Graffiti and City Sanitation Services: A Correlative Analysis

Data Warehousing and Analytics

City University of New York, Baruch College

December 2024

Introduction

This project aims to investigate the correlation between graffiti prevalence and the quality of city sanitation services across various community districts in New York City. The goal of this exploration is to investigate weather rates of graffiti correlates with sanitation services in each neighbourhood, which may indicate a lack of maintenance, resulting in infrequent or ineffective defaced property remediation. This then points towards larger social issues such as underfunding of public services or demographic discrimination.

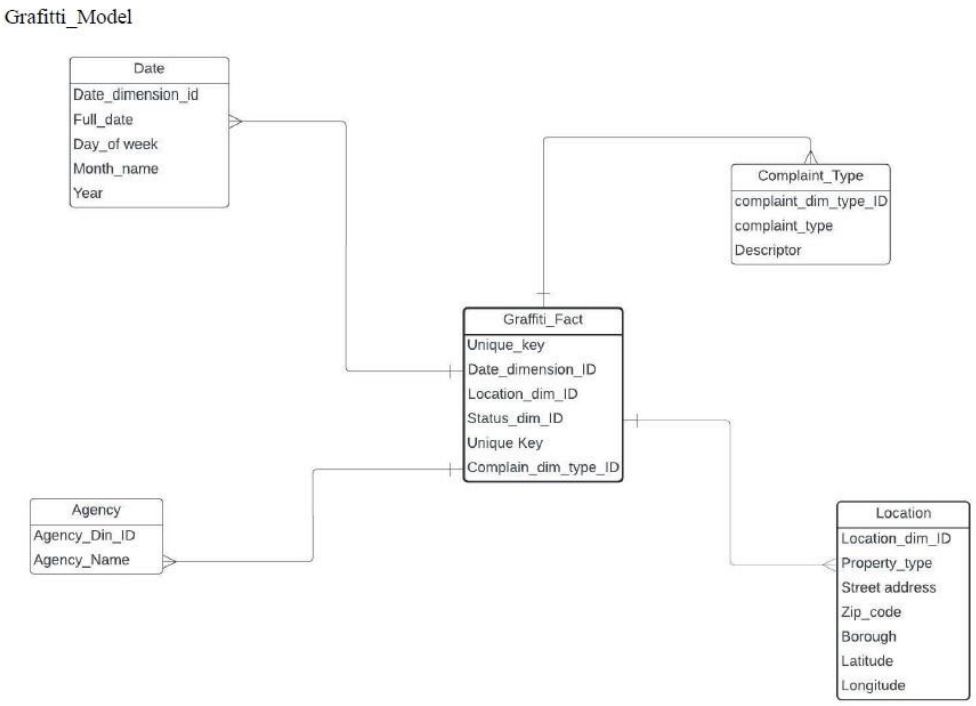
The hypothesis is that when a community district experiences irregular or inadequate sanitation services, it can contribute to this sense of "neglect," encouraging more graffiti and even other forms of vandalism. On the other hand, regular sanitation can serve as a deterrent and foster an environment of cleanliness. Understanding the correlation may guide policymakers in prioritizing sanitation in underserved communities and result in curbing graffiti and potentially other acts of vandalism.

Data Sources

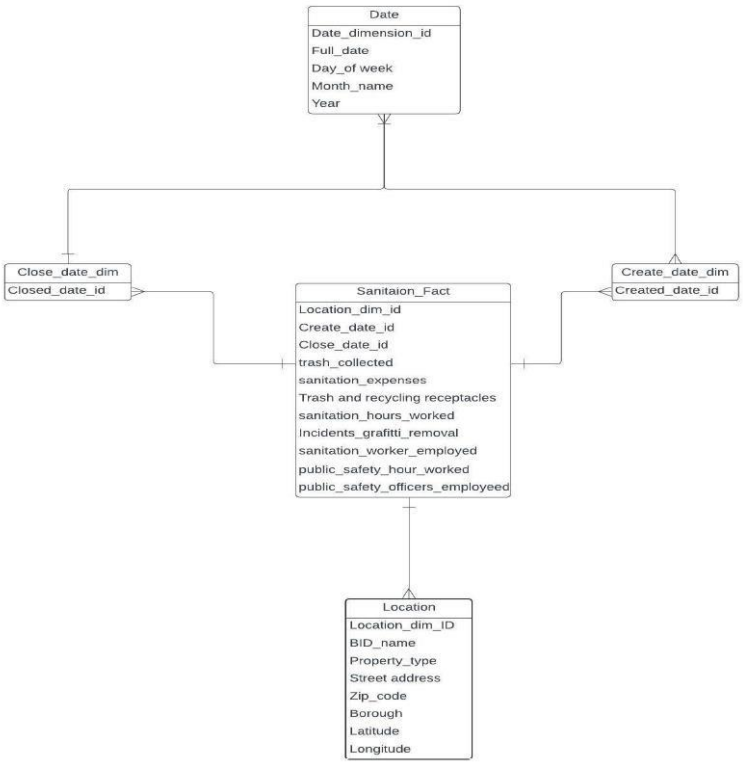
1. City Government - FY16 BID Trends Report Data
2. Social Services - NYC Graffiti
3. Social Services - Sanitation Complaints

Dimensional model diagram

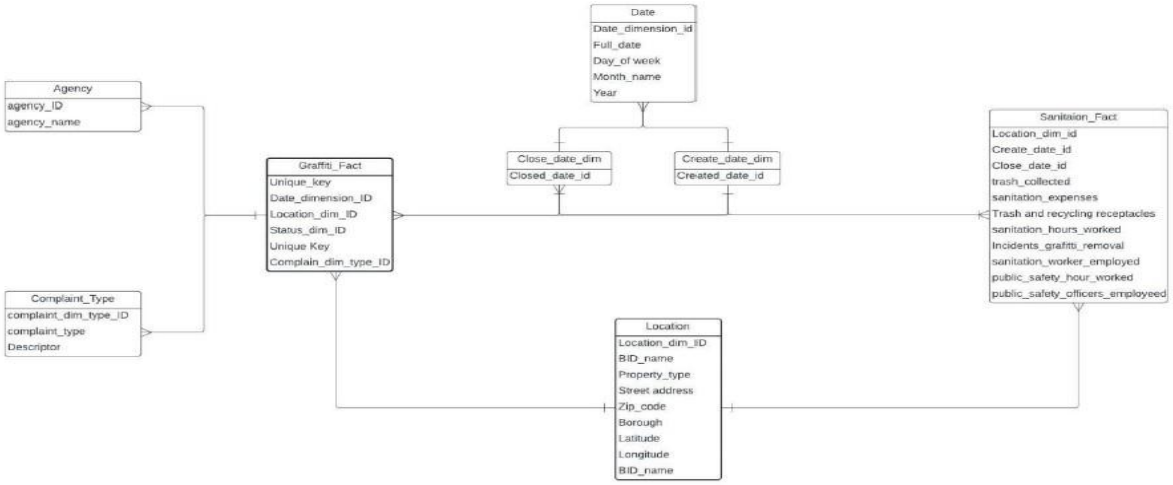
Graffiti_model



Sanitation_model



Final_integrated_model



Tools Used

The project utilized several tools across different phases. **Google BigQuery** served as the primary database, enabling efficient storage and querying of the transformed data. The **dbt (Data Build Tool)** was employed for the ETL process, allowing seamless data transformations and ensuring the data was analytics-ready. The programming languages **SQL** and **Python** were used for querying and preliminary data profiling, respectively. For data visualization, **Tableau** were used to create interactive dashboards and generate insightful KPI visualizations. Finally, the hosting environment for the project was **Google Cloud**, which provided a scalable and secure infrastructure for data storage and processing.

ETL Processes

The ETL (Extract, Transform, Load) workflow for the project was designed to integrate graffiti and sanitation data effectively.

1. Extract Phase

The data was sourced from the **NYC Open Data Portal**, a public platform that hosts datasets related to various city services. The required datasets, such as graffiti complaints and sanitation records, were downloaded in CSV format from this portal. This approach ensured that the raw data was accessible and ready for the subsequent transformation process.

2. Transform Phase

In this phase, multiple datasets were joined to create unified tables that facilitated analysis. Key transformations included merging data on graffiti complaints with sanitation service records based on common dimensions like location and date. These transformations aimed to standardize the data and prepare it for analytical queries in the final data warehouse.

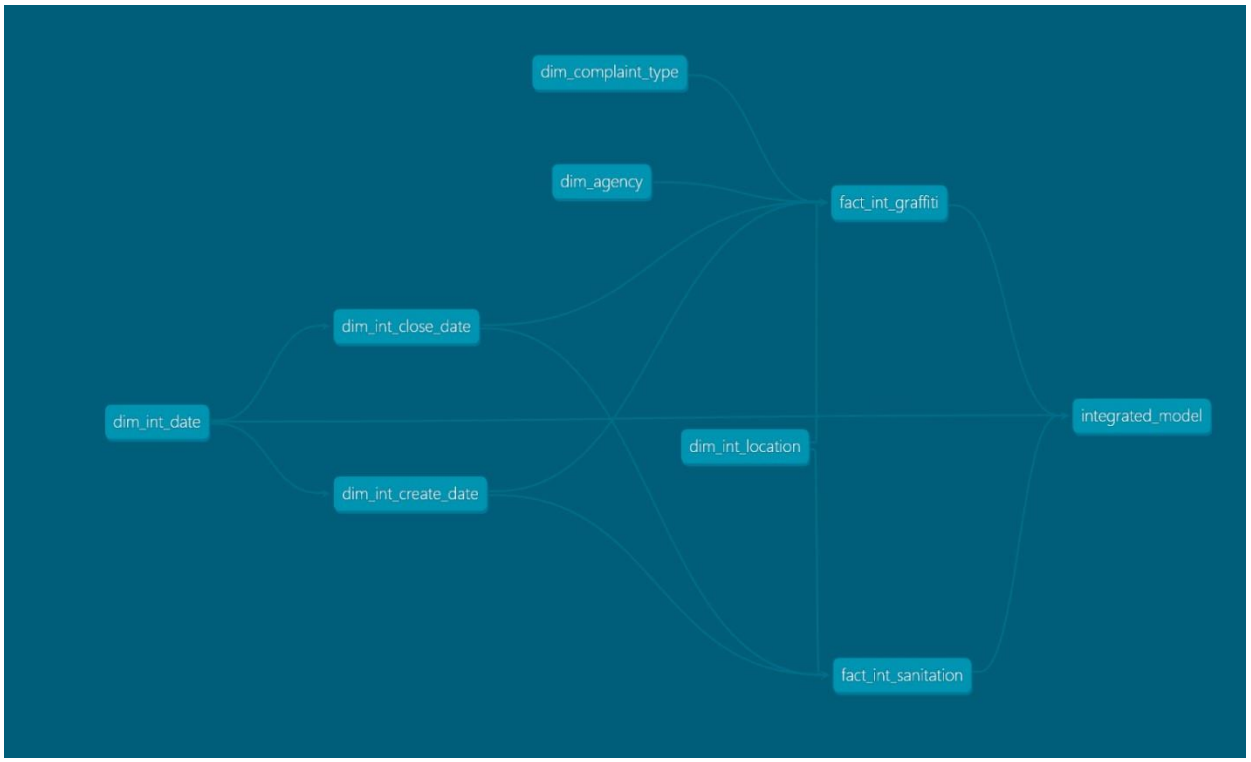
3. Load Phase

The transformed data was loaded into **Google BigQuery**, a cloud-based data warehouse designed for large-scale analytics. **dbt (Data Build Tool)** was utilized to define and execute the transformation models. This streamlined the ELT process by enabling SQL-based transformations directly within the warehouse.

4. ETL Lineage

The ETL lineage process focused on creating the **fact_graffiti** and **fact_sanitation** tables. These tables were modeled to aggregate data, ensuring that the relationships between facts and dimensions were correctly established. This lineage provided a clear mapping of how raw data evolved through extraction, transformation, and loading into analytics-ready structures.

DAG diagram:



Fact Graffiti:

```
SELECT
  c.`Unique_Key` AS unique_key,
  d.date_dimension_id,
  a.agency_dim_id,
  ct.complain_dim_type_id,
  l.location_dim_id,
FROM
  `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset` c
JOIN {{ ref('dim_date') }} d ON c.`Created_Date` = TIMESTAMP(d.full_date)
JOIN {{ ref('dim_agency') }} a ON c.Agency = a.agency_name
JOIN {{ ref('dim_complaint_type') }} ct ON CONCAT(c.`Complaint_Type`, '_', c.`Descript
JOIN {{ ref('dim_location') }} l ON CONCAT(c.`Location__Type`, '_', c.`Incident_Zip`,
GROUP BY
  unique_key, d.date_dimension_id, a.agency_dim_id, ct.complain_dim_type_id, l.locatio
```

Fact Sanitation

```
WITH base_data AS (
    SELECT
        BID_Name,
        Borough,
        `Incident Address`,
        CONCAT(`BID_Name`, '_', `Borough`, '_', `Incident Address`) AS location_dim_id,
        SAFE_CAST(`Sanitation Staff Employed` AS INT64) AS sanitation_worker_employed,
        SAFE_CAST(`Public Safety Employed` AS INT64) AS public_safety_officers_employe
        SAFE_CAST(`Sanitation hours Worked` AS FLOAT64) AS sanitation_hours_worked,
        SAFE_CAST(`Public Safety Hours Worked` AS FLOAT64) AS public_safety_hour_worke
        SAFE_CAST(`Trash Bags Collected` AS INT64) AS trash_collected,
        SAFE_CAST(`Trash Receptacles Services` AS INT64) AS trash_and_recycling_recept
        SAFE_CAST(`Graffiti Incidents Removed` AS INT64) AS incidents_graffiti_removal,
        SAFE_CAST(`Sanitation Expenses` AS FLOAT64) AS sanitation_expenses,
        SAFE_CAST(`Public Safety Expenses` AS FLOAT64) AS public_safety_expenses,
        PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Incident Date`)) AS create_
        PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Resolution Update Action Da
    FROM
        `cis9440finalproject-442001.source_dataset.bid_trends_sanitation`
```

```
SELECT
    bd.sanitation_worker_employed,
    bd.public_safety_officers_employed,
    bd.sanitation_hours_worked,
    bd.public_safety_hour_worked,
    bd.trash_collected,
    bd.trash_and_recycling_receptacles,
    bd.incidents_graffiti_removal,
    bd.sanitation_expenses,
    bd.public_safety_expenses,
    d.create_date_id,
    cd.close_date_id AS close_date_id,
    l.Location_dim_ID AS location_dim_id

FROM base_data bd
JOIN {{ ref('dim_create_date') }} d
    ON bd.create_date_id = d.create_date_id
JOIN {{ ref('dim_close_date') }} cd
    ON bd.close_date_id = cd.close_date_id
JOIN {{ref ('dim_Sanitation_location') }} l
    ON bd.location_dim_id = l.location_dim_id
```

Final Dimensional Schema

1. Dimensional Schema Overview

The project is based on a well-structured **star schema** designed to facilitate efficient analytical queries on available graffiti and sanitation data. The schema comprises two central fact tables **fact_graffiti** and **fact_sanitation** which store key metrics. These fact

tables are linked to their respective dimension tables, enabling comprehensive analysis of the correlation between graffiti incidents and sanitation services.

2. Components of the Schema

The schema includes the following components:

Fact Tables

The schema contains two primary fact tables:

1. **fact_graffiti**: This table captures metrics related to graffiti incidents, including complaint counts, types, and associated agencies. It is connected to dimensions such as **dim_date**, **dim_location**, **dim_complaint_type**, and **dim_agency**.

```
SELECT
  c.`Unique_Key` AS unique_key,
  d.date_dimension_id,
  a.agency_dim_id,
  ct.complain_dim_type_id,
  l.location_dim_id,
FROM
  `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset` c
JOIN {{ ref('dim_date') }} d ON c.`Created_Date` = TIMESTAMP(d.full_date)
JOIN {{ ref('dim_agency') }} a ON c.Agency = a.agency_name
JOIN {{ ref('dim_complaint_type') }} ct ON CONCAT(c.`Complaint_Type`, '_', c.`Descript
JOIN {{ ref('dim_location') }} l ON CONCAT(c.`Location__Type`, '_', c.`Incident_Zip`,
GROUP BY
  unique_key, d.date_dimension_id, a.agency_dim_id, ct.complain_dim_type_id, l.locatio
```

2. **fact_sanitation**: This table tracks sanitation metrics such as trash collected, hours worked, and sanitation expenses. It is linked to dimensions like **dim_date**, **dim_location**, **dim_close_date**, and **dim_create_date**.

```
WITH base_data AS (
    SELECT
        BID_Name,
        Borough,
        `Incident Address`,
        CONCAT(`BID_Name`, '_', `Borough`, '_', `Incident Address`) AS location_dim_id,
        SAFE_CAST(`Sanitation Staff Employed` AS INT64) AS sanitation_worker_employed,
        SAFE_CAST(`Public Safety Employed` AS INT64) AS public_safety_officers_employed,
        SAFE_CAST(`Sanitation hours Worked` AS FLOAT64) AS sanitation_hours_worked,
        SAFE_CAST(`Public Safety Hours Worked` AS FLOAT64) AS public_safety_hour_worked,
        SAFE_CAST(`Trash Bags Collected` AS INT64) AS trash_collected,
        SAFE_CAST(`Trash Receptacles Services` AS INT64) AS trash_and_recycling_receptacles,
        SAFE_CAST(`Graffiti Incidents Removed` AS INT64) AS incidents_graffiti_removal,
        SAFE_CAST(`Sanitation Expenses` AS FLOAT64) AS sanitation_expenses,
        SAFE_CAST(`Public Safety Expenses` AS FLOAT64) AS public_safety_expenses,
        PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Incident Date`)) AS create_
        PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Resolution Update Action Da
    FROM
        `cis9440finalproject-442001.source_dataset.bid_trends_sanitation`
```

```
SELECT
    bd.sanitation_worker_employed,
    bd.public_safety_officers_employed,
    bd.sanitation_hours_worked,
    bd.public_safety_hour_worked,
    bd.trash_collected,
    bd.trash_and_recycling_receptacles,
    bd.incidents_graffiti_removal,
    bd.sanitation_expenses,
    bd.public_safety_expenses,
    d.create_date_id,
    cd.close_date_id AS close_date_id,
    l.Location_dim_ID AS location_dim_id

FROM base_data bd
JOIN {{ ref('dim_create_date') }} d
    ON bd.create_date_id = d.create_date_id
JOIN {{ ref('dim_close_date') }} cd
    ON bd.close_date_id = cd.close_date_id
JOIN {{ref ('dim_Sanitation_location') }} l
    ON bd.location_dim_id = l.location_dim_id
```

Dimension Tables Graffiti:

The dimension tables provide descriptive attributes for the fact tables:

- **dim_date**: Contains date detail which includes date_dimention_id as primary key, full_date, day_of _week, month_name and year.

```
SELECT DISTINCT
    `Unique_Key`,
    DATE_TRUNC(CAST(`Created_Date` AS DATE), DAY) AS date_dimension_id,
    CAST(`Created_Date` AS DATE) AS full_date,
    FORMAT_TIMESTAMP('%A', CAST(`Created_Date` AS TIMESTAMP)) AS day_of_week,
    FORMAT_TIMESTAMP('%B', CAST(`Created_Date` AS TIMESTAMP)) AS month_name,
    FORMAT_TIMESTAMP('%Y', CAST(`Created_Date` AS TIMESTAMP)) AS year
FROM
    `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset`
```

- **dim_location**: Contains geographical data, including borough, zip code, and latitude/longitude, to support location-based analytics.

dim_location.sql

models / dim_graffiti / dim_location.sql

```sql
SELECT DISTINCT
 `Unique_Key`,
 CONCAT(`Incident_Zip`, '_', `Borough`) AS location_dim_id,
 `Location__Type` AS property_type,
 `Incident_Address` AS street_address,
 `Incident_Zip` AS zip_code,
 `Borough`
FROM
 `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset`
```

- **dim_complaint_type**: Provides detailed information about graffiti complaints and their descriptors.

dim_complaint_type.sql

models / dim_graffiti / dim_complaint_type.sql

```sql
SELECT DISTINCT
    `Unique_Key`,
    CONCAT(`Complaint_Type`, '_', `Descriptor`) AS complain_dim_type_id,
    `Complaint_Type`,
    `Descriptor`
FROM
    `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset`

```

- **dim_agency**: Includes details about agencies responsible for handling graffiti-related complaints.

dim_agency.sql

models / dim_graffiti / dim_agency.sql

```sql
SELECT DISTINCT
    CONCAT(`Agency`, '_',`Unique_Key`) AS agency_dim_id,
    `Unique_Key` as unique_Key,
    `Agency` AS agency_name
FROM
    `cis9440finalproject-442001.source_dataset.grafitti_complaint_dataset`
```

Sanitation:

- **dim_close_date** and **dim_create_date**: These tables focus on tracking sanitation event timings, such as when tasks were created and completed.

models / dim_sanitation / dim_create_date.sql

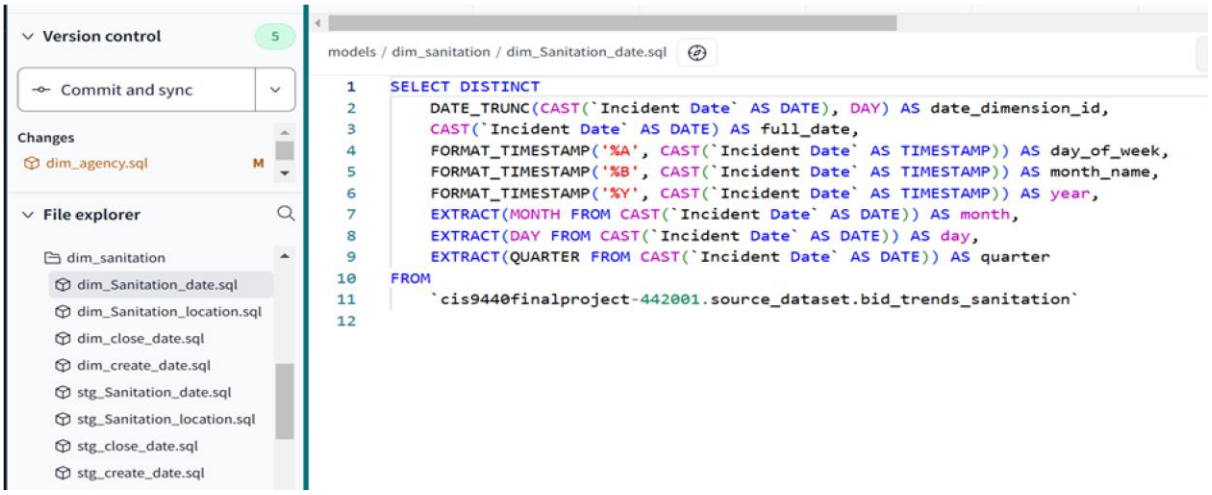
```sql
SELECT DISTINCT
    PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Incident Date`)) AS create_
FROM
    `cis9440finalproject-442001.source_dataset.bid_trends_sanitation` st
JOIN {{ ref('dim_Sanitation_date') }} ss
    ON CAST(st.`Incident Date` AS DATE) = ss.full_date

```

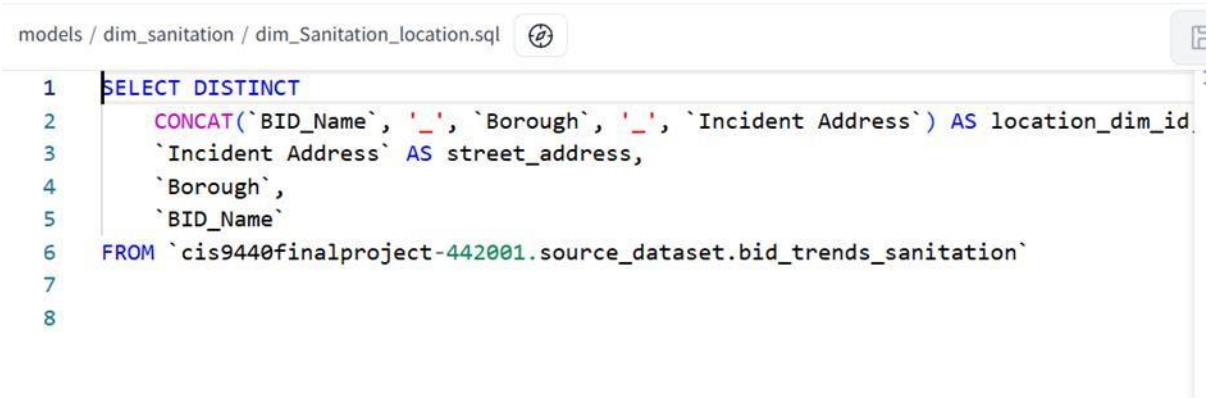
models / dim_sanitation / dim_close_date.sql

```sql
SELECT DISTINCT
    PARSE_DATE('%Y-%m-%d', FORMAT_DATETIME('%Y-%m-%d', `Resolution Update Action Da
FROM
    `cis9440finalproject-442001.source_dataset.bid_trends_sanitation` st
JOIN {{ ref('dim_Sanitation_date') }} ss
    ON CAST(st.`Resolution Update Action Date` AS DATE) = ss.full_date
```

- **date_dimension:** Contains date detail which includes date_dimention_id as primary key, full_date, day_of _week, month_name and year.



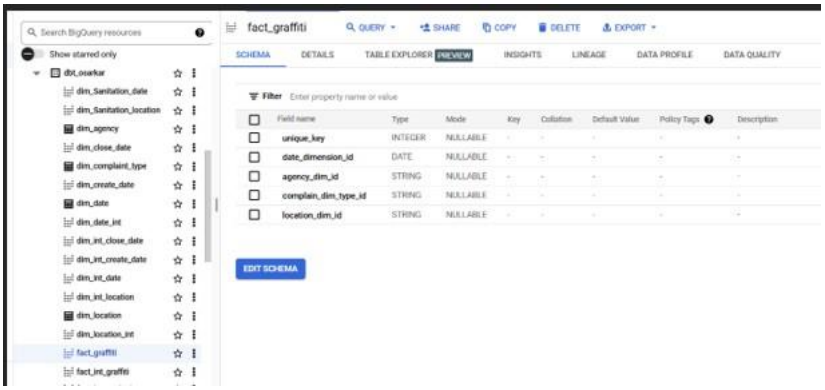
Location_dimension: This dimension contains descriptive information of the location data which includes location_dim_id as a primary key, followed with BID_Name, property_type, street, zip_code, borough, latitude and longitude.



Relationships

The fact tables are connected to their respective dimension tables using foreign keys. This structure creates a **star schema** that is optimized for querying and reporting, enabling seamless integration of fact and dimension data.

And this is the picture of how it looks like in the Google BigQurey



---

3. Schema Diagrams:

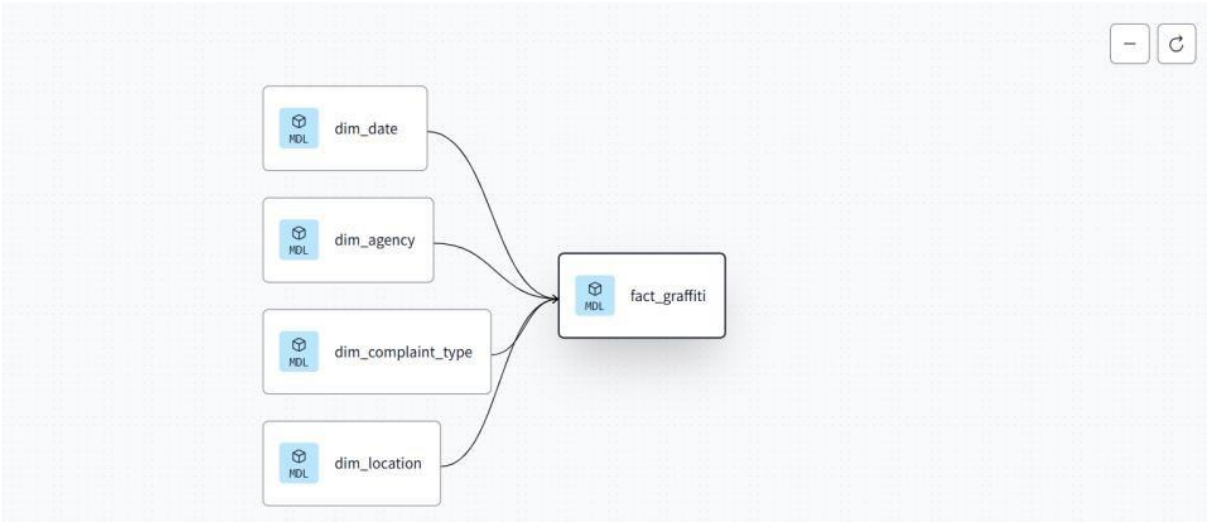
The following diagrams illustrate the dimensional schema and its components:

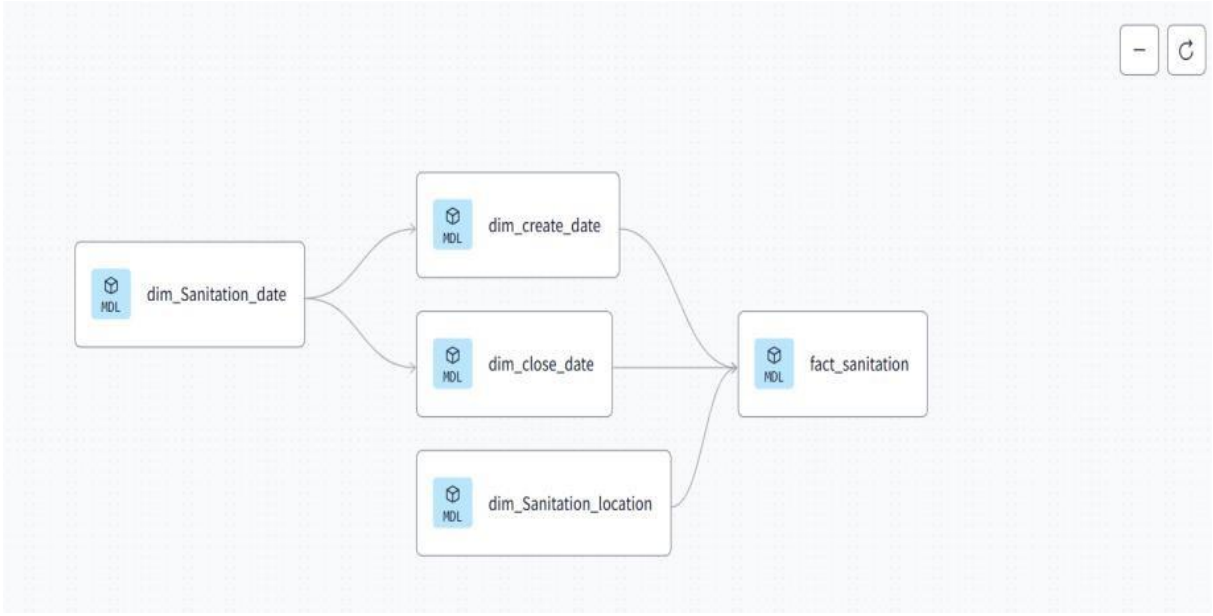
1. **Graffiti Schema**: Highlights the relationships between graffiti data and dimensions such as date, location, complaint type, and agency.

2. **Sanitation Schema**: Displays how sanitation data is linked to dimensions like date, location, and event timings.

3. **Integrated Schema**: Combines both graffiti and sanitation schemas for unified analysis.

These diagrams provide a clear visualization of the database structure and relationships, aiding in understanding and further analysis.

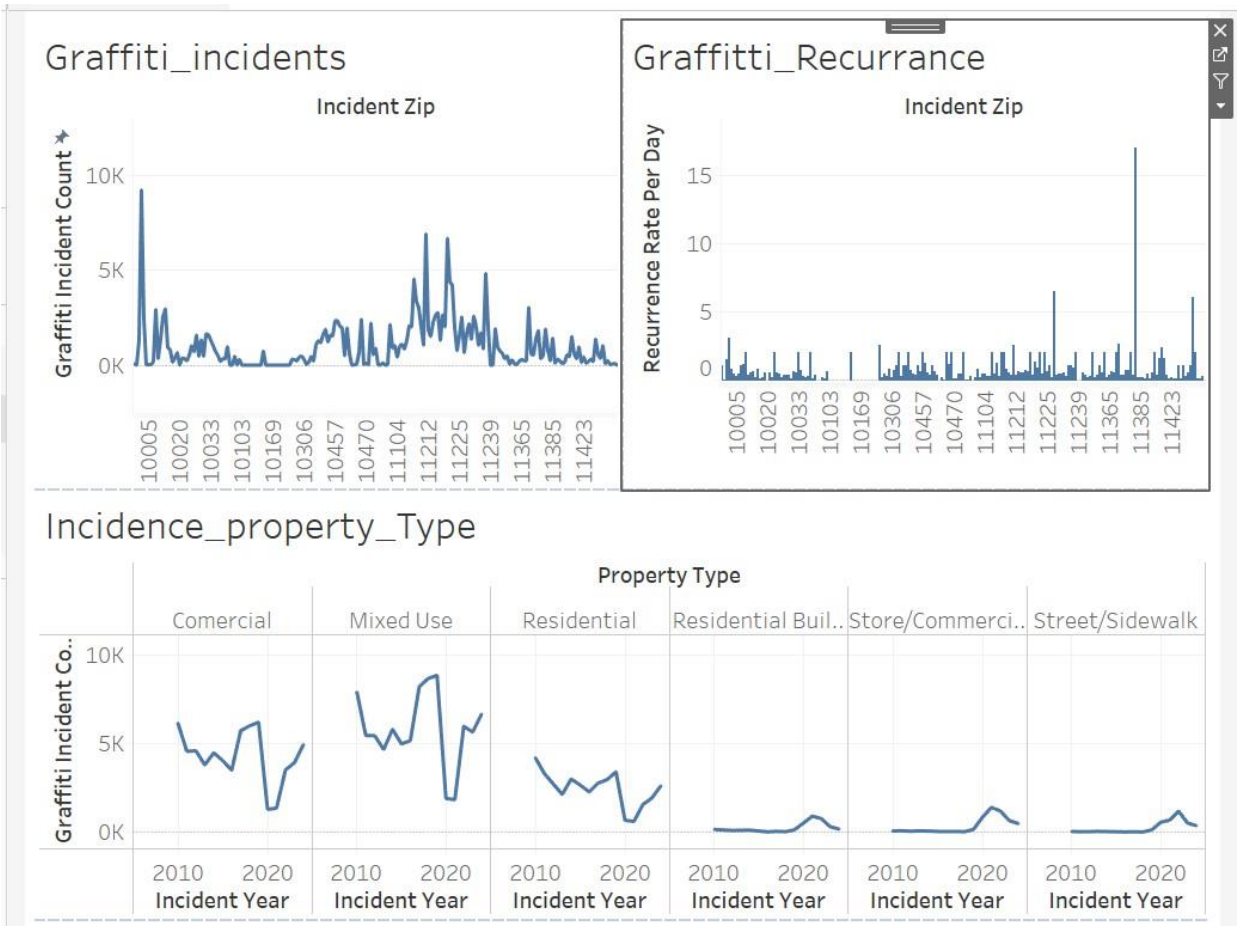




These diagrams provide a visual representation of the relationships between fact and dimension tables, facilitating easier understanding the database structural compostion.

KPI Visualizations

Graffiti KPIs Visualizations:



Description:

Graffiti_Incidents (Top Left):

This line chart represents the count of graffiti incidents across various ZIP codes. The x-axis lists ZIP codes, while the y-axis shows the number of incidents. The data highlights variations in graffiti incidents, with certain ZIP codes like 10005 and 11223 exhibiting significant spikes.

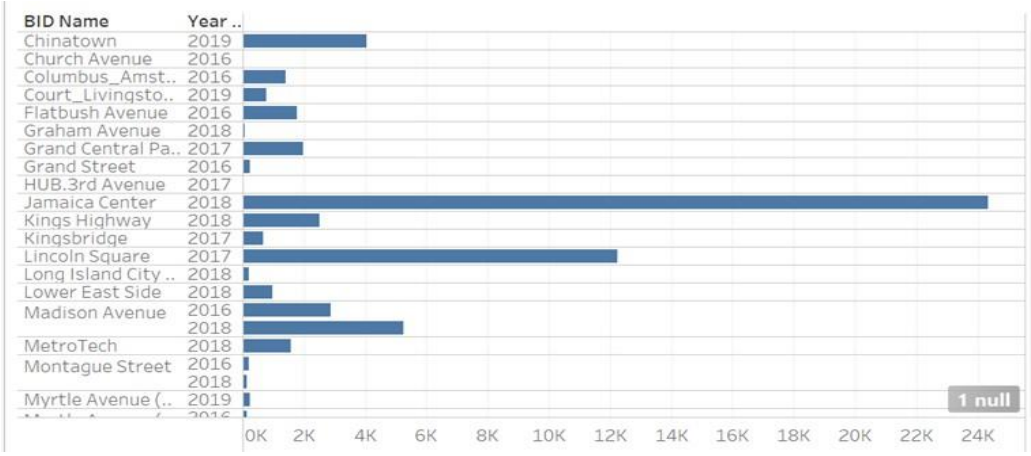
Graffiti_Recurrence (Top Right):

This chart displays the recurrence rate of graffiti incidents per day by ZIP code. The x-axis shows ZIP codes, and the y-axis indicates the average recurrence rate. ZIP codes like 11385 and 11423 have noticeable peaks, suggesting repeated graffiti occurrences.

Incidence_Property_Type (Bottom):

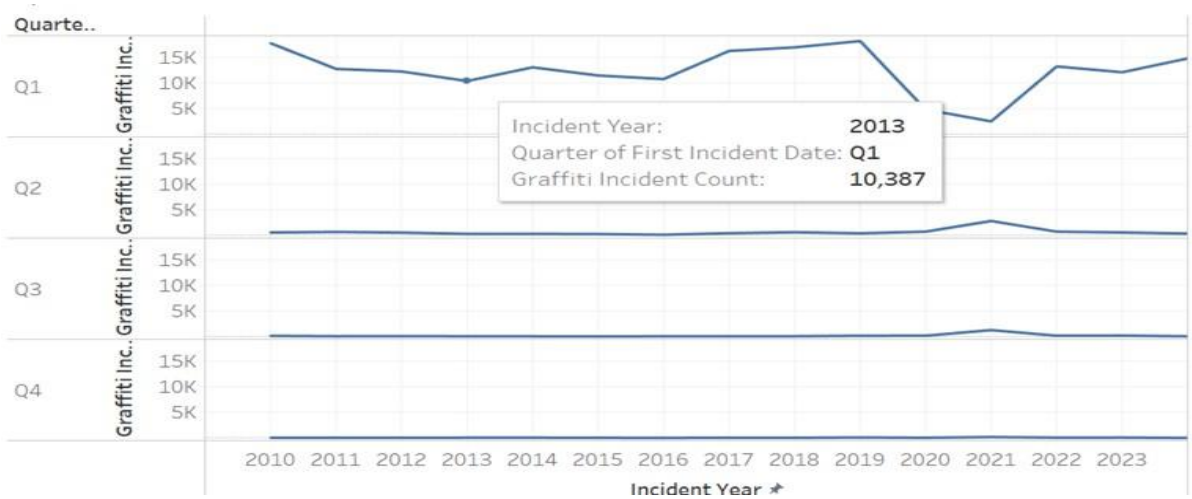
The series of line charts detail graffiti incidents by property type (e.g., Commercial, Mixed Use, Residential) over the years from 2010 to 2020. The x-axis represents the year, and the yaxis indicates incident counts. Trends reveal fluctuations across property types, with consistent activity in mixed-use and commercial properties, while residential areas display fewer incidents.

Graffiti Removal Sucess Rate



The bar chart illustrates the Graffiti removal success rate across New York City, grouped by year. The data highlights significant variations, with Jamaica Center and Kings Highway showing the highest values.

Graffiti Seasonality



- This graph illustrates the quarterly graffiti incident counts reported from 2010 to 2023. The data shows notable variations, with peak incidents in Q1 of 2013 (10,387 cases). Trends over time highlight key periods of fluctuation and potential impact from policy changes or external factors influencing graffiti occurrence.

Sanitation Visualisation:

Description:

Trash_Recycling (Top Left):

This bar chart visualizes the number of trash and recycling receptacles across various Business Improvement Districts (BIDs). The x-axis indicates the count of receptacles, while the y-axis lists BID names. Notable BIDs like 34th Street Partnership and 47th Street show a significantly higher number of receptacles, while some entries contain null values, highlighting missing data.

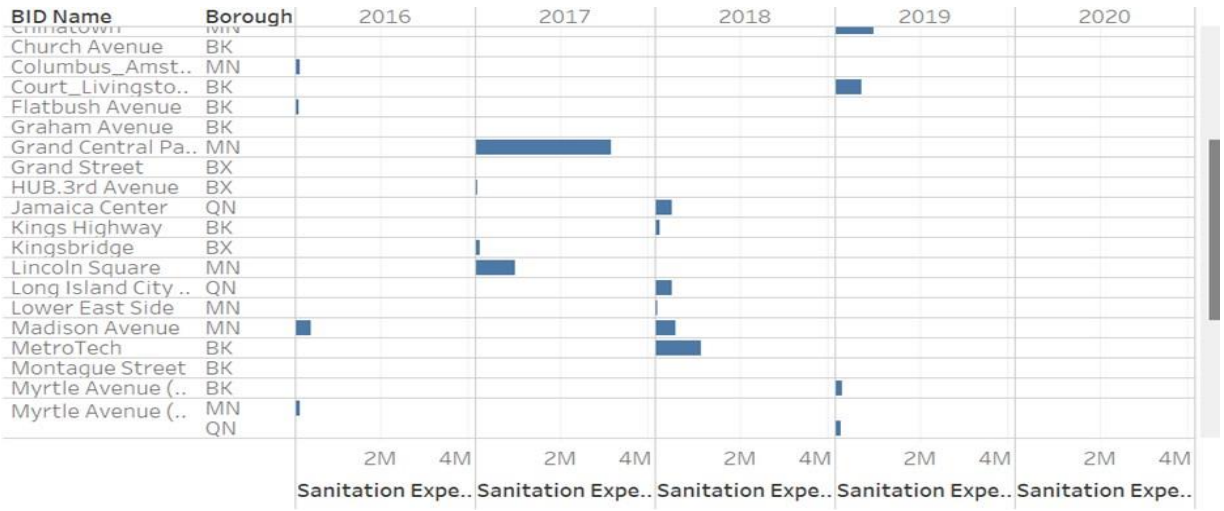
Trash_Collected_BR_Level (Top Right):

This bar chart represents the amount of trash collected (in thousands) across different BIDs and boroughs. The x-axis denotes trash collected, while the y-axis lists BID names and boroughs. 34th Street Partnership and Bryant Park Corporation in Manhattan (MN) display the highest volumes of trash collected.

Resolution_Time (Bottom):

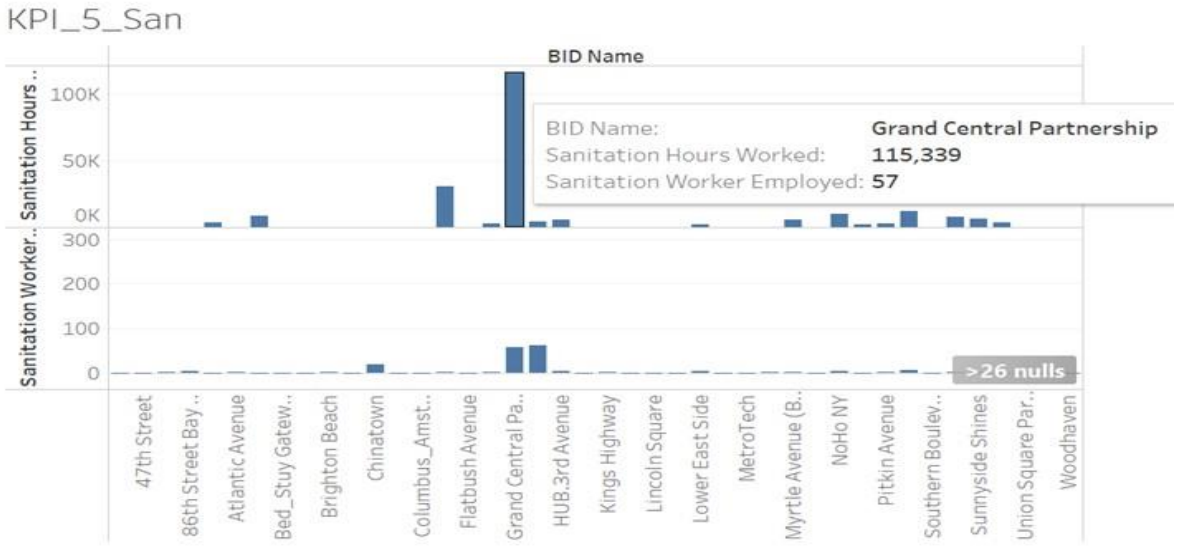
This bar chart illustrates the resolution time for issues across various BIDs and boroughs. The x-axis represents the resolution time (in hours/days), and the y-axis lists BID names. Chinatown and Columbus-Amsterdam exhibit the longest resolution times, while other BIDs show quicker issue resolution.

Sanitation Expenses



- This graph displays the sanitation expenditure trends across different Business Improvement Districts (BIDs) and boroughs from 2016 to 2020. The data highlights varying levels of investment in sanitation efforts, with noticeable spikes in specific BIDs such as Grand Central Partnership and Madison Avenue during certain years. The visualization provides insights into the prioritization of resources over time and across locations.

Sanitation Worker Hours



- This graph illustrates sanitation hours worked and workers employed across various Business Improvement Districts (BIDs). The Grand Central Partnership stands out with 115,339 sanitation hours and 57 workers, significantly higher than other districts. The data sheds light on the allocation of sanitation resources and workforce distribution across BIDs, providing insights into operational scale and focus areas.

Tools Used

The project made use of several tools across different phases to achieve its objectives efficiently. **Google BigQuery** served as the primary database for storing and analyzing the cleaned and transformed data. It provided scalable and efficient querying capabilities to handle the large datasets involved. For visualization, **Tableau** was utilized to create interactive dashboards, showcasing KPIs such as graffiti density, sanitation response times, and seasonal trends.

The ETL process was streamlined using **dbt (Data Build Tool)**, which allowed for the transformation of data directly within BigQuery. This tool facilitated defining models, writing SQL transformations, and managing the entire data pipeline. **Python** played a critical role in data cleaning, preliminary transformations, and profiling, with libraries like Pandas and NumPy being particularly useful. Additionally, **SQL** was used extensively for querying and manipulating data within BigQuery and for modeling transformations in dbt.

The project also relied on the **NYC Open Data Portal** as the primary source for raw datasets on graffiti complaints and sanitation services. Lastly, **Lucidchart** was employed for designing dimensional models and schema diagrams, ensuring clarity in visualizing data relationships.

Narrative Conclusion

Tools Used and Their Purposes

Several tools were integral to the success of this project. **Google BigQuery** served as the primary database for storing and querying data, while **dbt (Data Build Tool)** was pivotal in transforming raw data into an analytics-ready structure. **Tableau** and enabled the creation of interactive dashboards, providing effective visualization of KPIs. **Python** facilitated data cleaning, profiling, and static visualizations, while **Lucidchart** was used to design dimensional

models and schema diagrams. The **NYC Open Data Portal** acted as the source for all raw data related to graffiti and sanitation services. Each tool was chosen for its ability to handle specific tasks efficiently, ensuring seamless collaboration and accurate results.

Tools and Platforms

The tools and platforms used in this project included:

1. **Google BigQuery**: Cloud-based data warehouse for storing and querying project data.
2. **dbt (Data Build Tool)**: Used for modeling and transforming data.
3. **Tableau**: For creating dashboards and KPI visualizations.
4. **Lucidchart**: For designing dimensional models and schema diagrams.
5. **Python**: For data profiling and visualizations, utilizing libraries like Pandas, Matplotlib, and Seaborn.

Other References

- **NYC Open Data Documentation**: Provided detailed descriptions of the datasets used.
- **dbt Documentation**: Offered guidance on setting up and using dbt with BigQuery.
- **Tableau** Aided in designing effective visualizations.

***** Thank you*****