**ETL Case Study Project** – Restaurant inspections vs rat sightings vs customer reviews

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

**Overview**: For this project we extracted data from three different sources, normalized it in Jupyter notebook, extracted only sought-after elements and wrote it into a user-friendly databases. Cleaned up data was loaded into pgAdmin for convenient SQL quiring.

**Extraction**: 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 restaurant data credential review ratings, and much more info.

**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 –

1. NYC Open Data
2. Kaggle data- There were a total of 52 columns found in the original dataset. Then only relevant columns were extracted and entered into a new dataframe to perform data cleaning later on. A simple search revealed null values present in the dataset. A simple function was used to remove these, bringing the total number of rows to 92311 entries. Each row represents a rat finding by the DOHMH.
3. 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 catefory, ratings, lat & long, and review counts). Further we a) dropped number of insignificant columns (attributes, hours, is\_open); b) re-ordered columns; c) checked for absent/nul data fields; d) wrote new df into csv file.

**Load**: After cleaning and normalizing data during transformation step, we ended up with three data bases that were joined at “”” and loaded into pgAdmin where we created number tables (as detailed in ERD table) for convenience of segregation, maintenance and quiring process.