

Customer Service Request

DESCRIPTION Background of Problem Statement : NYC 311's mission is to provide the public with quick and easy access to all New York City government services and information while offering the best customer service. Each day, NYC311 receives thousands of requests related to several hundred types of non-emergency services, including noise complaints, plumbing issues, and illegally parked cars. These requests are received by NYC311 and forwarded to the relevant agencies such as the police, buildings, or transportation. The agency responds to the request, addresses it, and then closes it.

Problem Objective :

Perform a service request data analysis of New York City 311 calls. You will focus on the data wrangling techniques to understand the pattern in the data and also visualize the major complaint types.

Domain: Customer Service

Analysis Tasks to be performed:

(Perform a service request data analysis of New York City 311 calls)

- 1) Import a 311 NYC service request.
- 2) Read or convert the columns 'Created Date' and 'Closed Date' to datetime datatype and create a new column 'Request_Closing_Time' as the time elapsed between request creation and request closing. (Hint: Explore the package/module datetime)
- 3) Provide major insights/patterns that you can offer in a visual format (graphs or tables); at least 4 major conclusions that you can come up with after generic data mining.
- 4) Order the complaint types based on the average 'Request_Closing_Time', grouping them for different locations.
- 5) Perform a statistical test for the following: Please note: For the below statements you need to state the Null and Alternate and then provide a statistical test to accept or reject the Null Hypothesis along with the corresponding 'p-value'.

a)Whether the average response time across complaint types is similar or not (overall)?

b)Are the type of complaint or service requested and location related?

In [1]:

```
# Import Required Packages
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
import warnings
import scipy.stats as stats
```

1) Import 311 NYC service request.

In [2]:

```
csr_file = r"C:\Users\sinun\Data Science with Python Two\311_Service_Requests_from_2010_to_Present.csv"
csr_df = pd.read_csv(csr_file, low_memory=False) # reading file
csr_df
```

Out[2]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direct
0	32310363	12/31/2015 11:59:45 PM	01-01-16 0:55	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	...	NaN	N
1	32309934	12/31/2015 11:59:44 PM	01-01-16 1:26	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	...	NaN	N
2	32309159	12/31/2015 11:59:29 PM	01-01-16 4:51	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	...	NaN	N
3	32305098	12/31/2015 11:57:46 PM	01-01-16 7:43	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAISLEY AVENUE	...	NaN	N
4	32306529	12/31/2015 11:56:58 PM	01-01-16 3:24	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	...	NaN	N

		Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direct
300693	30281872		03/29/2015 12:33:41 AM		NAN	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	NAN	CRESCENT AVENUE	...	NaN
300694	30281230		03/29/2015 12:33:28 AM	03/29/2015 02:33:59 AM	NAN	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	11418.0	100-17 87 AVENUE	...	NaN
300695	30283424		03/29/2015 12:33:03 AM	03/29/2015 03:40:20 AM	NAN	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	11206.0	162 THROOP AVENUE	...	NaN
300696	30280004		03/29/2015 12:33:02 AM	03/29/2015 04:38:35 AM	NAN	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	10461.0	3151 EAST TREMONT AVENUE	...	NaN
300697	30281825		03/29/2015 12:33:01 AM	03/29/2015 04:41:50 AM	NAN	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Store/Commercial	10036.0	251 WEST 48 STREET	...	NaN

300698 rows × 53 columns



In [3]:

```
#To understand the data characteristics
csr_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300698 entries, 0 to 300697
Data columns (total 53 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unique Key      300698 non-null   int64  
 1   Created Date    300698 non-null   object  
 2   Closed Date     298534 non-null   object  
 3   Agency          300698 non-null   object  
 4   Agency Name     300698 non-null   object  
 5   Complaint Type 300698 non-null   object  
 ...   ...             ...             ...    
```

6	Descriptor	294784	non-null	object
7	Location Type	300567	non-null	object
8	Incident Zip	298083	non-null	float64
9	Incident Address	256288	non-null	object
10	Street Name	256288	non-null	object
11	Cross Street 1	251419	non-null	object
12	Cross Street 2	250919	non-null	object
13	Intersection Street 1	43858	non-null	object
14	Intersection Street 2	43362	non-null	object
15	Address Type	297883	non-null	object
16	City	298084	non-null	object
17	Landmark	349	non-null	object
18	Facility Type	298527	non-null	object
19	Status	300698	non-null	object
20	Due Date	300695	non-null	object
21	Resolution Description	300698	non-null	object
22	Resolution Action Updated Date	298511	non-null	object
23	Community Board	300698	non-null	object
24	Borough	300698	non-null	object
25	X Coordinate (State Plane)	297158	non-null	float64
26	Y Coordinate (State Plane)	297158	non-null	float64
27	Park Facility Name	300698	non-null	object
28	Park Borough	300698	non-null	object
29	School Name	300698	non-null	object
30	School Number	300698	non-null	object
31	School Region	300697	non-null	object
32	School Code	300697	non-null	object
33	School Phone Number	300698	non-null	object
34	School Address	300698	non-null	object
35	School City	300698	non-null	object
36	School State	300698	non-null	object
37	School Zip	300697	non-null	object
38	School Not Found	300698	non-null	object
39	School or Citywide Complaint	0	non-null	float64
40	Vehicle Type	0	non-null	float64
41	Taxi Company Borough	0	non-null	float64
42	Taxi Pick Up Location	0	non-null	float64
43	Bridge Highway Name	243	non-null	object
44	Bridge Highway Direction	243	non-null	object
45	Road Ramp	213	non-null	object
46	Bridge Highway Segment	213	non-null	object
47	Garage Lot Name	0	non-null	float64
48	Ferry Direction	1	non-null	object
49	Ferry Terminal Name	2	non-null	object
50	Latitude	297158	non-null	float64
51	Longitude	297158	non-null	float64
52	Location	297158	non-null	object

dtypes: float64(10), int64(1), object(42)
memory usage: 121.6+ MB

2) Read or convert the columns 'Created Date' and Closed Date' to datetime datatype

In [4]:

```
# Covert date coloumns to DateTime data format at import itself
date_fields = ['Created Date', 'Closed Date', 'Due Date']
csr_file = r"C:\Users\sinun\Data Science with Python Two\311_Service_Requests_from_2010_to_Present.csv"
csr_df = pd.read_csv(csr_file, parse_dates = date_fields, infer_datetime_format = True, low_memory=False) #datetime format
csr_df
```

Out[4]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direction	I
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	VERMILYEA AVENUE	71 ...	NaN	NaN	
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	27-07 23 AVENUE ...	NaN	NaN	
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	VALENTINE AVENUE	2897 ...	NaN	NaN	
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	BAISLEY AVENUE	2940 ...	NaN	NaN	
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	87-14 57 ROAD ...	NaN	NaN	
...	
300693	30281872	2015-03-29 00:33:41	NaT	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	NaN	CRESCENT AVENUE	CRESCENT AVENUE ...	NaN	NaN	

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direction	I
300694	30281230	2015-03-29 00:33:28	2015-03-29 02:33:59	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	11418.0	100-17 87 AVENUE	...	NaN	NaN	
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	11206.0	162 THROOP AVENUE	...	NaN	NaN	
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	10461.0	3151 EAST TREMONT AVENUE	...	NaN	NaN	
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Store/Commercial	10036.0	251 WEST 48 STREET	...	NaN	NaN	

300698 rows × 53 columns



In [5]:
To understand the Rows and Coloumns
 csr_df.shape

Out[5]: (300698, 53)

In [6]:
 csr_df.head()

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direction	Road Ramp	Hic Seq
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	...	NaN	NaN	NaN	

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Bridge Highway Name	Bridge Highway Direction	Road Ramp	Ramp	Hig Seq
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 AVENUE	...	NaN	NaN	NaN		
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	...	NaN	NaN	NaN		
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAISLEY AVENUE	...	NaN	NaN	NaN		
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	...	NaN	NaN	NaN		

5 rows x 53 columns



Data Pre- Processing

In [7]:

```
#To understand how much null non-values are present in each coloum in percentage
csr_df.isna().sum()/csr_df.shape[0] *100
```

Out[7]: Unique Key	0.000000
Created Date	0.000000
Closed Date	0.719659
Agency	0.000000
Agency Name	0.000000
Complaint Type	0.000000
Descriptor	1.966757
Location Type	0.043565
Incident Zip	0.869643
Incident Address	14.768971
Street Name	14.768971
Cross Street 1	16.388203
Cross Street 2	16.554483
Intersection Street 1	85.414602

```
Intersection Street 2           85.579552
Address Type                   0.936155
City                           0.869311
Landmark                        99.883937
Facility Type                  0.721987
Status                          0.000000
Due Date                        0.000998
Resolution Description          0.000000
Resolution Action Updated Date  0.727308
Community Board                 0.000000
Borough                         0.000000
X Coordinate (State Plane)    1.177261
Y Coordinate (State Plane)    1.177261
Park Facility Name             0.000000
Park Borough                    0.000000
School Name                     0.000000
School Number                   0.000000
School Region                   0.000333
School Code                     0.000333
School Phone Number            0.000000
School Address                  0.000000
School City                     0.000000
School State                    0.000000
School Zip                      0.000333
School Not Found               0.000000
School or Citywide Complaint   100.000000
Vehicle Type                    100.000000
Taxi Company Borough           100.000000
Taxi Pick Up Location          100.000000
Bridge Highway Name             99.919188
Bridge Highway Direction        99.919188
Road Ramp                       99.929165
Bridge Highway Segment          99.929165
Garage Lot Name                 100.000000
Ferry Direction                 99.999667
Ferry Terminal Name             99.999335
Latitude                        1.177261
Longitude                       1.177261
Location                         1.177261
dtype: float64
```

In [8]:

```
# To avoid the coloums having majority of null values
rel_cols = pd.DataFrame(csr_df.columns.to_list()).set_index(0)
rel_cols = rel_cols[csr_df.isna().sum()/csr_df.shape[0] *100 < 80].reset_index()
rel_cols
```

Out[8]:

0

0	Unique Key
1	Created Date
2	Closed Date
3	Agency
4	Agency Name
5	Complaint Type
6	Descriptor
7	Location Type
8	Incident Zip
9	Incident Address
10	Street Name
11	Cross Street 1
12	Cross Street 2
13	Address Type
14	City
15	Facility Type
16	Status
17	Due Date
18	Resolution Description
19	Resolution Action Updated Date
20	Community Board
21	Borough
22	X Coordinate (State Plane)
23	Y Coordinate (State Plane)

0

24	Park Facility Name
25	Park Borough
26	School Name
27	School Number
28	School Region
29	School Code
30	School Phone Number
31	School Address
32	School City
33	School State
34	School Zip
35	School Not Found
36	Latitude
37	Longitude
38	Location

In [9]:

```
csr_df = csr_df[rel_cols[0].to_list()]
csr_df
```

Out[9]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	School Code	School Phon Number
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	VERMILYEA AVENUE	71	Unspecified	Unspecified
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	27-07 23	Unspecified	Unspecified

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	School Code	School Phon Number
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	VALENTINE AVENUE	2897	Unspecified	Unspecified
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	BAISLEY AVENUE	2940	Unspecified	Unspecified
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	87-14 57 ROAD	Unspecified	Unspecified
...
300693	30281872	2015-03-29 00:33:41	NaT	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	NaN	CRESCENT AVENUE	...	Unspecified	Unspecified
300694	30281230	2015-03-29 00:33:28	2015-03-29 02:33:59	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	11418.0	100-17 87 AVENUE	100-17 87 AVENUE	Unspecified	Unspecified
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	11206.0	162 THROOP AVENUE	162 THROOP AVENUE	Unspecified	Unspecified
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	10461.0	3151 EAST TREMONT AVENUE	3151 EAST TREMONT AVENUE	Unspecified	Unspecified
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Store/Commercial	10036.0	251 WEST 48 STREET	251 WEST 48 STREET	Unspecified	Unspecified

300698 rows × 39 columns



In [10]:

```
#To understand how much coloumns are remaining after deletion of null valued coloumns
csr_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300698 entries, 0 to 300697
Data columns (total 39 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unique Key       300698 non-null   int64  
 1   Created Date     300698 non-null   datetime64[ns]
 2   Closed Date      298534 non-null   datetime64[ns]
 3   Agency           300698 non-null   object  
 4   Agency Name      300698 non-null   object  
 5   Complaint Type   300698 non-null   object  
 6   Descriptor        294784 non-null   object  
 7   Location Type    300567 non-null   object  
 8   Incident Zip     298083 non-null   float64 
 9   Incident Address 256288 non-null   object  
 10  Street Name      256288 non-null   object  
 11  Cross Street 1   251419 non-null   object  
 12  Cross Street 2   250919 non-null   object  
 13  Address Type     297883 non-null   object  
 14  City              298084 non-null   object  
 15  Facility Type    298527 non-null   object  
 16  Status            300698 non-null   object  
 17  Due Date          300695 non-null   datetime64[ns]
 18  Resolution Description 300698 non-null   object  
 19  Resolution Action Updated Date 298511 non-null   object  
 20  Community Board   300698 non-null   object  
 21  Borough           300698 non-null   object  
 22  X Coordinate (State Plane) 297158 non-null   float64 
 23  Y Coordinate (State Plane) 297158 non-null   float64 
 24  Park Facility Name 300698 non-null   object  
 25  Park Borough      300698 non-null   object  
 26  School Name       300698 non-null   object  
 27  School Number     300698 non-null   object  
 28  School Region     300697 non-null   object  
 29  School Code       300697 non-null   object  
 30  School Phone Number 300698 non-null   object  
 31  School Address    300698 non-null   object  
 32  School City       300698 non-null   object  
 33  School State      300698 non-null   object  
 34  School Zip        300697 non-null   object  
 35  School Not Found  300698 non-null   object  
 36  Latitude           297158 non-null   float64 
 37  Longitude          297158 non-null   float64 
 38  Location           297158 non-null   object  
dtypes: datetime64[ns](3), float64(5), int64(1), object(30)
memory usage: 89.5+ MB
```

In [11]: `csr_df['School Name'].value_counts() # The School related coloumns contains most of entries as 'unspecified'`

Out[11]: Unspecified 300697
Alley Pond Park - Nature Center 1
Name: School Name, dtype: int64

In [12]: `csr_df[csr_df['School Name'] != 'Unspecified']`

Out[12]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	School Code	School Phone Number	School Address	School City
283132	30427220	2015-04-18 09:44:55	2015-05-02 10:35:00	NYPD	New York City Police Department	Animal in a Park	Animal Waste	Park	NaN	NaN	...	NaN	7182176034	Grand Central Parkway, near the soccer field	QUEENS

1 rows × 16 columns

In [13]: `csr_df['School Code'].value_counts()`

Out[13]: Unspecified 300697
Name: School Code, dtype: int64

In [14]: `# Remove all school related coloumns from dataset as they contain 'Unspecified'`
`removed_sch_list = [col for col in csr_df.columns.to_list() if col.find('School') < 0]`
`removed_sch_list`

Out[14]: ['Unique Key',
'Created Date',
'Closed Date',
'Agency',
'Agency Name',
'Complaint Type',
'Descriptor',
'Location Type',
'Incident Zip',

```
'Incident Address',
'Street Name',
'Cross Street 1',
'Cross Street 2',
'Address Type',
'City',
'Facility Type',
>Status',
'Due Date',
'Resolution Description',
'Resolution Action Updated Date',
'Community Board',
'Borough',
'X Coordinate (State Plane)',
'Y Coordinate (State Plane)',
'Park Facility Name',
'Park Borough',
'Latitude',
'Longitude',
'Location']
```

In [15]:

```
# The dataset that excludes 'School' coloums
csr_df = csr_df[removed_sch_list]
csr_df
```

Out[15]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Resolution Action Updated Date	Commur Bo
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEAVENUE	...	01-01-16 0:55	MANHATT
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07-23 AVENUE	...	01-01-16 1:26	01 QUEE
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	...	01-01-16 4:51	07 BRO
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAISLEY AVENUE	...	01-01-16 7:43	10 BRO

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Resolution		Commur Bo:
												Action Updated Date	...	
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	...	01-01-16 3:24	04 QUEE	
300693	30281872	2015-03-29 00:33:41	NaT	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	NaN	CRESCENT AVENUE	...	NaN	0 Unspecif	
300694	30281230	2015-03-29 00:33:28	2015-03-29 02:33:59	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	11418.0	100-17 87 AVENUE	...	03/29/2015 02:33:59 AM	09 QUEE	
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	11206.0	162 THROOP AVENUE	...	03/29/2015 03:40:20 AM	BROOKL	
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	10461.0	3151 EAST TREMONT AVENUE	...	03/29/2015 04:38:35 AM	10 BRO	
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Store/Commercial	10036.0	251 WEST 48 STREET	...	03/29/2015 04:41:50 AM	MANHATT	

300698 rows × 29 columns



Drop Park Facility Name Location : since it is a combination of Latitute and Longitude

In [16]:

csr_df.info() # Remaining 29 coloums

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 300698 entries, 0 to 300697
Data columns (total 29 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unique Key       300698 non-null   int64
```

```

1 Created Date           300698 non-null datetime64[ns]
2 Closed Date           298534 non-null datetime64[ns]
3 Agency                300698 non-null object
4 Agency Name            300698 non-null object
5 Complaint Type         300698 non-null object
6 Descriptor              294784 non-null object
7 Location Type           300567 non-null object
8 Incident Zip             298083 non-null float64
9 Incident Address          256288 non-null object
10 Street Name             256288 non-null object
11 Cross Street 1           251419 non-null object
12 Cross Street 2           250919 non-null object
13 Address Type             297883 non-null object
14 City                   298084 non-null object
15 Facility Type            298527 non-null object
16 Status                  300698 non-null object
17 Due Date                 300695 non-null datetime64[ns]
18 Resolution Description      300698 non-null object
19 Resolution Action Updated Date 298511 non-null object
20 Community Board           300698 non-null object
21 Borough                  300698 non-null object
22 X Coordinate (State Plane) 297158 non-null float64
23 Y Coordinate (State Plane) 297158 non-null float64
24 Park Facility Name        300698 non-null object
25 Park Borough               300698 non-null object
26 Latitude                  297158 non-null float64
27 Longitude                  297158 non-null float64
28 Location                   297158 non-null object
dtypes: datetime64[ns](3), float64(5), int64(1), object(20)
memory usage: 66.5+ MB

```

In [17]:

```
# Checking for similar entries in 'Community Board' and 'Borough'
pd.crosstab(csr_df['Community Board'], csr_df['Borough'])
```

Out[17]:

	Borough	BRONX	BROOKLYN	MANHATTAN	QUEENS	STATEN ISLAND	Unspecified
--	---------	-------	----------	-----------	--------	---------------	-------------

Community Board	Borough	BRONX	BROOKLYN	MANHATTAN	QUEENS	STATEN ISLAND	Unspecified
0 Unspecified	0	0	0	0	0	2574	
01 BRONX	2146	0	0	0	0	0	
01 BROOKLYN	0	10920		0	0	0	
01 MANHATTAN	0	0	3615	0	0	0	

Community Board	Borough	BRONX	BROOKLYN	MANHATTAN	QUEENS	STATEN ISLAND	Unspecified
01 QUEENS		0	0	0	9197	0	0
...
84 QUEENS		0	0	0	11	0	0
95 STATEN ISLAND		0	0	0	0	32	0
Unspecified MANHATTAN		0	0	23	0	0	0
Unspecified QUEENS		0	0	0	2	0	0
Unspecified STATEN ISLAND		0	0	0	0	2	0

75 rows × 6 columns

In [18]: *#To Check the distinct values in Borough*
`csr_df['Borough'].value_counts()`

Out[18]:

BROOKLYN	98307
QUEENS	80641
MANHATTAN	66131
BRONX	40702
STATEN ISLAND	12343
Unspecified	2574
Name: Borough, dtype: int64	

In [19]: *#To Check the distinct values in Community Board*
`csr_df['Community Board'].value_counts()`

Out[19]:

12 MANHATTAN	12390
01 BROOKLYN	10920
05 QUEENS	9422
01 QUEENS	9197
09 QUEENS	8013
...	
84 QUEENS	11
56 BROOKLYN	9
80 QUEENS	7
Unspecified QUEENS	2

```
Unspecified STATEN ISLAND      2  
Name: Community Board, Length: 75, dtype: int64
```

```
In [20]: #The values in Community Board contains some numbers attached with the names  
# To remove the numbers prefixed with the names in entries of Community Board  
  
def remove_num(input_str):  
    out_str = "".join([c for c in input_str if c not in list("0123456789")])  
    return out_str  
  
warnings.filterwarnings ("ignore")  
csr_df['CB2'] = csr_df['Community Board'].apply(lambda x : x.replace("Unspecified", ""))  
csr_df['CB2'] = csr_df['CB2'].apply(remove_num)  
csr_df['CB2']
```

```
Out[20]: 0      MANHATTAN  
1      QUEENS  
2      BRONX  
3      BRONX  
4      QUEENS  
...  
300693  
300694      QUEENS  
300695      BROOKLYN  
300696      BRONX  
300697      MANHATTAN  
Name: CB2, Length: 300698, dtype: object
```

```
In [21]: # To find out the distinct values in Community board  
csr_df['CB2'].value_counts()
```

```
Out[21]: BROOKLYN      98307  
QUEENS        80641  
MANHATTAN     66131  
BRONX         40702  
STATEN ISLAND  12343  
              2574  
Name: CB2, dtype: int64
```

```
In [22]: # The coloums "Borough" and "Community Baoard " have similar entries, So DROP column 'Community Board'  
csr_df.drop(columns = ['Community Board', 'CB2'], inplace = True)  
csr_df
```

Out[22]:

		Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Resolution Description	Resolutic Actic Update Da
0	32310363		2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	...	The Police Department responded and upon arriv...	01-01-01
1	32309934		2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	...	The Police Department responded to the complai...	01-01-12
2	32309159		2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	...	The Police Department responded and upon arriv...	01-01-41
3	32305098		2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAISLEY AVENUE	...	The Police Department responded to the complai...	01-01-72
4	32306529		2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	87-14 57 ROAD	...	The Police Department responded and upon arriv...	01-01-32
...
300693	30281872		2015-03-29 00:33:41	NaT	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	NaN	CRESCENT AVENUE	...	Your complaint has been forwarded to the New Y...	Na

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	...	Resolution Description	Resolutic Actic Update Da
300694	30281230	2015-03-29 00:33:28	2015-03-29 02:33:59	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	11418.0	100-17 87 AVENUE	...	The Police Department responded and upon arriv...	03/29/2023: A
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	11206.0	162 THROOP AVENUE	...	The Police Department responded to the complai...	03/29/2023: A
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Club/Bar/Restaurant	10461.0	3151 EAST TREMONT AVENUE	...	The Police Department responded to the complai...	03/29/2023: A
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial	Loud Music/Party	Store/Commercial	10036.0	251 WEST 48 STREET	...	The Police Department responded to the complai...	03/29/2023: A

300698 rows x 28 columns



In [23]:

```
# To replace all other 'unspecified' entries as na values
csr_df.replace('Unspecified',np.NaN,inplace=True)
csr_df=csr_df.dropna(how='all')
#Remove all the rows containing NA values
csr_df= csr_df[(csr_df['Latitude'].notnull())& csr_df['Longitude'].notnull() & (csr_df['Descriptor'].notnull())]
csr_df.unique()
```

Out[23]: Unique Key
Created Date
Closed Date291317
252153
232128

```

Agency                               1
Agency Name                         2
Complaint Type                     15
Descriptor                          41
Location Type                      14
Incident Zip                        200
Incident Address                   104602
Street Name                         6624
Cross Street 1                      5786
Cross Street 2                      5801
Address Type                        5
City                                53
Facility Type                       1
Status                              4
Due Date                            252500
Resolution Description               14
Resolution Action Updated Date     232868
Borough                             5
X Coordinate (State Plane)        62809
Y Coordinate (State Plane)        73038
Park Facility Name                 0
Park Borough                        5
Latitude                            123070
Longitude                           123169
Location                            123980
dtype: int64

```

Create a new column 'Request_Closing_Time' as the time elapsed between request creation and request closing.

```
In [24]: csr_df['Request_Closing_Time'] = csr_df['Closed Date'] - csr_df['Created Date']# calculate the time taken for Action
```

```
In [25]: csr_df['Request_Closing_Time_Bin'] = pd.cut(csr_df['Request_Closing_Time'], 100)

csr_df['Request_Closing_Time_Bin'].value_counts()
```

```

Out[25]: (-1 days +23:25:25.718000, 0 days 05:56:42.820000]    228777
(0 days 05:56:42.820000, 0 days 11:52:25.640000]             45164
(0 days 11:52:25.640000, 0 days 17:48:08.460000]             10223
(0 days 17:48:08.460000, 0 days 23:43:51.280000]             3410
(0 days 23:43:51.280000, 1 days 05:39:34.100000]             1465
...
(15 days 01:39:32.020000, 15 days 07:35:14.840000]           0
(15 days 07:35:14.840000, 15 days 13:30:57.660000]           0
(15 days 13:30:57.660000, 15 days 19:26:40.480000]           0

```

```
(15 days 19:26:40.480000, 16 days 01:22:23.300000]          0
(12 days 08:26:41, 12 days 14:22:23.820000]            0
Name: Request_Closing_Time_Bin, Length: 100, dtype: int64
```

In [26]:

```
csr_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 291317 entries, 0 to 300697
Data columns (total 30 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unique Key       291317 non-null   int64  
 1   Created Date     291317 non-null   datetime64[ns]
 2   Closed Date      291261 non-null   datetime64[ns]
 3   Agency           291317 non-null   object  
 4   Agency Name      291317 non-null   object  
 5   Complaint Type   291317 non-null   object  
 6   Descriptor        291317 non-null   object  
 7   Location Type    291205 non-null   object  
 8   Incident Zip     291277 non-null   float64 
 9   Incident Address 249006 non-null   object  
 10  Street Name      249006 non-null   object  
 11  Cross Street 1   246163 non-null   object  
 12  Cross Street 2   246160 non-null   object  
 13  Address Type     291094 non-null   object  
 14  City              291277 non-null   object  
 15  Facility Type    291264 non-null   object  
 16  Status            291317 non-null   object  
 17  Due Date          291316 non-null   datetime64[ns]
 18  Resolution Description 291317 non-null   object  
 19  Resolution Action Updated Date 291239 non-null   object  
 20  Borough           291317 non-null   object  
 21  X Coordinate (State Plane) 291317 non-null   float64 
 22  Y Coordinate (State Plane) 291317 non-null   float64 
 23  Park Facility Name 0 non-null     object  
 24  Park Borough      291317 non-null   object  
 25  Latitude           291317 non-null   float64 
 26  Longitude          291317 non-null   float64 
 27  Location           291317 non-null   object  
 28  Request_Closing_Time 291261 non-null   timedelta64[ns]
 29  Request_Closing_Time_Bin 291261 non-null   category
dtypes: category(1), datetime64[ns](3), float64(5), int64(1), object(19), timedelta64[ns](1)
memory usage: 67.0+ MB
```

In [27]:

```
#Convert Request Closing time in Seconds
```

```
csr_df['Request_Closing_Time_Secs'] = csr_df['Request_Closing_Time'].apply(lambda x : x.seconds)
```

```
csr_df['Request_Closing_Time_Secs']=csr_df['Request_Closing_Time_Secs']
csr_df=csr_df[csr_df['Request_Closing_Time_Secs'].notnull()]
csr_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 291261 entries, 0 to 300697
Data columns (total 31 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unique Key      291261 non-null   int64  
 1   Created Date    291261 non-null   datetime64[ns]
 2   Closed Date     291261 non-null   datetime64[ns]
 3   Agency          291261 non-null   object  
 4   Agency Name     291261 non-null   object  
 5   Complaint Type 291261 non-null   object  
 6   Descriptor      291261 non-null   object  
 7   Location Type   291149 non-null   object  
 8   Incident Zip    291221 non-null   float64 
 9   Incident Address 248961 non-null   object  
 10  Street Name     248961 non-null   object  
 11  Cross Street 1  246125 non-null   object  
 12  Cross Street 2  246122 non-null   object  
 13  Address Type   291038 non-null   object  
 14  City            291221 non-null   object  
 15  Facility Type  291256 non-null   object  
 16  Status          291261 non-null   object  
 17  Due Date        291260 non-null   datetime64[ns]
 18  Resolution Description 291261 non-null   object  
 19  Resolution Action Updated Date 291224 non-null   object  
 20  Borough         291261 non-null   object  
 21  X Coordinate (State Plane) 291261 non-null   float64 
 22  Y Coordinate (State Plane) 291261 non-null   float64 
 23  Park Facility Name 0 non-null    object  
 24  Park Borough    291261 non-null   object  
 25  Latitude         291261 non-null   float64 
 26  Longitude        291261 non-null   float64 
 27  Location         291261 non-null   object  
 28  Request_Closing_Time 291261 non-null   timedelta64[ns]
 29  Request_Closing_Time_Bin 291261 non-null   category 
 30  Request_Closing_Time_Secs 291261 non-null   float64 
dtypes: category(1), datetime64[ns](3), float64(6), int64(1), object(19), timedelta64[ns](1)
memory usage: 69.2+ MB
```

3) Conclusions After Data Visualization

In [28]:

```
# To find out the most frequent types of Complaints
most_freq_complaints=csr_df.dropna(subset=["Complaint Type"])
most_freq_complaints=csr_df.groupby("Complaint Type")

sorted_ComplaintType = most_freq_complaints.size().sort_values(ascending = False) # to find how many times each complaint occurred
sorted_ComplaintType = sorted_ComplaintType.to_frame('count').reset_index()

sorted_ComplaintType
sorted_ComplaintType.head(10) # To show the TABLE of top 10 complaints
```

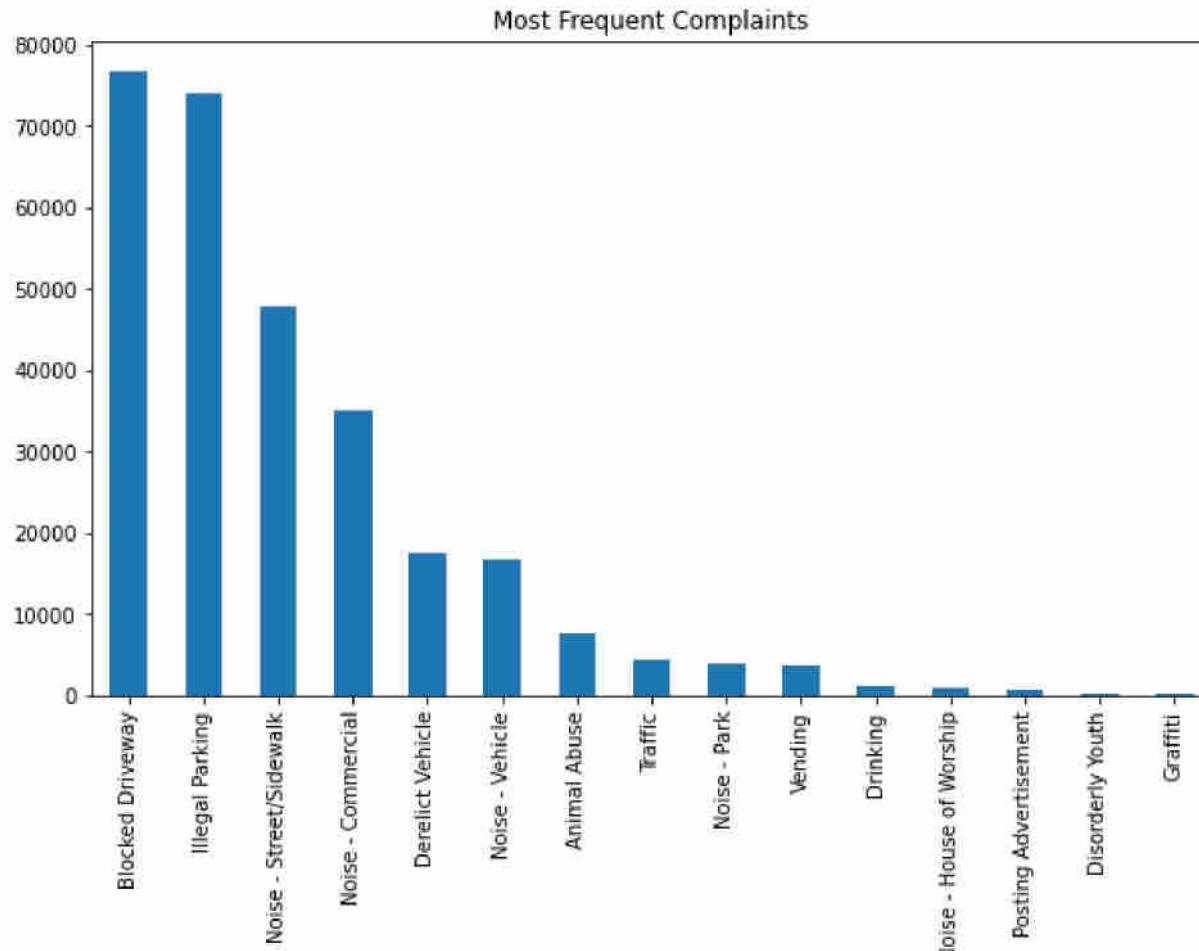
Out[28]:

	Complaint Type	count
0	Blocked Driveway	76714
1	Illegal Parking	74059
2	Noise - Street/Sidewalk	47779
3	Noise - Commercial	35161
4	Derelict Vehicle	17514
5	Noise - Vehicle	16871
6	Animal Abuse	7747
7	Traffic	4474
8	Noise - Park	3928
9	Vending	3775

In [29]:

```
# To plot the complaint types against the number of times they have occurred
(csr_df['Complaint Type'].value_counts()).head(15).plot(kind='bar', figsize=(10,6), title="Most Frequent Complaints")
```

Out[29]: <AxesSubplot:title={'center':'Most Frequent Complaints'}>



Conclusion 1:

- The most frequent complaint type received is about Blocked Driveway , followed by Illegal Parking.

In [30]:

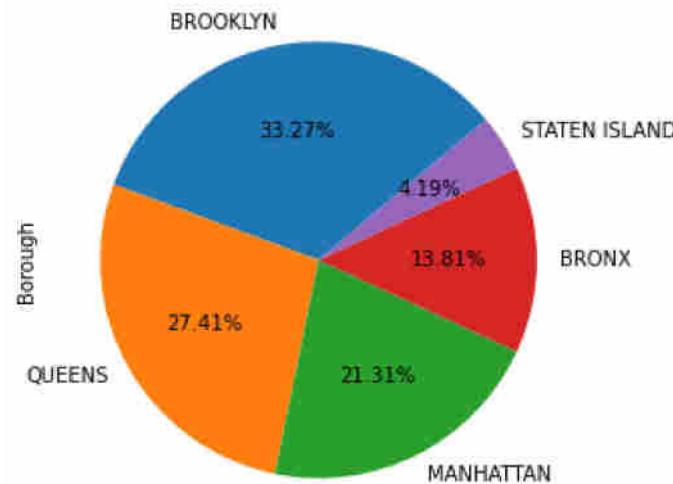
```
#To understand the Borough wise portion of total complaints received  
csr_df['Borough'].value_counts()
```

Out[30]: BROOKLYN 96899
QUEENS 79833
MANHATTAN 62081
BRONX 40231

```
STATEN ISLAND      12217
Name: Borough, dtype: int64
```

```
In [31]: # To plot a pie diagram showing the Borough wise percentage of complaints received
csr_df['Borough'].value_counts().plot(kind='pie', autopct='%1.2f%%', startangle=40, figsize=(10,5))
```

```
Out[31]: <AxesSubplot:ylabel='Borough'>
```

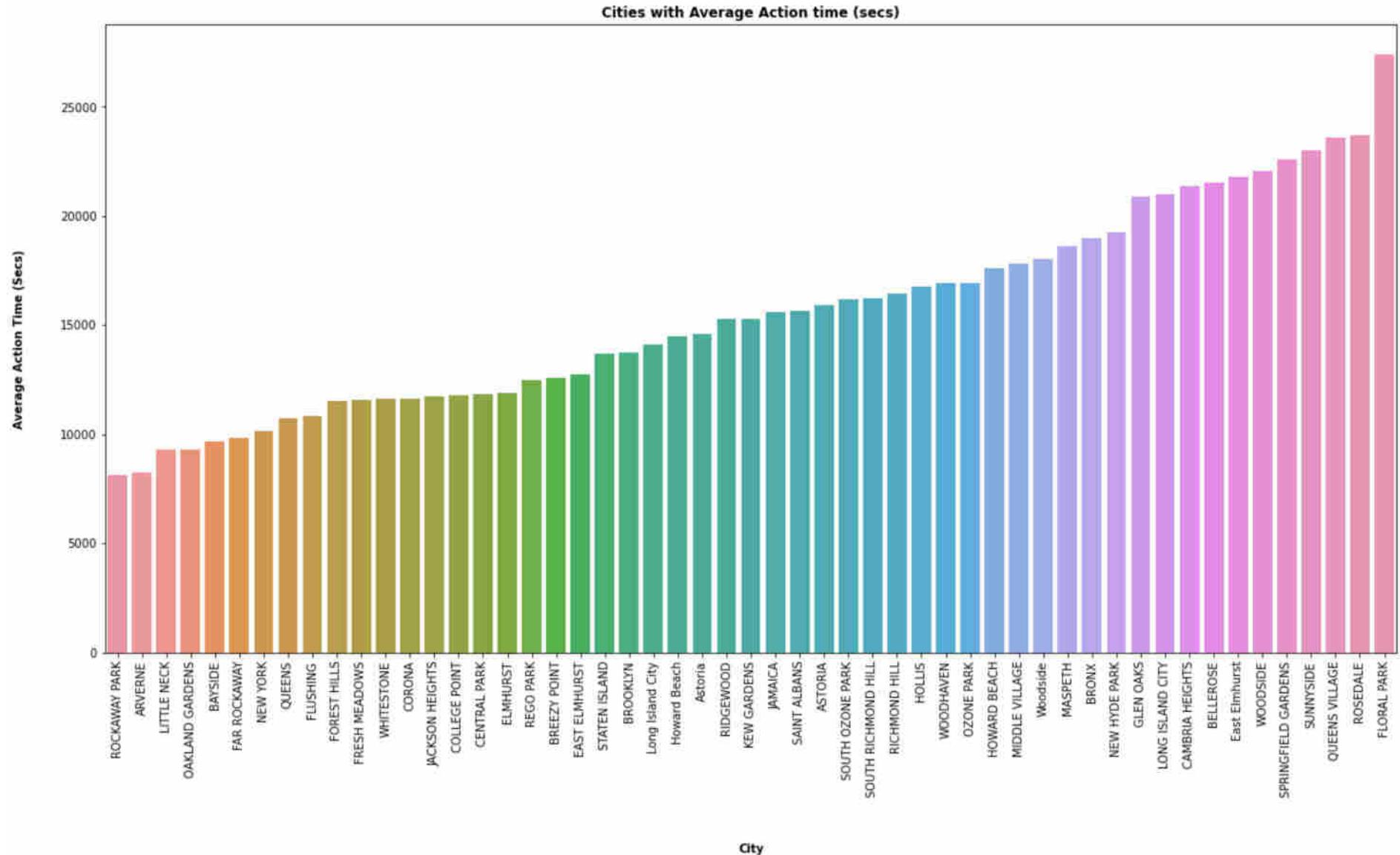


Conclusion 2 :

- Borough wise, Brooklyn has the most number of Complaints followed by Queens and Manhattan.

```
In [32]: # CITY vs Average Action time(Request Closing time) Required for resolving the complaint in that particular city
city_avg_action_time=csr_df[['City','Request_Closing_Time_Secs']]
city_avg_action_time_1= city_avg_action_time.groupby('City')['Request_Closing_Time_Secs'].mean().to_frame()# Average Action time r
city_avg_action_time_1= city_avg_action_time_1.sort_values('Request_Closing_Time_Secs')
city_avg_action_time_1['City']= city_avg_action_time_1.index
txt={'weight':'bold'}
plt.figure(figsize=(20,10))
sns.barplot(x='City',y='Request_Closing_Time_Secs',data=city_avg_action_time_1)
plt.title("Cities with Average Action time (secs)",fontdict=txt)
plt.ylabel("Average Action Time (Secs)",fontdict=txt,labelpad=30)
```

```
plt.xlabel("City",fontdict=txt,labelpad=40)
plt.xticks(rotation=90)
plt.show()
```



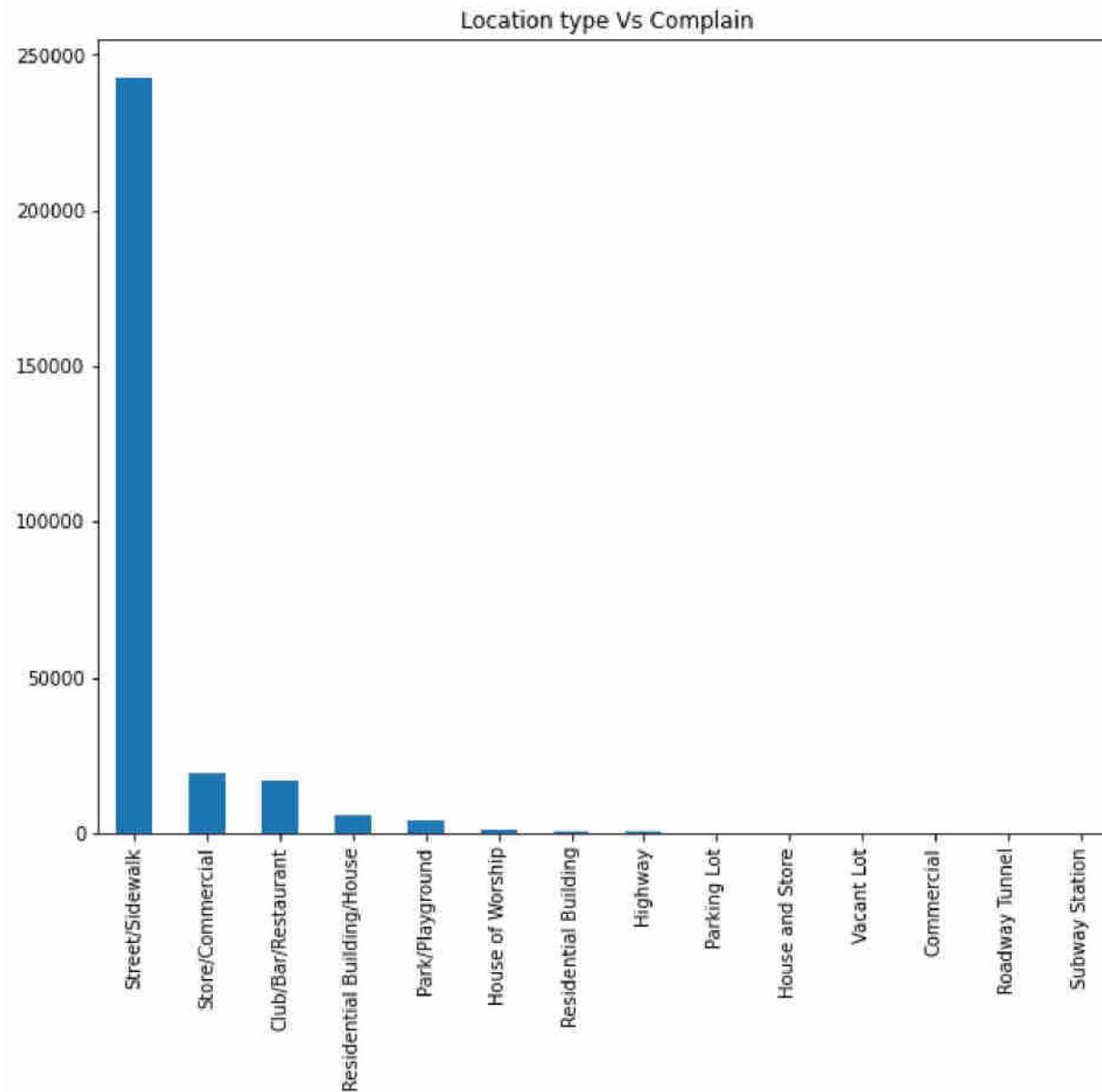
Conclusion 3

- Rockaway park and Arverne have minimum complaint Response time & Floral Park has maximum complaint Response time

In [33]:

```
#Location Type Vs Complaints
csr_df['Location Type'].value_counts().plot(kind='bar', figsize=(10,8), title='Location type Vs Complain')
```

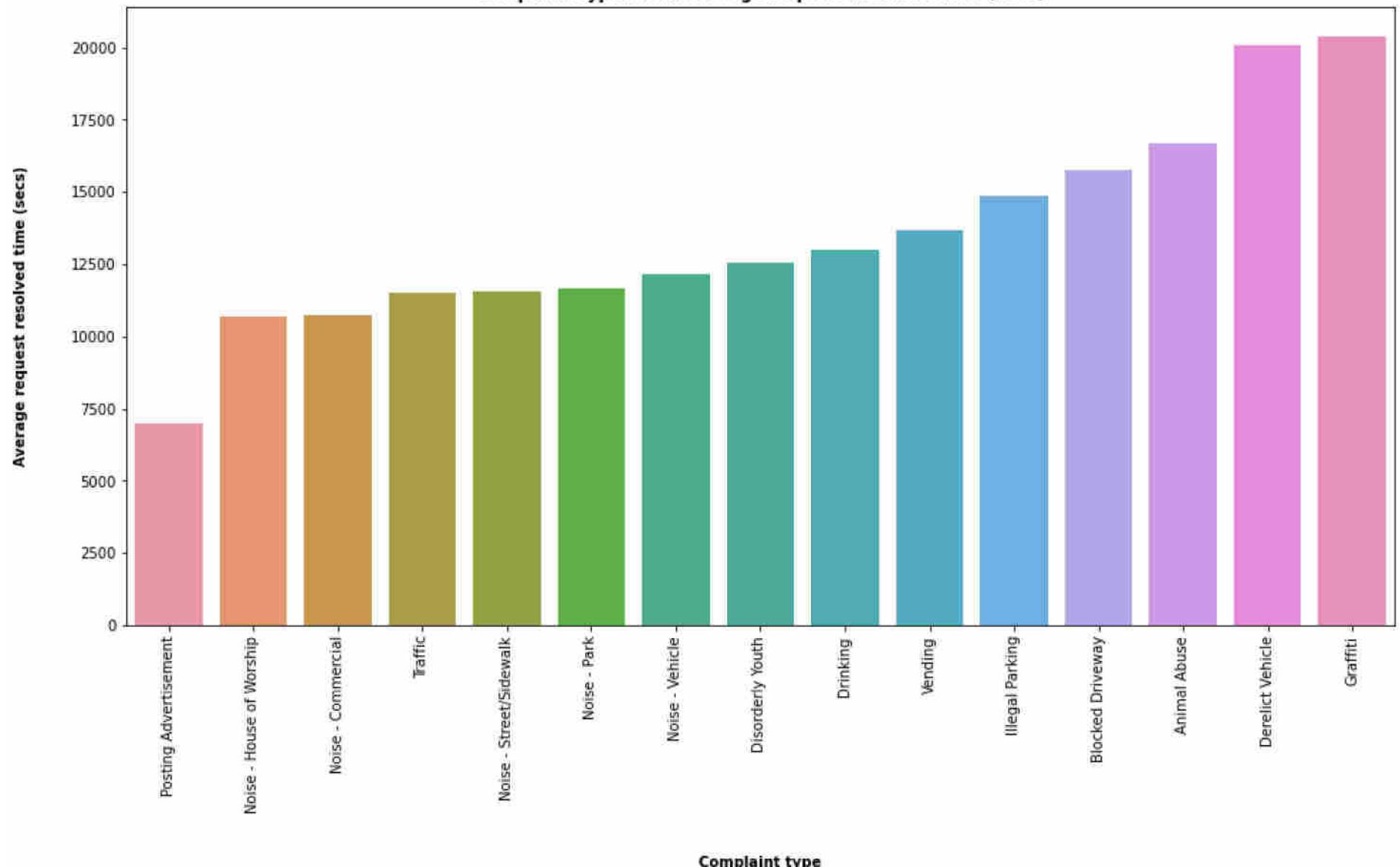
Out[33]: <AxesSubplot:title={'center':'Location type Vs Complain'}>



Conclusion 4

- Most of the Complaints are reported in Location Type 'Street/Sidewalk' followed by 'Store/ Commercial'.

```
In [34]: #visualizing Complaint types with average response time
Comp_Action_time= csr_df[['Complaint Type','Request_Closing_Time_Secs']]
Comp_Action_time2= Comp_Action_time.groupby('Complaint Type')['Request_Closing_Time_Secs'].mean().to_frame()
Comp_Action_time2= Comp_Action_time2.sort_values('Request_Closing_Time_Secs')
Comp_Action_time2['Complaint Type']= Comp_Action_time2.index
txt={'weight':'bold'}
plt.figure(figsize=(16,8))
sns.barplot(x='Complaint Type',y='Request_Closing_Time_Secs',data=Comp_Action_time2)
plt.title("Complaint types with average request resolved time (secs)",fontdict=txt)
plt.xlabel("Complaint type",fontdict=txt,labelpad=30)
plt.ylabel("Average request resolved time (secs)",fontdict=txt,labelpad=30)
plt.xticks(rotation=90)
plt.show()
```

Complaint types with average request resolved time (secs)

Conclusion 5

- Posting Advertistment complaints are responded faster and Derelict Vehicle, Graffiti complaints are responded slower.

4) Ordering the complaint types based on the average 'Request_Closing_Time', grouping them for different locations.

In [35]:

```
# Ordering complaint types based on average Request_Closing_Time, grouping them for different Location Types.
groupedby_complainttype= csr_df.sort_values('Request_Closing_Time_Secs').groupby(['Location Type','Complaint Type'])['Request_Clos
dataFrameByLocationType = pd.DataFrame(groupedby_complainttype)
dataFrameByLocationType.style.set_table_styles([{selector': 'tr', 'props': [({'font-size': '9pt'}, ('border-style', 'solid'), ('border-color', '#ccc'), ('padding', '5px')], 'apply': 'self'}, {'selector': 'td', 'props': [({'font-size': '9pt'}, ('border-style', 'solid'), ('border-color', '#ccc'), ('padding', '5px')], 'apply': 'self'}], [{}], [{}])
```

Out[35]:

		Request_Closing_Time_Secs
Location Type	Complaint Type	
Club/Bar/Restaurant	Drinking	14471.224658
	Noise - Commercial	10402.914743
Commercial	Animal Abuse	16446.870968
Highway	Derelict Vehicle	16212.230769
	Traffic	11947.123656
House and Store	Animal Abuse	16189.677419
House of Worship	Noise - House of Worship	10673.898913
	Animal Abuse	11761.100000
Park/Playground	Drinking	12082.458333
	Noise - Park	11677.570919
Parking Lot	Vending	12508.728155
	Animal Abuse	16005.137615
Residential Building	Posting Advertisement	7616.714286
	Animal Abuse	15844.650442
Residential Building/House	Animal Abuse	17467.388495
	Disorderly Youth	13876.116883
	Drinking	12942.851211

Request_Closing_Time_Secs		
Location Type	Complaint Type	
	Graffiti	18077.964286
	Posting Advertisement	12961.833333
	Vending	14679.651741
Roadway Tunnel	Derelict Vehicle	30111.000000
	Traffic	7837.379310
	Animal Abuse	14908.265385
	Disorderly Youth	10222.125000
	Drinking	11709.344444
Store/Commercial	Graffiti	20018.312500
	Noise - Commercial	11038.686505
	Posting Advertisement	8529.000000
	Vending	13905.817130
Street/Sidewalk	Animal Abuse	15269.866755
	Blocked Driveway	15776.283423
	Derelict Vehicle	20118.126299
	Disorderly Youth	12124.830000
	Drinking	12269.081395
	Graffiti	26055.080000
	Illegal Parking	14876.229142
	Noise - Street/Sidewalk	11584.398325
	Noise - Vehicle	12180.165876
	Posting Advertisement	6399.410345

Request_Closing_Time_Secs		
Location Type	Complaint Type	
	Traffic	11529.945933
	Vending	13629.661508
Subway Station	Animal Abuse	10928.181818
Vacant Lot	Derelict Vehicle	14563.272727

In [36]:

```
# Ordering complaint types based on average Request_Closing_Time, grouping them for cities.
groupedby_complainttype= csr_df.sort_values('Request_Closing_Time_Secs').groupby(['City','Complaint Type'])['Request_Closing_Time_Secs'].mean()
dataFrameByLocationType = pd.DataFrame(groupedby_complainttype).head(60) ## To display top 60 entries only
dataFrameByLocationType.style.set_table_styles([{ 'selector': 'tr', 'props': [ ('font-size', '9pt'), ('border-style', 'solid') ] }, { 'selector': 'th', 'props': [ ('background-color', '#f2f2f2'), ('text-align', 'center') ] }, { 'selector': 'td', 'props': [ ('text-align', 'right') ] }])
```

Out[36]:

Request_Closing_Time_Secs		
City	Complaint Type	
ARVERNE	Animal Abuse	7753.052632
	Blocked Driveway	9093.485714
	Derelict Vehicle	10685.592593
	Disorderly Youth	12928.500000
	Drinking	859.000000
	Graffiti	5520.000000
	Illegal Parking	8338.913793
	Noise - Commercial	8234.000000
	Noise - House of Worship	5623.909091
	Noise - Park	4620.000000
	Noise - Street/Sidewalk	7172.620690
	Noise - Vehicle	6695.571429

Request_Closing_Time_Secs		
City	Complaint Type	
	Vending	1740.000000
	Animal Abuse	18000.608000
	Blocked Driveway	17206.014897
	Derelict Vehicle	26265.914530
	Disorderly Youth	10449.333333
	Drinking	17000.714286
	Graffiti	29142.250000
	Illegal Parking	16832.938028
ASTORIA	Noise - Commercial	11261.369735
	Noise - House of Worship	7280.000000
	Noise - Park	10782.393443
	Noise - Street/Sidewalk	12436.763636
	Noise - Vehicle	12633.034314
	Posting Advertisement	21132.000000
	Traffic	19476.659574
	Vending	17767.500000
	Blocked Driveway	16947.836207
	Derelict Vehicle	15250.166667
Astoria	Illegal Parking	16555.009390
	Noise - Commercial	12419.770115
	Noise - Street/Sidewalk	13367.631579
BAYSIDE	Animal Abuse	11788.243243

Request_Closing_Time_Secs		
City	Complaint Type	
	Blocked Driveway	9227.058355
	Derelict Vehicle	12095.207071
	Disorderly Youth	10703.000000
	Drinking	6840.000000
	Graffiti	16385.666667
	Illegal Parking	9208.288499
	Noise - Commercial	8041.200000
	Noise - House of Worship	12725.500000
	Noise - Park	14320.000000
	Noise - Street/Sidewalk	5511.600000
	Noise - Vehicle	6150.312500
	Traffic	5497.000000
	Vending	6760.500000
BELLEROSE	Animal Abuse	33466.000000
	Blocked Driveway	21806.305263
	Derelict Vehicle	25886.213483
	Disorderly Youth	6660.000000
	Drinking	14105.000000
	Illegal Parking	21380.311321
	Noise - Commercial	14926.243243
	Noise - House of Worship	7909.000000
	Noise - Park	5091.000000

Request_Closing_Time_Secs	
City	Complaint Type
	Noise - Street/Sidewalk 19353.307692
	Noise - Vehicle 9299.400000
	Posting Advertisement 8136.000000
	Traffic 20731.857143

5) Statistical testing

a) Whether the average response time across complaint types is similar or not (overall)?

- **Null Hypothesis : H0 :** There is no significant difference in average response time across different complaint types
 - **Alternate Hypothesis : Ha :** There is a significant difference in average response time across different complaint types
 - To conduct our hypothesis test, we will conduct an ANOVA (analysis of variance) test as we have to compare the means of more than two groups.
 - Assumptions of ANOVA test are

- 1.The samples are independent.
- 2.Each sample is from a normally distributed population.
- 3.The population standard deviations of the groups are all equal.

- Since Alpha value is not given, Assume alpha value = 0.05 ie Testing at Confidence level(95%)

In [38]:

```
# Storing mean response time for various complaint types
complaint_type = csr_df['Complaint Type'].value_counts().index
for i in range(len(complaint_type)):
    exec("sample{} = csr_df.loc[(csr_df['Complaint Type'] == '{}'), 'Request_Closing_Time_Secs']".format(i+1,complaint_type[i]))
```

```
In [39]: # Perform one-way ANOVA.
fscore,pvalue = stats.f_oneway(sample1,sample2,sample3,sample4,sample5,sample6,sample7,sample8,sample9,sample10,sample11,sample12,
print("fscore : {:.2f} , pvalue : {:.2f}".format(fscore,pvalue))

fscore : 691.51 , pvalue : 0.00
```

pvalue (0.00) < alpha value(0.05)

- So we REJECT our Null Hypothesis (Null Hypothesis : H_0 : There is no significant difference in average response time across different complaint types)
- Hence there is a significant difference in average response time across different complaint types

ie **The average response time across different complaint types is not similar (overall)**

b) Are the type of complaint or service requested and location related ?

- Here we are using **Chi-Square Test of Independence**
 - Since Alpha value is not given, Assume alpha value = 0.05 ie Testing at Confidence level(95%)
 - **Null Hypothesis** : H_0 : There is no significant relation between type of complaint and location
 - **Alternate Hypothesis** : H_a : There is some significant relation between type of complaint and location

```
In [40]: # Chi-square test of independence
Chi_sq_complaint_Location_type = pd.crosstab(csr_df['Complaint Type'],csr_df['Location'])
```

```
In [41]: from scipy import stats
chisq,pval,df,et = stats.chi2_contingency(Chi_sq_complaint_Location_type)
print("chisq Score : {:.2f} , pvalue : {:.3f}".format(chisq,pval))
```

chisq Score : 3104085.04 , pvalue : 0.000

pvalue (0.00) < alpha value(0.05)

- So we REJECT our Null Hypothesis (Null Hypothesis : H_0 : There is no significant relation between type of complaint and location)
- There is some significant relation between type of complaint and location

(i.e)**The type of complaint or service requested and the location are related**