Segmentation and Clustering of Neighborhood in Toronto, Canada.

Introduction

```
1.Web-scraping of the Toronto Neighborhood data using requests and
BeautifulSoup Libraries.

2.Performing Exploratory and Data analysis Using the Pandas and Numpy
Library.

3.Using Foursquare API Explore the Neighborhoods of Toronto location.

4.Using Scikit-learn library to apply K-means Clustering alogrithm.

5.Analayzing diffferent clusters.
```

Table of Contents

1. Performing Web-scraping and Extracting Neighborhood and borough information from Wikipedia Page

In [1]:

```python
# Installing essential libraries in Web-scraping
# import the library we use to open the URL
import lxml
import requests # library to handle requests
# Import the beautifulSoup library so we can parse HTML and XML documents
from bs4 import BeautifulSoup as BS

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 completed the Foursquare API lab
from geopy.geocoders import Nominatim # convert an address into latitude
and longitude values

from pandas.io.json import json_normalize # tranform JSON file into a
pandas dataframe

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

# 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 haven't completed the Foursquare API lab
import folium # map rendering library

print('Libraries imported.')
```

```
Libraries imported.
```

In [2]:

```
# Storing the URL of wikipedia wiki_url
wiki_url =
'https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M'

# Getting the source code of the HTML page from the URL using the request
library
source = requests.get(wiki_url).text

# Converting the HTML souce code to the beautiful soup format
soup = BS(source,'html.parser')

# Reading the table Information and storing in the database
df_old = pd.read_html(wiki_url)

# Using the find method to find the exact borough information
df_old[0]
```

Out[2]:

	Postal Code	Borough	Neighbourhood

	Postal Code	Borough	Neighbourhood
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
5	M6A	North York	Lawrence Manor, Lawrence Heights
6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government
7	M8A	Not assigned	Not assigned
8	M9A	Etobicoke	Islington Avenue, Humber Valley Village
9	M1B	Scarborough	Malvern, Rouge
10	M2B	Not assigned	Not assigned
11	M3B	North York	Don Mills
12	M4B	East York	Parkview Hill, Woodbine Gardens
13	M5B	Downtown Toronto	Garden District, Ryerson
14	M6B	North York	Glencairn
15	M7B	Not assigned	Not assigned
16	M8B	Not assigned	Not assigned
17	M9B	Etobicoke	West Deane Park, Princess Gardens, Martin Grov...
18	M1C	Scarborough	Rouge Hill, Port Union, Highland Creek
19	M2C	Not assigned	Not assigned
20	M3C	North York	Don Mills
21	M4C	East York	Woodbine Heights
22	M5C	Downtown Toronto	St. James Town
23	M6C	York	Humewood-Cedarvale
24	M7C	Not assigned	Not assigned
25	M8C	Not assigned	Not assigned
26	M9C	Etobicoke	Eringate, Bloordale Gardens, Old Burnhamthorpe...
27	M1E	Scarborough	Guildwood, Morningside, West Hill
28	M2E	Not assigned	Not assigned
29	M3E	Not assigned	Not assigned
30	M4E	East Toronto	The Beaches
31	M5E	Downtown Toronto	Berczy Park
32	M6E	York	Caledonia-Fairbanks
33	M7E	Not assigned	Not assigned
34	M8E	Not assigned	Not assigned
35	M9E	Not assigned	Not assigned

	Postal Code	Borough	Neighbourhood
36	M1G	Scarborough	Woburn
37	M2G	Not assigned	Not assigned
38	M3G	Not assigned	Not assigned
39	M4G	East York	Leaside
40	M5G	Downtown Toronto	Central Bay Street
41	M6G	Downtown Toronto	Christie
42	M7G	Not assigned	Not assigned
43	M8G	Not assigned	Not assigned
44	M9G	Not assigned	Not assigned
45	M1H	Scarborough	Cedarbrae
46	M2H	North York	Hillcrest Village
47	M3H	North York	Bathurst Manor, Wilson Heights, Downsview North
48	M4H	East York	Thorncliffe Park
49	M5H	Downtown Toronto	Richmond, Adelaide, King
50	M6H	West Toronto	Dufferin, Dovercourt Village
51	M7H	Not assigned	Not assigned
52	M8H	Not assigned	Not assigned
53	M9H	Not assigned	Not assigned
54	M1J	Scarborough	Scarborough Village
55	M2J	North York	Fairview, Henry Farm, Oriole
56	M3J	North York	Northwood Park, York University
57	M4J	East York	East Toronto, Broadview North (Old East York)
58	M5J	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
59	M6J	West Toronto	Little Portugal, Trinity
60	M7J	Not assigned	Not assigned
61	M8J	Not assigned	Not assigned
62	M9J	Not assigned	Not assigned
63	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park
64	M2K	North York	Bayview Village
65	M3K	North York	Downsview
66	M4K	East Toronto	The Danforth West, Riverdale
67	M5K	Downtown Toronto	Toronto Dominion Centre, Design Exchange
68	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place
69	M7K	Not assigned	Not assigned
70	M8K	Not assigned	Not assigned
71	M9K	Not assigned	Not assigned

	Postal Code	Borough	Neighbourhood
72	M1L	Scarborough	Golden Mile, Clairlea, Oakridge
73	M2L	North York	York Mills, Silver Hills
74	M3L	North York	Downsview
75	M4L	East Toronto	India Bazaar, The Beaches West
76	M5L	Downtown Toronto	Commerce Court, Victoria Hotel
77	M6L	North York	North Park, Maple Leaf Park, Upwood Park
78	M7L	Not assigned	Not assigned
79	M8L	Not assigned	Not assigned
80	M9L	North York	Humber Summit
81	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
82	M2M	North York	Willowdale, Newtonbrook
83	M3M	North York	Downsview
84	M4M	East Toronto	Studio District
85	M5M	North York	Bedford Park, Lawrence Manor East
86	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
87	M7M	Not assigned	Not assigned
88	M8M	Not assigned	Not assigned
89	M9M	North York	Humberlea, Emery
90	M1N	Scarborough	Birch Cliff, Cliffside West
91	M2N	North York	Willowdale, Willowdale East
92	M3N	North York	Downsview
93	M4N	Central Toronto	Lawrence Park
94	M5N	Central Toronto	Roselawn
95	M6N	York	Runnymede, The Junction North
96	M7N	Not assigned	Not assigned
97	M8N	Not assigned	Not assigned
98	M9N	York	Weston
99	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
100	M2P	North York	York Mills West
101	M3P	Not assigned	Not assigned
102	M4P	Central Toronto	Davisville North
103	M5P	Central Toronto	Forest Hill North & West, Forest Hill Road Park
104	M6P	West Toronto	High Park, The Junction South
105	M7P	Not assigned	Not assigned
106	M8P	Not assigned	Not assigned
107	M9P	Etobicoke	Westmount

	Postal Code	Borough	Neighbourhood
108	M1R	Scarborough	Wexford, Maryvale
109	M2R	North York	Willowdale, Willowdale West
110	M3R	Not assigned	Not assigned
111	M4R	Central Toronto	North Toronto West, Lawrence Park
112	M5R	Central Toronto	The Annex, North Midtown, Yorkville
113	M6R	West Toronto	Parkdale, Roncesvalles
114	M7R	Mississauga	Canada Post Gateway Processing Centre
115	M8R	Not assigned	Not assigned
116	M9R	Etobicoke	Kingsview Village, St. Phillips, Martin Grove ...
117	M1S	Scarborough	Agincourt
118	M2S	Not assigned	Not assigned
119	M3S	Not assigned	Not assigned
120	M4S	Central Toronto	Davisville
121	M5S	Downtown Toronto	University of Toronto, Harbord
122	M6S	West Toronto	Runnymede, Swansea
123	M7S	Not assigned	Not assigned
124	M8S	Not assigned	Not assigned
125	M9S	Not assigned	Not assigned
126	M1T	Scarborough	Clarks Corners, Tam O'Shanter, Sullivan
127	M2T	Not assigned	Not assigned
128	M3T	Not assigned	Not assigned
129	M4T	Central Toronto	Moore Park, Summerhill East
130	M5T	Downtown Toronto	Kensington Market, Chinatown, Grange Park
131	M6T	Not assigned	Not assigned
132	M7T	Not assigned	Not assigned
133	M8T	Not assigned	Not assigned
134	M9T	Not assigned	Not assigned
135	M1V	Scarborough	Milliken, Agincourt North, Steeles East, L'Amo...
136	M2V	Not assigned	Not assigned
137	M3V	Not assigned	Not assigned
138	M4V	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...
139	M5V	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...
140	M6V	Not assigned	Not assigned
141	M7V	Not assigned	Not assigned
142	M8V	Etobicoke	New Toronto, Mimico South, Humber Bay Shores
143	M9V	Etobicoke	South Steeles, Silverstone, Humbergate, Jamest...

	Postal Code	Borough	Neighbourhood
144	M1W	Scarborough	Steeles West, L'Amoreaux West
145	M2W	Not assigned	Not assigned
146	M3W	Not assigned	Not assigned
147	M4W	Downtown Toronto	Rosedale
148	M5W	Downtown Toronto	Stn A PO Boxes
149	M6W	Not assigned	Not assigned
150	M7W	Not assigned	Not assigned
151	M8W	Etobicoke	Alderwood, Long Branch
152	M9W	Etobicoke	Northwest, West Humber - Clairville
153	M1X	Scarborough	Upper Rouge
154	M2X	Not assigned	Not assigned
155	M3X	Not assigned	Not assigned
156	M4X	Downtown Toronto	St. James Town, Cabbagetown
157	M5X	Downtown Toronto	First Canadian Place, Underground city
158	M6X	Not assigned	Not assigned
159	M7X	Not assigned	Not assigned
160	M8X	Etobicoke	The Kingsway, Montgomery Road, Old Mill North
161	M9X	Not assigned	Not assigned
162	M1Y	Not assigned	Not assigned
163	M2Y	Not assigned	Not assigned
164	M3Y	Not assigned	Not assigned
165	M4Y	Downtown Toronto	Church and Wellesley
166	M5Y	Not assigned	Not assigned
167	M6Y	Not assigned	Not assigned
168	M7Y	East Toronto	Business reply mail Processing Centre, South C...
169	M8Y	Etobicoke	Old Mill South, King's Mill Park, Sunnylea, Hu...
170	M9Y	Not assigned	Not assigned
171	M1Z	Not assigned	Not assigned
172	M2Z	Not assigned	Not assigned
173	M3Z	Not assigned	Not assigned
174	M4Z	Not assigned	Not assigned
175	M5Z	Not assigned	Not assigned
176	M6Z	Not assigned	Not assigned
177	M7Z	Not assigned	Not assigned
178	M8Z	Etobicoke	Mimico NW, The Queensway West, South of Bloor,...
179	M9Z	Not assigned	Not assigned

In [3]:

```
# Checking the  database for all the Unique Borough names and total number
of neighborhood
df = df_old[0].copy()
df.head()
```

Out[3]:

	Postal Code	Borough	Neighbourhood
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. Perfoming EDA

Neighborhood has a total of 3 boroughs and 177 neighborhoods. In order to segement the neighborhoods and explore them, we will essentially need a dataset that contains the 3 boroughs and the neighborhoods that exist in each borough as well as the the latitude and logitude coordinates of each neighborhood.

Data Handling and Cleaning

In [4]:

```
# Checking the dataFrame for missing Values
df.isna().sum()
```

Out[4]:

```
Postal Code      0
Borough          0
Neighbourhood    0
dtype: int64
```

In [5]:

```
# dropping the rows having 'Not assigned' in Borough column
df = df[~(df['Borough'] == 'Not assigned')]
df.head()
```

Out[5]:

	Postal Code	Borough	Neighbourhood
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]:

```
# Cheking again for any missing or null values
df.isna().sum()
```

Out[6]:

```
Postal Code      0
Borough          0
Neighbourhood    0
dtype: int64
```

In [7]:

```
df.shape
```

Out[7]:

```
(103, 3)
```

Using pgeocode library to get the latitudes and longitudes of each Neighborhood from the postal code given.

In [8]:

```
import pgeocode
nomi = pgeocode.Nominatim('CA')
location = nomi.query_postal_code("M3A")
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
43.7545 -79.33
```

In [9]:

```
df['Postal Code'].head()
```

Out[9]:

```
2    M3A
3    M4A
4    M5A
5    M6A
6    M7A
Name: Postal Code, dtype: object
```

In [10]:

```
# Creating a new DataFrame to store the latitudes and longitudes of the each neighborhood
Country_code = 'CA'
column_names = [ 'Postal Code', 'Borough', 'Neighbourhood', 'Latitude', 'Longitude']
Neighbourhood = pd.DataFrame(columns=column_names)

# We will define a instance of the geopy library as an tor_agent
# We cannot use the Nominatim method for large number of values hence we divide the dataset into two parts

# First Part of the dataset
for postal, bor, neigh in zip(df['Postal Code'], df['Borough'], df['Neighbourhood']):
    try:
        nomi = pgeocode.Nominatim(Country_code)
        location = nomi.query_postal_code(postal)
        Neighbourhood = Neighbourhood.append({
                                              'Postal Code':postal,
                                              'Borough':bor,
                                              'Neighbourhood':neigh,
                                              'Latitude':location.latitude,
                                              'Longitude':location.longitude
                                             },ignore_index=True)
    except:
        Neighbourhood = Neighbourhood.append({
                                              'Postal Code':postal,
                                              'Borough':bor,
                                              'Neighbourhood':neigh,
                                              'Latitude':np.NaN,
                                              'Longitude':np.NaN
                                             },ignore_index=True)



# Cheking for any null values
Neighbourhood.head()
```

Out[10]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M3A	North York	Parkwoods	43.7545	-79.3300
1	M4A	North York	Victoria Village	43.7276	-79.3148
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.6555	-79.3626
3	M6A	North York	Lawrence Manor, Lawrence Heights	43.7223	-79.4504
4	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.6641	-79.3889

In [11]:

```
# Checking for the missing values in the data frame
Neighbourhood.isna().sum()
```

Out[11]:

```
Postal Code      0
Borough          0
Neighbourhood    0
Latitude         1
Longitude        1
dtype: int64
```

In [12]:

```
# Checking the one row whose latitude and logitude were not extracted
properly
Neighbourhood[Neighbourhood.Latitude.isna()]
```

Out[12]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
76	M7R	Mississauga	Canada Post Gateway Processing Centre	NaN	NaN

In [13]:

```
# Removing the one row whose latitude and logitude were not extracted
properly
Neighbourhood = Neighbourhood[~(Neighbourhood.Latitude.isna())]
Neighbourhood.isna().sum()
```

Out[13]:

```
Postal Code      0
Borough          0
```

```
Neighbourhood    0
Latitude         0
Longitude        0
dtype: int64
```

In [14]:

```
# Checking all the different borough and the neighbour associated with it
Neighbourhood['Borough'].value_counts()
```

Out[14]:

```
North York         24
Downtown Toronto   19
Scarborough        17
Etobicoke          12
Central Toronto     9
West Toronto        6
East York           5
York                5
East Toronto        5
Name: Borough, dtype: int64
```

In [15]:

```
# Printing the information for Toronto Dataset
print('The dataframe has {} boroughs and {} neighborhoods.'.format(
        len(Neighbourhood['Borough'].unique()),
        Neighbourhood.shape[0]
    )
)
```

```
The dataframe has 9 boroughs and 102 neighborhoods.
```

3.Exploring the neighbourhood of Toronto

Using the visualization library Folium and the Neighbourhood datasets Co-ordinates

In [16]:

```
# Extracting the coordinates of Toronto city
nomi = pgeocode.Nominatim('CA')
location = nomi.query_postal_code("M4B")
toronto_latitude = location.latitude
toronto_longitude = location.longitude
print('The Coordinates of Toronto Canada is  {},
{}'.format(toronto_latitude, toronto_longitude))
```

```
The Coordinates of Toronto Canada is  43.7063, -79.3094
```

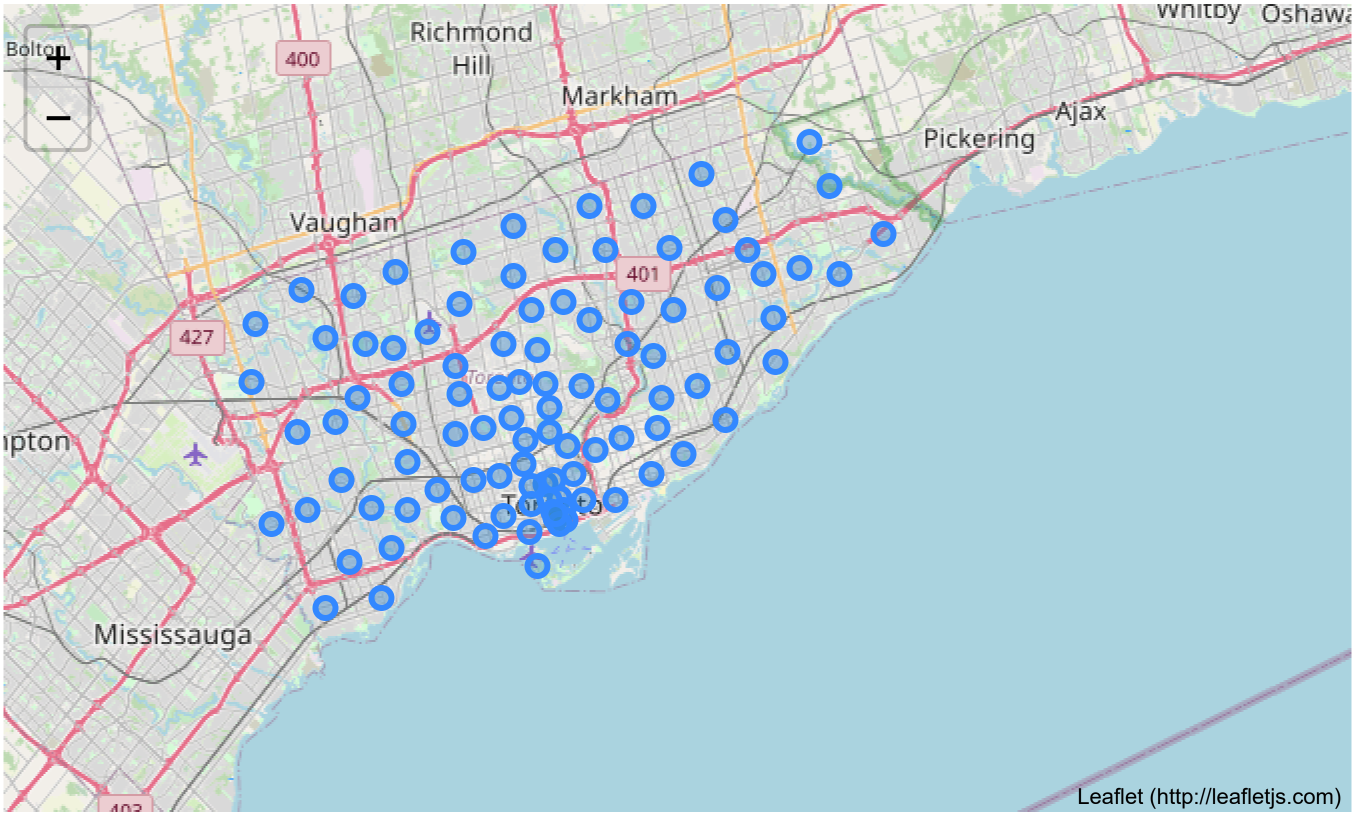
In [17]:

```
# Creating a map of Toronto Neighborhood using Folium Library and latitude
and longitude values
# Format of the Popup label = (Neighbourhood name , Borough name)
map_toronto = folium.Map(location=[toronto_latitude, toronto_longitude],
zoom_start=10)

for lat, long, borough, neigh in zip(Neighbourhood['Latitude'],
                                     Neighbourhood['Longitude'],
                                     Neighbourhood['Borough'],
                                     Neighbourhood['Neighbourhood']):
```

```
    label = '{}, {}'.format(neigh, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, long],
        radius=5,
        popup=label,
        colors='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.4,
        parse_html=False).add_to(map_toronto)
map_toronto
```

Out[17]:



3. Explore Neighbourhood Borough having Toronto name in it.

As we did for the all the neighborhood lets Visualize the Neiborhood of Scarborough

In [18]:

```
# Extracting all the neighbourhood of borough names with toronto and
creating a new dataframe
Toronto_Neighbourhood =
Neighbourhood[Neighbourhood.Borough.str.find('Toronto') != -1]
Toronto_Neighbourhood.reset_index(drop=True,).head()
```

Out[18]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M5A	Downtown Toronto	Regent Park, Harbourfront	43.6555	-79.3626

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
1	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.6641	-79.3889
2	M5B	Downtown Toronto	Garden District, Ryerson	43.6572	-79.3783
3	M5C	Downtown Toronto	St. James Town	43.6513	-79.3756
4	M4E	East Toronto	The Beaches	43.6784	-79.2941

The Toronto_Neighbourhood dataframe has 4 boroughs and 39 neighborhoods.

In [19]:

```
# Printing the information for New Data-set
print('The dataframe has {} boroughs and {} neighborhoods.'.format(
        len(Toronto_Neighbourhood['Borough'].unique()),
        Toronto_Neighbourhood.shape[0]
    )
)
```

```
The dataframe has 4 boroughs and 39 neighborhoods.
```

In [20]:

```
# Analyzing the dataframe and cheking for Unique Boroughs Neighbourhood
number
Toronto_Neighbourhood.Borough.value_counts()
```

Out[20]:

```
Downtown Toronto    19
Central Toronto      9
West Toronto         6
East Toronto         5
Name: Borough, dtype: int64
```

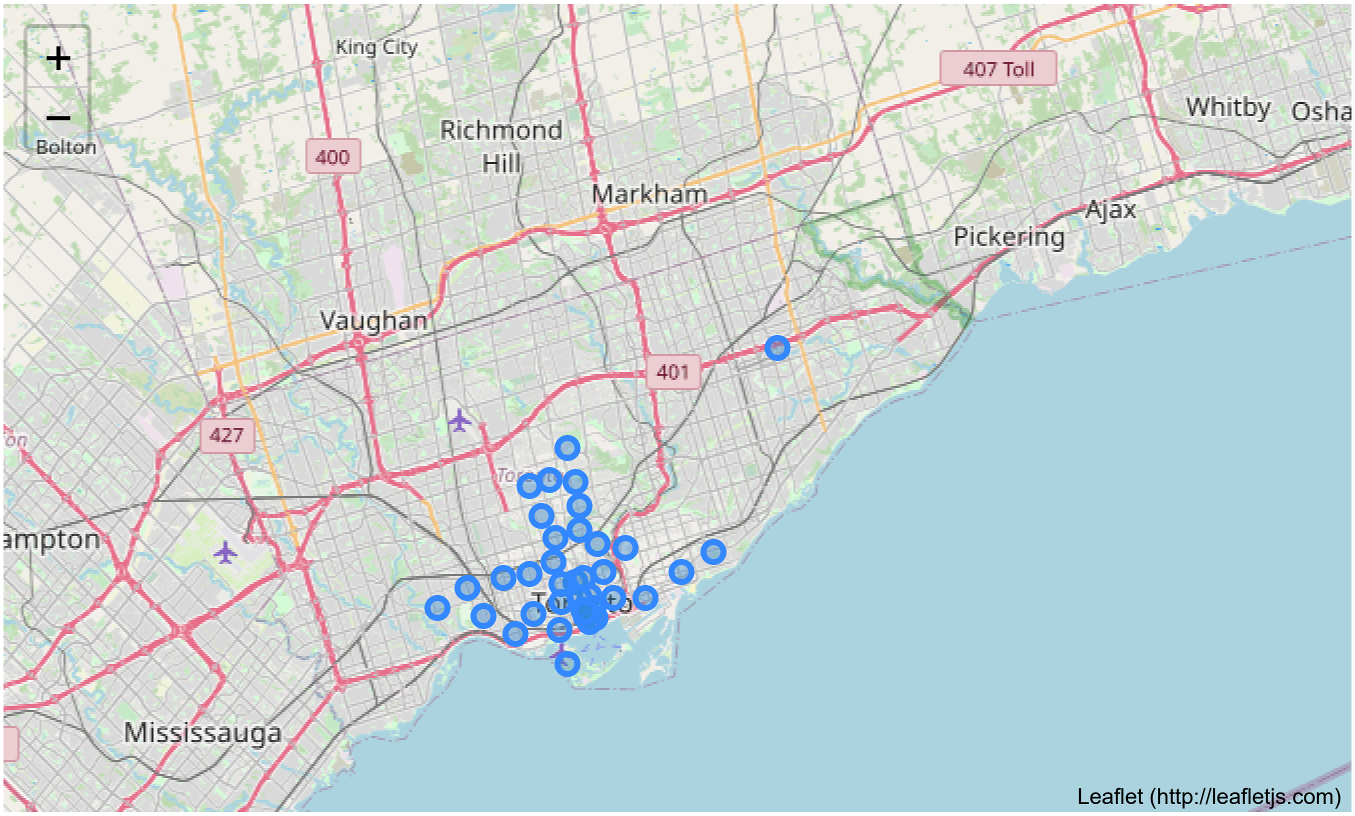
In [21]:

```
# Creating a map of Toronto Neighborhood using Folium Library and latitude
and longitude values
map_tor_neigh = folium.Map(location=[latitude, longitude], zoom_start=10)

for lat, long, borough, neigh in zip(Toronto_Neighbourhood['Latitude'],
                                     Toronto_Neighbourhood['Longitude'],
                                     Toronto_Neighbourhood['Borough'],

Toronto_Neighbourhood['Neighbourhood']):
    label = '{}, {}'.format(neigh, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, long],
        radius=5,
        popup=label,
        colors='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.4,
        parse_html=False).add_to(map_tor_neigh)
map_tor_neigh
```

Out[21]:



4. Utilizing the Foursquare Library to explore Analyze Each Neighborhood.

Define Foursquare Credentials and Version

In [22]:

```
CLIENT_ID = 'D4M1I000SE54SVKCXXL4NQGDHI5C4MSFPJ12LA3VNAATO0ZX' # your
Foursquare ID
CLIENT_SECRET = '3B2DHRPIPFPRIRYKT00OUDRSECRXIOOKVSGNQ5ZFV2HUUREQ' # your
Foursquare Secret
VERSION = '20200807' # Foursquare API version

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

```
Your credentails:
CLIENT_ID: D4M1I000SE54SVKCXXL4NQGDHI5C4MSFPJ12LA3VNAATO0ZX
CLIENT_SECRET:3B2DHRPIPFPRIRYKT00OUDRSECRXIOOKVSGNQ5ZFV2HUUREQ
```

In [23]:

```
# Exploring the First Neighbourhood in the dataFrame
Toronto_Neighbourhood = Toronto_Neighbourhood.reset_index(drop=True)
print('The first Row has same neighboor hood with same postalcode and hence
from here same co-ordinates : {}'.
      format(Toronto_Neighbourhood.loc[0,'Neighbourhood']))
```

```
# Select the first element as both have the same data Neighbourhood select
the first element
print('Selecting the first Neighbourhood : {}'.
      format(Toronto_Neighbourhood.loc[0,'Neighbourhood'].split(',')[0]))

print('Latitudes and Longitudes : {},
{}'.format(Toronto_Neighbourhood.loc[0,'Latitude'] , Toronto_Neighbourhood
                           .loc[0,'Longitude']))
```

```
The first Row has same neighboor hood with same postalcode and hence from here same co-o
rdinates : Regent Park, Harbourfront
Selecting the first Neighbourhood : Regent Park
Latitudes and Longitudes : 43.6555, -79.3626
```

Now, let's get the top 100 venues that are in Regent Park within a radius of 500 meters.

First, let's create the GET request URL. Name your URL **url**.

In [24]:

```
# type your answer here
LIMIT = 100 #limit the number of venues returned by Foursquare API
radius = 500 # define radius
# Create a URL to send the GET request
url = 'https://api.foursquare.com/v2/venues/explore?&client_id=
{}&client_secret={}&v={}&ll={},{}&\
radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET, VERSION,
                           Toronto_Neighbourhood.loc[0,'Latitude'],
                           Toronto_Neighbourhood.loc[0,'Longitude'],
                           radius, LIMIT )
url
```

Out[24]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=D4M1I000SE54SVKCXXL4NQGDHI5C4MS
FPJ12LA3VNAATO0ZX&client_secret=3B2DHRPIPFPRIRYKT00OUDRSECRXIOOKVSGNQ5ZFV2HUUREQ&v=20200
807&ll=43.6555,-79.3626&radius=500&limit=100'
```

In [25]:

```
# Send the GET request and examine the resutls
result = requests.get(url).json()
result
```

Out[25]:

```
{'meta': {'code': 200, 'requestId': '5f2e2ed388ec7f202b6e7409'},
 'response': {'headerLocation': 'Corktown',
  'headerFullLocation': 'Corktown, Toronto',
  'headerLocationGranularity': 'neighborhood',
  'totalResults': 23,
  'suggestedBounds': {'ne': {'lat': 43.660000004500006,
    'lng': -79.3563918719477},
   'sw': {'lat': 43.6509999955, 'lng': -79.36880812805231}},
  'groups': [{'type': 'Recommended Places',
    'name': 'recommended',
    'items': [{'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '53b8466a498e83df908c3f21',
  'name': 'Tandem Coffee',
  'location': {'address': '368 King St E',
   'crossStreet': 'at Trinity St',
   'lat': 43.65355870959944,
   'lng': -79.36180945913513,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65355870959944,
     'lng': -79.36180945913513}],
   'distance': 225,
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['368 King St E (at Trinity St)',
    'Toronto ON',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d1e0931735',
    'name': 'Coffee Shop',
    'pluralName': 'Coffee Shops',
    'shortName': 'Coffee Shop',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-53b8466a498e83df908c3f21-0'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '54ea41ad498e9a11e9e13308',
  'name': 'Roselle Desserts',
  'location': {'address': '362 King St E',
   'crossStreet': 'Trinity St',
   'lat': 43.653446723052674,
   'lng': -79.3620167174383,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.653446723052674,
     'lng': -79.3620167174383}],
   'distance': 233,
   'postalCode': 'M5A 1K9',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['362 King St E (Trinity St)',
    'Toronto ON M5A 1K9',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d16a941735',
    'name': 'Bakery',
    'pluralName': 'Bakeries',
    'shortName': 'Bakery',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/bakery_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-54ea41ad498e9a11e9e13308-1'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4af59046f964a520e0f921e3',
  'name': 'Figs Breakfast & Lunch',
  'location': {'address': '344 Queen St. E.',
   'crossStreet': 'at Parliament St.',
   'lat': 43.65567455427388,
   'lng': -79.3645032892494,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65567455427388,
     'lng': -79.3645032892494}],
   'distance': 154,
        'postalCode': 'M5A 1S8',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['344 Queen St. E. (at Parliament St.)',
         'Toronto ON M5A 1S8',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d143941735',
         'name': 'Breakfast Spot',
         'pluralName': 'Breakfast Spots',
         'shortName': 'Breakfast',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/breakfast_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4af59046f964a520e0f921e3-2'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4b58dd55f964a5208f6f28e3',
       'name': 'The Yoga Lounge',
       'location': {'address': '106 Sherbourne St.',
        'crossStreet': 'at Adelaide St. East',
        'lat': 43.65551522261721,
        'lng': -79.36495542526245,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.65551522261721,
          'lng': -79.36495542526245}],
        'distance': 189,
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['106 Sherbourne St. (at Adelaide St. East)',
         'Toronto ON',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d102941735',
         'name': 'Yoga Studio',
         'pluralName': 'Yoga Studios',
         'shortName': 'Yoga Studio',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/gym_yogastudio
_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4b58dd55f964a5208f6f28e3-3'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '5619551a498e9e35fce2256b',
       'name': 'Sumach Espresso',
       'location': {'address': '118 Sumach St',
        'crossStreet': 'at Shuter St',
        'lat': 43.65813540553308,
        'lng': -79.35951549011845,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.65813540553308,
          'lng': -79.35951549011845}],
        'distance': 384,
        'postalCode': 'M5A 3J9',
        'cc': 'CA',
        'neighborhood': 'Downtown Toronto',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['118 Sumach St (at Shuter St)',
         'Toronto ON M5A 3J9',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d1e0931735',
    'name': 'Coffee Shop',
    'pluralName': 'Coffee Shops',
    'shortName': 'Coffee Shop',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-5619551a498e9e35fce2256b-4'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '50760559e4b0e8c7babe2497',
  'name': 'Body Blitz Spa East',
  'location': {'address': '497 King Street East',
   'crossStreet': 'btwn Sackville St and Sumach St',
   'lat': 43.65473505045365,
   'lng': -79.35987433132891,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65473505045365,
     'lng': -79.35987433132891}],
   'distance': 235,
   'postalCode': 'M5A 1L9',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['497 King Street East (btwn Sackville St and Sumach St)',
    'Toronto ON M5A 1L9',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d1ed941735',
    'name': 'Spa',
    'pluralName': 'Spas',
    'shortName': 'Spa',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/spa_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-50760559e4b0e8c7babe2497-5'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4ada57aff964a520972121e3',
  'name': 'Sukhothai',
  'location': {'address': '274 Parliament St',
   'crossStreet': 'btwn Coatsworth St. & Shuter St.',
   'lat': 43.658444465539525,
   'lng': -79.36568085121628,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.658444465539525,
     'lng': -79.36568085121628}],
   'distance': 411,
   'postalCode': 'M5A 2Z6',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['274 Parliament St (btwn Coatsworth St. & Shuter St.)',
    'Toronto ON M5A 2Z6',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d149941735',
    'name': 'Thai Restaurant',
    'pluralName': 'Thai Restaurants',
    'shortName': 'Thai',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/thai_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-4ada57aff964a520972121e3-6'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '566e1294498e3f6629006bc3',
  'name': 'Dominion Pub and Kitchen',
  'location': {'address': '500 Queen Street East',
   'lat': 43.65691857501867,
   'lng': -79.35896684476664,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65691857501867,
     'lng': -79.35896684476664}],
   'distance': 332,
   'postalCode': 'M5A 1T9',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['500 Queen Street East',
    'Toronto ON M5A 1T9',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d11b941735',
    'name': 'Pub',
    'pluralName': 'Pubs',
    'shortName': 'Pub',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightlife/pub_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-566e1294498e3f6629006bc3-7'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4ae5b91ff964a520a6a121e3',
  'name': 'Morning Glory Cafe',
  'location': {'address': '457 King St. E',
   'crossStreet': 'Gilead Place',
   'lat': 43.653946942635294,
   'lng': -79.36114884214422,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.653946942635294,
     'lng': -79.36114884214422}],
   'distance': 208,
   'postalCode': 'M5A 1L6',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['457 King St. E (Gilead Place)',
    'Toronto ON M5A 1L6',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d143941735',
    'name': 'Breakfast Spot',
    'pluralName': 'Breakfast Spots',
    'shortName': 'Breakfast',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/breakfast_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []},
  'venuePage': {'id': '39686393'}},
 'referralId': 'e-0-4ae5b91ff964a520a6a121e3-8'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '51853a73498e4d97a8b20831',
  'name': 'Rooster Coffee',
          'location': {'address': '343 King St E',
           'crossStreet': 'btwn Princess & Berkeley St',
           'lat': 43.65189965670432,
           'lng': -79.36560912104514,
           'labeledLatLngs': [{'label': 'display',
             'lat': 43.65189965670432,
             'lng': -79.36560912104514}],
           'distance': 468,
           'postalCode': 'M5A 1L1',
           'cc': 'CA',
           'city': 'Toronto',
           'state': 'ON',
           'country': 'Canada',
           'formattedAddress': ['343 King St E (btwn Princess & Berkeley St)',
            'Toronto ON M5A 1L1',
            'Canada']},
          'categories': [{'id': '4bf58dd8d48988d1e0931735',
            'name': 'Coffee Shop',
            'pluralName': 'Coffee Shops',
            'shortName': 'Coffee Shop',
            'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
             'suffix': '.png'},
            'primary': True}],
          'photos': {'count': 0, 'groups': []}},
         'referralId': 'e-0-51853a73498e4d97a8b20831-9'},
        {'reasons': {'count': 0,
          'items': [{'summary': 'This spot is popular',
            'type': 'general',
            'reasonName': 'globalInteractionReason'}]},
         'venue': {'id': '4ade8ea8f964a5205a7621e3',
          'name': 'Berkeley Church',
          'location': {'address': '315 Queen St E',
           'crossStreet': 'at Berkeley St',
           'lat': 43.65512324174501,
           'lng': -79.36587330410705,
           'labeledLatLngs': [{'label': 'display',
             'lat': 43.65512324174501,
             'lng': -79.36587330410705}],
           'distance': 266,
           'postalCode': 'M5A 1S7',
           'cc': 'CA',
           'city': 'Toronto',
           'state': 'ON',
           'country': 'Canada',
           'formattedAddress': ['315 Queen St E (at Berkeley St)',
            'Toronto ON M5A 1S7',
            'Canada']},
          'categories': [{'id': '4bf58dd8d48988d171941735',
            'name': 'Event Space',
            'pluralName': 'Event Spaces',
            'shortName': 'Event Space',
            'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/eventspace
   _',
             'suffix': '.png'},
            'primary': True}],
          'photos': {'count': 0, 'groups': []}},
         'referralId': 'e-0-4ade8ea8f964a5205a7621e3-10'},
        {'reasons': {'count': 0,
          'items': [{'summary': 'This spot is popular',
            'type': 'general',
            'reasonName': 'globalInteractionReason'}]},
         'venue': {'id': '5612b1cc498e3dd742af0dc8',
          'name': 'Impact Kitchen',
          'location': {'address': '573 King St E',
           'crossStreet': 'at St Lawrence St',
           'lat': 43.65636850543279,
           'lng': -79.35697968750694,
           'labeledLatLngs': [{'label': 'display',
             'lat': 43.65636850543279,
             'lng': -79.35697968750694}],
        'distance': 462,
        'postalCode': 'M5A 4L3',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['573 King St E (at St Lawrence St)',
         'Toronto ON M5A 4L3',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d1c4941735',
         'name': 'Restaurant',
         'pluralName': 'Restaurants',
         'shortName': 'Restaurant',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/default_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-5612b1cc498e3dd742af0dc8-11'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4b8c46f3f964a5200cc832e3',
       'name': 'Alumnae Theatre',
       'location': {'address': '70 Berkley St.',
        'crossStreet': 'at Adelaide St.',
        'lat': 43.65275554626444,
        'lng': -79.36475283805089,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.65275554626444,
          'lng': -79.36475283805089}],
        'distance': 351,
        'postalCode': 'M5A 2W6',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['70 Berkley St. (at Adelaide St.)',
         'Toronto ON M5A 2W6',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d137941735',
         'name': 'Theater',
         'pluralName': 'Theaters',
         'shortName': 'Theater',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/arts_entertainment/p
erformingarts_theater_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4b8c46f3f964a5200cc832e3-12'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '58c7fbf7424f9373e6427e99',
       'name': 'Starbucks',
       'location': {'address': '351 King St E, 60',
        'lat': 43.651613,
        'lng': -79.364917,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.651613,
          'lng': -79.364917}],
        'distance': 471,
        'postalCode': 'M5A 1L1',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['351 King St E, 60',
         'Toronto ON M5A 1L1',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d1e0931735',
    'name': 'Coffee Shop',
    'pluralName': 'Coffee Shops',
    'shortName': 'Coffee Shop',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-58c7fbf7424f9373e6427e99-13'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4ad776eef964a520e20a21e3',
  'name': 'Mangia and Bevi Resto-Bar',
  'location': {'address': '260 King St E',
   'crossStreet': 'Princess',
   'lat': 43.652249517927686,
   'lng': -79.36635530500347,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.652249517927686,
     'lng': -79.36635530500347}],
   'distance': 471,
   'postalCode': 'M5R 4L5',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['260 King St E (Princess)',
    'Toronto ON M5R 4L5',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d110941735',
    'name': 'Italian Restaurant',
    'pluralName': 'Italian Restaurants',
    'shortName': 'Italian',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/italian_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-4ad776eef964a520e20a21e3-14'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '574c229e498ebb5c6b257902',
  'name': 'Cooper Koo Family YMCA',
  'location': {'address': '461 Cherry St',
   'lat': 43.65324910177244,
   'lng': -79.35800826343677,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65324910177244,
     'lng': -79.35800826343677}],
   'distance': 446,
   'postalCode': 'M5A 0H7',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['461 Cherry St', 'Toronto ON M5A 0H7', 'Canada']},
  'categories': [{'id': '52e81612bcbc57f1066b7a37',
    'name': 'Distribution Center',
    'pluralName': 'Distribution Centers',
    'shortName': 'Distributor',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/default_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-574c229e498ebb5c6b257902-15'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4e8b7fa1cc2112f67517660a',
  'name': 'The Extension Room',
  'location': {'address': '30 Eastern Ave',
   'crossStreet': 'Sackville St.',
   'lat': 43.65331304337331,
   'lng': -79.35972538072777,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65331304337331,
     'lng': -79.35972538072777}],
   'distance': 335,
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['30 Eastern Ave (Sackville St.)',
    'Toronto ON',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d175941735',
    'name': 'Gym / Fitness Center',
    'pluralName': 'Gyms or Fitness Centers',
    'shortName': 'Gym / Fitness',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/gym_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-4e8b7fa1cc2112f67517660a-16'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '50899387e4b055a756b254fc',
  'name': 'I Love Churros',
  'location': {'address': '1017 Dufferin St',
   'crossStreet': 'Bloor St West',
   'lat': 43.65836361668554,
   'lng': -79.36558277878224,
   'labeledLatLngs': [{'label': 'display',
     'lat': 43.65836361668554,
     'lng': -79.36558277878224}],
   'distance': 399,
   'postalCode': 'M6H 4B5',
   'cc': 'CA',
   'city': 'Toronto',
   'state': 'ON',
   'country': 'Canada',
   'formattedAddress': ['1017 Dufferin St (Bloor St West)',
    'Toronto ON M6H 4B5',
    'Canada']},
  'categories': [{'id': '4bf58dd8d48988d1cb941735',
    'name': 'Food Truck',
    'pluralName': 'Food Trucks',
    'shortName': 'Food Truck',
    'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/streetfood_',
     'suffix': '.png'},
    'primary': True}],
  'photos': {'count': 0, 'groups': []}},
 'referralId': 'e-0-50899387e4b055a756b254fc-17'},
{'reasons': {'count': 0,
  'items': [{'summary': 'This spot is popular',
    'type': 'general',
    'reasonName': 'globalInteractionReason'}]},
 'venue': {'id': '4ad8d551f964a5201f1521e3',
  'name': 'Vistek',
  'location': {'address': '496 Queen St E',
   'crossStreet': 'at Sumach',
   'lat': 43.65704640025721,
   'lng': -79.35966695607046,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.65704640025721,
          'lng': -79.35966695607046}],
        'distance': 292,
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['496 Queen St E (at Sumach)',
         'Toronto ON',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d122951735',
         'name': 'Electronics Store',
         'pluralName': 'Electronics Stores',
         'shortName': 'Electronics',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/technology_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4ad8d551f964a5201f1521e3-18'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4b8acdfdf964a520378232e3',
       'name': 'The Beer Store',
       'location': {'address': '28 River Street',
        'crossStreet': 'Queen St E',
        'lat': 43.657773,
        'lng': -79.3574632,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.657773,
          'lng': -79.3574632}],
        'distance': 484,
        'postalCode': 'M5A 3N9',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['28 River Street (Queen St E)',
         'Toronto ON M5A 3N9',
         'Canada']},
       'categories': [{'id': '5370f356bcbc57f1066c94c2',
         'name': 'Beer Store',
         'pluralName': 'Beer Stores',
         'shortName': 'Beer Store',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/nightlife/beergarden
_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4b8acdfdf964a520378232e3-19'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4b2cf78cf964a52077cb24e3',
       'name': 'Savoury Grounds',
       'location': {'address': '481 Queen St. East',
        'crossStreet': 'Sumach St',
        'lat': 43.656820970004496,
        'lng': -79.3589698353747,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.656820970004496,
          'lng': -79.3589698353747}],
        'distance': 327,
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['481 Queen St. East (Sumach St)',
         'Toronto ON',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d1e0931735',
         'name': 'Coffee Shop',
         'pluralName': 'Coffee Shops',
         'shortName': 'Coffee Shop',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/coffeeshop_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4b2cf78cf964a52077cb24e3-20'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '5292543a498ec4d4c99c0c64',
       'name': 'The Healthy Road',
       'location': {'address': '518 King Street East',
        'crossStreet': 'Sumach',
        'lat': 43.656264585886454,
        'lng': -79.35711882680904,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.656264585886454,
          'lng': -79.35711882680904}],
        'distance': 449,
        'postalCode': 'M5A 1M1',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['518 King Street East (Sumach)',
         'Toronto ON M5A 1M1',
         'Canada']},
       'categories': [{'id': '50aa9e744b90af0d42d5de0e',
         'name': 'Health Food Store',
         'pluralName': 'Health Food Stores',
         'shortName': 'Health Food Store',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/food_grocery
_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-5292543a498ec4d4c99c0c64-21'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '5834fe851470b05200a49735',
       'name': 'Wine Rack',
       'location': {'lat': 43.656573,
        'lng': -79.356928,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.656573,
          'lng': -79.356928}],
        'distance': 472,
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['Toronto ON', 'Canada']},
       'categories': [{'id': '4bf58dd8d48988d119951735',
         'name': 'Wine Shop',
         'pluralName': 'Wine Shops',
         'shortName': 'Wine Shop',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/food_wineshop
_',
          'suffix': '.png'},
         'primary': True}],
        'photos': {'count': 0, 'groups': []}},
       'referralId': 'e-0-5834fe851470b05200a49735-22'}]}]}}
```

In [26]:

```
# 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']
```

In [27]:

```
## Now we are ready to clean the json and structure it into a pandas
dataframe.
# result['response']['groups'][0]['items'][3]['venue']
venues = result['response']['groups'][0]['items']
nearby_venues = pd.json_normalize(venues)
print(nearby_venues.shape[0])
nearby_venues.head(2) # Flattening the json
```

```
23
```

Out[27]:

	referralId	reasons.count	reasons.items	venue.id	venue.name	venue.locati
0	e-0-53b8466a498e83df908c3f21-0	0	[{'summary': 'This spot is popular', 'type': '...	53b8466a498e83df908c3f21	Tandem Coffee	3(
1	e-0-54ea41ad498e9a11e9e13308-1	0	[{'summary': 'This spot is popular', 'type': '...	54ea41ad498e9a11e9e13308	Roselle Desserts	3(

In [28]:

```
# Extracting the categories information from the json file of each venue
categories = []
for x in range(nearby_venues.shape[0]):
    categories.append(pd.json_normalize(result['response']['groups'][0]
['items'][x]
                                         ['venue']['categories'])['name']
[0])
categories[0:5]
```

Out[28]:

```
['Coffee Shop', 'Bakery', 'Breakfast Spot', 'Yoga Studio', 'Coffee Shop']
```

In [29]:

```
# Create a dataframe to store the shop name, categories, and latitude
longitude
```

```
Column_names = ['Shop_Name','Category','Latitude','Longitude']
venues_df = pd.DataFrame(columns=Column_names)
for shop_name, Cat, lat, lng in zip(nearby_venues['venue.name'],
                                    categories,
                                    nearby_venues['venue.location.lat'],
                                   nearby_venues['venue.location.lng']):


    venues_df = venues_df.append({
                                  'Shop_Name':shop_name,
                                  'Category':Cat,
                                  'Latitude':lat,
                                  'Longitude':lng
                                          },ignore_index=True)



venues_df.head()
```

Out[29]:

	Shop_Name	Category	Latitude	Longitude
0	Tandem Coffee	Coffee Shop	43.653559	-79.361809
1	Roselle Desserts	Bakery	43.653447	-79.362017
2	Figs Breakfast & Lunch	Breakfast Spot	43.655675	-79.364503
3	The Yoga Lounge	Yoga Studio	43.655515	-79.364955
4	Sumach Espresso	Coffee Shop	43.658135	-79.359515

In [30]:

```
# Determining the number of venues returned by Foursquare API
print('{} venues were returned by Foursquare.'.format(venues_df.shape[0]) )
```

```
23 venues were returned by Foursquare.
```

In [31]:

```
# Creating a function to extract all the venues in all the neighbourhood in
Toronto_Neighbourhood
def get_Nearby_Venues(latitude, longitude, Neighbourhood):


    CLIENT_ID = 'D4M1I000SE54SVKCXXL4NQGDHI5C4MSFPJ12LA3VNAATO0ZX' # your
Foursquare ID
    CLIENT_SECRET = '3B2DHRPIPFPRIRYKT00OUDRSECRXIOOKVSGNQ5ZFV2HUUREQ' #
your Foursquare Secret
    VERSION = '20200807' # Foursquare API version
    # Create a dataframe to store the shop name, categories, and latitude
longitude
    Column_names =
['Neighbourhood','Venue_Name','Venue_Category','Latitude','Longitude']
    venues_df = pd.DataFrame(columns=Column_names)
    # Extracting Venues of Each Neighbourhood
    for lat, lng, neigh in zip(latitude, longitude, Neighbourhood):
        neigh = neigh.split(',')[0]
        LIMIT = 100 #limit the number of venues returned by Foursquare API
        radius = 500 # define radius
        # Create a URL to send the GET request
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id=
{}&client_secret={}&v={}&ll={},{}&\
        radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET, VERSION, lat,
lng, radius, LIMIT )
        # Send the GET request and examine the resutls
        result = requests.get(url).json()
        venues = result['response']['groups'][0]['items']
        nearby_venues = pd.json_normalize(venues)



        # Extracting category of each venue and sot
        categories = []
        for x in range(nearby_venues.shape[0]):
            categories.append(pd.json_normalize(result['response']
['groups'][0]['items'][x]
                                        ['venue']['categories'])['name']
[0])



        for shop_name, Cat, lat, lng in zip(nearby_venues['venue.name'],
                                        categories,

nearby_venues['venue.location.lat'],

nearby_venues['venue.location.lng']):

            venues_df = venues_df.append({'Neighbourhood':neigh,
                                          'Venue_Name':shop_name,
                                          'Venue_Category':Cat,
                                          'Latitude':lat,
                                          'Longitude':lng
                                           },ignore_index=True)
    return(venues_df)     # Returning the dataframe having the venues
information
```

In [32]:

```
# Using the above function to Extract all the venues of all the
Neighbourhood in Toronto_Neighbourhood datase
```

```
# get_Nearby_Venues(latitude, longitude, Neighbourhood):
Toronto_Neighbourhood_Venues =
get_Nearby_Venues(Toronto_Neighbourhood.Latitude,

Toronto_Neighbourhood.Longitude,

Toronto_Neighbourhood.Neighbourhood)
Toronto_Neighbourhood_Venues.head()
```

Out[32]:

	Neighbourhood	Venue_Name	Venue_Category	Latitude	Longitude
0	Regent Park	Tandem Coffee	Coffee Shop	43.653559	-79.361809
1	Regent Park	Roselle Desserts	Bakery	43.653447	-79.362017
2	Regent Park	Sumach Espresso	Coffee Shop	43.658135	-79.359515
3	Regent Park	Rooster Coffee	Coffee Shop	43.651900	-79.365609
4	Regent Park	Sukhothai	Thai Restaurant	43.658444	-79.365681

In [33]:

```
# This dataframe has 3848 venues information
Toronto_Neighbourhood_Venues.shape
```

Out[33]:

```
(3848, 5)
```

In [34]:

```
# Analyzing the number of Venues from each Neigghbourhood
Toronto_Neighbourhood_Venues.groupby('Neighbourhood').count()
```

Out[34]:

	Venue_Name	Venue_Category	Latitude	Longitude
Neighbourhood				
Berczy Park	100	100	100	100
Brockton	100	100	100	100
Business reply mail Processing Centre	100	100	100	100
CN Tower	100	100	100	100
Central Bay Street	100	100	100	100
Christie	100	100	100	100
Church and Wellesley	100	100	100	100
Commerce Court	100	100	100	100
Davisville	100	100	100	100
Davisville North	100	100	100	100
Dufferin	100	100	100	100
First Canadian Place	100	100	100	100
Forest Hill North & West	100	100	100	100
Garden District	100	100	100	100

	Venue_Name	Venue_Category	Latitude	Longitude
Neighbourhood				
Harbourfront East	88	88	88	88
High Park	100	100	100	100
India Bazaar	100	100	100	100
Kensington Market	100	100	100	100
Lawrence Park	100	100	100	100
Little Portugal	100	100	100	100
Moore Park	100	100	100	100
North Toronto West	100	100	100	100
Parkdale	100	100	100	100
Queen's Park	60	60	60	60
Regent Park	100	100	100	100
Richmond	100	100	100	100
Rosedale	100	100	100	100
Roselawn	100	100	100	100
Runnymede	100	100	100	100
St. James Town	200	200	200	200
Stn A PO Boxes	100	100	100	100
Studio District	100	100	100	100
Summerhill West	100	100	100	100
The Annex	100	100	100	100
The Beaches	100	100	100	100
The Danforth West	100	100	100	100
Toronto Dominion Centre	100	100	100	100
University of Toronto	100	100	100	100

In [35]:

```
# Let's find out how many unique categories can be curated from all the
returned venues
print('Unique Venues in all the Neighbourhood :{}'.
      format(len(Toronto_Neighbourhood_Venues.Venue_Category.unique())) )
```

```
Unique Venues in all the Neighbourhood :273
```

Analyze Each Neighborhood with respect to the Venues

In [36]:

```
# One hot encoding
Toronto_Venues_onehot =
pd.get_dummies(Toronto_Neighbourhood_Venues[['Venue_Category']],
                                                 prefix='',
                                                              prefix_sep='')
Toronto_Venues_onehot.head()
# add Neighbour column to the dataframe
Toronto_Venues_onehot['Neighbourhood'] =
Toronto_Neighbourhood_Venues['Neighbourhood']

# Move the Neighbourhood column to the first column
column_names = [Toronto_Venues_onehot.columns[-1]] +
list(Toronto_Venues_onehot.columns[:-1])
Toronto_Venues_onehot = Toronto_Venues_onehot[column_names]
Toronto_Venues_onehot.head()
```

Out[36]:

	Neighbourhood	Afghan Restaurant	Airport	American Restaurant	Amphitheater	Animal Shelter	Antique Shop	Aquarium	Arcade	Art Gallery
0	Regent Park	0	0	0	0	0	0	0	0	0
1	Regent Park	0	0	0	0	0	0	0	0	0
2	Regent Park	0	0	0	0	0	0	0	0	0
3	Regent Park	0	0	0	0	0	0	0	0	0
4	Regent Park	0	0	0	0	0	0	0	0	0

Next, let's group rows by neighborhood and by taking the mean of the frequency of occurrence of each category

In [40]:

```
Toronto_Venues_grouped =
Toronto_Venues_onehot.groupby('Neighbourhood').mean().reset_index()
Toronto_Venues_grouped.head()
```

Out[40]:

	Neighbourhood	Afghan Restaurant	Airport	American Restaurant	Amphitheater	Animal Shelter	Antique Shop	Aquarium	Arcade	Art Gallery
0	Berczy Park	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	0.02
1	Brockton	0.0	0.0	0.01	0.0	0.0	0.0	0.0	0.0	0.00
2	Business reply mail Processing Centre	0.0	0.0	0.01	0.0	0.0	0.0	0.0	0.0	0.00
3	CN Tower	0.0	0.0	0.01	0.0	0.0	0.0	0.0	0.0	0.01
4	Central Bay Street	0.0	0.0	0.00	0.0	0.0	0.0	0.0	0.0	0.03

In [41]:

```
# Conforming the new size of the dataFrame
Toronto_Venues_grouped.shape
```

Out[41]:

```
(38, 274)
```

In [42]:

```
# Let's print each neighborhood along with the top 5 most common venues
num_top_venues = 5

for hood in Toronto_Venues_grouped['Neighbourhood']:
    print("----"+hood+"----")
    temp = Toronto_Venues_grouped[Toronto_Venues_grouped['Neighbourhood']
== 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_top_venues))
    print('\n')
```

```
----Berczy Park----
                 venue  freq
0          Coffee Shop  0.12
1                Hotel  0.06
2           Restaurant  0.05
3  Japanese Restaurant  0.04
4               Bakery  0.04


----Brockton----
         venue  freq
0         Café  0.06
1  Coffee Shop  0.06
2          Bar  0.04
3       Bakery  0.04
4   Restaurant  0.04


----Business reply mail Processing Centre----
                venue  freq
0         Coffee Shop  0.07
1  Chinese Restaurant  0.04
2         Supermarket  0.04
3      Clothing Store  0.04
4   Indian Restaurant  0.04


----CN Tower----
         venue  freq
0  Coffee Shop  0.09
1  Yoga Studio  0.06
2         Café  0.05
3          Gym  0.05
4   Restaurant  0.04


----Central Bay Street----
                venue  freq
0         Coffee Shop  0.13
1      Clothing Store  0.06
2                Café  0.03
3  Italian Restaurant  0.03
4         Art Gallery  0.03


----Christie----
```

```
                           venue  freq
0                           Café  0.12
1                    Coffee Shop  0.06
2                            Bar  0.05
3  Vegetarian / Vegan Restaurant  0.04
4              Korean Restaurant  0.04


----Church and Wellesley----
                 venue  freq
0          Coffee Shop  0.10
1  Japanese Restaurant  0.06
2     Sushi Restaurant  0.04
3                 Café  0.04
4           Restaurant  0.04


----Commerce Court----
                 venue  freq
0          Coffee Shop  0.09
1                Hotel  0.09
2                 Café  0.07
3     Asian Restaurant  0.03
4  American Restaurant  0.03


----Davisville----
                venue  freq
0         Coffee Shop  0.08
1  Italian Restaurant  0.06
2              Bakery  0.04
3          Restaurant  0.04
4                Park  0.04


----Davisville North----
                venue  freq
0         Coffee Shop  0.08
1  Italian Restaurant  0.07
2              Bakery  0.05
3                Café  0.04
4                Park  0.04


----Dufferin----
                venue  freq
0                Café  0.14
1         Coffee Shop  0.11
2  Italian Restaurant  0.09
3                 Bar  0.06
4                Park  0.05


----First Canadian Place----
                 venue  freq
0          Coffee Shop  0.09
1                Hotel  0.09
2                 Café  0.07
3     Asian Restaurant  0.03
4  American Restaurant  0.03


----Forest Hill North & West----
                       venue  freq
0                Coffee Shop  0.10
1         Italian Restaurant  0.07
2           Sushi Restaurant  0.06
3  Middle Eastern Restaurant  0.03
4                       Bank  0.03
```

```
----Garden District----
                 venue  freq
0       Clothing Store  0.08
1          Coffee Shop  0.08
2  Japanese Restaurant  0.03
3   Italian Restaurant  0.03
4                 Café  0.03


----Harbourfront East----
             venue  freq
0             Park  0.12
1      Coffee Shop  0.06
2             Café  0.06
3  Harbor / Marina  0.05
4    Boat or Ferry  0.05


----High Park----
                venue  freq
0                Café  0.10
1         Coffee Shop  0.08
2              Bakery  0.06
3                 Bar  0.06
4  Italian Restaurant  0.05


----India Bazaar----
         venue  freq
0  Coffee Shop  0.06
1         Park  0.06
2        Beach  0.05
3         Café  0.05
4      Brewery  0.05


----Kensington Market----
                           venue  freq
0                           Café  0.08
1                    Coffee Shop  0.06
2  Vegetarian / Vegan Restaurant  0.06
3                            Bar  0.04
4                   Dessert Shop  0.03


----Lawrence Park----
                venue  freq
0         Coffee Shop  0.09
1  Italian Restaurant  0.06
2                Café  0.05
3    Sushi Restaurant  0.05
4          Restaurant  0.03


----Little Portugal----
         venue  freq
0         Café  0.07
1          Bar  0.06
2       Bakery  0.05
3   Restaurant  0.04
4  Pizza Place  0.04


----Moore Park----
                venue  freq
0                Park  0.09
1  Italian Restaurant  0.09
2                Café  0.05
3    Sushi Restaurant  0.05
```

```
4         Coffee Shop  0.04


----North Toronto West----
                venue  freq
0  Italian Restaurant  0.11
1         Coffee Shop  0.10
2                Café  0.04
3              Bakery  0.04
4    Sushi Restaurant  0.03


----Parkdale----
         venue  freq
0  Coffee Shop  0.07
1       Bakery  0.06
2         Park  0.06
3          Bar  0.05
4         Café  0.04


----Queen's Park----
                 venue  freq
0          Coffee Shop  0.15
1                 Café  0.05
2          Yoga Studio  0.03
3      Bubble Tea Shop  0.03
4  Japanese Restaurant  0.03


----Regent Park----
         venue  freq
0  Coffee Shop  0.12
1         Café  0.05
2      Theater  0.04
3   Restaurant  0.04
4    Gastropub  0.04


----Richmond----
         venue  freq
0  Coffee Shop  0.12
1        Hotel  0.09
2         Café  0.07
3   Restaurant  0.04
4      Theater  0.03


----Rosedale----
                venue  freq
0  Italian Restaurant  0.08
1                Park  0.08
2         Coffee Shop  0.06
3                Café  0.06
4                 Spa  0.04


----Roselawn----
                 venue  freq
0          Coffee Shop  0.12
1   Italian Restaurant  0.07
2           Restaurant  0.04
3     Sushi Restaurant  0.03
4  Japanese Restaurant  0.03


----Runnymede----
                venue  freq
0                Café  0.06
1              Bakery  0.06
```

```
2         Coffee Shop  0.06
3                Park  0.06
4  Italian Restaurant  0.06


----St. James Town----
                 venue  freq
0          Coffee Shop  0.07
1                 Café  0.06
2  Japanese Restaurant  0.04
3           Restaurant  0.04
4                 Park  0.03


----Stn A PO Boxes----
         venue  freq
0  Coffee Shop  0.10
1         Café  0.08
2        Hotel  0.05
3   Restaurant  0.04
4         Park  0.03


----Studio District----
                   venue  freq
0            Coffee Shop  0.09
1                   Café  0.06
2                   Park  0.06
3  Vietnamese Restaurant  0.05
4                Brewery  0.04


----Summerhill West----
                venue  freq
0  Italian Restaurant  0.09
1                Park  0.06
2                Café  0.06
3         Coffee Shop  0.06
4    Sushi Restaurant  0.04


----The Annex----
                           venue  freq
0             Italian Restaurant  0.06
1                           Café  0.04
2                    Coffee Shop  0.04
3  Vegetarian / Vegan Restaurant  0.04
4                           Park  0.04


----The Beaches----
         venue  freq
0         Park  0.07
1        Beach  0.06
2  Coffee Shop  0.06
3         Café  0.05
4          Pub  0.05


----The Danforth West----
                   venue  freq
0                   Café  0.09
1       Greek Restaurant  0.07
2                   Park  0.06
3                 Bakery  0.05
4  Vietnamese Restaurant  0.04


----Toronto Dominion Centre----
                 venue  freq
```

```
0          Coffee Shop  0.11
1                Hotel  0.09
2                 Café  0.08
3  American Restaurant  0.03
4         Concert Hall  0.03


----University of Toronto----
                            venue  freq
0                            Café  0.08
1                     Coffee Shop  0.06
2  Vegetarian / Vegan Restaurant  0.05
3                      Restaurant  0.04
4                          Bakery  0.04
```

In [43]:

```
# Putting the above output in pandas dataframe
# Creating a function for sorting the venues in descending order of
frequency
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 [44]:

```
# Now let's create the new dataframe and display the top 10 venues for each
neighborhood.
num_top_venues = 10

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

# create columns according to number of top venues
columns = ['Neighbourhood']
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['Neighbourhood'] =
Toronto_Venues_grouped['Neighbourhood']

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

Toronto_Venues_grouped.iloc[ind, :],
num_top_venues)

neighborhoods_venues_sorted.head()
```

Out[44]:

	Neighbourhood	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
0	Berczy Park	Coffee Shop	Hotel	Restaurant	Japanese Restaurant	Beer Bar	Café	Bakery	Italian Restaurant
1	Brockton	Coffee Shop	Café	Restaurant	Bar	Bakery	Athletics & Sports	Asian Restaurant	Gift Shop
2	Business reply mail Processing Centre	Coffee Shop	Restaurant	Supermarket	Chinese Restaurant	Clothing Store	Indian Restaurant	Bakery	Caribbean Restaurant
3	CN Tower	Coffee Shop	Yoga Studio	Gym	Café	Restaurant	Italian Restaurant	Park	Bar
4	Central Bay Street	Coffee Shop	Clothing Store	Italian Restaurant	Café	Plaza	Art Gallery	Department Store	Electronics Store

5. Cluster Neighbourhood

Run *k*-means to cluster the neighborhood into 5 clusters.

In [45]:

```
# set number of clusters
kclusters = 5

toronto_grouped_clustering = Toronto_Venues_grouped.drop('Neighbourhood',
1)

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

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]
```

Out[45]:

```
array([3, 2, 1, 1, 1, 4, 1, 3, 1, 1])
```

In [46]:

```
# Processing the Neighbourhood column having two names with with the same
postal code to be replaced
# with the first one
for index, item in zip(Toronto_Neighbourhood['Neighbourhood'].index,
                       Toronto_Neighbourhood['Neighbourhood']):

    Toronto_Neighbourhood['Neighbourhood'][index] =
Toronto_Neighbourhood.loc[index,'Neighbourhood'].split(',')[0]
```

```
# Toronto_Neighbourhood.loc[0,'Neighbourhood'].split(',')[0]
Toronto_Neighbourhood.head()
```

```
C:\Users\rajat\anaconda3\lib\site-packages\ipykernel_launcher.py:6: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_
guide/indexing.html#returning-a-view-versus-a-copy
```

Out[46]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude
0	M5A	Downtown Toronto	Regent Park	43.6555	-79.3626
1	M7A	Downtown Toronto	Queen's Park	43.6641	-79.3889
2	M5B	Downtown Toronto	Garden District	43.6572	-79.3783
3	M5C	Downtown Toronto	St. James Town	43.6513	-79.3756
4	M4E	East Toronto	The Beaches	43.6784	-79.2941

In [47]:

```
# Creating a new dataframe that includes the cluster as well as the top 10
venues of each neighbourhood
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

Toronto_merged =  Toronto_Neighbourhood

# merge Toronto_grouped with Toronto_Neighbourhood to add
latitude/longitude for each neighborhood
Toronto_merged =
Toronto_merged.join(neighborhoods_venues_sorted.set_index('Neighbourhood'),
                                                    on='Neighbourhood')


Toronto_merged.head() # check the last columns!
```

Out[47]:

	Postal Code	Borough	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue
0	M5A	Downtown Toronto	Regent Park	43.6555	-79.3626	1	Coffee Shop	Café	Bakery	Restaurant
1	M7A	Downtown Toronto	Queen's Park	43.6641	-79.3889	1	Coffee Shop	Café	Bubble Tea Shop	Japanese Restaurant
2	M5B	Downtown Toronto	Garden District	43.6572	-79.3783	1	Clothing Store	Coffee Shop	Italian Restaurant	Cosmetics Shop
3	M5C	Downtown Toronto	St. James Town	43.6513	-79.3756	1	Coffee Shop	Café	Restaurant	Japanese Restaurant
4	M4E	East Toronto	The Beaches	43.6784	-79.2941	2	Park	Beach	Coffee Shop	Pub

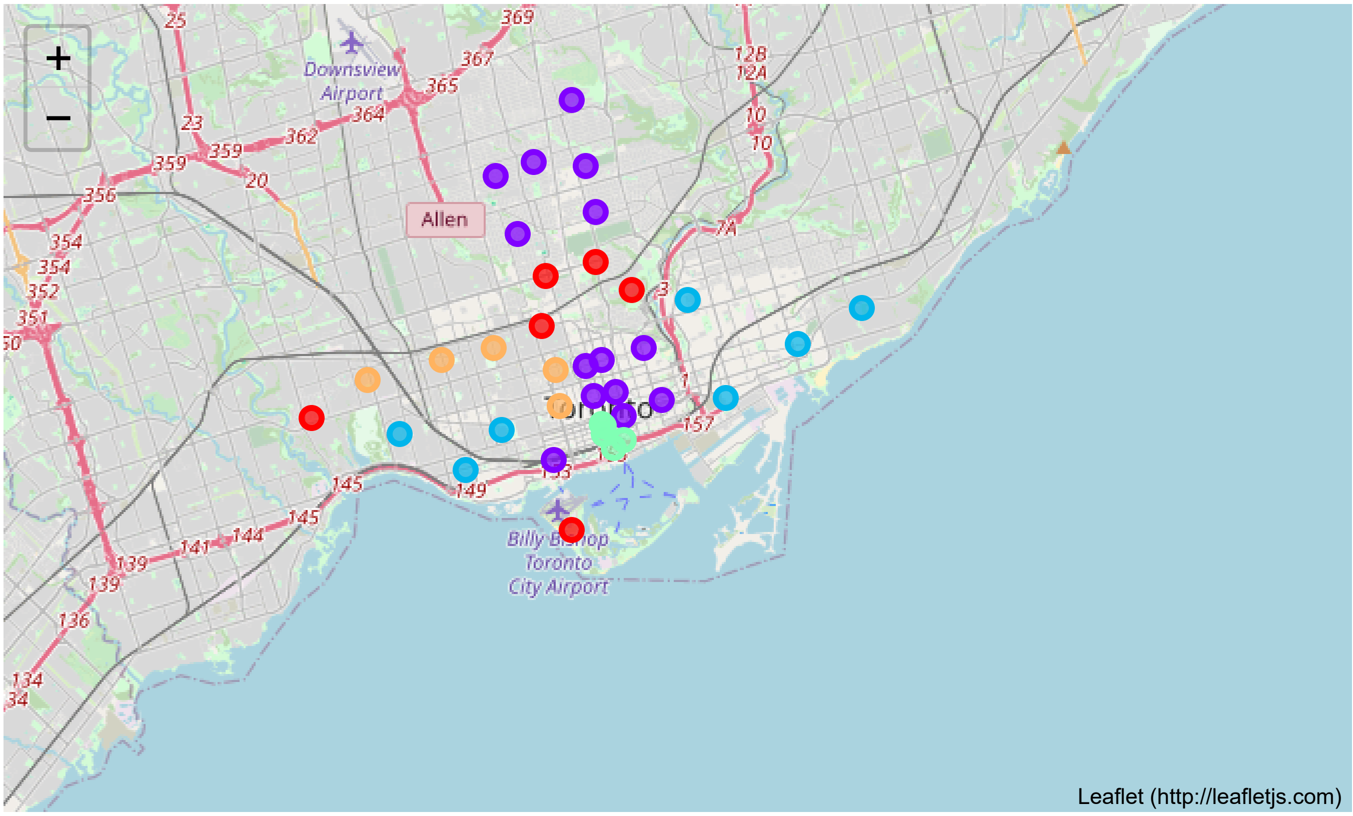
In [48]:

```
# Finally Visualizing the Results
# create map
map_clusters = folium.Map(location=[43.6555, -79.3626], zoom_start=11)

# 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['Longitude'], Toronto_merged['Neighbourhood'],
Toronto_merged['Cluster Labels']):
    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[48]:



6. Examine Clusters

Now, examining each cluster and determine the discriminating venue categories that distinguish each cluster.

Cluster 1

In [49]:

```
Toronto_merged.loc[Toronto_merged['Cluster Labels'] == 0,
Toronto_merged.columns[[1] + list(range(6, Toronto_merged.shape[1]))]]
```

Out[49]:

	Borough	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 Mos Commo Venu
10	Downtown Toronto	Park	Café	Coffee Shop	Harbor / Marina	Boat or Ferry	Hotel	Restaurant	Pizza Place	Brewer
24	Central Toronto	Italian Restaurant	Restaurant	Café	Coffee Shop	Vegetarian / Vegan Restaurant	Park	Spa	Middle Eastern Restaurant	Boutiqu
28	West Toronto	Coffee Shop	Park	Café	Italian Restaurant	Bakery	Bar	Brewery	Pizza Place	Gastropu
29	Central Toronto	Park	Italian Restaurant	Café	Sushi Restaurant	Bakery	Coffee Shop	Dessert Shop	Restaurant	India Restaurar
31	Central Toronto	Italian Restaurant	Park	Coffee Shop	Café	Sushi Restaurant	Grocery Store	Yoga Studio	Middle Eastern Restaurant	Vegetaria / Vega Restaurar
33	Downtown Toronto	Park	Italian Restaurant	Café	Coffee Shop	Restaurant	Spa	Yoga Studio	Ice Cream Shop	Pu

Cluster 2

In [50]:

```
Toronto_merged.loc[Toronto_merged['Cluster Labels'] == 1,
Toronto_merged.columns[[1] + list(range(6, Toronto_merged.shape[1]))]]
```

Out[50]:

	Borough	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
0	Downtown Toronto	Coffee Shop	Café	Bakery	Restaurant	Gastropub	Theater	Breakfast Spot	Diner
1	Downtown Toronto	Coffee Shop	Café	Bubble Tea Shop	Japanese Restaurant	Diner	Yoga Studio	Sushi Restaurant	Bookstore
2	Downtown Toronto	Clothing Store	Coffee Shop	Italian Restaurant	Cosmetics Shop	Café	Japanese Restaurant	Bubble Tea Shop	Lingerie Store
3	Downtown Toronto	Coffee Shop	Café	Restaurant	Japanese Restaurant	Park	Gastropub	Bakery	Diner
6	Downtown Toronto	Coffee Shop	Clothing Store	Italian Restaurant	Café	Plaza	Art Gallery	Department Store	Electronics Store
18	Central Toronto	Coffee Shop	Italian Restaurant	Café	Sushi Restaurant	Bakery	Restaurant	Ice Cream Shop	Japanese Restaurant

	Borough	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
19	Central Toronto	Coffee Shop	Italian Restaurant	Restaurant	Japanese Restaurant	Café	Bakery	Bagel Shop	Sushi Restaurant
20	Central Toronto	Coffee Shop	Italian Restaurant	Bakery	Indian Restaurant	Café	Park	Restaurant	Pizza Place
21	Central Toronto	Coffee Shop	Italian Restaurant	Sushi Restaurant	Gym	Park	Yoga Studio	Restaurant	Ice Cream Shop
23	Central Toronto	Italian Restaurant	Coffee Shop	Café	Bakery	Fast Food Restaurant	Park	Sushi Restaurant	Restaurant
26	Central Toronto	Coffee Shop	Italian Restaurant	Café	Gym	Restaurant	Bakery	Park	Sushi Restaurant
32	Downtown Toronto	Coffee Shop	Yoga Studio	Gym	Café	Restaurant	Italian Restaurant	Park	Bar
35	Downtown Toronto	Coffee Shop	Café	Restaurant	Japanese Restaurant	Park	Gastropub	Bakery	Diner
37	Downtown Toronto	Coffee Shop	Japanese Restaurant	Restaurant	Sushi Restaurant	Café	Gay Bar	Diner	Mediterranean Restaurant
38	East Toronto	Coffee Shop	Restaurant	Supermarket	Chinese Restaurant	Clothing Store	Indian Restaurant	Bakery	Caribbean Restaurant

Cluster 3

In [51]:

```
Toronto_merged.loc[Toronto_merged['Cluster Labels'] == 2,
Toronto_merged.columns[[1] + list(range(6, Toronto_merged.shape[1]))]]
```

Out[51]:

	Borough	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 Mo Comm Ven
4	East Toronto	Park	Beach	Coffee Shop	Pub	Café	Bakery	Indian Restaurant	Breakfast Spot	Ice Crea Sh
11	West Toronto	Café	Bar	Bakery	Pizza Place	Asian Restaurant	Coffee Shop	Restaurant	Yoga Studio	Cockt E
12	East Toronto	Café	Greek Restaurant	Park	Bakery	Vietnamese Restaurant	Ice Cream Shop	American Restaurant	Italian Restaurant	Gastrop
14	West Toronto	Coffee Shop	Café	Restaurant	Bar	Bakery	Athletics & Sports	Asian Restaurant	Gift Shop	Itali Restaura
15	East Toronto	Coffee Shop	Park	Café	Beach	Brewery	Bar	Pizza Place	Indian Restaurant	Breakfa Sp
17	East Toronto	Coffee Shop	Park	Café	Vietnamese Restaurant	Brewery	Bar	Bakery	Pizza Place	Ice Crea Sh
25	West Toronto	Coffee Shop	Park	Bakery	Bar	Café	Restaurant	Eastern European Restaurant	Sushi Restaurant	Indi Restaura

Cluster 4

In [52]:

```
Toronto_merged.loc[Toronto_merged['Cluster Labels'] == 3,
Toronto_merged.columns[[1] + list(range(6, Toronto_merged.shape[1]))]]
```

Out[52]:

	Borough	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 Mos Commo Venu
5	Downtown Toronto	Coffee Shop	Hotel	Restaurant	Japanese Restaurant	Beer Bar	Café	Bakery	Italian Restaurant	Seafoo Restaurar
8	Downtown Toronto	Coffee Shop	Hotel	Café	Restaurant	Theater	Gym	Concert Hall	Beer Bar	Cosmetic Sho
13	Downtown Toronto	Coffee Shop	Hotel	Café	American Restaurant	Concert Hall	Restaurant	Seafood Restaurant	Gym	Japanes Restaurar
16	Downtown Toronto	Hotel	Coffee Shop	Café	Japanese Restaurant	Asian Restaurant	American Restaurant	Restaurant	Gym	Italia Restaurar
34	Downtown Toronto	Coffee Shop	Café	Hotel	Restaurant	Park	Beer Bar	Japanese Restaurant	Art Gallery	Vegetaria / Vega Restaurar
36	Downtown Toronto	Hotel	Coffee Shop	Café	Japanese Restaurant	Asian Restaurant	American Restaurant	Restaurant	Gym	Italia Restaurar

Cluster 5

In [53]:

```
Toronto_merged.loc[Toronto_merged['Cluster Labels'] == 4,
Toronto_merged.columns[[1] + list(range(6, Toronto_merged.shape[1]))]]
```

Out[53]:

	Borough	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 Mos Commo Venu
7	Downtown Toronto	Café	Coffee Shop	Bar	Vegetarian / Vegan Restaurant	Korean Restaurant	Grocery Store	Italian Restaurant	Dessert Shop	Restaurar
9	West Toronto	Café	Coffee Shop	Italian Restaurant	Bar	Park	Cocktail Bar	Bakery	Restaurant	Brewer
22	West Toronto	Café	Coffee Shop	Bar	Bakery	Italian Restaurant	Brewery	Gastropub	Restaurant	Pizz Plac
27	Downtown Toronto	Café	Coffee Shop	Vegetarian / Vegan Restaurant	Bakery	Restaurant	Bar	Italian Restaurant	Bookstore	Ice Crear Sho
30	Downtown Toronto	Café	Vegetarian / Vegan Restaurant	Coffee Shop	Bar	Dessert Shop	Ice Cream Shop	Mexican Restaurant	Arts & Crafts Store	Grocer Stor

In [ ]:

```

```