Capstone Project - The Battle of Neighbourhoods

Problem Description

Nowadays, there are a lot of food bloggers in each and every city, who go around discovering places to fill up our stomach and heart every time we feel hungry.

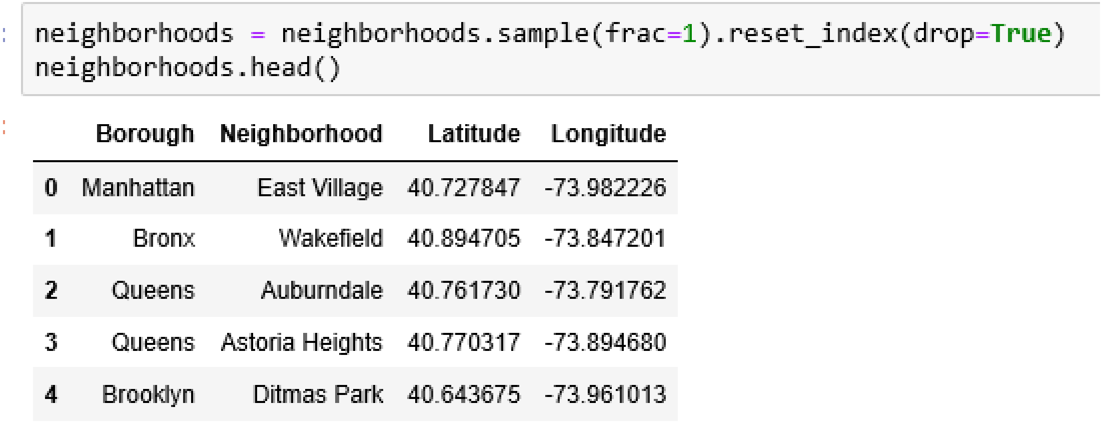
So here I planned to group together places where you would find different genres of food.

Data Description

The dataset to be used would be the New-York dataset which was previously used in Optional Assignment of Week-2.

The types of establishments I'll be targeting are (but not limited to) restaurants, cafes, bars, delis, bakeries, bistros and more.

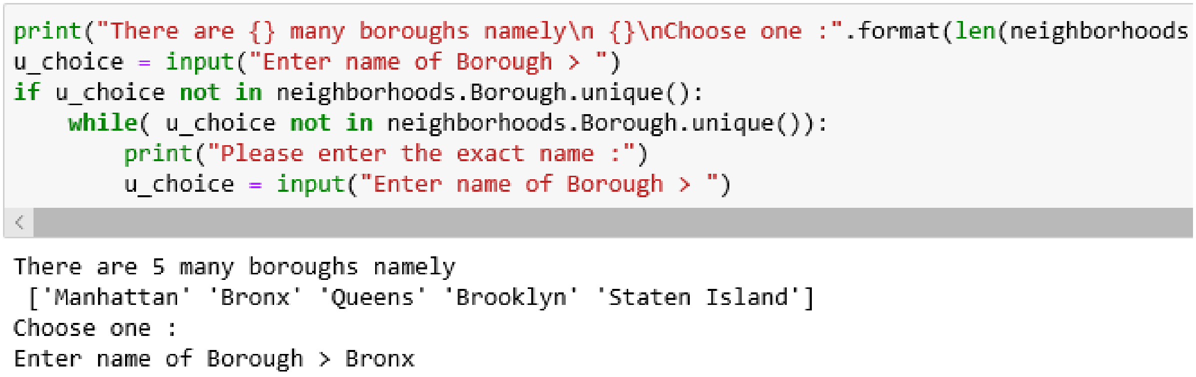
I plan segregate these places from others for each borough, and group them together. If possible, I may also provide a ranking for each cluster.



```
neighborhoods = neighborhoods.sample(frac=1).reset_index(drop=True)
neighborhoods.head()
```

	Borough	Neighborhood	Latitude	Longitude
0	Manhattan	East Village	40.727847	-73.982226
1	Bronx	Wakefield	40.894705	-73.847201
2	Queens	Auburndale	40.761730	-73.791762
3	Queens	Astoria Heights	40.770317	-73.894680
4	Brooklyn	Ditmas Park	40.643675	-73.961013

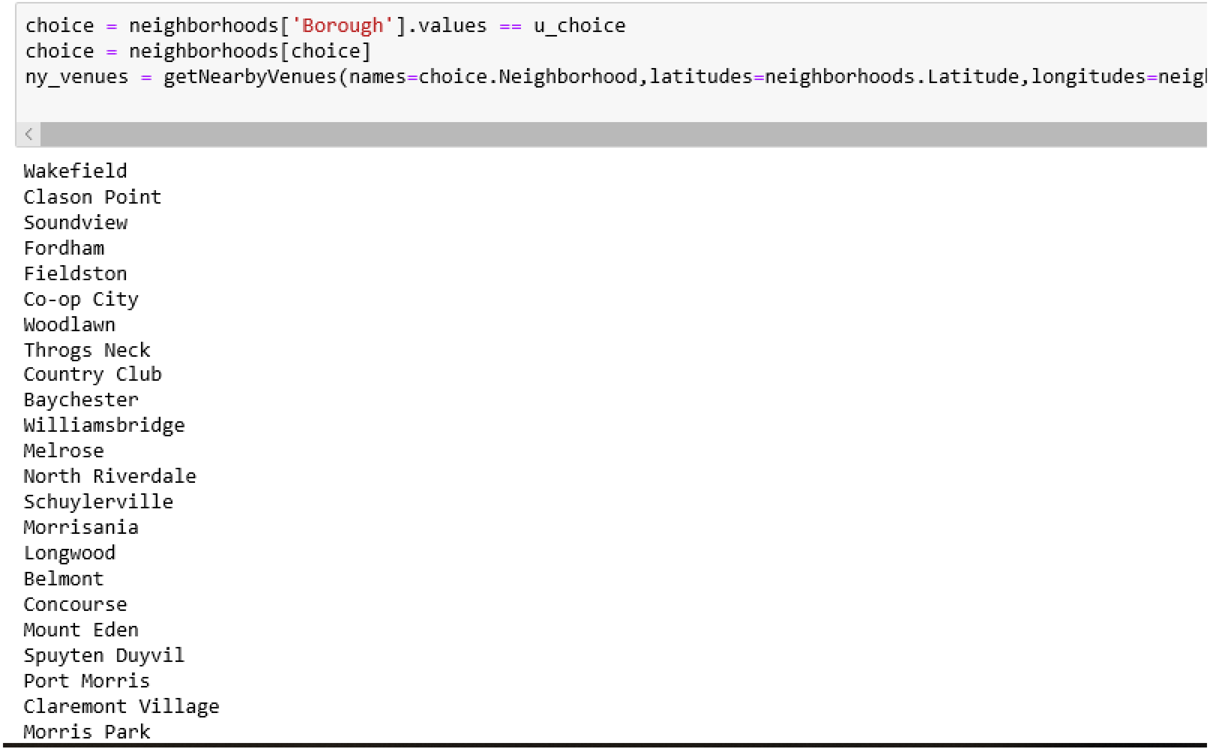
Working



```
print("There are {} many boroughs namely\n {}\nChoose one :".format(len(neighborhoods
u_choice = input("Enter name of Borough > ")
if u_choice not in neighborhoods.Borough.unique():
    while( u_choice not in neighborhoods.Borough.unique()):
        print("Please enter the exact name :")
        u_choice = input("Enter name of Borough > ")
```

```
There are 5 many boroughs namely
 ['Manhattan' 'Bronx' 'Queens' 'Brooklyn' 'Staten Island']
Choose one :
Enter name of Borough > Bronx
```

Here we see the user selects one Borough out of 5, this selection helps to narrow down the list of neighbours in later cells.

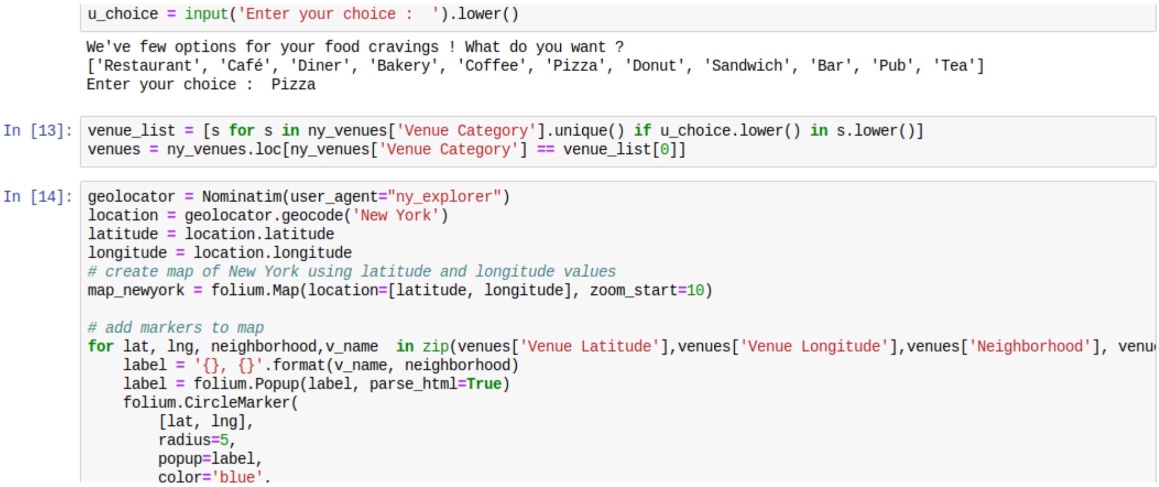


```
choice = neighborhoods['Borough'].values == u_choice
choice = neighborhoods[choice]
ny_venues = getNearbyVenues(names=choice.Neighborhood,latitudes=neighborhoods.Latitude,longitudes=neig
```

```
Wakefield
Clason Point
Soundview
Fordham
Fieldston
Co-op City
Woodlawn
Throgs Neck
Country Club
Baychester
Williamsbridge
Melrose
North Riverdale
Schuylerville
Morrisania
Longwood
Belmont
Concourse
Mount Eden
Spuyten Duyvil
Port Morris
Claremont Village
Morris Park
```

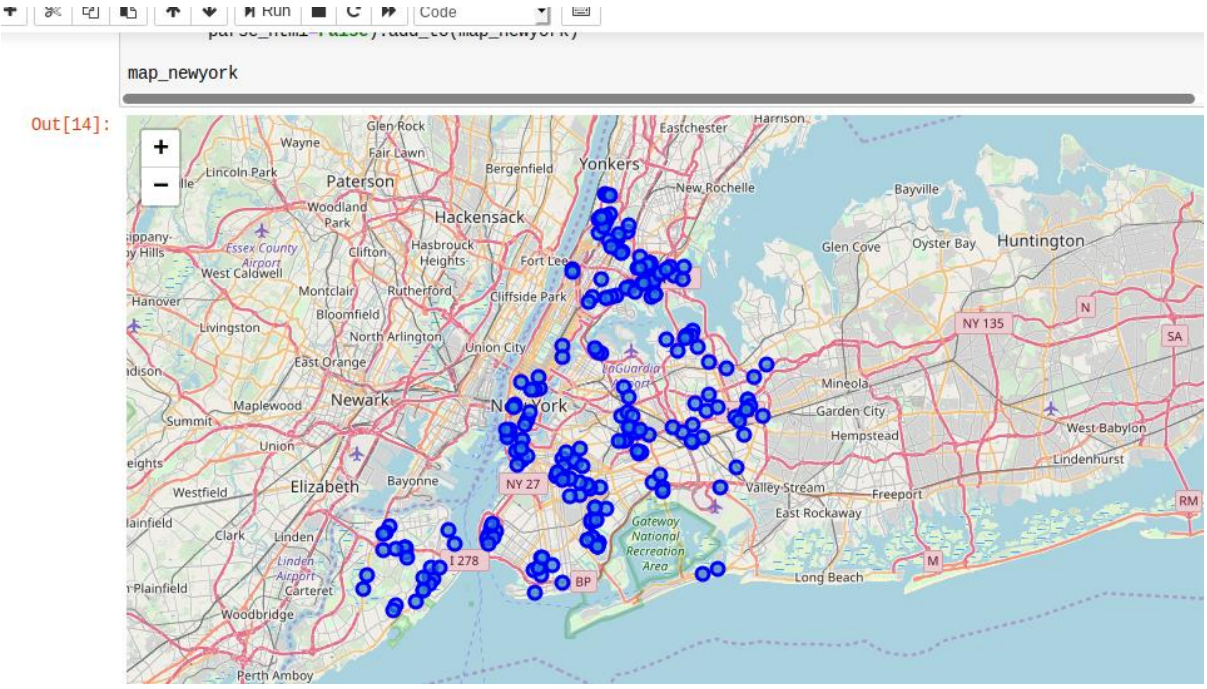
Since the list of neighbours are only from the selected borough, this process speeds up quite a bit.

Results



```
u_choice = input('Enter your choice :  ').lower()

We've few options for your food cravings ! What do you want ?
['Restaurant', 'Café', 'Diner', 'Bakery', 'Coffee', 'Pizza', 'Donut', 'Sandwich', 'Bar', 'Pub', 'Tea']
Enter your choice :  Pizza

In [13]: venue_list = [s for s in ny_venues['Venue Category'].unique() if u_choice.lower() in s.lower()]
         venues = ny_venues.loc[ny_venues['Venue Category'] == venue_list[0]]

In [14]: geolocator = Nominatim(user_agent="ny_explorer")
         location = geolocator.geocode('New York')
         latitude = location.latitude
         longitude = location.longitude
         # 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, neighborhood,v_name  in zip(venues['Venue Latitude'],venues['Venue Longitude'],venues['Neighborhood'], venu
             label = '{}, {}'.format(v_name, neighborhood)
             label = folium.Popup(label, parse_html=True)
             folium.CircleMarker(
                 [lat, lng],
                 radius=5,
                 popup=label,
                 color='blue',
```

Here, I’ve searched for Pizza places within the borough Bronx. Then we get the results in map.



```
    parse_html=False).add_to(map_newyork)

map_newyork

Out[14]:
```

Conclusion

This work is performed on limited data. If a good amount of data is available, I could have introduced a ranking system for each place. Also a bigger dataset could mean bigger area to search for.