In [6]:

```
 #Import reqiered pakage
```

In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [7]:

```
#Import data into Python
```

In [3]:

```
df=pd.read_csv('/Users/admin/Downloads/Comcast_telecom_complaints_data.csv')
```

In [4]:

```
df.head()
```

Out[4]:

	Ticket #	Customer Complaint	Date	Date_month_year	Time	Received Via	City	State	Zip code	Status	Filing on Behalf of Someone
0	250635	Comcast Cable Internet Speeds	22-04-15	22-Apr-15	3:53:50 PM	Customer Care Call	Abingdon	Maryland	21009	Closed	No
1	223441	Payment disappear - service got disconnected	04-08-15	04-Aug-15	10:22:56 AM	Internet	Acworth	Georgia	30102	Closed	No
2	242732	Speed and Service	18-04-15	18-Apr-15	9:55:47 AM	Internet	Acworth	Georgia	30101	Closed	Yes
3	277946	Comcast Imposed a New Usage Cap of 300GB that ...	05-07-15	05-Jul-15	11:59:35 AM	Internet	Acworth	Georgia	30101	Open	Yes
4	307175	Comcast not working and no service to boot	26-05-15	26-May-15	1:25:26 PM	Internet	Acworth	Georgia	30101	Solved	No

In [5]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2224 entries, 0 to 2223
Data columns (total 11 columns):
 #   Column                       Non-Null Count  Dtype 
---  ------                       --------------  ----- 
 0   Ticket #                     2224 non-null   object
 1   Customer Complaint           2224 non-null   object
 2   Date                         2224 non-null   object
 3   Date_month_year              2224 non-null   object
 4   Time                         2224 non-null   object
 5   Received Via                 2224 non-null   object
 6   City                         2224 non-null   object
 7   State                        2224 non-null   object
 8   Zip code                     2224 non-null   int64 
 9   Status                       2224 non-null   object
 10  Filing on Behalf of Someone  2224 non-null   object
dtypes: int64(1), object(10)
memory usage: 191.2+ KB
```

In [6]:

```
df.isnull().sum()
```

Out[6]:

```
Ticket #                        0
Customer Complaint              0
Date                            0
Date_month_year                 0
Time                            0
Received Via                    0
City                            0
State                           0
Zip code                        0
Status                          0
Filing on Behalf of Someone     0
dtype: int64
```

In [8]:

```
df=df.drop(['Ticket #','Time'],axis=1)
```

In [9]:

```
df.head()
```

Out[9]:

	Customer Complaint	Date	Date_month_year	Received Via	City	State	Zip code	Status	Filing on Behalf of Someone
0	Comcast Cable Internet Speeds	22-04-15	22-Apr-15	Customer Care Call	Abingdon	Maryland	21009	Closed	No
1	Payment disappear - service got disconnected	04-08-15	04-Aug-15	Internet	Acworth	Georgia	30102	Closed	No
2	Speed and Service	18-04-15	18-Apr-15	Internet	Acworth	Georgia	30101	Closed	Yes
3	Comcast Imposed a New Usage Cap of 300GB that ...	05-07-15	05-Jul-15	Internet	Acworth	Georgia	30101	Open	Yes
4	Comcast not working and no service to boot	26-05-15	26-May-15	Internet	Acworth	Georgia	30101	Solved	No

In [10]:

```
df['Date_month_year']=df['Date_month_year'].apply(pd.to_datetime)
```

In [11]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2224 entries, 0 to 2223
Data columns (total 9 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   Customer Complaint           2224 non-null   object
 1   Date                         2224 non-null   object
 2   Date_month_year              2224 non-null   datetime64[ns]
 3   Received Via                 2224 non-null   object
 4   City                         2224 non-null   object
 5   State                        2224 non-null   object
 6   Zip code                     2224 non-null   int64
 7   Status                       2224 non-null   object
 8   Filing on Behalf of Someone  2224 non-null   object
dtypes: datetime64[ns](1), int64(1), object(7)
memory usage: 156.5+ KB
```

In [12]:

```
df=df.set_index('Date_month_year')
```

In [13]:

```
df.head()
```

Out[13]:

	Customer Complaint	Date	Received Via	City	State	Zip code	Status	Filing on Behalf of Someone
Date_month_year								
2015-04-22	Comcast Cable Internet Speeds	22-04-15	Customer Care Call	Abingdon	Maryland	21009	Closed	No
2015-08-04	Payment disappear - service got disconnected	04-08-15	Internet	Acworth	Georgia	30102	Closed	No
2015-04-18	Speed and Service	18-04-15	Internet	Acworth	Georgia	30101	Closed	Yes
2015-07-05	Comcast Imposed a New Usage Cap of 300GB that ...	05-07-15	Internet	Acworth	Georgia	30101	Open	Yes
2015-05-26	Comcast not working and no service to boot	26-05-15	Internet	Acworth	Georgia	30101	Solved	No

In [8]:

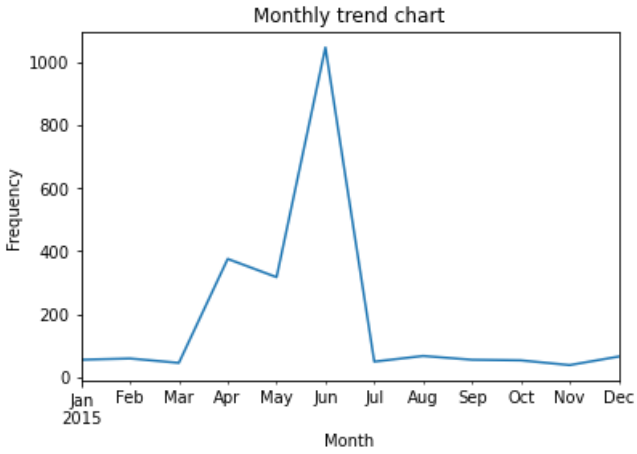
```
#Plotting trend chart for monthly
```

In [14]:

```
df.groupby(pd.Grouper(freq='M')).size().plot()
plt.xlabel('Month')
plt.ylabel('Frequency')
plt.title('Monthly trend chart')
```

Out[14]:

```
Text(0.5, 1.0, 'Monthly trend chart')
```



In [9]:

```
#INSIGHTS: - From the above trend chart, we can clearly see that complaints for the month of June are ma
```

In [10]:

```
#Plotting trend chart for daily
```

In [16]:

```
df['Date'].value_counts()[:8]
```

Out[16]:

```
24-06-15    218
23-06-15    190
25-06-15     98
26-06-15     55
30-06-15     53
29-06-15     51
18-06-15     47
06-12-15     43
Name: Date, dtype: int64
```

In [17]:

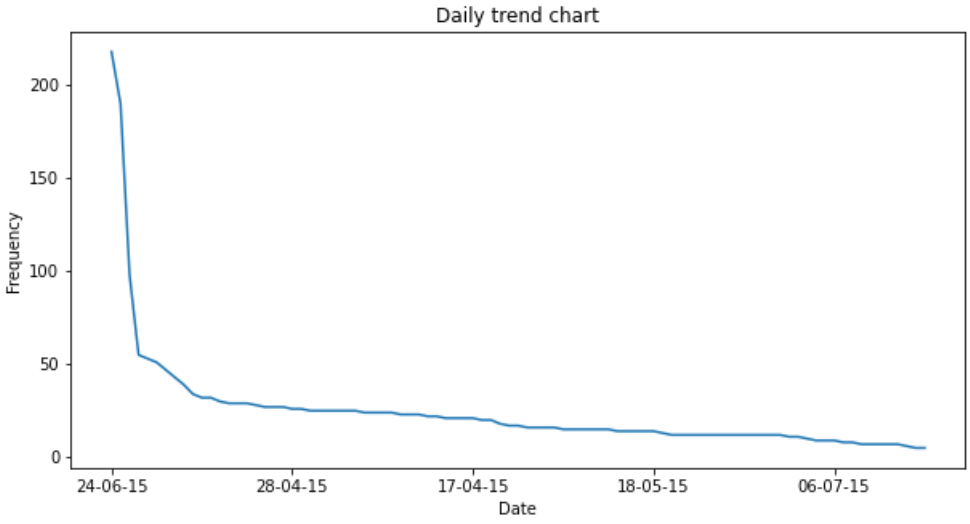
```
#plotting Daily chart

df=df.sort_values(by='Date')
plt.figure(figsize=(10,5))
```

```
df['Date'].value_counts().plot()
plt.xlabel('Date')
plt.ylabel('Frequency')
plt.title('Daily trend chart')
```

Out[17]:

```
Text(0.5, 1.0, 'Daily trend chart')
```



In [18]:

```
# Provide a table with the frequency of complaint types.
```

In [20]:

```
df['Customer Complaint'].value_counts()[:9]
```

Out[20]:

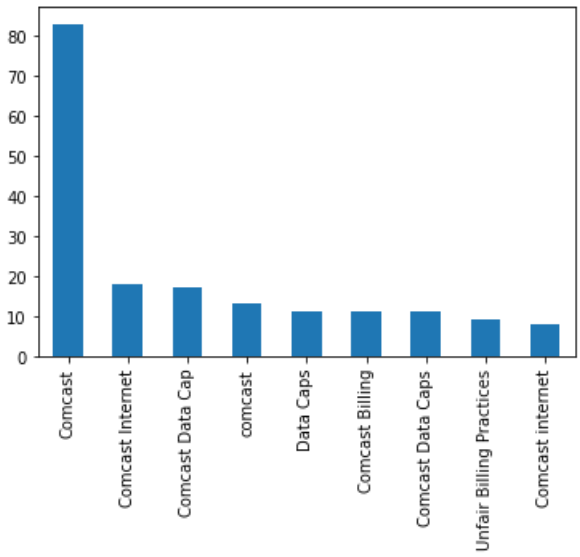
```
Comcast                      83
Comcast Internet             18
Comcast Data Cap             17
comcast                      13
Data Caps                    11
Comcast Billing              11
Comcast Data Caps            11
Unfair Billing Practices      9
Comcast internet              8
Name: Customer Complaint, dtype: int64
```

In [21]:

```
df['Customer Complaint'].value_counts()[:9].plot.bar()
```

Out[21]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x120357a00>
```



In [22]:

```
#Which complaint types are maximum i.e., around internet, network issues, or across any other domains
```

In [23]:

```
df['Customer Complaint'].unique()
```

Out[23]:

```
array(['Fraudulent claims reported to collections agency',
       'Comcast refusal of service', 'Comcast Cable', ...,
       'Comcast of East Windsor NJ Complaint',
       'Complaint against Comcast for incredibly bad service',
       'Questionable internet slowdown'], dtype=object)
```

In [30]:

```
internet_issue1=df[df['Customer Complaint'].str.contains('network')].count()
```

In [31]:

```
internet_issue2=df[df['Customer Complaint'].str.contains('speed')].count()
```

In [32]:

```
internet_issue3=df[df['Customer Complaint'].str.contains('data')].count()
```

In [33]:

```
internet_issue4=df[df['Customer Complaint'].str.contains('internet')].count()
```

In [34]:

```
billing_issue1=df[df['Customer Complaint'].str.contains('billing')].count()
```

In [35]:

```
billing_issue2=df[df['Customer Complaint'].str.contains('charges')].count()
```

In [36]:

```
billing_issue3=df[df['Customer Complaint'].str.contains('bill')].count()
```

In [37]:

```
service_issue1=df[df['Customer Complaint'].str.contains('service')].count()
```

In [38]:

```
service_issue2=df[df['Customer Complaint'].str.contains('customer')].count()
```

In [39]:

```
total_issue_internet=internet_issue1+internet_issue2+internet_issue3+internet_issue4
```

In [40]:

```
total_issue_internet
```

Out[40]:

```
Customer Complaint             374
Date                           374
Received Via                   374
City                           374
State                          374
Zip code                       374
Status                         374
Filing on Behalf of Someone    374
dtype: int64
```

In [41]:

```
total_billing_issues=billing_issue1+billing_issue2+billing_issue3
```

In [42]:

```
total_billing_issues
```

Out[42]:

```
Customer Complaint             353
Date                           353
Received Via                   353
City                           353
State                          353
Zip code                       353
Status                         353
Filing on Behalf of Someone    353
dtype: int64
```

In [43]:

```
total_service_issues=service_issue1+service_issue2
```

In [44]:

```
total_service_issues
```

Out[44]:

```
Customer Complaint             360
Date                           360
Received Via                   360
City                           360
State                          360
Zip code                       360
Status                         360
Filing on Behalf of Someone    360
dtype: int64
```

In [45]:

```
df.shape
```

Out[45]:

```
(2224, 8)
```

In [46]:

```
other_issues=2224-(total_billing_issues+total_issue_internet+total_service_issues)
```

In [47]:

```
other_issues
```

Out[47]:

```
Customer Complaint             1137
Date                           1137
Received Via                   1137
City                           1137
State                          1137
Zip code                       1137
Status                         1137
Filing on Behalf of Someone    1137
dtype: int64
```

In [ ]:

```
#INSIGHTS: - From the above Dataset, we can clearly see that other_issues has maximum complaints.
```

In [48]:

```
# Create a new categorical variable with value as Open and Closed. Open & Pending is to be categorized a.
```

In [ ]:

```
#######---OPEN/CLOSED & OPEN/PENDINNG Complaint categorical -----#####
```

In [49]:

```
df['newStatus']=['Open' if Status=='Open' or Status=='Pending' else 'Closed' for Status in df['Status']]
```

In [52]:

```
#task :Which state has the maximum complaints
```

In [53]:

```
df.groupby(['State']).size().sort_values(ascending=False)[:5]
```

Out[53]:

```
State
Georgia       288
Florida       240
California    220
Illinois      164
Tennessee     143
dtype: int64
```

In [54]:

```
#Provide state wise status of complaints in a stacked bar chart. Use the categorized variable from Q3. P
```

In [55]:

```
state_complain=df.groupby(['State','newStatus']).size().unstack()
```

In [56]:

```
state_complain
```

Out[56]:

newStatus	Closed	Open
State		
Alabama	17.0	9.0
Arizona	14.0	6.0
Arkansas	6.0	NaN
California	159.0	61.0
Colorado	58.0	22.0
Connecticut	9.0	3.0
Delaware	8.0	4.0
District Of Columbia	14.0	2.0
District of Columbia	1.0	NaN
Florida	201.0	39.0
Georgia	208.0	80.0
Illinois	135.0	29.0
Indiana	50.0	9.0
Iowa	1.0	NaN
Kansas	1.0	1.0
Kentucky	4.0	3.0
Louisiana	12.0	1.0
Maine	3.0	2.0
Maryland	63.0	15.0
Massachusetts	50.0	11.0
Michigan	92.0	23.0
Minnesota	29.0	4.0
Mississippi	23.0	16.0
Missouri	3.0	1.0
Montana	1.0	NaN
Nevada	1.0	NaN
New Hampshire	8.0	4.0
New Jersey	56.0	19.0
New Mexico	11.0	4.0
New York	6.0	NaN
North Carolina	3.0	NaN
Ohio	3.0	NaN
Oregon	36.0	13.0
Pennsylvania	110.0	20.0
Rhode Island	1.0	NaN
South Carolina	15.0	3.0
Tennessee	96.0	47.0
Texas	49.0	22.0
Utah	16.0	6.0
Vermont	2.0	1.0
Virginia	49.0	11.0
Washington	75.0	23.0
West Virginia	8.0	3.0

In [ ]:

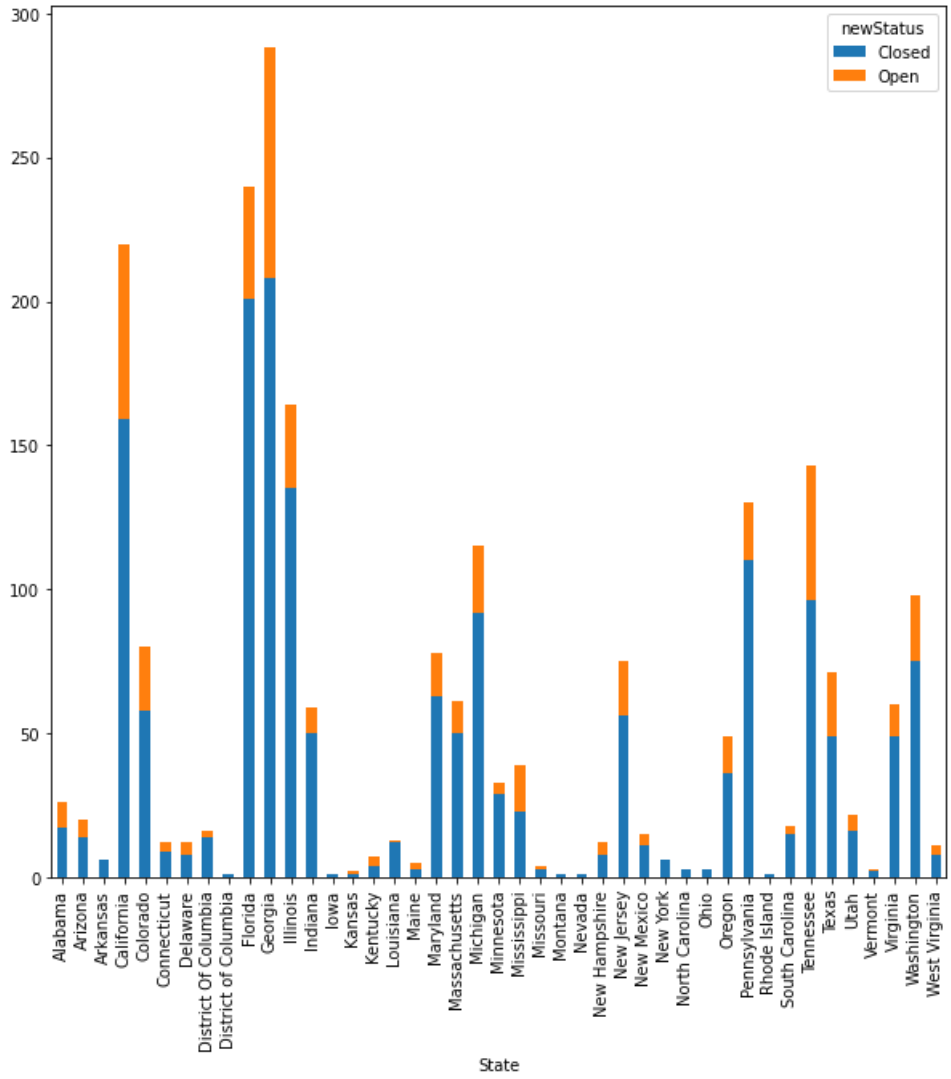
```
######----Plotting on stacked bar chart-----#########
```

In [57]:

```
state_complain.plot.bar(figsize=(10,10),stacked=True)
```

Out[57]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x11fb03100>
```



In [58]:

```
#Which state has the highest percentage of unresolved complaints
```

In [59]:

```
df.newStatus.value_counts()
```

Out[59]:

```
Closed    1707
Open       517
Name: newStatus, dtype: int64
```

In [61]:

```
unresolved_data=df.groupby(['State','newStatus']).size().unstack().fillna(0).sort_values(by='Open',ascend
```

In [62]:

```
unresolved_data
```

Out[62]:

newStatus	Closed	Open
State		
Georgia	208.0	80.0
California	159.0	61.0
Tennessee	96.0	47.0
Florida	201.0	39.0
Illinois	135.0	29.0
Washington	75.0	23.0
Michigan	92.0	23.0
Colorado	58.0	22.0
Texas	49.0	22.0
Pennsylvania	110.0	20.0
New Jersey	56.0	19.0
Mississippi	23.0	16.0
Maryland	63.0	15.0
Oregon	36.0	13.0
Virginia	49.0	11.0
Massachusetts	50.0	11.0
Alabama	17.0	9.0
Indiana	50.0	9.0
Utah	16.0	6.0
Arizona	14.0	6.0
New Hampshire	8.0	4.0
New Mexico	11.0	4.0
Minnesota	29.0	4.0
Delaware	8.0	4.0
West Virginia	8.0	3.0
Connecticut	9.0	3.0
Kentucky	4.0	3.0
South Carolina	15.0	3.0
Maine	3.0	2.0
District Of Columbia	14.0	2.0
Kansas	1.0	1.0
Vermont	2.0	1.0
Missouri	3.0	1.0
Louisiana	12.0	1.0
Montana	1.0	0.0
Rhode Island	1.0	0.0
Ohio	3.0	0.0
District of Columbia	1.0	0.0
North Carolina	3.0	0.0
New York	6.0	0.0
Nevada	1.0	0.0
Arkansas	6.0	0.0
Iowa	1.0	0.0

In [63]:

```
unresolved_data['unresolved_cmp_prct']=unresolved_data['Open']/unresolved_data['Open'].sum()*100
```

In [64]:

```
unresolved_data
```

Out[64]:

newStatus	Closed	Open	unresolved_cmp_prct
State			
Georgia	208.0	80.0	15.473888
California	159.0	61.0	11.798839
Tennessee	96.0	47.0	9.090909
Florida	201.0	39.0	7.543520
Illinois	135.0	29.0	5.609284
Washington	75.0	23.0	4.448743
Michigan	92.0	23.0	4.448743
Colorado	58.0	22.0	4.255319
Texas	49.0	22.0	4.255319
Pennsylvania	110.0	20.0	3.868472
New Jersey	56.0	19.0	3.675048
Mississippi	23.0	16.0	3.094778
Maryland	63.0	15.0	2.901354
Oregon	36.0	13.0	2.514507
Virginia	49.0	11.0	2.127660
Massachusetts	50.0	11.0	2.127660
Alabama	17.0	9.0	1.740812
Indiana	50.0	9.0	1.740812
Utah	16.0	6.0	1.160542
Arizona	14.0	6.0	1.160542
New Hampshire	8.0	4.0	0.773694
New Mexico	11.0	4.0	0.773694
Minnesota	29.0	4.0	0.773694
Delaware	8.0	4.0	0.773694
West Virginia	8.0	3.0	0.580271
Connecticut	9.0	3.0	0.580271
Kentucky	4.0	3.0	0.580271
South Carolina	15.0	3.0	0.580271
Maine	3.0	2.0	0.386847
District Of Columbia	14.0	2.0	0.386847
Kansas	1.0	1.0	0.193424
Vermont	2.0	1.0	0.193424
Missouri	3.0	1.0	0.193424
Louisiana	12.0	1.0	0.193424
Montana	1.0	0.0	0.000000
Rhode Island	1.0	0.0	0.000000
Ohio	3.0	0.0	0.000000
District of Columbia	1.0	0.0	0.000000
North Carolina	3.0	0.0	0.000000
New York	6.0	0.0	0.000000
Nevada	1.0	0.0	0.000000
Arkansas	6.0	0.0	0.000000
Iowa	1.0	0.0	0.000000

In [ ]:

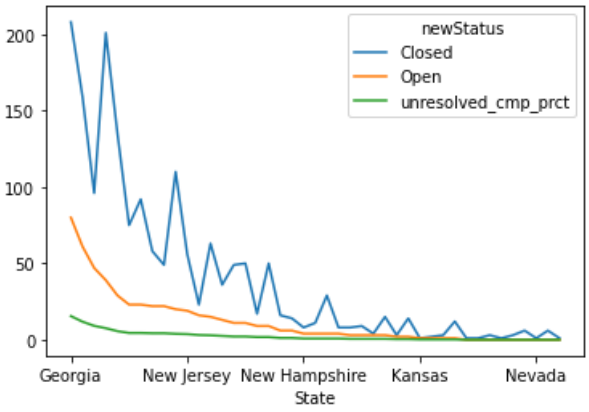
```
#INSIGHTS: - From the above chart, we can clearly see that Georgia has maximum complaints.
```

In [65]:

```
unresolved_data.plot()
```

Out[65]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1233ae460>
```



In [66]:

```
#Provide the percentage of complaints resolved till date, which were received through the Internet and c
```

In [68]:

```
resolved_data=df.groupby(['Received Via','newStatus']).size().unstack().fillna(0)
```

In [69]:

```
resolved_data['resolved']=resolved_data['Closed']/resolved_data['Closed'].sum()*100
```

In [70]:

```
resolved_data
```

Out[70]:

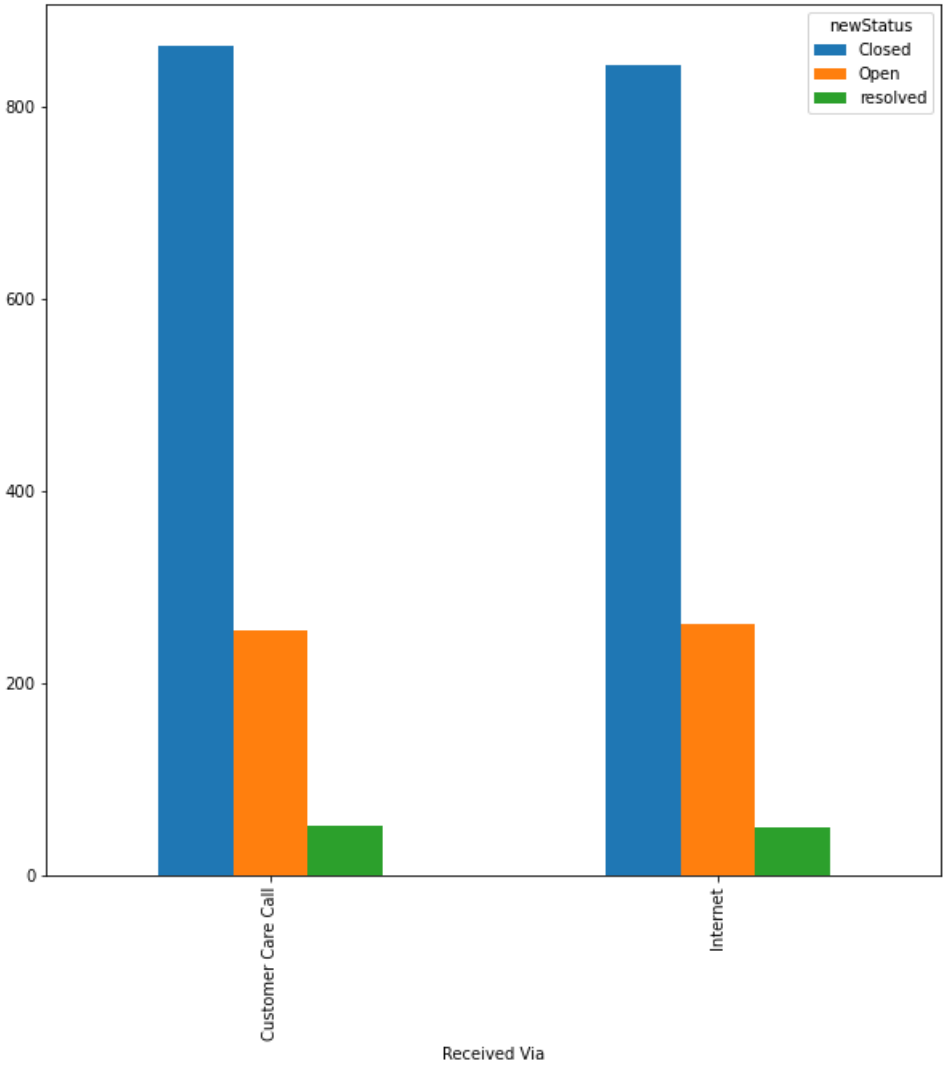
newStatus	Closed	Open	resolved
Received Via			
Customer Care Call	864	255	50.615114
Internet	843	262	49.384886

In [71]:

```
resolved_data.plot(kind='bar',figsize=(10,10))
```

Out[71]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x125025400>
```



In [ ]: