Clustering Assignment part2

In [1]:
```
import pandas as pd
import numpy as np
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
import requests
from bs4 import BeautifulSoup
import os
import folium # map rendering library
from geopy.geocoders import Nominatim # convert an address into latitude and l
ongitude values
# Matplotlib and associated plotting modules
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import matplotlib.colors as colors
%matplotlib inline
```

In [2]:
```
def geo_location(address):
    geolocator = Nominatim(user_agent="canada")
    location = geolocator.geocode(address)
    latitude = location.latitude
    longitude = location.longitude
    return latitude,longitude
```

In [3]:
```
def get_venues(lat,lng):

    #set variables
    CLIENT_ID='AD3YDLBLIW54WQIF33CSDJTEBNZ5EBKTDZV0SY5H3LURMJYZ'
    CLIENT_SECRET='MJFLOCTIU2VQ4N5HB2H1YOC23WXW32TENKPFSQIEA0ATMUGC'
    VERSION = '20190325'
    #url to fetch data from foursquare api
    url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_s
ecret={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng)

    # get all the data
    results = requests.get(url).json()
    venue_data=results["response"]['groups'][0]['items']
    venue_details=[]
    for row in venue_data:
        try:
            venue_id=row['venue']['id']
            venue_name=row['venue']['name']
            venue_category=row['venue']['categories'][0]['name']
            venue_details.append([venue_id,venue_name,venue_category])
        except KeyError:
            pass

    column_names=['ID','Name','Category']
    df = pd.DataFrame(venue_details,columns=column_names)
    return df
```

In [5]:
```
df = pd.read_csv('toronto.csv')
df.head(12)
```

Out[5]:

	Unnamed: 0	Postcode	Borough	Neighbourhood
0	0	M5L	Downtown Toronto	Commerce Court
1	1	M3A	North York	Parkwoods
2	2	M9N	York	Weston
3	3	M3J	North York	Northwood Park, York University
4	4	M4H	East York	Thorncliffe Park
5	5	M5S	Downtown Toronto	University of Toronto
6	6	M9R	Etobicoke	Kingsview Village
7	7	M5A	Downtown Toronto	Harbourfront, Regent Park
8	8	M1G	Scarborough	Woburn
9	9	M1K	Scarborough	Ionview, Kennedy Park
10	10	M4T	Central Toronto	Moore Park
11	11	M4E	East Toronto	The Beaches

In [8]:
```
df.columns=['Postalcode','Latitude','Longitude']
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-8-b5f27f84df5f> in <module>
----> 1 df.columns=['Postalcode','Latitude','Longitude']

~\Anaconda3\lib\site-packages\pandas\core\generic.py in __setattr__(self, nam
e, value)
   5078         try:
   5079             object.__getattribute__(self, name)
-> 5080             return object.__setattr__(self, name, value)
   5081         except AttributeError:
   5082             pass

pandas/_libs/properties.pyx in pandas._libs.properties.AxisProperty.__set__()

~\Anaconda3\lib\site-packages\pandas\core\generic.py in _set_axis(self, axis,
labels)
    636
    637     def _set_axis(self, axis, labels):
--> 638         self._data.set_axis(axis, labels)
    639         self._clear_item_cache()
    640

~\Anaconda3\lib\site-packages\pandas\core\internals\managers.py in set_axis(s
elf, axis, new_labels)
    153             raise ValueError(
    154                 'Length mismatch: Expected axis has {old} elements, n
ew '
--> 155                 'values have {new} elements'.format(old=old_len, new=
new_len))
    156
    157         self.axes[axis] = new_labels

ValueError: Length mismatch: Expected axis has 4 elements, new values have 3
 elements
```

In [13]:
```
toronto_data
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-13-f2105f3293b7> in <module>
----> 1 toronto_data

NameError: name 'toronto_data' is not defined
```

In [ ]:
```

```