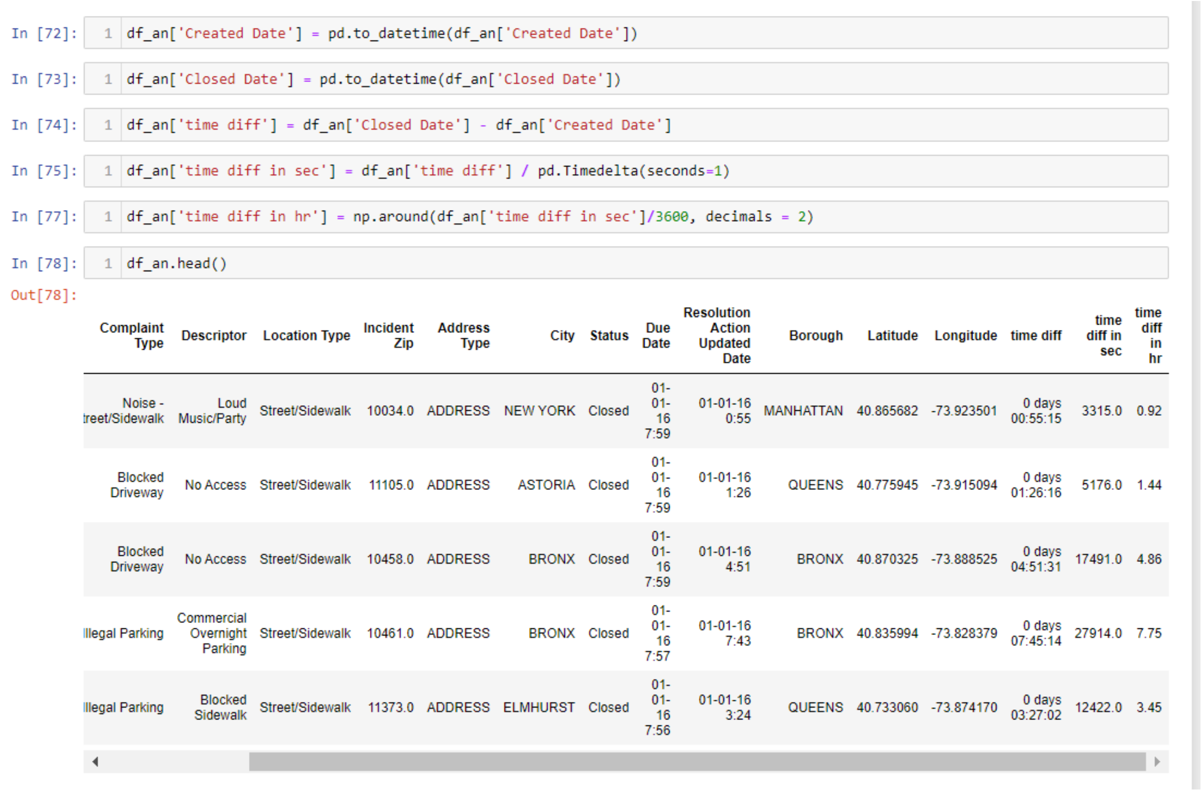
Finding Request Closing Time

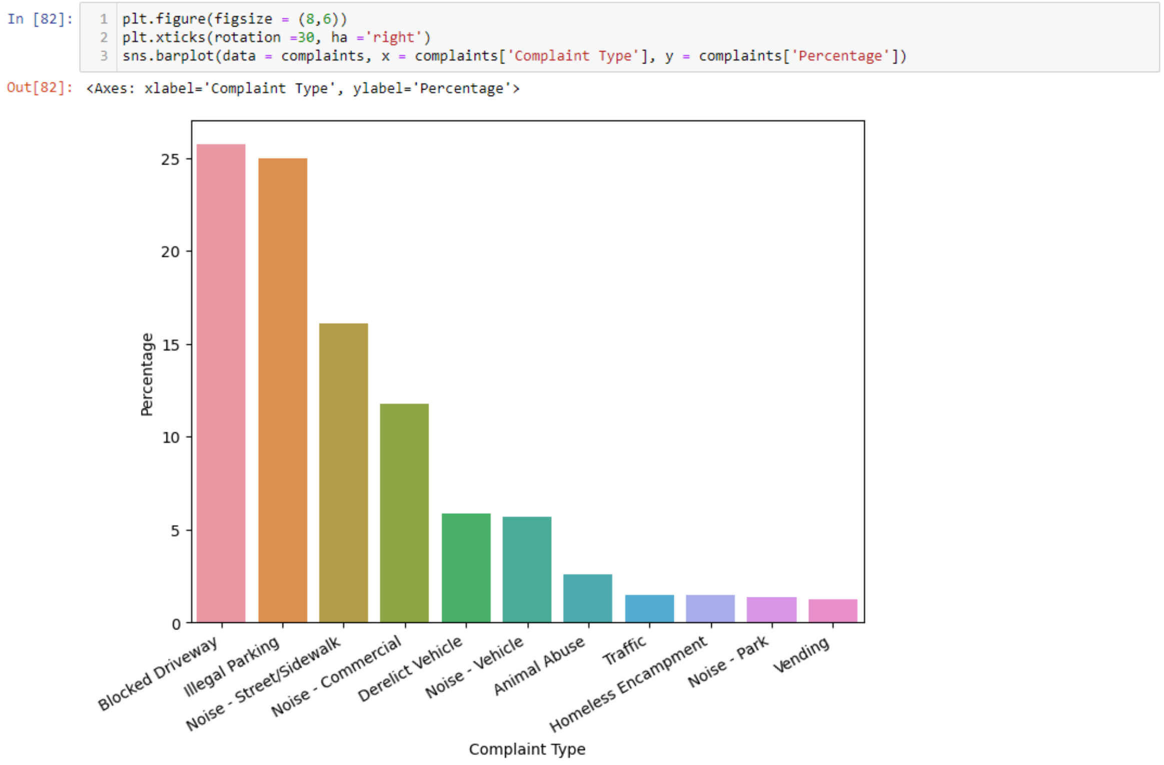


```
In [72]:  df_an['Created Date'] = pd.to_datetime(df_an['Created Date'])

In [73]:  df_an['Closed Date'] = pd.to_datetime(df_an['Closed Date'])

In [74]:  df_an['time diff'] = df_an['Closed Date'] - df_an['Created Date']

In [75]:  df_an['time diff in sec'] = df_an['time diff'] / pd.Timedelta(seconds=1)

In [77]:  df_an['time diff in hr'] = np.around(df_an['time diff in sec']/3600, decimals = 2)

In [78]:  df_an.head()
```

Out[78]:

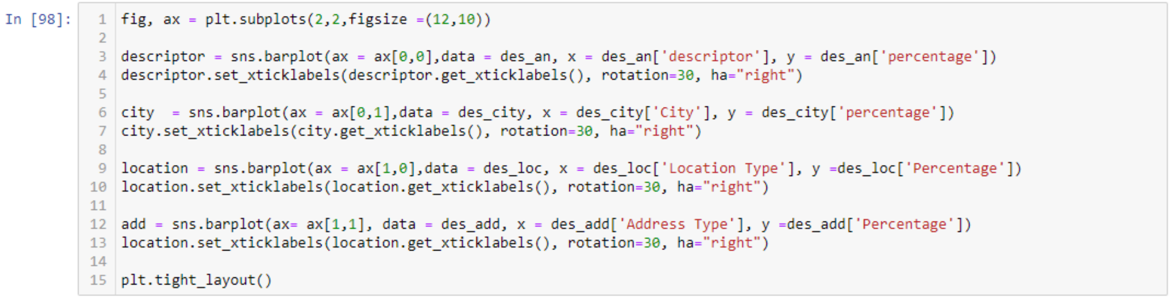
Complaint Type	Descriptor	Location Type	Incident Zip	Address Type	City	Status	Due Date	Resolution Action Updated Date	Borough	Latitude	Longitude	time diff	time diff in sec	time diff in hr
Noise - treet/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	ADDRESS	NEW YORK	Closed	01-01-16 7:59	01-01-16 0:55	MANHATTAN	40.865682	-73.923501	0 days 00:55:15	3315.0	0.92
Blocked Driveway	No Access	Street/Sidewalk	11105.0	ADDRESS	ASTORIA	Closed	01-01-16 7:59	01-01-16 1:26	QUEENS	40.775945	-73.915094	0 days 01:26:16	5176.0	1.44
Blocked Driveway	No Access	Street/Sidewalk	10458.0	ADDRESS	BRONX	Closed	01-01-16 7:59	01-01-16 4:51	BRONX	40.870325	-73.888525	0 days 04:51:31	17491.0	4.86
llegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	ADDRESS	BRONX	Closed	01-01-16 7:57	01-01-16 7:43	BRONX	40.835994	-73.828379	0 days 07:45:14	27914.0	7.75
llegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	ADDRESS	ELMHURST	Closed	01-01-16 7:56	01-01-16 3:24	QUEENS	40.733060	-73.874170	0 days 03:27:02	12422.0	3.45

Percentage of complaint types

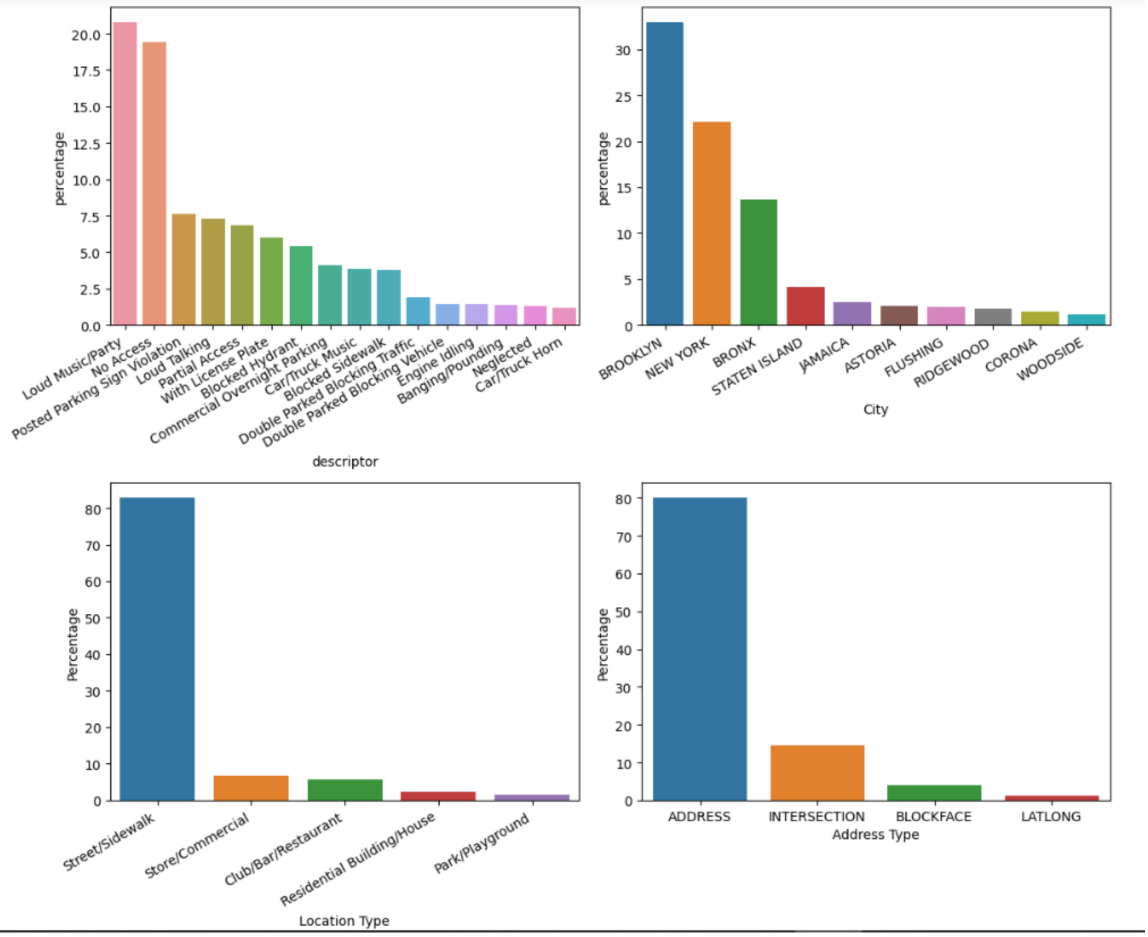


```
In [82]:  plt.figure(figsize = (8,6))
          plt.xticks(rotation =30, ha ='right')
          sns.barplot(data = complaints, x = complaints['Complaint Type'], y = complaints['Percentage'])

Out[82]:  <Axes: xlabel='Complaint Type', ylabel='Percentage'>
```

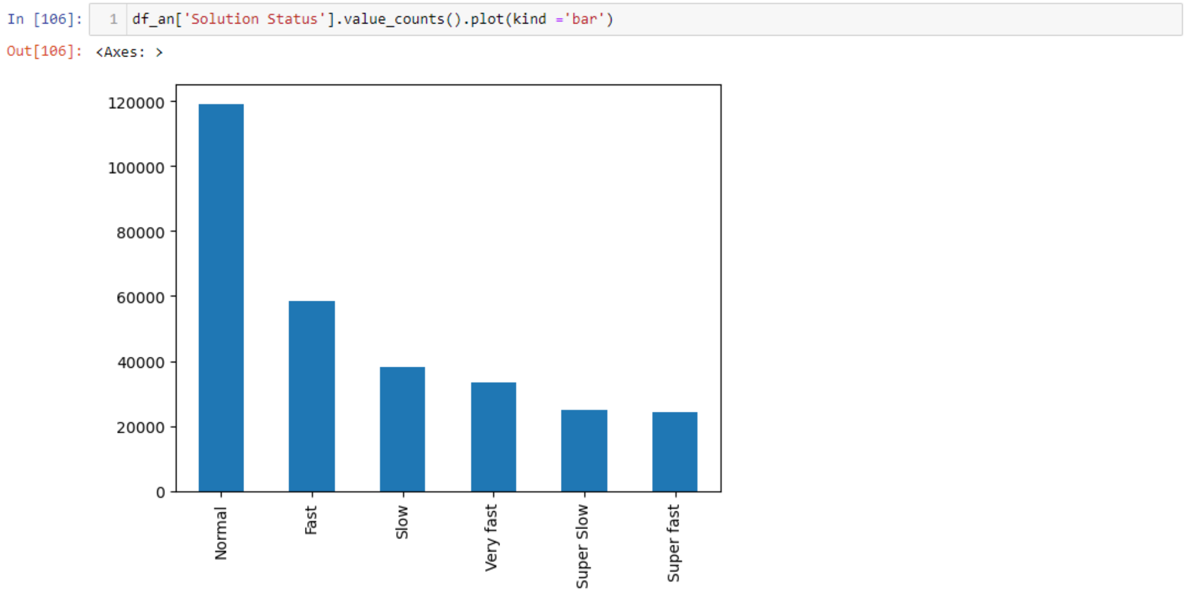
Plotting percentage of different features



```
In [98]:
fig, ax = plt.subplots(2,2,figsize =(12,10))

descriptor = sns.barplot(ax = ax[0,0],data = des_an, x = des_an['descriptor'], y = des_an['percentage'])
descriptor.set_xticklabels(descriptor.get_xticklabels(), rotation=30, ha="right")

city  = sns.barplot(ax = ax[0,1],data = des_city, x = des_city['City'], y = des_city['percentage'])
city.set_xticklabels(city.get_xticklabels(), rotation=30, ha="right")

location = sns.barplot(ax = ax[1,0],data = des_loc, x = des_loc['Location Type'], y =des_loc['Percentage'])
location.set_xticklabels(location.get_xticklabels(), rotation=30, ha="right")

add = sns.barplot(ax= ax[1,1], data = des_add, x = des_add['Address Type'], y =des_add['Percentage'])
location.set_xticklabels(location.get_xticklabels(), rotation=30, ha="right")

plt.tight_layout()
```

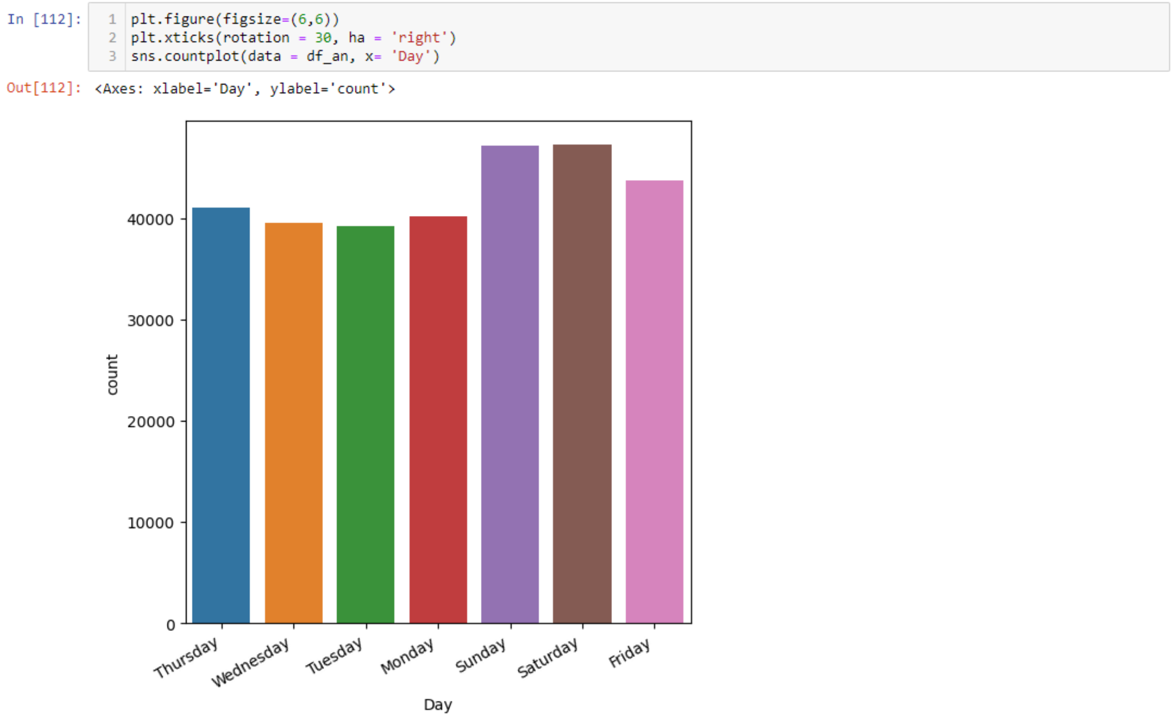


Labeled complaints according to their name



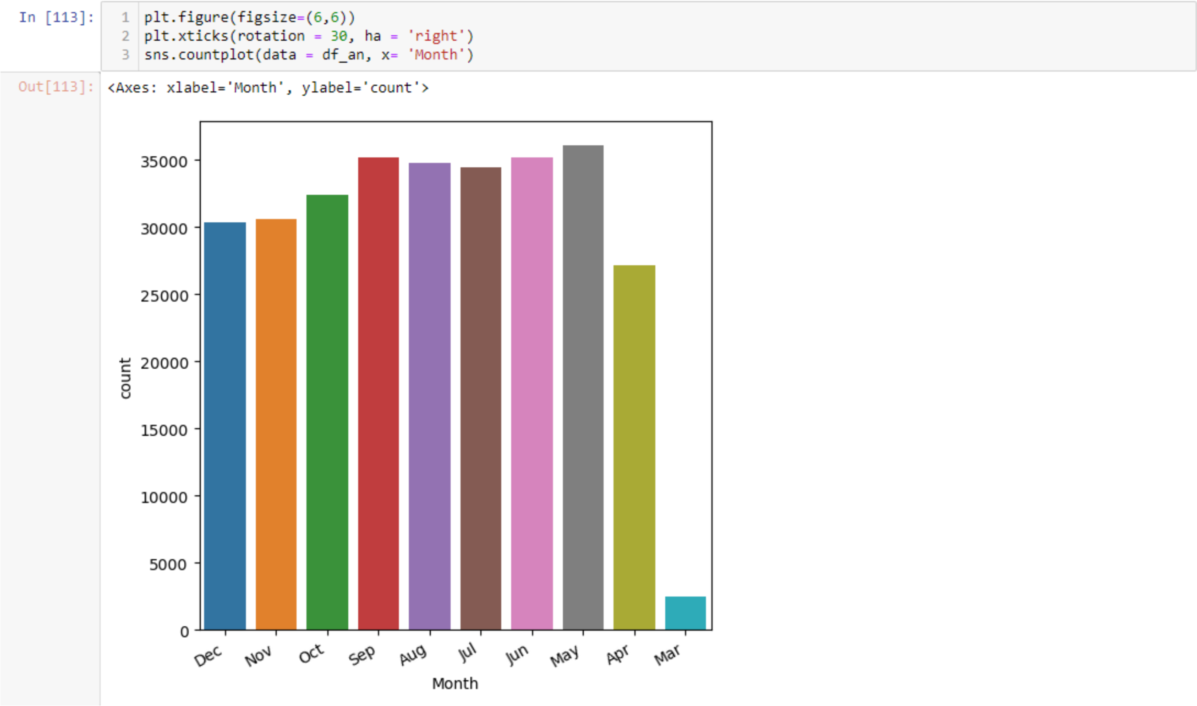
```
In [106]: df_an['Solution Status'].value_counts().plot(kind ='bar')
Out[106]: <Axes: >
```

Complaints according to days



```
In [112]: plt.figure(figsize=(6,6))
          plt.xticks(rotation = 30, ha = 'right')
          sns.countplot(data = df_an, x= 'Day')
Out[112]: <Axes: xlabel='Day', ylabel='count'>
```

Complaints according to months



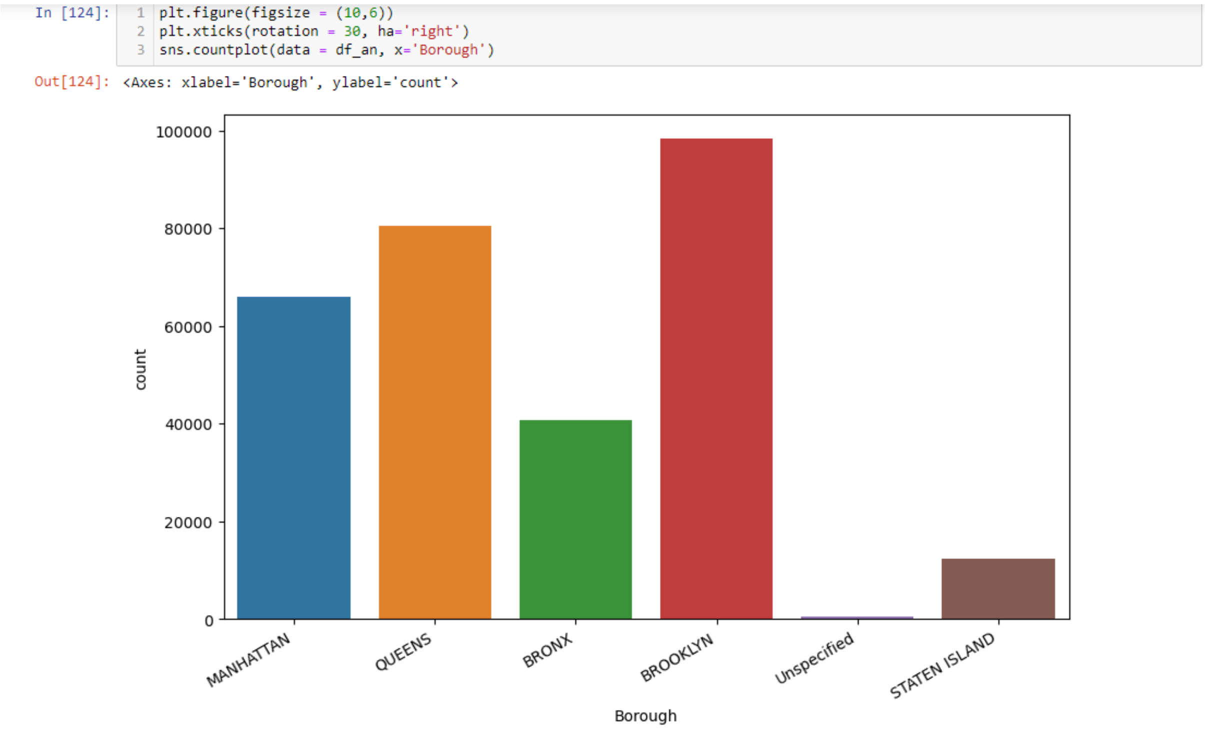
In [113]:

```
plt.figure(figsize=(6,6))
plt.xticks(rotation = 30, ha = 'right')
sns.countplot(data = df_an, x= 'Month')
```

Out[113]:

```
<Axes: xlabel='Month', ylabel='count'>
```

Complaints according to Boroughs



In [124]:

```
plt.figure(figsize = (10,6))
plt.xticks(rotation = 30, ha='right')
sns.countplot(data = df_an, x='Borough')
```

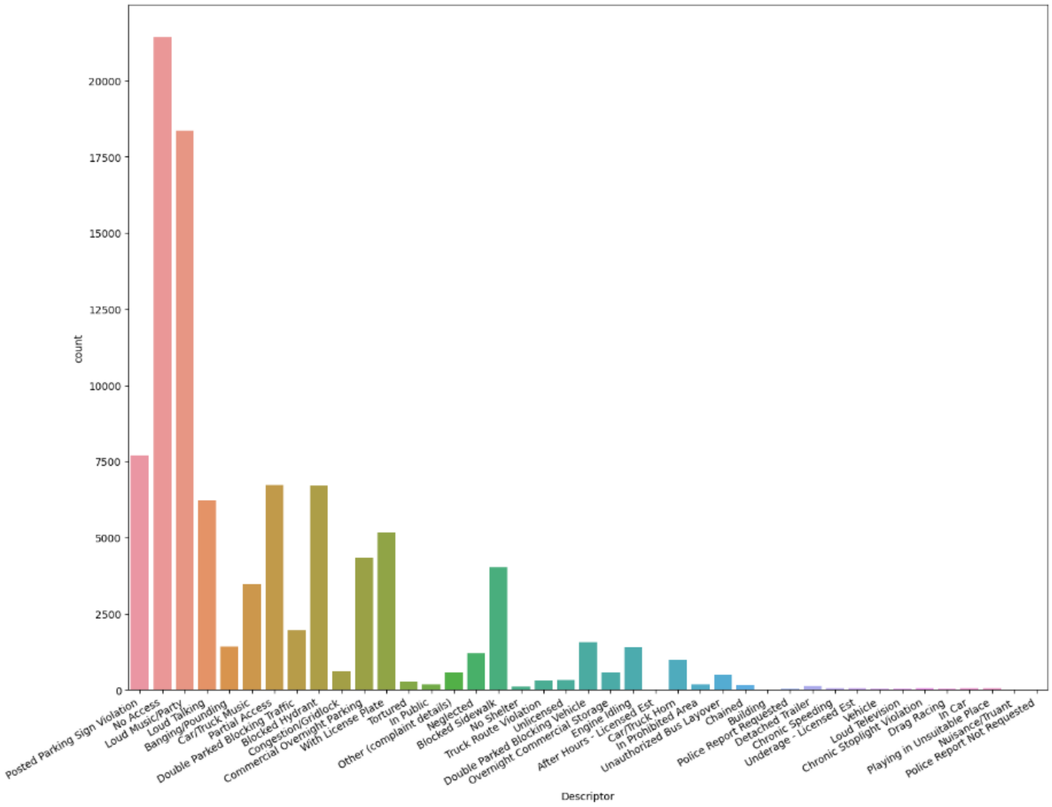
Out[124]:

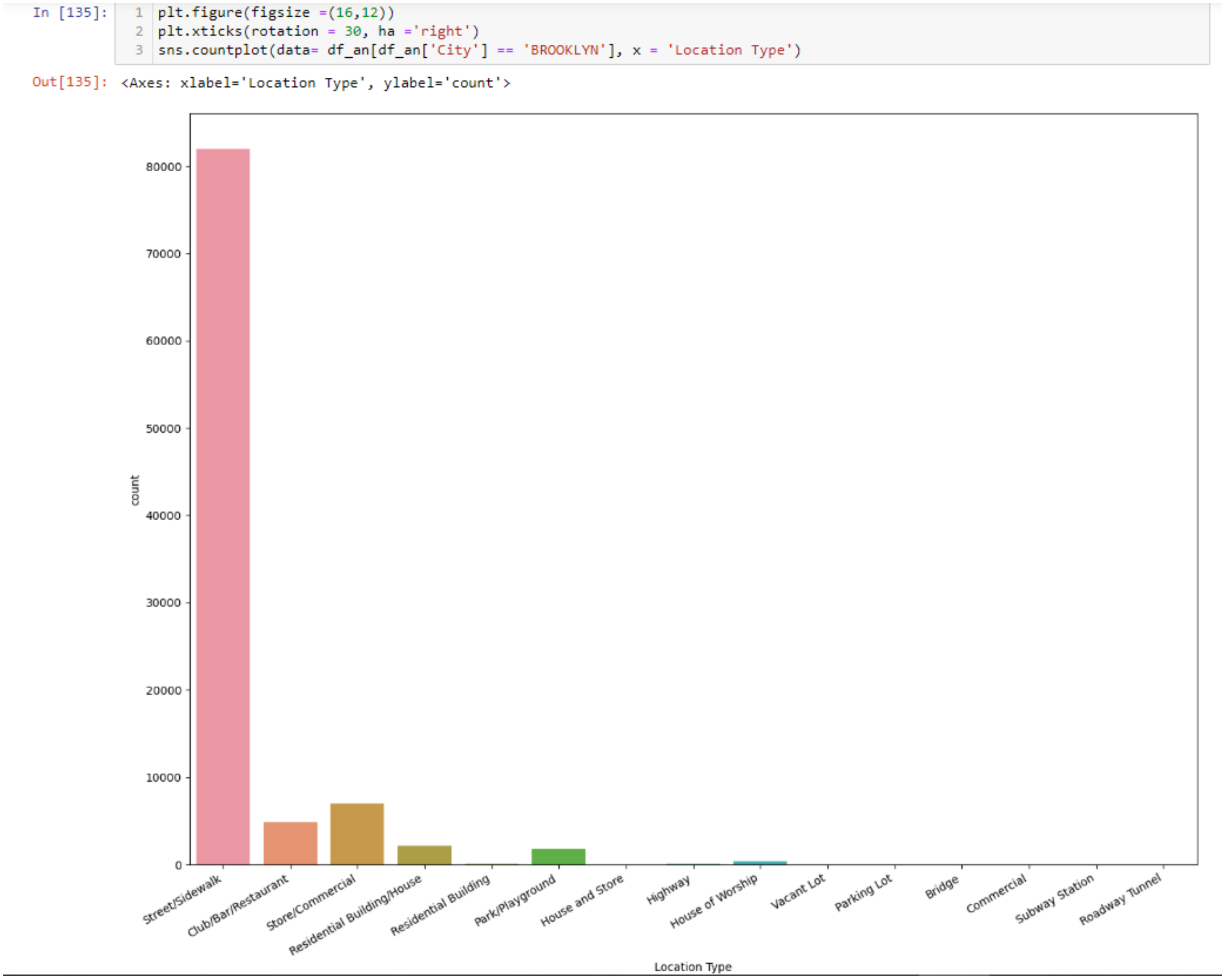
```
<Axes: xlabel='Borough', ylabel='count'>
```

Brooklyn analysis

In [133]:
```
plt.figure(figsize =(16,12))
plt.xticks(rotation = 30, ha ='right')
sns.countplot(data= df_an[df_an['City'] == 'BROOKLYN'], x = 'Descriptor')
```

Out[133]:
```
<Axes: xlabel='Descriptor', ylabel='count'>
```





In [135]:

```
plt.figure(figsize =(16,12))
plt.xticks(rotation = 30, ha ='right')
sns.countplot(data= df_an[df_an['City'] == 'BROOKLYN'], x = 'Location Type')
```

Out[135]:

```
<Axes: xlabel='Location Type', ylabel='count'>
```

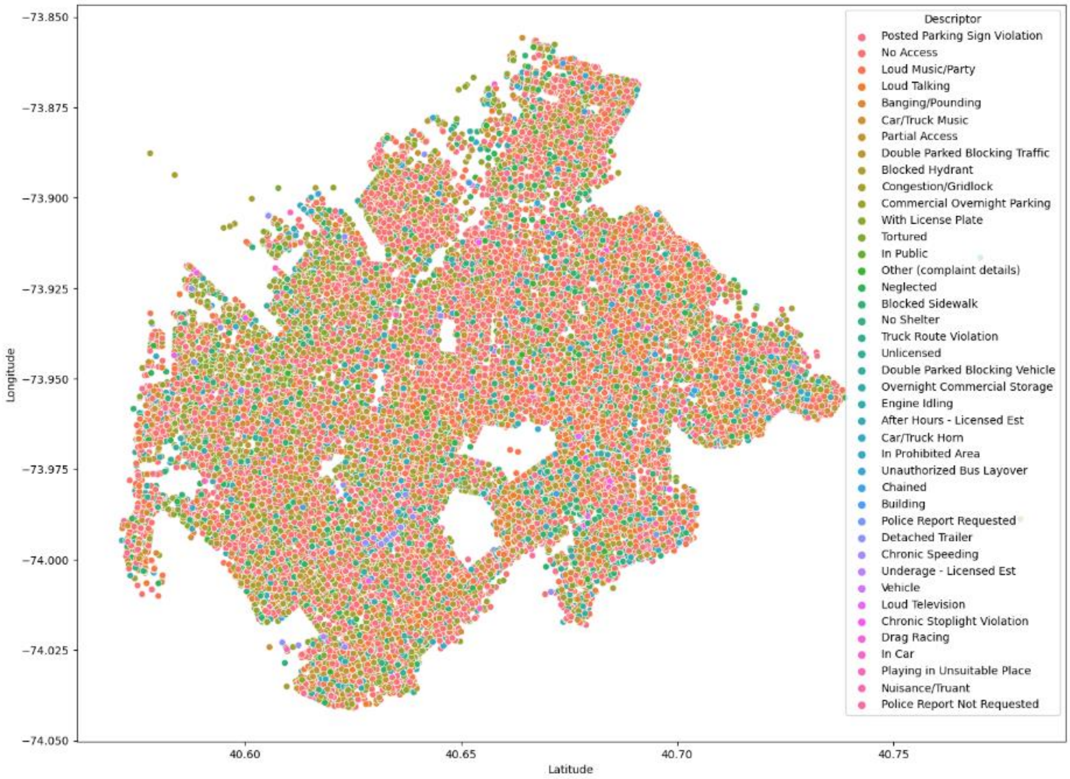
Will plot descriptor and lcation type in scatter plot with using latitude and longitude to understand how this data is distributed around brooklyn

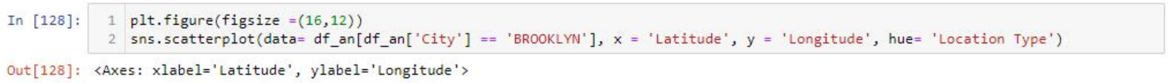
In [126]:

```
plt.figure(figsize =(16,12))
sns.scatterplot(data= df_an[df_an['City'] == 'BROOKLYN'], x = 'Latitude', y = 'Longitude', hue= 'Descriptor')
```

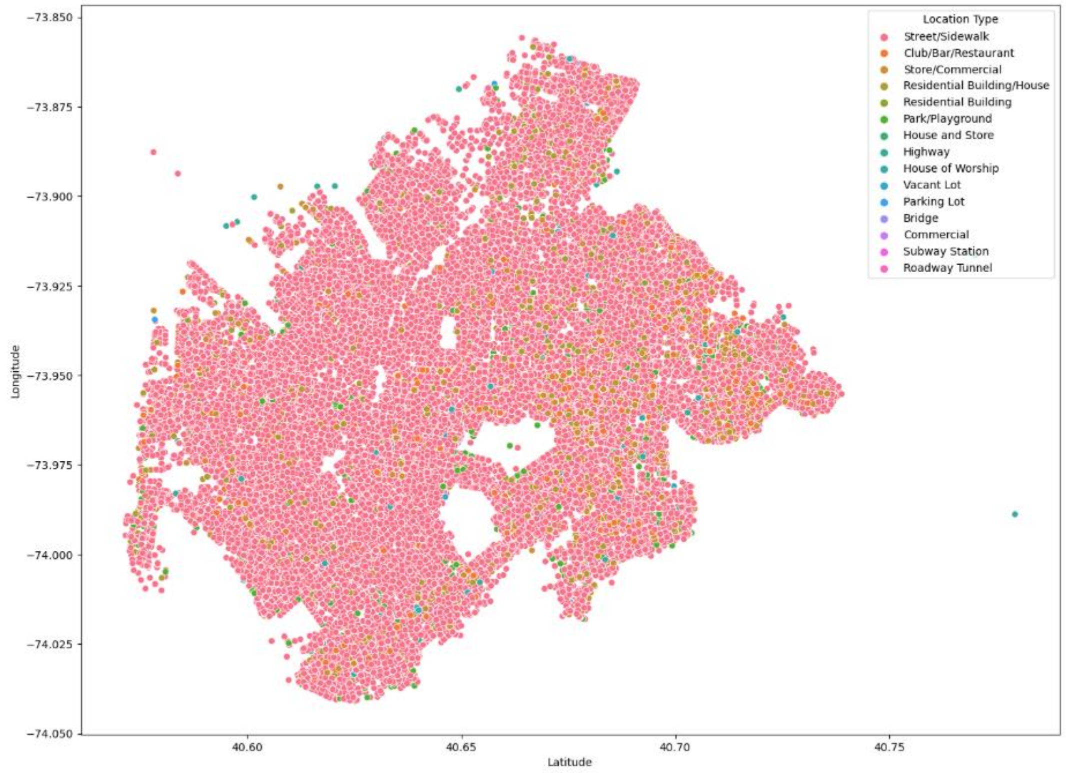
Out[126]:

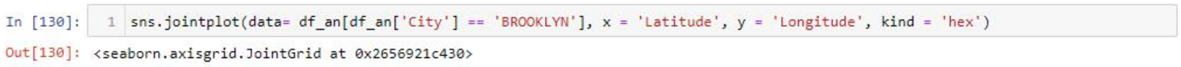
```
<Axes: xlabel='Latitude', ylabel='Longitude'>
```



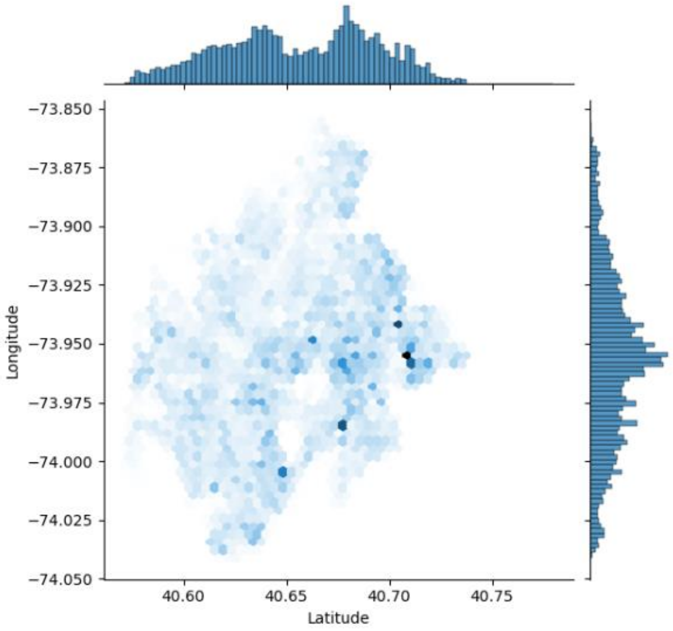


```
In [128]:  plt.figure(figsize =(16,12))
           sns.scatterplot(data= df_an[df_an['City'] == 'BROOKLYN'], x = 'Latitude', y = 'Longitude', hue= 'Location Type')

Out[128]: <Axes: xlabel='Latitude', ylabel='Longitude'>
```





```
In [130]:  sns.jointplot(data= df_an[df_an['City'] == 'BROOKLYN'], x = 'Latitude', y = 'Longitude', kind = 'hex')

Out[130]: <seaborn.axisgrid.JointGrid at 0x2656921c430>
```



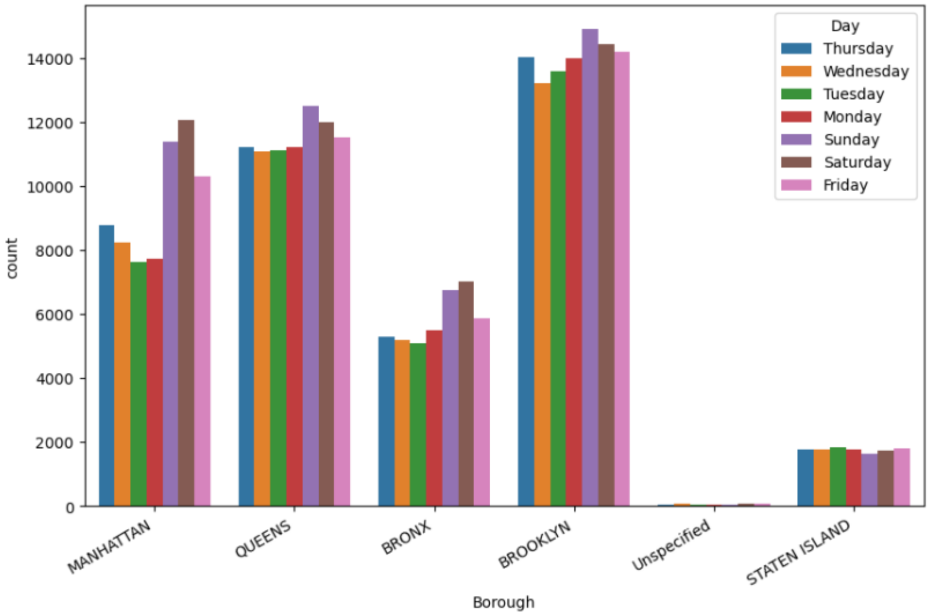
We can compare these plots to Brooklyn's map to see where service request concentration is highest.

In [118]:

```
plt.figure(figsize = (10,6))
plt.xticks(rotation = 30, ha='right')
sns.countplot(data = df_an, x='Borough', hue = 'Day')
```

Out[118]:

```
<Axes: xlabel='Borough', ylabel='count'>
```

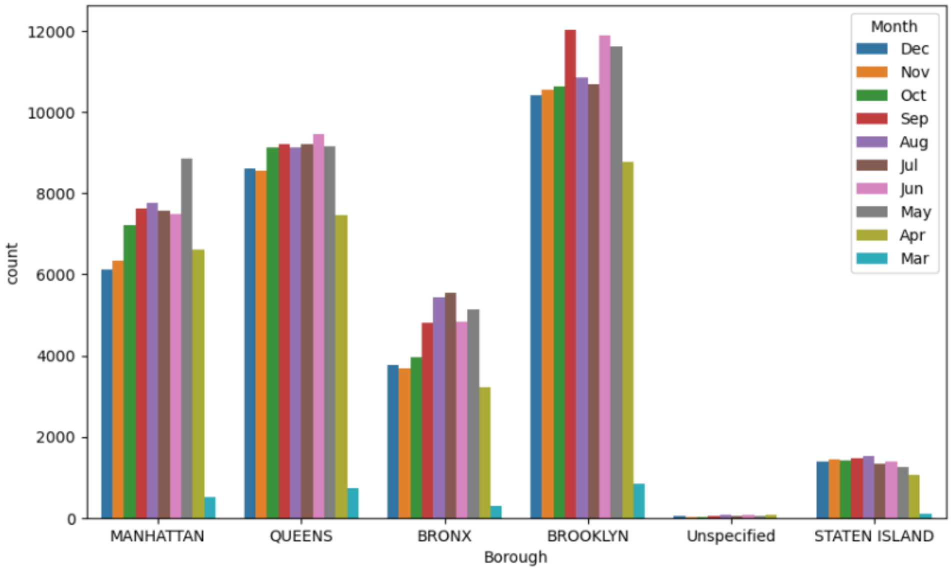


In [123]:

```
plt.figure(figsize = (10,6))
sns.countplot(data = df_an, x='Borough', hue = 'Month')
```

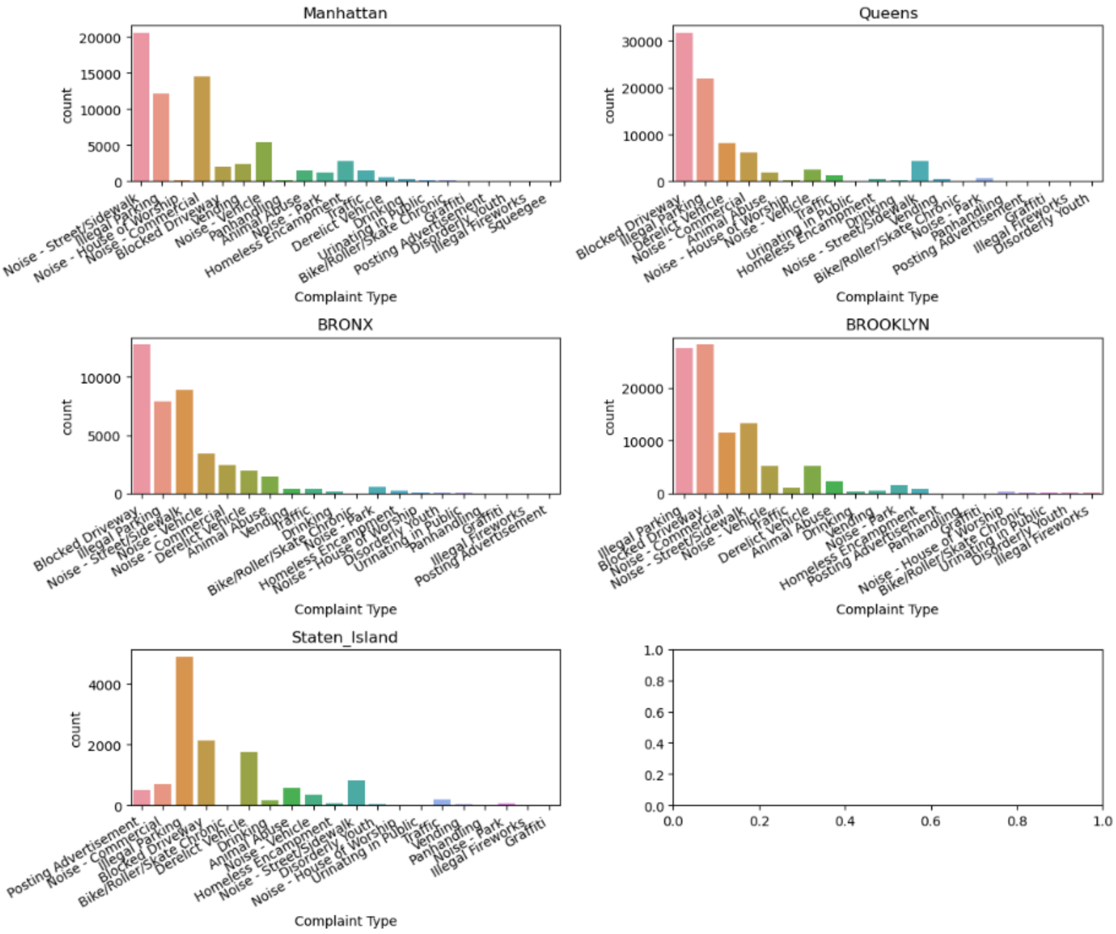
Out[123]:

```
<Axes: xlabel='Borough', ylabel='count'>
```



In [121]:

```python
fig, ax = plt.subplots(3,2,figsize =(12,10))

Manhattan = sns.countplot(ax = ax[0,0],data = df_an[df_an['Borough'] == 'MANHATTAN'], x ='Complaint Type')
Manhattan.set_xticklabels(Manhattan.get_xticklabels(), rotation=30, ha="right")
Manhattan.set_title('Manhattan')

Queens = sns.countplot(ax = ax[0,1],data = df_an[df_an['Borough'] == 'QUEENS'], x ='Complaint Type')
Queens.set_xticklabels(Queens.get_xticklabels(), rotation=30, ha="right")
Queens.set_title('Queens')

BRONX = sns.countplot(ax = ax[1,0],data = df_an[df_an['Borough'] == 'BRONX'], x ='Complaint Type')
BRONX.set_xticklabels(BRONX.get_xticklabels(), rotation=30, ha="right")
BRONX.set_title('BRONX')

BROOKLYN = sns.countplot(ax = ax[1,1],data = df_an[df_an['Borough'] == 'BROOKLYN'], x ='Complaint Type')
BROOKLYN.set_xticklabels(BROOKLYN.get_xticklabels(), rotation=30, ha="right")
BROOKLYN.set_title('BROOKLYN')

Staten_Island = sns.countplot(ax = ax[2,0],data = df_an[df_an['Borough'] == 'STATEN ISLAND'], x ='Complaint Type')
Staten_Island.set_xticklabels(Staten_Island.get_xticklabels(), rotation=30, ha="right")
Staten_Island.set_title('Staten_Island')

plt.tight_layout()
```


Manhattan
count
Noise - Street/Sidewalk
Illegal Parking
Noise - House of Worship
Noise - Commercial
Blocked Driveway
Vending
Noise - Vehicle
Panhandling
Animal Abuse
Noise - Park
Homeless Encampment
Traffic
Derelict Vehicle
Drinking
Urinating in Public
Bike/Roller/Skate Chronic
Graffiti
Posting Advertisement
Disorderly Youth
Illegal Fireworks
Squeegee
Complaint Type
Queens
count
Blocked Driveway
Illegal Parking
Derelict Vehicle
Noise - Commercial
Animal Abuse
Noise - House of Worship
Noise - Vehicle
Traffic
Urinating in Public
Homeless Encampment
Drinking
Noise - Street/Sidewalk
Vending
Bike/Roller/Skate Chronic
Noise - Park
Panhandling
Posting Advertisement
Graffiti
Illegal Fireworks
Disorderly Youth
Complaint Type
BRONX
count
Blocked Driveway
Illegal Parking
Noise - Street/Sidewalk
Noise - Vehicle
Noise - Commercial
Derelict Vehicle
Animal Abuse
Vending
Traffic
Drinking
Bike/Roller/Skate Chronic
Noise - Park
Homeless Encampment
Noise - House of Worship
Disorderly Youth
Urinating in Public
Panhandling
Graffiti
Illegal Fireworks
Posting Advertisement
Complaint Type
BROOKLYN
count
Illegal Parking
Blocked Driveway
Noise - Commercial
Noise - Street/Sidewalk
Noise - Vehicle
Traffic
Derelict Vehicle
Animal Abuse
Drinking
Vending
Noise - Park
Homeless Encampment
Posting Advertisement
Panhandling
Graffiti
Noise - House of Worship
Bike/Roller/Skate Chronic
Urinating in Public
Disorderly Youth
Illegal Fireworks
Complaint Type
Staten_Island
count
Posting Advertisement
Noise - Commercial
Illegal Parking
Blocked Driveway
Bike/Roller/Skate Chronic
Derelict Vehicle
Drinking
Animal Abuse
Noise - Vehicle
Homeless Encampment
Noise - Street/Sidewalk
Disorderly Youth
Noise - House of Worship
Urinating in Public
Traffic
Vending
Panhandling
Noise - Park
Illegal Fireworks
Graffiti
Complaint Type