ETL PROJECT

**Toronto New Restaurant Location Analysis**

Submitted By: Dong Yi Kim | Saif Gorges | Saloni Gupta | Sooyeon Kim

Date: December 17th, 2020

horizontal line

# 

# Introduction

Where to open a new restaurant? Choosing a new restaurant location is the most important, at the same time, the most difficult decision throughout the whole process. This project is to look at restaurant data in Toronto as well as ethnicity demographics, neighbourhood average income/crime. The goal of this new layer of analysis will be to help new restaurant owners decide as to where the best placement of a new restaurant could be. Our analysis will consider ethnicity, local competition, income and crime per neighborhood to help determine whether a restaurant could potentially be profitable or not each neighborhood.

## Table of Contents

## Data Modeling / ERD

## Data Extraction / Data Transformation

* 1. Datasets Sources
  2. Toronto Restaurants Data - Extraction & Transformation
  3. Toronto Neighbourhood Data - Extraction & Transformation
  4. Toronto Ethnicity Data - Extraction & Transformation
  5. Toronto Crime Data - Extraction & Transformation
  6. Restaurants Rating & Review Data from Yelp API - Extraction & Transformation
  7. Toronto Income Data - Extraction & Transformation

## Data Load to SQL Database

* 1. Load onto SQL
  2. Final Database
     1. Restaurant Table
     2. Neighbourhood\_Restaurant Table
     3. Neighbourhood Table
     4. Income Table
     5. Crime Table
     6. Ethnicity Table
     7. Yelp\_Rating Table
  3. Join Tables (SQL queries)
     1. Join crime, income, ethnicity, neighbourhood\_restaurnant tables
     2. Join restaurant and yelp\_ratings tables

## Sample Analysis

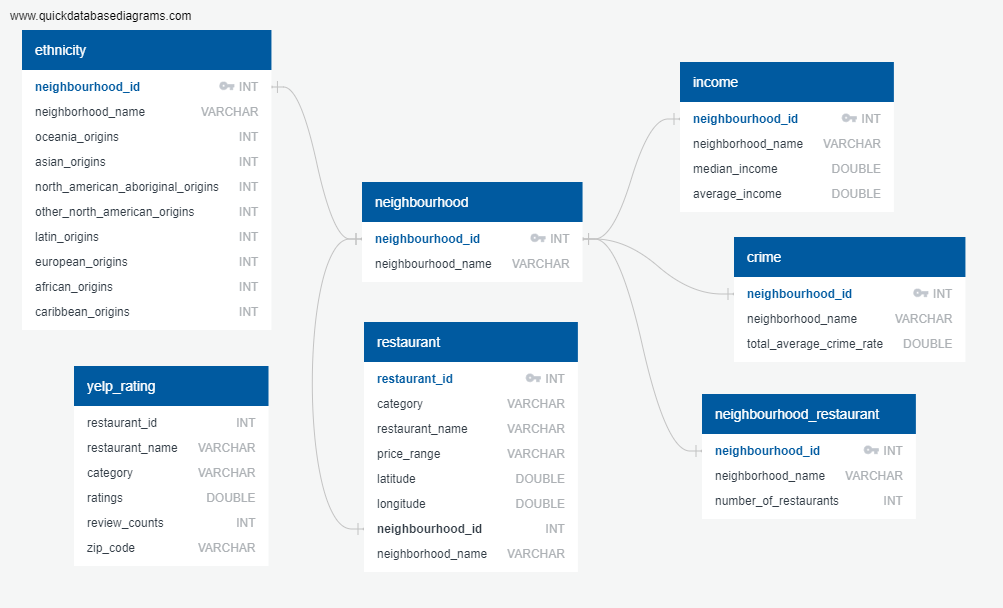
## Discussion

4.1 Limitations

4.2 Next Steps

# Data Modeling

ERD (Entity Relationship Diagram) of the tables was sketched out using a tool called 'Quick DBD.'



# Data Extraction / Data Transformation

### 2.1 Datasets Sources:

In this project we extracted, transformed, and loaded these datasets:

* Toronto Neighbourhood Data - Toronto City Open Data
* Toronto Neighbourhood Income - Toronto City Open Data
* Toronto Crime Data - Toronto City Open Data
* Toronto Ethnicity Data - Toronto City Open Data Json API
* Toronto Restaurant Data - Kaggle
* Restaurant Ratings & number of reviews - Yelp API

The extraction of these datasets were completed using two methods; CSV’s and API’s. It was important to choose datasets from reliable sources such as Kaggle, ‘Yelp API’, ‘Toronto Police API portal’ and the ‘City of Toronto Open API’. These three sources were all credible and trustworthy sources of data for our analysis. The primary key to almost every dataset was the neighborhood. Since our analysis primarily focuses on the behaviour of certain variables (crime, ethnicity, income, etc..) it was imperative that every dataset we sourced was in some way linkable to a neighborhood. The methods in which we chose to link to a neighborhood will be further discussed in detail below.

### 2.2 Toronto Restaurant Data

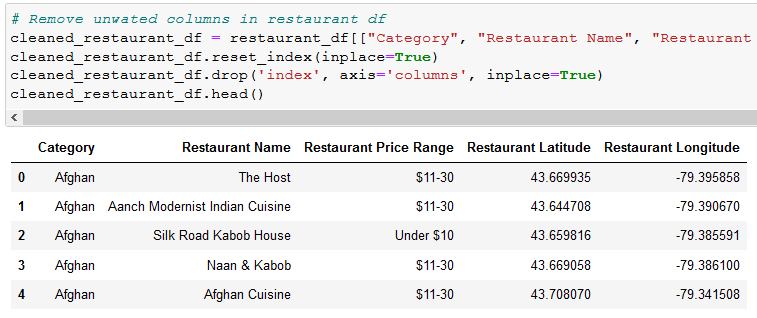
*Extract*

The restaurant data that we sourced from Kaggle came in a CSV format. From the screenshot below, we can see that we get all of the data we need for this analysis except for the neighborhood in which the restaurants belong to.

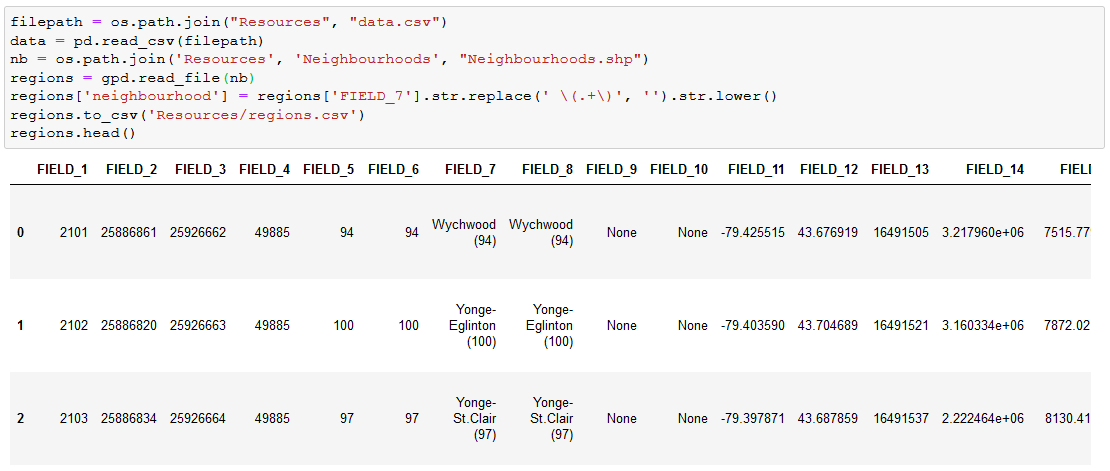


*Transform*

Initial data cleanup consisted of removing any unwanted columns and removing NaN rows.



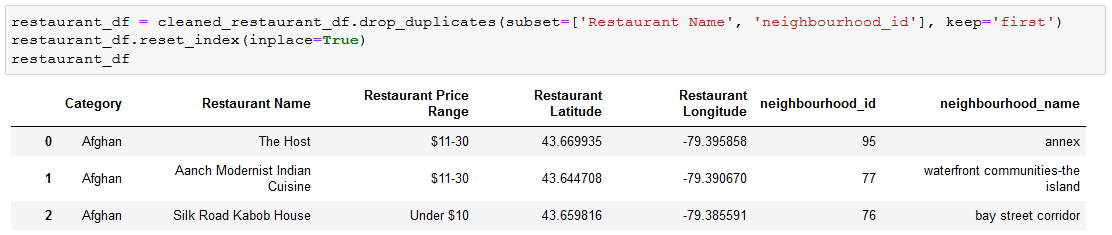
In order for us to connect this dataset to our main database, we need to identify which neighborhood each restaurant belongs to. This issue can be solved using GeoPandas. From the screenshots below, we load shapes file into Pandas so that we can get the borders of each neighborhood based on lat/long.



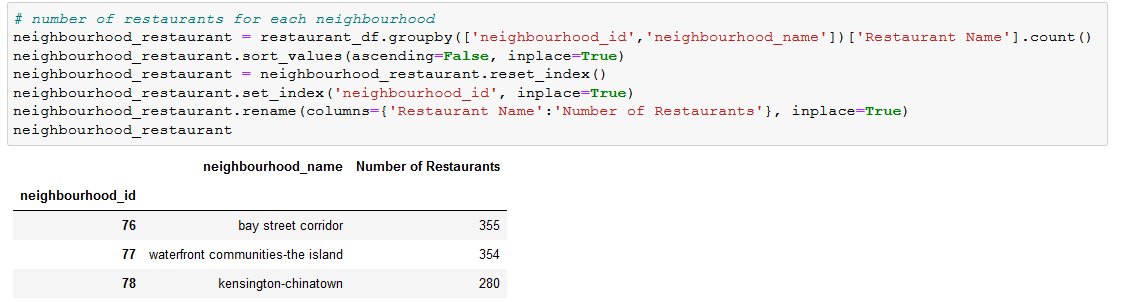
We then create two new columns (Neighbourhood ID and Name) and link each restaurant based on lat/long with the above Geopandas table.



Many of the duplicate values included a restaurant having more than one category, or multiple locations but same name. We dropped the duplicated that included having multiple categories.



Finally, the total number of restaurants in each neighborhood were aggregated to come up with a cleaned list of the total number of restaurants per neighborhood.



### 2.3 Toronto Neighbourhood Data

Toronto Neighbourhood Data was the base of other tables as the objective was to look at the difference between neighbourhoods. The final neighbourhood table has only two columns: ‘neighbourhood\_id’ and the other is ‘neighbourhood\_name’, and both can be primary keys as the values are unique.

*Extract*

The Toronto Neighbourhood dataset was found in Toronto City Open Data. The raw dataset was a csv file and it has many columns including ‘AREA\_SHORT\_CODE’, ‘AREA\_NAME’, ‘LATITUDE’, ‘LONGITUDE’, ‘geometry’ and so on. This dataset was imported by using Pandas library.

*Transform*

Firstly, the neighbourhood names had to be edited using Python string method replace(), as the raw data neighbourhood name had unrequired strings. For example, the raw data had a neighbourhood name, Yonge-Eglinton (100), from here ‘(100)’ was not required. With the code below, ‘neighbourhood\_name’ column was added to the dataframe. 

After that, the unwanted columns were dropped, renamed some columns based on our ERD, and finally set ‘neighbourhood\_id’ as an index of the dataframe.

### 

### 

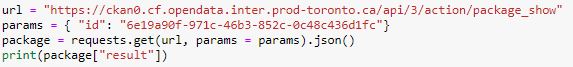
### 2.4 Toronto Ethnicity Data

Toronto Neighborhood profiles is a free open source data which is available to the public in many different file formats (CSV, XML and JSON). Using this data, our goal here was to extract the information about the ethnicity distribution by neighborhood in Toronto. The steps that were taken to achieve this goal will be explained in more details below.

*Extract*

Step1

As response to the API call, the ethnicity data from Toronto Neighborhood profiles was received in a JSON format. Toronto Open Data website provided sample code snippets on how to use the API to access the dataset. However, this sample code only fetched a maximum of 100 records (regardless of how many actual records existed) and this meant that if the result (from the API call) had more than 100 records, it would only fetch 100 records and the rest of the record would be disregarded. It was not the plausible scenario for our case because the result we get from the API call, would contain more than 100 records. In order to address this issue, the pagination had to be implemented. How this pagination works is by, using the while loop. First, the API call was made to try to fetch the first 100 records. If the records and the next page existed, while loop kicked in, and then made a call to the API again to fetch the next 100 records in the next page. In each iteration, the result was converted to the DataFrame and then appended to a list. The iteration was continued until there were no more records to fetch. At this point, it broke out of the while loop and stopped calling the API. And then, multiple DataFrames in the list were concatenated into a single DataFrame containing the total ethnicity data from Toronto Neighborhood profiles. To demonstrate how this was done, the actual code is shown below.





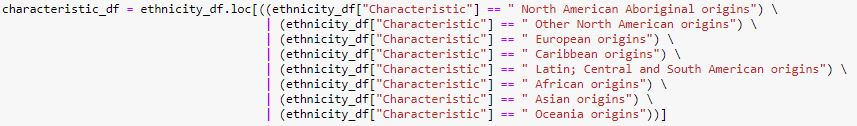
Step2

Toronto Neighborhood profiles provide a portrait of the demographic, social and economic characteristics of the people and households in each City of Toronto neighborhood. From this data, we extracted the information about the ethnicity distribution (Ethnic origin and Ethnic origin population) in Toronto by using pandas. 

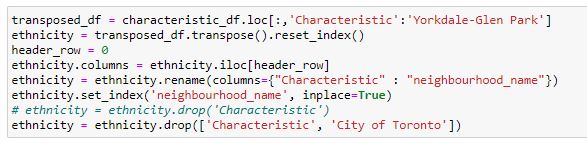
*Transform*

Step3

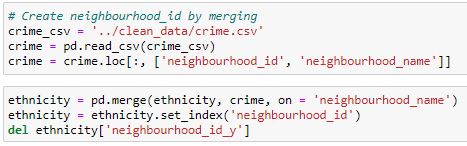
The only 8 ethnic origins representing each ethnicity (Aboriginal, North American, European, Caribbean, Latins, African, Asian, Oceanian) were selected. Columns and rows are transposed, and the column called ‘characteristic’ is renamed to ‘neighbourhood\_name’ which makes it easier to be merged with other data. Also, the columns which are not necessary for this project are eliminated.







Step 4

Finally, ‘neighbourhood\_id’ columns were added up and set as index by merging with the crime csv file.

### 

### 

### 

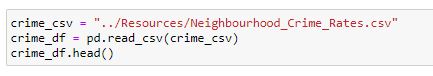
### 2.5 Toronto Crime Data

Neighbourhood Crime Rates dataset from Toronto City Open Data includes the 2014-2019 Crime Data by Neighbourhood. Counts are available for Assault, Auto Theft, Break and Enter, Robbery, Theft Over and Homicide. Data also includes five-year averages and crime rates per 100,000 people by neighbourhood based on 2016 Census Population. Using this data, our goal was to extract the information about the crime rate by neighborhood in Toronto. The steps will be explained in more details below.

*Extract*

Step1

Neighbourhood Crime Rates dataset came in CSV format from Toronto City Open Data Website.

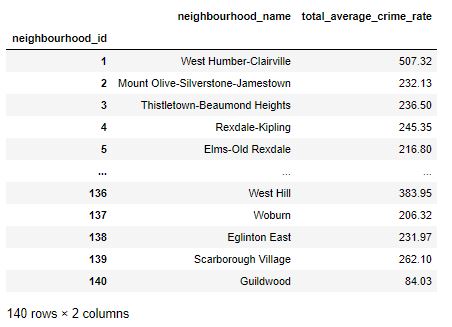


*Transform*

Step2

All unnecessary columns were eliminated since we only needed the 2019 crime data. Total average crime rates were calculated which is the total crime rate of crimes for 2019 per 100,00 population including Assault, Auto Theft, Break and Enter, Robbery, Theft Over and Homicide rates. Also, we created a new column for the values. Rest of the columns were removed and renamed until the data only has ‘Neighbourhood Id’, ‘Neighbourhood Name’ and ‘Average Crime Rate’ for columns. The result table and codes are down below.

### 



### 

### 

### 

### 

### 2.6 Restaurants Rating & Review Data from Yelp API

Yelp is a crowd-sourced local business review site where users can get lots of information about local restaurants, bars, cafes, etc. For the competition analysis, we wanted to look at which restaurants are popular to people, ratings, and the number of reviews information on restaurants on Yelp could be very valuable. Fortunately, Yelp allows the users to use API to get restaurant information on their site, and on Python, there is a library called ‘YelpAPI’, which makes it easier to use those API to get wanted information. Each step will be explained in detail.

*Extract*

#### Step 1

In order to use Yelp API, API key will be required from Yelp API site. You will need to use *‘from yelpapi import YelpAPI’* to import YelpAPI library.

#### Step 2

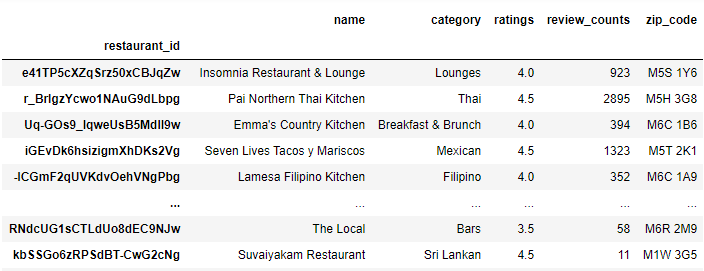
Toronto neighbourhood information with latitude and longitude was required as well, since Yelp allowed us to query based on latitude and longitude. 200-245 restaurants per neighbourhood were queried. The number of Toronto neighbourhoods is 140. By using each neighbourhood’s latitude and longitude, and for loop, we were able to get restaurant information about name, category, rating, the number of reviews, zip code, and hold these information in lists. You can see the code in the screenshot below:



*Transform*

#### Step 3

With the lists created in Step 2, a dataframe was created by using Pandas. The duplicated rows were dropped, and set the ‘restaurant\_id’ column as an index. There were no null values in the dataframe; removing null values was not needed.



### 2.7 Toronto Income Data

Toronto income dataset from Toronto City Open Data includes the Neighbourhood Profiles which provide a portrait of the demographic, social and economic characteristics of the people and households in each City of Toronto neighbourhood. The data is based on tabulations of 2016 Census of Population data from Statistics Canada. Using this data, we were extracting the information about household income by neighbourhood in Toronto.

*Extract*

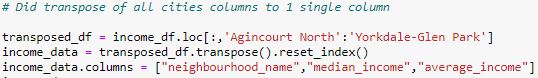
#### Step 1

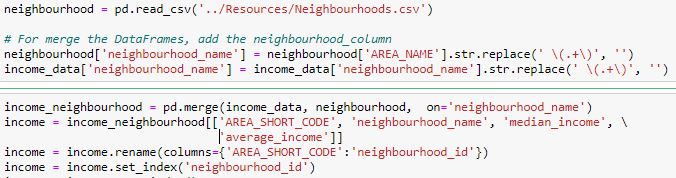
Toronto household income dataset came in CSV format from Toronto City Open Data Website.



*Transform*

Step 2

‘Income of households’ in the income category is selected as needed. Also, median and average of household total income is calculated for further analysis and all cities columns were transposed to one single column. Finally, income data and neighbourhood data was merged on neighbourhood name, and neighbourhood id was set as index. The result table and codes are down below.





# Data Load to SQL Database

### 3.1 Load onto PostgreSQL

#### Step 1

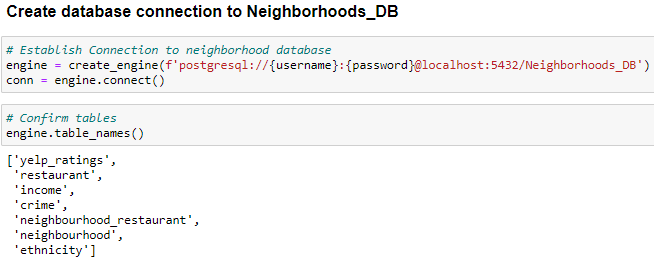
In pgAdmin4, create a table schema for each of the 7 transformed dataframe using the information and specified data types, primary keys, foreign keys, and other constraints.

Link to DB-Schema-Tables Creation:

[Neighbourhoods-DB-Schema-Tables-Creation.sql](https://github.com/saif-gorges/Toronto-New-Restaurant-Location-Analysis/blob/main/ERD/Neighbourhoods-DB-Schema-Tables-Creation.sql)

Step 2

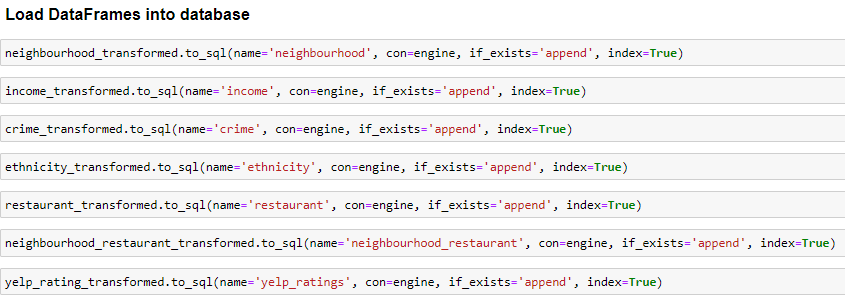
Import the module sqlalchemy and create an engine with the parameters user, password, and database name to connect and log in to the PostgreSQL database.



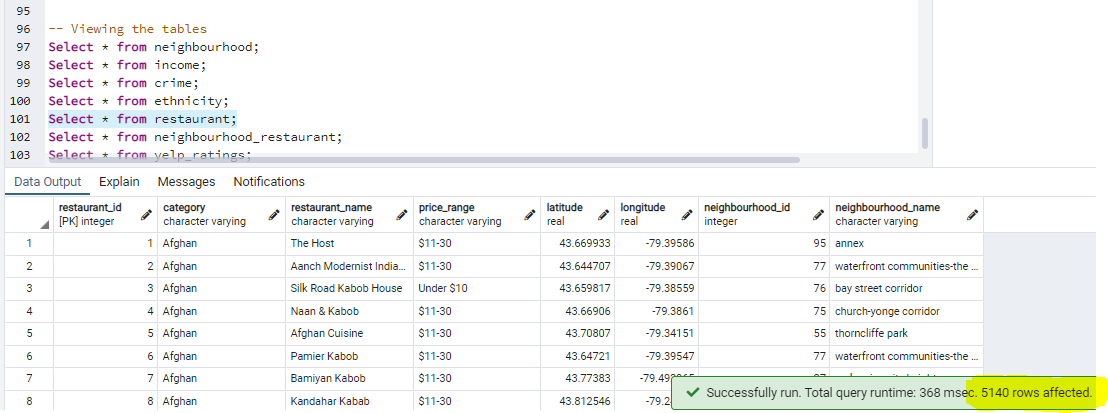
#### Step 3

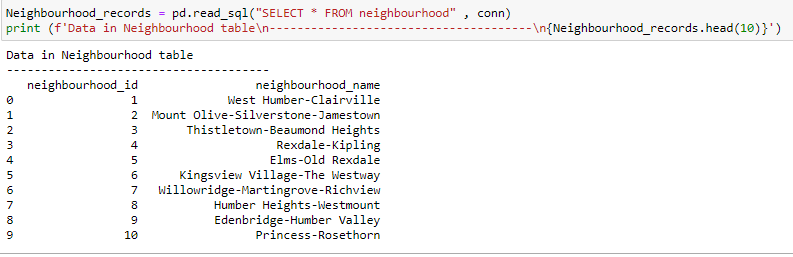
Load Final Transformed data into the tables using the to\_sql() function with the parameters table name, engine name, if\_exists, and index.

This approach accomplishes data loading in a more direct way, and allows us to add a whole dataframe to a PostgreSQL database all at once.



Checked if the tables are successfully populated with data in postgres - Neighbourhoods\_DB using the SELECT command. It also shows the number of rows affected successfully.

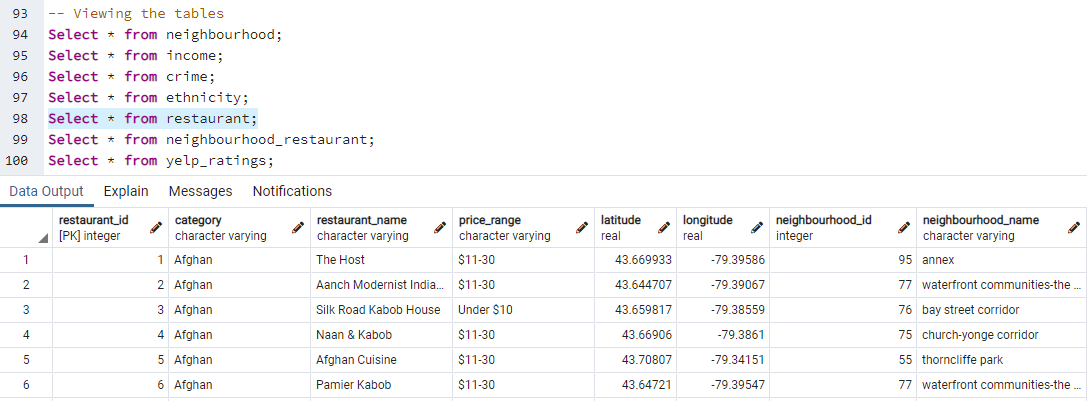


We further analysed the same set of data within Pandas using read\_sql command for all the tables.

### 3.2. Final Database

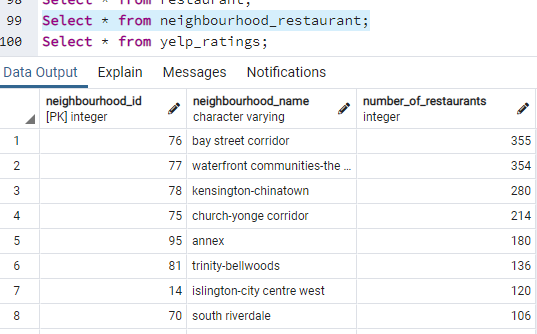
Confirm successful **Load** by querying database.

##### **3.2.1 Restaurant Table**

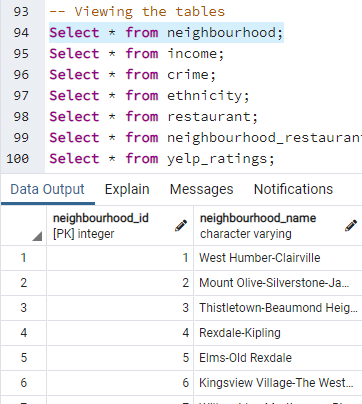


##### 

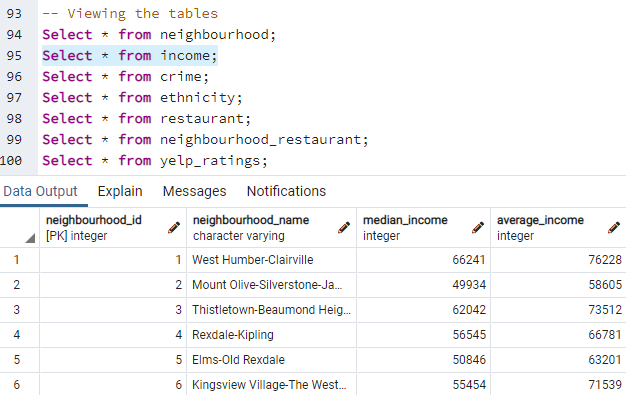
##### **3.2.2 Neighbourhood\_restaurant Table**



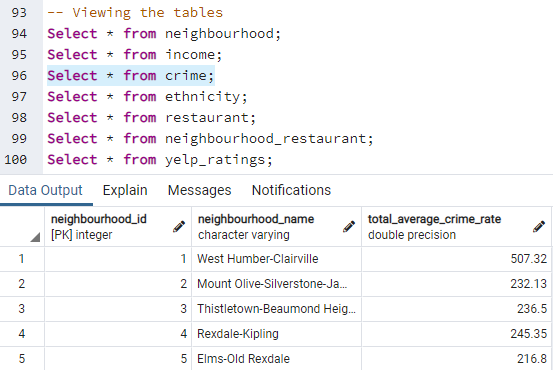
##### **3.2.3 Neighbourhood Table**



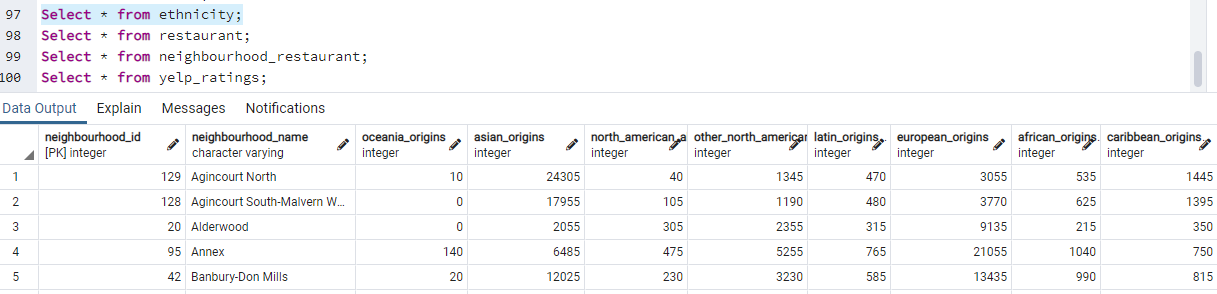
##### **3.2.4 Income Table**



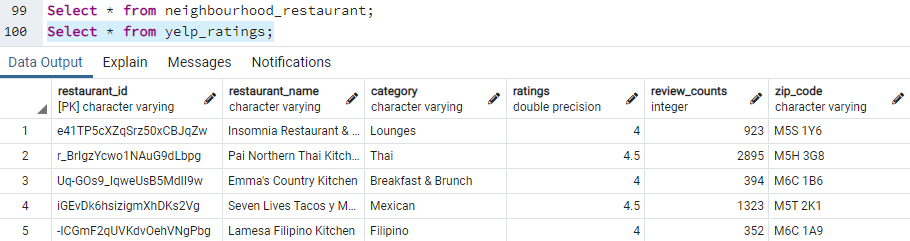
##### **3.2.5 Crime Table**



##### **3.2.6 Ethnicity Table**



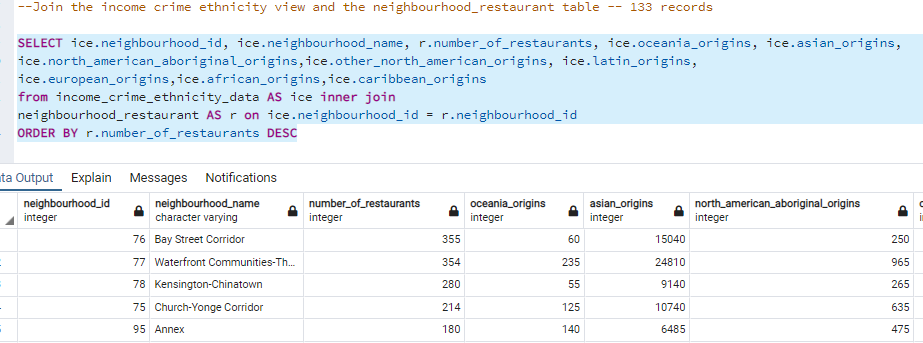
##### **3.2.7 Yelp\_Ratings Table**



### 3.3 Join Tables

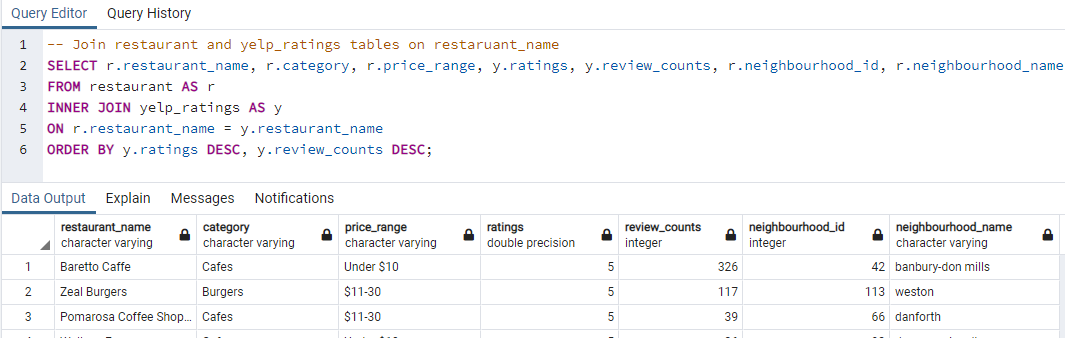
3.3.1 Join crime, income, ethnicity, and neighbourhood\_restaurant Tables on ‘neighbourhood\_id’

[Query-joining-Income-Crime-Ethnicity-Restaurant.sql](https://github.com/saif-gorges/Toronto-New-Restaurant-Location-Analysis/blob/main/ERD/Query-joining-Income-Crime-Ethnicity_Restaurant_number.sql)

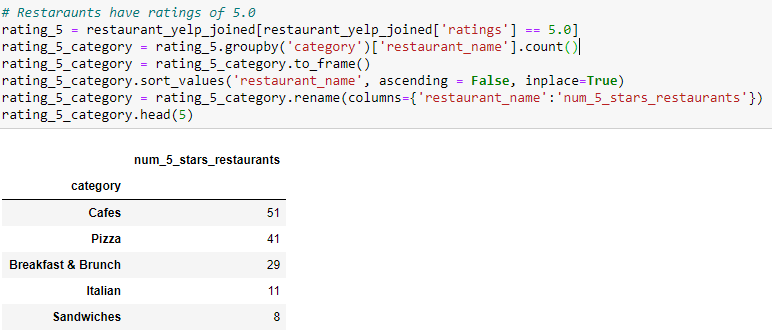


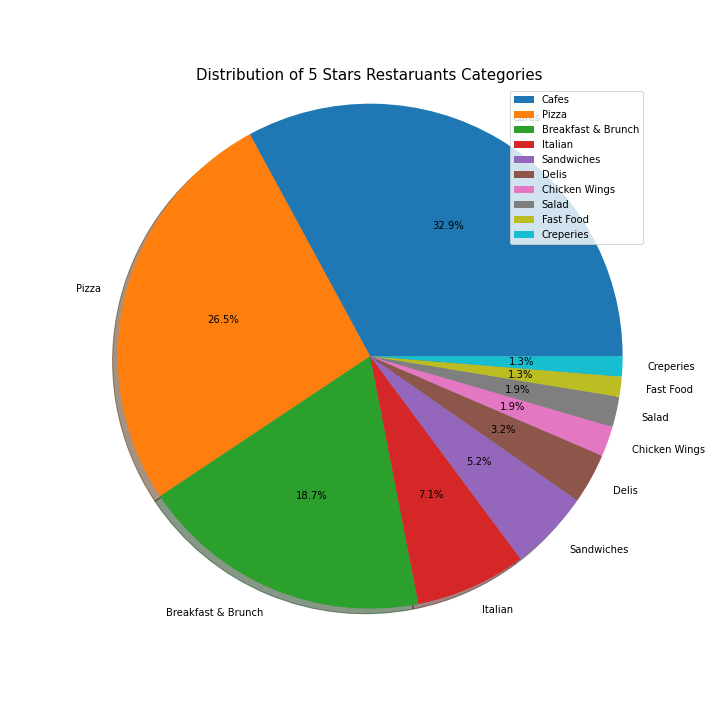
3.3.2 Join restaurant table and yelp\_ratings table on ‘restaurant\_name’

[Query-joining-Restaurants-and-Yelp-Ratings.sql](https://github.com/saif-gorges/Toronto-New-Restaurant-Location-Analysis/blob/main/ERD/restaurant_yelp_join_query.sql)

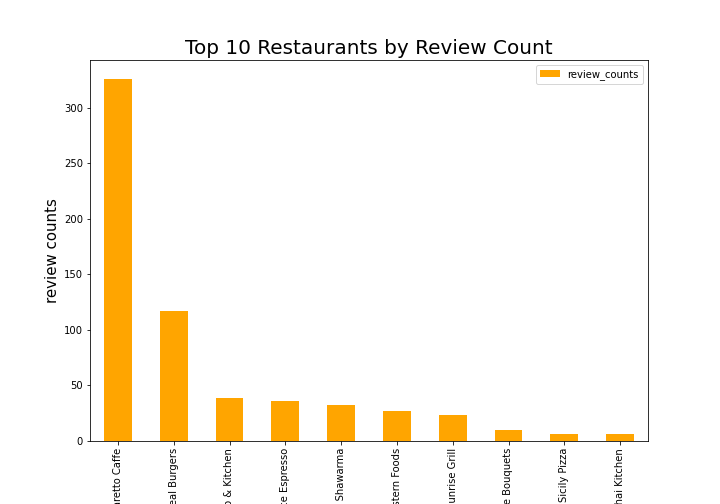


# Sample Analysis

With the joined table, the bar plot and pie plot were created by using Pandas and matplotlib. We were able to look at the distribution of 5 stars restaurants categories in the pie chart. The table was filtered using the code ‘restaurant\_yelp\_joined[restaurant\_yelp\_joined['ratings'] == 5.0]’, grouped by ‘category’, and count() method, we were able to get the dataframe to plot the pie chart. 



Also, the we sorted the joined table to get top 10 restaurants by review count, and created the bar chart:



# Discussion

### 5.1 Limitations

Some of the limitations encountered in this project was the availability of lat/long data in many of the datasets. There were some incredibly interesting potentials for analysis from many of our sources such as date/time of police interaction. Lat/long data from the ‘Toronto Police API’ would have allowed us to pinpoint crime rates based on date and time **within each neighborhood**. This could allow for restaurant owners to make better decisions for opening/closing times, staffing volume during certain times of the day or even extra security measures during certain times of the day in order to minimize cost. As we searched for data sources, we found many other very interesting types of data that would have made a very positive impact on our analysis, however unfortunately many of these datasets do not contain information that we could link to a neighborhood.

### 5.2 Next Steps

Considering the limitations, we believe that we have consolidated a valuable set of data for the purpose of our analysis. Moving forward, we will be taking careful consideration for what types of data is available. We have learned that some data points are not as easy to collect as others such as geodata. However, with the onset of a new digital age where everyone has a phone, these limitations may not be around for long.

In the next part of the project, we will be analysing our data to answer the following questions:

* What percent of restaurants are getting more than 250 reviews?
* What percent of restaurants are getting a rating of more than 4?
* Which localities have higher footfall? (areas with higher review counts are likely to be in neighbourhoods where residents dine out more often)
* Localities with saturated restaurant market space - review count less than 250 & rating greater than 4 (worst areas to open Restaurants, since there are few people who dine out and neighbourhood has higher number of good restaurants already existing )
* Neighborhoods with high potential for restaurant opening(business review count greater than 300 and rating less than 4)
* Market segmentation based on ethnicity (most popular neighborhoods by ethnicity) - can create heat maps if time permits to showcase the most popular ethnicity in each neighborhood or horizontal bar graph.