**CUNY Baruch College**

**Service on the Street and at the Table: Investigating Taxi and Restaurant Issues in NYC**

Final Project Report

Group-3 Members:

Dhruv Sharma

Pasang Syangba

Maung Aung

Sujasna Tamang

BUS 9440-27334

*DataWarehousing and Analytics*

Fall 2024

Professor: Ramah Al Balawi

December 13, 2024

**Introduction**

Narrative description of the project

New York City, renowned for its fast-paced lifestyle and diverse cultural offerings, heavily relies on two vital service sectors: transportation and dining. However, both sectors face critical issues that impact the city's residents and visitors. Taxi services often attract complaints related to unsafe driving, route disputes, and pickup refusals, while restaurants are frequently evaluated for compliance with health standards, with some receiving poor grades due to critical violations. Addressing these challenges requires an integrated approach to uncover patterns and correlations that may exist between service quality in these sectors.

This project aims to create a comprehensive data warehouse to investigate the interplay between taxi complaints and restaurant inspection results. By analyzing these datasets, we aim to answer the central question: **Can the safety and the service quality of a borough be evaluated through restaurant inspection results and taxi complaints?**

#### **Overview of Source Data**

To achieve this goal, we leverage two critical datasets:

1. 311 Taxi Complaints Dataset: This dataset, sourced from NYC Open Data, documents public complaints regarding taxi services, including unsafe driving, route disputes, service refusals, and unauthorized pickups. By analyzing the geographical and temporal distribution of these complaints, we aim to identify systemic service issues across the city.

* Source: NYC Open Data - 311 Service Requests

<https://data.cityofnewyork.us/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9/about_data>

1. DOHMH New York City Restaurant Inspection Results Dataset: This dataset, also sourced from NYC Open Data, provides detailed health inspection records of New York City restaurants. It includes scores, inspection grades, critical violation flags, and compliance indicators. The data allows us to evaluate neighborhood-level public health standards.

* Source: NYC Open Data - DOHMH Restaurant Inspection Results

<https://data.cityofnewyork.us/Health/DOHMH-New-York-City-Restaurant-Inspection-Results/43nn-pn8j/about_data>

The group will analyze the relationship between taxi complaints such as unsafe driving, route issues, service refusals, and New York City restaurant inspection results, including inspection grades, health violations, and compliance flags. By examining both datasets, the group aims to uncover patterns between taxi service issues and restaurant standards within specific boroughs, seeking to determine whether areas with lower restaurant inspection scores also experience higher rates of taxi complaints. This combined analysis could reveal broader trends in borough-wide safety and service quality, providing insights that may guide improvement in transportation and public health regulation across New York City.

**Key Performance Indicators (KPIs):**

Taxi Complaints Dataset

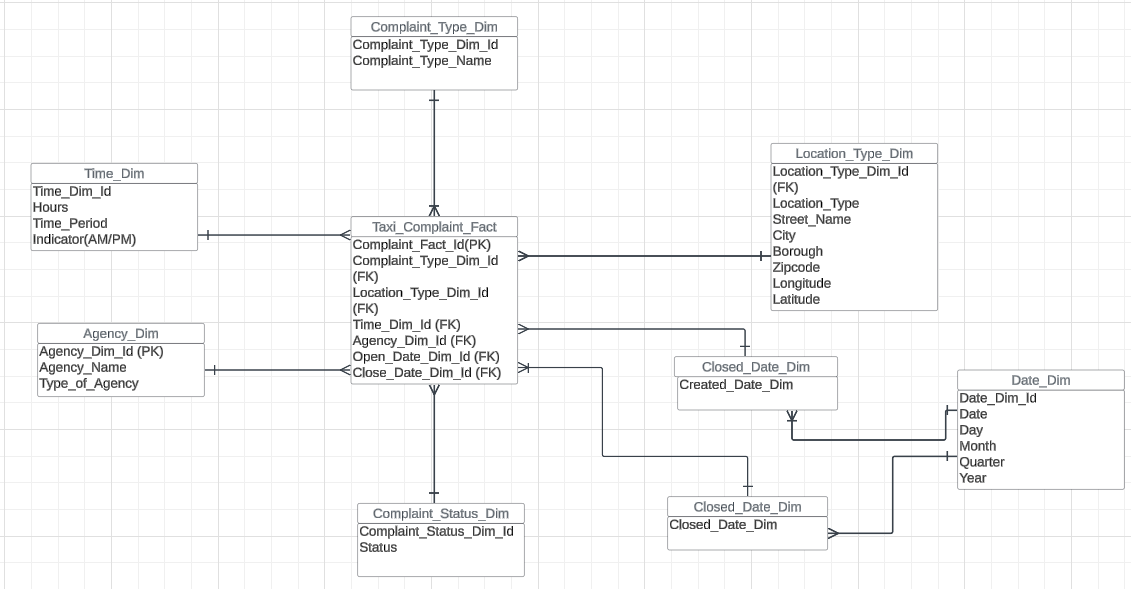
1. Total Number of Complaints by Type
2. Number of Complaints by Location Type
3. Complaint Frequency by Time of Day
4. Average Day of Complaint Closure by Agency
5. Number of Complaints by Status
6. Number of Complaints by Month

Restaurant Inspections Dataset

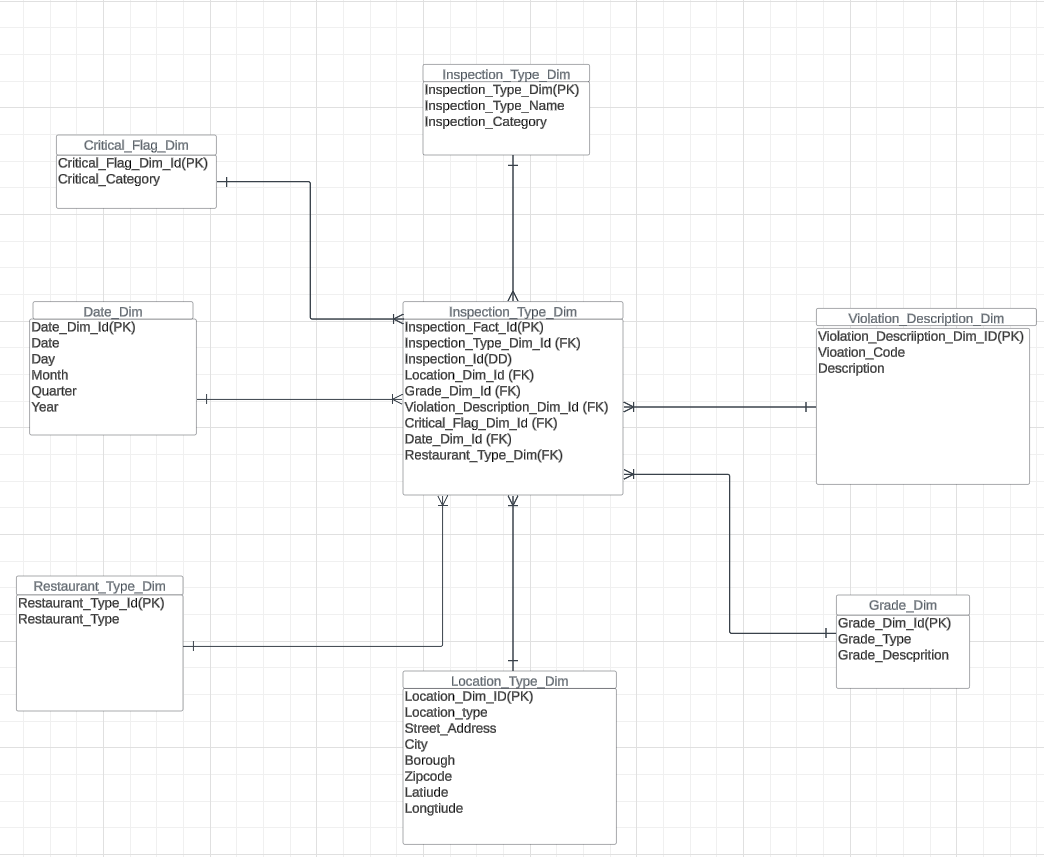
1. Number of inspections by Inspection Type.
2. Number of Inspection scores by Zipcode
3. Number of Inspections by Grade
4. Total Number of Complaints by Critical Flag
5. Number of Inspections by Month

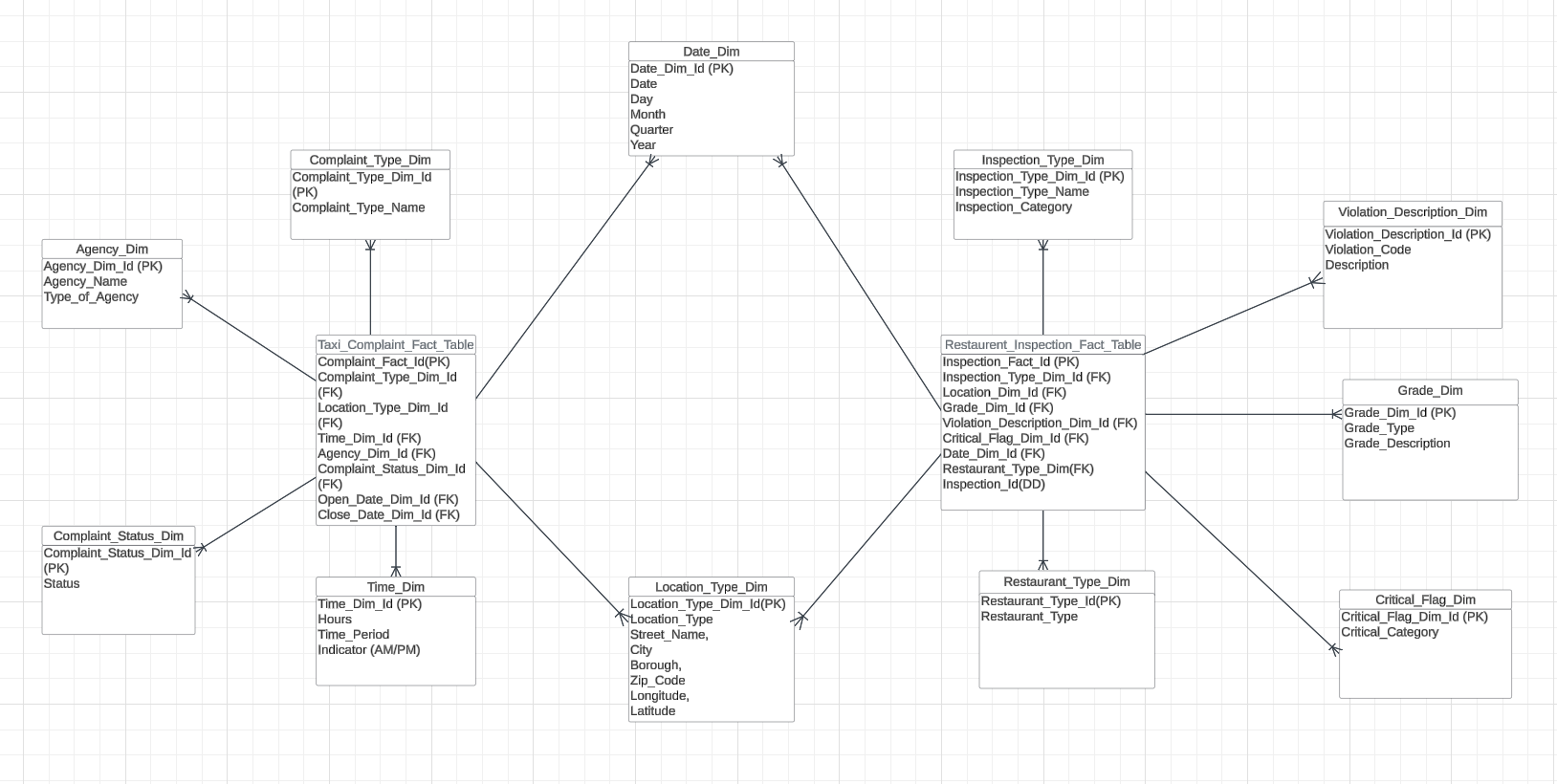
**Dimension Model Diagrams**

**Taxi complaints dimension model**



**Restaurant Inspection Dimension Model**



**Integrated Data Warehouse Model**

### **ETL Process**

We used Python for the ETL (Extract, Transform, Load) process of our project. The process was carried out as follows:

1. **Extraction and data profiling**

* **Tool Used:** Sodapy API in Jupyter Notebook

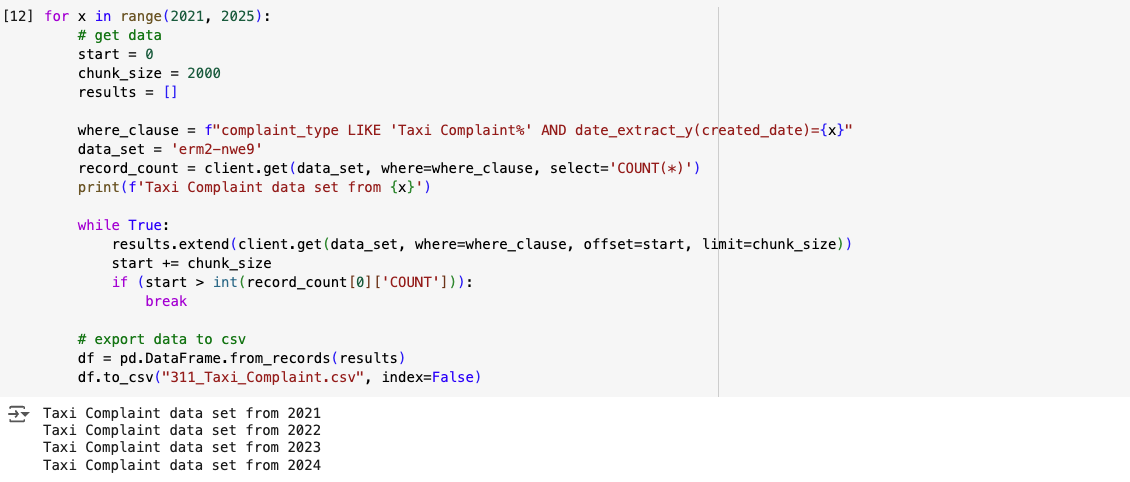
**Process Overview**:

We applied the same methodology to extract and profile both datasets. In this report, we will only provide the code for data extraction and data profiling for the “Taxi Complaints”.

**Extraction:**

The source data was a publicly available dataset hosted on NYC Open Data and retrieved using sodapy library. The extraction process was streamlined using defined parameters such as filters, date ranges of 01/01/2021 to 10/31/2024, to reduce noise and optimize data transfer. Then, the raw data was saved as CSV files for further processing and backup.



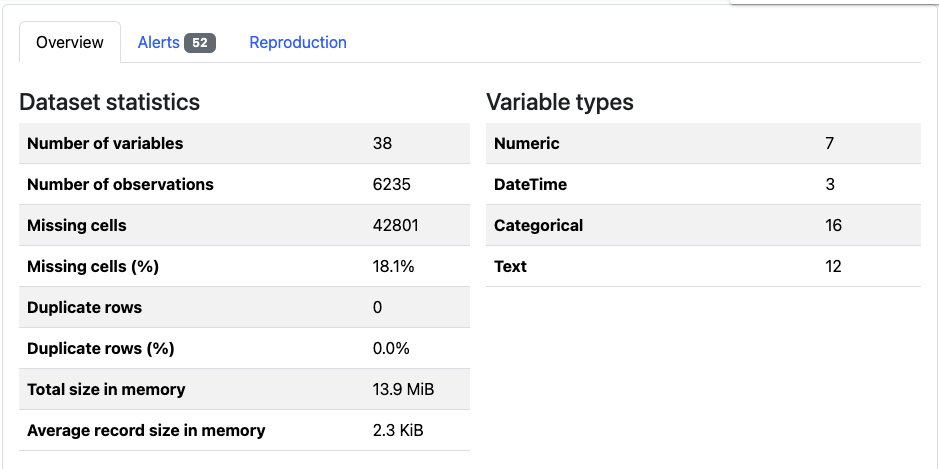


**Data profiling:**

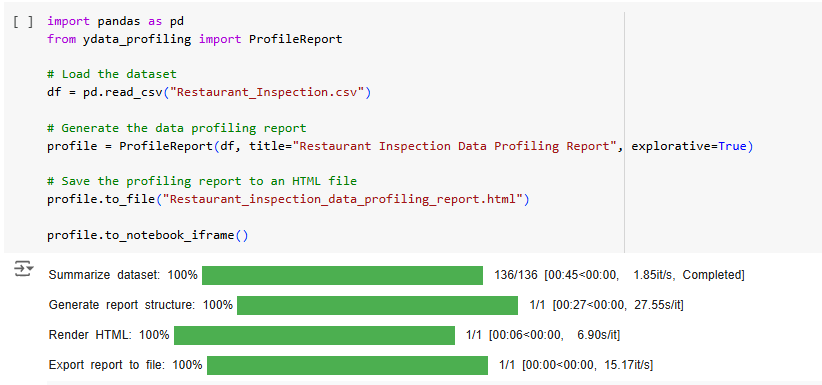
After extracting, data profiling was conducted to gain insights into the structure of our data structure and quality. This helped us to identify potential issues of missing values, outliers or inconsistencies.

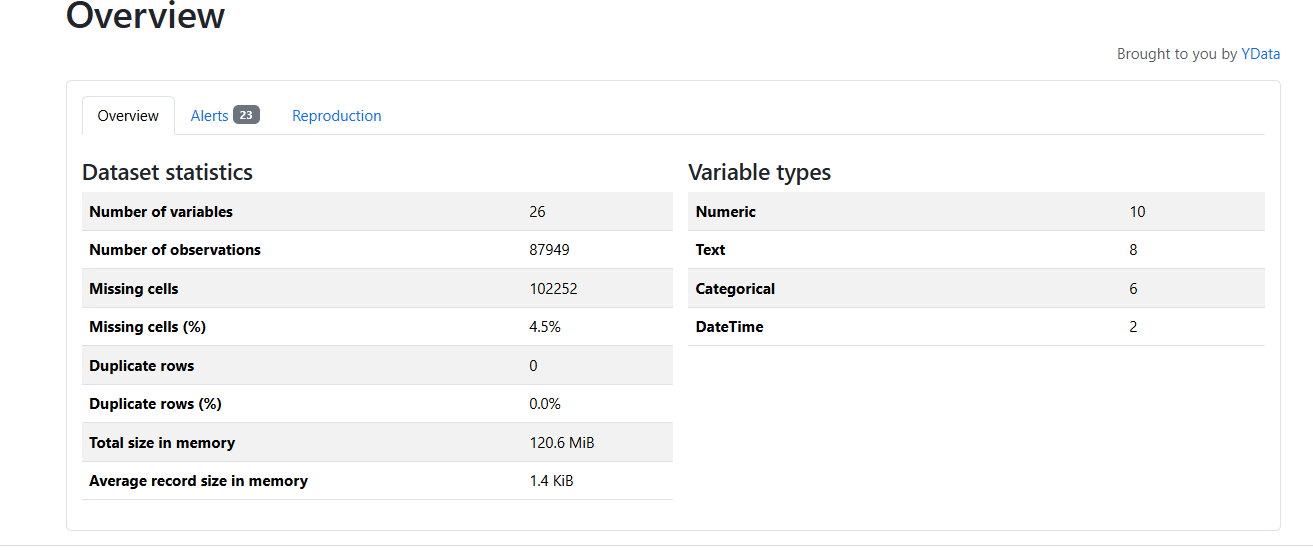
Taxi Complaint:





Restaurant Inspection:

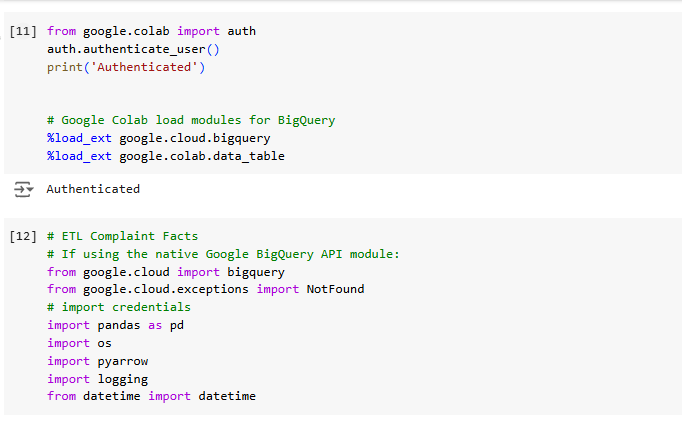




#### 

1. **Transformation**

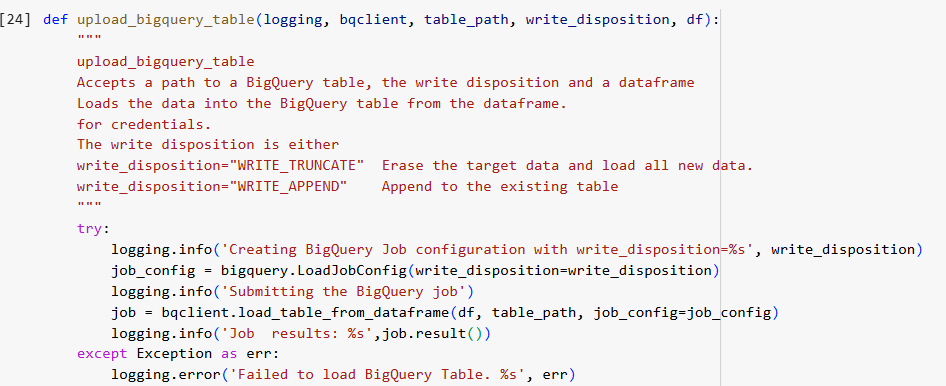
After extraction, we performed several data cleansing and transformation steps to ensure the data was accurate and suitable for analysis. Before we defined functions, we imported some necessary libraries. Below are our steps for the transformation process for both data sets.

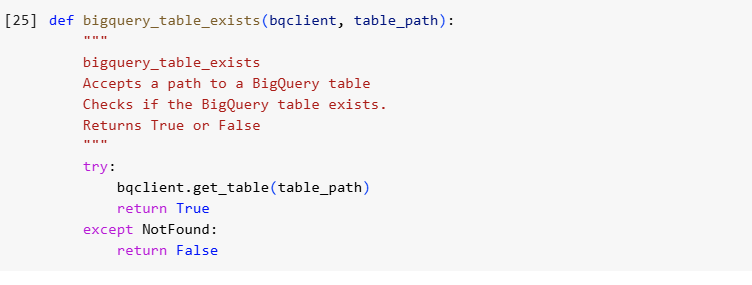


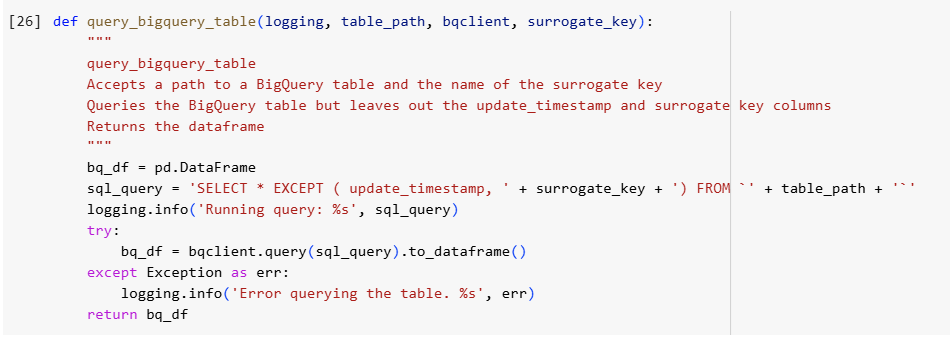
**Taxi Complaints:**



The code sets up a connection to BigQuery, a data warehouse service. It first tries to find a service account key from an environment variable or a specific file path. If successful, it uses this key to authenticate and create a BigQuery client. This client can then be used to perform various data operations, such as creating datasets. The code includes error handling to catch potential issues during the connection process.



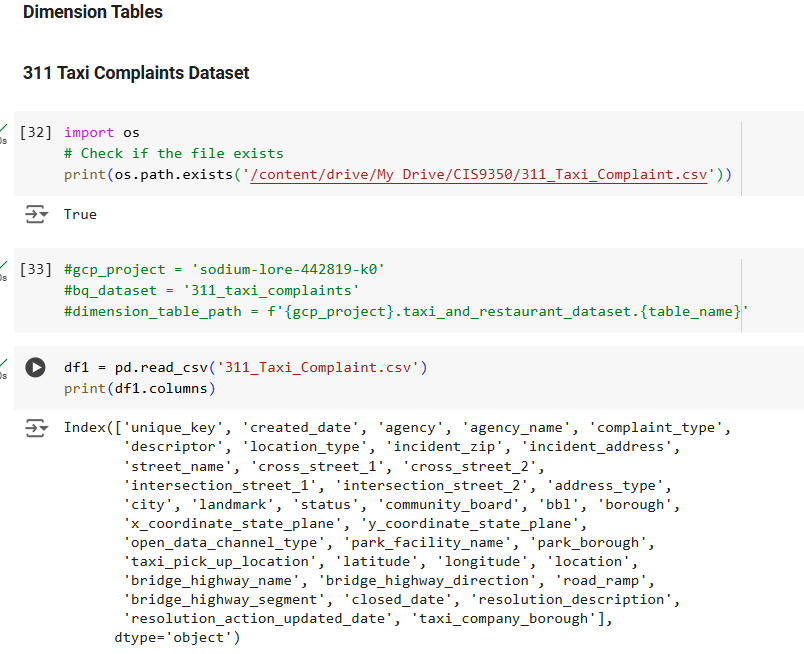






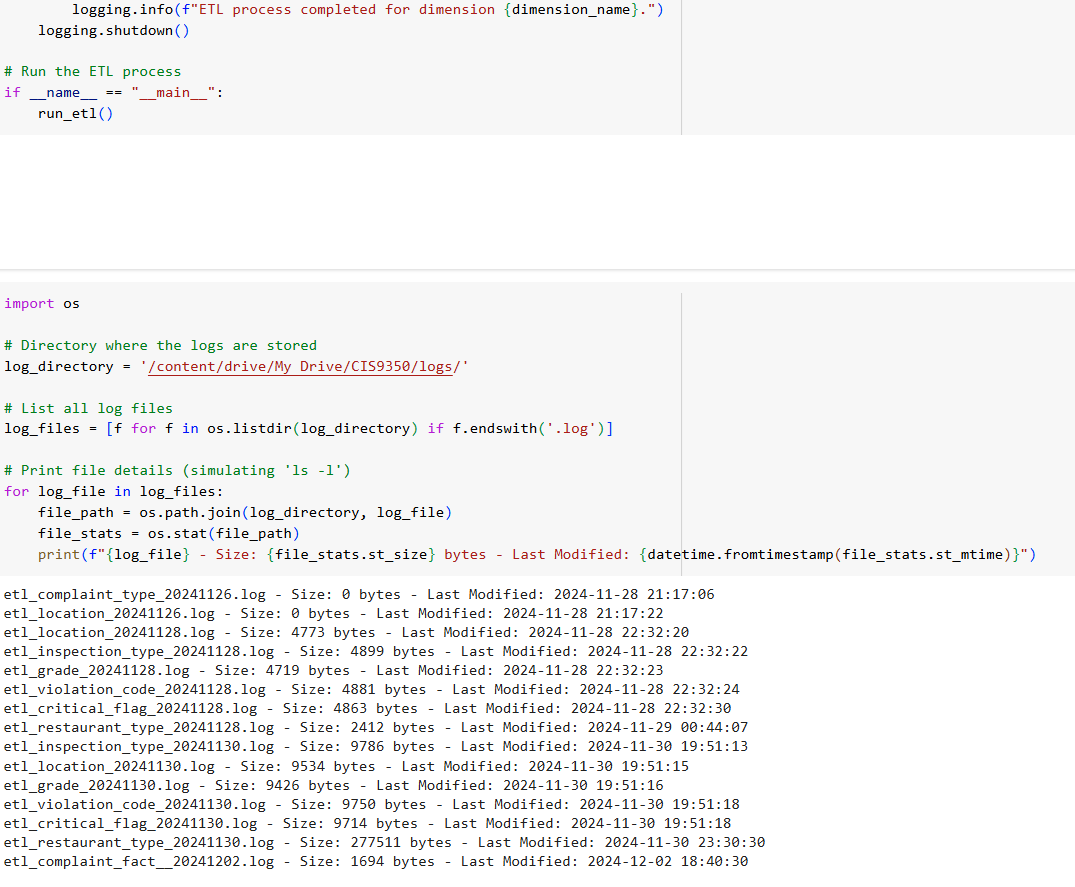


These functions handle data operations in BigQuery, including creating new dimensional tables, inserting new records, and updating existing ones. They check for existing data, add timestamps and unique identifiers, and upload data to the specified tables.

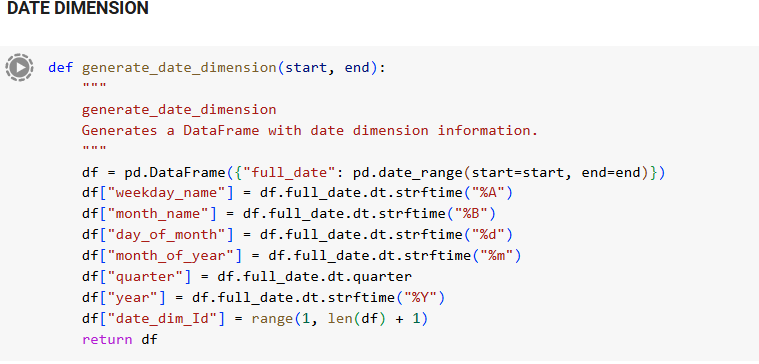








These codes helped us create dimensions for 311\_Taxi\_Complaints dataset other than time and date which are shown below in while creating the location dimension we replaced all the NaN values to “Unknown” in location type, Street name, and city and replaced NaN values to “0” for incident\_zip, Latitude and Longitude.

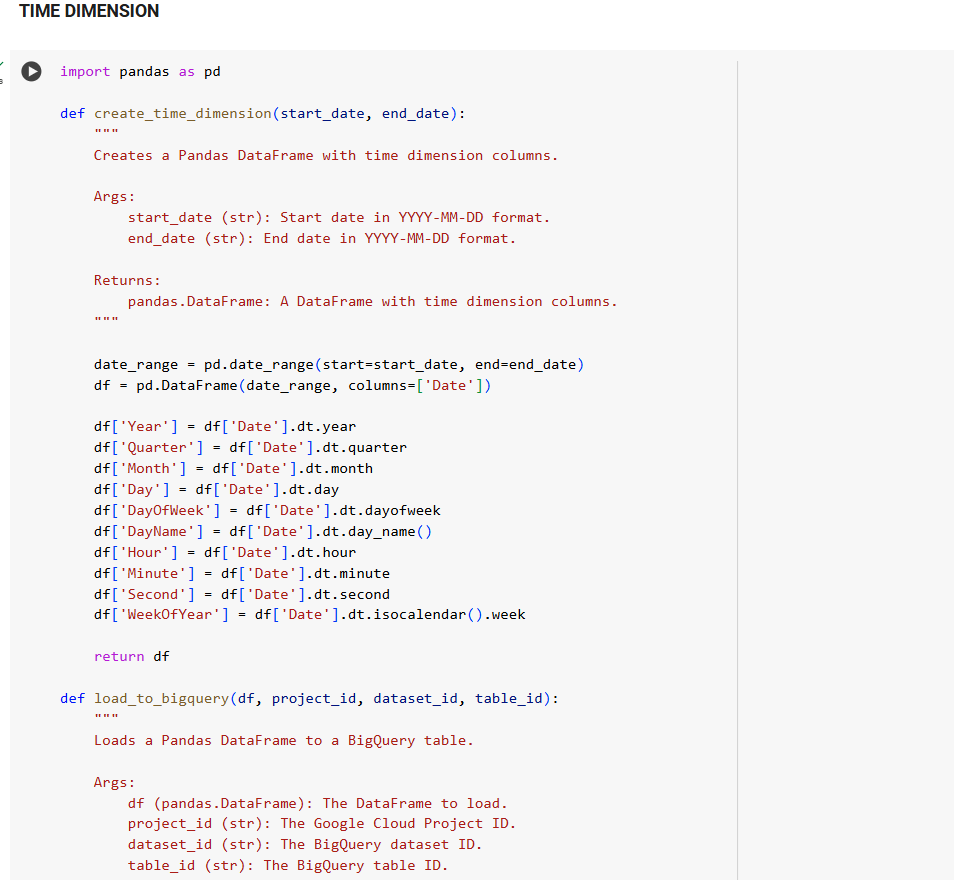


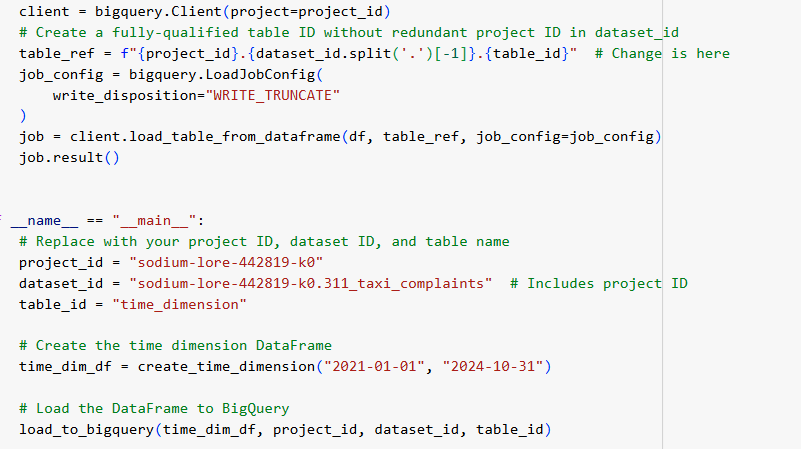
Each date is transformed into columns such as the Day of the month can be month and quarter can be 1 and follows.





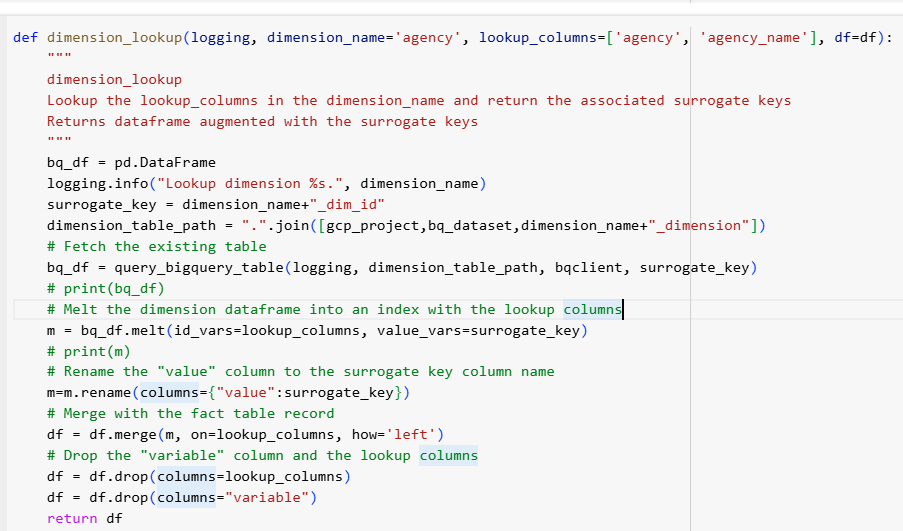
For date dimensions, we only extracted the ranges from 2021-01-01 to 2024-10-30 as we will only be analyzing from that range.



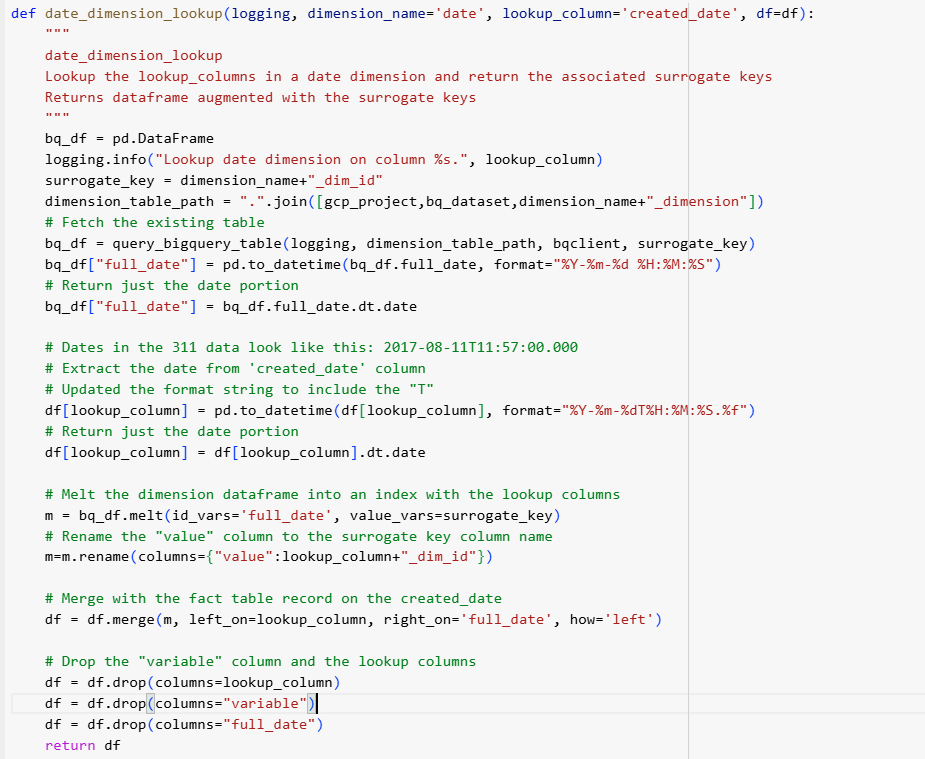


It first generates a range of dates and then extracts various time components like year, quarter, month, day of week, and more. This enriched dataset is then loaded into a specified BigQuery table, providing a valuable resource for time-based analysis and reporting.

**Creating Fact Table**



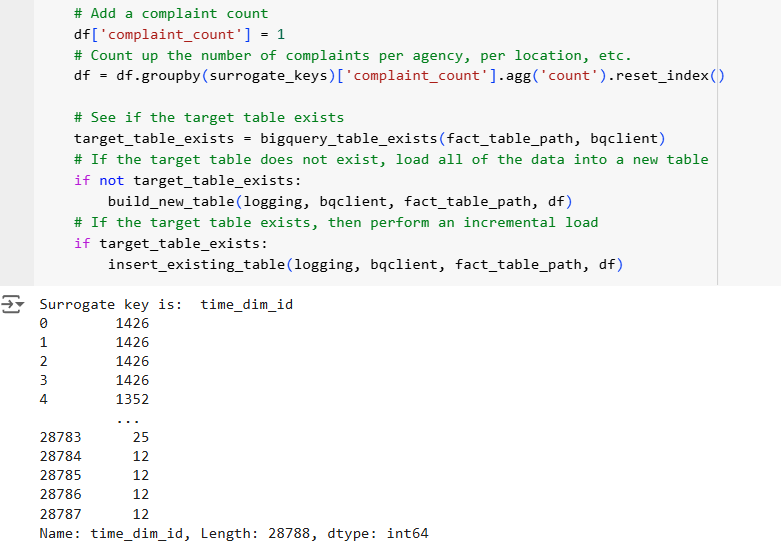
This helps merge a fact table with a dimension table to replace natural keys in the fact table with their corresponding surrogate keys from the dimension table.

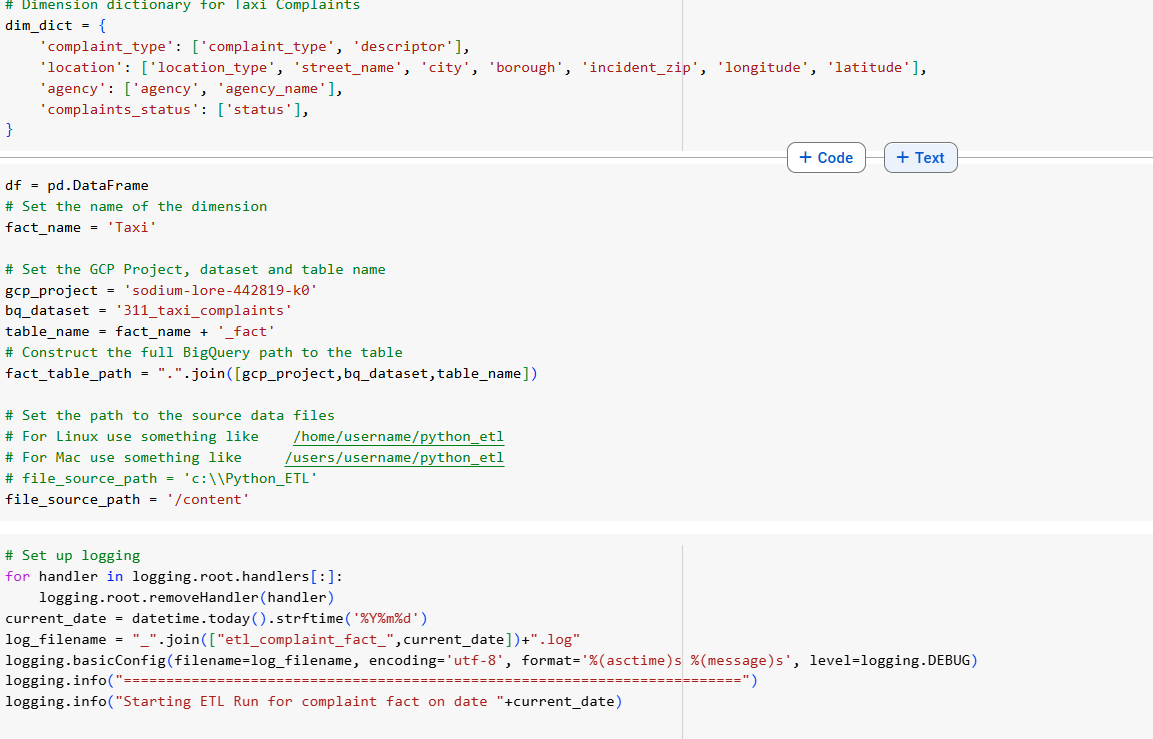












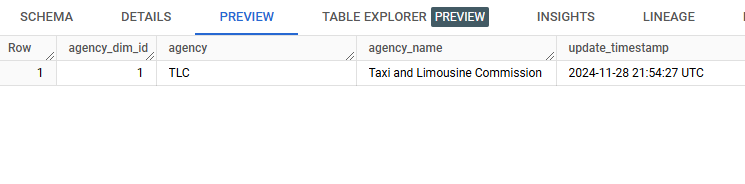
It aggregates the data by counting complaints for each combination of dimensions and loads the results into a BigQuery table.

**Final Dimension Schema:**

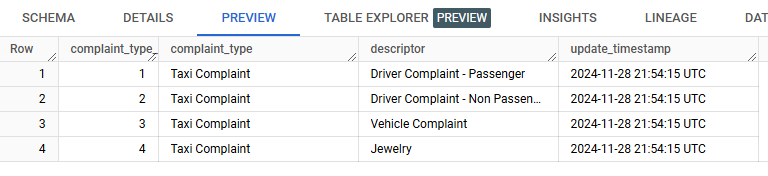
**Taxi Complaints:**

The following screenshots display the final dimensional schema in Google BigQuery, illustrating each dimension after the ETL process has been successfully completed.

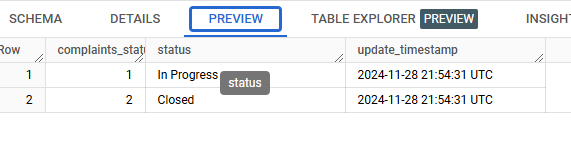
1. **Agency Dimension:**



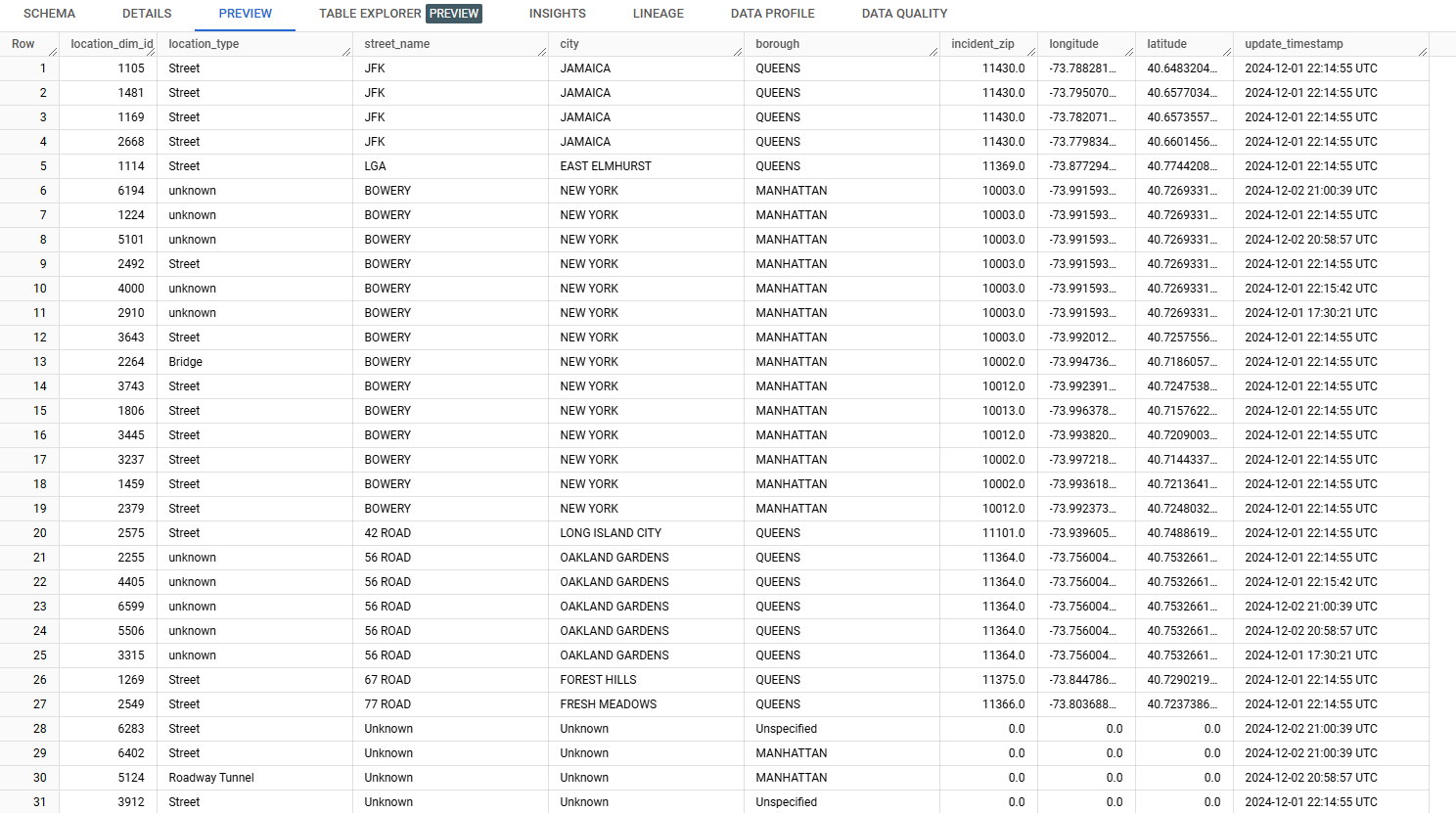
1. **Complaint Type Dimension:**



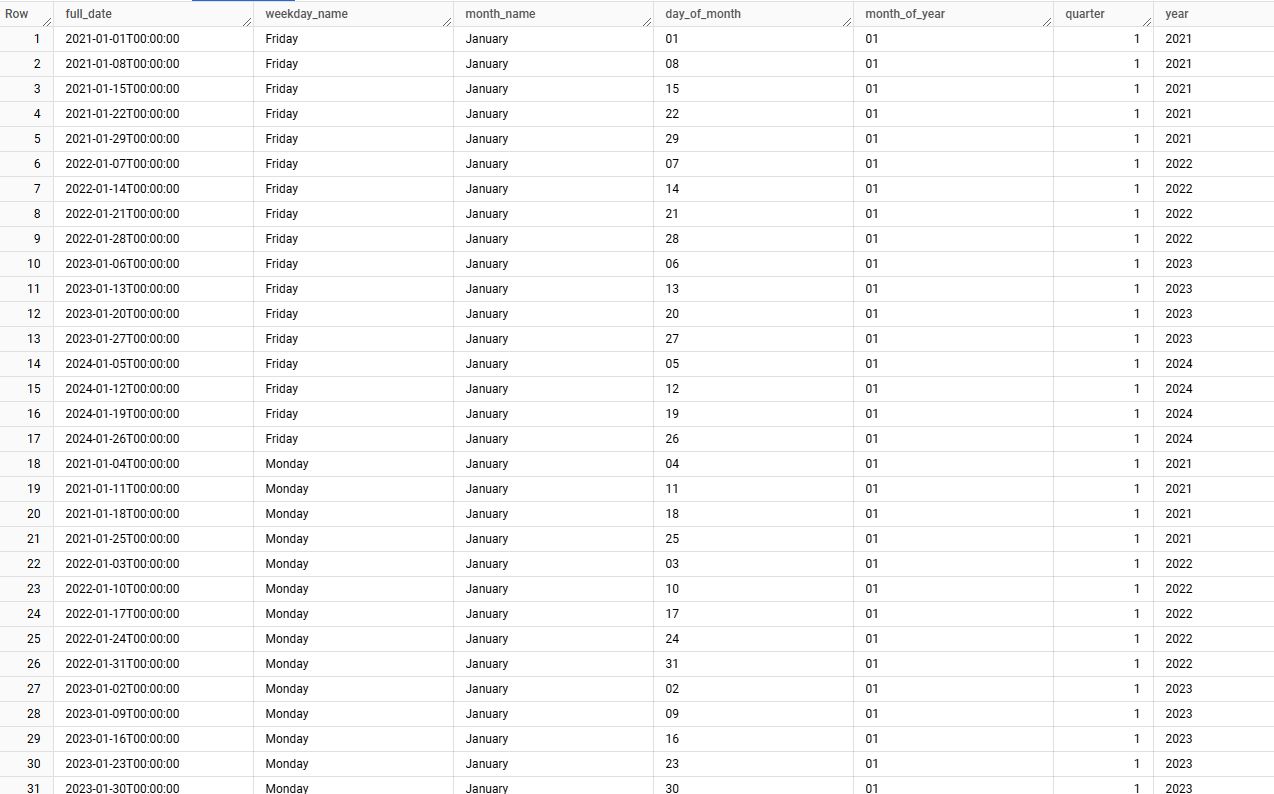
1. **Complaint Status Dimension:**



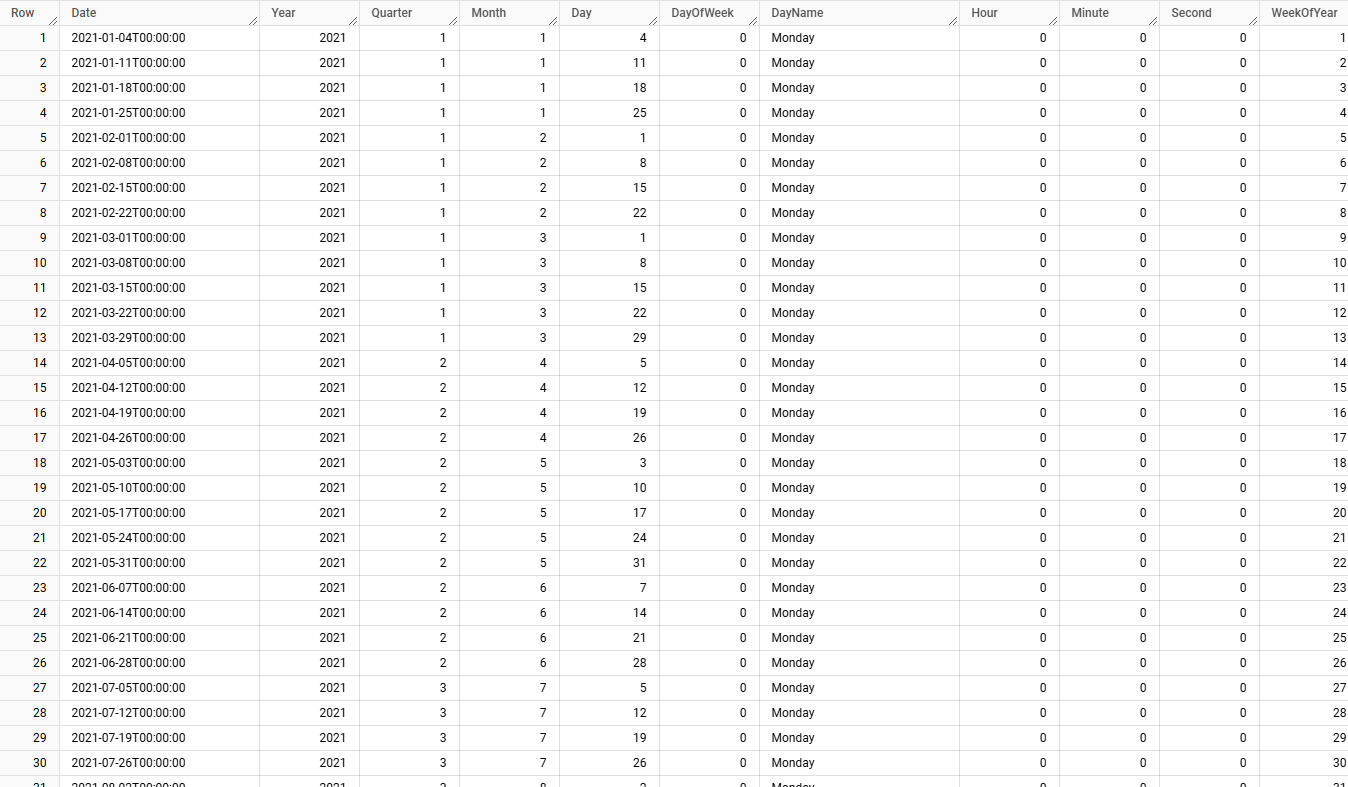
1. **Location Dimension:**



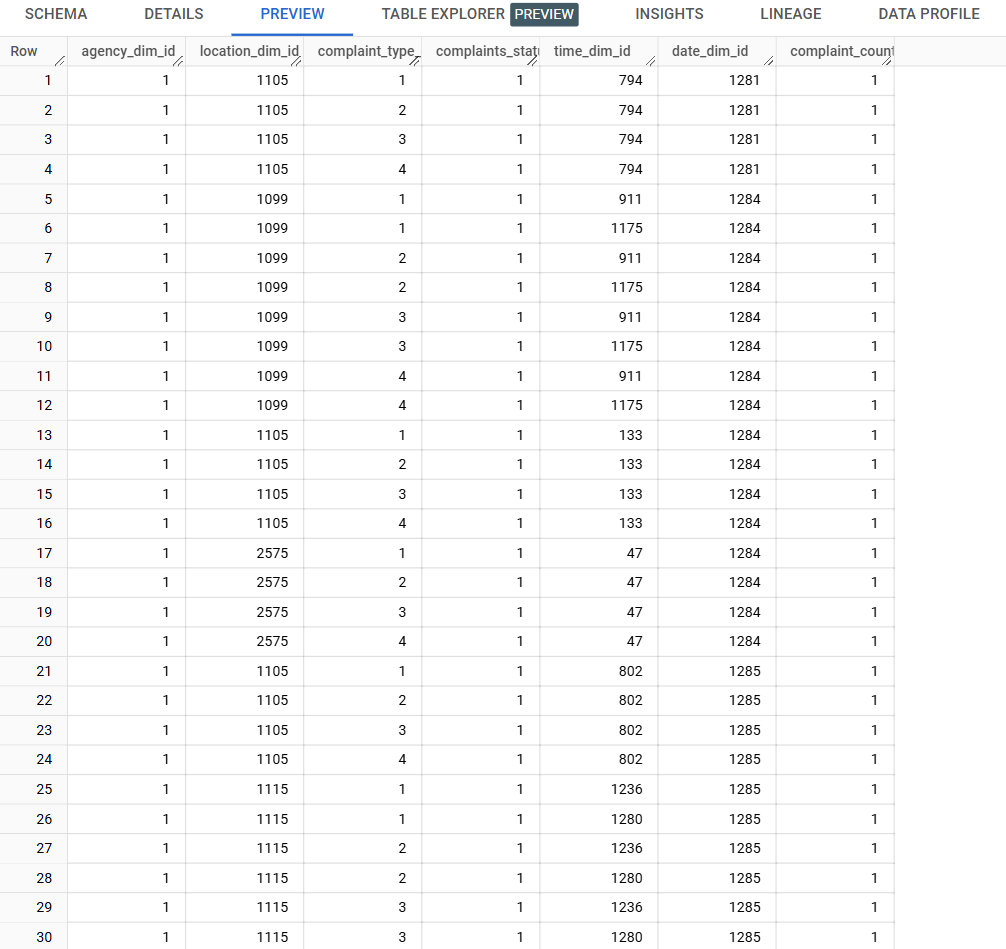
1. **Date Dimension:**



1. **Time Dimension:**



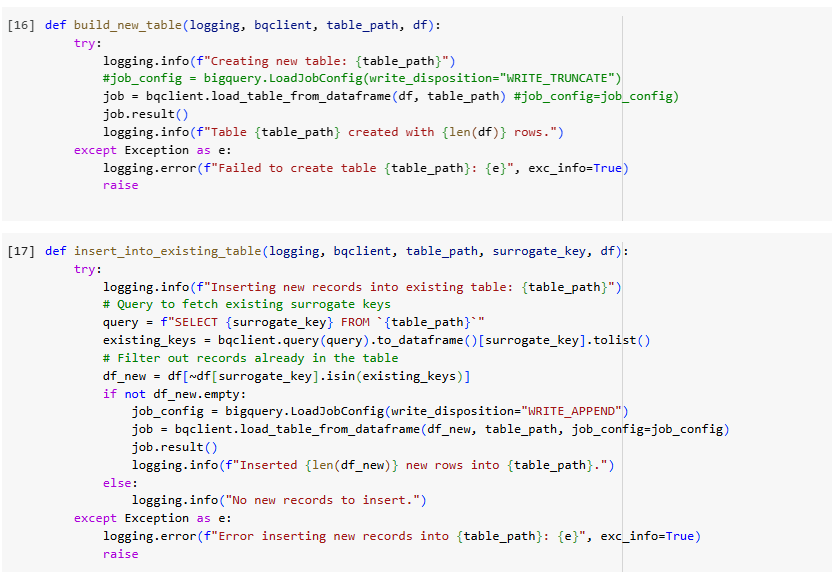
**Taxi Complaints Facts Table:**

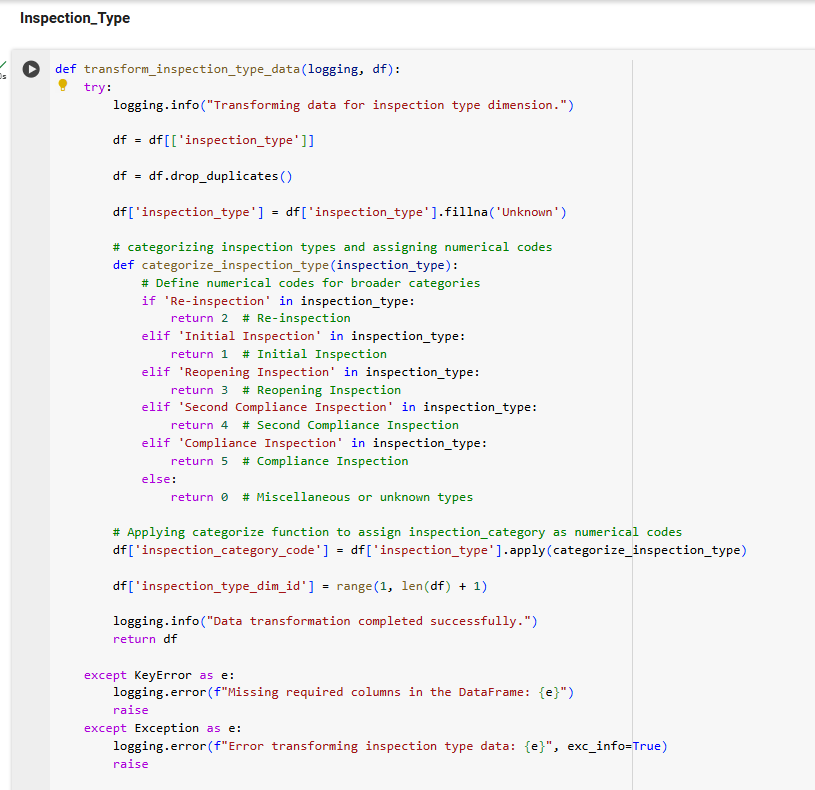


**Restaurant Inspection:**

We began initiating the transformation phase, we developed reusable Python functions to standardize and streamline the ETL process. These functions help us ensure consistency in data handling and allow us to efficiently apply the same transformations across multiple datasets.







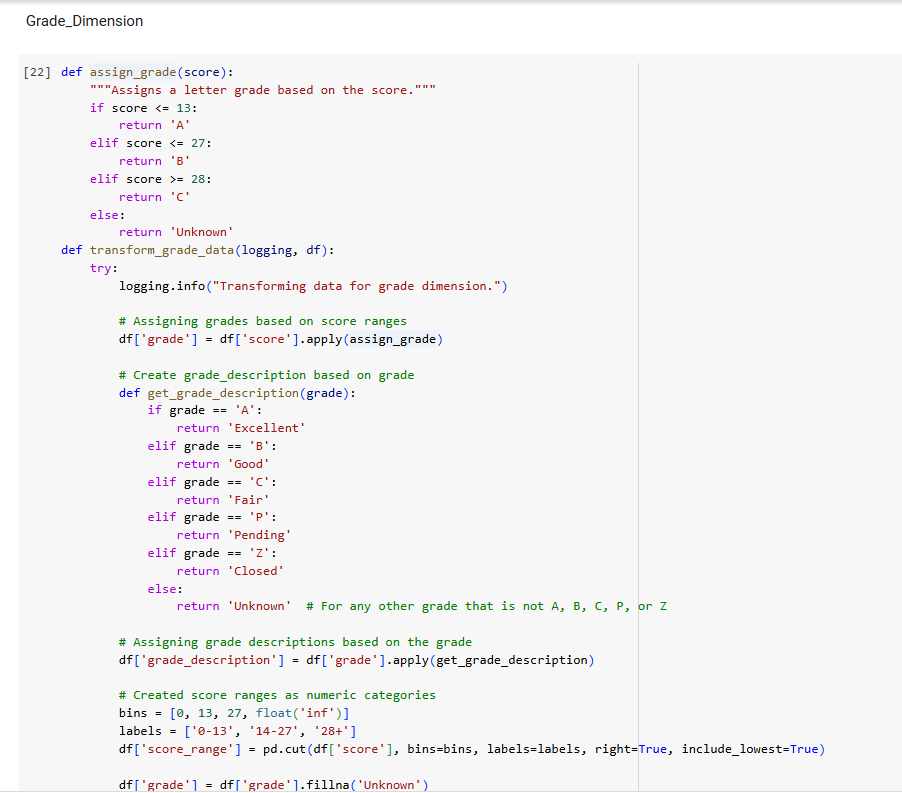
The inspection type field was mapped to corresponding numeric values to create a new field, inspection\_category\_code. Each inspection type was assigned a unique numeric code for clarity and traceability. For instance, an “initial Inspection” was mapped to 1, a “Re-inspection” to 2 and so on.

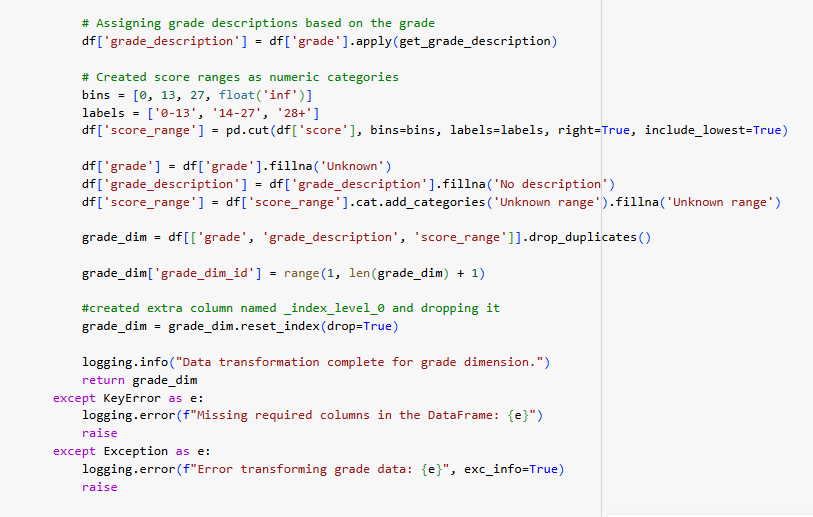




Since some columns had duplicates and some columns were missing, we replaced the missing zipcode with 99999 and street as unknown and added columns named “city” and assigned as “New York”.







The restaurant score follows the below grades:

* *0-13 points = Grade A (Excellent)*
* *14-27 points = Grade B (Good)*
* *28+ points = Grade C (Fair)*
* *Grade P* (Pending) for reinspection, Grade N (Not rated) or Grade Z (Closed) if severe issues are identified

The grade function steps:

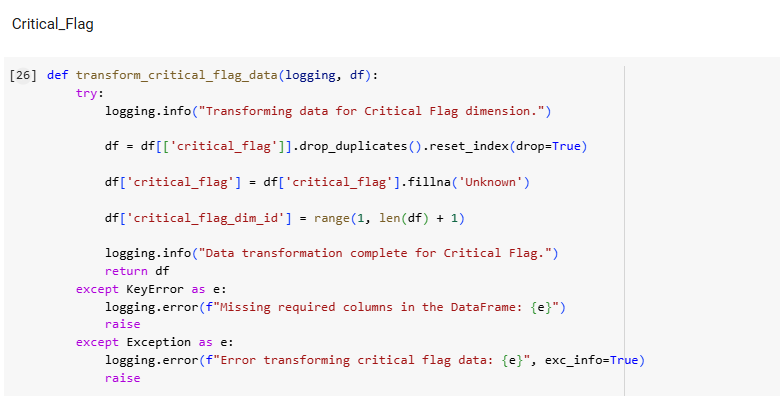
1. Assign Letter Grade: Based on the score, a letter grade is assigned (e.g., a score of 0-13 gets an "A" and follows).
2. Add Grade Description: After assigning the letter grade, a description is added to explain it (e.g., "A" is described as "Excellent" and follows).
3. Define Score Range: Finally, the score range is categorized based on score into three ranges 0-13, 14-27, and 28+ using pd. cut() array elements into different groups.
4. Exclusion of Grades P, Z and N from binning: They are not related to the numerical scoring system as these grades are assigned based on inspection status or conditions and unrelated to point values.

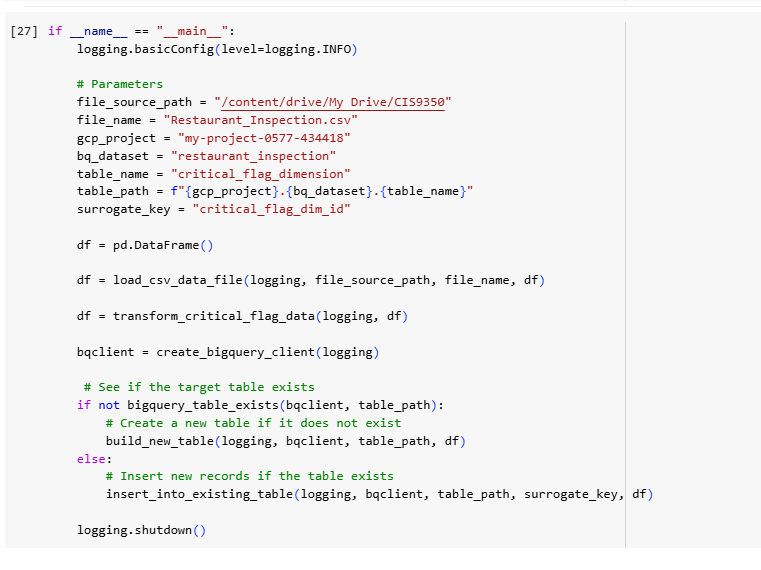
This process helps organize the data clearly, linking scores to grades, descriptions, and ranges without repeating the same letter grades for different records.



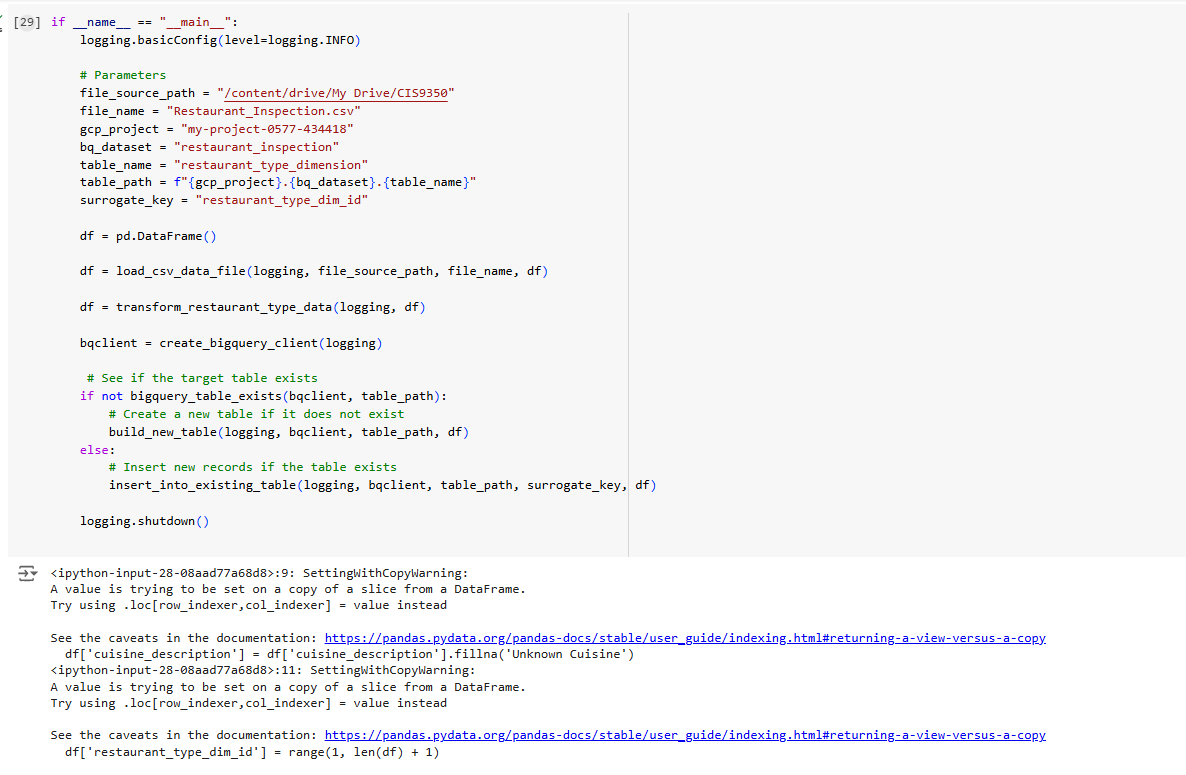






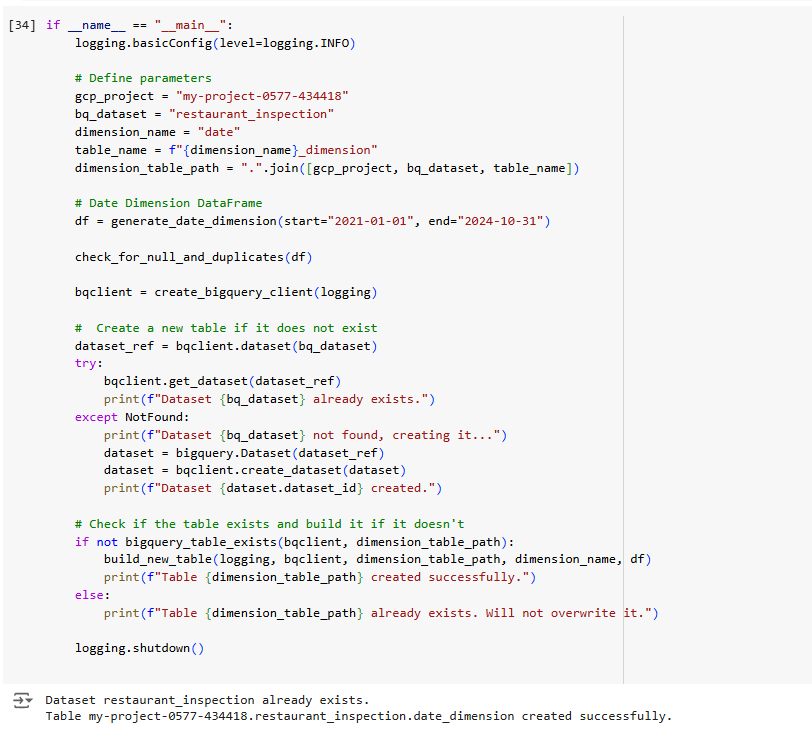




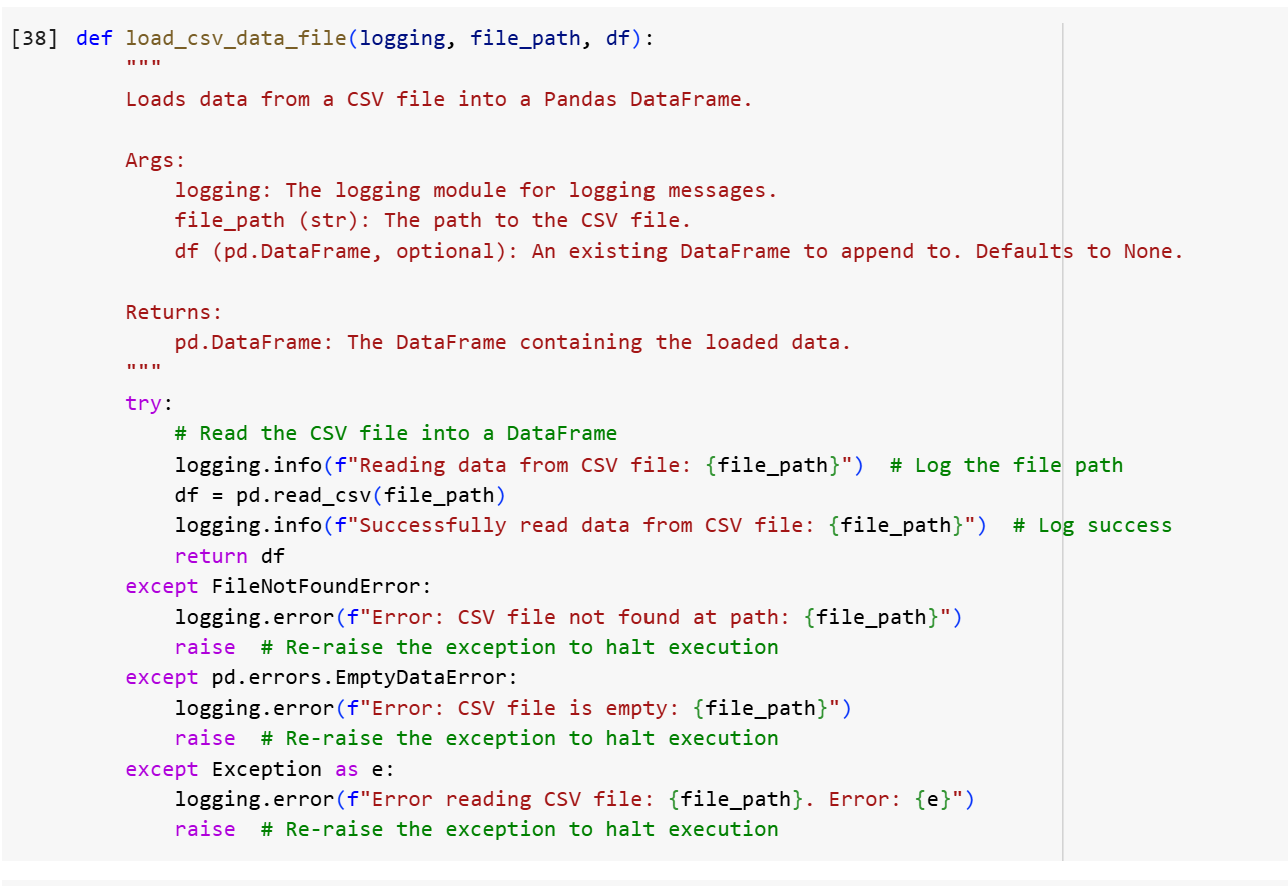




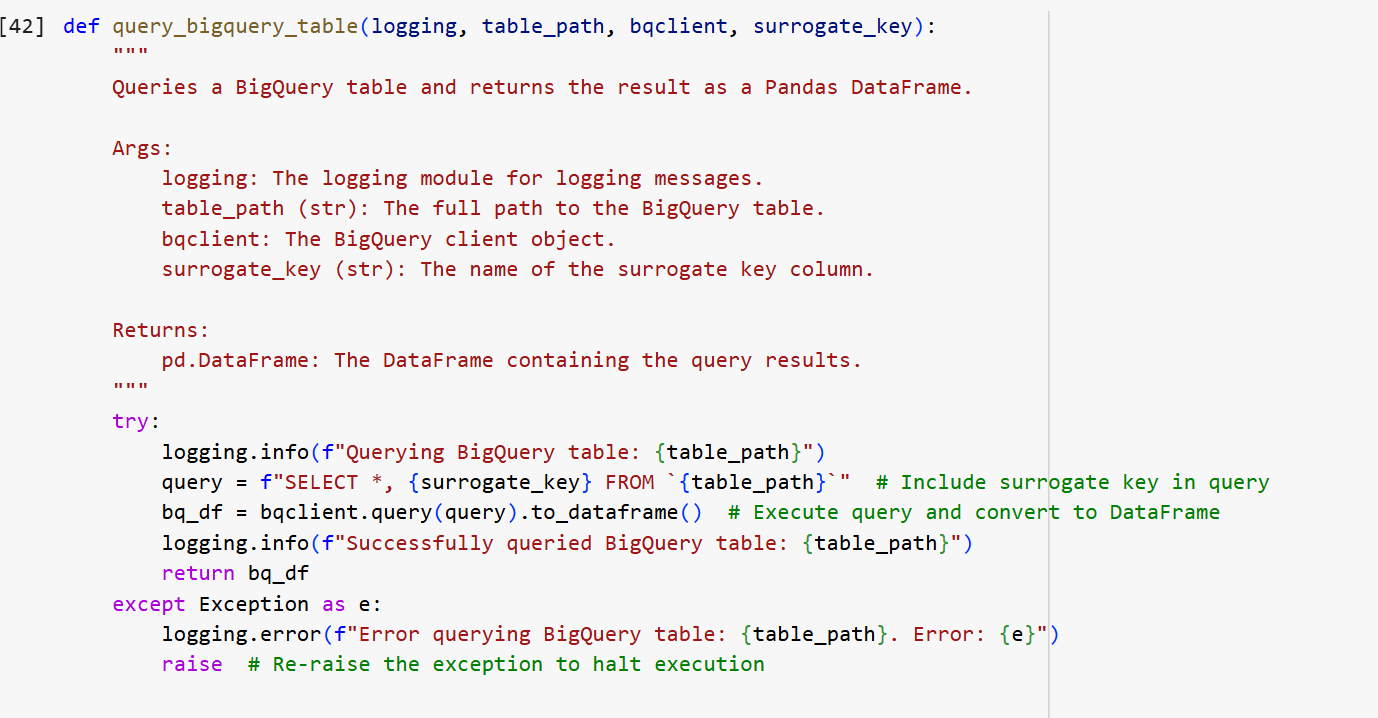


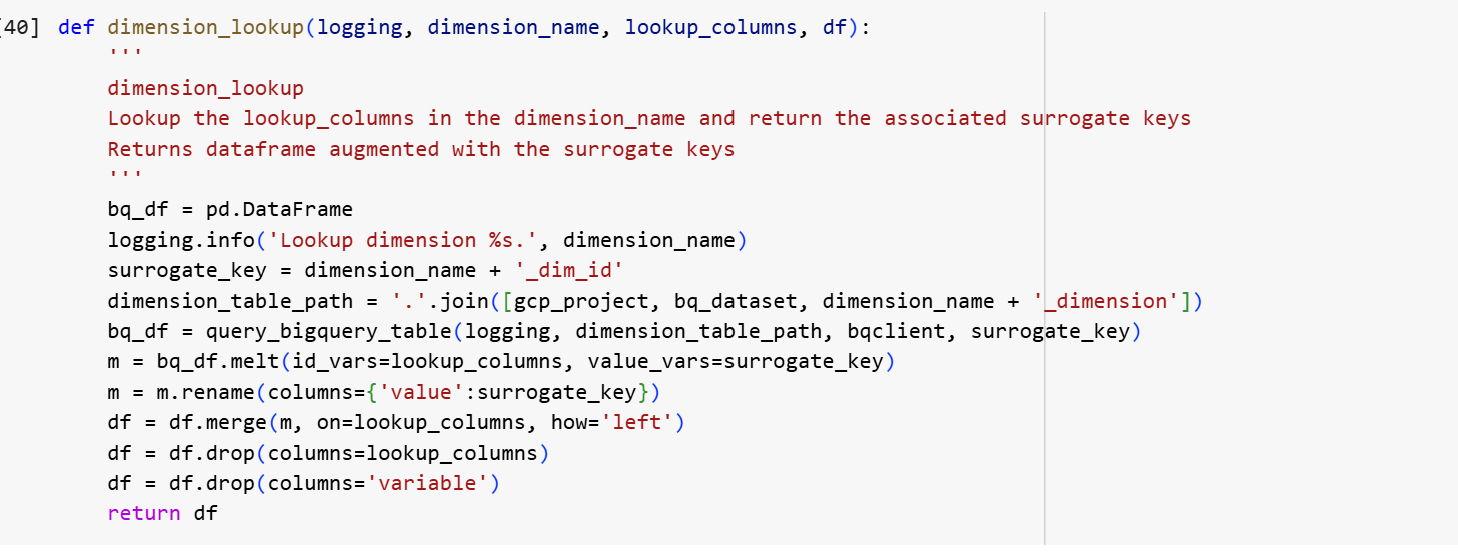


**Creating Fact Table**

****









It generates inspection\_id by setting an index on the DataFrame to assign a unique identifier. The schema is defined for a table that is being loaded, which matches the structure of the data frame.



Idf['inspection\_id'] = df. index: creates a new column to assign values

df = df. group by (surrogate\_keys + ['inspection\_id'])['inspection\_count'].agg('count').reset\_index()

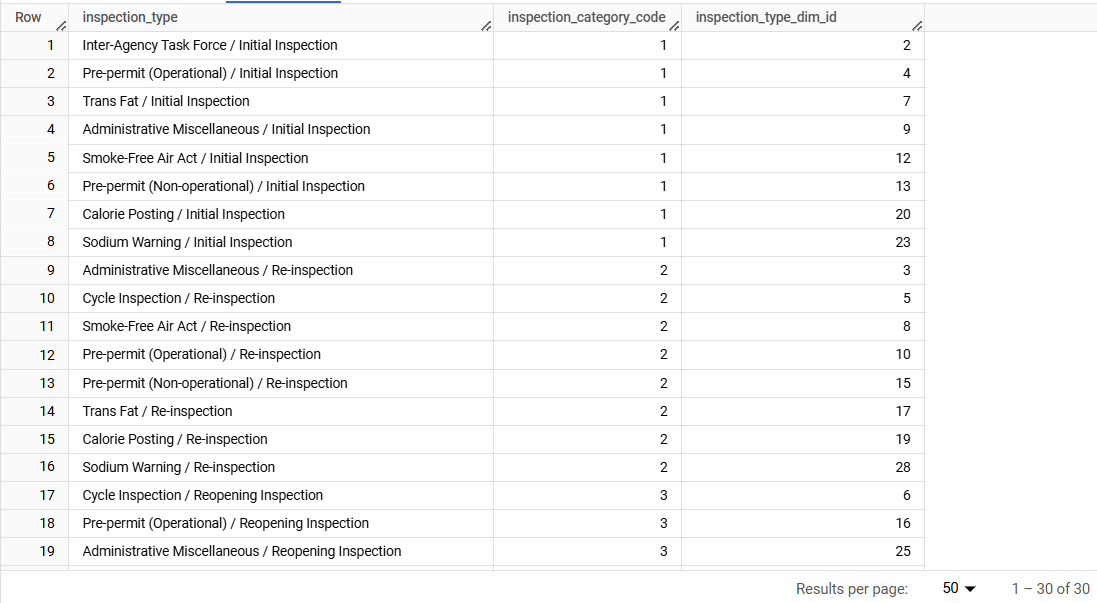
df = df[surrogate\_keys + ['inspection\_id']]

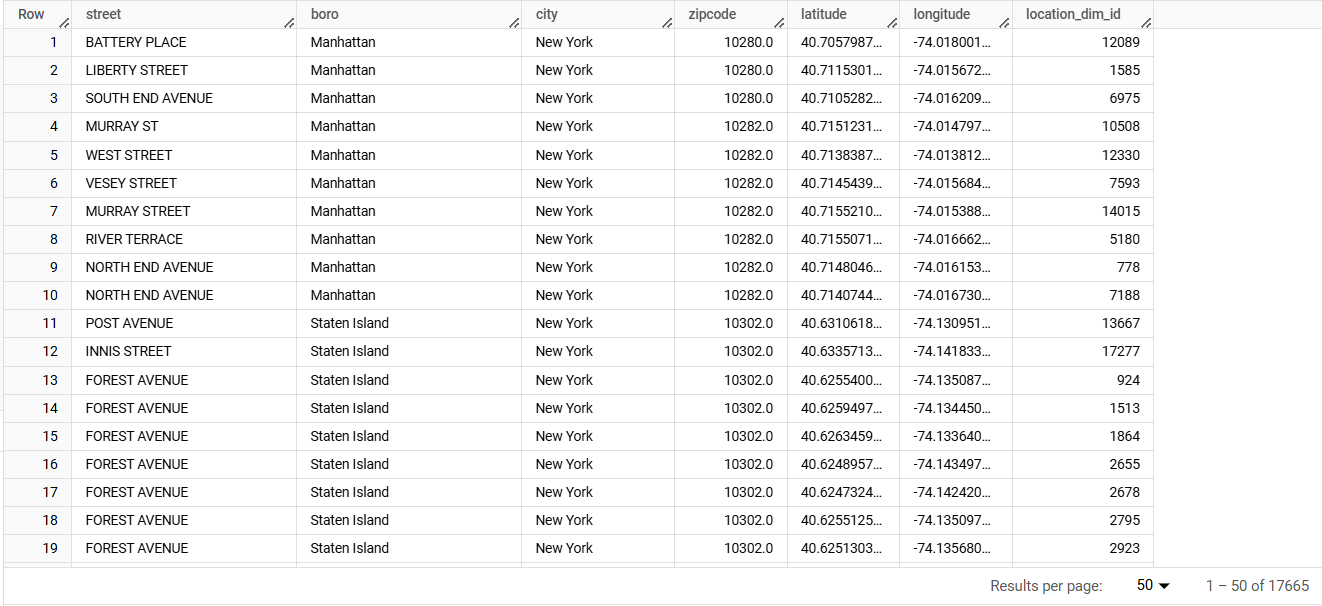
It assigns a constant value of 1 to insepction\_id and then groups the DataFrame by surrogate keys and insepction\_id to count the occurrences of each group. The record represents a single transaction of inspections. This captures a granular snapshot of surrogate keys with dimensions.

**Final Dimension Schema:**

**Restaurant Inspection:**

The following screenshots display the final dimensional schema in Google BigQuery, illustrating each dimension after the ETL process has been successfully completed.

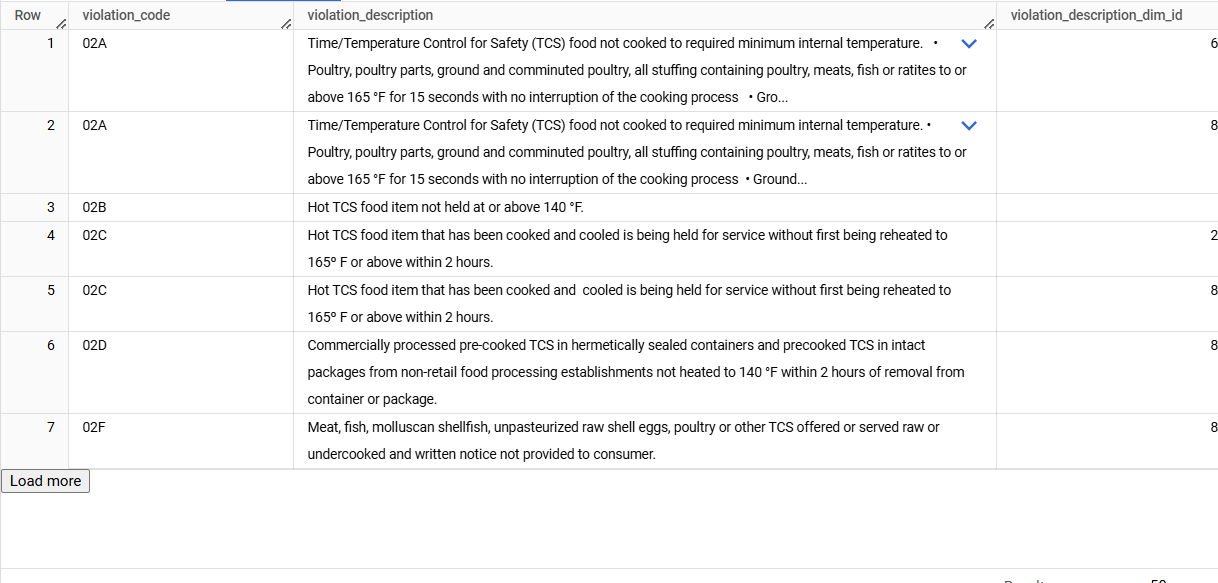
1. **Inspection Type Dimension**
2. **Location Dimension**



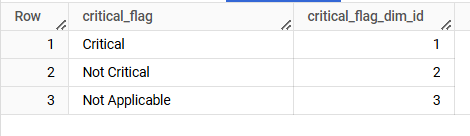
1. **Grade Dimension**



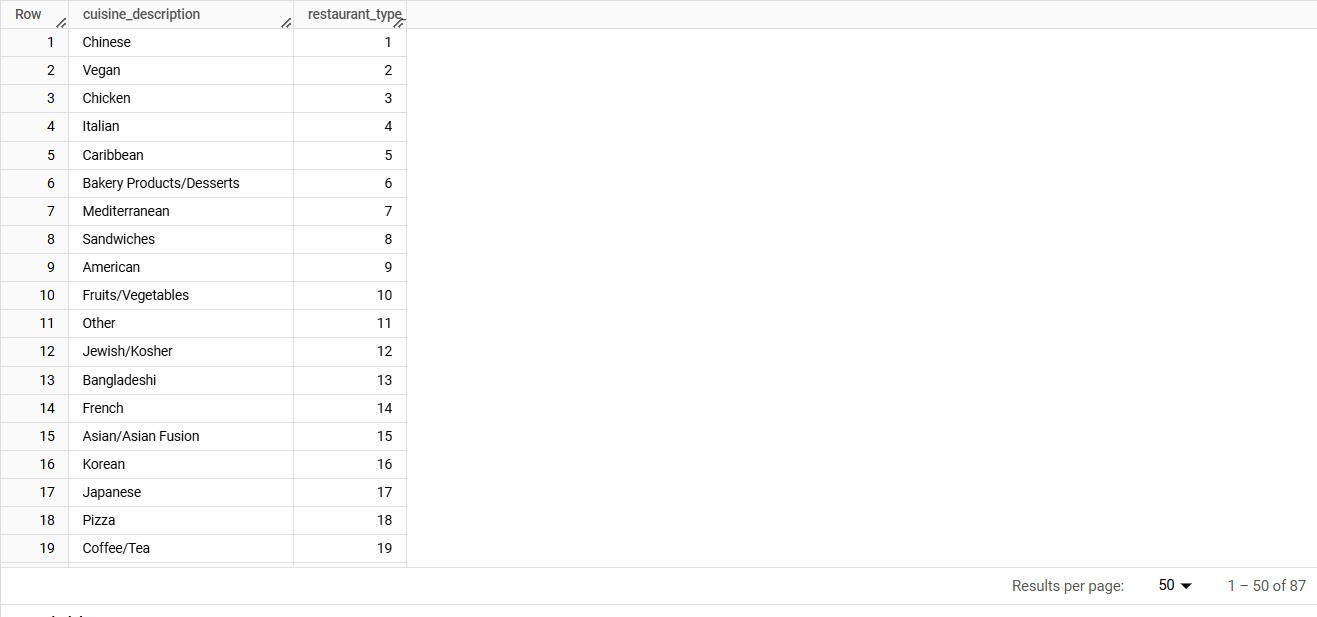
1. **Violation Code Dimensions**



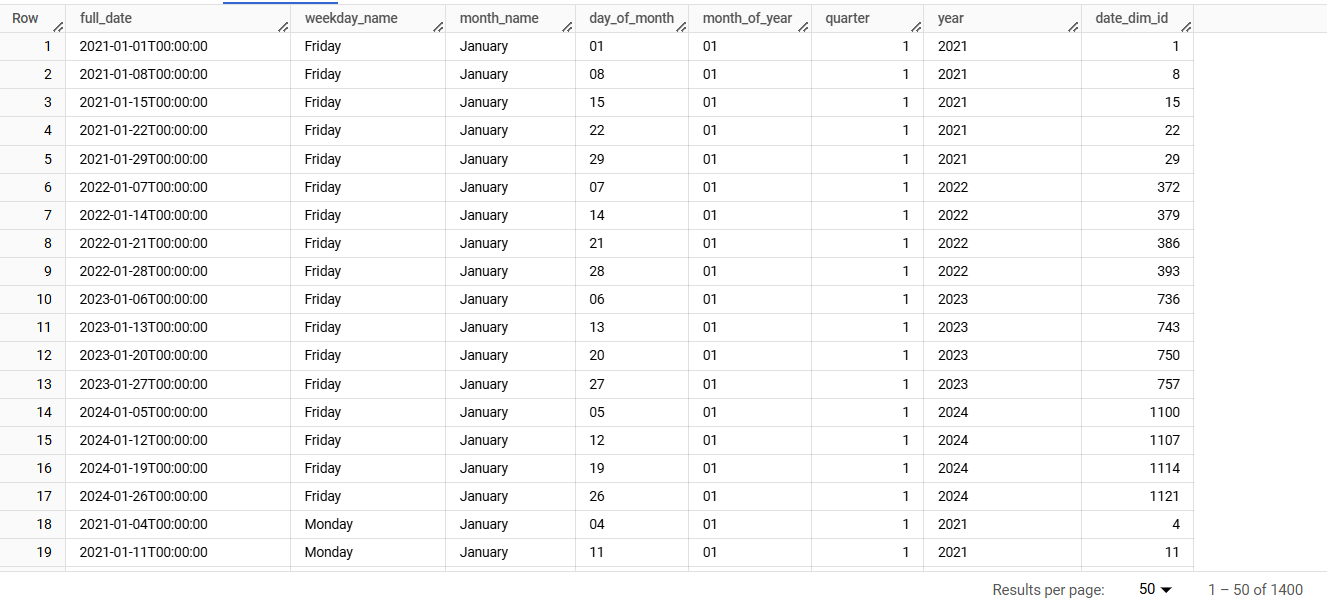
1. **Critical Flag Dimension**



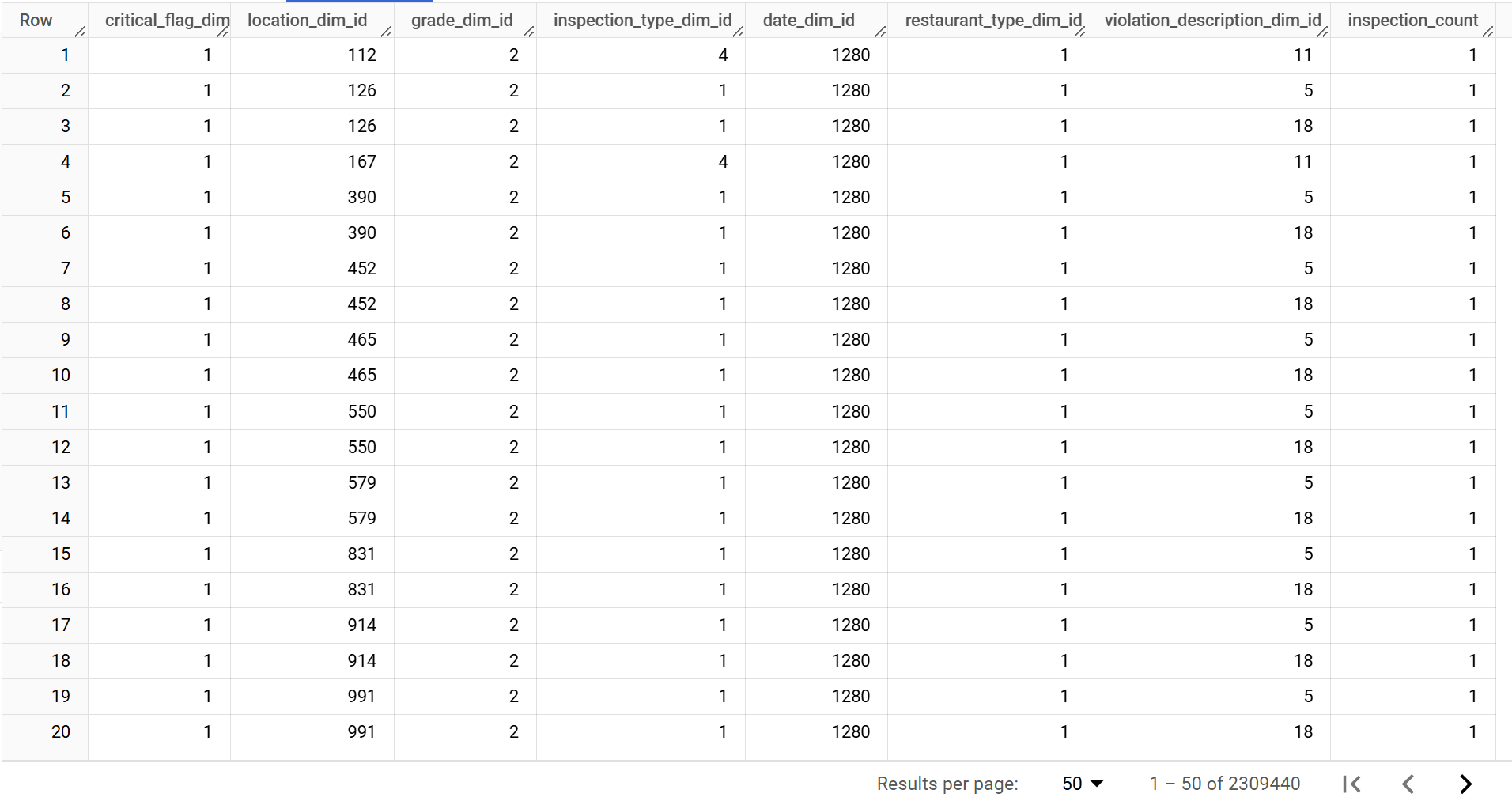
1. **Restaurant Type Dimension**



1. **Date Dimension**



**Restaurant Inspection Fact Table:**



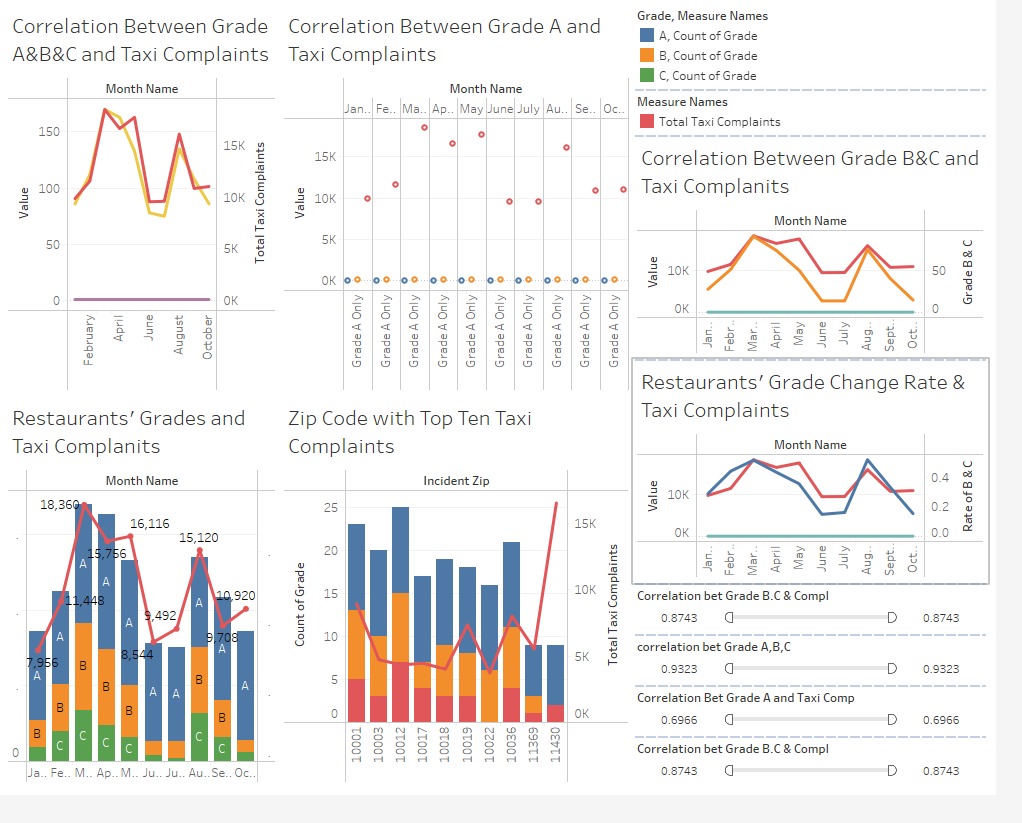
#### **3. Loading**

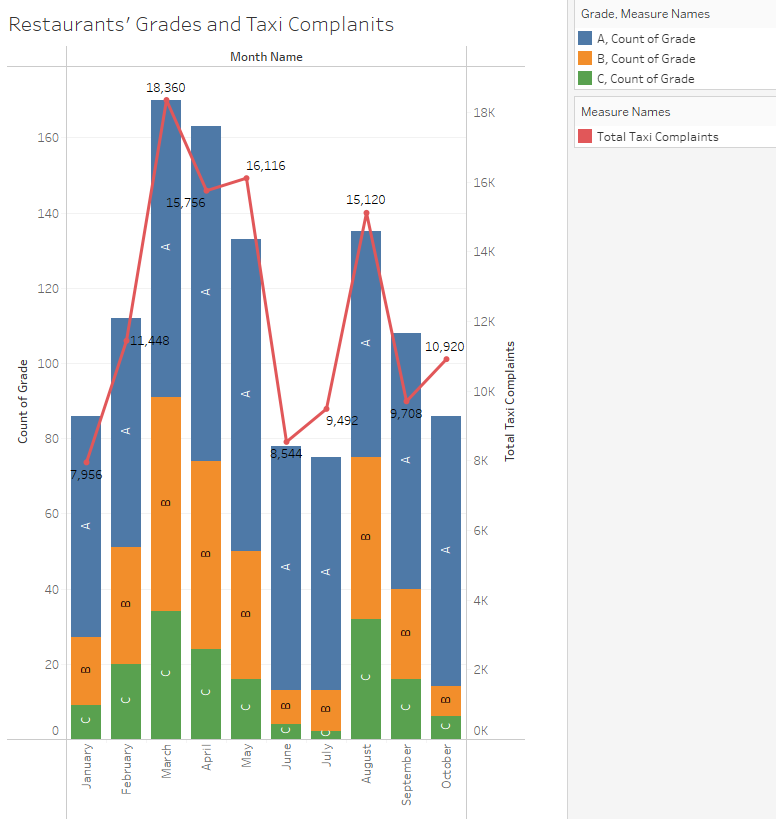
* **Tools Used:** Python and BigQuery on Google Cloud Platform
* The transformed data was loaded into our cloud-based data warehouse. We defined the table structures and loaded the transformed data into our dimensional schema and BigQuery served as the final storage and querying tool for our data warehouse.
* Tables were organized into fact and dimension tables based on the star schema design.
  + **Fact Table:** Stored the aggregated metrics and KPIs.
  + **Dimension Tables:** Contained descriptive information such as restaurant types, inspection grades, and geographic locations.

**Data visualization**

After identifying dimensions and fact tables, we integrated both datasets to generate visualizations. Tableau was selected as the primary visualization tool due to its robust interactive features, user-friendly interface, and ability to connect with Google BigQuery for real-time data analysis.

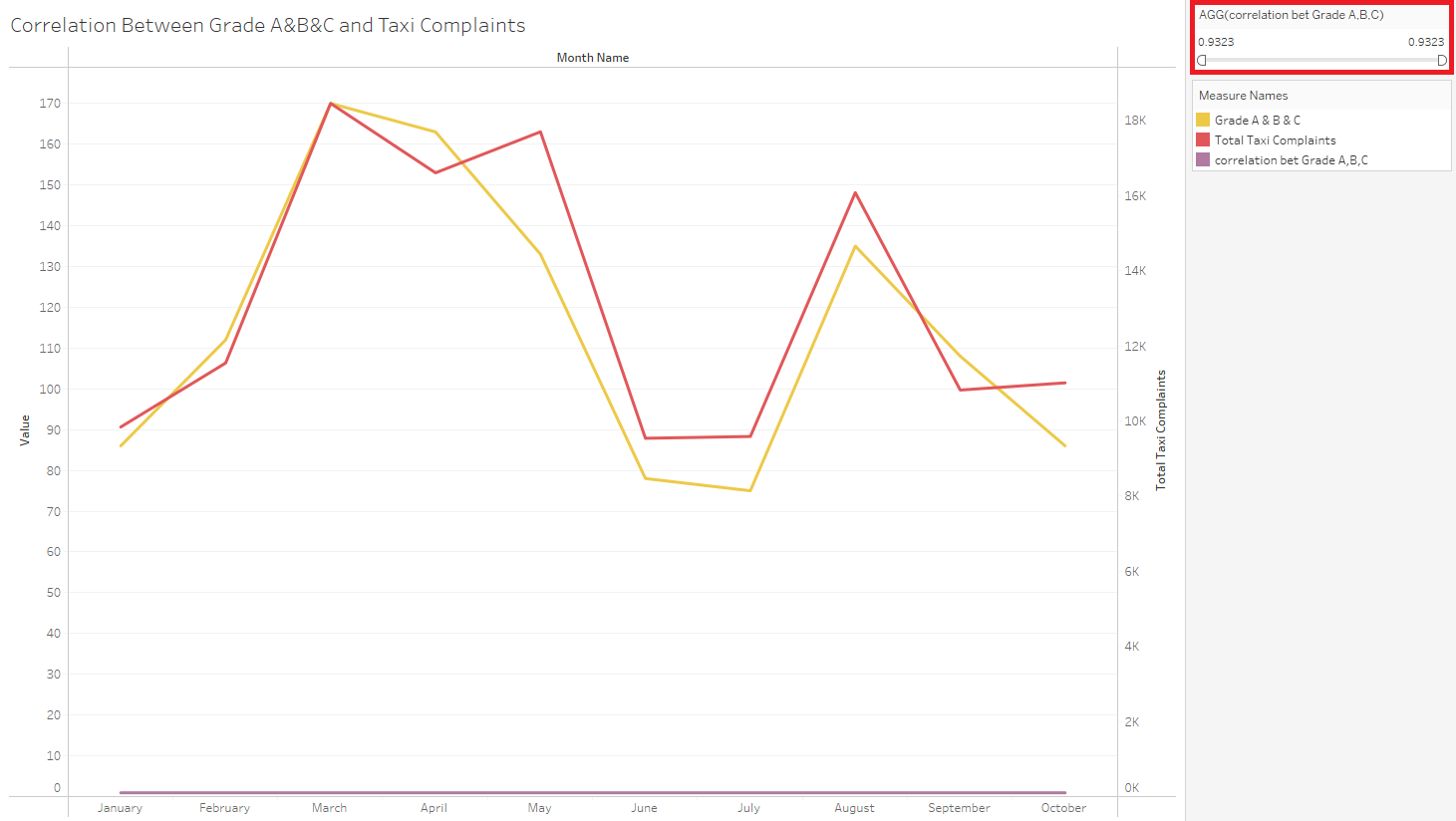
The screenshot below showcases a data visualization dashboard. Below is a detailed explanation of each graph included in the dashboard.

****



For this analysis, we selected the year 2024 to ensure that observations and correlations are based on recent and relevant data only. Using this approach, we can capture current patterns and relationships that reflect current realities and are more actionable.

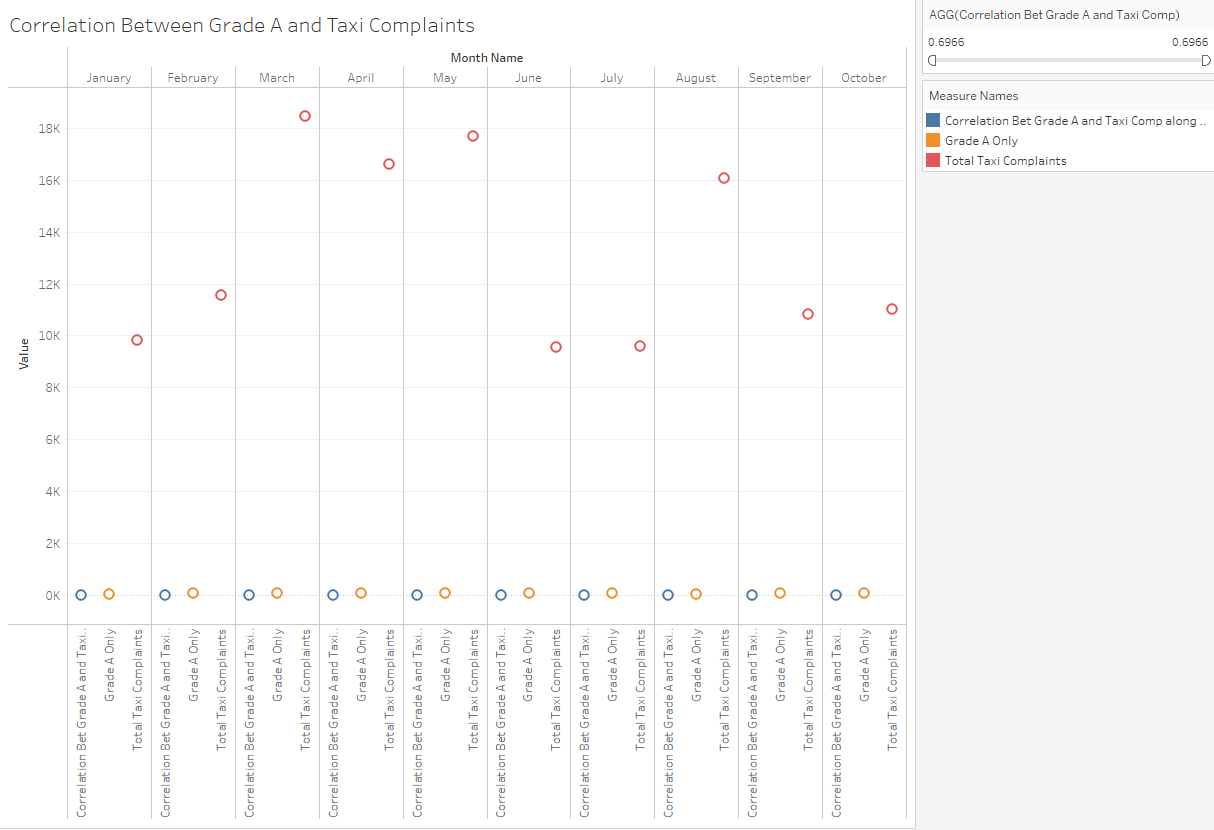
The above graph displays the count of restaurant inspections categorized by their respective grades (A, B, and C) alongside the total number of taxi complaints, organized by month. Observing the trends, it appears that months with an increase in Grade B and C counts often correspond to a rise in taxi complaints. However, this is merely an observation and should not be interpreted as a definitive correlation without further analysis or statistical validation.



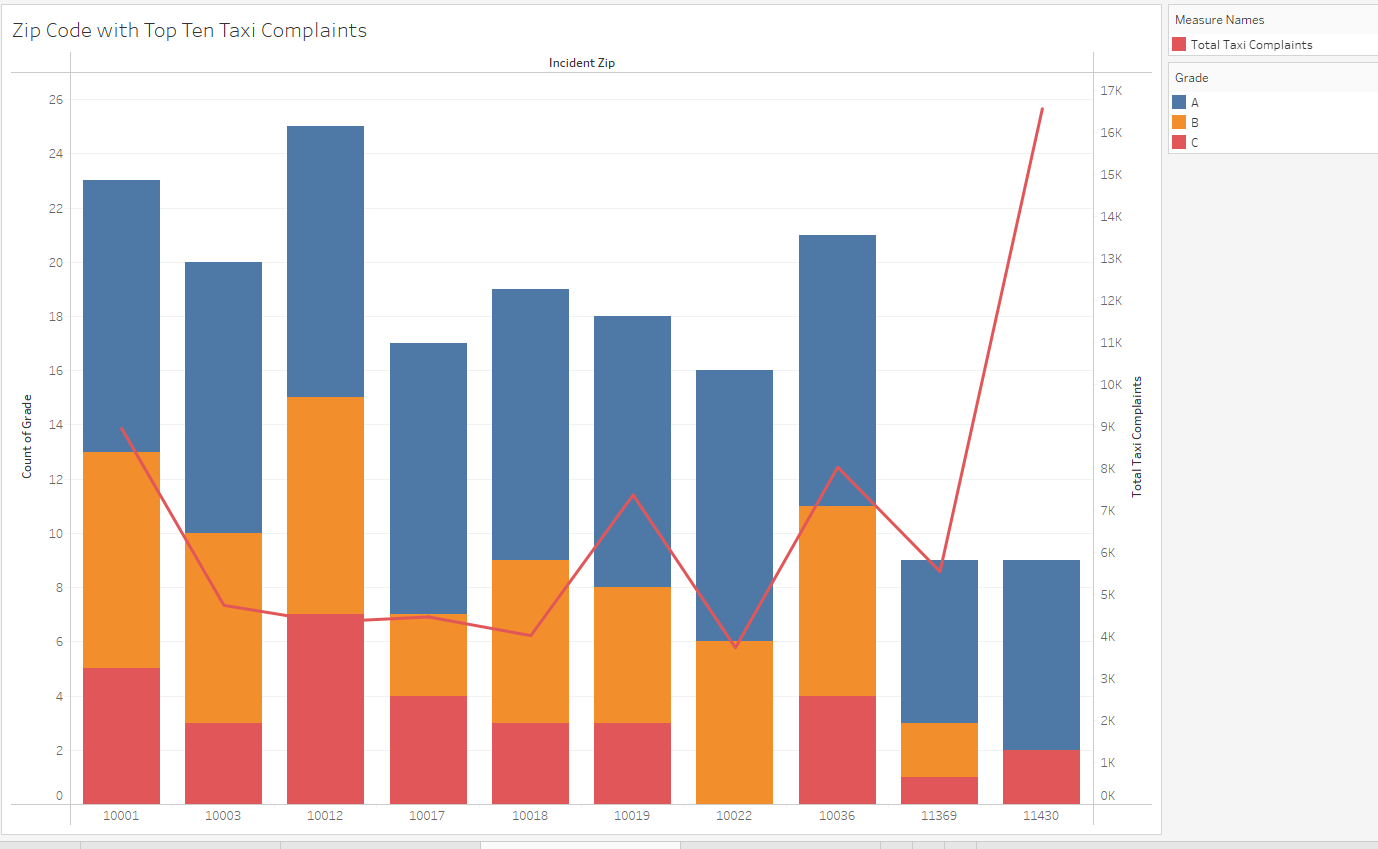
To confirm the relationship, we analyzed the correlation between restaurant grades and taxi complaints. The results show a correlation value of 0.9, indicating a strong positive relationship between neighborhoods with varying restaurant grades and the volume of taxi complaints. This supports the hypothesis that restaurant quality impacts taxi complaints.



Narrowing the focus to neighborhoods with a higher concentration of Grade B and C restaurants, the correlation value is 0.87, still a strong positive relationship. This highlights that areas with more Grade B and C restaurants are likely to experience higher taxi complaints.



When focusing exclusively on Grade A restaurants, the correlation with taxi complaints drops to 0.69, a weaker but still positive relationship. This aligns with the hypothesis that neighborhoods with better-graded restaurants tend to experience fewer taxi complaints.



We wanted to examine the neighborhoods with the highest taxi complaints. In this graph, the zip codes are displayed along the bottom, with the red line representing the number of taxi complaints. If you focus on the bars, which indicate the count of Grade B and C restaurants, you’ll notice a pattern: as the count of Grade B and C restaurants increases, the taxi complaints line also rises. Similarly, when the bars decrease, the line for taxi complaints drops as well. This suggests a potential relationship between the prevalence of Grade B and C restaurants and the volume of taxi complaints.



This above graph examines the ratio of lower grades (B and C) to all restaurant grades over time. This ratio represents the proportion of lower-graded restaurants (B and C) compared to the total number of restaurants, including Grade A, providing a relative measure of the concentration of lower-grade restaurants in each period. The blue line represents this ratio, while the red line shows taxi complaints. Both lines follow similar trends, with a correlation value of 0.87, reinforcing the hypothesis that neighborhoods with a higher proportion of lower-grade restaurants tend to have more taxi complaints.

### 

### **Descriptions of Tools Used**

Our project leveraged various tools and technologies to design, implement, and analyze the data warehouse. Below is an overview of the tools used across different milestones of the project:

Databases

Google BigQuery: Used as the primary data warehouse to efficiently store, manage, and query large datasets. It enabled seamless integration of taxi complaints and restaurant inspection data.

Lucidchart: Designed dimensional models, star schemas, and integrated schema diagrams.

ETL (Extract, Transform, Load)

Python: Utilized for data profiling, transformation, and cleaning tasks. Python libraries like Pandas were used to handle missing values, format data, and apply business logic.

Google Colab: Used for running ETL scripts and generating surrogate keys for all the dimensional tables.

Visualization Tools

Tableau: Developed KPI dashboards and visualizations to analyze trends and relationships between taxi complaints and restaurant inspections.

**Conclusion**

a) Software and Database Tools Used

* Lucidchart: The tool was used to visualize and design the dimensional models and schema diagrams.
* Google BigQuery: It was the primary database for storing and querying large datasets efficiently.
* Python: Mainly used for ETL processes, data profiling, and transformations with libraries like Pandas.
* Tableau: Created interactive dashboards for visualizing main KPIs and trends.
* Google Colab: Used to execute Python scripts and manage ETL processes collaboratively.

b) Team’s Experience with the Project

Challenges:

* The major challenges we faced were when doing the ETL process, particularly understanding DBT’s setup and capabilities. Switching to Python simplified the task but required significant learning to optimize transformations.
* Integrating two distinct datasets (taxi complaints and restaurant inspections) posed difficulties in aligning dimensions and handling missing data.

Easiest Step:

Building dimensional models in Lucidchart was relatively seamless once the schema design was finalized.

Key Learnings:

* We gained insights into advanced data warehouse concepts, including star schemas and surrogate key creation.
* Mastered each step of ETL techniques and handled large datasets efficiently.
* Realized the importance of clear communication amongst groups and collaboration tools for managing group tasks.

**What We Would Do Differently:**

* Allocate more time for ETL tool exploration as it was the most challenging part to avoid mid-project changes.
* Establish stricter deadlines for all milestones to ensure better time management.
* Conduct a pilot test with smaller datasets to identify integration issues earlier.

c) Proposed Benefits Realization

This project provides actionable insights by correlating taxi complaints and restaurant inspections, enabling targeted interventions to improve public health and service quality. It supports efficient resource allocation for regulatory agencies, educates the public, and lays a foundation for broader application and future research.

d) Final Comments and Conclusions

This project has been a valuable learning experience for us. It has equipped the whole team with collaborative strategies and technical skills. The results also demonstrate the potential for integrated datasets to address real-world issues comprehensively. While challenges were encountered, the project’s outcomes validate the effort and showcase the significance of data warehousing in decision-making processes.

### 

### 

### 

### **References**

1. **NYC Open Data**Dataset: Taxi Complaints  
   Source:<https://data.cityofnewyork.us/Social-Services/311-Service-Requests-from-2010-to-Present/erm2-nwe9/about_data>
2. **NYC Open Data**Dataset: Restaurant Inspections  
   Source:<https://data.cityofnewyork.us/Health/DOHMH-New-York-City-Restaurant-Inspection-Results/43nn-pn8j/about_data>
3. **Professor's Code**

Code provided by Professor Ramah Al Balaw for extracting and transforming inspection data, which was used to preprocess the Restaurant Inspections dataset. The code was not available on a public website.

1. **GitHub Repository: NYC 311 Complaints Data Warehouse**

The repository provided by [GitHub User: jli82](https://github.com/jli82) was used as a guide to create the fact table for restaurant inspections. https://github.com/jli82/nyc-311-complaints-data-warehouse/blob/main/Final%20Project%20Report.pdf

Date Accessed: 11/15/2024