PART 1

Imporyting required libraries

In [1]:

```
import pandas as pd
import numpy as np
import requests
```

Installing required libraries for retriving and processing data from web

In [2]:

```
!pip install beautifulsoup4
from bs4 import BeautifulSoup
!pip install lxml
```

```
Requirement already satisfied: beautifulsoup4 in /home/jupyterlab/conda/envs/python/
lib/python3.6/site-packages (4.9.1)
Requirement already satisfied: soupsieve>1.2 in /home/jupyterlab/conda/envs/python/l
ib/python3.6/site-packages (from beautifulsoup4) (2.0.1)
Requirement already satisfied: lxml in /home/jupyterlab/conda/envs/python/lib/python
3.6/site-packages (4.5.2)
```

In [3]:

```
import numpy as np
import pandas
import seaborn # 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

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

# 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 # uncomment this line if you haven't completed the Foursquare API lab
import folium # map rendering library

print('Libraries imported.')
```

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


==> WARNING: A newer version of conda exists. <==
  current version: 4.8.3
  latest version: 4.8.4

Please update conda by running

    $ conda update -n base -c defaults conda



## Package Plan ##

  environment location: /home/jupyterlab/conda/envs/python

  added / updated specs:
    - geopy


The following NEW packages will be INSTALLED:

  geographiclib      conda-forge/noarch::geographiclib-1.50-py_0
  geopy              conda-forge/noarch::geopy-2.0.0-pyh9f0ad1d_0

The following packages will be UPDATED:

  openssl                                    1.1.1g-h516909a_0 --> 1.1.1g-h516909a_1


Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Libraries imported.
```

In [4]:

```
wiki = 'https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M'
wiki_page = requests.get(wiki)

wiki_raw = pd.read_html(wiki_page.content, header = 0)[0]
df = wiki_raw[wiki_raw.District != 'Not assigned']
df.reset_index(inplace = True)
df.head()
```

Out[4]:

	index	Postal Code	District	Neighbourhood
0	2	M3A	North York	Parkwoods
1	3	M4A	North York	Victoria Village
2	4	M5A	Downtown Toronto	Regent Park, Harbourfront
3	5	M6A	North York	Lawrence Manor, Lawrence Heights
4	6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

In [5]:

```
df.groupby(['Postal Code']).first()
```

Out[5]:

	index	District	Neighbourhood
Postal Code			
M1B	9	Scarborough	Malvern, Rouge
M1C	18	Scarborough	Rouge Hill, Port Union, Highland Creek
M1E	27	Scarborough	Guildwood, Morningside, West Hill
M1G	36	Scarborough	Woburn
M1H	45	Scarborough	Cedarbrae
M1J	54	Scarborough	Scarborough Village
M1K	63	Scarborough	Kennedy Park, Ionview, East Birchmount Park
M1L	72	Scarborough	Golden Mile, Clairlea, Oakridge
M1M	81	Scarborough	Cliffside, Cliffcrest, Scarborough Village West
M1N	90	Scarborough	Birch Cliff, Cliffside West
M1P	99	Scarborough	Dorset Park, Wexford Heights, Scarborough Town...
M1R	108	Scarborough	Wexford, Maryvale
M1S	117	Scarborough	Agincourt
M1T	126	Scarborough	Clarks Corners, Tam O'Shanter, Sullivan
M1V	135	Scarborough	Milliken, Agincourt North, Steeles East, L'Amo...
M1W	144	Scarborough	Steeles West, L'Amoreaux West
M1X	153	Scarborough	Upper Rouge
M2H	46	North York	Hillcrest Village
M2J	55	North York	Fairview, Henry Farm, Oriole
M2K	64	North York	Bayview Village
M2L	73	North York	York Mills, Silver Hills
M2M	82	North York	Willowdale, Newtonbrook
M2N	91	North York	Willowdale, Willowdale East
M2P	100	North York	York Mills West
M2R	109	North York	Willowdale, Willowdale West
M3A	2	North York	Parkwoods
M3B	11	North York	Don Mills
M3C	20	North York	Don Mills
M3H	47	North York	Bathurst Manor, Wilson Heights, Downsview North
M3J	56	North York	Northwood Park, York University
M3K	65	North York	Downsview
M3L	74	North York	Downsview
M3M	83	North York	Downsview
M3N	92	North York	Downsview
M4A	3	North York	Victoria Village
M4B	12	East York	Parkview Hill, Woodbine Gardens
M4C	21	East York	Woodbine Heights
M4E	30	East Toronto	The Beaches

	index	District	Neighbourhood
Postal Code			
M4G	39	East York	Leaside
M4H	48	East York	Thorncliffe Park
M4J	57	East York	East Toronto, Broadview North (Old East York)
M4K	66	East Toronto	The Danforth West, Riverdale
M4L	75	East Toronto	India Bazaar, The Beaches West
M4M	84	East Toronto	Studio District
M4N	93	Central Toronto	Lawrence Park
M4P	102	Central Toronto	Davisville North
M4R	111	Central Toronto	North Toronto West, Lawrence Park
M4S	120	Central Toronto	Davisville
M4T	129	Central Toronto	Moore Park, Summerhill East
M4V	138	Central Toronto	Summerhill West, Rathnelly, South Hill, Forest...
M4W	147	Downtown Toronto	Rosedale
M4X	156	Downtown Toronto	St. James Town, Cabbagetown
M4Y	165	Downtown Toronto	Church and Wellesley
M5A	4	Downtown Toronto	Regent Park, Harbourfront
M5B	13	Downtown Toronto	Garden District, Ryerson
M5C	22	Downtown Toronto	St. James Town
M5E	31	Downtown Toronto	Berczy Park
M5G	40	Downtown Toronto	Central Bay Street
M5H	49	Downtown Toronto	Richmond, Adelaide, King
M5J	58	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands
M5K	67	Downtown Toronto	Toronto Dominion Centre, Design Exchange
M5L	76	Downtown Toronto	Commerce Court, Victoria Hotel
M5M	85	North York	Bedford Park, Lawrence Manor East
M5N	94	Central Toronto	Roselawn
M5P	103	Central Toronto	Forest Hill North & West, Forest Hill Road Park
M5R	112	Central Toronto	The Annex, North Midtown, Yorkville
M5S	121	Downtown Toronto	University of Toronto, Harbord
M5T	130	Downtown Toronto	Kensington Market, Chinatown, Grange Park
M5V	139	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...
M5W	148	Downtown Toronto	Stn A PO Boxes
M5X	157	Downtown Toronto	First Canadian Place, Underground city
M6A	5	North York	Lawrence Manor, Lawrence Heights
M6B	14	North York	Glencairn
M6C	23	York	Humewood-Cedarvale
M6E	32	York	Caledonia-Fairbanks
M6G	41	Downtown Toronto	Christie
M6H	50	West Toronto	Dufferin, Dovercourt Village

	index	District	Neighbourhood
Postal Code			
M6J	59	West Toronto	Little Portugal, Trinity
M6K	68	West Toronto	Brockton, Parkdale Village, Exhibition Place
M6L	77	North York	North Park, Maple Leaf Park, Upwood Park
M6M	86	York	Del Ray, Mount Dennis, Keelsdale and Silverthorn
M6N	95	York	Runnymede, The Junction North
M6P	104	West Toronto	High Park, The Junction South
M6R	113	West Toronto	Parkdale, Roncesvalles
M6S	122	West Toronto	Runnymede, Swansea
M7A	6	Downtown Toronto	Queen's Park, Ontario Provincial Government
M7R	114	Mississauga	Canada Post Gateway Processing Centre
M7Y	168	East Toronto	Business reply mail Processing Centre, South C...
M8V	142	Etobicoke	New Toronto, Mimico South, Humber Bay Shores
M8W	151	Etobicoke	Alderwood, Long Branch
M8X	160	Etobicoke	The Kingsway, Montgomery Road, Old Mill North
M8Y	169	Etobicoke	Old Mill South, King's Mill Park, Sunnylea, Hu...
M8Z	178	Etobicoke	Mimico NW, The Queensway West, South of Bloor,...
M9A	8	Etobicoke	Islington Avenue, Humber Valley Village
M9B	17	Etobicoke	West Deane Park, Princess Gardens, Martin Grov...
M9C	26	Etobicoke	Eringate, Bloordale Gardens, Old Burnhamthorpe...
M9L	80	North York	Humber Summit
M9M	89	North York	Humberlea, Emery
M9N	98	York	Weston
M9P	107	Etobicoke	Westmount
M9R	116	Etobicoke	Kingsview Village, St. Phillips, Martin Grove ...
M9V	143	Etobicoke	South Steeles, Silverstone, Humbergate, Jamest...
M9W	152	Etobicoke	Northwest, West Humber - Clairville

```
In [6]: len(df['Postal Code'].unique())
```

```
Out[6]: 103
```

removing coloumns which contains "Not assigned"

In [7]:
```
df[df.District!= 'Not assigned'].head()
```

Out[7]:

	index	Postal Code	District	Neighbourhood
0	2	M3A	North York	Parkwoods
1	3	M4A	North York	Victoria Village
2	4	M5A	Downtown Toronto	Regent Park, Harbourfront
3	5	M6A	North York	Lawrence Manor, Lawrence Heights
4	6	M7A	Downtown Toronto	Queen's Park, Ontario Provincial Government

In [ ]:

In [8]:
```
df.shape
```

Out[8]:
```
(103, 4)
```

In [ ]:

PART 2

In [9]:
```
pip install geocoder
```

```
Collecting geocoder
  Using cached https://files.pythonhosted.org/packages/4f/6b/13166c909ad2f2d76b929a4
227c952630ebaf0d729f6317eb09cbceccbab/geocoder-1.38.1-py2.py3-none-any.whl
Collecting ratelim (from geocoder)
  Using cached https://files.pythonhosted.org/packages/f2/98/7e6d147fd16a10a5f821db6
e25f192265d6ecca3d82957a4fdd592cad49c/ratelim-0.1.6-py2.py3-none-any.whl
Requirement already satisfied: six in /home/jupyterlab/conda/envs/python/lib/python
3.6/site-packages (from geocoder) (1.15.0)
Collecting future (from geocoder)
  Using cached https://files.pythonhosted.org/packages/45/0b/38b06fd9b92dc2b68d58b75
f900e97884c45bedd2ff83203d933cf5851c9/future-0.18.2.tar.gz
Collecting click (from geocoder)
  Using cached https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4f8d338c2f
d15c8d33fdd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/py
thon3.6/site-packages (from geocoder) (2.24.0)
Requirement already satisfied: decorator in /home/jupyterlab/conda/envs/python/lib/p
ython3.6/site-packages (from ratelim->geocoder) (4.4.2)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/li
b/python3.6/site-packages (from requests->geocoder) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/pyt
hon/lib/python3.6/site-packages (from requests->geocoder) (2020.6.20)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /home/jupy
terlab/conda/envs/python/lib/python3.6/site-packages (from requests->geocoder) (1.2
5.10)
Requirement already satisfied: chardet<4,>=3.0.2 in /home/jupyterlab/conda/envs/pyth
on/lib/python3.6/site-packages (from requests->geocoder) (3.0.4)
Building wheels for collected packages: future
  Building wheel for future (setup.py) ... done
  Stored in directory: /home/jupyterlab/.cache/pip/wheels/8b/99/a0/81daf51dcd359a937
7b110a8a886b3895921802d2fc1b2397e
Successfully built future
Installing collected packages: ratelim, future, click, geocoder
Successfully installed click-7.1.2 future-0.18.2 geocoder-1.38.1 ratelim-0.1.6
Note: you may need to restart the kernel to use updated packages.
```

In [10]:
```
import geocoder
```

In [11]:
```
url = 'http://cocl.us/Geospatial_data'
```

In [12]:
```
df_geo = pd.read_csv(url)
df_geo.head()
```

Out[12]:

	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 [13]:
```
import pandas as pd
df = pd.concat([df.reset_index(),df_geo], axis=1)
df.head()
df = df.loc[:,~df.columns.duplicated()]
df
df.drop('Postal Code',axis = 1, inplace = True)
df.head()
```

Out[13]:

	level_0	index	District	Neighbourhood	Latitude	Longitude
0	0	2	North York	Parkwoods	43.806686	-79.194353
1	1	3	North York	Victoria Village	43.784535	-79.160497
2	2	4	Downtown Toronto	Regent Park, Harbourfront	43.763573	-79.188711
3	3	5	North York	Lawrence Manor, Lawrence Heights	43.770992	-79.216917
4	4	6	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.773136	-79.239476

In [14]:
```
df.head()
df.shape
```

Out[14]:
```
(103, 6)
```

PART 3

In [15]:
```
df1 = df.copy()
df1.head()
```

Out[15]:

	level_0	index	District	Neighbourhood	Latitude	Longitude
0	0	2	North York	Parkwoods	43.806686	-79.194353
1	1	3	North York	Victoria Village	43.784535	-79.160497
2	2	4	Downtown Toronto	Regent Park, Harbourfront	43.763573	-79.188711
3	3	5	North York	Lawrence Manor, Lawrence Heights	43.770992	-79.216917
4	4	6	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.773136	-79.239476

In [16]:
```
print(len(df.District.unique()))
len(df.Neighbourhood.unique())
print(df.District.unique())
```

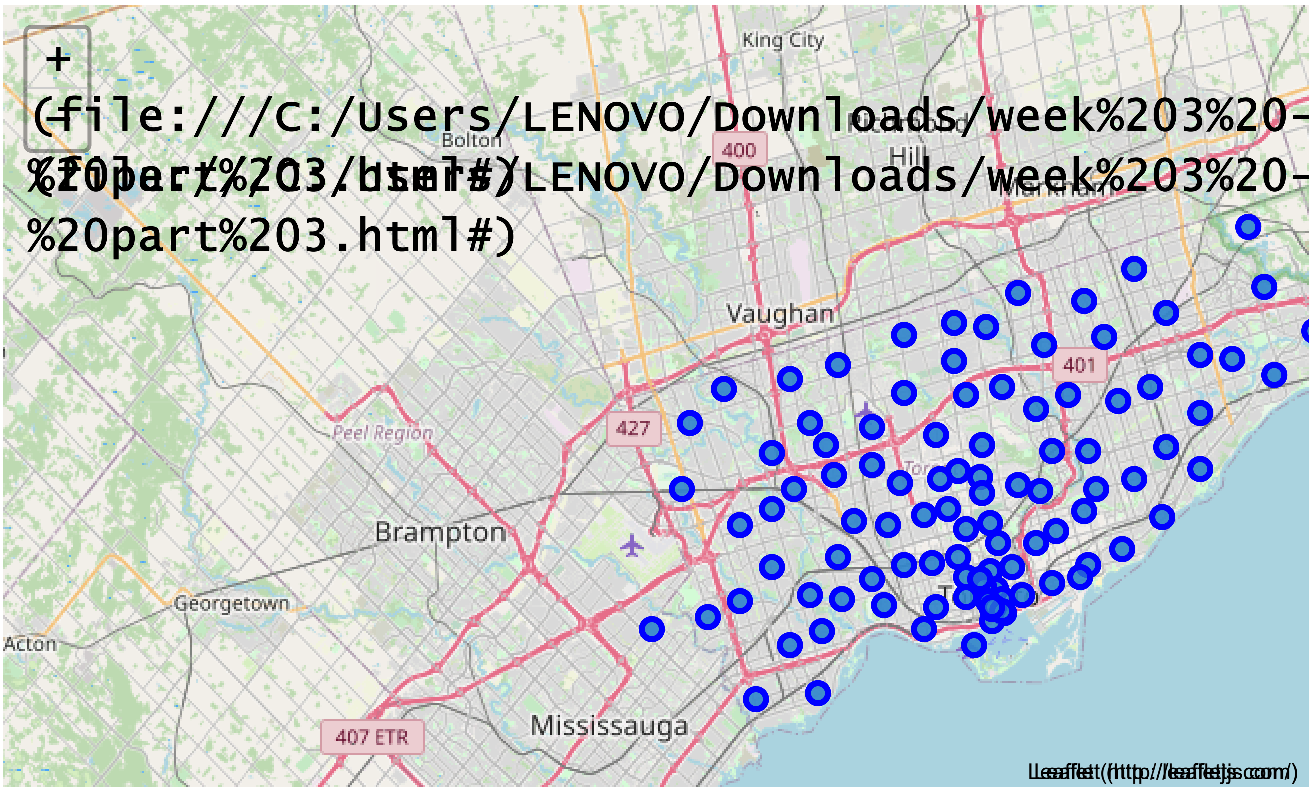
```
10
['North York' 'Downtown Toronto' 'Etobicoke' 'Scarborough' 'East York'
 'York' 'East Toronto' 'West Toronto' 'Central Toronto' 'Mississauga']
```

In [17]:
```
address = 'Toronto, ON'

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

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

In [18]:
```
# create map of New York using latitude and longitude values
map_newyork = folium.Map(location=[latitude, longitude], zoom_start=10)

# add markers to map
for lat, lng, borough, df in zip(df['Latitude'], df['Longitude'], df['District'], df[
'Neighbourhood']):
    label = '{}, {}'.format(df, 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_newyork)

map_newyork
```

Out[18]:


+
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
King City
Bolton
400
Richmond Hill
Markham
Vaughan
401
427
Peel Region
Brampton
Georgetown
Acton
Mississauga
407 ETR
Leaflet (http://leafletjs.com/)

taking toronto data

T_data = Toronto data

In [19]:

```
T_data = df1.loc[df1['District'] == 'Downtown Toronto']
T_data
```

Out[19]:

	level_0	index	District	Neighbourhood	Latitude	Longitude
2	2	4	Downtown Toronto	Regent Park, Harbourfront	43.763573	-79.188711
4	4	6	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.773136	-79.239476
9	9	13	Downtown Toronto	Garden District, Ryerson	43.692657	-79.264848
15	15	22	Downtown Toronto	St. James Town	43.799525	-79.318389
20	20	31	Downtown Toronto	Berczy Park	43.757490	-79.374714
24	24	40	Downtown Toronto	Central Bay Street	43.782736	-79.442259
25	25	41	Downtown Toronto	Christie	43.753259	-79.329656
30	30	49	Downtown Toronto	Richmond, Adelaide, King	43.737473	-79.464763
36	36	58	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.695344	-79.318389
42	42	67	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.668999	-79.315572
48	48	76	Downtown Toronto	Commerce Court, Victoria Hotel	43.689574	-79.383160
80	80	121	Downtown Toronto	University of Toronto, Harbord	43.691116	-79.476013
84	84	130	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.651571	-79.484450
87	87	139	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.662744	-79.321558
91	91	147	Downtown Toronto	Rosedale	43.636258	-79.498509
92	92	148	Downtown Toronto	Stn A PO Boxes	43.628841	-79.520999
96	96	156	Downtown Toronto	St. James Town, Cabbagetown	43.756303	-79.565963
97	97	157	Downtown Toronto	First Canadian Place, Underground city	43.724766	-79.532242
99	99	165	Downtown Toronto	Church and Wellesley	43.696319	-79.532242

In [20]:

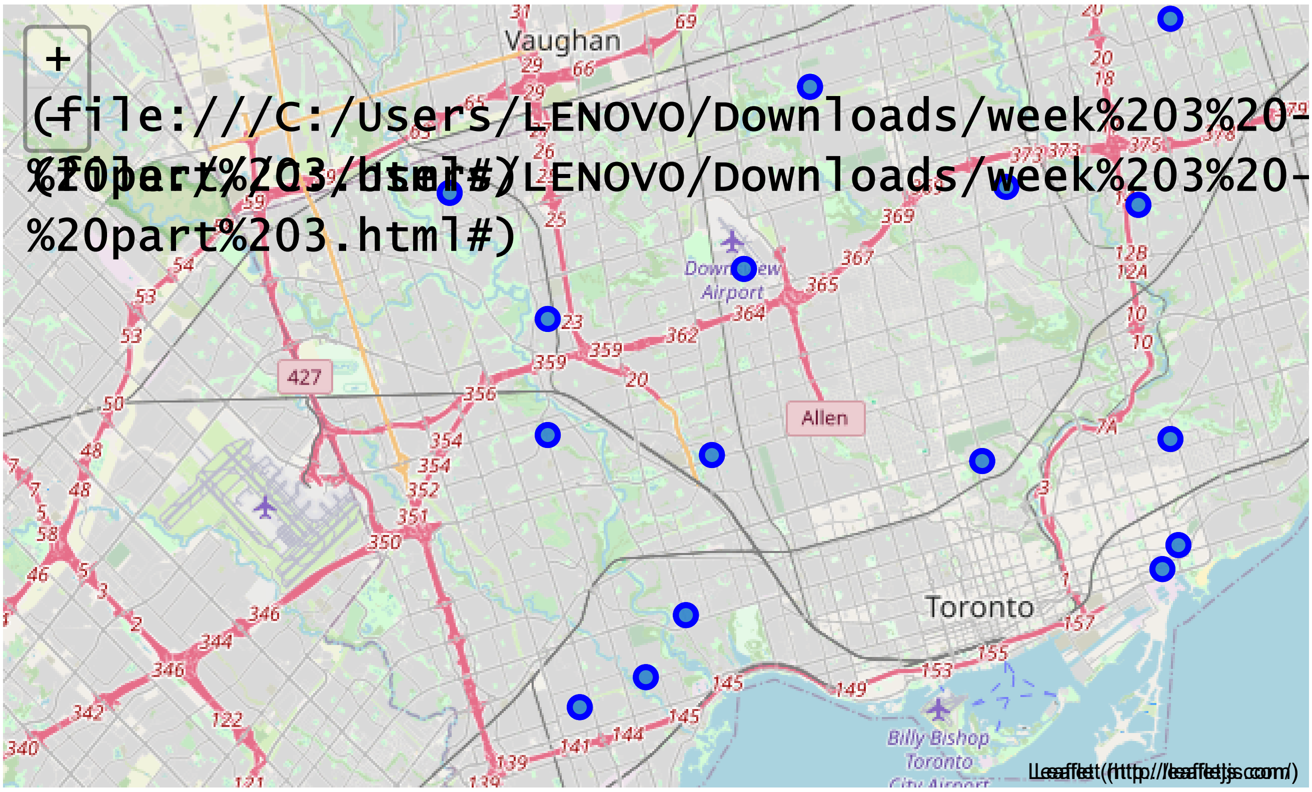
```
address = 'Downtown Toronto, ON'

geolocator = Nominatim(user_agent="ny_explorer")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print('The geograpical coordinate of Manhattan are {}, {}.'.format(latitude, longitud
e))
```

```
The geograpical coordinate of Manhattan are 43.6563221, -79.3809161.
```

In [21]:

```python
# create map of Manhattan using latitude and longitude values
map_Downtowntoronto= folium.Map(location=[latitude, longitude], zoom_start=11)

# add markers to map
for lat, lng, label in zip(T_data['Latitude'], T_data['Longitude'], T_data['Neighbourhood']):
    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_Downtowntoronto)

map_Downtowntoronto
```

Out[21]:


+
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
Vaughan
Downsview Airport
427
Allen
Toronto
Billy Bishop Toronto City Airport
Leaflet (http://leafletjs.com/)

In [22]:

```python
CLIENT_ID = 'FIOXIFMDA5IYSCFWXWT21UWH1Y2B0TPUHLQCMNFBEBQAW0RY' # your Foursquare ID
CLIENT_SECRET = 'KWUO4IYF1TE1A41EKUNRM2RMY5V5OJASGXJTAO2AVXZH0P5Z' # your Foursquare Secret
VERSION = '20200827' # Foursquare API version

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

```
Your credentails:
CLIENT_ID: FIOXIFMDA5IYSCFWXWT21UWH1Y2B0TPUHLQCMNFBEBQAW0RY
CLIENT_SECRET:KWUO4IYF1TE1A41EKUNRM2RMY5V5OJASGXJTAO2AVXZH0P5Z
```

In [57]:

```python
T_data.reset_index(inplace = True)
```

In [58]:

```
T_data.loc[0,'Neighbourhood']
```

Out[58]:

```
'Regent Park, Harbourfront'
```

In [59]:

```
neighborhood_latitude = T_data.loc[0,'Latitude'] # neighborhood latitude value
neighborhood_longitude =T_data.loc[0,'Longitude'] # neighborhood longitude value

neighborhood_name = T_data.loc[0,'Neighbourhood']# neighborhood name

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

```
Latitude and longitude values of Regent Park, Harbourfront are 43.7635726, -79.18871
15.
```

In [62]:

```
LIMIT = 100 # limit of number of venues returned by Foursquare API

radius = 500 # define radius

# create URL
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v=
{}&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
url # display URL
```

Out[62]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=FIOXIFMDA5IYSCFWXWT21UWH1Y2
B0TPUHLQCMNFBEBQAW0RY&client_secret=KWUO4IYF1TE1A41EKUNRM2RMY5V5OJASGXJTAO2AVXZH0P5Z
&v=20200827&ll=43.7635726,-79.1887115&radius=500&limit=100'
```

In [63]:

```
results = requests.get(url).json()
results
```

```
Out[63]: {'meta': {'code': 200, 'requestId': '5f48df933c862528834220c0'},
          'response': {'headerLocation': 'Scarborough Village',
           'headerFullLocation': 'Scarborough Village, Toronto',
           'headerLocationGranularity': 'neighborhood',
           'totalResults': 8,
           'suggestedBounds': {'ne': {'lat': 43.768072604500006,
             'lng': -79.18249216787879},
            'sw': {'lat': 43.7590725955, 'lng': -79.1949308321212}},
           'groups': [{'type': 'Recommended Places',
             'name': 'recommended',
             'items': [{'reasons': {'count': 0,
                'items': [{'summary': 'This spot is popular',
                  'type': 'general',
                  'reasonName': 'globalInteractionReason'}]},
               'venue': {'id': '4beee041e24d20a1cd857314',
                'name': 'RBC Royal Bank',
                'location': {'address': '4374 KINGSTON RD',
                 'crossStreet': 'Kingston & Lawrence',
                 'lat': 43.76678992471017,
                 'lng': -79.19115118872593,
                 'labeledLatLngs': [{'label': 'display',
                   'lat': 43.76678992471017,
                   'lng': -79.19115118872593}],
                 'distance': 408,
                 'postalCode': 'M1E 2M8',
                 'cc': 'CA',
                 'city': 'Scarborough',
                 'state': 'ON',
                 'country': 'Canada',
                 'formattedAddress': ['4374 KINGSTON RD (Kingston & Lawrence)',
                  'Scarborough ON M1E 2M8',
                  'Canada']},
                'categories': [{'id': '4bf58dd8d48988d10a951735',
                  'name': 'Bank',
                  'pluralName': 'Banks',
                  'shortName': 'Bank',
                  'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/financial
_',
                   'suffix': '.png'},
                  'primary': True}],
                'photos': {'count': 0, 'groups': []}},
               'referralId': 'e-0-4beee041e24d20a1cd857314-0'},
              {'reasons': {'count': 0,
                'items': [{'summary': 'This spot is popular',
                  'type': 'general',
                  'reasonName': 'globalInteractionReason'}]},
               'venue': {'id': '4c62f34bde1b2d7fec89e370',
                'name': 'G & G Electronics',
                'location': {'address': '4371 kingston road',
                 'lat': 43.765309,
                 'lng': -79.191537,
                 'labeledLatLngs': [{'label': 'display',
                   'lat': 43.765309,
                   'lng': -79.191537}],
                 'distance': 298,
                 'postalCode': 'M1E 2M9',
                 'cc': 'CA',
                 'city': 'Toronto',
                 'state': 'ON',
                 'country': 'Canada',
                 'formattedAddress': ['4371 kingston road',
                  'Toronto ON M1E 2M9',
                  '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-0-4c62f34bde1b2d7fec89e370-1'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '57fd24f6cd1083addfd77bf9',
       'name': 'Sail Sushi',
       'location': {'address': '9-4352 Kingston Rd',
        'lat': 43.765951,
        'lng': -79.191275,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.765951,
          'lng': -79.191275}],
        'distance': 335,
        'postalCode': 'M1E 2M8',
        'cc': 'CA',
        'city': 'Scarborough',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['9-4352 Kingston Rd',
         'Scarborough ON M1E 2M8',
         '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-0-57fd24f6cd1083addfd77bf9-2'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '5411f741498e9ebd5e35d8bd',
       'name': 'Big Bite Burrito',
       'location': {'address': '4383 Kingston rd.',
        'lat': 43.766299084470795,
        'lng': -79.19071980583941,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.766299084470795,
          'lng': -79.19071980583941}],
        'distance': 343,
        'cc': 'CA',
        'city': 'Scarborough',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['4383 Kingston rd.', 'Scarborough ON', 'Canada']},
       'categories': [{'id': '4bf58dd8d48988d1c1941735',
         'name': 'Mexican Restaurant',
         'pluralName': 'Mexican Restaurants',
         'shortName': 'Mexican',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/food/mexican_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-5411f741498e9ebd5e35d8bd-3'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4c1c7f9bb306c9288f0464b7',
       'name': 'Enterprise Rent-A-Car',
       'location': {'address': '4304 Kingston Rd',
        'lat': 43.7640757,
        'lng': -79.1934057,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.7640757,
          'lng': -79.1934057}],
        'distance': 381,
        'postalCode': 'M1E 2M8',
        'cc': 'CA',
        'city': 'Scarborough',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['4304 Kingston Rd',
         'Scarborough ON M1E 2M8',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d1ef941735',
         'name': 'Rental Car Location',
         'pluralName': 'Rental Car Locations',
         'shortName': 'Rental Car',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/shops/airport_re
ntalcar_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4c1c7f9bb306c9288f0464b7-4'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4b9008acf964a520737133e3',
       'name': 'Woburn Medical Centre',
       'location': {'address': '4125 Lawrence Ave E',
        'crossStreet': 'Kingston Rd',
        'lat': 43.766631153138455,
        'lng': -79.19228553771973,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.766631153138455,
          'lng': -79.19228553771973}],
        'distance': 445,
        'cc': 'CA',
        'city': 'West Hill',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['4125 Lawrence Ave E (Kingston Rd)',
         'West Hill ON',
         'Canada']},
       'categories': [{'id': '4bf58dd8d48988d104941735',
         'name': 'Medical Center',
         'pluralName': 'Medical Centers',
         'shortName': 'Medical',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/medical
_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-4b9008acf964a520737133e3-5'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '513b84e6e4b0ef935943c0ad',
       'name': 'Lawrence Ave E & Kingston Rd',
       'location': {'address': 'Lawrence Ave E & Kingston Rd',
        'crossStreet': 'Lawrence Ave E & Kingston Rd',
        'lat': 43.767704,
        'lng': -79.18949,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.767704,
          'lng': -79.18949}],
        'distance': 464,
        'postalCode': 'M1E',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['Lawrence Ave E & Kingston Rd (Lawrence Ave E & Kingsto
n Rd)',
         'Toronto ON M1E',
         'Canada']},
       'categories': [{'id': '52f2ab2ebcbc57f1066b8b4c',
         'name': 'Intersection',
         'pluralName': 'Intersections',
         'shortName': 'Intersection',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/travel/default
_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-513b84e6e4b0ef935943c0ad-6'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4d95bd2ddaec224bf7bf043e',
       'name': 'Eggsmart',
       'location': {'address': '4410 Kingston Rd.',
        'crossStreet': 'at Lawerence Ave. E',
        'lat': 43.7678,
        'lng': -79.190466,
        'labeledLatLngs': [{'label': 'display',
          'lat': 43.7678,
          'lng': -79.190466}],
        'distance': 491,
        'postalCode': 'M1E 2N4',
        'cc': 'CA',
        'city': 'Toronto',
        'state': 'ON',
        'country': 'Canada',
        'formattedAddress': ['4410 Kingston Rd. (at Lawerence Ave. E)',
         'Toronto ON M1E 2N4',
         '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-0-4d95bd2ddaec224bf7bf043e-7'}]}]}}
```

In [64]:

```python
# 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 [65]:

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

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.lo
cation.lng']
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()
```

```
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel_launcher.p
y:3: FutureWarning: pandas.io.json.json_normalize is deprecated, use pandas.json_nor
malize instead
  This is separate from the ipykernel package so we can avoid doing imports until
```

Out[65]:

	name	categories	lat	lng
0	RBC Royal Bank	Bank	43.766790	-79.191151
1	G & G Electronics	Electronics Store	43.765309	-79.191537
2	Sail Sushi	Restaurant	43.765951	-79.191275
3	Big Bite Burrito	Mexican Restaurant	43.766299	-79.190720
4	Enterprise Rent-A-Car	Rental Car Location	43.764076	-79.193406

In [66]:

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

```
8 venues were returned by Foursquare.
```

In [67]:

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

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

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

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

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

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

    return(nearby_venues)
```

Now write the code to run the above function on each neighborhood and create a new dataframe called manhattan_venues.

In [72]:
```
# type your answer here

T_venues = getNearbyVenues(names=T_data['Neighbourhood'],
                                   latitudes=T_data['Latitude'],
                                   longitudes=T_data['Longitude']
                                  )
```

```
Regent Park, Harbourfront
Queen's Park, Ontario Provincial Government
Garden District, Ryerson
St. James Town
Berczy Park
Central Bay Street
Christie
Richmond, Adelaide, King
Harbourfront East, Union Station, Toronto Islands
Toronto Dominion Centre, Design Exchange
Commerce Court, Victoria Hotel
University of Toronto, Harbord
Kensington Market, Chinatown, Grange Park
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South N
iagara, Island airport
Rosedale
Stn A PO Boxes
St. James Town, Cabbagetown
First Canadian Place, Underground city
Church and Wellesley
```

In [73]:
```
print(T_venues.shape)
T_venues.head()
```

```
(159, 7)
```

Out[73]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Regent Park, Harbourfront	43.763573	-79.188711	RBC Royal Bank	43.766790	-79.191151	Bank
1	Regent Park, Harbourfront	43.763573	-79.188711	G & G Electronics	43.765309	-79.191537	Electronics Store
2	Regent Park, Harbourfront	43.763573	-79.188711	Sail Sushi	43.765951	-79.191275	Restaurant
3	Regent Park, Harbourfront	43.763573	-79.188711	Big Bite Burrito	43.766299	-79.190720	Mexican Restaurant
4	Regent Park, Harbourfront	43.763573	-79.188711	Enterprise Rent-A-Car	43.764076	-79.193406	Rental Car Location

In [76]:

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

Out[76]:

	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Neighborhood						
CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport	18	18	18	18	18	18
Central Bay Street	5	5	5	5	5	5
Christie	2	2	2	2	2	2
Church and Wellesley	8	8	8	8	8	8
Commerce Court, Victoria Hotel	3	3	3	3	3	3
First Canadian Place, Underground city	1	1	1	1	1	1
Garden District, Ryerson	4	4	4	4	4	4
Harbourfront East, Union Station, Toronto Islands	8	8	8	8	8	8
Kensington Market, Chinatown, Grange Park	35	35	35	35	35	35
Queen's Park, Ontario Provincial Government	8	8	8	8	8	8
Regent Park, Harbourfront	8	8	8	8	8	8
Richmond, Adelaide, King	3	3	3	3	3	3
Rosedale	2	2	2	2	2	2
St. James Town	15	15	15	15	15	15
St. James Town, Cabbagetown	2	2	2	2	2	2
Stn A PO Boxes	13	13	13	13	13	13
Toronto Dominion Centre, Design Exchange	20	20	20	20	20	20
University of Toronto, Harbord	4	4	4	4	4	4

Let's find out how many unique categories can be curated from all the returned venues

In [78]:

```
print('There are {} uniques categories.'.format(len(T_venues['Venue Category'].unique
())))
```

```
There are 87 uniques categories.
```

3. Analyze Each Neighborhood

In [80]:

```
# one hot encoding
T_onehot = pd.get_dummies(T_venues[['Venue Category']], prefix="", prefix_sep="")

# add neighborhood column back to dataframe
T_onehot['Neighborhood'] = T_venues['Neighborhood']

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

T_onehot.head()
```

Out[80]:

	Neighborhood	Airport	Athletics & Sports	Auto Workshop	Bakery	Bank	Bar	Baseball Field	Beer Store	Board Shop	Bookstore
0	Regent Park, Harbourfront	0	0	0	0	1	0	0	0	0	0
1	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0	0
2	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0	0
3	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0	0
4	Regent Park, Harbourfront	0	0	0	0	0	0	0	0	0	0

In [82]:

```
T_onehot.shape
```

Out[82]:

```
(159, 88)
```

In [83]:

```
T_grouped = T_onehot.groupby('Neighborhood').mean().reset_index()
T_grouped
```

Out[83]:

	Neighborhood	Airport	Athletics & Sports	Auto Workshop	Bakery	Bank	Bar	Baseball Field	Beer Store	Board Shop
0	CN Tower, King and Spadina, Railway Lands, Har...	0.000000	0.000	0.055556	0.000000	0.000000	0.000000	0.0	0.000	0.00
1	Central Bay Street	0.000000	0.000	0.000000	0.000000	0.200000	0.000000	0.0	0.000	0.00
2	Christie	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
3	Church and Wellesley	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
4	Commerce Court, Victoria Hotel	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
5	First Canadian Place, Underground city	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	1.0	0.000	0.00
6	Garden District, Ryerson	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
7	Harbourfront East, Union Station, Toronto Islands	0.000000	0.125	0.000000	0.000000	0.000000	0.000000	0.0	0.125	0.00
8	Kensington Market, Chinatown, Grange Park	0.000000	0.000	0.000000	0.000000	0.028571	0.028571	0.0	0.000	0.00
9	Queen's Park, Ontario Provincial Government	0.000000	0.125	0.000000	0.125000	0.125000	0.000000	0.0	0.000	0.00
10	Regent Park, Harbourfront	0.000000	0.000	0.000000	0.000000	0.125000	0.000000	0.0	0.000	0.00
11	Richmond, Adelaide, King	0.333333	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
12	Rosedale	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.5	0.000	0.00
13	St. James Town	0.000000	0.000	0.000000	0.000000	0.066667	0.000000	0.0	0.000	0.00
14	St. James Town, Cabbagetown	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00
15	Stn A PO Boxes	0.000000	0.000	0.000000	0.076923	0.000000	0.000000	0.0	0.000	0.00
16	Toronto Dominion Centre, Design Exchange	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.05
17	University of Toronto, Harbord	0.000000	0.000	0.000000	0.000000	0.000000	0.000000	0.0	0.000	0.00

In [84]:

```
T_grouped.shape
```

Out[84]:

```
(18, 88)
```

Let's print each neighborhood along with the top 5 most common venues

In [87]:

```
num_top_venues = 5

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

```
----CN Tower, King and Spadina, Railway Lands, Harbourfront West, Bathurst Quay, South Niagara, Island airport----
                  venue  freq
0    Light Rail Station  0.11
1  Gym / Fitness Center  0.06
2               Butcher  0.06
3            Skate Park  0.06
4            Restaurant  0.06


----Central Bay Street----
           venue  freq
0       Pharmacy   0.2
1           Bank   0.2
2  Grocery Store   0.2
3    Coffee Shop   0.2
4    Pizza Place   0.2


----Christie----
               venue  freq
0               Park   0.5
1  Food & Drink Shop   0.5
2            Airport   0.0
3           Gym Pool   0.0
4          Pet Store   0.0


----Church and Wellesley----
                venue  freq
0         Pizza Place  0.25
1      Discount Store  0.12
2        Intersection  0.12
3         Coffee Shop  0.12
4  Chinese Restaurant  0.12


----Commerce Court, Victoria Hotel----
               venue  freq
0         Playground  0.33
1               Park  0.33
2             Lawyer  0.33
3  Health Food Store  0.00
4     Ice Cream Shop  0.00


----First Canadian Place, Underground city----
                venue  freq
0      Baseball Field   1.0
1             Airport   0.0
2  Light Rail Station   0.0
3           Pet Store   0.0
4                Park   0.0


----Garden District, Ryerson----
                   venue  freq
0           Skating Rink  0.25
1        College Stadium  0.25
2  General Entertainment  0.25
3                   Café  0.25
4                Airport  0.00
```

```
----Harbourfront East, Union Station, Toronto Islands----
                venue  freq
0        Skating Rink  0.12
1  Athletics & Sports  0.12
2         Video Store  0.12
3         Curling Ice  0.12
4                Park  0.12


----Kensington Market, Chinatown, Grange Park----
                venue  freq
0                Café  0.09
1  Italian Restaurant  0.06
2         Pizza Place  0.06
3         Coffee Shop  0.06
4                 Pub  0.06


----Queen's Park, Ontario Provincial Government----
                venue  freq
0         Gas Station  0.12
1    Hakka Restaurant  0.12
2              Bakery  0.12
3                Bank  0.12
4  Athletics & Sports  0.12


----Regent Park, Harbourfront----
                 venue  freq
0       Medical Center  0.12
1    Electronics Store  0.12
2                 Bank  0.12
3  Rental Car Location  0.12
4         Intersection  0.12


----Richmond, Adelaide, King----
               venue  freq
0            Airport  0.33
1               Park  0.33
2        Snack Place  0.33
3  Health Food Store  0.00
4       Liquor Store  0.00


----Rosedale----
                venue  freq
0                Park   0.5
1      Baseball Field   0.5
2             Airport   0.0
3  Light Rail Station   0.0
4           Pet Store   0.0


----St. James Town----
                  venue  freq
0  Fast Food Restaurant  0.13
1        Discount Store  0.07
2        Breakfast Spot  0.07
3           Coffee Shop  0.07
4            Nail Salon  0.07
```

```
----St. James Town, Cabbagetown----
                    venue  freq
0             Pizza Place   0.5
1  Furniture / Home Store   0.5
2      Light Rail Station   0.0
3               Pet Store   0.0
4                    Park   0.0


----Stn A PO Boxes----
                  venue  freq
0        Hardware Store  0.08
1       Supplement Shop  0.08
2  Fast Food Restaurant  0.08
3     Convenience Store  0.08
4        Sandwich Place  0.08


----Toronto Dominion Centre, Design Exchange----
                  venue  freq
0        Ice Cream Shop  0.05
1         Burrito Place  0.05
2  Fast Food Restaurant  0.05
3           Pizza Place  0.05
4     Fish & Chips Shop  0.05


----University of Toronto, Harbord----
                  venue  freq
0        Discount Store  0.25
1           Coffee Shop  0.25
2        Sandwich Place  0.25
3  Fast Food Restaurant  0.25
4               Airport  0.00
```

Let's put that into a pandas dataframe

In [88]:

```
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 [ ]:

```

```

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

In [93]:

```
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'] = T_grouped['Neighborhood']

for ind in np.arange(T_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(T_grouped.i
loc[ind, :], num_top_venues)

neighborhoods_venues_sorted.head()
```

Out[93]:

	Neighborhood	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Most Common Venue	8th I Com Ve
0	CN Tower, King and Spadina, Railway Lands, Har...	Light Rail Station	Yoga Studio	Restaurant	Auto Workshop	Brewery	Burrito Place	Butcher	C S
1	Central Bay Street	Pharmacy	Grocery Store	Pizza Place	Coffee Shop	Bank	Farmers Market	Curling Ice	De S
2	Christie	Park	Food & Drink Shop	Yoga Studio	Fast Food Restaurant	Curling Ice	Dessert Shop	Diner	Disc S
3	Church and Wellesley	Pizza Place	Coffee Shop	Chinese Restaurant	Middle Eastern Restaurant	Discount Store	Intersection	Sandwich Place	Fa Restau
4	Commerce Court, Victoria Hotel	Park	Playground	Lawyer	Farmers Market	Convenience Store	Curling Ice	Dessert Shop	[

4. Cluster Neighborhoods

In [94]:

```
# set number of clusters
kclusters = 5

T_grouped_clustering = T_grouped.drop('Neighborhood', 1)

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

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

Out[94]:

```
array([1, 1, 3, 1, 3, 0, 1, 1, 1, 1], dtype=int32)
```

Let's create a new dataframe that includes the cluster as well as the top 10 venues for each neighborhood.

In [133]:

```
T_merged = T_data.copy()
# merge toronto_grouped with toronto_data to add latitude/longitude for each neighbor
hood
T_merged =T_merged.join(neighborhoods_venues_sorted.set_index('Neighborhood'), on='Ne
ighbourhood')

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

Out[133]:

	index	District	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
0	2	Downtown Toronto	Regent Park, Harbourfront	43.763573	-79.188711	1.0	Intersection	Mexican Restaurant	Medical Center
1	4	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.773136	-79.239476	1.0	Fried Chicken Joint	Athletics & Sports	Thai Restaurant
2	9	Downtown Toronto	Garden District, Ryerson	43.692657	-79.264848	1.0	College Stadium	Skating Rink	General Entertainment
3	15	Downtown Toronto	St. James Town	43.799525	-79.318389	1.0	Fast Food Restaurant	Coffee Shop	Breakfast Spot
4	20	Downtown Toronto	Berczy Park	43.757490	-79.374714	NaN	NaN	NaN	NaN

In [188]:

```
T_merged[T_merged['Cluster Labels'].isnull()].replace(np.nan, 0, regex=True)
```

Out[188]:

	index	District	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th M Comm Ver
4	20	Downtown Toronto	Berczy Park	43.75749	-79.374714	0.0	0	0	0	

```
In [190]:
```

```
Out[190]:
```

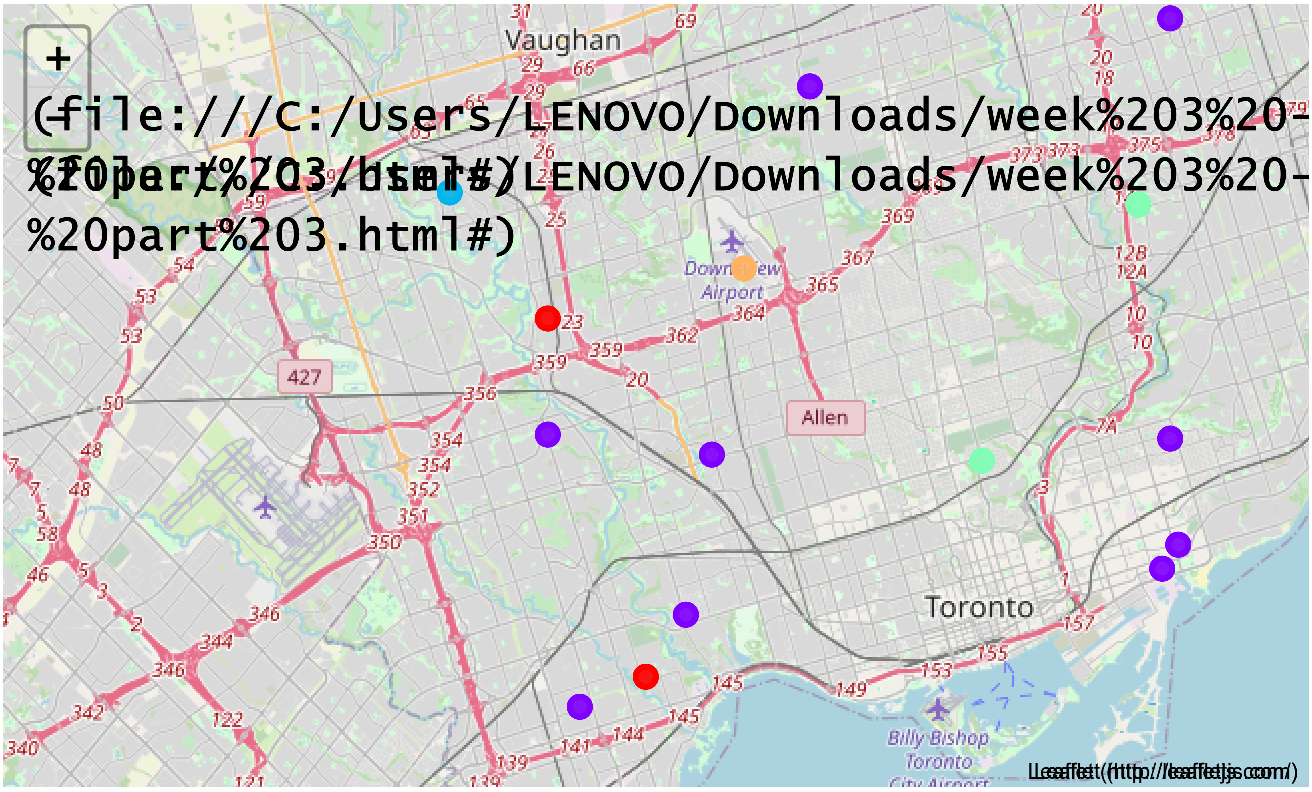
	index	District	Neighbourhood	Latitude	Longitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Mos Commo Venu
0	2	Downtown Toronto	Regent Park, Harbourfront	43.763573	-79.188711	1.0	Intersection	Mexican Restaurant	Medic Cente
1	4	Downtown Toronto	Queen's Park, Ontario Provincial Government	43.773136	-79.239476	1.0	Fried Chicken Joint	Athletics & Sports	Tha Restaurai
2	9	Downtown Toronto	Garden District, Ryerson	43.692657	-79.264848	1.0	College Stadium	Skating Rink	Genera Entertainmei
3	15	Downtown Toronto	St. James Town	43.799525	-79.318389	1.0	Fast Food Restaurant	Coffee Shop	Breakfa Spo
5	24	Downtown Toronto	Central Bay Street	43.782736	-79.442259	1.0	Pharmacy	Grocery Store	Pizza Plac
6	25	Downtown Toronto	Christie	43.753259	-79.329656	3.0	Park	Food & Drink Shop	Yoga Stud
7	30	Downtown Toronto	Richmond, Adelaide, King	43.737473	-79.464763	4.0	Airport	Snack Place	Pai
8	36	Downtown Toronto	Harbourfront East, Union Station, Toronto Islands	43.695344	-79.318389	1.0	Video Store	Athletics & Sports	Pai
9	42	Downtown Toronto	Toronto Dominion Centre, Design Exchange	43.668999	-79.315572	1.0	Restaurant	Movie Theater	Ice Crea Sho
10	48	Downtown Toronto	Commerce Court, Victoria Hotel	43.689574	-79.383160	3.0	Park	Playground	Lawye
11	80	Downtown Toronto	University of Toronto, Harbord	43.691116	-79.476013	1.0	Fast Food Restaurant	Coffee Shop	Discou Sto
12	84	Downtown Toronto	Kensington Market, Chinatown, Grange Park	43.651571	-79.484450	1.0	Café	Sushi Restaurant	Coffee Sho
13	87	Downtown Toronto	CN Tower, King and Spadina, Railway Lands, Har...	43.662744	-79.321558	1.0	Light Rail Station	Yoga Studio	Restaurai
14	91	Downtown Toronto	Rosedale	43.636258	-79.498509	0.0	Park	Baseball Field	Fish & Chip Sho
15	92	Downtown Toronto	Stn A PO Boxes	43.628841	-79.520999	1.0	Kids Store	Grocery Store	Gy
16	96	Downtown Toronto	St. James Town, Cabbagetown	43.756303	-79.565963	2.0	Pizza Place	Furniture / Home Store	Fast Foo Restaurai
17	97	Downtown Toronto	First Canadian Place, Underground city	43.724766	-79.532242	0.0	Baseball Field	Yoga Studio	Fish & Chip Sho
18	99	Downtown Toronto	Church and Wellesley	43.696319	-79.532242	1.0	Pizza Place	Coffee Shop	Chines Restaurai

Finally, let's visualize the resulting clusters

In [192]:

```
T_merged.drop([4],inplace = True)




import matplotlib.cm as cm
import matplotlib.colors as colors

# create map
map_clusters = folium.Map(location=[latitude, longitude], 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]

T_nonan = T_merged.dropna(subset=['Cluster Labels'])

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

map_clusters
```

Out[192]:


+
−
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
(file:///C:/Users/LENOVO/Downloads/week%203%20-%20part%203.html#)
Vaughan
Downsview Airport
Allen
Toronto
Billy Bishop Toronto City Airport
Leaflet (http://leafletjs.com/)

In [ ]:

5. Examine Clusters

cluster 1

```
In [122]: T_merged.loc[T_merged['Cluster Labels'] == 0, T_merged.columns[[1] + list(range(5, T_
          merged.shape[1]))]]
```

Out[122]:

	District	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 Mos Commo Venu
14	Downtown Toronto	0.0	Park	Baseball Field	Fish & Chips Shop	Curling Ice	Dessert Shop	Diner	Discount Store	Electronic Stor
17	Downtown Toronto	0.0	Baseball Field	Yoga Studio	Fish & Chips Shop	Curling Ice	Dessert Shop	Diner	Discount Store	Electronic Stor

cluster 2

In [123]:
```
T_merged.loc[T_merged['Cluster Labels'] == 1, T_merged.columns[[1] + list(range(5, T_merged.shape[1]))]]
```

Out[123]:

	District	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 Mo Comm Ven
0	Downtown Toronto	1.0	Intersection	Mexican Restaurant	Medical Center	Bank	Electronics Store	Rental Car Location	Restaura
1	Downtown Toronto	1.0	Fried Chicken Joint	Athletics & Sports	Thai Restaurant	Bakery	Bank	Hakka Restaurant	Gas Stati
2	Downtown Toronto	1.0	College Stadium	Skating Rink	General Entertainment	Café	Curling Ice	Dessert Shop	Dir
3	Downtown Toronto	1.0	Fast Food Restaurant	Coffee Shop	Breakfast Spot	Nail Salon	Discount Store	Pharmacy	Pizza Pla
5	Downtown Toronto	1.0	Pharmacy	Grocery Store	Pizza Place	Coffee Shop	Bank	Farmers Market	Curling I
8	Downtown Toronto	1.0	Video Store	Athletics & Sports	Park	Curling Ice	Spa	Beer Store	Skati Ri
9	Downtown Toronto	1.0	Restaurant	Movie Theater	Ice Cream Shop	Pub	Burrito Place	Pet Store	Pa
11	Downtown Toronto	1.0	Fast Food Restaurant	Coffee Shop	Discount Store	Sandwich Place	Yoga Studio	Curling Ice	Dess Sh
12	Downtown Toronto	1.0	Café	Sushi Restaurant	Coffee Shop	Italian Restaurant	Pub	Pizza Place	Gourm Sh
13	Downtown Toronto	1.0	Light Rail Station	Yoga Studio	Restaurant	Auto Workshop	Brewery	Burrito Place	Butch
15	Downtown Toronto	1.0	Kids Store	Grocery Store	Gym	Bakery	Tanning Salon	Wings Joint	Suppleme Sh
18	Downtown Toronto	1.0	Pizza Place	Coffee Shop	Chinese Restaurant	Middle Eastern Restaurant	Discount Store	Intersection	Sandwi Pla

cluster 3

In [124]:
```
T_merged.loc[T_merged['Cluster Labels'] == 2, T_merged.columns[[1] + list(range(5, T_merged.shape[1]))]]
```

Out[124]:

	District	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 M Com Ve
16	Downtown Toronto	2.0	Pizza Place	Furniture / Home Store	Fast Food Restaurant	Convenience Store	Curling Ice	Dessert Shop	Diner	Disc S

cluster 4

In [125]:

```
T_merged.loc[T_merged['Cluster Labels'] == 3, T_merged.columns[[1] + list(range(5, T_
merged.shape[1]))]]
```

Out[125]:

	District	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 Cor \
6	Downtown Toronto	3.0	Park	Food & Drink Shop	Yoga Studio	Fast Food Restaurant	Curling Ice	Dessert Shop	Diner	Dis
10	Downtown Toronto	3.0	Park	Playground	Lawyer	Farmers Market	Convenience Store	Curling Ice	Dessert Shop	

cluster 5

In [126]:

```
T_merged.loc[T_merged['Cluster Labels'] == 4, T_merged.columns[[1] + list(range(5, T_
merged.shape[1]))]]
```

Out[126]:

	District	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 Mos Commo Venu
7	Downtown Toronto	4.0	Airport	Snack Place	Park	Gas Station	Farmers Market	Convenience Store	Curling Ice	Desse Sho

thank you

In [ ]:

```
```