

```
Project title: Customer service requests analysis
April 30,2023

In [159]: #Import important libraries for this dataset
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

In [160]: #Uploading a csv file
df=pd.read_csv('311_Service_Requests_from_2010_to_Present.csv')

/usr/local/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3063: DtypeWarning: Columns (48,49) have mixed
Dtype. Specify dtype option on import or set low_memory=False.
interactivity=interactivity, compiler=compiler, result=result)

In [161]: #Finding the first five rows of the dataset
df.head()

Out[161]:
   Unique Key  Created Date  Closed Date  Agency  Agency Name  Complaint Type  Descriptor  Location Type  Incident Zip  Incident Address  Bridge Highway Name  Bridge Highway Direction  Bridge Ramp  Bt High
0  32310363  12/31/2015  01/01/2016  NYPD  New York City Police Department  Noise - Street/Sidewalk  Loud Music/Party  Street/Sidewalk  10034.0  VERMILION AVENUE  N  NaN  NaN  NaN
1  32309934  11/30/15  01/01/2016  NYPD  New York City Police Department  Blocked Driveway  No Access  Street/Sidewalk  11105.0  27-07-23 AVENUE  N  NaN  NaN  NaN
2  32309159  11/30/15  04/10/16  NYPD  New York City Police Department  Blocked Driveway  No Access  Street/Sidewalk  10458.0  2897 VALENTINE AVENUE  N  NaN  NaN  NaN
3  32305088  11/30/15  07/13/17  NYPD  New York City Police Department  Commercial Overweight Parking  Street/Sidewalk  10461.0  2940 BAELEY AVENUE  N  NaN  NaN  NaN
4  32306529  12/31/2015  03/24/16  NYPD  New York City Police Department  Illegal Parking  Blocked Driveway  Street/Sidewalk  11373.0  87-14-57 ROAD  N  NaN  NaN  NaN

5 rows x 53 columns

In [162]: #Finding the last five rows of the dataset
df.tail()

Out[162]:
   Unique Key  Created Date  Closed Date  Agency  Agency Name  Complaint Type  Descriptor  Location Type  Incident Zip  Incident Address  Bridge Highway Name  Bridge Highway Direction  Bridge Ramp  Road
36455 2600993 01/01/2015 01/01/2016 NYPD New York City Police Department Illegal Parking Blocked Driveway Street/Sidewalk 11421.0 84-25-85 ROAD N NaN NaN NaN
36456 2600362 12/04/15 02/28/16 NYPD New York City Police Department Noise - Vehicle Car/Truck Non Street/Sidewalk 10488.0 SEDGWICK AVENUE N NaN NaN NaN
36458 2600759 01/01/2015 01/01/2015 NYPD New York City Police Department Noise - Street/Sidewalk Load Street/Sidewalk 10031.0 908 WEST STREET N NaN NaN NaN
36459 2961089 01/01/2015 01/01/2015 NYPD New York City Police Department Blocked Driveway No Access Street/Sidewalk 10468.0 931 EAST 135 STREET N NaN NaN NaN
36457 2961185 12/09/15 02/47/16 NYPD New York City Police Department Blocked Driveway No Access Street/Sidewalk 11420.0 123-19 135 STREET N NaN NaN NaN

5 rows x 53 columns

In [163]: #df gives the descriptive statistics of numerical columns#
df.describe()

Out[163]:
   Unique Key  Incident Zip  X Coordinate (State Plane)  Y Coordinate (State Plane)  School Of Citywide Complaint  Vehicle Type  Taxi Company Borough  Taxi Pick Up Location  Carage Lot  Latitude  Longitude
count  3.64558e+05  361660.000000  3.605280e+06  360528.000000  0.0  0.0  0.0  0.0  0.0  360528.000000  360528.000000
mean  3.106959e+07  8078.496599  1.005043e+06  203425.305782  NaN  NaN  NaN  NaN  NaN  40.724960  -73.824960
std  7.331531e+05  678.263114  2.190382e+04  26842.120267  NaN  NaN  NaN  NaN  NaN  0.081067  0.079213
min  2.960737e+07  83.000000  9.133070e+05  12118.000000  NaN  NaN  NaN  NaN  NaN  40.499040  -74.264970
25%  3.048999e+07  10018.000000  9.838400e+05  162948.000000  NaN  NaN  NaN  NaN  NaN  40.809742  -73.972283
50%  3.106787e+07  11249.000000  1.020470e+06  203250.000000  NaN  NaN  NaN  NaN  NaN  40.718465  -73.820643
75%  3.187433e+07  11238.000000  1.039134e+06  222780.000000  NaN  NaN  NaN  NaN  NaN  40.778166  -73.874008
max  3.231058e+07  11697.000000  1.067386e+06  271876.000000  NaN  NaN  NaN  NaN  NaN  40.912669  -73.700715

In [164]: #Finding the information of the dataset
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 364558 entries, 0 to 364557
Data columns (total 53 columns):
 # Column Non-Null Count Dtype
 0 Unique Key 364558 non-null int64
 1 Created Date 364558 non-null object
 2 Closed Date 362177 non-null object
 3 Agency 364558 non-null object
 4 Agency Name 364558 non-null object
 5 Complaint Type 364558 non-null object
 6 Descriptor 358057 non-null object
 7 Location Type 364425 non-null object
 8 Incident Zip 316568 non-null float64
 9 Incident Address 312859 non-null object
10 Street Name 312859 non-null object
11 Cross Street 1 307378 non-null object
12 Cross Street 2 308753 non-null object
13 Intersection Street 1 51129 non-null object
14 Intersection Street 2 58512 non-null object
15 Address Type 561386 non-null object
16 City 361561 non-null object
17 Landmark 375 non-null object
18 Facility Type 362189 non-null object
19 Status 364558 non-null object
20 Due Date 36455 non-null object
21 Resolution Description 364558 non-null object
22 Resolution Action updated Date 362156 non-null object
23 Community Board 364558 non-null object
24 Borough 364558 non-null object
25 X Coordinate (State Plane) 368528 non-null float64
26 Y Coordinate (State Plane) 368528 non-null float64
27 Park Facility Name 364558 non-null object
28 Park Borough 364558 non-null object
29 School Name 364558 non-null object
30 School Number 364558 non-null object
31 School Region 364557 non-null object
32 School Code 364557 non-null object
33 School Phone Number 364558 non-null object
34 School Address 364558 non-null object
35 School City 364557 non-null object
36 School State 364558 non-null object
37 School Zip 364557 non-null object
38 School Not Found 364558 non-null object
39 School or Citywide Complaint 9 non-null float64
40 Vehicle Type 9 non-null float64
41 Taxi Company Borough 9 non-null float64
42 Taxi Pick Up Location 252 non-null float64
43 Bridge Highway Name 297 non-null object
44 Bridge Highway Direction 297 non-null object
45 Road Ramp 262 non-null object
46 Bridge Highway Segment 0 non-null float64
47 Garage Lot Name 6 non-null float64
48 Ferry Direction 1 non-null object
49 Ferry Terminal Name 364556 non-null object
50 Latitude 368528 non-null float64
51 Longitude 368528 non-null float64
52 Location 368528 non-null object
dtypes: float64(18), int64(1), object(42)
memory usage: 347.4+ MB

In [165]: #df gives dimensional of dataset i.e row and column
df.ndim

Out[165]: 2

In [166]: #Identifying the shape of the dataset
df.shape

Out[166]: (364558, 53)

In [167]: #We finding the null columns in the dataset
df.isna().sum()

Out[167]:
Unique Key      0
Created Date      0
Closed Date      2381
Agency           0
Agency Name      0
Complaint Type    0
Descriptor        6551
Location Type     133
Incident Zip     51999
Incident Address 51699
Street Name      57186
Cross Street 1   57805
Cross Street 2   57805
Intersection Street 1 313458
Intersection Street 2 314846
Address Type     3152
City             2997
Landmark         36453
Facility Type    2389
Status           3
Due Date         3
Resolution Description 8
Resolution Action updated Date 2462
Community Board  8
Borough          8
X Coordinate (State Plane) 4838
Y Coordinate (State Plane) 4838
Park Facility Name 8
Park Borough     8
School Name      1
School Number    8
School Region    71
School Code      1
School Phone Number 8
School Address   8
School City      8
School State     8
School Zip       8
School Not Found 8
School or Citywide Complaint 364558
Vehicle Type     364558
Taxi Company Borough 364558
Taxi Pick Up Location 364558
Bridge Highway Name 364261
Bridge Highway Direction 364261
Road Ramp        364296
Bridge Highway Segment 364296
Garage Lot Name  364558
Ferry Direction  364557
Ferry Terminal Name 364556
Latitude         4838
Longitude        4838
Location         4838
dtype: int64

In [168]: #Replacing the missing values,but,here instead of filling na values(fillna()),nd dropping na columns(dropna())##
df = df.drop(['School or Citywide Complaint', 'Vehicle Type', 'Taxi Company Borough', 'Taxi Pick Up Location', 'Garage Lot Name'], axis=1)
df.head()

Out[168]:
   Unique Key  Created Date  Closed Date  Agency  Agency Name  Complaint Type  Descriptor  Location Type  Incident Zip  Incident Address  Bridge Highway Name  Bridge Highway Direction  Bridge Ramp  Bt Run
0  32310363  12/31/2015  01/01/2016  NYPD  New York City Police Department  Noise - Street/Sidewalk  Loud Music/Party  Street/Sidewalk  10034.0  VERMILION AVENUE  N  NaN  NaN  NaN
1  32309934  11/30/15  01/01/2016  NYPD  New York City Police Department  Blocked Driveway  No Access  Street/Sidewalk  11105.0  27-07-23 AVENUE  N  NaN  NaN  NaN
2  32309159  11/30/15  04/10/16  NYPD  New York City Police Department  Blocked Driveway  No Access  Street/Sidewalk  10458.0  2897 VALENTINE AVENUE  N  NaN  NaN  NaN
3  32305088  11/30/15  07/13/17  NYPD  New York City Police Department  Commercial Overweight Parking  Street/Sidewalk  10461.0  2940 BAELEY AVENUE  N  NaN  NaN  NaN
4  32306529  12/31/2015  03/24/16  NYPD  New York City Police Department  Illegal Parking  Blocked Driveway  Street/Sidewalk  11373.0  87-14-57 ROAD  N  NaN  NaN  NaN

5 rows x 48 columns

In [194]: ##Converting date and time columns into DO-MN-YYY format##
df['Created date']=pd.to_datetime(df['Created Date'])
df['Closed date']=pd.to_datetime(df['Closed Date'])

In [169]: #df.isac(df['Created date']==df['Closed date']) shape

Out[169]: (67408, 48)

In [51]: #display city names
df['City'].unique()

Out[51]: array(['NEW YORK', 'ASTORIA', 'BROOK', 'ELMHURST', 'BROOKLYN',
       'NEW GARDENS', 'JACKSON HEIGHTS', 'MIDDLE VILLAGE', 'REGO PARK',
       'SAINT ALBANS', 'JANICKA', 'SOUTH RICHMOND HILL', nan, 'KINGSWOOD',
       'HOWARD BEACH', 'FOREST HILLS', 'STATEN ISLAND', 'OZONE PARK',
       'RICHMOND HILL', 'WOODHAVEN', 'FLUSHING', 'CORONA',
       'QUEENS VILLAGE', 'OAKLAND GARDENS', 'HOLLIS', 'MASPETH',
       'EAST ELMHURST', 'SOUTH OZONE PARK', 'WOODSIDE', 'FRESH MEADOWS',
       'LONG ISLAND CITY', 'ROCKAWAY PARK', 'SPRINGFIELD GARDENS',
       'COLLEGE POINT', 'BAYSIDE', 'GLEN DAMS', 'FAR ROCKAWAY',
       'MELROSE', 'LITTLE NECK', 'CAMBRIA HEIGHTS', 'ROSELIA',
       'SUNNYSIDE', 'WHITESTONE', 'ARVERNE', 'FLORAL PARK',
       'NEW HYDE PARK', 'CENTRAL PARK', 'BREEZY POINT', 'QUEENS',
       'ASTORIA', 'Long Island City', 'Woodside', 'East Elmhurst',
       'Howard Beach'], dtype=object)

In [96]: #looking for a different types of complaints
df['Complaint Type'].value_counts()

Out[96]:
Blocked Driveway    108881
Illegal Parking     92679
Noise - Street/Sidewalk 51692
Noise - Commercial  44109
Derelict Vehicle    21661
Noise - Vehicle     19352
Animal Abuse        18541
Traffic             5198
Homeless Encampment 4879
Vending            4192
Noise - Park       4109
Drinking           1409
Noise - House of Worship 1070
Posting Advertisement 681
Urinating in Public 641
Bike/Roller/Skate Chronic 478
Panhandling         327
Disorderly Youth    315
Illegal Fireworks   172
Graffiti           157
Agency Issues      8
Squeepage           4
Ferry Complaint     2
Animal in a Park    1
Name: Complaint Type, dtype: int64

In [52]: #number of complaint by city wise
df.groupby(['City', 'Complaint Type']).size()

Out[52]:
City Complaint Type
ARVERNE Animal Abuse      46
         Blocked Driveway  58
         Derelict Vehicle  52
         Disorderly Youth  2
         Drinking         1
         ...
         Woodside Blocked Driveway 27
         Derelict Vehicle 8
         Illegal Parking 124
         Noise - Commercial 2
         Noise - Street/Sidewalk 5
Length: 777, dtype: int64

In [53]:
def to_title(city):
    try:
        city = city.title()
        return city
    except:
        return np.nan

df['City'] = df['City'].apply(to_title)
df['City'].value_counts()

Out[53]:
Brooklyn    118862
New York    77512
Bronx       49172
Staten Island 15348
Jamaica     8852
Astoria     7487
Flushing    6392
Corona      5383
Woodside   4523
East Elmhurst 3588
Ozone Park  3448
Elmhurst    3438
South Richmond Hill 3431
Long Island City 3198
Maspeth     3118
Woodhaven  3103
South Ozone Park 2868
Fresh Meadows 2453
Richmond Hill 2335
Middle Village 2251
Forest Hills 2122
Jackson Heights 2106
Rego Park   1807
Bayside     1558
College Point 1544
Far Rockaway 1369
Whitestone  1369
Hollis      1145
Howard Beach 1094
Springfield Gardens 1091
Saint Albans 847
Kew Gardens 808
Sunnyside   844
Rockaway Park 831
Oakland Gardens 717
Little Neck 712
Cambria Heights 617
Belrose     487
Glen Oaks  408
Arverne     259
Floral Park 119
New Hyde Park 37
Central Park 3
Queens      1
Breezy Point 1
Name: City, dtype: int64

In [99]: #We frequency plot for city wise complaints in count
plt.figure(figsize=(12,10))
plt.title('Citywise total complaints')
sns.countplot('City', data=df)
plt.show()

In [100]: #Scatter plot for complaint concentration across Brooklyn
df_bkn = df.loc[df['City'] == 'Brooklyn']## here, we are taking the location of Brooklyn via 'loc'method and copying it to df_bkn and plotting ScatterPlot on it
df_bkn[['Longitude', 'Latitude']].plot(kind = 'scatter', x='Longitude', y='Latitude', title = 'Complaints Concentration in Brooklyn', figsize = (18, 8));

In [101]: #A heatmap plot for complaint concentration across Brooklyn##
df_bkn[['Longitude', 'Latitude']].plot(kind = 'heatmap', x='Longitude', y='Latitude', gridsize=(40,colormap='jet',vmin=0,title='Complaints Concentration in Brooklyn',figsize=(10,8))

Out[101]:
<AxesSubplot:title='center': 'Complaints Concentration in Brooklyn', xlabel='Longitude', ylabel='Latitude'>

In [54]: #Accessing the 'complaint type',through 'loc'method
high_complaints=df.loc[:, 'Complaint Type']
high_complaints.value_counts()

Out[54]:
0 Noise - Street/Sidewalk    108881
1 Blocked Driveway          92679
2 Blocked Driveway          51692
3 Illegal Parking           44109
4 Illegal Parking           21661
...
36453 Illegal Parking        18541
36454 Noise - Vehicle        5198
36455 Noise - Street/Sidewalk 4879
36456 Blocked Driveway       4192
36457 Blocked Driveway       4109
Name: Complaint Type, Length: 36458, dtype: object

In [55]: high_complaints.nunique()

Out[55]: 24

In [56]: high_complaints.value_counts()
head

Out[56]:
Blocked Driveway    108881
Illegal Parking     92679
Noise - Street/Sidewalk 51692
Noise - Commercial  44109
Derelict Vehicle    21661
Noise - Vehicle     19352
Animal Abuse        18541
Traffic             5198
Homeless Encampment 4879
Vending            4192
Noise - Park       4109
Drinking           1409
Noise - House of Worship 1070
Posting Advertisement 681
Urinating in Public 641
Bike/Roller/Skate Chronic 478
Panhandling         327
Disorderly Youth    315
Illegal Fireworks   172
Graffiti           157
Agency Issues      8
Squeepage           4
Ferry Complaint     2
Animal in a Park    1
Name: Complaint Type, dtype: int64

In [57]: tail-high_complaints.value_counts()
tail

Out[57]:
Blocked Driveway    108881
Illegal Parking     92679
Noise - Street/Sidewalk 51692
Noise - Commercial  44109
Derelict Vehicle    21661
Noise - Vehicle     19352
Animal Abuse        18541
Traffic             5198
Homeless Encampment 4879
Vending            4192
Noise - Park       4109
Drinking           1409
Noise - House of Worship 1070
Posting Advertisement 681
Urinating in Public 641
Bike/Roller/Skate Chronic 478
Panhandling         327
Disorderly Youth    315
Illegal Fireworks   172
Graffiti           157
Agency Issues      8
Squeepage           4
Ferry Complaint     2
Animal in a Park    1
Name: Complaint Type, dtype: int64

In [106]: high_complaints.value_counts().plot(kind='bar',title='count vs. complaint types')

Out[106]:
<AxesSubplot:title='center': 'count vs. complaint types'>

In [107]: #visualizing the frequency of top 10 complaint types
df['Complaint Type'].value_counts().head(10).plot(kind = 'bar', title = 'Overall Complaint Type Frequency')

Out[107]:
<AxesSubplot:title='center': 'Overall Complaint Type Frequency'>

In [58]: df1=df.groupby(['City', 'Complaint Type']).size()
df2=df1.reset_index()

In [59]: #displaying the 'city'and 'types of complaints' in a separate dataset
df2

Out[59]:
   City  Complaint Type  0
Arverne Blocked Driveway 46
         Blocked Driveway 50
         Derelict Vehicle 32
         Disorderly Youth 2
         Drinking         1
         ...
         Noise - Street/Sidewalk 256
         Noise - Vehicle 45
         Unraining in Public 8
         Vending         15
759 rows x 3 columns

In [68]: df2.value_counts().nunique()

Out[68]: 16
```