1. INTRODUCTION

1.1 BACKGROUND

The Greater Toronto Area (GTA) is the most populous metropolitan area in Canada. It consists of the City of Toronto and 4 surrounding regional municipalities: Durham, Halton, Peel, and York. There are 25 incorporated municipalities in the GTA area, and is recognized as one of the most multicultural and cosmopolitan cities in the world. The metropolitan area continues to grow and attract immigrants from all over the world.

1.2 PROBLEM

For new immigrants and families interested in Toronto as a place to live, and would like to find cultural/entertainment opportunities and establishments (restaurants, museums, parks, etc.) best suited for their own needs, this analysis will attempt to find things to do for new comers in Toronto that have a relatively high number of venues.

1.3 Interest

The key audience for this research is new immigrants who are considering a move to Toronto. Personally, I'm also interested in this analysis. Having lived in GTA for almost 6 years, I am looking forward to applying data science methods and location data to gain a better understanding of the key neighborhoods in the Toronto.

2. DATA ACQUISITION AND CLEANING

2.1 DATA SOURCES

Canada Post provides a free postal code look-up tool on its website, and for this analysis, we will be using data from this Wikipedia page:https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:M (https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:M) As for geographical coordinates of the neighborhoods, we will be using this link to a csv file: http://cocl.us/Geospatial_data (http://cocl.us/Geospatial_data).

The list of neighborhoods I will analyze include:

- Berczy Park
- Brockton, Parkdale Village, Exhibition Place
- Business reply mail Processing Centre, South Central Letter Processing Plant Toronto
- CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport
- Central Bay Street
- Christie
- Church and Wellesley
- Commerce Court, Victoria Hotel
- Davisville
- Davisville North

- Dufferin, Dovercourt Village
- First Canadian Place, Underground city
- Forest Hill North & West, Forest Hill Road Park
- Garden District, Ryerson
- Harbourfront East, Union Station, Toronto Islands
- High Park, The Junction South
- India Bazaar, The Beaches West
- Kensington Market, Chinatown, Grange Park
- Lawrence Park
- Little Portugal, Trinity
- Moore Park, Summerhill East
- North Toronto West, Lawrence Park
- Parkdale, Roncesvalles
- Queen's Park, Ontario Provincial Government
- Regent Park, Harbourfront
- Richmond, Adelaide, King
- Rosedale
- Roselawn
- Runnymede, Swansea
- St. James Town
- St. James Town, Cabbagetown
- Stn A PO Boxes
- Studio District
- Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park
- The Annex, North Midtown, Yorkville
- The Beaches
- The Danforth West, Riverdale
- Toronto Dominion Centre, Design Exchange
- University of Toronto, Harbord

For each neighborhood, I use the following types off data (source in parenthesis):

- Latitude (Foursquare API)
- Longitude (Foursquare API)
- Venues (Foursquare API)
- Type of Venue (Foursquare API)
- Postal code (Wikipedia)

In [1]:

```
# step 1: scraping data from Wikipedia
```

In [2]:

```
from bs4 import BeautifulSoup
import requests
import pandas as pd
import numpy as np

#get html from wiki page and create soup object
source = requests.get("https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:
soup = BeautifulSoup(source.text, "html.parser")

#using soup object, iterate the .wikitable to get the data from the HTML page and st
data = []
columns = []
table = soup.find(class_='wikitable')
for index, tr in enumerate(table.find_all('tr')):
    section = []
    for td in tr.find_all(['th','td']):
        section.append(td.text.rstrip())

    #First row of data is the header
    if (index == 0):
        columns = section
    else:
        data.append(section)

#convert list into Pandas DataFrame
toronto_df = pd.DataFrame(data = data,columns = columns)
toronto_df.head()
```

Out[2]:

	Postal Code	Borough	Neighborhood
0	M1A	Not assigned	Not assigned
1	M2A	Not assigned	Not assigned
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront

2.2 DATA CLEANING

Data for this analysis was pulled from listed sources, and compiled into one table. The data scraping and wrangling process using Pandas/Python was used to access neighborhood and venue data from the Foursquare API. All data are very recnet, from 2020. Please find the table below for your review:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017	Bakery
1	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809	Coffee Shop
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008	Distribution Center
3	Regent Park, Harbourfront	43.65426	-79.360636	Body Blitz Spa East	43.654735	-79.359874	Spa
4	Regent Park, Harbourfront	43.65426	-79.360636	Corktown Common	43.655618	-79.356211	Park

For a better review of our data, we only pick the top 10 most popular venues in each neighborhood, and each venue is catagorized for the type of service they provide.

Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th Most Common Venue	9th Most Common Venue	10th I Comr Venu
Berczy Park	Coffee Shop	Cocktail Bar	Restaurant	Seafood Restaurant	Beer Bar	Bakery	Cheese Shop	Café	Diner	Shopp Mall
Brockton, Parkdale Village, Exhibition Place	Café	Breakfast Spot	Performing Arts Venue	Coffee Shop	Climbing Gym	Burrito Place	Restaurant	Italian Restaurant	Intersection	Stadi
Business reply mail Processing Centre, South C...	Light Rail Station	Yoga Studio	Pizza Place	Smoke Shop	Skate Park	Brewery	Burrito Place	Restaurant	Recording Studio	Comi Shop
CN Tower, King and Spadina, Railway Lands, Har...	Airport Service	Airport Lounge	Coffee Shop	Harbor / Marina	Boutique	Boat or Ferry	Rental Car Location	Plane	Sculpture Garden	Airpo
Central Bay Street	Coffee Shop	Italian Restaurant	Sandwich Place	Japanese Restaurant	Café	Salad Place	Department Store	Dessert Shop	Thai Restaurant	Middl Easte Resta

In [4]:

```
# Data Cleaning 1: Remove Boroughs that are not assigned
toronto_df = toronto_df[toronto_df['Borough'] != 'Not assigned']
toronto_df.head()
```

Out[4]:

	Postal Code	Borough	Neighborhood
2	M3A	North York	Parkwoods
3	M4A	North York	Victoria Village
4	M5A	Downtown Toronto	Regent Park, Harbourfront
5	M6A	North York	Lawrence Manor, Lawrence Heights
6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

In [6]:

```
# Data Cleaning 2: groupby postacode to combine neighbourhoods nested in one postal
toronto_df["Neighborhood"] = toronto_df.groupby("Postal Code")["Neighborhood"].trans
```

In [7]:

```
# Data Cleaning 3: remove duplicates
toronto_df = toronto_df.drop_duplicates()
```

In [8]:

```
# Update index to be postal code if it isn't already
if(toronto_df.index.name != 'Postal Code'):
    toronto_df = toronto_df.set_index('Postal Code')

toronto_df.head()
```

Out[8]:

Postal Code	Borough	Neighborhood
M3A	North York	Parkwoods
M4A	North York	Victoria Village
M5A	Downtown Toronto	Regent Park, Harbourfront
M6A	North York	Lawrence Manor, Lawrence Heights
M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

In [9]:

```
#Get data lat/long data from link:http://cocl.us/Geospatial_data
data_path = 'http://cocl.us/Geospatial_data'
coord_df = pd.read_csv(data_path)
coord_df.head()
```

Out[9]:

	Postal Code	Latitude	Longitude
0	M1B	43.806686	-79.194353
1	M1C	43.784535	-79.160497
2	M1E	43.763573	-79.188711
3	M1G	43.770992	-79.216917
4	M1H	43.773136	-79.239476

In [10]:

```
#set the index to be Postcode
if(coord_df.index.name != 'Postal Code'):
    coord_df = coord_df.set_index('Postal Code')

coord_df.head()
```

Out[10]:

	Latitude	Longitude
Postal Code		
M1B	43.806686	-79.194353
M1C	43.784535	-79.160497
M1E	43.763573	-79.188711
M1G	43.770992	-79.216917
M1H	43.773136	-79.239476

In [18]:

```
# join the two dataframe together
toronto_df1= toronto_df.merge(coord_df, on='Postal Code', how='left')
toronto_df1.head()
```

Out[18]:

Postal Code	Borough	Neighborhood	Latitude	Longitude
M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
M4E	East Toronto	The Beaches	43.676357	-79.293031

2.3 LIMITS OF DATA

There are few limits of this analysis because of information access. For example, only 100 of the top venue results for any specific location are returned, which may not be a true representation of the cultrural/enterainment venues for neighborhoods with hundreds of venues, but rather a glimpse into how many venues are in the top 100. While this is not perfect, this analysis will still serve as a proxy for understanding relatively higher levels of resturants interests in these neighborhoods.

3. METHODOLOGY: EXPLORATORY DATA ANALYSIS

3.1 Built dataframe of the postal code, latitude and the longitude coordinates of each neighborhood.

As we have discussed in the DATA aquisition and cleaning section, the data frame will consist of three columns: postalcode, borough, and neighborhood. only process the cell sthat hace an assigned borough, and combine neighborhood with the same postal code. Now that you have built a dataframe of the postal code of each neighborhood along with borough name and neighborhood name, in order to utilize the Foursquare location data, we need to get the latitude and longitude coordinates of each neighborhood. Leveraging the Foursquare API to get these coordinates.

3.2 Segmenting and clustering neighborhoods in Toronto

We will use the explore function to get the most common venue categories in each neighborhood, and then use this feature to group the neighborhoods into clusters. We will then use the K-means clustering algorithm to complete this task. Let's create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood. Finally, we will use teh Folium library to visualize the neighborhoods in New York City and their emerging clusters.

In [15]:

```
# Explore and cluster the neighborhoods in Toronto based on the data we get from par
import numpy as np # library to handle data in a vectorized manner

import pandas as pd # library for data analsysis
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json # library to handle JSON files


#!conda install -c conda-forge geopy --yes # uncomment this line if you haven't comp
from geopy.geocoders import Nominatim # convert an address into latitude and longitu

import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas datafra

# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors

!pip3 install sklearn
!pip3 install folium

# import k-means from clustering stage
from sklearn.cluster import KMeans

#!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haver
import folium # map rendering library

print('Libraries imported.')
```

```
Requirement already satisfied: sklearn in /Library/Frameworks/Python.f
ramework/Versions/3.7/lib/python3.7/site-packages (0.0)
Requirement already satisfied: scikit-learn in /Library/Frameworks/Pyt
hon.framework/Versions/3.7/lib/python3.7/site-packages (from sklearn)
(0.23.1)
Requirement already satisfied: numpy>=1.13.3 in /Library/Frameworks/Py
thon.framework/Versions/3.7/lib/python3.7/site-packages (from scikit-l
earn->sklearn) (1.18.5)
Requirement already satisfied: scipy>=0.19.1 in /Library/Frameworks/Py
thon.framework/Versions/3.7/lib/python3.7/site-packages (from scikit-l
earn->sklearn) (1.4.1)
Requirement already satisfied: joblib>=0.11 in /Library/Frameworks/Pyt
hon.framework/Versions/3.7/lib/python3.7/site-packages (from scikit-le
arn->sklearn) (0.15.1)
Requirement already satisfied: threadpoolctl>=2.0.0 in /Library/Framew
orks/Python.framework/Versions/3.7/lib/python3.7/site-packages (from s
cikit-learn->sklearn) (2.1.0)
You are using pip version 10.0.1, however version 20.2b1 is available.
You should consider upgrading via the 'pip install --upgrade pip' comm
and.
Requirement already satisfied: folium in /Library/Frameworks/Python.fr
amework/Versions/3.7/lib/python3.7/site-packages (0.11.0)
Requirement already satisfied: numpy in /Library/Frameworks/Python.fra
mework/Versions/3.7/lib/python3.7/site-packages (from folium) (1.18.5)
Requirement already satisfied: branca>=0.3.0 in /Library/Frameworks/Py
thon.framework/Versions/3.7/lib/python3.7/site-packages (from folium)
```

```
 (0.4.1)
Requirement already satisfied: jinja2>=2.9 in /Library/Frameworks/Pyth
on.framework/Versions/3.7/lib/python3.7/site-packages (from folium)
 (2.11.2)
Requirement already satisfied: requests in /Library/Frameworks/Python.
framework/Versions/3.7/lib/python3.7/site-packages (from folium) (2.2
4.0)
Requirement already satisfied: MarkupSafe>=0.23 in /Library/Framework
s/Python.framework/Versions/3.7/lib/python3.7/site-packages (from jinj
a2>=2.9->folium) (1.1.1)
Requirement already satisfied: chardet<4,>=3.0.2 in /Library/Framework
s/Python.framework/Versions/3.7/lib/python3.7/site-packages (from requ
ests->folium) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1
in /Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/sit
e-packages (from requests->folium) (1.25.9)
Requirement already satisfied: certifi>=2017.4.17 in /Library/Framewor
ks/Python.framework/Versions/3.7/lib/python3.7/site-packages (from req
uests->folium) (2020.4.5.2)
Requirement already satisfied: idna<3,>=2.5 in /Library/Frameworks/Pyt
hon.framework/Versions/3.7/lib/python3.7/site-packages (from requests-
>folium) (2.9)
You are using pip version 10.0.1, however version 20.2b1 is available.
You should consider upgrading via the 'pip install --upgrade pip' comm
and.
Libraries imported.
```

In [20]:

```
#Filter dataframe to only use boroughs in Toronto
toronto_df1 = toronto_df1[toronto_df['Borough'].str.contains('Toronto')]
toronto_df1.head()
```

Out[20]:

Postal Code	Borough	Neighborhood	Latitude	Longitude
M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494
M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
M4E	East Toronto	The Beaches	43.676357	-79.293031

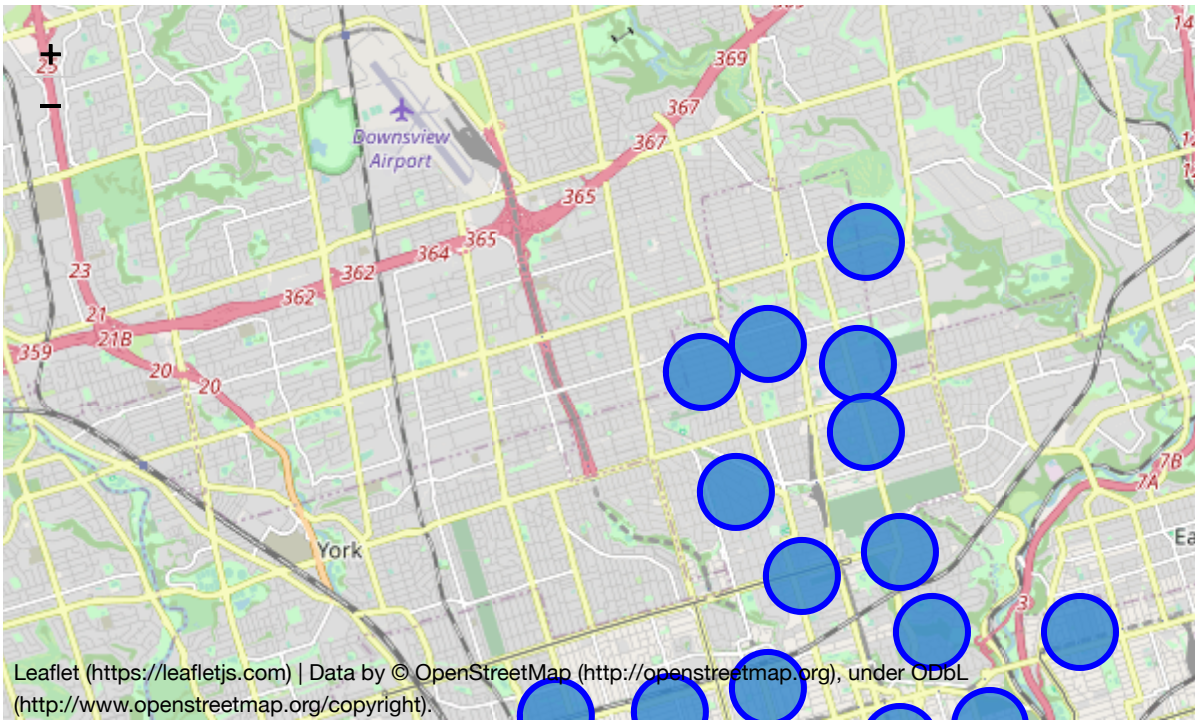
In [28]:

```
#Show an initial map of the neighborhoods in Toronto
# create map of Toronto using first entries latitude and longitude values
map_toronto = folium.Map(location=[toronto_df1["Latitude"][0], toronto_df1["Longitud

# add markers to map
for lat, lng, borough, neighborhood in zip(toronto_df1['Latitude'], toronto_df1['Lor
    label = '{}, {}'.format(neighborhood, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=18,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.8,
        parse_html=False).add_to(map_toronto)

map_toronto
```

Out[28]:



##

In [30]:

```
# Define Foursquare credentials and version
CLIENT_ID = 'J0JF35C2TJMAN1JDPPHQ3BRZ1K5HL1GRST2C5IN1RSF2VWJK' # your Foursquare ID
CLIENT_SECRET = 'SDKUVPCZELGDRXSO42041RKE53BOAVN4Y5XKBHYWMANVDQUR' # your Foursquare
VERSION = '20200617' # Foursquare API version

print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: J0JF35C2TJMAN1JDPPHQ3BRZ1K5HL1GRST2C5IN1RSF2VWJK
CLIENT_SECRET:SDKUVPCZELGDRXSO42041RKE53BOAVN4Y5XKBHYWMANVDQUR
```

In [33]:

```
# let's explore the first neighborhood in our dataframe
neighborhood_latitude = toronto_df1.loc['M5A', 'Latitude'] # neighborhood latitude v
neighborhood_longitude = toronto_df1.loc['M5A', 'Longitude'] # neighborhood longitud

neighborhood_name = toronto_df.loc['M5A', 'Neighborhood'] # neighborhood name

print('Latitude and longitude values of {} are {}, {}.'.format(neighborhood_name,
                                                               neighborhood_latitude
                                                               neighborhood_longitud
```

```
Latitude and longitude values of Regent Park, Harbourfront are 43.6542
599, -79.3606359.
```

In [35]:

```
# Setup API URL to explore venues near by
LIMIT = 100
radius = 500
url = 'https://api.foursquare.com/v2/venues/explore?client_id={}&client_secret={}&ll
neighborhood_json = requests.get(url).json()

# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']

venues = neighborhood_json['response']['groups'][0]['items']

nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.l
nearby_venues =nearby_venues.loc[:, filtered_columns]

# filter the category for each row
nearby_venues['venue.categories'] = nearby_venues.apply(get_category_type, axis=1)

# clean columns
nearby_venues.columns = [col.split(".")[-1] for col in nearby_venues.columns]

nearby_venues.head(20)
```

```
/Library/Frameworks/Python.framework/Versions/3.7/lib/python3.7/site-p
ackages/ipykernel_launcher.py:21: FutureWarning: pandas.io.json.json_n
ormalize is deprecated, use pandas.json_normalize instead
```

Out[35]:

	name	categories	lat	lng
0	Roselle Desserts	Bakery	43.653447	-79.362017
1	Tandem Coffee	Coffee Shop	43.653559	-79.361809
2	Cooper Koo Family YMCA	Distribution Center	43.653249	-79.358008
3	Body Blitz Spa East	Spa	43.654735	-79.359874
4	Dominion Pub and Kitchen	Pub	43.656919	-79.358967
5	Corktown Common	Park	43.655618	-79.356211
6	Impact Kitchen	Restaurant	43.656369	-79.356980
7	Morning Glory Cafe	Breakfast Spot	43.653947	-79.361149
8	The Extension Room	Gym / Fitness Center	43.653313	-79.359725
9	The Distillery Historic District	Historic Site	43.650244	-79.359323

	name	categories	lat	lng
10	Distillery Sunday Market	Farmers Market	43.650075	-79.361832
11	Figs Breakfast & Lunch	Breakfast Spot	43.655675	-79.364503
12	Sumach Espresso	Coffee Shop	43.658135	-79.359515
13	Arvo	Coffee Shop	43.649963	-79.361442
14	Young Centre for the Performing Arts	Performing Arts Venue	43.650825	-79.357593
15	Rooster Coffee	Coffee Shop	43.651900	-79.365609
16	Starbucks	Coffee Shop	43.651613	-79.364917
17	SOMA chocolatemaker	Chocolate Shop	43.650622	-79.358127
18	Underpass Park	Park	43.655764	-79.354806

In [36]:

```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
47 venues were returned by Foursquare.
```

In [37]:

```
# let's create a function to repeat the same process to all the neighborhoods in Tor

def getNearbyVenues(names, latitudes, longitudes, radius=500):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_sec
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            LIMIT)

        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']

        # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

    nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in ven
    nearby_venues.columns = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

In [39]:

```
toronto_venues = getNearbyVenues(names=toronto_df1['Neighborhood'],
                                 latitudes=toronto_df1['Latitude'],
                                 longitudes=toronto_df1['Longitude']
                                )
```

```
Regent Park, Harbourfront
Queen's Park, Ontario Provincial Government
Garden District, Ryerson
St. James Town
The Beaches
Berczy Park
Central Bay Street
Christie
Richmond, Adelaide, King
Dufferin, Dovercourt Village
Harbourfront East, Union Station, Toronto Islands
Little Portugal, Trinity
The Danforth West, Riverdale
Toronto Dominion Centre, Design Exchange
Brockton, Parkdale Village, Exhibition Place
India Bazaar, The Beaches West
Commerce Court, Victoria Hotel
Studio District
Lawrence Park
Roselawn
Davisville North
Forest Hill North & West, Forest Hill Road Park
High Park, The Junction South
North Toronto West,  Lawrence Park
The Annex, North Midtown, Yorkville
Parkdale, Roncesvalles
Davisville
University of Toronto, Harbord
Runnymede, Swansea
Moore Park, Summerhill East
Kensington Market, Chinatown, Grange Park
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst
Quay, South Niagara, Island airport
Rosedale
Stn A PO Boxes
St. James Town, Cabbagetown
First Canadian Place, Underground city
Church and Wellesley
Business reply mail Processing Centre, South Central Letter Processing
Plant Toronto
```

In [40]:

```
# Let's check the size of the resulting dataframe
print(toronto_venues.shape)
toronto_venues.head()
```

```
(1627, 7)
```

Out[40]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017	Bakery
1	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809	Coffee Shop
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008	Distribution Center
3	Regent Park, Harbourfront	43.65426	-79.360636	Body Blitz Spa East	43.654735	-79.359874	Spa
4	Regent Park, Harbourfront	43.65426	-79.360636	Dominion Pub and Kitchen	43.656919	-79.358967	Pub

In [41]:

```
toronto_venues.groupby('Neighborhood').count()
```

Out[41]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Berczy Park	58	58	58	58	58	58
Brockton, Parkdale Village, Exhibition Place	24	24	24	24	24	24
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto	17	17	17	17	17	17
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	16	16	16	16	16	16
Central Bay Street	66	66	66	66	66	66
Christie	17	17	17	17	17	17
Church and Wellesley	83	83	83	83	83	83
Commerce Court, Victoria Hotel	100	100	100	100	100	100
Davisville	33	33	33	33	33	33
Davisville North	9	9	9	9	9	9
Dufferin, Dovercourt Village	15	15	15	15	15	15
First Canadian Place, Underground city	100	100	100	100	100	100
Forest Hill North & West, Forest Hill Road Park	4	4	4	4	4	4
Garden District, Ryerson	100	100	100	100	100	100
Harbourfront East, Union Station, Toronto Islands	100	100	100	100	100	100
High Park, The Junction South	24	24	24	24	24	24
India Bazaar, The Beaches West	20	20	20	20	20	20
Kensington Market, Chinatown, Grange Park	60	60	60	60	60	60
Lawrence Park	3	3	3	3	3	3
Little Portugal, Trinity	44	44	44	44	44	44
Moore Park, Summerhill East	2	2	2	2	2	2
North Toronto West, Lawrence Park	20	20	20	20	20	20

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Parkdale, Roncesvalles	15	15	15	15	15	15
Queen's Park, Ontario Provincial Government	32	32	32	32	32	32
Regent Park, Harbourfront	47	47	47	47	47	47
Richmond, Adelaide, King	93	93	93	93	93	93
Rosedale	4	4	4	4	4	4
Roselawn	1	1	1	1	1	1
Runnymede, Swansea	34	34	34	34	34	34
St. James Town	80	80	80	80	80	80
St. James Town, Cabbagetown	47	47	47	47	47	47
Stn A PO Boxes	97	97	97	97	97	97
Studio District	41	41	41	41	41	41
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park	16	16	16	16	16	16
The Annex, North Midtown, Yorkville	20	20	20	20	20	20
The Beaches	6	6	6	6	6	6
The Danforth West, Riverdale	43	43	43	43	43	43
Toronto Dominion Centre, Design Exchange	100	100	100	100	100	100
University of Toronto, Harbord	36	36	36	36	36	36

In [42]:

```
# let's find out how many unique categories can be curated from all the returned ver
print('There are {} uniques categories.'.format(len(toronto_venues['Venue Category']
```

```
There are 234 uniques categories.
```

In [45]:

```
toronto_venues.head()
```

Out[45]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Regent Park, Harbourfront	43.65426	-79.360636	Roselle Desserts	43.653447	-79.362017	Bakery
1	Regent Park, Harbourfront	43.65426	-79.360636	Tandem Coffee	43.653559	-79.361809	Coffee Shop
2	Regent Park, Harbourfront	43.65426	-79.360636	Cooper Koo Family YMCA	43.653249	-79.358008	Distribution Center
3	Regent Park, Harbourfront	43.65426	-79.360636	Body Blitz Spa East	43.654735	-79.359874	Spa
4	Regent Park, Harbourfront	43.65426	-79.360636	Dominion Pub and Kitchen	43.656919	-79.358967	Pub

In [55]:

```
toronto_venues_count = toronto_venues.groupby('Neighborhood').count()
toronto_venues_count.sort_values(by = 'Venue', ascending = False, axis = 0, inplace
toronto_venues_count
```

Out[55]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Harbourfront East, Union Station, Toronto Islands	100	100	100	100	100	100
Toronto Dominion Centre, Design Exchange	100	100	100	100	100	100
Commerce Court, Victoria Hotel	100	100	100	100	100	100
First Canadian Place, Underground city	100	100	100	100	100	100
Garden District, Ryerson	100	100	100	100	100	100
Stn A PO Boxes	97	97	97	97	97	97
Richmond, Adelaide, King	93	93	93	93	93	93
Church and Wellesley	83	83	83	83	83	83
St. James Town	80	80	80	80	80	80
Central Bay Street	66	66	66	66	66	66
Kensington Market, Chinatown, Grange Park	60	60	60	60	60	60
Berczy Park	58	58	58	58	58	58
St. James Town, Cabbagetown	47	47	47	47	47	47
Regent Park, Harbourfront	47	47	47	47	47	47
Little Portugal, Trinity	44	44	44	44	44	44
The Danforth West, Riverdale	43	43	43	43	43	43
Studio District	41	41	41	41	41	41
University of Toronto, Harbord	36	36	36	36	36	36
Runnymede, Swansea	34	34	34	34	34	34
Davisville	33	33	33	33	33	33
Queen's Park, Ontario Provincial Government	32	32	32	32	32	32
High Park, The Junction South	24	24	24	24	24	24
Brockton, Parkdale Village, Exhibition Place	24	24	24	24	24	24
North Toronto West, Lawrence Park	20	20	20	20	20	20

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
The Annex, North Midtown, Yorkville	20	20	20	20	20	20
India Bazaar, The Beaches West	20	20	20	20	20	20
Christie	17	17	17	17	17	17
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto	17	17	17	17	17	17
Summerhill West, Rathnelly, South Hill, Forest Hill SE, Deer Park	16	16	16	16	16	16
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	16	16	16	16	16	16
Parkdale, Roncesvalles	15	15	15	15	15	15
Dufferin, Dovercourt Village	15	15	15	15	15	15
Davisville North	9	9	9	9	9	9
The Beaches	6	6	6	6	6	6
Rosedale	4	4	4	4	4	4
Forest Hill North & West, Forest Hill Road Park	4	4	4	4	4	4
Lawrence Park	3	3	3	3	3	3
Moore Park, Summerhill East	2	2	2	2	2	2
Roselawn	1	1	1	1	1	1

In [57]:

```
# Analyze each neighborhood
# one hot encoding
toronto_onehot = pd.get_dummies(toronto_venues[['Venue Category']], prefix="", prefi

# add neighborhood column back to dataframe
toronto_onehot['Neighborhood'] = toronto_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [toronto_onehot.columns[-1]] + list(toronto_onehot.columns[:-1])
toronto_onehot = toronto_onehot[fixed_columns]

toronto_onehot.head()
```

Out[57]:

	Yoga Studio	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Antique Shop
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0

In [58]:

```
toronto_onehot.shape
```

Out[58]:

```
(1627, 234)
```

In [59]:

```
toronto_grouped = toronto_onehot.groupby('Neighborhood').mean().reset_index()
toronto_grouped
```

Out[59]:

	Neighborhood	Yoga Studio	Afghan Restaurant	Airport	Airport Food Court	Airport Gate	Airport Lounge	Airport Service	Airport Terminal	American Restaurant	Ar
0	Berczy Park	0.000000	0.000000	0.0000	0.0000	0.0000	0.0000	0.0000	0.000	0.000000	0.0(
1	Brockton, Parkdale Village, Exhibition Place	0.041667	0.000000	0.0000	0.0000	0.0000	0.0000	0.0000	0.000	0.000000	0.0(
2	Business reply mail Processing Centre, South C...	0.058824	0.000000	0.0000	0.0000	0.0000	0.0000	0.0000	0.000	0.000000	0.0(
	CN Tower, King and										

In [60]:

```
# because of the built-in limit from the Foursquare API, where only 100 of the top v
# let's print each neighborhood along with the top 5 most common venues
num_top_venues = 5

for hood in toronto_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = toronto_grouped[toronto_grouped['Neighborhood'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_
    print('\n')
```

```
----Berczy Park----
          venue  freq
0   Coffee Shop  0.09
1  Cocktail Bar  0.05
2    Restaurant  0.03
3          Café  0.03
4   Cheese Shop  0.03


----Brockton, Parkdale Village, Exhibition Place----
                   venue  freq
0                   Café  0.12
1  Performing Arts Venue  0.08
2            Coffee Shop  0.08
3         Breakfast Spot  0.08
4            Yoga Studio  0.04


----Business reply mail Processing Centre, South Central Letter Proce
```

In [61]:

```
def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

In [64]:

```
num_top_venues = 5

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Neighborhood']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = toronto_grouped['Neighborhood']

for ind in np.arange(toronto_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(toronto_gr

neighborhoods_venues_sorted.head()
```

Out[64]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Berczy Park	Coffee Shop	Cocktail Bar	Café	Bakery	Seafood Restaurant
1	Brockton, Parkdale Village, Exhibition Place	Café	Coffee Shop	Breakfast Spot	Performing Arts Venue	Furniture / Home Store
2	Business reply mail Processing Centre, South C...	Yoga Studio	Auto Workshop	Garden Center	Gym / Fitness Center	Fast Food Restaurant
3	CN Tower, King and Spadina, Railway Lands, Har...	Airport Service	Airport Terminal	Coffee Shop	Plane	Rental Car Location
4	Central Bay Street	Coffee Shop	Sandwich Place	Italian Restaurant	Japanese Restaurant	Café

In [65]:

```
# set index to Neighborhood
if(neighborhoods_venues_sorted.index.name != 'Neighborhood'):
    neighborhoods_venues_sorted = neighborhoods_venues_sorted.set_index('Neighborhoo
neighborhoods_venues_sorted.head()
```

Out[65]:

Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Berczy Park	Coffee Shop	Cocktail Bar	Café	Bakery	Seafood Restaurant
Brockton, Parkdale Village, Exhibition Place	Café	Coffee Shop	Breakfast Spot	Performing Arts Venue	Furniture / Home Store
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto	Yoga Studio	Auto Workshop	Garden Center	Gym / Fitness Center	Fast Food Restaurant
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	Airport Service	Airport Terminal	Coffee Shop	Plane	Rental Car Location
Central Bay Street	Coffee Shop	Sandwich Place	Italian Restaurant	Japanese Restaurant	Café

In [69]:

```
# merge toronto_venues_count and neighborhoods_venues_sorted
toronto_venues_popularity=toronto_venues_count.merge(neighborhoods_venues_sorted, or
toronto_venues_popularity.head(10)
```

Out[69]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	1st Most Common Venue	2
Harbourfront East, Union Station, Toronto Islands	100	100	100	100	100	100	Coffee Shop	
Toronto Dominion Centre, Design Exchange	100	100	100	100	100	100	Coffee Shop	
Commerce Court, Victoria Hotel	100	100	100	100	100	100	Coffee Shop	
First Canadian Place, Underground city	100	100	100	100	100	100	Coffee Shop	
Garden District, Ryerson	100	100	100	100	100	100	Clothing Store	
Stn A PO Boxes	97	97	97	97	97	97	Coffee Shop	
Richmond, Adelaide, King	93	93	93	93	93	93	Coffee Shop	
Church and Wellesley	83	83	83	83	83	83	Coffee Shop	R
St. James Town	80	80	80	80	80	80	Coffee Shop	
Central Bay Street	66	66	66	66	66	66	Coffee Shop	

Top 10 most popular Neighborhoods

In [83]:

```
toronto_top10 = toronto_df1
toronto_top10 = toronto_top10.merge(toronto_venues_popularity, on='Neighborhood', ho
toronto_top10 = toronto_top10.head(10)
toronto_top10
```

hborhood	Latitude	Longitude	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category	1st Mos Commo Venu
gent Park, ırbourfront	43.654260	-79.360636	47	47	47	47	47	47	Coffe Sho
en's Park, Ontario Provincial overnment	43.662301	-79.389494	32	32	32	32	32	32	Coffe Sho
Garden District, Ryerson	43.657162	-79.378937	100	100	100	100	100	100	Clothin Stor
St. James Town	43.651494	-79.375418	80	80	80	80	80	80	Coffe Sho

In [109]:

```
neighborhoods_venues_sorted.head()
```

Out[109]:

Neighborhood	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
Berczy Park	2	Coffee Shop	Cocktail Bar	Café	Bakery	Seafood Restaurant
Brockton, Parkdale Village, Exhibition Place	2	Café	Coffee Shop	Breakfast Spot	Performing Arts Venue	Furniture / Home Store
Business reply mail Processing Centre, South Central Letter Processing Plant Toronto	2	Yoga Studio	Auto Workshop	Garden Center	Gym / Fitness Center	Fast Food Restaurant
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	2	Airport Service	Airport Terminal	Coffee Shop	Plane	Rental Car Location
Central Bay Street	2	Coffee Shop	Sandwich Place	Italian Restaurant	Japanese Restaurant	Café

In [113]:

```
# Clustering Neighborhoods
```

In [110]:

```
# Run K-means to cluster the neighborhood into 10 clusters
# set number of clusters
kclusters = 5

toronto_grouped_clustering = toronto_grouped.drop('Neighborhood', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(toronto_grouped_clustering

# merge toronto_grouped with toronto_data to add latitude/longtitude for each neighb
toronto_merged = toronto_df1.merge(neighborhoods_venues_sorted, on = 'Neighborhood',

toronto_merged.head()
```

Out[110]:

	Borough	Neighborhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	
0	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636	2	Coffee Shop	Park	Pub	
1	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.662301	-79.389494	2	Coffee Shop	Sushi Restaurant	Yoga Studio	F
2	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937	2	Clothing Store	Coffee Shop	Cosmetics Shop	
3	Downtown Toronto	St. James Town	43.651494	-79.375418	2	Coffee Shop	Café	Cocktail Bar	F
4	East Toronto	The Beaches	43.676357	-79.293031	0	Asian Restaurant	Pizza Place	Pub	

In [112]:

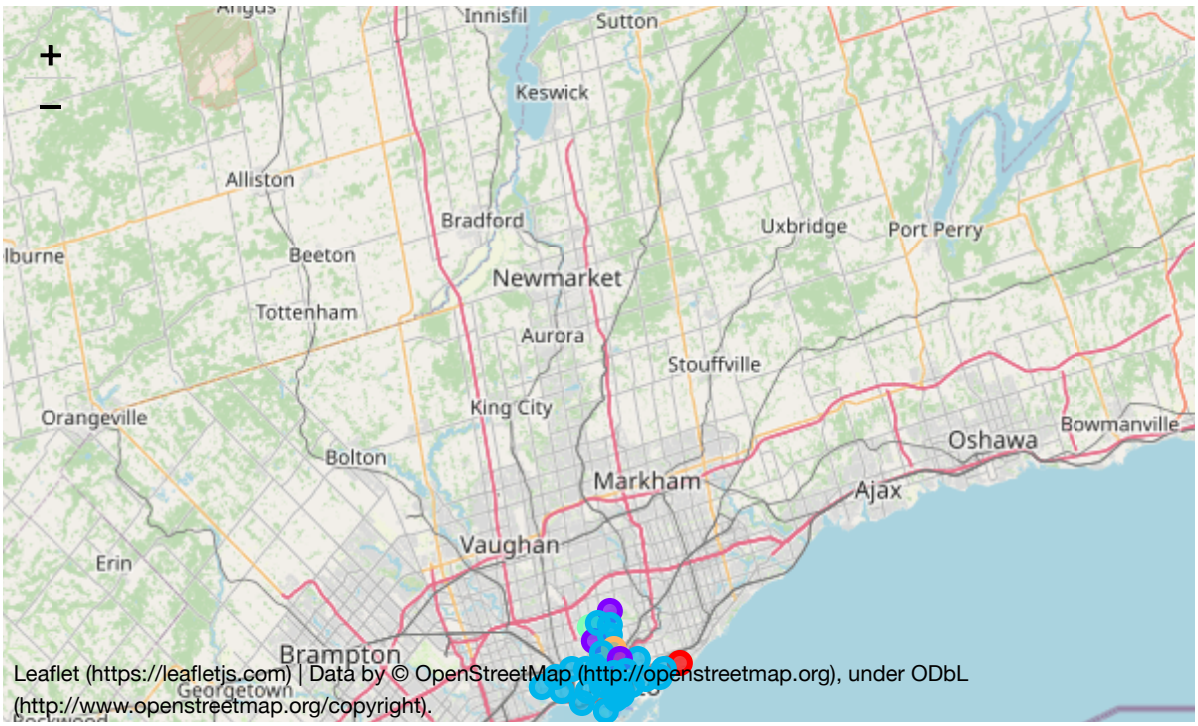
```
# create map
map_clusters = folium.Map(location=[toronto_df1["Latitude"][0], toronto_df1["Longitu

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i + x + (i*x)**2 for i in range(kclusters)]
colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
rainbow = [colors.rgb2hex(i) for i in colors_array]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(toronto_merged['Latitude'], toronto_merged['Longit
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[112]:



4. CONCLUSION

4.1 Top 10 most popular neighborhood

The most popular neighborhood in Toronto is listed below:

- Harbourfront, Regent Park

- Queen's park
- Garden District, Ryerson
- St. James Town
- The Beaches
- Berczy Park
- Central Bay Street
- Christie
- Richmond, Adelaide, King
- Dufferin, Dovercourt Village

4.2 Most popular venues in each neighborhood

In [129]:

```
toronto_top10[['Neighborhood', '1st Most Common Venue', '2nd Most Common Venue', '3r
```

Out[129]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
0	Regent Park, Harbourfront	Coffee Shop	Park	Pub	Bakery	Café
1	Queen's Park, Ontario Provincial Government	Coffee Shop	Sushi Restaurant	Yoga Studio	Restaurant	Bar
2	Garden District, Ryerson	Clothing Store	Coffee Shop	Cosmetics Shop	Café	Bubble Tea Shop
3	St. James Town	Coffee Shop	Café	Cocktail Bar	American Restaurant	Gastropub
4	The Beaches	Asian Restaurant	Pizza Place	Pub	Trail	Health Food Store
5	Berczy Park	Coffee Shop	Cocktail Bar	Café	Bakery	Seafood Restaurant
6	Central Bay Street	Coffee Shop	Sandwich Place	Italian Restaurant	Japanese Restaurant	Café
7	Christie	Grocery Store	Café	Park	Athletics & Sports	Candy Store
8	Richmond, Adelaide, King	Coffee Shop	Café	Restaurant	Deli / Bodega	Gym
9	Dufferin, Dovercourt Village	Bakery	Pharmacy	Park	Pizza Place	Recording Studio

In [ ]:

```

```