In [87]:
```
#conda update -n base -c defaults conda
```

In [88]:
```
# pip install folium
# !conda install -c conda-forge folium=0.5.0 --yes
```

In [89]:
```
# pip install geopy
```

In [ ]:
```
!pip install beautifulsoup4
!pip install lxml
```

In [17]:
```
import pandas as pd
from bs4 import BeautifulSoup
import requests
import numpy as np
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

import folium # map rendering library

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

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

In [82]:
```
source = requests.get("https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M").text
soup = BeautifulSoup(source, 'lxml')

table = soup.find("table")
table_rows = table.tbody.find_all("tr")

# print(table_rows)
data = []
for tr in table_rows:
    td = tr.find_all("td")
    row = [tr.text for tr in td]
#     print(row)
#     print('_____________________________________')
# Only process the cells that have an assigned borough. Ignore cells with a borough that is Not assigned.
#     print(row)
    for temp in row:
#         print(temp)
        if temp != []:
            temp = temp.strip()
            temp1 = temp[0:3]
```

```
            temp2 = temp[3:]

            if temp2.strip() != "Not assigned":
                res = temp2.split("(")
                temp2 = res[0]
                temp3 = res[1].rstrip(')')
                temp3 = temp3.replace(")",",")
                temp3 = temp3.replace(" / ",", ")
                res = [temp1, temp2, temp3]
            data.append(res)
# print(data)
# Dataframe with 3 columns
df = pd.DataFrame(data, columns = ["PostalCode", "Borough", "Neighborhood"])

df.drop_duplicates(subset = ["PostalCode","Borough","Neighborhood"],keep = False, inplace = True)
df.head(200)
```

Out[82]:

	PostalCode	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
8	M9A	Etobicoke	Islington Avenue
...	...	...	...
147	M4W	Downtown Toronto	Rosedale
151	M8W	Etobicoke	Alderwood, Long Branch
152	M9W	EtobicokeNorthwest	Clairville, Humberwood, Woodbine Downs, West H...
156	M4X	Downtown Toronto	St. James Town, Cabbagetown
168	M7Y	East TorontoBusiness reply mail Processing Cen...	Enclave of M4L

70 rows × 3 columns

2) you have built a dataframe of the postal code of each neighborhood along with the borough name and neighborhood name, in order to utilize the Foursquare location data, we need to get the latitude and the longitude coordinates of each neighborhood. Put the Latitude and Longtitude to Dataframes.

In [84]:

```
geo_data = pd.read_csv("Geospatial_Coordinates.csv")
geo_data.head()
```

Out[84]:

	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 [86]:
```
df_toronto = pd.merge(df, geo_data, how='left', left_on = 'PostalCode', right_on = 'Postal Code')
# remove the "Postal Code" column
df_toronto.drop("Postal Code", axis=1, inplace=True)
df_toronto.head()
```

Out[86]:

	PostalCode	Borough	Neighborhood	Latitude	Longitude
0	M3A	North York	Parkwoods	43.753259	-79.329656
1	M4A	North York	Victoria Village	43.725882	-79.315572
2	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
3	M6A	North York	Lawrence Manor, Lawrence Heights	43.718518	-79.464763
4	M9A	Etobicoke	Islington Avenue	43.667856	-79.532242

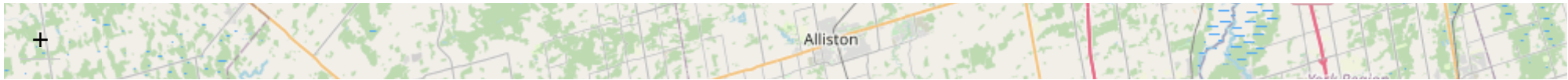
3) Explore and cluster the neighborhoods in Toronto. You can decide to work with only boroughs that contain the word Toronto and then replicate the same analysis we did to the New York City data. It is up to you.

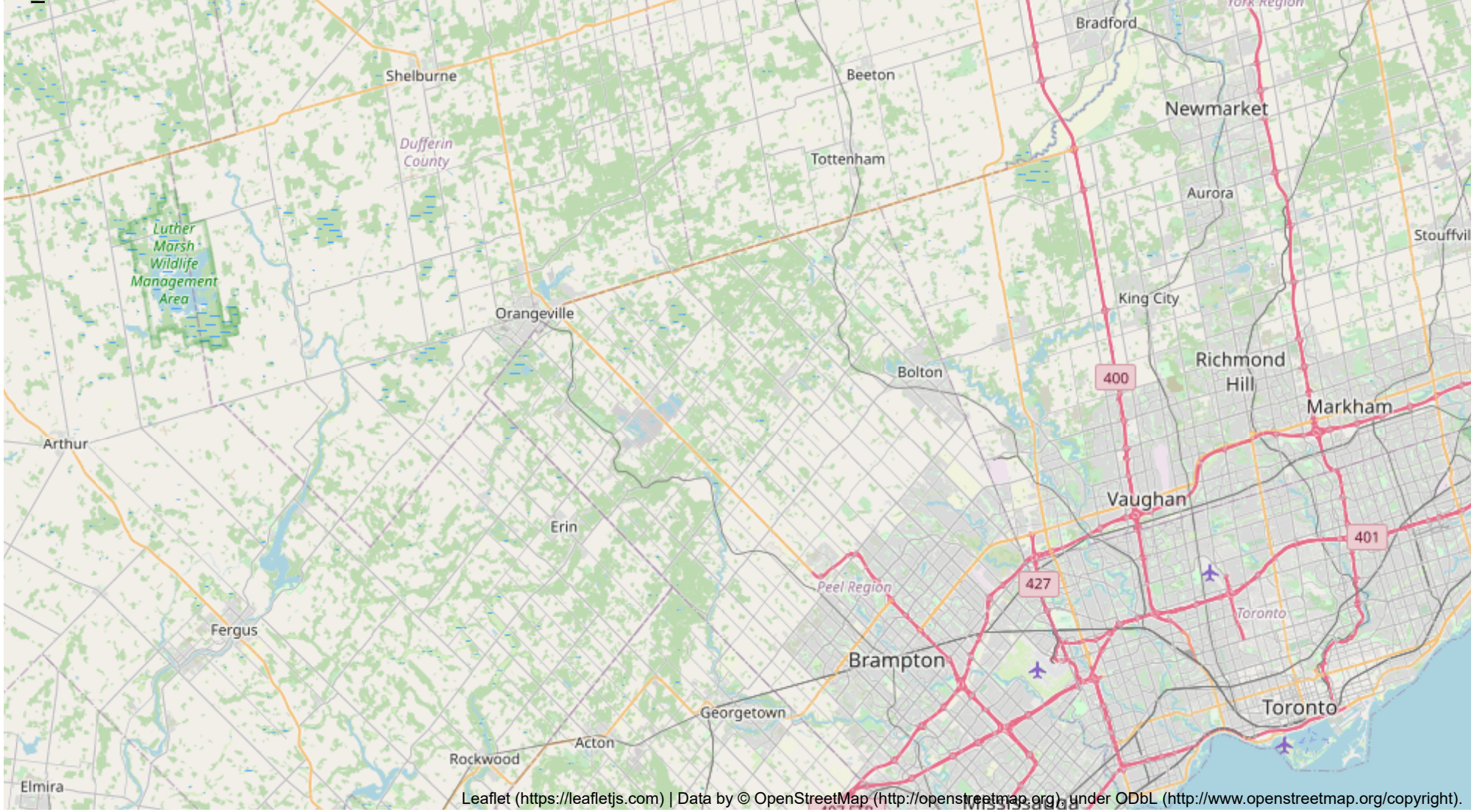
In [90]:
```
address = "Toronto, ON"

geolocator = Nominatim(user_agent="toronto_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Toronto city are {}, {}.'.format(latitude, longitude))
```

```
The geograpical coordinate of Toronto city are 43.6534817, -79.3839347.
```

In [91]:
```
# create map of Toronto using latitude and longitude values
map_toronto = folium.Map(location=[latitude, longitude], zoom_start=10)
map_toronto
```

Out[91]:



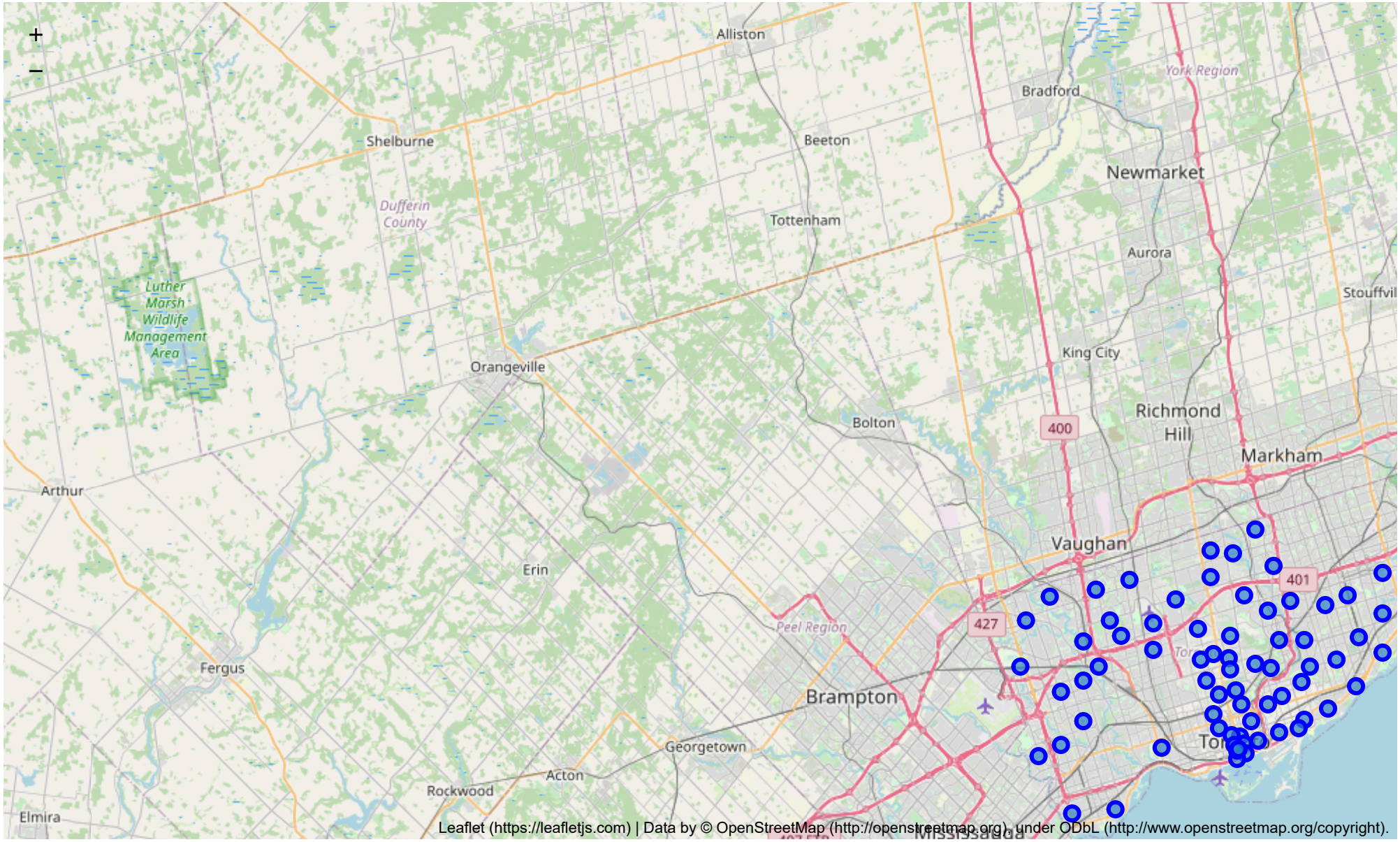


In [92]:

```
for lat, lng, borough, neighborhood in zip(
        df_toronto['Latitude'],
        df_toronto['Longitude'],
        df_toronto['Borough'],
        df_toronto['Neighborhood']):
    label = '{}, {}'.format(neighborhood, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
```

```
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_toronto)

map_toronto
```

Out[92]:



Map of Toronto City

In [93]:

```
# "denc" = [D]owntown Toronto, [E]ast Toronto, [N]orth Toronto, [C]entral Toronto
df_toronto_denc = df_toronto[df_toronto['Borough'].str.contains("Toronto")].reset_index(drop=True)
```

```
df_toronto_denc.head()
```

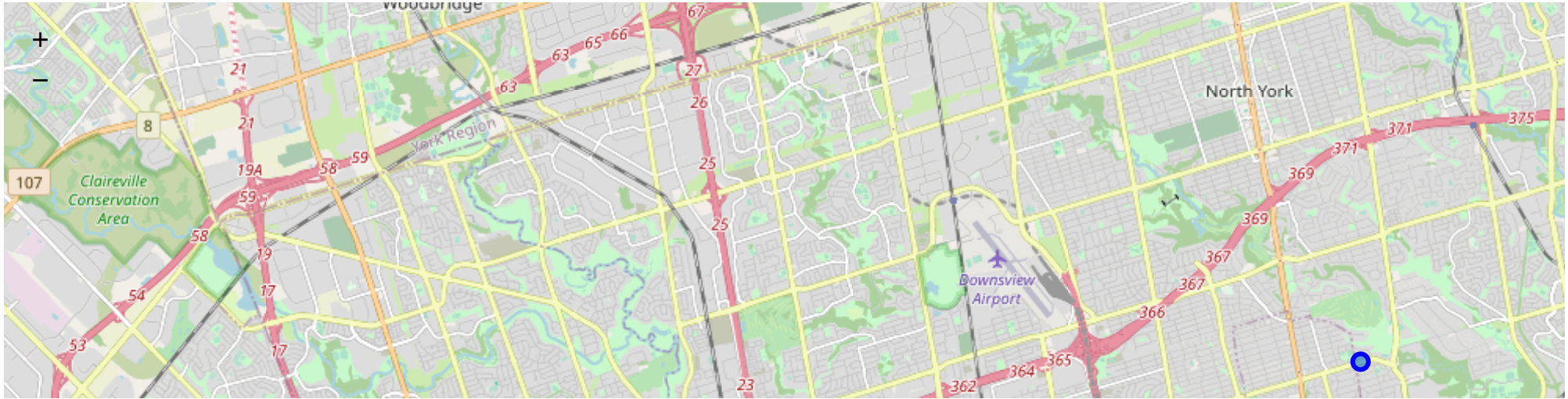
Out[93]:

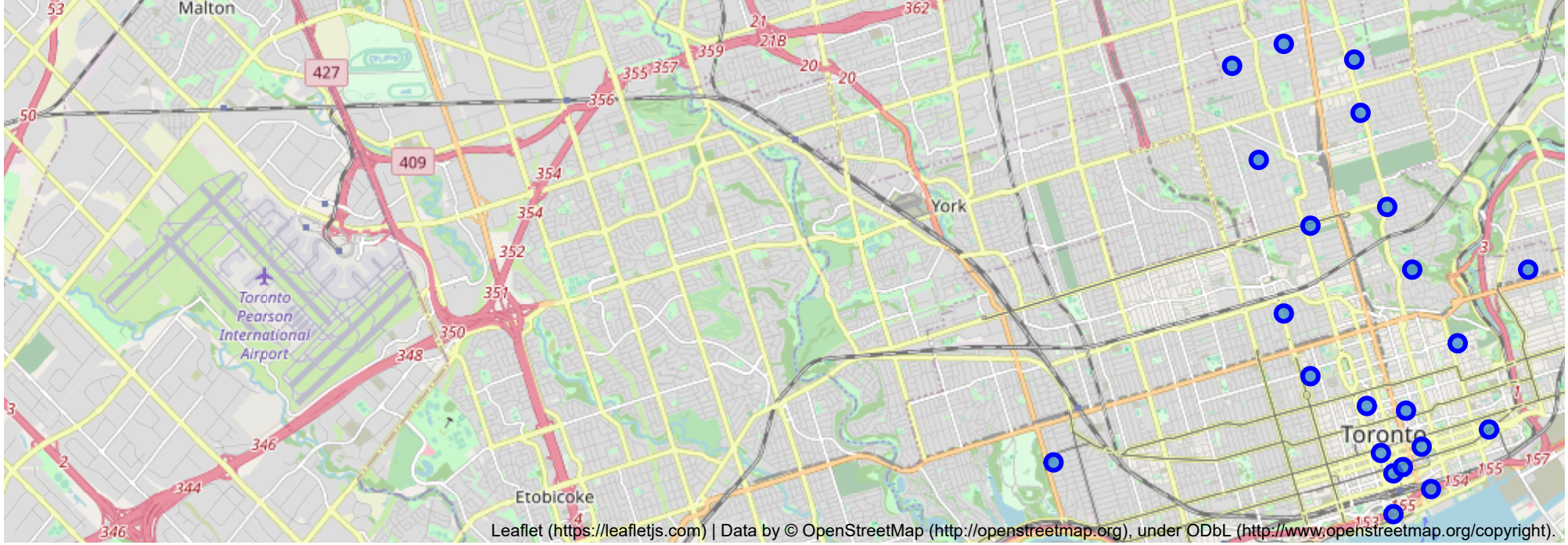
	PostalCode	Borough	Neighborhood	Latitude	Longitude
0	M5A	Downtown Toronto	Regent Park, Harbourfront	43.654260	-79.360636
1	M5B	Downtown Toronto	Garden District, Ryerson	43.657162	-79.378937
2	M5C	Downtown Toronto	St. James Town	43.651494	-79.375418
3	M4E	East Toronto	The Beaches	43.676357	-79.293031
4	M5E	Downtown Toronto	Berczy Park	43.644771	-79.373306

In [94]:

```
map_toronto_denc = folium.Map(location=[latitude, longitude], zoom_start=12)
for lat, lng, borough, neighborhood in zip(
        df_toronto_denc['Latitude'],
        df_toronto_denc['Longitude'],
        df_toronto_denc['Borough'],
        df_toronto_denc['Neighborhood']):
    label = '{}, {}'.format(neighborhood, borough)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(map_toronto_denc)

map_toronto_denc
```

Out[94]:




Malton
427
409
York
Toronto Pearson International Airport
Etobicoke
Toronto
Leaflet (https://leafletjs.com) | Data by © OpenStreetMap (http://openstreetmap.org), under ODbL (http://www.openstreetmap.org/copyright).