Comcast Telecom Consumer Complaints

In [50]:
```
# importing packages

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

%matplotlib inline
```

Working Directory

In [8]:
```
pwd
```

Out[8]:
```
'C:\\Users\\aruls\\OneDrive\\Documents\\data\\SimpliLearn'
```

Analysing the Dataset

In [9]:
```
#Importing the Comcast Telecom Dataset
df=pd.read_csv('C:\\Users\\aruls\\OneDrive\\Documents\\data\\SimpliLearn\\Comcast_telecom_complaints_data_2.csv')
```

In [10]:
```
#Viewing the dataset
df
```

Out[10]:

	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
...	...	...	...	...	...	...	...	...	...	...	...
2219	213550	Service Availability	04-02-15	04-Feb-15	9:13:18 AM	Customer Care Call	Youngstown	Florida	32466	Closed	No
2220	318775	Comcast Monthly Billing for Returned Modem	06-02-15	06-Feb-15	1:24:39 PM	Customer Care Call	Ypsilanti	Michigan	48197	Solved	No
2221	331188	complaint about comcast	06-09-15	06-Sep-15	5:28:41 PM	Internet	Ypsilanti	Michigan	48197	Solved	No
2222	360489	Extremely unsatisfied Comcast customer	23-06-15	23-Jun-15	11:13:30 PM	Customer Care Call	Ypsilanti	Michigan	48197	Solved	No
2223	363614	Comcast, Ypsilanti MI Internet Speed	24-06-15	24-Jun-15	10:28:33 PM	Customer Care Call	Ypsilanti	Michigan	48198	Open	Yes

2224 rows × 11 columns

In [11]:
```
# shape of dataset
df.shape
```

Out[11]:
```
(2224, 11)
```

11 columns , 2224 rows

In [12]:
```
# first five rows
df.head(5)
```

Out[12]:

	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 [13]:
```
#columns
df.columns
```

Out[13]:
```
Index(['Ticket #', 'Customer Complaint', 'Date', 'Date_month_year', 'Time',
       'Received Via', 'City', 'State', 'Zip code', 'Status',
       'Filing on Behalf of Someone'],
      dtype='object')
```

In [14]:
```
#Data Type of the columns
df.dtypes
```

Out[14]:
```
Ticket #                      object
Customer Complaint            object
Date                          object
Date_month_year               object
Time                          object
Received Via                  object
City                          object
State                         object
Zip code                       int64
Status                        object
Filing on Behalf of Someone   object
dtype: object
```

In [15]:
```
type(df["Date"][0])
#date is stored as string
```

Out[15]:
```
str
```

In [16]:
```
type(df['Date_month_year'][0])
#month is also stored as string
```

Out[16]:
```
str
```

In [17]:
```
#converting column to date format and making new dataset with index date time (this is for ploting ease)

df["Date"] = df["Date_month_year"] + " " + df["Time"]
```

In [18]:
```
df["Date"] = pd.to_datetime(df["Date"])
```

In [19]:
```
#checking date column

type(df["Date"][0])
```

Out[19]:
```
pandas._libs.tslibs.timestamps.Timestamp
```

In [20]:
```
#converting column to date format
df["Date_month_year"] = pd.to_datetime(df["Date_month_year"])
```

In [21]:
```
#checking date column

type(df["Date_month_year"][0])
```

Out[21]:
```
pandas._libs.tslibs.timestamps.Timestamp
```

In [22]:
```
#sorting data frame according to date

df_sorted_bydate = df.sort_values(by="Date")
```

In [23]:
```
df_sorted_bydate
```

Out[23]:

	Ticket #	Customer Complaint	Date	Date_month_year	Time	Received Via	City	State	Zip code	Status	Filing on Behalf of Someone
1852	211255	Comcast harassment	2015-01-04 00:18:47	2015-01-04	12:18:47 AM	Customer Care Call	Schaumburg	Illinois	60193	Closed	No
1160	211472	comcast cable	2015-01-04 10:43:20	2015-01-04	10:43:20 AM	Customer Care Call	Lockport	Illinois	60441	Closed	No
1430	211478	Comcast	2015-01-04 10:47:35	2015-01-04	10:47:35 AM	Internet	North Huntingdon	Pennsylvania	15642	Closed	No
2144	211677	Comcast refusal of service	2015-01-04 12:01:06	2015-01-04	12:01:06 PM	Customer Care Call	Wayne	Pennsylvania	19087	Closed	No
1237	211775	Horrible Service	2015-01-04 12:28:58	2015-01-04	12:28:58 PM	Customer Care Call	Mckeesport	Pennsylvania	15132	Closed	No
...	...	...	...	...	...	...	...	...	...	...	...
1798	338467	Cable internet unavailable	2015-12-06 21:18:18	2015-12-06	9:18:18 PM	Internet	San Mateo	California	94402	Closed	No
440	338507	Comcast Internet, cable, and phone outtages	2015-12-06 21:46:12	2015-12-06	9:46:12 PM	Customer Care Call	Clarkston	Michigan	48346	Solved	No
1363	338510	Comcast	2015-12-06 21:51:40	2015-12-06	9:51:40 PM	Internet	Muskegon	Michigan	49445	Solved	No
5	338519	ISP Charging for arbitrary data limits with ov...	2015-12-06 21:59:40	2015-12-06	9:59:40 PM	Internet	Acworth	Georgia	30101	Solved	No
441	338606	Internet connection outage	2015-12-06 23:52:11	2015-12-06	11:52:11 PM	Customer Care Call	Clarkston	Michigan	48346	Solved	No

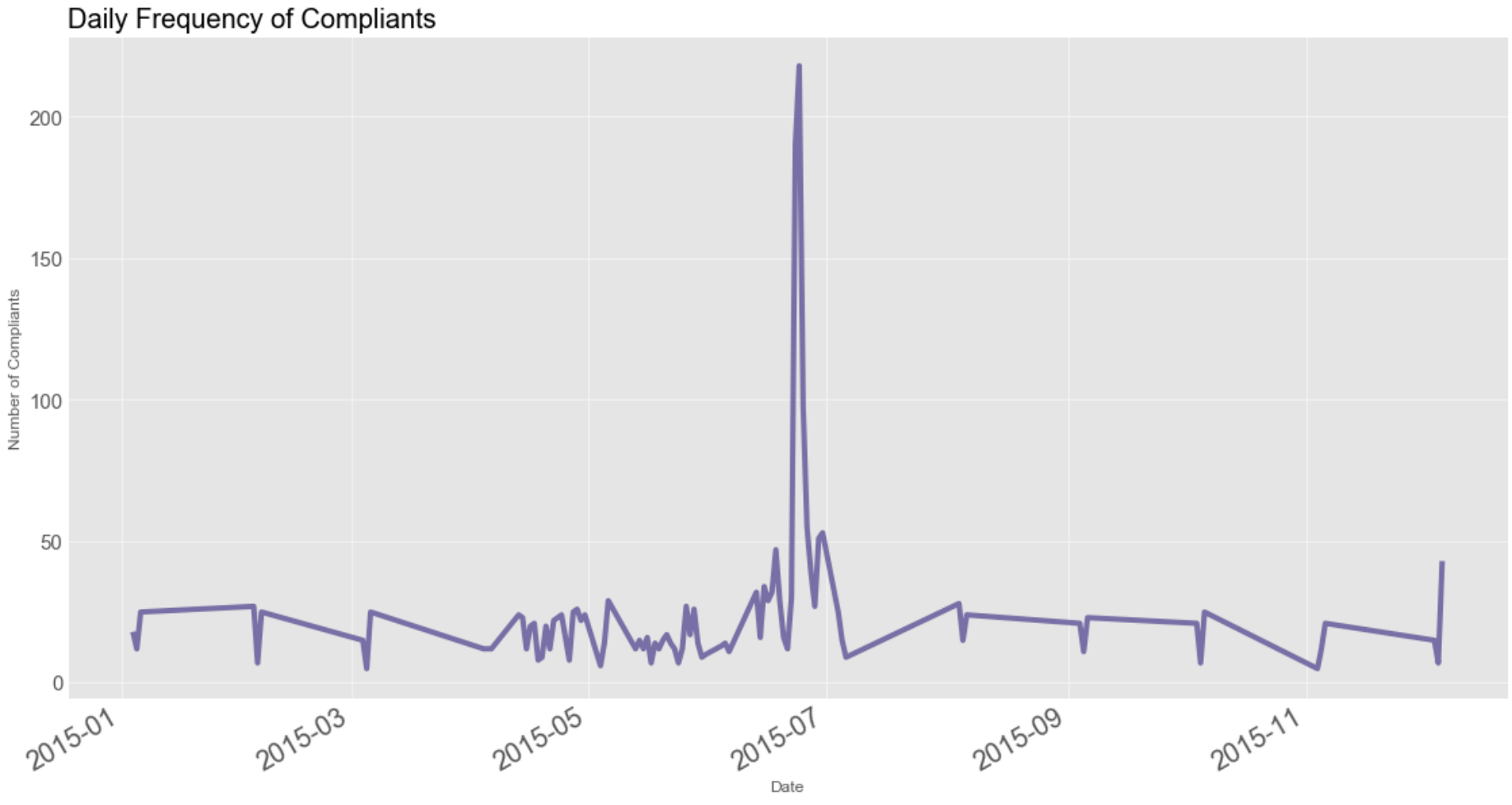
2224 rows × 11 columns

In [24]:
```
# Counting the compliants on a perticular date
df_trend=df_sorted_bydate["Date_month_year"].value_counts()
```

Trend chart for the number of complaints by daily granularity levels

In [125]:
```
plt.style.use('ggplot')
df_trend.plot(figsize=(17,10),linestyle='-' ,linewidth=4 ,color="darkslateblue",alpha=0.7)
plt.title("Daily Frequency of Compliants", loc='left', fontsize=20, fontweight=6, color='Black')
plt.yticks(fontsize=15)
plt.xticks(fontsize=20)
plt.xlabel("Date")
plt.ylabel("Number of Compliants")
```

Out[125]:
```
Text(0, 0.5, 'Number of Compliants')
```



In [26]:
```
#setting inedx of dataset as Date (this is for the ease of plotting)
df_date_asIndex = df.set_index(df["Date"])
```

In [27]:
```
df_date_asIndex
```

Out[27]:

Date	Ticket #	Customer Complaint	Date	Date_month_year	Time	Received Via	City	State	Zip code	Status	Filing on Behalf of Someone
2015-04-22 15:53:50	250635	Comcast Cable Internet Speeds	2015-04-22 15:53:50	2015-04-22	3:53:50 PM	Customer Care Call	Abingdon	Maryland	21009	Closed	No
2015-08-04 10:22:56	223441	Payment disappear - service got disconnected	2015-08-04 10:22:56	2015-08-04	10:22:56 AM	Internet	Acworth	Georgia	30102	Closed	No
2015-04-18 09:55:47	242732	Speed and Service	2015-04-18 09:55:47	2015-04-18	9:55:47 AM	Internet	Acworth	Georgia	30101	Closed	Yes
2015-07-05 11:59:35	277946	Comcast Imposed a New Usage Cap of 300GB that ...	2015-07-05 11:59:35	2015-07-05	11:59:35 AM	Internet	Acworth	Georgia	30101	Open	Yes
2015-05-26 13:25:26	307175	Comcast not working and no service to boot	2015-05-26 13:25:26	2015-05-26	1:25:26 PM	Internet	Acworth	Georgia	30101	Solved	No
...	...	...	...	...	...	...	...	...	...	...	...
2015-02-04 09:13:18	213550	Service Availability	2015-02-04 09:13:18	2015-02-04	9:13:18 AM	Customer Care Call	Youngstown	Florida	32466	Closed	No
2015-02-06 13:24:39	318775	Comcast Monthly Billing for Returned Modem	2015-02-06 13:24:39	2015-02-06	1:24:39 PM	Customer Care Call	Ypsilanti	Michigan	48197	Solved	No
2015-09-06 17:28:41	331188	complaint about comcast	2015-09-06 17:28:41	2015-09-06	5:28:41 PM	Internet	Ypsilanti	Michigan	48197	Solved	No
2015-06-23 23:13:30	360489	Extremely unsatisfied Comcast customer	2015-06-23 23:13:30	2015-06-23	11:13:30 PM	Customer Care Call	Ypsilanti	Michigan	48197	Solved	No
2015-06-24 22:28:33	363614	Comcast, Ypsilanti MI Internet Speed	2015-06-24 22:28:33	2015-06-24	10:28:33 PM	Customer Care Call	Ypsilanti	Michigan	48198	Open	Yes

2224 rows × 11 columns

In [28]:
```
df_date_asIndex.groupby(pd.Grouper(freq="M")).size()
```

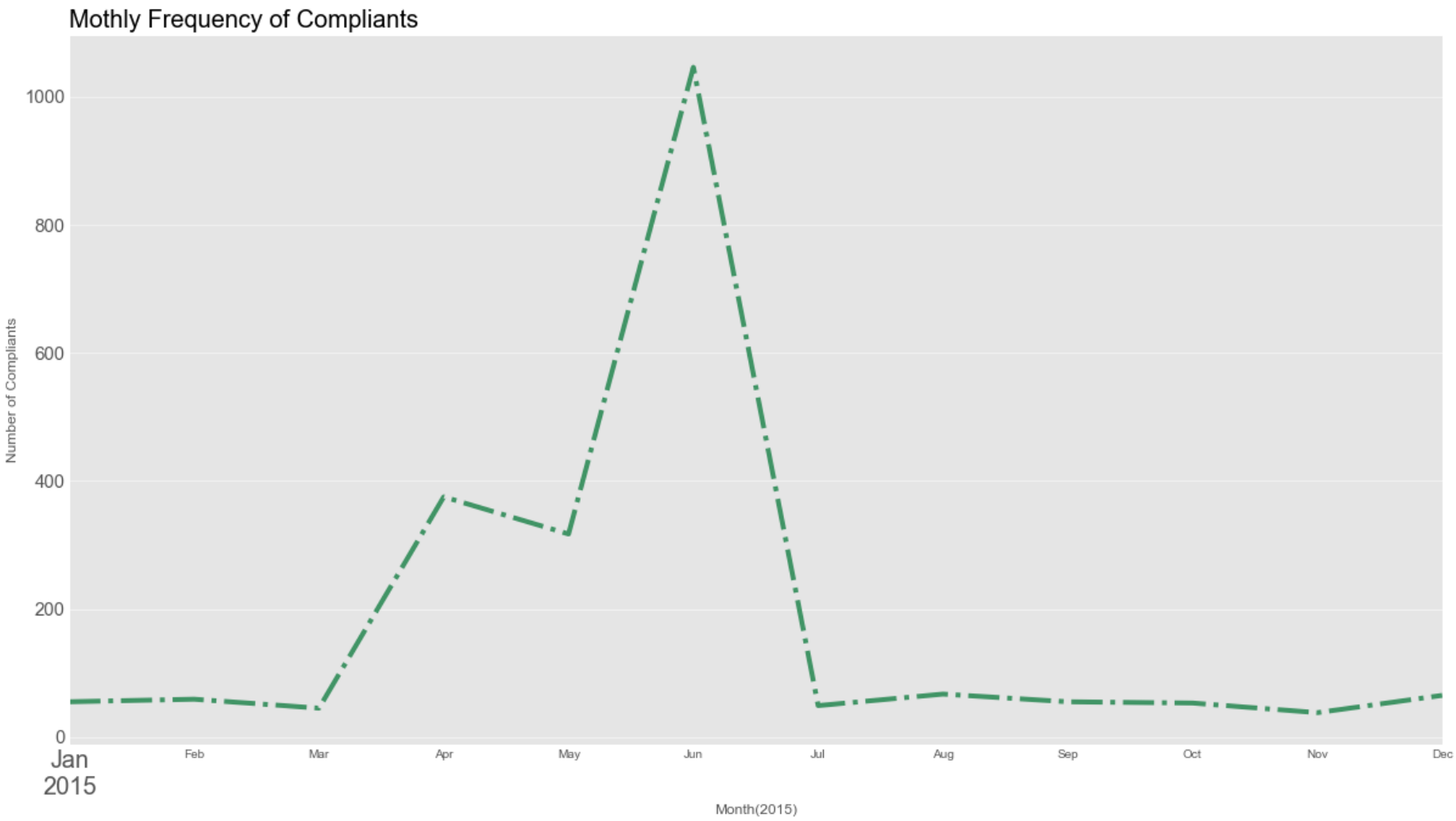
Out[28]:
```
Date
2015-01-31      55
2015-02-28      59
2015-03-31      45
2015-04-30     375
2015-05-31     317
2015-06-30    1046
2015-07-31      49
2015-08-31      67
2015-09-30      55
2015-10-31      53
2015-11-30      38
2015-12-31      65
Freq: M, dtype: int64
```

Trend chart for the number of complaints by monthly granularity levels

In [124]:
```
plt.style.use('ggplot')
df_date_asIndex.groupby(pd.Grouper(freq="M")).size().plot(figsize=(18,10),linestyle='-.' ,linewidth=4 ,color="seagreen",a

plt.title("Mothly Frequency of Compliants", loc='left', fontsize=20, fontweight=6, color='Black')
plt.yticks(fontsize=15)
plt.xticks(fontsize=20)
plt.xlabel("Month(2015)")
plt.ylabel("Number of Compliants")
```

Out[124]: Text(0, 0.5, 'Number of Compliants')



In [30]:
```
df.groupby('Status').size()
```

Out[30]:
```
Status
Closed     734
Open       363
Pending    154
Solved     973
dtype: int64
```

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.

In [31]:
```
df['Status_New']=["Open" if x=="Open" or x=="Pending" else "Closed" for x in df["Status"]]
```

In [32]:
```
df.groupby('Status_New').size()
```

Out[32]:
```
Status_New
Closed    1707
Open       517
dtype: int64
```

In [33]:
```
#checking the new column
df["Status_New"]
```

Out[33]:
```
0       Closed
1       Closed
2       Closed
3         Open
4       Closed
         ...
2219    Closed
2220    Closed
2221    Closed
2222    Closed
2223      Open
Name: Status_New, Length: 2224, dtype: object
```

Percentage of complaints resolved till date, which were received through the Internet and customer care calls.

In [130]:
```
Total_Cases=2224
Resolved=1707

percentage=100*(Resolved/Total_Cases)

print(percentage)
```

```
76.75359712230215
```

In [34]:
```
#Grouping by State
state_complaints=df.groupby("State").size()
```

Which state has the maximum complaints

In [35]:
```
state_complaints.sort_values(ascending=False)

# The state of Georgia has the max compliants
```

Out[35]:
```
State
Georgia                 288
Florida                 240
California              220
Illinois                164
Tennessee               143
Pennsylvania            130
Michigan                115
Washington               98
Colorado                 80
Maryland                 78
New Jersey               75
Texas                    71
Massachusetts            61
Virginia                 60
Indiana                  59
Oregon                   49
Mississippi              39
Minnesota                33
Alabama                  26
Utah                     22
Arizona                  20
South Carolina           18
District Of Columbia     16
New Mexico               15
Louisiana                13
New Hampshire            12
Connecticut              12
Delaware                 12
West Virginia            11
Kentucky                  7
New York                  6
Arkansas                  6
Maine                     5
Missouri                  4
North Carolina            3
Vermont                   3
Ohio                      3
Kansas                    2
District of Columbia      1
Rhode Island              1
Montana                   1
Iowa                      1
Nevada                    1
dtype: int64
```

In [36]:
```
State_Unresolved=df.groupby(['State','Status_New']).size()
```

In [37]:
```
State_Unresolved
```

Out[37]:
```
State          Status_New
Alabama        Closed        17
               Open           9
Arizona        Closed        14
               Open           6
Arkansas       Closed         6
                             ..
Virginia       Open          11
Washington     Closed        75
               Open          23
West Virginia  Closed         8
               Open           3
Length: 77, dtype: int64
```

In [38]:
```
State_Unresolved=State_Unresolved.unstack().fillna(0)
```

Which state has the highest percentage of unresolved complaints

In [39]:
```
State_Unresolved.sort_values(by='Open',ascending=False)

# The State of Georgia has the highest percentage of unresolved complaints
```

Out[39]:

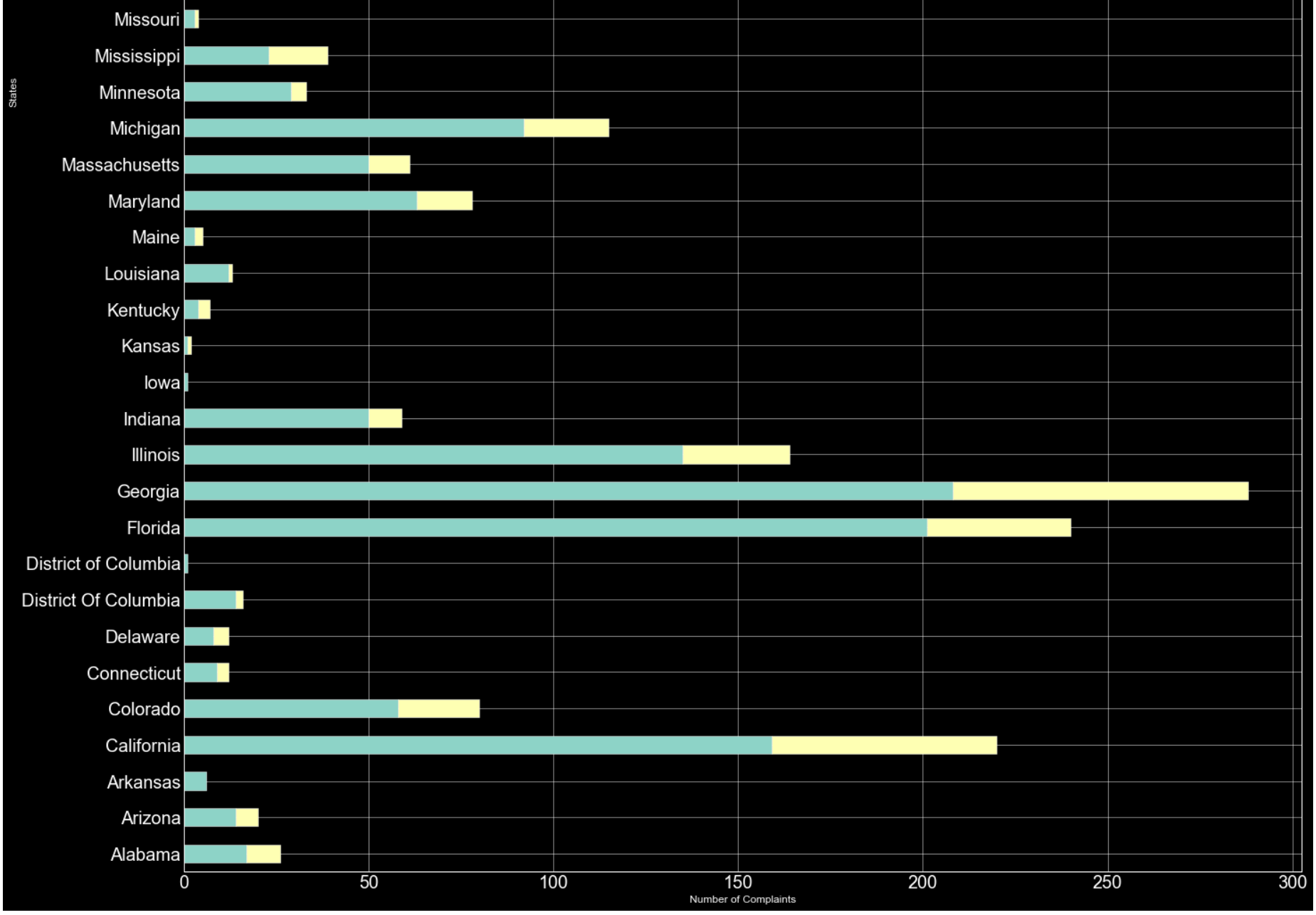
Status_New	Closed	Open
State		
Georgia	208.0	80.0
California	159.0	61.0
Tennessee	96.0	47.0
Florida	201.0	39.0
Illinois	135.0	29.0
Washington	75.0	23.0
Michigan	92.0	23.0
Colorado	58.0	22.0
Texas	49.0	22.0
Pennsylvania	110.0	20.0
New Jersey	56.0	19.0
Mississippi	23.0	16.0
Maryland	63.0	15.0
Oregon	36.0	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

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

In [122]:
```
plt.style.use('dark_background')
State_Unresolved.plot(kind="barh", figsize=(20,30), stacked=True).legend(
    loc='upper right', ncol=2, title="Complaint Status",title_fontsize=20,fontsize="xx-large"
)
plt.title("Ratio of Open and Closed Complaints ", loc='left', fontsize=25, fontweight=6, color='White')
plt.yticks(fontsize=20)
plt.xticks(fontsize=20)
plt.xlabel("Number of Complaints")

plt.ylabel("States")
```

Out[122]: Text(0, 0.5, 'States')



In [ ]: