

# Moleboheng Wendy Mokoena

Jupyter

Used 20.4 of 50 hours in May, 2023

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

Customer Service Requests Analysis.  
Course-end Project 1

DESCRIPTION

You've been asked to perform data analysis of service request (311) calls from New York City. You've also been asked to utilize data wrangling techniques to understand the pattern in the data and visualize the major types of complaints.

Note: Download 311-service-requests-nyc.zip file using the link given in the Customer Service Requests Analysis project problem statement and extract the 311\_Service\_Requests\_from\_2010\_to\_Present.csv file

```
[1]: import pandas as pd
import numpy as np
import scipy.stats as stats
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import warnings

[2]: pd.set_option('display.max_columns',30)
pd.set_option('display.max_rows',800)
warnings.simplefilter('ignore')

[3]: data = pd.read_csv('customer service/311_Service_Requests_from_2010_to_Present.csv')
data
```

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
0	32310363	12/31/2015 11:59:45 PM	12/31/2015 12:55:15 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	ACADUWAY STREET	WEST 204 STREET

Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb

Used 20.4 of 50 hours in May, 2023

Jupyter

Used 20.4 of 50 hours in May, 2023

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

```
[3]: data = pd.read_csv('customer service/311_Service_Requests_from_2010_to_Present.csv')
data
```

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
0	32310363	12/31/2015 11:59:45 PM	12/31/2015 12:55:15 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	ACADUWAY STREET	WEST 204 STREET
1	32309934	12/31/2015 11:59:44 PM	01/01/2016 01:26:57 AM	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	23 AVENUE	27 STREET 28 STREET
2	32309159	12/31/2015 11:59:29 PM	04/31/2016 04:51:03 AM	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	EAST 190 STREET	EAST 199 STREET
3	32305096	12/31/2015 11:57:46 PM	07/43/13 AM	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAYSLEY AVENUE	BAYSLEY AVENUE	EDISON AVENUE 8 STREET
4	32306529	12/31/2015 11:56:58 PM	03/24/42 AM	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	67-14 57 ROAD	57 ROAD	SEABURY DRIVE HOFFMAN DRIVE
191402	31024399	07/07/2015 08:47:21 PM	07/08/2015 12:32:03 AM	NYPD	New York City Police Department	Noise - Commercial	Banging/Pounding	Store/Commercial	11207.0	258 JAMAICA AVENUE	JAMAICA AVENUE	VAN SICKLEN COURT HENDRIX STREET
191403	31025431	07/07/2015 08:43:41 PM	11/13/32 PM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10009.0	134 AVENUE D	EAST 9 STREET	EAST 10 STREET
191404	31023497	07/07/2015 08:42:44 PM	02/59/07 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10473.0	NaN	NaN	NaN

Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb

Used 20.4 of 50 hours in May, 2023

Jupyter

Used 20.4 of 50 hours in May, 2023

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

```
[4]: data.head()
```

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
0	32310363	12/31/2015 11:59:45 PM	12/31/2015 12:55:15 AM	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034.0	71 VERMILYEA AVENUE	ACADUWAY STREET	WEST 204 STREET	NaN
1	32309934	12/31/2015 11:59:44 PM	01/01/2016 01:26:57 AM	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	11105.0	27-07 23 AVENUE	23 AVENUE	27 STREET 28 STREET	NaN
2	32309159	12/31/2015 11:59:29 PM	04/31/2016 04:51:03 AM	NYPD	New York City Police Department	Blocked Driveway	No Access	Street/Sidewalk	10458.0	2897 VALENTINE AVENUE	EAST 190 STREET	EAST 199 STREET	NaN
3	32305096	12/31/2015 11:57:46 PM	07/43/13 AM	NYPD	New York City Police Department	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461.0	2940 BAYSLEY AVENUE	BAYSLEY AVENUE	EDISON AVENUE 8 STREET	NaN
4	32306529	12/31/2015 11:56:58 PM	03/24/42 AM	NYPD	New York City Police Department	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373.0	67-14 57 ROAD	57 ROAD	SEABURY DRIVE HOFFMAN DRIVE	NaN

5 rows x 53 columns

```
[5]: data.columns
```

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)', 'Block-Facility-Name', 'Block-Borough'])

Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb

Used 20.4 of 50 hours in May, 2023

# Moleboheng Wendy Mokoena

JupyterLab interface showing a file explorer on the left and a code editor on the right. The code editor displays the following Python code and its output:

```
[6]: data.shape
[6]: (191407, 53)
[7]: data[['Complaint Type', 'City']]
[7]:
```

Complaint Type	City
0 Noise - Street/Sidewalk	NEW YORK
1 Blocked Driveway	ASTORIA
2 Blocked Driveway	BRONX
3 Illegal Parking	BRONX
4 Illegal Parking	ELMHURST
...	...
191402 Noise - Commercial	BROOKLYN
191403 Noise - Street/Sidewalk	NEW YORK
191404 Noise - Street/Sidewalk	BRONX
191405 Blocked Driveway	BROOKLYN
191406 Noise - Street/Sidewalk	NEW YORK

```
[8]: data.isnull()
[8]:
```

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	School Not Found	School City Compl
0	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
1	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
2	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
3	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
4	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
191402	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191403	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191404	False	False	False	False	False	False	False	False	False	True	True	True	True	True	...	False
191405	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191406	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	True

191407 rows x 2 columns

JupyterLab interface showing a file explorer on the left and a code editor on the right. The code editor displays the following Python code and its output:

```
[8]: data.isnull()
[8]:
```

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	School Not Found	School City Compl
0	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
1	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
2	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
3	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
4	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
191402	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191403	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191404	False	False	False	False	False	False	False	False	False	True	True	True	True	True	...	False
191405	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191406	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	True

191407 rows x 53 columns

```
[9]: data.info()
[9]:
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 191407 entries, 0 to 191406
Data columns (total 53 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   Unique Key          191407 non-null  int64
1   Created Date         191407 non-null  object
2   Closed Date          189985 non-null  object
3   ...                  ...              ...
```

JupyterLab interface showing a file explorer on the left and a code editor on the right. The code editor displays the following Python code and its output:

```
[8]: data.isnull()
[8]:
```

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	School Not Found	School City Compl
0	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
1	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
2	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
3	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
4	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
191402	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191403	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191404	False	False	False	False	False	False	False	False	False	True	True	True	True	True	...	False
191405	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	False
191406	False	False	False	False	False	False	False	False	False	False	False	False	True	True	...	True

191407 rows x 53 columns

```
[9]: data.info()
[9]:
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 191407 entries, 0 to 191406
Data columns (total 53 columns):
#   Column              Non-Null Count  Dtype
---  ---
0   Unique Key          191407 non-null  int64
1   Created Date         191407 non-null  object
2   Closed Date          189985 non-null  object
3   ...                  ...              ...
```

# Moleboheng Wendy Mokoena

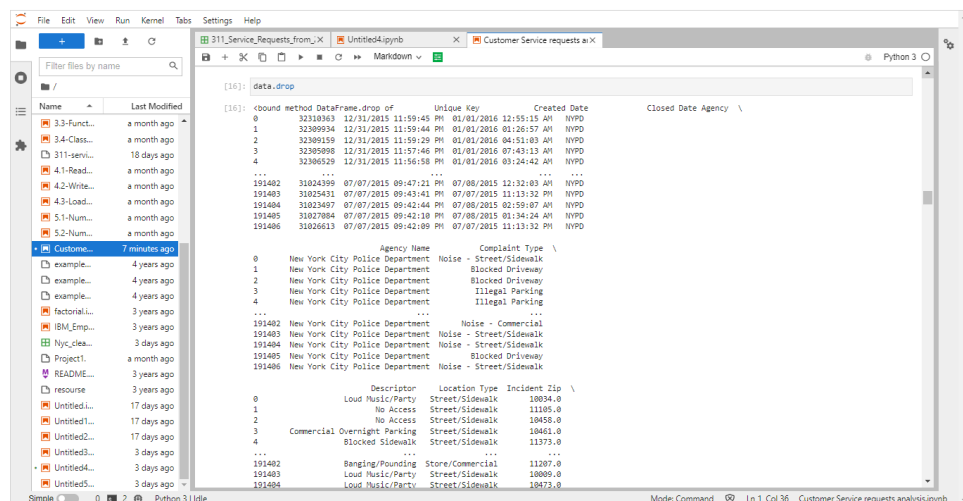
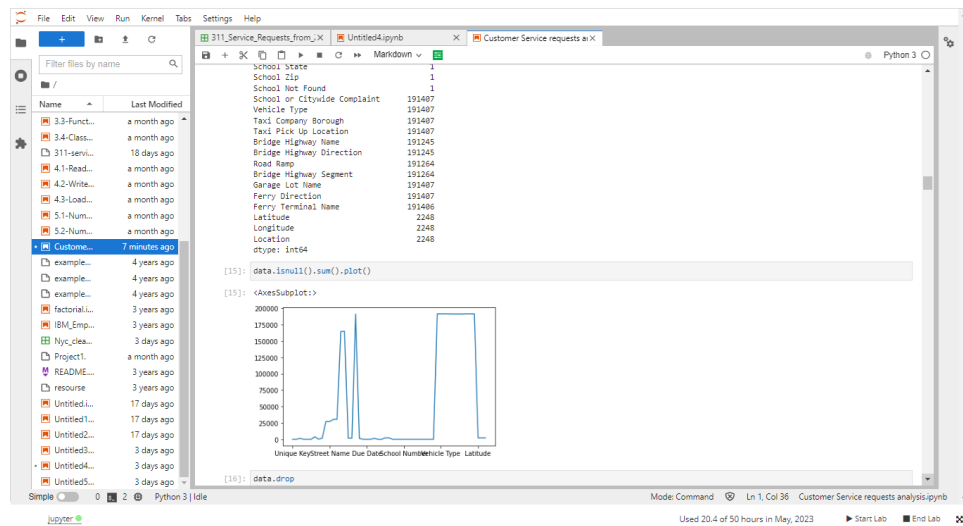
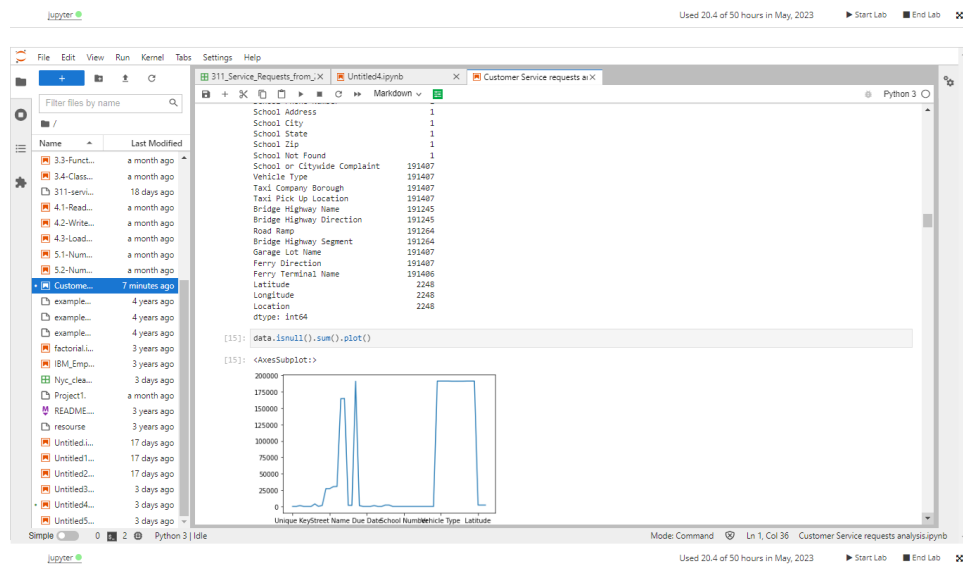
The image displays three sequential screenshots of a Jupyter Notebook interface, showing the analysis of 311 service requests data. The notebook is titled "Customer Service requests analysis.ipynb" and is running on a Python 3 kernel.

**First Screenshot:** The notebook shows the initial data loading and inspection. The code cell [9] displays the output of `data.info()`, indicating that the data is a pandas DataFrame with 191407 entries and 53 columns. The output lists the columns and their data types, including 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 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, Bridge Highway Segment, Garage Lot Name, Ferry Direction, Ferry Terminal Name, Latitude, Longitude, and Location.

**Second Screenshot:** The notebook shows the next step in the analysis, where the data is filtered to identify duplicates. The code cell [13] displays the output of `data.duplicated().sum()`, which returns 0, indicating that there are no duplicate rows in the data. The code cell [14] displays the output of `data.isna().sum()`, which returns 0, indicating that there are no missing values in the data.

**Third Screenshot:** The notebook shows the final step in the analysis, where the data is summarized. The code cell [14] displays the output of `data.isna().sum()`, which returns 0, indicating that there are no missing values in the data. The code cell [15] displays the output of `data.duplicated().sum()`, which returns 0, indicating that there are no duplicate rows in the data. The code cell [16] displays the output of `data.info()`, which shows the data is a pandas DataFrame with 191407 entries and 53 columns. The output lists the columns and their data types, including 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 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, Bridge Highway Segment, Garage Lot Name, Ferry Direction, Ferry Terminal Name, Latitude, Longitude, and Location.

# Moleboheng Wendy Mokoena



# Moleboheng Wendy Mokoena

JupyterLab interface showing a Jupyter Notebook with a file explorer on the left and a code editor on the right. The notebook is titled "Customer Service requests analysis.ipynb" and is running Python 3.

The code in the notebook is as follows:

```
[17]: data.dropna(subset=['Closed Date'])
```

Unique Key	Created Date	Closed Date	Agency	Complaint Type	Descriptor	Location Type	Incident Zip	Incident Address	Street Name	Cross Street 1	Cross Street 2
0	12/31/2015 11:59:45 PM	12/31/2015 12:55:15 AM	NYPD	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10034-0	VERMILYEA AVENUE	VERMILYEA AVENUE	ACADEMY STREET	WEST 204 STREET
1	12/31/2015 11:59:44 PM	01/26/2016 AM	NYPD	Blocked Driveway	No Access	Street/Sidewalk	11105-0	27-07 23 AVENUE	23 AVENUE	27 STREET	28 STREET
2	12/31/2015 11:59:29 PM	04/51/2016 AM	NYPD	Blocked Driveway	No Access	Street/Sidewalk	10458-0	VALENTINE AVENUE	VALENTINE AVENUE	EAST 198 STREET	EAST 199 STREET
3	12/31/2015 11:57:46 PM	07-42/2015 AM	NYPD	Illegal Parking	Commercial Overnight Parking	Street/Sidewalk	10461-0	2940 BALSLEY AVENUE	BALSLEY AVENUE	EDISON AVENUE	8 STREET
4	12/31/2015 11:56:58 PM	03/24/2016 AM	NYPD	Illegal Parking	Blocked Sidewalk	Street/Sidewalk	11373-0	87-14 57 ROAD	57 ROAD	SEABURY STREET	HOFFMAN DRIVE
191402	07/07/2015 09:47:21 PM	12/32/2015 AM	NYPD	Noise - Commercial	Banging/Pounding	Store/Commercial	11207-0	258 JAMAICA AVENUE	JAMAICA AVENUE	VAN SICLEN COURT	HENDRIX STREET
191403	07/07/2015 08:43:41 PM	11/13/2015 PM	NYPD	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10009-0	134 AVENUE D	AVENUE D	EAST 9 STREET	EAST 10 STREET
191404	07/07/2015 08:42:44 PM	02/29/2016 AM	NYPD	Noise - Street/Sidewalk	Loud Music/Party	Street/Sidewalk	10473-0	NaN	NaN	NaN	NaN
191405	07/07/2015 08:42:10 PM	01/34/2016 AM	NYPD	Blocked Driveway	No Access	Street/Sidewalk	11248-0	376 WYTHE AVENUE	WYTHE AVENUE	SOUTH 3 STREET	SOUTH 4 STREET

```
[18]: import time
[19]: Closed_Time = time.time()
Creation_Date = time.time()
elapsed_time = Closed_Time - Creation_Date
print(elapsed_time)
-3.1948889599689375e-05
[20]: from datetime import datetime
seconds = -3.218659817870946e-05
data = datetime.fromtimestamp(seconds).strftime('%Y-%m-%d %H:%M:%S.%f')
print('seconds is:', data)
seconds is: 1969-12-31 23:59:59.999968
[21]: import numpy as np
import pandas as pd
data = pd.read_csv(r'customer service/311_Service_Requests_from_2010_to_Present.csv')
import datetime as datetime
data.describe()
```

	Unique Key	Incident Zip	X Coordinate (State Plane)	Y Coordinate (State Plane)	School or Citywide Complaint	Vehicle Type	Taxi Company Borough	Taxi Pick Up Location	Garage Lot Name	Ferry Direction	Latitude	Longitude
count	1.914070e+05	18972.000000	1.891590e+05	1.891590e+05	0.0	0.0	0.0	0.0	0.0	0.0	189159.000000	189159.000000
mean	3.166035e+07	10855.280398	1.004803e+06	203402.653628	NaN	NaN	NaN	NaN	NaN	NaN	40.724918	-73.825812
std	3.692353e+05	574.951853	2.195427e+04	29962.071501	NaN	NaN	NaN	NaN	NaN	NaN	0.082237	0.079177
min	3.101905e+07	83.000000	9.133570e+05	121411.000000	NaN	NaN	NaN	NaN	NaN	NaN	40.499673	-74.254937
25%	3.134816e+07	10312.000000	9.919740e+05	182791.500000	NaN	NaN	NaN	NaN	NaN	NaN	40.666300	-73.972159

Moleboheng Wendy Mokoena

The screenshot displays a Jupyter Notebook environment with a file explorer on the left, a top menu bar, and a status bar at the bottom. The main area shows a pandas DataFrame with the following columns: 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, School Not Found, and School or Citywide Complaint. The data is filtered by 'City' and shows 53 columns and 0 rows. The interface includes a file explorer on the left, a top menu bar, and a status bar at the bottom.

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	School Not Found	School or Citywide Complaint
33	12/31/2015 11:18:10 PM	01/02/2016 01:04:03 AM	NYPD	New York City Police Department	Illegal Parking	Double Parking Blocking Traffic	Street/Sidewalk	NaN	23-0-23-54 12TH STREET	12TH STREET	NaN	NaN	NaN	NaN	NaN	NaN
283	12/31/2015 05:40:16 PM	01/01/2016 10:59:11 AM	NYPD	New York City Police Department	Illegal Parking	Blocked Hydrant	Street/Sidewalk	NaN	72-23-72-2 72ND PLACE	72ND PLACE	NaN	NaN	NaN	NaN	NaN	NaN
302	12/31/2015 06:09:00 AM	01/01/2016 08:17:35 PM	NYPD	New York City Police Department	Blocked Driveway	Partial Access	Street/Sidewalk	NaN	9-15-9-99 CORNAGA AVENUE	CORNAGA AVENUE	NaN	NaN	NaN	NaN	NaN	NaN
416	12/31/2015 02:16:04 PM	NaN	NYPD	New York City Police Department	Illegal Parking	Posted Parking Sign Violation	Street/Sidewalk	NaN	5426-5526 90TH ST	90TH ST	NaN	NaN	NaN	NaN	NaN	NaN
611	12/31/2015 09:50:06 AM	NaN	NYPD	New York City Police Department	Noise - Street/Sidewalk	Loose Music/Loud	Street/Sidewalk	NaN	30 STREET	30 STREET	30 AVENUE	NaN	NaN	NaN	NaN	NaN

The screenshot shows the JupyterLab interface. On the left is a file explorer with a sidebar showing a list of files and folders. The main area is a code editor displaying a Jupyter Notebook cell. The code in the cell is as follows:

```
[1]: data['City'].isna().mode()

[88]: 0 False
dtype: bool

[86]: data['City'] = data['City'].fillna(data['City'].mode())

[89]: data['City'].isna().mode()

[89]: 0 False
dtype: bool

[3]: import numpy as np
import pandas as pd

data = pd.read_csv('customer service/311_Service_Requests_from_2010_to_Present.csv')
data
print(data.fillna('unknown_City'))

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

interactivity=Interactivity, Created Date Closed Date Agency \
0 32330363 12/31/2015 11:59:45 PM 01/01/2016 12:55:15 AM NYPD
1 32309594 12/31/2015 11:59:44 PM 01/01/2016 01:20:57 AM NYPD
2 32380159 12/31/2015 11:59:29 PM 01/01/2016 04:51:03 AM NYPD
3 32350808 12/31/2015 11:57:46 PM 01/01/2016 07:43:13 AM NYPD
4 32380529 12/31/2015 11:56:58 PM 01/01/2016 03:24:42 AM NYPD
...
191482 31024399 07/07/2015 09:47:21 PM 07/08/2015 12:32:03 AM NYPD
191483 31025431 07/07/2015 09:43:41 PM 07/07/2015 11:13:32 PM NYPD
191484 31024597 07/07/2015 09:42:44 PM 07/08/2015 02:59:07 AM NYPD
191485 31027864 07/07/2015 09:42:10 PM 07/08/2015 01:34:24 AM NYPD
191486 31026613 07/07/2015 09:42:09 PM 07/07/2015 11:13:32 PM NYPD

Agency Name Complaint Type \
0 311 Service Request 311 Service Request
1 311 Service Request 311 Service Request
2 311 Service Request 311 Service Request
3 311 Service Request 311 Service Request
4 311 Service Request 311 Service Request
...
191482 311 Service Request 311 Service Request
191483 311 Service Request 311 Service Request
191484 311 Service Request 311 Service Request
191485 311 Service Request 311 Service Request
191486 311 Service Request 311 Service Request
```

The status bar at the bottom indicates 'Ln 1, Col 36' and 'Customer Service requests analysis.ipynb'.

# Moleboheng Wendy Mokoena

Used 20.4 of 50 hours in May, 2023 ▶ Start Lab ■ End Lab ✕

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

```
191486 464 EAST 10 STREET ... unknown_City
```

	Bridge Highway Direction	Road Ramp Bridge Highway Segment \	
0	unknown_City	unknown_City	unknown_City
1	unknown_City	unknown_City	unknown_City
2	unknown_City	unknown_City	unknown_City
3	unknown_City	unknown_City	unknown_City
4	unknown_City	unknown_City	unknown_City
...	...	...	...
191482	unknown_City	unknown_City	unknown_City
191483	unknown_City	unknown_City	unknown_City
191484	unknown_City	unknown_City	unknown_City
191485	unknown_City	unknown_City	unknown_City
191486	unknown_City	unknown_City	unknown_City

	Garage Lot Name Ferry Direction Ferry Terminal Name	Latitude \
0	unknown_City	unknown_City
1	unknown_City	unknown_City
2	unknown_City	unknown_City
3	unknown_City	unknown_City
4	unknown_City	unknown_City
...	...	...
191482	unknown_City	unknown_City
191483	unknown_City	unknown_City
191484	unknown_City	unknown_City
191485	unknown_City	unknown_City
191486	unknown_City	unknown_City

	Longitude	Location
0	-73.9235	(40.86568153631767, -73.92358095371746)
1	-73.9151	(40.775945312321885, -73.915093938986685)
2	-73.8885	(40.870324522111424, -73.88852464418646)
3	-73.8284	(40.8350444683083, -73.82837939540286)
4	-73.8742	(40.733859618956813, -73.87416975818375)
...	...	...
191482	-73.8918	(40.68061718662283, -73.8917426953243)
191483	-73.8755	(40.72444683419667, -73.8754633228889)
191484	-73.8582	(40.81329732471253, -73.85821295914048)
191485	-73.9656	(40.713287275795135, -73.96564173122595)

Simple 0 2 Python 3 Idle Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb

Used 20.4 of 50 hours in May, 2023 ▶ Start Lab ■ End Lab ✕

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

```
[191487 rows x 53 columns]
```

```
[87]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[30]: plt.title('Complaint Types Distribution')
sns.countplot(y='Complaint Type', data=data)
plt.show()
```

```
[31]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data = pd.read_csv('customer service/311_Service_Requests_from_2010_to_Present.csv')
data
```

```
data[['Longitude', 'Latitude']].plot(kind='scatter', x='Longitude', y='Latitude', title='Complaints Concentration in Brooklyn', figsize=(10, 10))
```

Simple 0 2 Python 3 Idle Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb

Used 20.4 of 50 hours in May, 2023 ▶ Start Lab ■ End Lab ✕

File Edit View Run Kernel Tabs Settings Help

Filter files by name

311\_Service\_Requests\_from\_X Untitled4.ipynb Customer Service requests a/x

Python 3

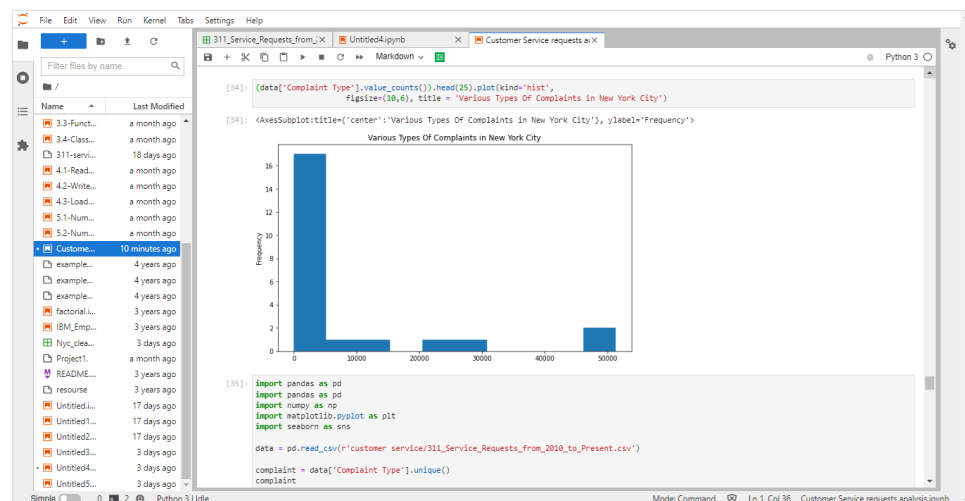
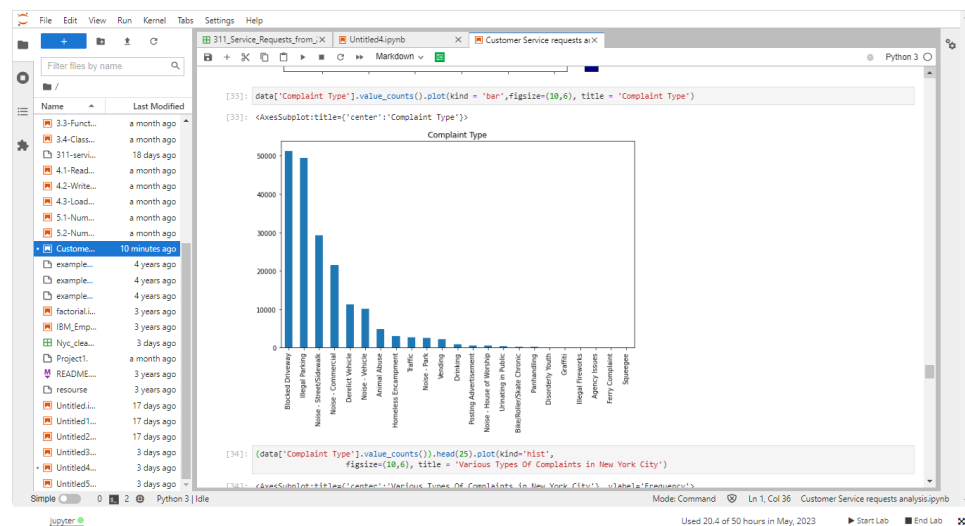
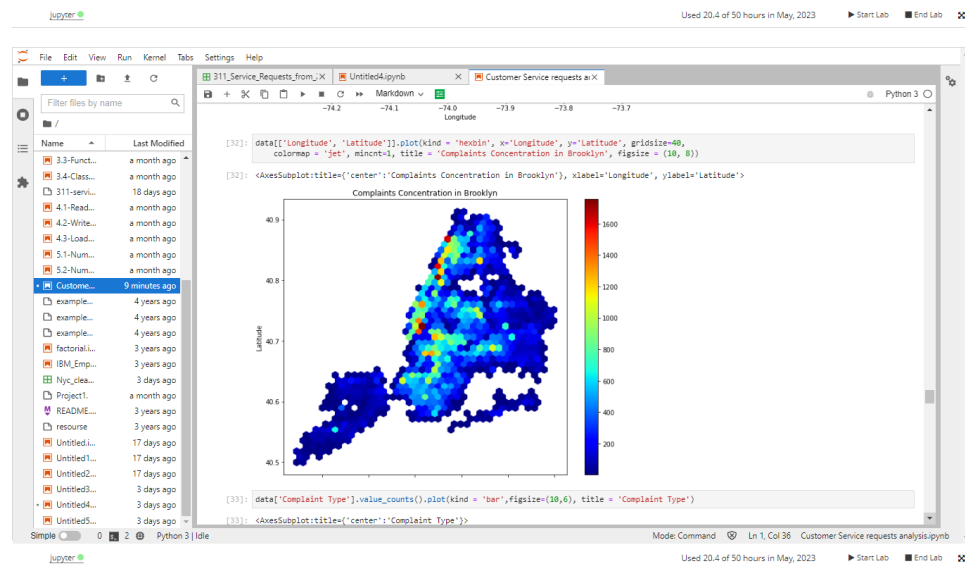
```
[31]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data = pd.read_csv('customer service/311_Service_Requests_from_2010_to_Present.csv')
data
```

```
data[['Longitude', 'Latitude']].plot(kind='scatter', x='Longitude', y='Latitude', title='Complaints Concentration in Brooklyn', figsize=(10, 10))
```

```
[31]: (axesSubplot,title='center':'Complaints Concentration in Brooklyn', xlabel='Longitude', ylabel='Latitude')
```

Simple 0 2 Python 3 Idle Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb





# Moleboheng Wendy Mokoena

JupyterLab interface showing the initial data loading and preprocessing steps for the 311 Service Requests dataset.

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

data = pd.read_csv(r'customer service/311_Service_Requests_from_2010_to_Present.csv')
complaint = data['Complaint Type'].unique()
complaint

complaint[10]
```

```
[35]: array(['Noise - Street/Sidewalk', 'Blocked Driveway', 'Illegal Parking',
        'Derelict Vehicle', 'Noise - Commercial',
        'Noise - House of Worship', 'Posting Advertisement',
        'Noise - Vehicle', 'Animal Abuse', 'Vending'], dtype=object)
```

```
[36]: import pandas as pd
import numpy as np

data = pd.read_csv(r'customer service/311_Service_Requests_from_2010_to_Present.csv')
a = pd.DataFrame(data)
a['Complaint Type', 'City']
```

```
[37]:
```

Complaint Type	City
0 Noise - Street/Sidewalk	NEW YORK
1 Blocked Driveway	ASTORIA
2 Blocked Driveway	BROOKLYN
3 Illegal Parking	BROOKLYN
4 Illegal Parking	ELMHURST
...	...
191402 Noise - Commercial	BROOKLYN

JupyterLab interface showing the creation of a pivot table to analyze complaint types across different cities.

```
[37]: df_new = pd.crosstab(data['Complaint Type'], data['City'])
df_new
```

```
[37]:
```

Complaint Type	ARVERNE	ASTORIA	ASTORIA	BAYSIDE	BELLEROSSE	BREEZY POINT	BROOKLYN	CAMBRIA HEIGHTS	CENTRAL PARK	COLLEGE POINT	CORONA	EAST ELMHURST	ELMHURST	Elm
Animal Abuse	26	75	0	24	6	1	901	1465	7	0	21	33	41	23
Bike/Roller/Skate Chronic	0	14	0	0	1	0	12	69	0	0	0	0	0	1
Blocked Driveway	26	1803	60	240	60	3	8396	18612	99	0	287	1880	935	956
Derelict Vehicle	11	234	9	128	51	0	1272	3369	69	0	95	35	78	61
Disorderly Youth	1	2	0	1	1	0	27	40	0	0	0	2	0	0
Drinking	0	24	0	0	0	0	120	154	0	0	0	22	4	4
Graffiti	0	0	0	3	0	0	7	24	0	0	1	0	2	0
Homeless Encampment	3	24	0	2	1	0	148	595	2	0	1	14	1	26
Illegal Fireworks	0	1	0	0	0	0	8	13	0	0	0	0	0	1
Illegal Parking	40	709	111	315	71	13	5189	17919	53	0	217	452	616	419
Noise - Commercial	2	695	100	31	9	3	1636	6777	7	0	25	154	27	45
Noise - House of Worship	3	6	0	1	0	0	50	231	0	0	0	3	11	2
Noise - Park	0	43	0	3	1	0	337	1008	0	0	1	17	1	25
Noise - Street/Sidewalk	14	242	53	5	8	1	5320	8099	12	50	25	133	62	161

JupyterLab interface showing the visualization of complaint types using a pie chart.

```
[38]: (data['Complaint Type'].value_counts().head(25).plot(kind='pie',
        figsize=(10,6), title = 'Types Of Complaints in Each City'))
```

```
[38]: <AxesSubplot:title='center': 'Types Of Complaints in Each City', ylabel='Complaint Type'>
```

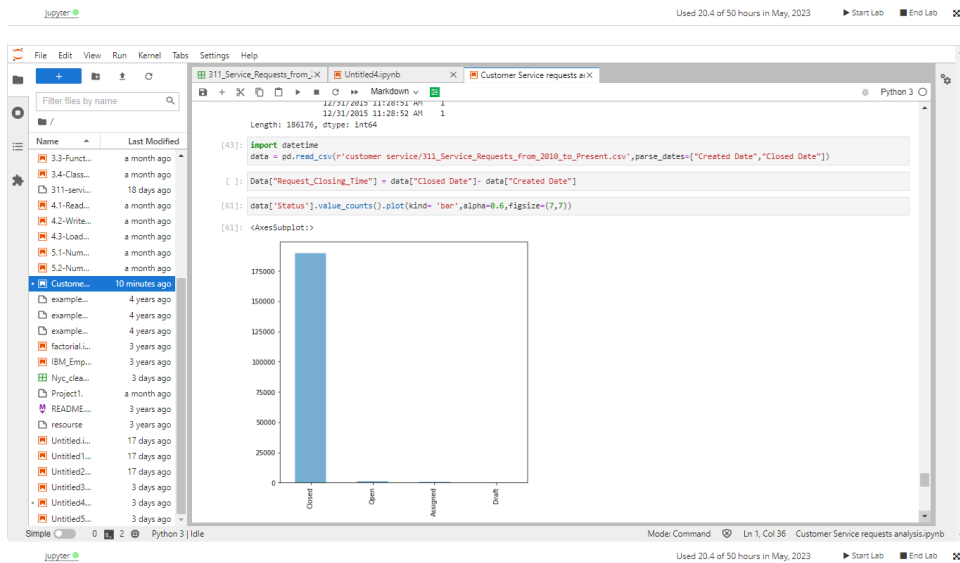
```
[40]: import numpy as np
import pandas as pd

data = pd.read_csv(r'customer service/311_Service_Requests_from_2010_to_Present.csv')
data.groupby(['Complaint Type', 'Closed Date']).size()
```

```
[40]:
```

Complaint Type	Closed Date	Size
Agency Issues	07/11/2015 04:43:58 PM	1
Animal Abuse	07/13/2015 12:33:00 AM	1
Animal Abuse	07/08/2015 01:04:04 PM	1
Animal Abuse	07/08/2015 01:12:46 PM	1

# Moleboheng Wendy Mokoena



Jupyter

Used 20.4 of 50 hours in May, 2023 ▶ Start Lab ■ End Lab ✕

File Edit View Run Kernel Tabs Settings Help

311\_Service\_Requests\_from\_...X Untitled4.ipynb Customer Service requests a1X

Python 3

```
[4]: from numpy.random import seed
from numpy.random import randn
from scipy.stats import wilcoxon
# contingency table
seed(1)

#generate two independent samples
data1 = 5 * randn(100) + 50
data2 = 5 * randn(100) + 51

#interpret test-statistic
stat, p = wilcoxon(data1, data2)
print('statistics=%.3f, p=%.3f' % (stat, p))

#interpret p values
alpha = 0.05
if p > alpha:
    print('Same distribution (fail to reject H0)')
else:
    print('different distribution (reject H0)')

statistics=1886.000, p=0.028
different distribution (reject H0)
```

Based on all the analyses done, the NYC311 Service is a very well-liked and dependable channel and resource for the NYC neighborhoods to inform the neighborhood organizations and citizen service providers about a variety of issues that are crucial to the society's health and well-being. I was able to pinpoint general trends and issues affecting the major boroughs of the New York Metropolitan Area, but even more significantly, I was able to identify distinctive themes and patterns that were more prevalent in each one. This gave me a better understanding of the unique needs, challenges, and local dynamics that each borough community faces on a daily basis. Surprisingly, the majority of the "nyc311" tweets' sentiment analysis results show that they are favorable or neutral (67%), which is a relatively small percentage.

Simple 0 2 Python 3 Idle Mode Command Ln 1, Col 36 Customer Service requests analysis.ipynb