the number of complaints at monthly and daily granularity levels.**

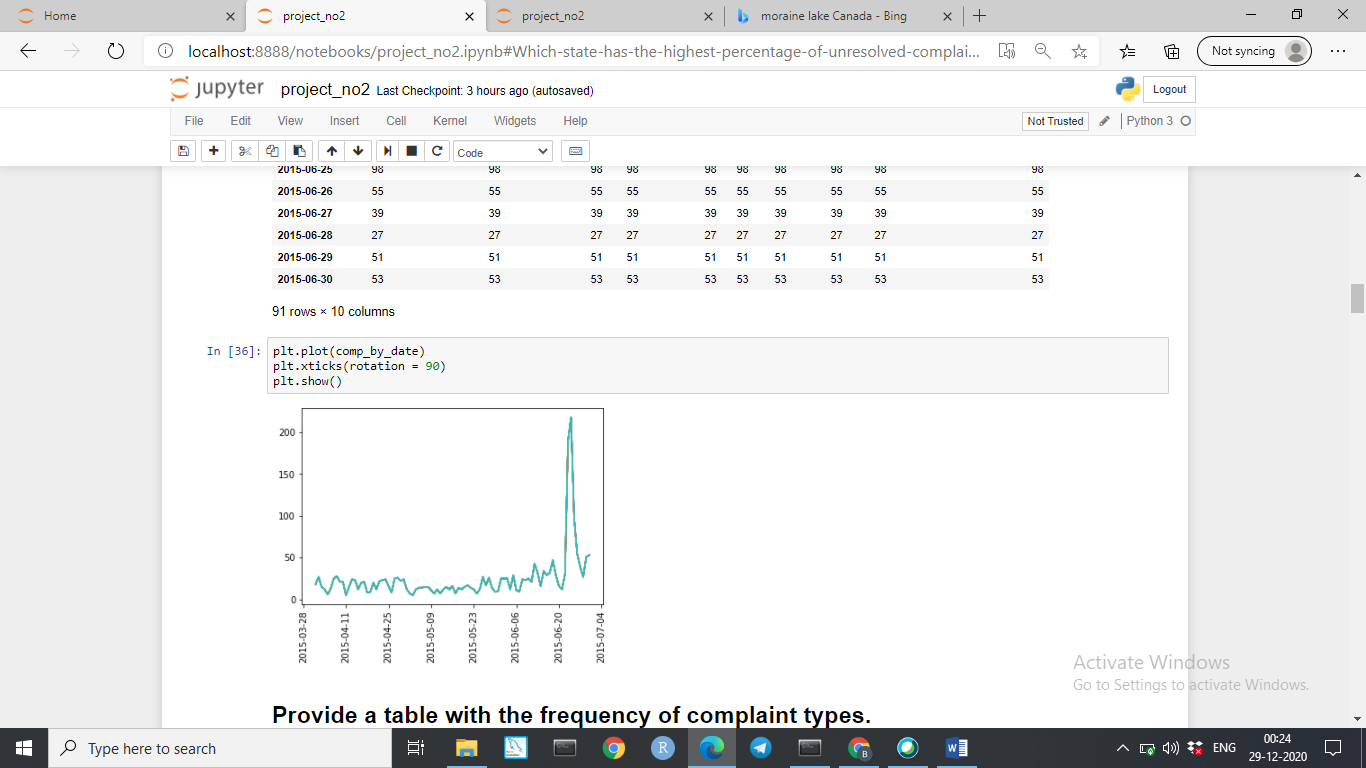
import matplotlib.pyplot as plt

% matplotlib inline

import seaborn as sns

telecom['Date'] = pd.to\_datetime(telecom['Date'])



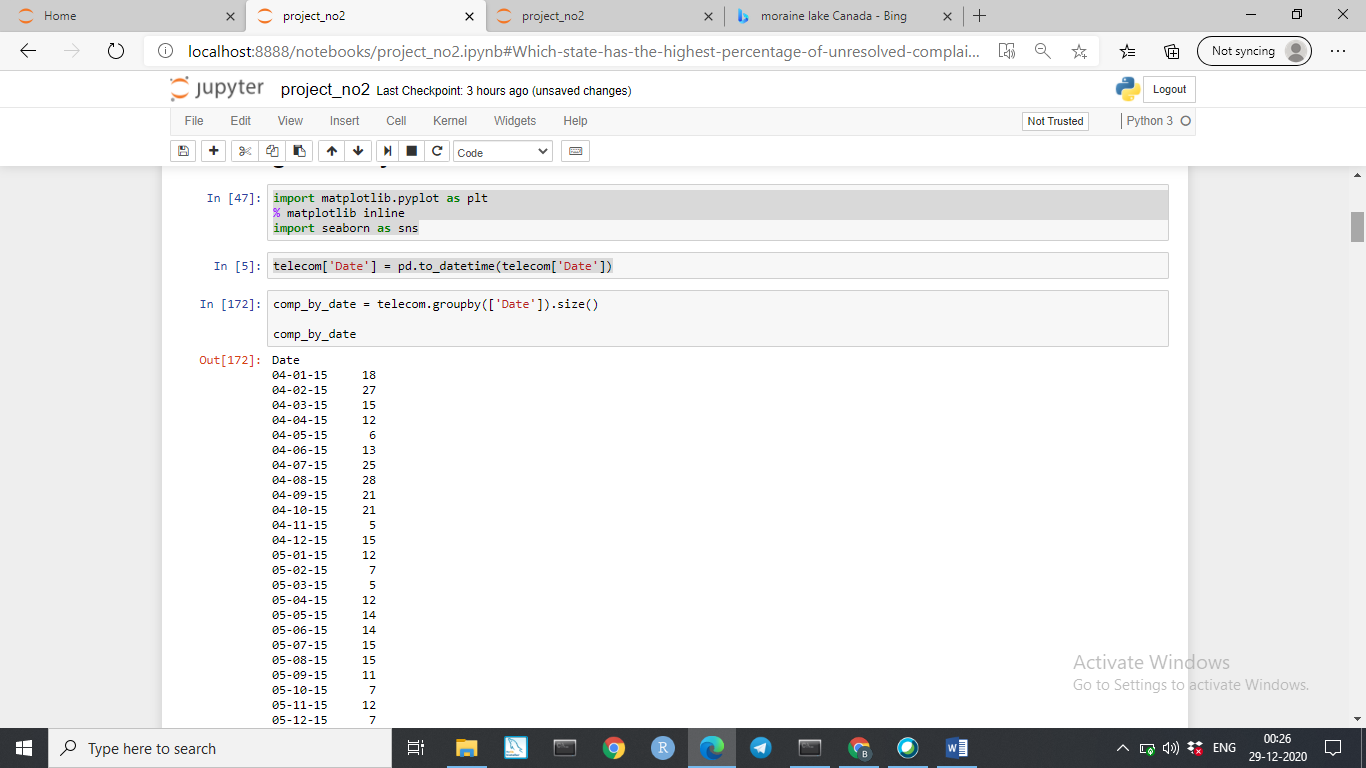


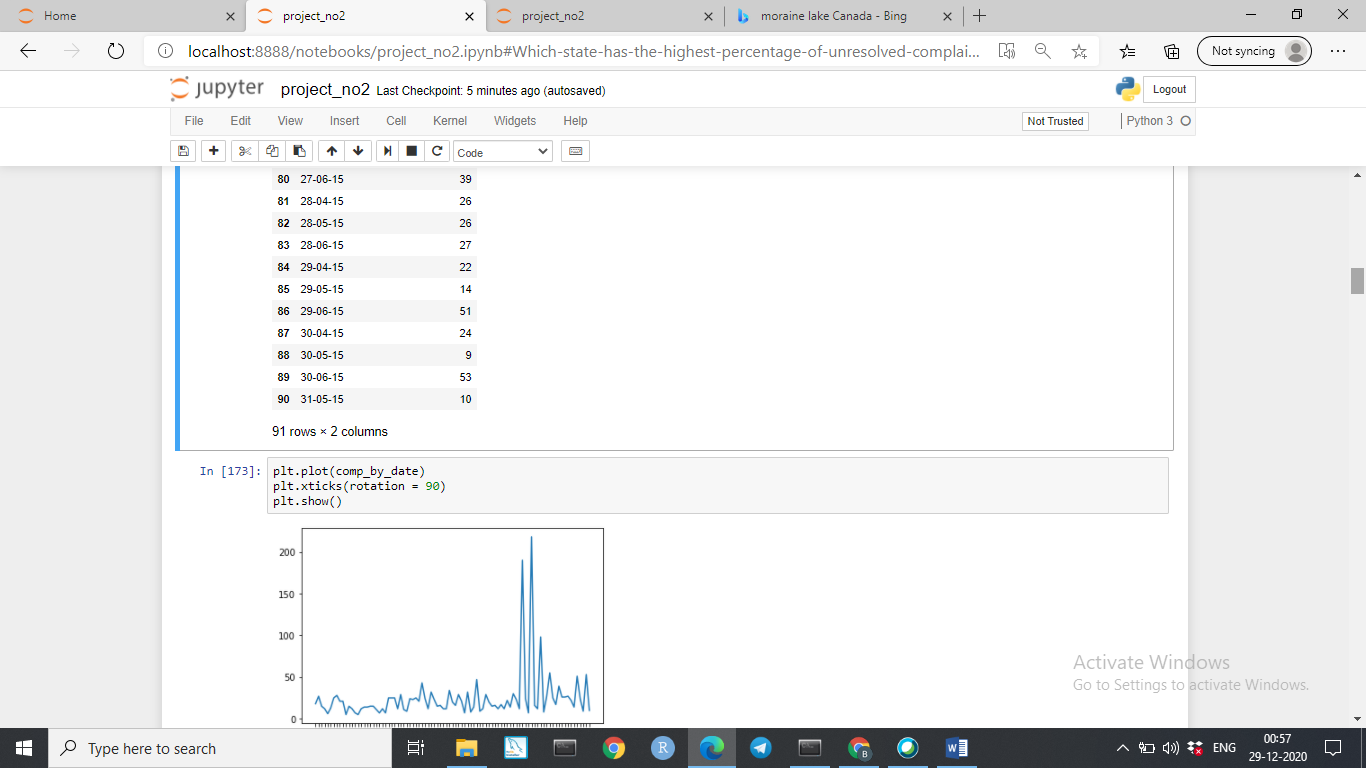
**# Each day no of Complaints**

comp\_by\_date = telecom.groupby(['Date']).size()

new\_df = comp\_by\_date.to\_frame(name = 'no\_of-comp\_each\_day').reset\_index()

new\_df





======================================================================================================

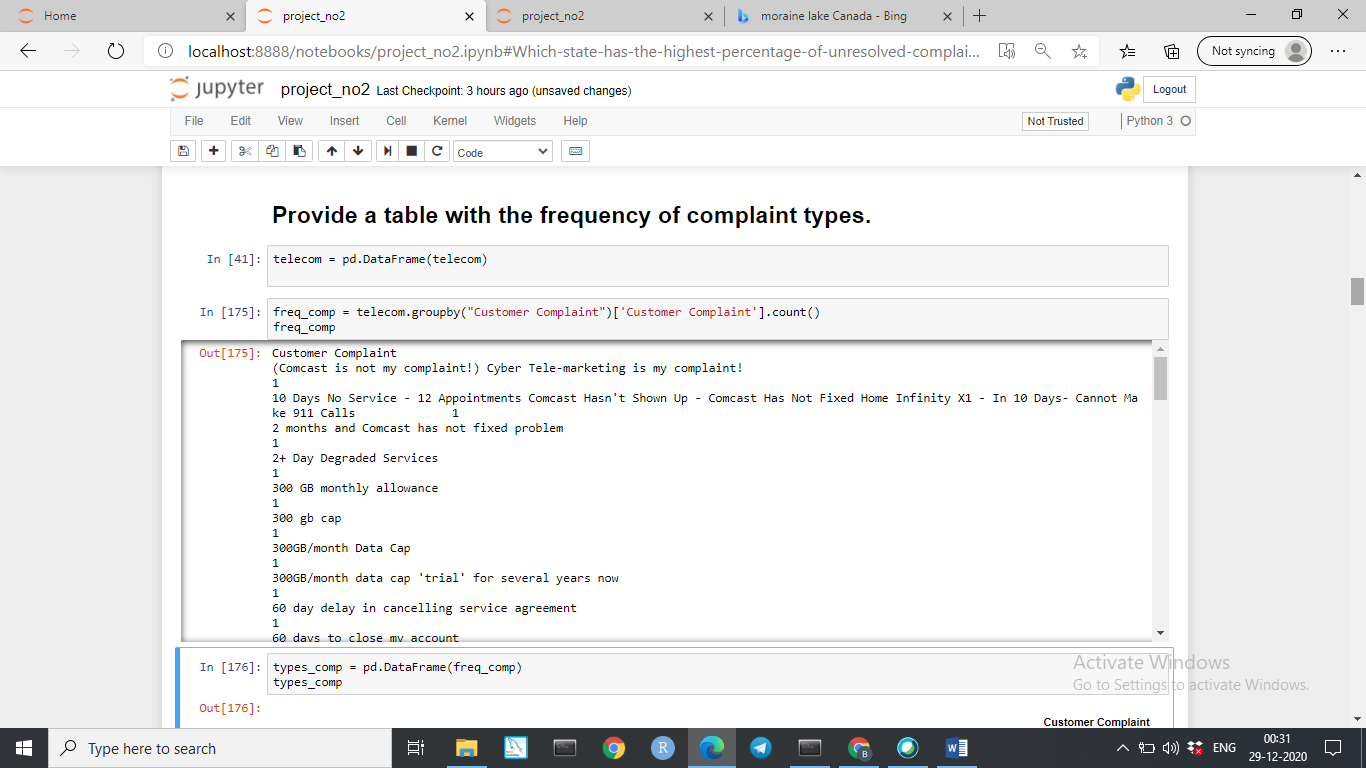
=======================================================================================================

**3. Provide a table with the frequency of complaint types.**

telecom = pd.DataFrame(telecom)

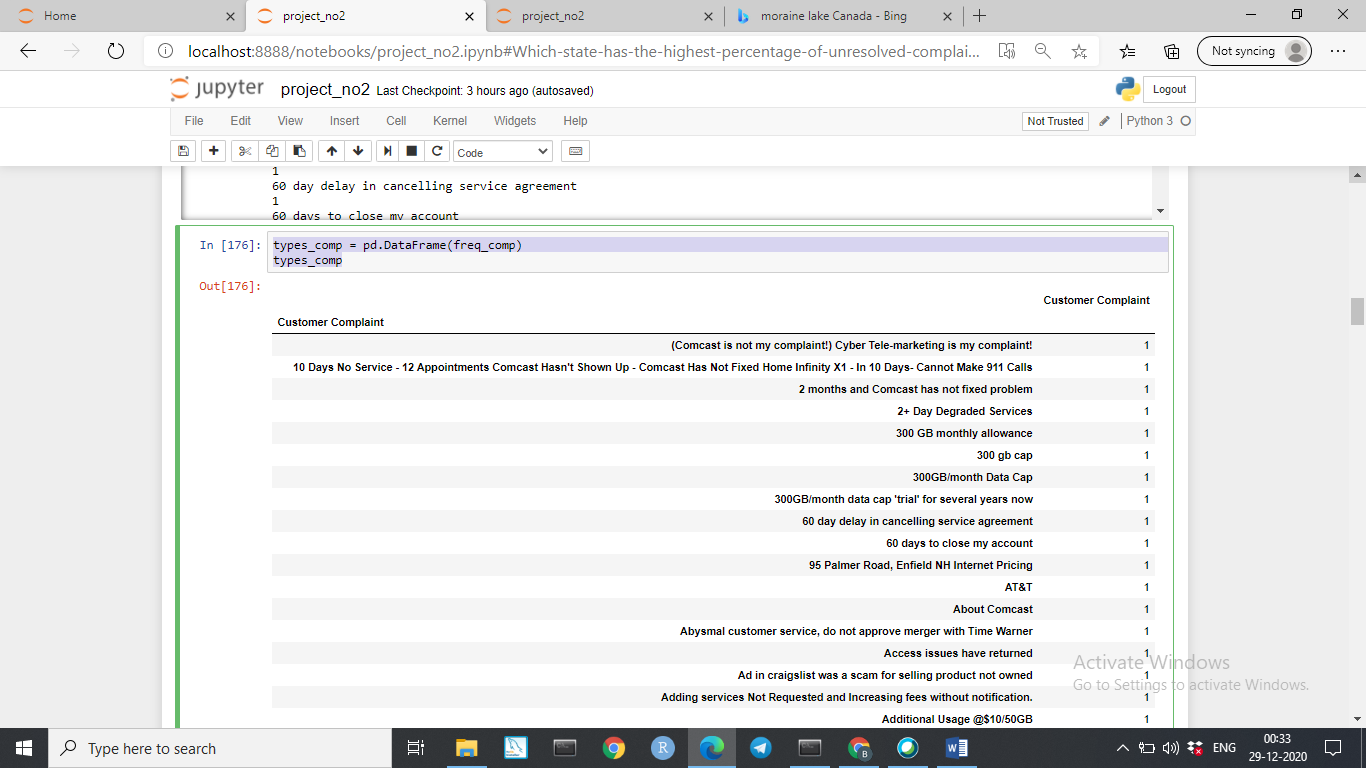
freq\_comp = telecom.groupby("Customer Complaint")['Customer Complaint'].count()

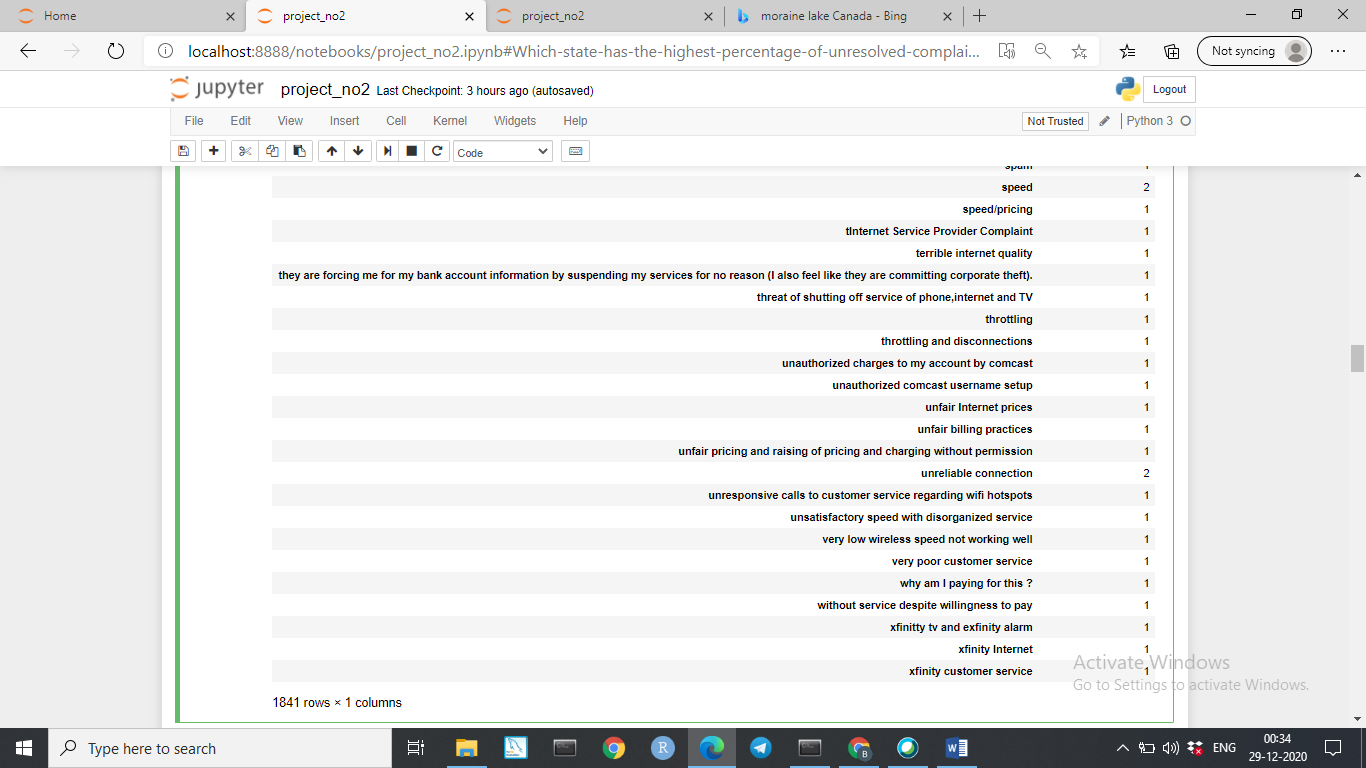
freq\_comp



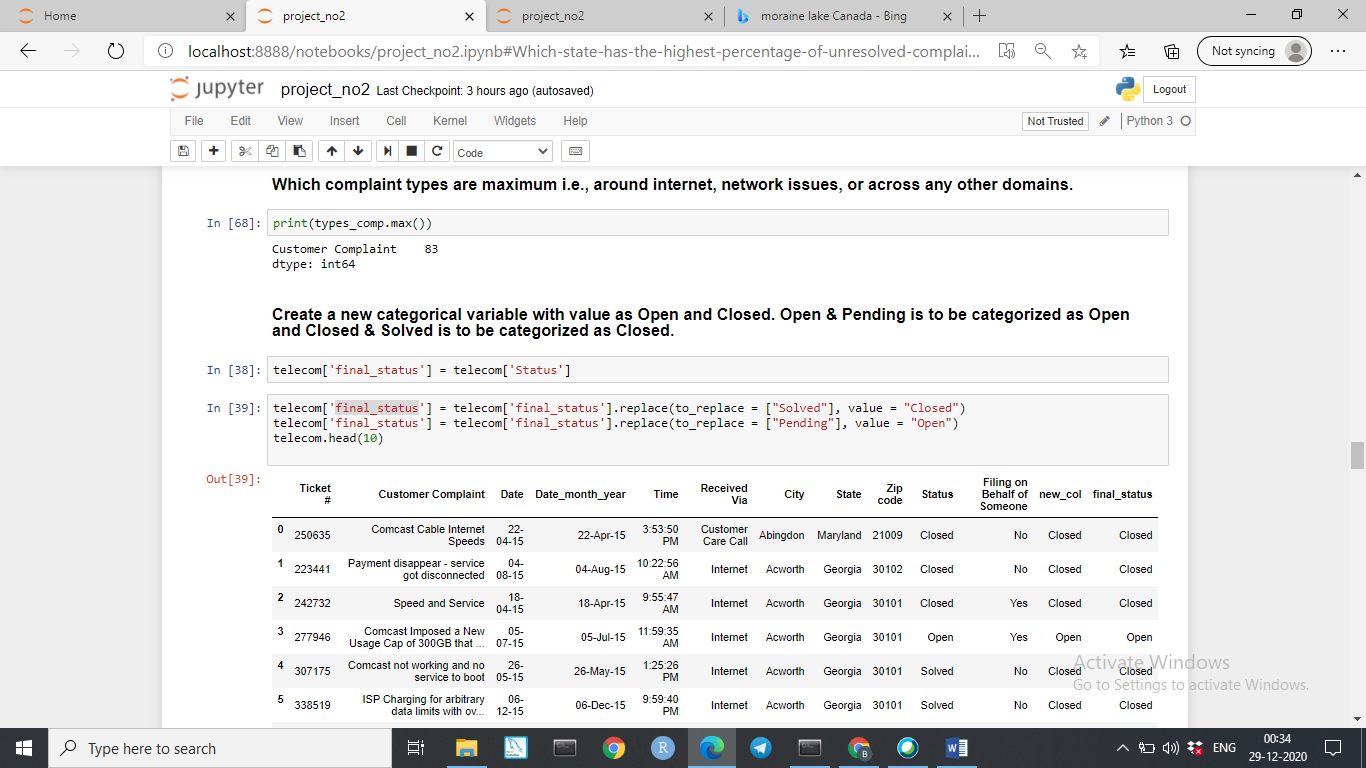
types\_comp = pd.DataFrame(freq\_comp)

types\_comp





* **4. Which complaint types are maximum i.e., around internet, network issues, or across any other domains.**
* print(types\_comp.max())



=======================================================================================================

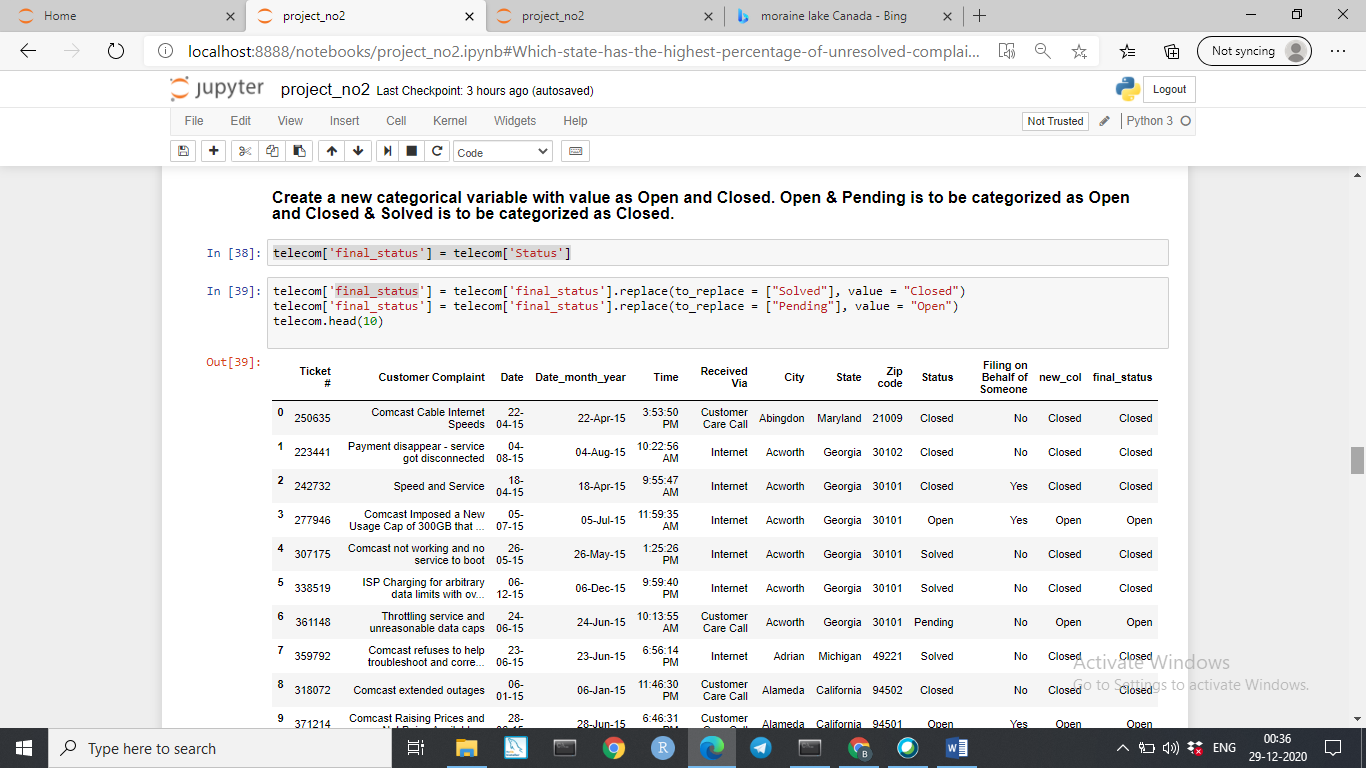
**5. Create a new categorical variable with value as**Open **and**Closed**. Open & Pending is to be categorized as Open and Closed & Solved is to be categorized as Closed.**

telecom['final\_status'] = telecom['Status']

telecom['final\_status'] = telecom['final\_status'].replace(to\_replace = ["Solved"], value = "Closed")

telecom['final\_status'] = telecom['final\_status'].replace(to\_replace = ["Pending"], value = "Open")

telecom.head(10)

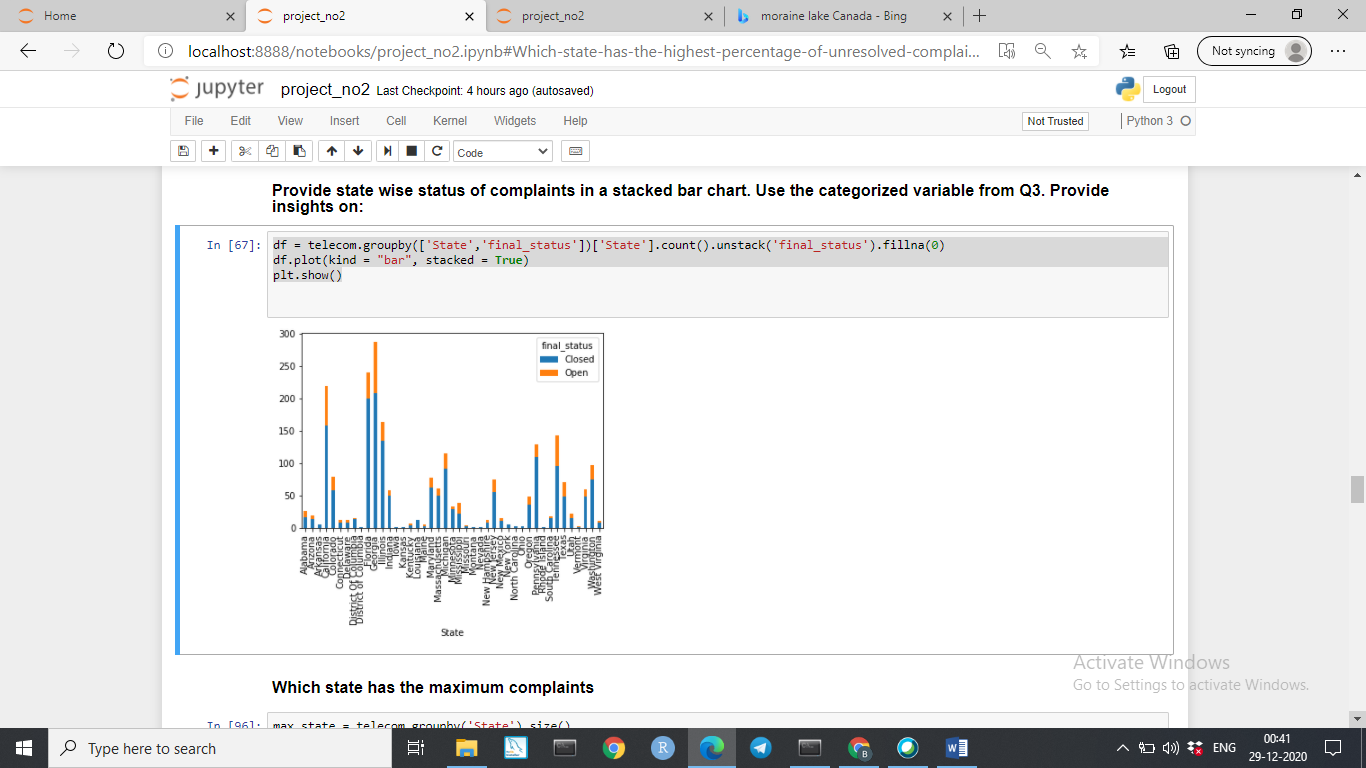


**6. Provide state wise status of complaints in a stacked bar chart. Use the categorized variable from Q3. Provide insights on:**

df = telecom.groupby(['State','final\_status'])['State'].count().unstack('final\_status').fillna(0)

df.plot(kind = "bar", stacked = True)

plt.show()

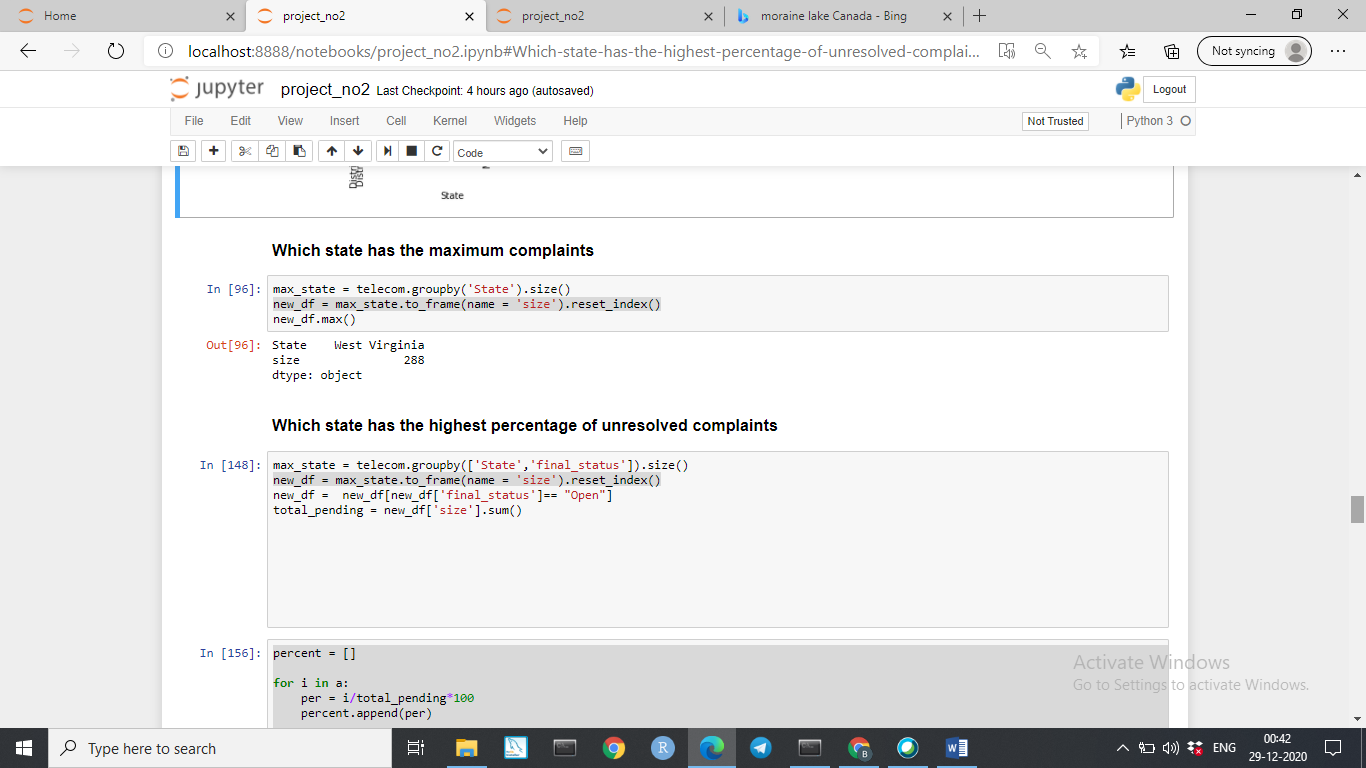


* **7. Which state has the maximum complaints**

max\_state = telecom.groupby('State').size()

new\_df = max\_state.to\_frame(name = 'size').reset\_index()

new\_df.max()



**8. Which state has the highest percentage of unresolved** complaints

max\_state = telecom.groupby(['State','final\_status']).size()

new\_df = max\_state.to\_frame(name = 'size').reset\_index()

new\_df = new\_df[new\_df['final\_status']== "Open"]

total\_pending = new\_df['size'].sum()

percent = []

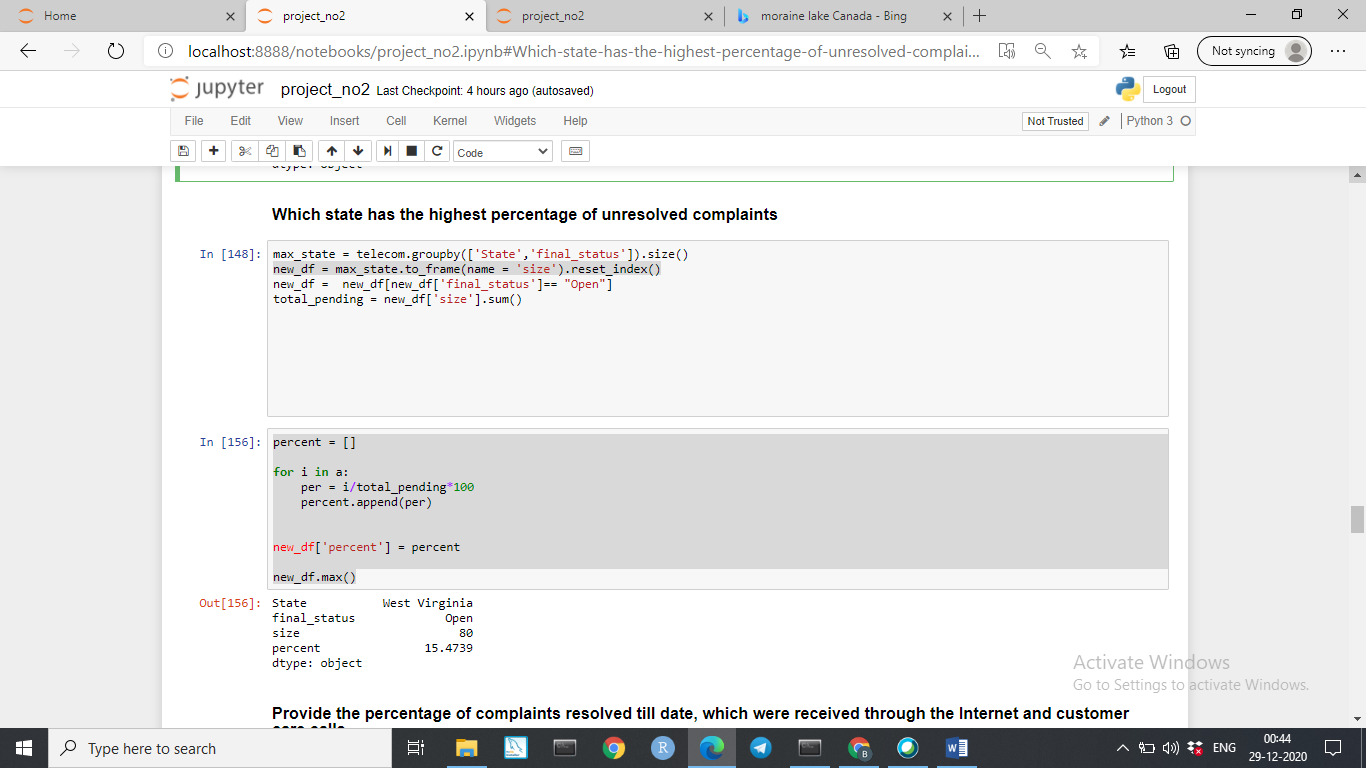
for i in a:

per = i/total\_pending\*100

percent.append(per)

new\_df['percent'] = percent

new\_df.max()



**9. Provide the percentage of complaints resolved till date, which were received through the Internet and customer care calls.**

total\_comp = telecom1.groupby(['Received Via','final\_status']).size()

new\_df = total\_comp.to\_frame(name = 'size').reset\_index()

new\_df

new\_df = new\_df[new\_df['final\_status'] == "Closed"]

total\_solved\_queries = new\_df['size'].sum()

total\_solved\_queries

percent = []

for i in new\_df['size']:

per = i/total\_solved\_queries\*100

percent.append(per)

new\_df['percent'] = percent

new\_df

