

# DATA SCIENCE WITH PYTHON

## PROJECT 1

### CUSTOMER SERVICE REQUEST ANALYSIS

Date : 07.08.2020

# 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

```
In [75]: # Import Libraries
```

```
In [76]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
In [77]: # Import NYC 311's dataset
```

```
In [78]: nyc = pd.read_csv("C:/users/admin/Desktop/311_Service_Request
s_from_2010_to_Present.csv")
```

```
D:\anaconda\lib\site-packages\IPython\core\interactiveshell.
py:3063: DtypeWarning: Columns (48,49) have mixed types.Spec
ify dtype option on import or set low_memory=False.
interactivity=interactivity, compiler=compiler, result=res
ult)
```

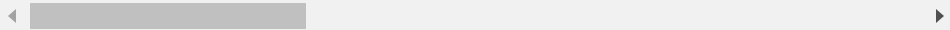
In [79]:

```
nyc
```

Out[79]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complai Tyl
0	32310363	12/31/2015 11:59:45 PM	01-01-16 0:55	NYPD	New York City Police Department	Noise Street/Sidewa
1	32309934	12/31/2015 11:59:44 PM	01-01-16 1:26	NYPD	New York City Police Department	Blocke Drivew:
2	32309159	12/31/2015 11:59:29 PM	01-01-16 4:51	NYPD	New York City Police Department	Blocke Drivew:
3	32305098	12/31/2015 11:57:46 PM	01-01-16 7:43	NYPD	New York City Police Department	Illegal Parkin
4	32306529	12/31/2015 11:56:58 PM	01-01-16 3:24	NYPD	New York City Police Department	Illegal Parkin
...	...	...	...	...	...	...
300693	30281872	03/29/2015 12:33:41 AM	NaN	NYPD	New York City Police Department	Noise Commerc
300694	30281230	03/29/2015 12:33:28 AM	03/29/2015 02:33:59 AM	NYPD	New York City Police Department	Blocke Drivew:
300695	30283424	03/29/2015 12:33:03 AM	03/29/2015 03:40:20 AM	NYPD	New York City Police Department	Noise Commerc
300696	30280004	03/29/2015 12:33:02 AM	03/29/2015 04:38:35 AM	NYPD	New York City Police Department	Noise Commerc
300697	30281825	03/29/2015 12:33:01 AM	03/29/2015 04:41:50 AM	NYPD	New York City Police Department	Noise Commerc

300698 rows × 53 columns



In [80]:

```
# summary of the dataframe
```

In [81]:

```
nyc.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
4			
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
4			
26	Y Coordinate (State Plane)	297158 non-null	float64
4			
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
4			
40	Vehicle Type	0 non-null	float64

```

4
41 Taxi Company Borough      0 non-null    float6
4
42 Taxi Pick Up Location     0 non-null    float6
4
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    float6
4
48 Ferry Direction           1 non-null    object
49 Ferry Terminal Name        2 non-null    object
50 Latitude                   297158 non-null float6
4
51 Longitude                  297158 non-null float6
4
52 Location                   297158 non-null object
dtypes: float64(10), int64(1), object(42)
memory usage: 121.6+ MB

```

In [82]: `nyc.columns`

```

Out[82]: Index(['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',
               'Intersection Street 1', 'Intersection Street 2', 'Address Type',
               'City', 'Landmark', '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',
               'School Name', 'School Number', 'School Region', 'School Code',
               'School Phone Number', 'School Address', 'School City', 'School State',
               'School Zip', 'School Not Found', 'School or Citywide Complaint',
               'Vehicle Type', 'Taxi Company Borough', 'Taxi Pick Up Location',
               'Bridge Highway Name', 'Bridge Highway Direction', 'Road Ramp',
               'Bridge Highway Segment', 'Garage Lot Name', 'Ferry Direction',
               'Ferry Terminal Name', 'Latitude', 'Longitude', 'Location'],
              dtype='object')

```

```

In [83]: # row indexes
         nyc.index.values

```

```

Out[83]: array([    0,     1,     2, ..., 300695, 300696, 300697],
              dtype=int64)

```

```

In [84]: import datetime

```

```

In [85]: t= datetime.datetime.now()
         t

```

```

Out[85]: datetime.datetime(2020, 8, 7, 18, 44, 9, 57214)

```

```

In [86]: print('date is',t.strftime('%x')) # Return date/month/year
         date is 08/07/20

```

```
In [87]: print('hour',t.strftime('%H')) # Return hours
```

```
hour 18
```

```
In [88]: t.strftime('%d') # Return current date
```

```
Out[88]: '07'
```

```
In [89]: t.strftime('%Y') # Return current year
```

```
Out[89]: '2020'
```

```
In [90]: t.strftime('%y') # Return current year
```

```
Out[90]: '20'
```

```
In [91]: # minutes value  
t.strftime('%M') # Return minute
```

```
Out[91]: '44'
```

```
In [92]: t.strftime('%m') # Return current month
```

```
Out[92]: '08'
```

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

```
In [93]: nyc['Created Date']= pd.to_datetime(nyc['Created Date'])  
nyc['Created Date']
```

```
Out[93]: 0          2015-12-31 23:59:45  
1          2015-12-31 23:59:44  
2          2015-12-31 23:59:29  
3          2015-12-31 23:57:46  
4          2015-12-31 23:56:58  
...  
300693     2015-03-29 00:33:41  
300694     2015-03-29 00:33:28  
300695     2015-03-29 00:33:03  
300696     2015-03-29 00:33:02  
300697     2015-03-29 00:33:01  
Name: Created Date, Length: 300698, dtype: datetime64[ns]
```

```
In [94]: nyc['Closed Date']= pd.to_datetime(nyc['Closed Date'])  
nyc['Closed Date']
```

```

Out[94]: 0      2016-01-01 00:55:00
        1      2016-01-01 01:26:00
        2      2016-01-01 04:51:00
        3      2016-01-01 07:43:00
        4      2016-01-01 03:24:00
        ...
        300693      NaT
        300694      2015-03-29 02:33:59
        300695      2015-03-29 03:40:20
        300696      2015-03-29 04:38:35
        300697      2015-03-29 04:41:50
        Name: Closed Date, Length: 300698, dtype: datetime64[ns]

```

```
In [95]: nyc.head()
```

```

Out[95]:

```

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	Descr
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	Music/I
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway	No Ac
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	No Ac
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	Comme Over Pa
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking	Blo Side

5 rows × 53 columns

# create a new column 'Request\_Closing\_Time' as the time elapsed between request creation and request closing.

```

In [96]: nyc['Request_Closing_Time'] = nyc['Closed Date'] - nyc['Created Date']
        nyc['Request_Closing_Time']

```

```
Out[96]: 0      00:55:15
         1      01:26:16
         2      04:51:31
         3      07:45:14
         4      03:27:02
         ...
        300693      NaT
        300694      02:00:31
        300695      03:07:17
        300696      04:05:33
        300697      04:08:49
        Name: Request_Closing_Time, Length: 300698, dtype: timedelta64[ns]
```

```
In [97]: nyc['response_time_days'] = nyc['Request_Closing_Time'] / np.timedelta64(1, 'D')
        nyc['response_time_days']
```

```
Out[97]: 0      0.038368
         1      0.059907
         2      0.202442
         3      0.323079
         4      0.143773
         ...
        300693      NaN
        300694      0.083692
        300695      0.130058
        300696      0.170521
        300697      0.172789
        Name: response_time_days, Length: 300698, dtype: float64
```

```
In [98]: nyc['Request_Closing_Time'] / np.timedelta64(1, 'h')
```

```
Out[98]: 0      0.920833
         1      1.437778
         2      4.858611
         3      7.753889
         4      3.450556
         ...
        300693      NaN
        300694      2.008611
        300695      3.121389
        300696      4.092500
        300697      4.146944
        Name: Request_Closing_Time, Length: 300698, dtype: float64
```

```
In [99]: # missing values for the columns
        nyc.isnull().sum()
```



```

Out[99]: Unique Key                0
Created Date                      0
Closed Date                      2164
Agency                          0
Agency Name                     0
Complaint Type                   0
Descriptor                       5914
Location Type                   131
Incident Zip                     2615
Incident Address                 44410
Street Name                     44410
Cross Street 1                  49279
Cross Street 2                  49779
Intersection Street 1           256840
Intersection Street 2           257336
Address Type                    2815
City                           2614
Landmark                       300349
Facility Type                   2171
Status                          0
Due Date                        3
Resolution Description           0
Resolution Action Updated Date  2187
Community Board                 0
Borough                        0
X Coordinate (State Plane)      3540
Y Coordinate (State Plane)      3540
Park Facility Name              0
Park Borough                    0
School Name                     0
School Number                   0
School Region                   1
School Code                     1
School Phone Number             0
School Address                  0
School City                     0
School State                    0
School Zip                      1
School Not Found                0
School or Citywide Complaint    300698
Vehicle Type                    300698
Taxi Company Borough            300698
Taxi Pick Up Location           300698
Bridge Highway Name             300455
Bridge Highway Direction        300455
Road Ramp                      300485
Bridge Highway Segment          300485
Garage Lot Name                 300698
Ferry Direction                 300697
Ferry Terminal Name             300696

```

Latitude	3540
Longitude	3540
Location	3540
Request_Closing_Time	2164
response_time_days	2164
dtype:	int64

```
In [100]: # Distribution Of Type Complaints
nyc['Complaint Type'].value_counts()
```

```
Out[100]: Blocked Driveway          77044
Illegal Parking          75361
Noise - Street/Sidewalk  48612
Noise - Commercial      35577
Derelict Vehicle        17718
Noise - Vehicle         17083
Animal Abuse            7778
Traffic                 4498
Homeless Encampment     4416
Noise - Park            4042
Vending                 3802
Drinking                1280
Noise - House of Worship 931
Posting Advertisement    650
Urinating in Public      592
Bike/Roller/Skate Chronic 427
Panhandling             307
Disorderly Youth        286
Illegal Fireworks        168
Graffiti               113
Agency Issues           6
Squeegee                4
Ferry Complaint          2
Animal in a Park         1
Name: Complaint Type, dtype: int64
```

```
In [101]: # suppose we want top 10 Complaints
top_10= nyc['Complaint Type'].value_counts()[:10]
top_10
```

```
Out[101]: Blocked Driveway          77044
          Illegal Parking           75361
          Noise - Street/Sidewalk   48612
          Noise - Commercial        35577
          Derelict Vehicle          17718
          Noise - Vehicle           17083
          Animal Abuse              7778
          Traffic                   4498
          Homeless Encampment       4416
          Noise - Park              4042
          Name: Complaint Type, dtype: int64
```

```
In [102]: # For top 20 Complaints
          top_20= nyc['Complaint Type'].value_counts()[:20]
          top_20
```

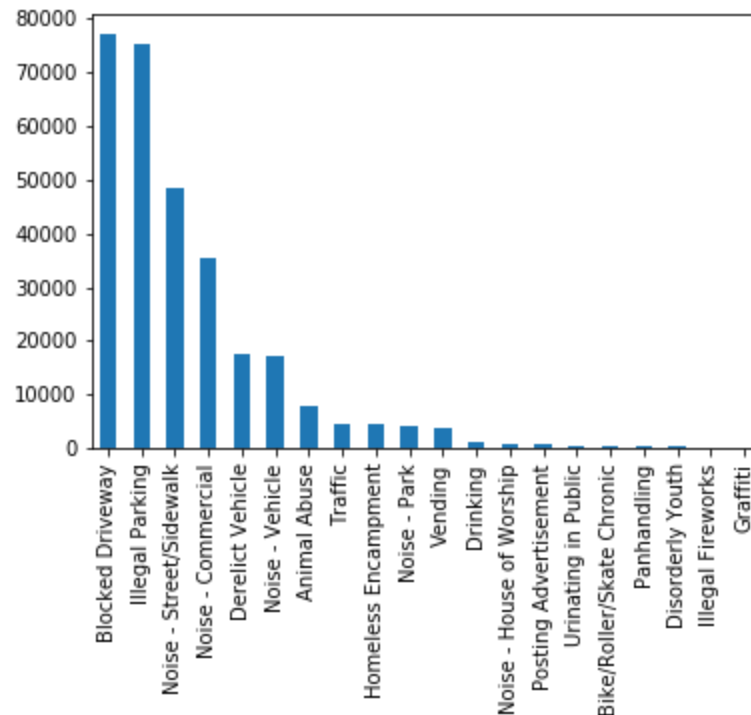
```
Out[102]: Blocked Driveway          77044
          Illegal Parking           75361
          Noise - Street/Sidewalk   48612
          Noise - Commercial        35577
          Derelict Vehicle          17718
          Noise - Vehicle           17083
          Animal Abuse              7778
          Traffic                   4498
          Homeless Encampment       4416
          Noise - Park              4042
          Vending                   3802
          Drinking                  1280
          Noise - House of Worship   931
          Posting Advertisement      650
          Urinating in Public        592
          Bike/Roller/Skate Chronic  427
          Panhandling                307
          Disorderly Youth           286
          Illegal Fireworks          168
          Graffiti                  113
          Name: Complaint Type, dtype: int64
```

```
In [103]: # For top 15 Complaints
          top_15= nyc['Complaint Type'].value_counts()[:20]
          top_15
```

```
Out[103]: Blocked Driveway          77044
          Illegal Parking           75361
          Noise - Street/Sidewalk   48612
          Noise - Commercial        35577
          Derelict Vehicle          17718
          Noise - Vehicle           17083
          Animal Abuse              7778
          Traffic                   4498
          Homeless Encampment       4416
          Noise - Park              4042
          Vending                   3802
          Drinking                  1280
          Noise - House of Worship  931
          Posting Advertisement     650
          Urinating in Public       592
          Bike/Roller/Skate Chronic 427
          Panhandling               307
          Disorderly Youth          286
          Illegal Fireworks         168
          Graffiti                 113
          Name: Complaint Type, dtype: int64
```

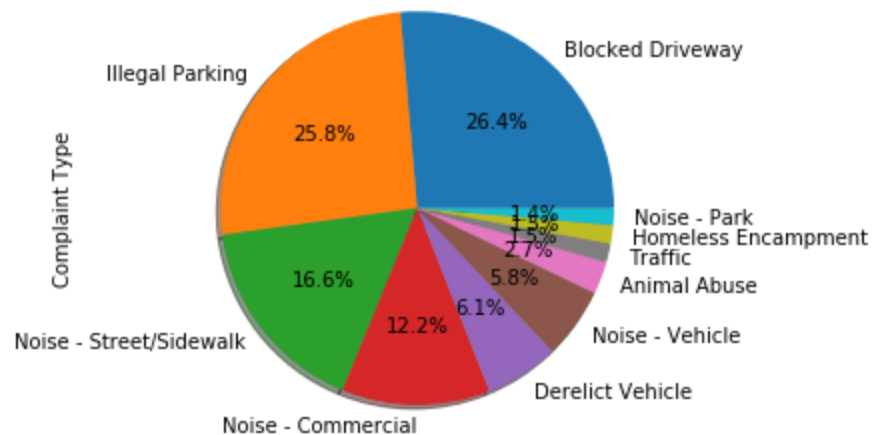
```
In [104]: # Provide major insights/patterns.
```

```
In [105]: # Created Bar plot for the Complaint Types
          top_15.plot(kind='bar')
          plt.show()
```



# Here from the bar plot we observe that, the value of 'Blocked Driveway' Complaint is more than the other types of Complaints.

```
In [106]: top_10.plot(kind='pie', shadow=True, autopct='%1.1f%%')
plt.axis('equal')
plt.show()
```

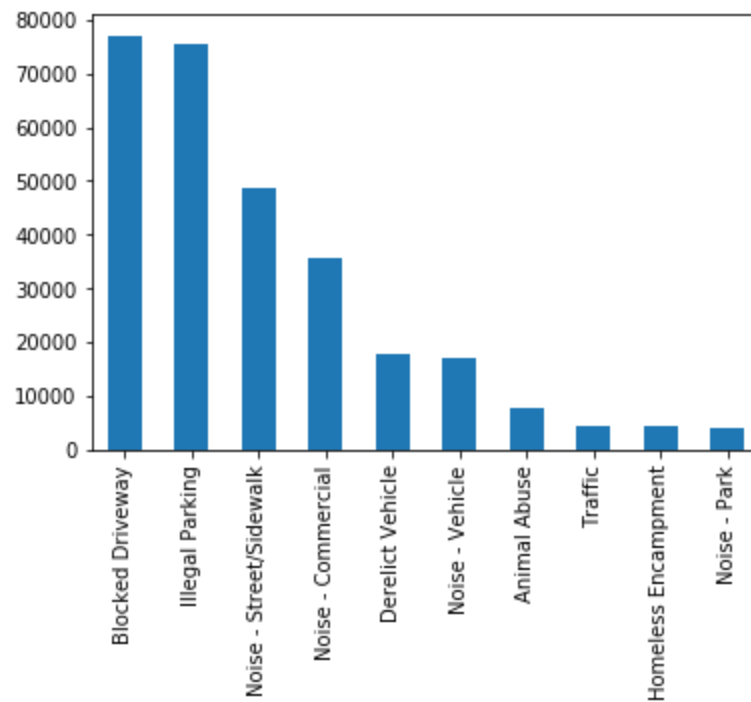


# Here from the above pie chart we observe that the percentage of Blocked Driveway Complaint is higher i.e. about 26.4% than the other type of Complaints.

```
In [107]: # To get Top cities having Complaints
nyc['City'].value_counts()[:10]
```

```
Out[107]: BROOKLYN      98307
          NEW YORK      65994
          BRONX         40702
          STATEN ISLAND 12343
          JAMAICA        7296
          ASTORIA        6330
          FLUSHING       5971
          RIDGEWOOD      5163
          CORONA         4295
          WOODSIDE       3544
          Name: City, dtype: int64
```

```
In [108]: top_10.plot(kind= 'bar')
          plt.show()
```

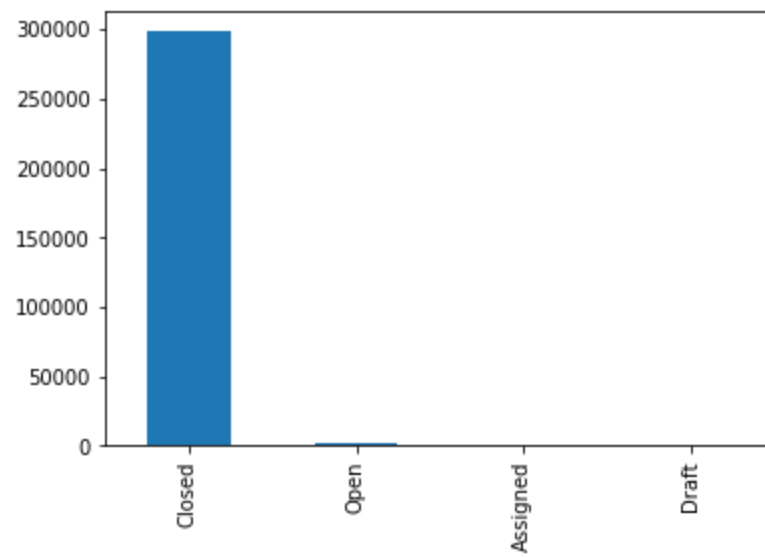


```
In [109]: # Complaint by Status
          nyc['Status'].value_counts()
```

```
Out[109]: Closed      298471
          Open        1439
          Assigned     786
          Draft        2
          Name: Status, dtype: int64
```

```
In [110]: # Complaint by Status bar
          nyc['Status'].value_counts().plot(kind= 'bar')
```

Out[110]: <matplotlib.axes.\_subplots.AxesSubplot at 0x23e50a69bc8>



```
In [111]: nyc['Descriptor'].value_counts()
```

```

Out[111]: Loud Music/Party          61430
          No Access                  56976
          Posted Parking Sign Violation 22440
          Loud Talking                21584
          Partial Access              20068
          With License Plate          17718
          Blocked Hydrant             16081
          Commercial Overnight Parking 12189
          Car/Truck Music             11273
          Blocked Sidewalk            11121
          Double Parked Blocking Traffic 5731
          Double Parked Blocking Vehicle 4211
          Engine Idling               4189
          Banging/Pounding            4165
          Neglected                  3787
          Car/Truck Horn              3511
          Congestion/Gridlock         2761
          In Prohibited Area          2025
          Other (complaint details)   1969
          Unlicensed                  1777
          Overnight Commercial Storage 1757
          Unauthorized Bus Layover     1367
          Truck Route Violation       1014
          In Public                   932
          Tortured                    854
          Vehicle                     590
          Chained                     535
          Detached Trailer            464
          No Shelter                  382
          Chronic Stoplight Violation 280
          Underage - Licensed Est     271
          Chronic Speeding            268
          In Car                      251
          Playing in Unsuitable Place 245
          Drag Racing                 175
          Loud Television              93
          Police Report Requested     90
          After Hours - Licensed Est  77
          Building                    60
          Nuisance/Truant             41
          Police Report Not Requested 23
          Language Access Complaint    6
          Disruptive Passenger         1
          Homeless Issue               1
          Animal Waste                1
          Name: Descriptor, dtype: int64

```

```

In [112]: nyc['Descriptor'].value_counts()[0:10]

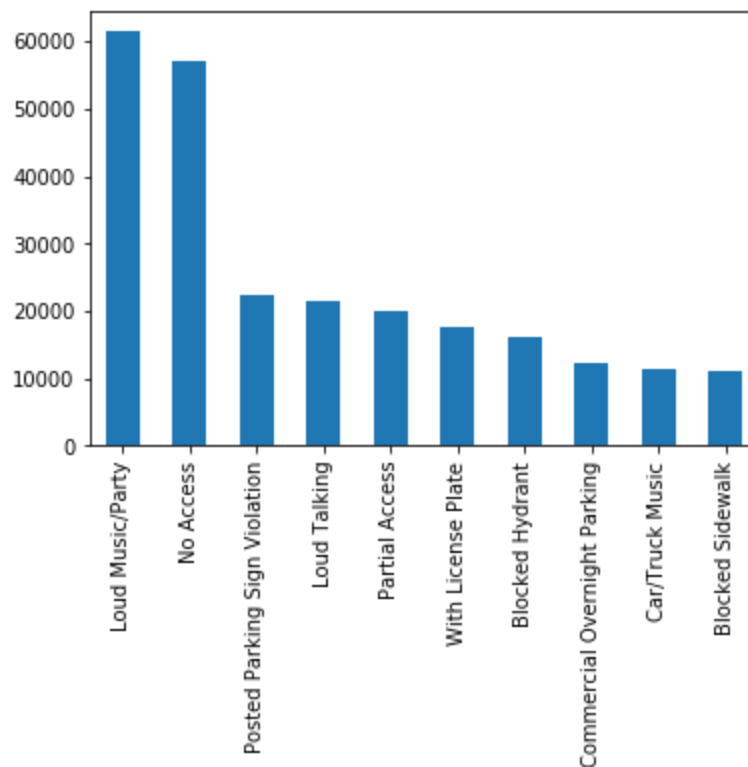
```



```
Out[112]: Loud Music/Party          61430
          No Access                56976
          Posted Parking Sign Violation 22440
          Loud Talking             21584
          Partial Access           20068
          With License Plate       17718
          Blocked Hydrant          16081
          Commercial Overnight Parking 12189
          Car/Truck Music          11273
          Blocked Sidewalk         11121
          Name: Descriptor, dtype: int64
```

```
In [113]: nyc['Descriptor'].value_counts()[0:10].plot(kind='bar')
```

```
Out[113]: <matplotlib.axes._subplots.AxesSubplot at 0x23e52adb908>
```



```
In [114]: # Get the Complaint Types by the City
```

```
In [115]: City_Type = nyc.groupby(['City', 'Complaint Type'])
          City_Type
```

```
Out[115]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x000023E566A1848>
```

```
In [116]: City_Type_Cnt= nyc.groupby(['City','Complaint Type']).size()  
City_Type_Cnt
```

```
Out[116]: City      Complaint Type  
ARVERNE  Animal Abuse      38  
          Blocked Driveway  35  
          Derelict Vehicle  27  
          Disorderly Youth   2  
          Drinking          1  
          ...  
Woodside  Blocked Driveway  11  
          Derelict Vehicle   2  
          Illegal Parking    100  
          Noise - Commercial  2  
          Noise - Street/Sidewalk  5  
Length: 764, dtype: int64
```

```
In [117]: # If we want the City - Complaint Type in df
```

```
In [118]: City_Type_1= City_Type_Cnt.unstack()  
City_Type_1
```

Out[118]:

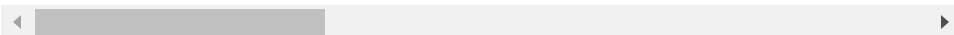
	Complaint Type	Animal Abuse	Blocked Driveway	Derelict Vehicle	Disorderly Youth	Drinking	Graffiti	En
City								
ARVERNE		38.0	35.0	27.0	2.0	1.0	1.0	
ASTORIA		125.0	2618.0	351.0	3.0	35.0	4.0	
Astoria		NaN	116.0	12.0	NaN	NaN	NaN	
BAYSIDE		37.0	377.0	198.0	1.0	1.0	3.0	
BELLEROSE		7.0	95.0	89.0	2.0	1.0	NaN	
BREEZY POINT		2.0	3.0	3.0	NaN	1.0	NaN	
BRONX		1415.0	12755.0	1953.0	63.0	188.0	9.0	
BROOKLYN		2394.0	28148.0	5181.0	72.0	257.0	43.0	
CAMBRIA HEIGHTS		11.0	147.0	115.0	NaN	NaN	NaN	
CENTRAL PARK		NaN	NaN	NaN	NaN	NaN	NaN	
COLLEGE POINT		28.0	435.0	184.0	1.0	NaN	1.0	
CORONA		61.0	2761.0	57.0	6.0	33.0	2.0	
EAST ELMHURST		59.0	1408.0	113.0	1.0	9.0	3.0	
ELMHURST		38.0	1446.0	78.0	2.0	13.0	NaN	
East Elmhurst		NaN	NaN	1.0	NaN	NaN	NaN	
FAR ROCKAWAY		89.0	284.0	187.0	1.0	4.0	NaN	
FLORAL PARK		2.0	20.0	56.0	1.0	1.0	NaN	
FLUSHING		143.0	2795.0	440.0	2.0	40.0	4.0	
FOREST HILLS		45.0	663.0	52.0	1.0	1.0	3.0	
FRESH MEADOWS		45.0	503.0	291.0	NaN	2.0	NaN	
GLEN OAKS		5.0	30.0	49.0	NaN	NaN	NaN	
HOLLIS		33.0	342.0	143.0	1.0	3.0	NaN	
HOWARD BEACH		31.0	167.0	138.0	1.0	4.0	NaN	



Out[118]:

	Complaint Type	Animal Abuse	Blocked Driveway	Derelict Vehicle	Disorderly Youth	Drinking	Graffiti	En
City								
RICHMOND HILL		26.0	1548.0	289.0	2.0	23.0	NaN	
SPRINGFIELD GARDENS		24.0	262.0	210.0	NaN	6.0	NaN	
STATEN ISLAND		557.0	2142.0	1766.0	23.0	175.0	2.0	
SUNNYSIDE		35.0	206.0	10.0	2.0	10.0	1.0	
WHITESTONE		28.0	208.0	227.0	1.0	2.0	1.0	
WOODHAVEN		45.0	1060.0	308.0	NaN	3.0	NaN	
WOODSIDE		69.0	1613.0	247.0	1.0	15.0	3.0	
Woodside		NaN	11.0	2.0	NaN	NaN	NaN	

53 rows × 22 columns



In [119]: City\_Type\_1.sum(axis=0)

```
Out[119]: Complaint Type
Animal Abuse          7767.0
Blocked Driveway     76761.0
Derelict Vehicle     17547.0
Disorderly Youth      286.0
Drinking             1272.0
Graffiti             113.0
Homeless Encampment   4415.0
Illegal Parking      74318.0
Noise - Commercial   35200.0
Noise - House of Worship  929.0
Noise - Park         4016.0
Noise - Street/Sidewalk 48000.0
Noise - Vehicle     17029.0
Panhandling           305.0
Urinating in Public   592.0
Vending              3795.0
Bike/Roller/Skate Chronic  422.0
Illegal Fireworks     168.0
Posting Advertisement  649.0
Traffic               4495.0
Squeegee              4.0
Animal in a Park       1.0
dtype: float64
```

```
In [120]: # Get Total Complaint by City (row wise)
```

```
In [121]: City_Type_1.sum(axis=1)
```

```

Out[121]: City
          ARVERNE                220.0
          ASTORIA                6330.0
          Astoria                717.0
          BAYSIDE                1221.0
          BELLEROSE              375.0
          BREEZY POINT           30.0
          BRONX                  40702.0
          BROOKLYN              98307.0
          CAMBRIA HEIGHTS        477.0
          CENTRAL PARK           97.0
          COLLEGE POINT          1220.0
          CORONA                 4295.0
          EAST ELMHURST          2734.0
          ELMHURST               2673.0
          East Elmhurst          14.0
          FAR ROCKAWAY           1179.0
          FLORAL PARK            152.0
          FLUSHING               5971.0
          FOREST HILLS           1688.0
          FRESH MEADOWS          1899.0
          GLEN OAKS              306.0
          HOLLIS                 1012.0
          HOWARD BEACH           931.0
          Howard Beach           1.0
          JACKSON HEIGHTS        1689.0
          JAMAICA                7296.0
          KEW GARDENS            771.0
          LITTLE NECK           559.0
          LONG ISLAND CITY        2437.0
          Long Island City       134.0
          MASPETH                2462.0
          MIDDLE VILLAGE         1765.0
          NEW HYDE PARK           98.0
          NEW YORK               65994.0
          OAKLAND GARDENS        551.0
          OZONE PARK             2755.0
          QUEENS                 32.0
          QUEENS VILLAGE         1814.0
          REGO PARK              1486.0
          RICHMOND HILL          1904.0
          RIDGEWOOD              5163.0
          ROCKAWAY PARK          745.0
          ROSEDALE               922.0
          SAINT ALBANS           834.0
          SOUTH OZONE PARK        2173.0
          SOUTH RICHMOND HILL     2774.0
          SPRINGFIELD GARDENS     883.0
          STATEN ISLAND          12343.0
          SUNNYSIDE              723.0

```

```
WHITESTONE          1098.0
WOODHAVEN           2464.0
WOODSIDE            3544.0
Woodside            120.0
dtype: float64
```

```
In [122]: # Average request Closing Time by the Type Of Complaint
nyc.groupby('Complaint Type')['response_time_days'].mean()
```

```
Out[122]: Complaint Type
Agency Issues          0.219180
Animal Abuse           0.217218
Animal in a Park       14.034780
Bike/Roller/Skate Chronic 0.156936
Blocked Driveway       0.197538
Derelict Vehicle       0.306839
Disorderly Youth       0.148274
Drinking               0.160910
Ferry Complaint        NaN
Graffiti              0.297969
Homeless Encampment    0.181899
Illegal Fireworks      0.115047
Illegal Parking        0.187548
Noise - Commercial     0.131131
Noise - House of Worship 0.133054
Noise - Park           0.142114
Noise - Street/Sidewalk 0.143551
Noise - Vehicle        0.149541
Panhandling            0.182199
Posting Advertisement   0.082325
Squeegee              0.168568
Traffic                0.143695
Urinating in Public    0.151111
Vending               0.167247
Name: response_time_days, dtype: float64
```

```
In [123]: nyc.columns
```



```

Out[123]: Index(['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',
                'Intersection Street 1', 'Intersection Street 2', 'Address Type',
                'City', 'Landmark', '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',
                'School Name', 'School Number', 'School Region', 'School Code',
                'School Phone Number', 'School Address', 'School City', 'School State',
                'School Zip', 'School Not Found', 'School or Citywide Complaint',
                'Vehicle Type', 'Taxi Company Borough', 'Taxi Pick Up Location',
                'Bridge Highway Name', 'Bridge Highway Direction', 'Road Ramp',
                'Bridge Highway Segment', 'Garage Lot Name', 'Ferry Direction',
                'Ferry Terminal Name', 'Latitude', 'Longitude', 'Location',
                'Request_Closing_Time', 'response_time_days'],
                dtype='object')

```

```

In [124]: # Average Request Closing Time by the type of Complaints

```

```

In [125]: nyc.groupby(['Complaint Type', 'City'])['response_time_days'].mean()

```

```
Out[125]: Complaint Type City
Animal Abuse ARVERNE 0.089734
ASTORIA 0.208340
BAYSIDE 0.136438
BELLEROSE 0.530195
BREEZY POINT 0.108895
...
Vending STATEN ISLAND 0.177979
SUNNYSIDE 0.405143
WHITESTONE 0.097222
WOODHAVEN 0.118436
WOODSIDE 0.304254
Name: response_time_days, Length: 764, dtype: float64
```

```
In [126]: Type_resonse_days= nyc.groupby(['Complaint Type','City'])['response_time_days'].mean()
Type_resonse_days
```

```
Out[126]: Complaint Type City
Animal Abuse ARVERNE 0.089734
ASTORIA 0.208340
BAYSIDE 0.136438
BELLEROSE 0.530195
BREEZY POINT 0.108895
...
Vending STATEN ISLAND 0.177979
SUNNYSIDE 0.405143
WHITESTONE 0.097222
WOODHAVEN 0.118436
WOODSIDE 0.304254
Name: response_time_days, Length: 764, dtype: float64
```

```
In [127]: # Perform a statistical test
```

# Whether the average response time across Complaint Types is similar or not (overall) #  
 response\_time\_days value is changing by different Complaint Types. # H0: The average  
 response\_time\_days is the same for the Complaints. # H1: The average response\_time\_days is  
 not the same for the Complaints.

```
In [128]: nyc['Complaint Type'].value_counts()
```

```

Out[128]: Blocked Driveway          77044
          Illegal Parking           75361
          Noise - Street/Sidewalk    48612
          Noise - Commercial         35577
          Derelict Vehicle           17718
          Noise - Vehicle            17083
          Animal Abuse               7778
          Traffic                    4498
          Homeless Encampment        4416
          Noise - Park               4042
          Vending                    3802
          Drinking                   1280
          Noise - House of Worship    931
          Posting Advertisement       650
          Urinating in Public         592
          Bike/Roller/Skate Chronic   427
          Panhandling                307
          Disorderly Youth           286
          Illegal Fireworks           168
          Graffiti                   113
          Agency Issues               6
          Squeegee                   4
          Ferry Complaint             2
          Animal in a Park            1
          Name: Complaint Type, dtype: int64

```

# Creat a dataframe where we have these top 5 reasons. # Here, Independent variable is response time and dependent variable is complaint type group.

```

In [129]: df_1= nyc[nyc['Complaint Type']=='Blocked Driveway']

          df_1.head()

          Type1= df_1.response_time_days.dropna()

```

# Remove the missing values.

```

In [130]: df_2= nyc[nyc['Complaint Type']=='Illegal parking']
          df_3= nyc[nyc['Complaint Type']=='Noise_street/ Sidewalk']
          df_4= nyc[nyc['Complaint Type']=='Noise- commercial']
          df_5= nyc[nyc['Complaint Type']=='Derelict vehicle']

```

```

In [131]: Type2= df_2.response_time_days.dropna()
          Type3= df_3.response_time_days.dropna()
          Type4= df_4.response_time_days.dropna()
          Type5= df_5.response_time_days.dropna()

```

```

In [132]: sum(Type2.dropna().isnull())

```

Out[132]: 0

```
In [133]: sum(Type1.dropna().isnull())
```

Out[133]: 0

# Get the corresponding values for response\_time\_days

```
In [134]: from scipy import stats
```

```
In [135]: stats.f_oneway(Type1, Type2, Type3, Type4, Type5)
```

```
D:\anaconda\lib\site-packages\scipy\stats\stats.py:3339: RuntimeWarning: invalid value encountered in double_scalars
  ssbn += _square_of_sums(a - offset) / len(a)
```

Out[135]: F\_onewayResult(statistic=nan, pvalue=nan)

# If p- value is less than 0.05, then we reject H0. Hence the average response time across the top5 Type is different. # The dependent variables are type of Complaints. H0 : Two categories Complaint Type and Location is dependent. H1 : Two categories Complaint Type and Location is independent. # This is chisquare test . Here we take top 5 city and top 5 States and get distinct count. # top 5 Complaint types by city

```
In [136]: top_city = nyc['City'].value_counts()[:5]
top_city
```

Out[136]:

BROOKLYN	98307
NEW YORK	65994
BRONX	40702
STATEN ISLAND	12343
JAMAICA	7296

Name: City, dtype: int64

```
In [137]: top_city.index
```

Out[137]: Index(['BROOKLYN', 'NEW YORK', 'BRONX', 'STATEN ISLAND', 'JAMAICA'], dtype='object')

```
In [138]: top_Complaints = nyc['Complaint Type'].value_counts()[0:5]
top_Complaints
```

Out[138]:

Blocked Driveway	77044
Illegal Parking	75361
Noise - Street/Sidewalk	48612
Noise - Commercial	35577
Derelict Vehicle	17718

Name: Complaint Type, dtype: int64

```
In [139]: top_Complaints.index
```

```
Out[139]: Index(['Blocked Driveway', 'Illegal Parking', 'Noise - Street/Sidewalk',  
                'Noise - Commercial', 'Derelict Vehicle'],  
               dtype='object')
```

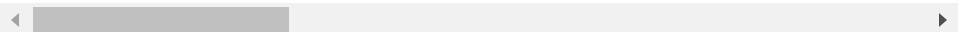
# Now we can filter dataframe for the top5 Complaint Types

```
In [140]: df1= nyc[nyc['Complaint Type'].isin(top_Complaints.index)]  
df1
```

Out[140]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk
1	32309934	2015-12-31 23:59:44	2016-01-01 01:26:00	NYPD	New York City Police Department	Blocked Driveway
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking
4	32306529	2015-12-31 23:56:58	2016-01-01 03:24:00	NYPD	New York City Police Department	Illegal Parking
...	...	...	...	...	...	...
300693	30281872	2015-03-29 00:33:41	NaT	NYPD	New York City Police Department	Noise - Commercial
300694	30281230	2015-03-29 00:33:28	2015-03-29 02:33:59	NYPD	New York City Police Department	Blocked Driveway
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial

254312 rows × 55 columns

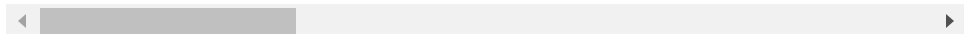


```
In [141]: df2= df1[df1['City'].isin(top_city.index)]
df2
```

Out[141]:

	Unique Key	Created Date	Closed Date	Agency	Agency Name	Complaint Type	
0	32310363	2015-12-31 23:59:45	2016-01-01 00:55:00	NYPD	New York City Police Department	Noise - Street/Sidewalk	I
2	32309159	2015-12-31 23:59:29	2016-01-01 04:51:00	NYPD	New York City Police Department	Blocked Driveway	
3	32305098	2015-12-31 23:57:46	2016-01-01 07:43:00	NYPD	New York City Police Department	Illegal Parking	(
5	32306554	2015-12-31 23:56:30	2016-01-01 01:50:00	NYPD	New York City Police Department	Illegal Parking	
6	32306559	2015-12-31 23:55:32	2016-01-01 01:53:00	NYPD	New York City Police Department	Illegal Parking	
...	...	...	...	...	...	...	
300691	30279999	2015-03-29 00:35:04	2015-03-29 02:52:28	NYPD	New York City Police Department	Noise - Commercial	I
300692	30281370	2015-03-29 00:34:32	2015-03-29 01:13:01	NYPD	New York City Police Department	Noise - Commercial	I
300695	30283424	2015-03-29 00:33:03	2015-03-29 03:40:20	NYPD	New York City Police Department	Noise - Commercial	I
300696	30280004	2015-03-29 00:33:02	2015-03-29 04:38:35	NYPD	New York City Police Department	Noise - Commercial	I
300697	30281825	2015-03-29 00:33:01	2015-03-29 04:41:50	NYPD	New York City Police Department	Noise - Commercial	I

185475 rows × 55 columns



# Now, Get the cross tab frequency

```
In [142]: city_com_top5= pd.crosstab(df2['Complaint Type'],df2['City'])
city_com_top5
```

Out[142]:

	City	BRONX	BROOKLYN	JAMAICA	NEW YORK	STATEN ISLAND
Complaint Type						
Blocked Driveway		12755	28148	2818	2072	2142
Derelict Vehicle		1953	5181	954	537	1766
Illegal Parking		7859	27462	1421	12128	4886
Noise - Commercial		2434	11463	429	14550	678
Noise - Street/Sidewalk		8892	13356	339	20433	819

```
In [143]: stats.chi2_contingency(city_com_top5)
```

```
Out[143]: (40522.79928349593,  
0.0,  
16,  
array([[ 8759.46060116, 22125.43658175, 1540.58786899, 128  
49.86224559,  
2659.65270252],  
[ 1898.81203936, 4796.19091522, 333.95741205, 27  
85.49950128,  
576.54013209],  
[ 9823.16812508, 24812.24510042, 1727.6695835 , 144  
10.28882599,  
2982.62836501],  
[ 5400.58618143, 13641.28825987, 949.83902952, 79  
22.49564631,  
1639.79088287],  
[ 8010.97305297, 20234.83914274, 1408.94610594, 117  
51.85378083,  
2432.38791751]]))
```

```
In [144]: chi_square,p_value,degrees_of_freedom,exp_freq=stats.chi2_con  
tingency(city_com_top5)
```

```
In [145]: p_value
```

Out[145]: 0.0

# Since the p- value is less than 0.05 , then we reject the null hypothesis. Hence the city and the Complaint Types are not independent on each other,they are related.

```
In [146]: degrees_of_freedom
```

Out[146]: 16



```
In [147]: chi_square
```

```
Out[147]: 40522.79928349593
```

# If the p- value is less than 0.05 , then we reject the null hypothesis. Here, from the above result we observe that p-value is 0 with 16 degrees of freedom. Hence we reject the null hypothesis i.e. the city and the Complaint Types are not independent on each other i.e. they are related. # The value of chi-square test is 40522.79928349593

```
In [148]: ## THANK YOU ##
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```