This notebook will be mainly used for the capstone project offered by Coursera.

In [1]:
```
import numpy as np
import pandas as pd
print("Hello Capstone Project Course!")
```

```
Hello Capstone Project Course!
```

In [7]:
```
import urllib.request
url="https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
page = urllib.request.urlopen(url)
from bs4 import BeautifulSoup
soup = BeautifulSoup(page, "lxml")
all_tables=soup.find_all("table")
right_table=soup.find('table', class_='wikitable sortable')
```

In [8]:
```
A=[]
B=[]
C=[]
for row in right_table.findAll('tr'):
    cells=row.findAll('td')
    if len(cells)==3:
        A.append(cells[0].find(text=True).rstrip())

        B.append(cells[1].find(text=True).rstrip())
        C.append(cells[2].find(text=True).rstrip())
```

In [100]:
```
data=pd.DataFrame(A,columns=['Postal Code'])
data['Borough']=B
data['Neighborhood']=C
df=data.copy()
df['Borough'].replace(to_replace = "Not assigned" , value = np.nan,inplace=True)
df.dropna(axis=0,inplace=True)
df.reset_index(inplace = True)
df_post=df.groupby("Postal Code",as_index=False).agg(lambda x:','.join(set(x)))
df_post
```

87	M7Y	East Toronto	Business reply mail Processing Centre, South C...
88	M8V	Etobicoke	New Toronto, Mimico South, Humber Bay Shores
89	M8W	Etobicoke	Alderwood, Long Branch
90	M8X	Etobicoke	The Kingsway, Montgomery Road, Old Mill North
91	M8Y	Etobicoke	Old Mill South, King's Mill Park, Sunnylea, Hu...
92	M8Z	Etobicoke	Mimico NW, The Queensway West, South of Bloor,...
93	M9A	Etobicoke	Islington Avenue, Humber Valley Village
94	M9B	Etobicoke	West Deane Park, Princess Gardens, Martin Grov...
95	M9C	Etobicoke	Eringate, Bloordale Gardens, Old Burnhamthorpe...
96	M9L	North York	Humber Summit
97	M9M	North York	Humberlea, Emery
98	M9N	York	Weston
99	M9P	Etobicoke	Westmount

GETTING LATITUDE AND LONGITUDE FOR CORRESPONDING POSTAL CODES

In [9]:
```
location=pd.read_csv("https://cocl.us/Geospatial_data")
location
```

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
...	...	...	...
98	M9N	43.706876	-79.518188
99	M9P	43.696319	-79.532242
100	M9R	43.688905	-79.554724
101	M9V	43.739416	-79.588437
102	M9W	43.706748	-79.594054

103 rows × 3 columns

In [101]:
```
df_post["latitude"]=location['Latitude'].values
df_post["longitude"]=location['Longitude'].values
df_post
```

72	M6B	North York	Glencairn	43.709577	-79.445073
73	M6C	York	Humewood-Cedarvale	43.693781	-79.428191
74	M6E	York	Caledonia-Fairbanks	43.689026	-79.453512
75	M6G	Downtown Toronto	Christie	43.669542	-79.422564
76	M6H	West Toronto	Dufferin, Dovercourt Village	43.669005	-79.442259
77	M6J	West Toronto	Little Portugal, Trinity	43.647927	-79.419750
78	M6K	West Toronto	Brockton, Parkdale Village, Exhibition Place	43.636847	-79.428191
79	M6L	North York	North Park, Maple Leaf Park, Upwood Park	43.713756	-79.490074
80	M6M	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn	43.691116	-79.476013
81	M6N	York	Runnymede, The Junction North	43.673185	-79.487262

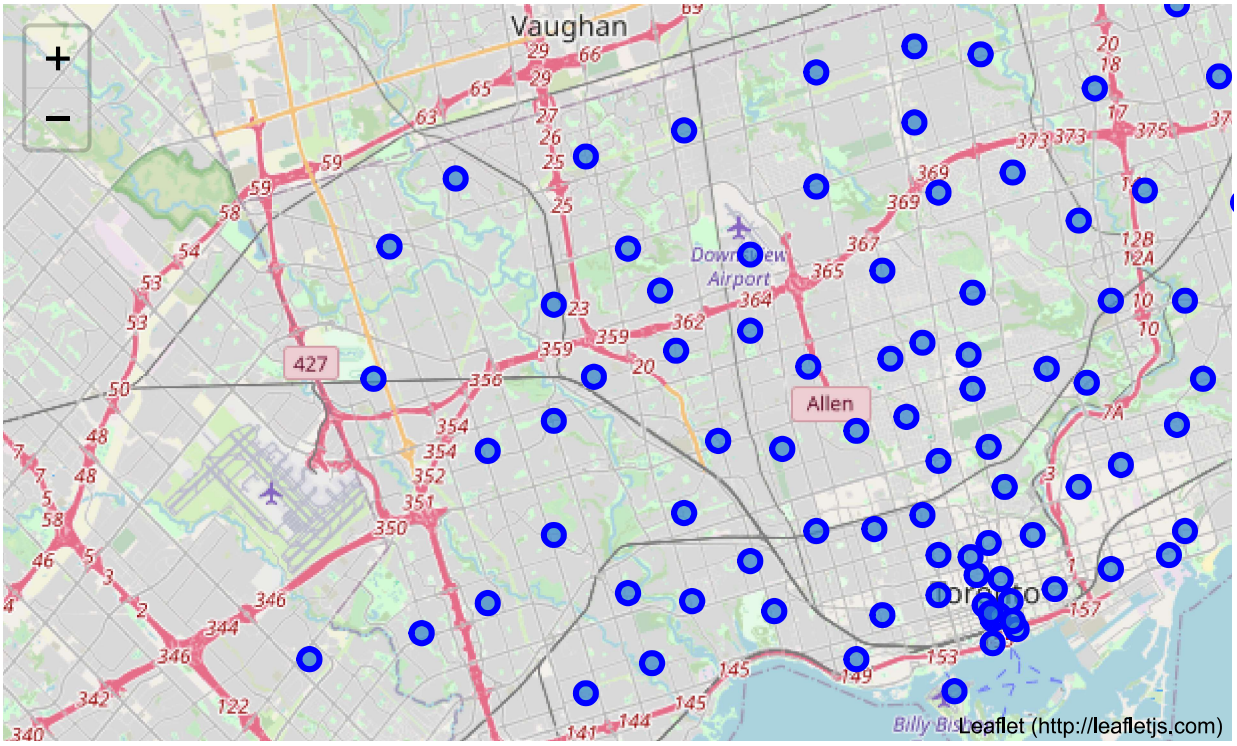
In [13]:

```
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

import json

!conda install -c conda-forge geopy --yes
from geopy.geocoders import Nominatim # convert an address into latitude and long

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

# Matplotlib and associated plotting modules
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
import folium # map rendering library

print('Libraries imported.')
```

```
Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

# All requested packages already installed.

Collecting package metadata (current_repodata.json): ...working... done
Solving environment: ...working... done

# All requested packages already installed.

Libraries imported.
```

In [102]:
```
toronto_latitude = 43.6532; toronto_longitude = -79.3832
map_toronto = folium.Map(location = [toronto_latitude, toronto_longitude], zoom_s
for lat, lng, borough, neighborhood in zip(df_post['latitude'], df_post['longitud
    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).add_to(map_toronto)

map_toronto
```

Out[102]:



In [23]:
```
CLIENT_ID = 'N1UH4XLAXAKK23JT2OYNKYVJKNLKN5IJ4T5JRLVKBWE1RXCE' # your Foursquare
CLIENT_SECRET = 'GFGY513UVIVSVRRRMSYMBPN3X2YGJAY05UAGRWH3SSZ14CFP' # your Foursqu
VERSION = '20190530' # Foursquare API version
```

In [103]:
```
scarborough_data = df_post[df_post['Borough'] == 'Scarborough'].reset_index(drop=
scarborough_data.head(7)
```

Out[103]:

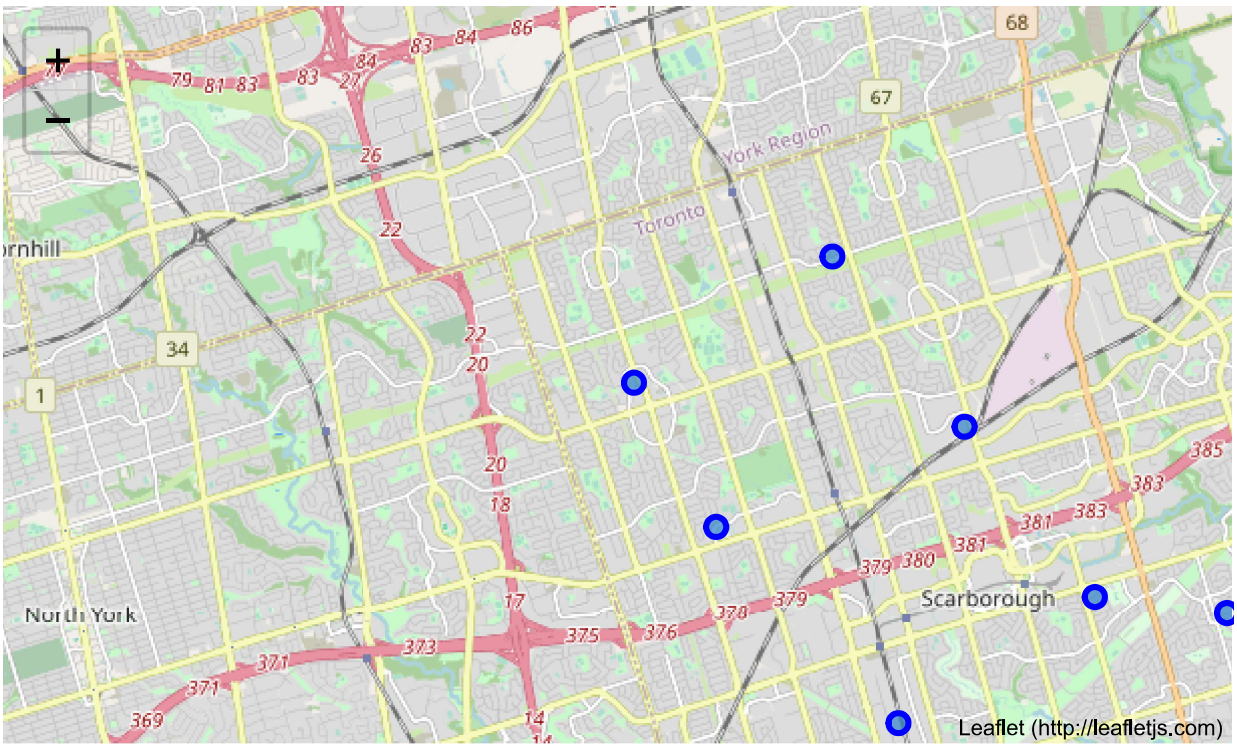
	Postal Code	Borough	Neighborhood	latitude	longitude
0	M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353
1	M1C	Scarborough	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
3	M1G	Scarborough	Woburn	43.770992	-79.216917
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476
5	M1J	Scarborough	Scarborough Village	43.744734	-79.239476
6	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park	43.727929	-79.262029

In [26]:
```
address_scar = 'Scarborough,Toronto'
latitude_scar = 43.773077
longitude_scar = -79.257774
print('The geograpical coordinate of Scarborough are {}, {}.'.format(latitude_sca
```

```
The geograpical coordinate of Scarborough are 43.773077, -79.257774.
```

In [104]:
```
map_scarb = folium.Map(location=[latitude_scar, longitude_scar], zoom_start=12)

# add markers to map
for lat, lng, label in zip(scarborough_data['latitude'], scarborough_data['longit
    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).add_to(map_scarb)
map_scarb
```

Out[104]:



In [105]:
```
neighborhood_latitude = scarborough_data.loc[0, 'latitude'] # neighbourhood latit
neighborhood_longitude = scarborough_data.loc[0, 'longitude'] # neighbourhood lor

neighborhood_name = scarborough_data.loc[0, 'Neighborhood'] # neighbourhood name

print('Latitude and longitude values of "{}" are {}, {}.'.format(neighborhood_nam
                                                                 neighborhood_latit
                                                                 neighborhood_longi
```

```
Latitude and longitude values of "Malvern, Rouge" are 43.806686299999996, -79.1
9435340000001.
```

In [33]:
```
= 100
s = 500
'https://api.foursquare.com/v2/venues/explore?client_id={}&client_secret={}&ll={}
```

In [34]:
```
results = requests.get(url).json()
```

In [35]:
```
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 [106]:
```
import json # library to handle JSON files
from pandas.io.json import json_normalize # tranform JSON file into a pandas data

venues = results['response']['groups'][0]['items']
nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venu
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(10)
```

```
C:\Users\mahesh venkat reesu\anaconda3\lib\site-packages\ipykernel_launcher.py:
5: FutureWarning: pandas.io.json.json_normalize is deprecated, use pandas.json_
normalize instead
  """
```

Out[106]:

	name	categories	lat	lng
0	SEPHORA	Cosmetics Shop	43.775017	-79.258109
1	Disney Store	Toy / Game Store	43.775537	-79.256833
2	American Eagle Store	Clothing Store	43.776012	-79.258334
3	DAVIDsTEA	Tea Room	43.776320	-79.258688
4	Tommy Hilfiger	Clothing Store	43.776015	-79.257369
5	Chipotle Mexican Grill	Mexican Restaurant	43.776410	-79.258069
6	St. Louis Bar & Grill	Bar	43.774157	-79.253808
7	Hot Topic	Clothing Store	43.775450	-79.257929
8	St. Andrews Fish & Chips	Fish & Chips Shop	43.771865	-79.252645
9	Shoppers Drug Mart	Pharmacy	43.773305	-79.251662

In [39]:
```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
40 venues were returned by Foursquare.
```

In [107]:
```
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_secr
        CLIENT_ID,
        CLIENT_SECRET,
        VERSION,
        lat,
        lng,
        radius,
        LIMIT)

    # make the GET request
    results = requests.get(url).json()["response"]['groups'][0]['items']
    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 venu
nearby_venues.columns = ['Neighborhood',
              'Neighborhood Latitude',
              'Neighborhood Longitude',
              'Venue',
              'Venue Latitude',
              'Venue Longitude',
              'Venue Category']

return(nearby_venues)
```

Get venues for each neighborhood in Scarborough

In [108]:

```
scarborough_venues = getNearbyVenues(names=scarborough_data['Neighborhood'],
                                     latitudes=scarborough_data['latitude'],
                                     longitudes=scarborough_data['longitude']
                                    )
```

```
Malvern, Rouge
Rouge Hill, Port Union, Highland Creek
Guildwood, Morningside, West Hill
Woburn
Cedarbrae
Scarborough Village
Kennedy Park, Ionview, East Birchmount Park
Golden Mile, Clairlea, Oakridge
Cliffside, Cliffcrest, Scarborough Village West
Birch Cliff, Cliffside West
Dorset Park, Wexford Heights, Scarborough Town Centre
Wexford, Maryvale
Agincourt
Clarks Corners, Tam O'Shanter, Sullivan
Milliken, Agincourt North, Steeles East, L'Amoreaux East
Steeles West, L'Amoreaux West
Upper Rouge
```

In [45]:
```
scarborough_venues.head(10)
```

Out[45]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Malvern, Rouge	43.806686	-79.194353	Wendy's	43.807448	-79.199056	Fast Food Restaurant
1	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497	Chris Effects Painting	43.784343	-79.163742	Construction & Landscaping
2	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497	Great Shine Window Cleaning	43.783145	-79.157431	Home Service
3	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497	Royal Canadian Legion	43.782533	-79.163085	Bar
4	Guildwood, Morningside, West Hill	43.763573	-79.188711	RBC Royal Bank	43.766790	-79.191151	Bank
5	Guildwood, Morningside, West Hill	43.763573	-79.188711	G & G Electronics	43.765309	-79.191537	Electronics Store
6	Guildwood, Morningside, West Hill	43.763573	-79.188711	Sail Sushi	43.765951	-79.191275	Restaurant
7	Guildwood, Morningside, West Hill	43.763573	-79.188711	Big Bite Burrito	43.766299	-79.190720	Mexican Restaurant
8	Guildwood, Morningside, West Hill	43.763573	-79.188711	Enterprise Rent-A-Car	43.764076	-79.193406	Rental Car Location
9	Guildwood, Morningside, West Hill	43.763573	-79.188711	Wood Floor Polishing Inc	43.766500	-79.185207	Moving Target

In [109]:
```
scarborough_venues.groupby('Neighborhood').count()
```

Out[109]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Agincourt	5	5	5	5	5	5
Birch Cliff, Cliffside West	4	4	4	4	4	4
Cedarbrae	8	8	8	8	8	8
Clarks Corners, Tam O'Shanter, Sullivan	12	12	12	12	12	12
Cliffside, Cliffcrest, Scarborough Village West	3	3	3	3	3	3
Dorset Park, Wexford Heights, Scarborough Town Centre	5	5	5	5	5	5
Golden Mile, Clairlea, Oakridge	8	8	8	8	8	8
Guildwood, Morningside, West Hill	9	9	9	9	9	9
Kennedy Park, Ionview, East Birchmount Park	5	5	5	5	5	5
Malvern, Rouge	1	1	1	1	1	1
Milliken, Agincourt North, Steeles East, L'Amoreaux East	3	3	3	3	3	3
Rouge Hill, Port Union, Highland Creek	3	3	3	3	3	3
Scarborough Village	2	2	2	2	2	2
Steeles West, L'Amoreaux West	16	16	16	16	16	16
Wexford, Maryvale	4	4	4	4	4	4
Woburn	5	5	5	5	5	5

In [126]:
```
print('There are {} uniques categories.'.format(len(scarborough_venues['Venue Cat
```

```
There are 55 uniques categories.
```

In [110]:
```
# one hot encoding
scarborough_onehot = pd.get_dummies(scarborough_venues[['Venue Category']], prefi

# add neighborhood column back to dataframe
scarborough_onehot['Neighborhood'] = scarborough_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [scarborough_onehot.columns[-1]] + list(scarborough_onehot.colum
scarborough_onehot = scarborough_onehot[fixed_columns]

scarborough_onehot.head()
```

Out[110]:

American Restaurant	Athletics & Sports	Bakery	Bank	Bar	Breakfast Spot	Bus Station	Café	Caribbean Restaurant	Chinese Restaurant	Clothing Store
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0

In [48]:
```
scarborough_onehot.shape
```

Out[48]: (93, 56)

In [111]:
```
scarborough_grouped = scarborough_onehot.groupby('Neighborhood').mean().reset_inc
scarborough_grouped
```

Out[111]:

Middle Eastern Restaurant	Motel	Moving Target	Noodle House	Park	Pet Store	Pharmacy	Pizza Place	Playground	Renta Ca Locatio
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.083333	0.000000	0.0	0.166667	0.166667	0.000000	0.00000
0.00	0.333333	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.2	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.125000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.111111	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.11111
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.333333	0.0	0.000000	0.000000	0.333333	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.500000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.062500	0.062500	0.000000	0.00000
0.25	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000
0.00	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	0.000000	0.000000	0.00000

In [112]:
```
scarborough_grouped.shape
```

Out[112]:
```
(16, 56)
```

each neighborhood along with the top 5 most common venues

In [113]:
```
num_top_venues = 5

for hood in scarborough_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = scarborough_grouped[scarborough_grouped['Neighborhood'] == hood].T.res
    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(r
    print('\n')
```

```
4         Intersection   0.0


----Steeles West, L'Amoreaux West----
                  venue  freq
0    Chinese Restaurant  0.12
1  Fast Food Restaurant  0.12
2         Grocery Store  0.12
3        Discount Store  0.06
4     Electronics Store  0.06


----Wexford, Maryvale----
                       venue  freq
0                     Bakery  0.25
1             Breakfast Spot  0.25
2                 Smoke Shop  0.25
3  Middle Eastern Restaurant  0.25
4        American Restaurant  0.00
```

descending order of venues

In [114]:
```
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 [130]:
```
num_top_venues = 10

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'] = scarborough_grouped['Neighborhood']

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

neighborhoods_venues_sorted
```

Out[130]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Com V
0	Agincourt	Skating Rink	Breakfast Spot	Latin American Restaurant	Lounge	Clothing Store	Vietnai Resta
1	Birch Cliff, Cliffside West	College Stadium	Skating Rink	General Entertainment	Café	Construction & Landscaping	
2	Cedarbrae	Thai Restaurant	Athletics & Sports	Bakery	Bank	Hakka Restaurant	Carib Resta
3	Clarks Corners, Tam O'Shanter, Sullivan	Pizza Place	Pharmacy	Italian Restaurant	Chinese Restaurant	Thai Restaurant	
4	Cliffside, Cliffcrest, Scarborough Village West	American Restaurant	Motel	Skating Rink	Grocery Store	General Entertainment	Gas St
5	Dorset Park, Wexford Heights, Scarborough Town...	Indian Restaurant	Vietnamese Restaurant	Chinese Restaurant	Pet Store	Construction & Landscaping	Grocery
6	Golden Mile, Clairlea, Oakridge	Bakery	Bus Station	Ice Cream Shop	Park	Intersection	Metro St
7	Guildwood, Morningside, West Hill	Mexican Restaurant	Medical Center	Bank	Moving Target	Restaurant	Renta Loc

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Com V
8	Kennedy Park, Ionview, East Birchmount Park	Convenience Store	Discount Store	Department Store	Coffee Shop	Hobby Shop	Vietnai Resta
9	Malvern, Rouge	Fast Food Restaurant	Vietnamese Restaurant	Construction & Landscaping	Gym	Grocery Store	Ge Entertain
10	Milliken, Agincourt North, Steeles East, L'Amo...	Gym	Playground	Park	College Stadium	Grocery Store	Ge Entertain
11	Rouge Hill, Port Union, Highland Creek	Home Service	Construction & Landscaping	Bar	Gym	Grocery Store	Ge Entertain
12	Scarborough Village	Smoke Shop	Playground	Vietnamese Restaurant	College Stadium	Grocery Store	Ge Entertain
13	Steeles West, L'Amoreaux West	Grocery Store	Chinese Restaurant	Fast Food Restaurant	Gym	Pharmacy	Dis
14	Wexford, Maryvale	Middle Eastern Restaurant	Smoke Shop	Bakery	Breakfast Spot	Vietnamese Restaurant	Grocery
15	Woburn	Coffee Shop	Soccer Field	Korean Restaurant	Convenience Store	Vietnamese Restaurant	Constru Landsc

Cluster Neighborhoods

In [139]:
```
kclusters = 5

scarborough_grouped_clustering = scarborough_grouped.drop('Neighborhood', 1)

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

# check cluster labels generated for each row in the dataframe
kmeans.labels_
```

Out[139]:
```
array([0, 3, 0, 0, 3, 0, 0, 0, 4, 1, 0, 0, 2, 0, 0, 4])
```

In [134]:
```
scarborough_data
```

Out[134]:

	Postal Code	Borough	Neighborhood	latitude	longitude
0	M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353
1	M1C	Scarborough	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
3	M1G	Scarborough	Woburn	43.770992	-79.216917
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476
5	M1J	Scarborough	Scarborough Village	43.744734	-79.239476
6	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park	43.727929	-79.262029
7	M1L	Scarborough	Golden Mile, Clairlea, Oakridge	43.711112	-79.284577
8	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West	43.716316	-79.239476
9	M1N	Scarborough	Birch Cliff, Cliffside West	43.692657	-79.264848
10	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...	43.757410	-79.273304
11	M1R	Scarborough	Wexford, Maryvale	43.750072	-79.295849
12	M1S	Scarborough	Agincourt	43.794200	-79.262029
13	M1T	Scarborough	Clarks Corners, Tam O'Shanter, Sullivan	43.781638	-79.304302
14	M1V	Scarborough	Milliken, Agincourt North, Steeles East, L'Amo...	43.815252	-79.284577
15	M1W	Scarborough	Steeles West, L'Amoreaux West	43.799525	-79.318389
16	M1X	Scarborough	Upper Rouge	43.836125	-79.205636

In [149]:
```
scarborough_merged = scarborough_data[0:16]

# add clustering labels
scarborough_merged['Cluster Labels'] = kmeans.labels_

# merge toronto_grouped with toronto_data to add latitude/longitude for each neig
neighborhoods_venues_sorted = scarborough_merged.join(neighborhoods_venues_sortec

neighborhoods_venues_sorted
```

```
C:\Users\mahesh venkat reesu\anaconda3\lib\site-packages\ipykernel_launcher.py:
4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/sta
ble/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pyd
ata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-c
opy)
  after removing the cwd from sys.path.
```

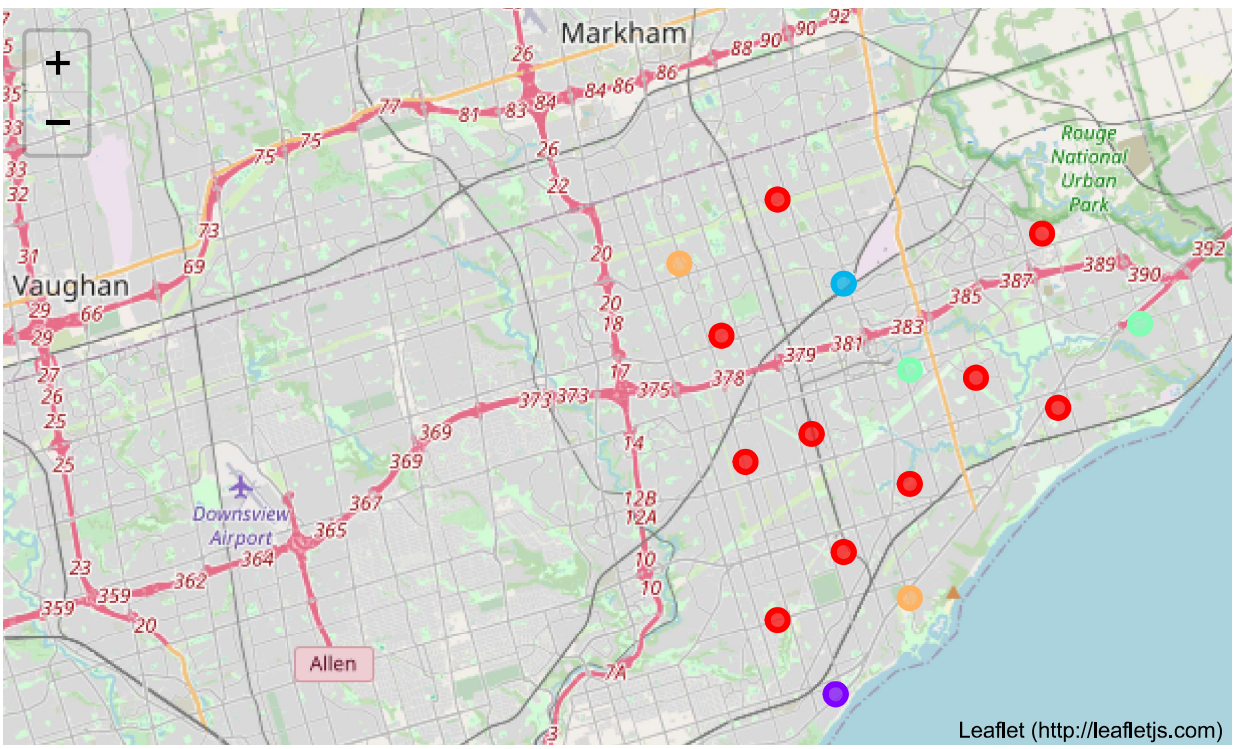
Out[149]:

	Postal Code	Borough	Neighborhood	latitude	longitude	Cluster Labels	1st Most Common Venue	2nd Mo Commc Venu
0	M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353	0	Fast Food Restaurant	Vietnames Restaura
1	M1C	Scarborough	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497	3	Home Service	Constructic Landscapir
2	M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711	0	Mexican Restaurant	Medic Cent
3	M1G	Scarborough	Woburn	43.770992	-79.216917	0	Coffee Shop	Soccer Fie
4	M1H	Scarborough	Cedarbrae	43.773136	-79.239476	3	Thai Restaurant	Athletics Spor
5	M1J	Scarborough	Scarborough Village	43.744734	-79.239476	0	Smoke Shop	Playgrour
6	M1K	Scarborough	Kennedy Park, Ionview, East Birchmount Park	43.727929	-79.262029	0	Convenience Store	Discou Sto
7	M1L	Scarborough	Golden Mile, Clairlea, Oakridge	43.711112	-79.284577	0	Bakery	Bus Static
8	M1M	Scarborough	Cliffside, Cliffcrest, Scarborough Village West	43.716316	-79.239476	4	American Restaurant	Mot

	Postal Code	Borough	Neighborhood	latitude	longitude	Cluster Labels	1st Most Common Venue	2nd Mo Commo Venu
9	M1N	Scarborough	Birch Cliff, Cliffside West	43.692657	-79.264848	1	College Stadium	Skating Rir
10	M1P	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...	43.757410	-79.273304	0	Indian Restaurant	Vietnames Restaura
11	M1R	Scarborough	Wexford, Maryvale	43.750072	-79.295849	0	Middle Eastern Restaurant	Smoke Sho
12	M1S	Scarborough	Agincourt	43.794200	-79.262029	2	Skating Rink	Breakfa Sp
13	M1T	Scarborough	Clarks Corners, Tam O'Shanter, Sullivan	43.781638	-79.304302	0	Pizza Place	Pharma
14	M1V	Scarborough	Milliken, Agincourt North, Steeles East, L'Amo...	43.815252	-79.284577	0	Gym	Playgrour
15	M1W	Scarborough	Steeles West, L'Amoreaux West	43.799525	-79.318389	4	Grocery Store	Chines Restaura

In [152]:
```
map_clusters = folium.Map(location = [latitude_scar, longitude_scar], zoom_start=

# 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(scarborough_merged['latitude'], scarborough_mer
    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[152]:



In [ ]:

Examine each of the five clusters¶

In [155]:
```
ods_venues_sorted['Cluster Labels'] == 0, neighborhoods_venues_sorted.columns[[1]
```

Out[155]:

	Borough	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th I Com V
0	Scarborough	0	Fast Food Restaurant	Vietnamese Restaurant	Construction & Landscaping	Gym	Grocery Store	Ge Entertain
2	Scarborough	0	Mexican Restaurant	Medical Center	Bank	Moving Target	Restaurant	Renta Loc
3	Scarborough	0	Coffee Shop	Soccer Field	Korean Restaurant	Convenience Store	Vietnamese Restaurant	Constru Landsca
5	Scarborough	0	Smoke Shop	Playground	Vietnamese Restaurant	College Stadium	Grocery Store	Ge Entertain
6	Scarborough	0	Convenience Store	Discount Store	Department Store	Coffee Shop	Hobby Shop	Vietnar Resta
7	Scarborough	0	Bakery	Bus Station	Ice Cream Shop	Park	Intersection	Metro St
10	Scarborough	0	Indian Restaurant	Vietnamese Restaurant	Chinese Restaurant	Pet Store	Construction & Landscaping	Grocery S
11	Scarborough	0	Middle Eastern Restaurant	Smoke Shop	Bakery	Breakfast Spot	Vietnamese Restaurant	Grocery S
13	Scarborough	0	Pizza Place	Pharmacy	Italian Restaurant	Chinese Restaurant	Thai Restaurant	
14	Scarborough	0	Gym	Playground	Park	College Stadium	Grocery Store	Ge Entertain

In [156]:
```
es_sorted['Cluster Labels'] == 1, neighborhoods_venues_sorted.columns[[1] + list(
```

Out[156]:

	Borough	Cluster Labels	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 M Comm Ven
9	Scarborough	1	College Stadium	Skating Rink	General Entertainment	Café	Construction & Landscaping	Gym	Groc St

In [157]:
```
es_sorted['Cluster Labels'] == 2, neighborhoods_venues_sorted.columns[[1] + list(
```

Out[157]:

	Borough	Cluster Labels	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 M Comr Ve
12	Scarborough	2	Skating Rink	Breakfast Spot	Latin American Restaurant	Lounge	Clothing Store	Vietnamese Restaurant	Convenie S

In [158]:
```
es_sorted['Cluster Labels'] == 3, neighborhoods_venues_sorted.columns[[1] + list(
```

Out[158]:

	Borough	Cluster Labels	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 Con V
1	Scarborough	3	Home Service	Construction & Landscaping	Bar	Gym	Grocery Store	General Entertainment	S
4	Scarborough	3	Thai Restaurant	Athletics & Sports	Bakery	Bank	Hakka Restaurant	Caribbean Restaurant	S

In [159]:
```
es_sorted['Cluster Labels'] == 4, neighborhoods_venues_sorted.columns[[1] + list(
```

Out[159]:

	Borough	Cluster Labels	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 Com V
8	Scarborough	4	American Restaurant	Motel	Skating Rink	Grocery Store	General Entertainment	Gas Station	Furr / H
15	Scarborough	4	Grocery Store	Chinese Restaurant	Fast Food Restaurant	Gym	Pharmacy	Discount Store	C

In [160]:
```
es_sorted['Cluster Labels'] == 5, neighborhoods_venues_sorted.columns[[1] + list(
```

Out[160]:

Borough	Cluster Labels	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 Mc Comm Ven

In [ ]: