**ETL Project**

**Restaurant inspections vs rat sightings**

**Team**: Alla Zbirun, Hernan Hernandez, Ryan Michel

**Executive Summary**: Rats in NYC persist to be an issue. As residents of NY, we volunteered to investigate this issue and determine if there is a correlation between restaurant inspections by NY Health department and rat sightings in the area. This document summarizes step by set process the team performed from extraction of the data, through data clean up and normalization, resulting in creation of databases finally uploaded into PostgreSQL for convenient SQL quiring.

**Extraction**: For this project we extracted data from three different sources -

1. NYC Open Data (<https://data.cityofnewyork.us/Health/DOHMH-New-York-City-Restaurant-Inspection-Results/43nn-pn8j/data>) – we were able to download csv files of data containing information on violations determined under inspection conducted during preceding three years for restaurants and college cafeterias
2. Kaggle.com ( <https://www.kaggle.com/new-york-city/nyc-rat-sightings>) – we were able to download csv files on rats sightings in the NYC areas.
3. Yelp.com ( <https://www.yelp.com/dataset>) – we were able to download a JASON file which contained data of restaurant credentials, review ratings and much more info on a global scale.

**Transformation**: The data we received was more than we needed and held number of inconsistencies or missing information. In order to normalize the data we performed following steps in Jupyter notebook to extract only sought-after elements and wrote it nto a databases –

1. NYC Open Data - The restaurant dataset consisted of 394,890 rows with 26 columns. In viewing the dataset we noted obvious columns that we removed as the data had nothing to do with the focus of our project. In total we removed 8 columns. We next looked for the sum of nulls in each column. Out of the 18 columns remaining, only 5 didn't have nulls. We decided to drop the rows that had null values. Lastly, the dataset contained 7 rows with duplications and we removed all of them. After the cleaning, the final dataset contains 193,397 rows of the restaurant inspection data.
2. Kaggle data- There were a total of 52 columns found in the original dataset. Since the goal was to see if there was a relationship with health grade ratings and number of rat findings, we only preserved a zipcode column. Further, we used a groupby function to determine number of rat findings by zipcode; which we then saved the total number of observations as a column in the new dataframe. We removed all null values and one entry that listed incomplete zipcode. The final dataset contains 186 ros of rat sightings by zipcode.

Yelp data – Dataset file was downloaded in TAR archive format. In order to extract data files we downloaded/utilized 7-Zip, which decompressed the file and extracted contents of the archive into a folder. Out of the data available (see <https://www.yelp.com/dataset/documentation/main> for details) we chose business.json file as it contained most relevant information (i.e. - address, restaurant category, ratings, lat & long, and review counts). After detailed review of the file ocntents we dropped number of insignificant columns (attributes, hours, is\_open). We re-ordered columns for more convenient read of the data by sending column headerst o a list; the changing the order of the apparentce and reflecting it in the dataframe.; Furhter, we groupped data by states and noted that the dataset is not exclusive to USA and contains only 22 entries for NYS. We extracted NYS data and were able to determine that this set contained only one entry for NYC which was a beauty salon. At this point we decided to drop this dataset from the analysis.

1. **Load**: After cleaning and normalizing data during transformation step, we generated following dataframes to be uploaded into pgAdmin –
   1. Based on NYC OpenData set –
      1. Violations – holds a list of violations and their descriptions
      2. Boro\_Zip – holds a list of zipcodes in the dataset with relative bourogh name and inspection claim id
      3. Restaurant – holds a list of restaurants in the database including name of the business, borough, zipcode, type of cusine and grade.
      4. Inspections – holds a list of inspection claim ids, violation does, zip-codes and inspection date
      5. Geo-Location – holds a list of inspection claim ids with relevant latitude and longitude of the restaurant.
   2. Rats – based on the rat sighting analysis which shows zipcode and associated rat sightings.

ended up with three data bases that were joined at “”” and loaded into PostgreSQL where we created number tables (as detailed in ERD table) for convenience of segregation, maintenance and quiring process.