# Deliverable 2

## Introduction

This document outlines the plan for analyzing the NYC311 data to answer two primary questions:

1. **How did Hurricane Sandy impact New York City base on NYC311 data?**
2. **What was the impact of COVID-19 on New York City based on NYC311 data?**

The analysis involves summarizing and visualizing the data to identify patterns and trends related to these events.

## Plan for Data Analysis

The analysis will include the following steps:

**Data Preprocessing:** Convert date columns to DateTime format, handle missing values, and filter data based on the specified time frames.

**Data Summarization:** Summarize the total number of service requests and categorize the types of service requests before and after each event.

**Data Visualization:** Plot the number of service requests over time, create bar charts for common complaint types, and map the geographical distribution of requests.

**Specific Analyses:**

Hurricane Sandy: Compare service requests before and after the event to see how they have increased/decreased, identify the common complaint types and compare how they have changes before/after the event, map affected areas, and identify agencies with the highest requests.

COVID-19: Compare service requests before and after the event to see how they have increased/decreased, identify common complaint types before/after the event, map affected areas, and identify agencies with the highest requests. Analyze which areas (Incident Zip) were receiving the most complaints before COVID and then after to see which areas were most affected by COVID

****The codes on the next page might have errors that we will address as we progress during the project. These are only sample codes of how we will be extracting data but we will be analyzing other variables within the dataset as mentioned above under "Specific Analyses"****

**Sample Code for the Analysis**

```
import pandas as pd

import datetime

import matplotlib.pyplot as plt

# Reads the CSV File

data_311 = pd.read_csv('311data.csv', index_col='id')

data_covid = pd.read_csv('311_COVID.csv', index_col='Unique Key', low_memory=False)

# Convert date columns to datetime

data_311["created_date"] = pd.to_datetime(data_311["created_date"])

data_covid["Created Date"] = pd.to_datetime(data_covid["Created Date"])

# Filter data for Hurricane Sandy period

preHurricaneData = data_311.loc[(data_311["created_date"]< "2012-10-29") &
(data_311["created_date"]> "2012-09-29") ]

postHurricaneData = data_311.loc[(data_311["created_date"]< "2012-11-29") &
(data_311["created_date"]> "2012-10-29") ]

preHurricaneData.head()

postHurricaneData.head()

# The below code will be ran for different variables

preHurricaneData['city'].describe()

postHurricaneData['city'].describe()

 # Filter data for COVID-19 period

preCovidData = data_covid.loc[(data_covid["Created Date"]< "2020-03-22") &
(data_covid["Created Date"]> "2020-01-01") ]

postCovidData = data_covid.loc[(data_covid["Created Date"]< "2020-06-01") &
(data_covid["Created Date"]> "2020-03-22") ]

 # Summarize and visualize data

    def summarize_data(df, period):

        if df.empty:
```

```
        print(f"No data available for {period}.")
        return
    print(f"Summary for {period}:")
    print(df['complaint_type'].value_counts().head(10))
    df['complaint_type'].value_counts().head(10).plot(kind='bar')
    plt.title(f"Top 10 Complaint Types - {period}")
    plt.show()
# Summarize and visualize Hurricane Sandy data
summarize_data(preHurricaneData, 'Before Hurricane Sandy')
summarize_data(postHurricaneData, 'After Hurricane Sandy')

# Summarize and visualize COVID-19 data
summarize_data(preCovidData, 'Before COVID-19 Lockdown')
summarize_data(postCovidData, 'After COVID-19 Lockdown')
```