In [1]:
```
import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
import requests
```

In [2]:
```
url="https://en.wikipedia.org/wiki/List_of_postal_codes_of_Canada:_M"
header = {
  "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KH
TML, like Gecko) Chrome/50.0.2661.75 Safari/537.36",
  "X-Requested-With": "XMLHttpRequest"
}

r = requests.get(url, headers=header)

tables = pd.read_html(r.text)
```

In [3]:
```
df=pd.DataFrame(tables[0])

# The dataframe will consist of three columns: PostalCode, Borough, and
 Neighborhood

df.columns=['Postcode','Borough','Neighbourhood']

df.drop([0],axis=0,inplace=True)

df.reset_index()

# Only process the cells that have an assigned borough. Ignore cells wi
th a borough that is Not assigned.

df.drop(df[df['Borough']=="Not assigned"].index,axis=0, inplace=True)

# More than one neighborhood can exist in one postal code area.
# For example, in the table on the Wikipedia page,
# you will notice that M5A is listed twice and has two neighborhoods:
```

```
# Harbourfront and Regent Park. These two rows will be combined into on
e row with the neighborhoods
# separated with a comma as shown in row 11 in the above table.

df1=df.groupby("Postcode").agg(lambda x:','.join(set(x)))

# If a cell has a borough but a Not assigned neighborhood,
# then the neighborhood will be the same as the borough.
# So for the 9th cell in the table on the Wikipedia page,
# the value of the Borough and the Neighborhood columns will be Queen's
 Park.

df1.loc[df1['Neighbourhood']=="Not assigned",'Neighbourhood']=df1.loc[d
f1['Neighbourhood']=="Not assigned",'Borough']

df1.shape
```

Out[3]: (103, 2)

In [4]:
```
from bs4 import BeautifulSoup
soup = BeautifulSoup(requests.get(url).text,'lxml')
My_table = soup.find('table',{'class':'wikitable sortable'})

My_table
table_rows = My_table.find_all('tr')

t=[]
for tr in table_rows:
    td = tr.find_all('td')
    row = [tr.text.rstrip('\n') for tr in td]
    t.append(row)
df=pd.DataFrame(t)

df.columns=['Postcode','Borough','Neighbourhood']

df.drop([0],axis=0,inplace=True)

df.reset_index()
```

```
# Only process the cells that have an assigned borough. Ignore cells wi
th a borough that is Not assigned.

df.drop(df[df['Borough']=="Not assigned"].index,axis=0, inplace=True)

# More than one neighborhood can exist in one postal code area.
# For example, in the table on the Wikipedia page,
# you will notice that M5A is listed twice and has two neighborhoods:
# Harbourfront and Regent Park. These two rows will be combined into on
e row with the neighborhoods
# separated with a comma as shown in row 11 in the above table.

df1=df.groupby("Postcode").agg(lambda x:','.join(set(x)))

# If a cell has a borough but a Not assigned neighborhood,
# then the neighborhood will be the same as the borough.
# So for the 9th cell in the table on the Wikipedia page,
# the value of the Borough and the Neighborhood columns will be Queen's
 Park.

df1.loc[df1['Neighbourhood']=="Not assigned",'Neighbourhood']=df1.loc[d
f1['Neighbourhood']=="Not assigned",'Borough']

df1.shape
df1
```

Out[4]:

Postcode	Borough	Neighbourhood
M1B	Scarborough	Malvern, Rouge
M1C	Scarborough	Rouge Hill, Port Union, Highland Creek
M1E	Scarborough	Guildwood, Morningside, West Hill
M1G	Scarborough	Woburn
M1H	Scarborough	Cedarbrae
...	...	...
M9N	York	Weston

Postcode	Borough	Neighbourhood
M9P	Etobicoke	Westmount
M9R	Etobicoke	Kingsview Village, St. Phillips, Martin Grove ...
M9V	Etobicoke	South Steeles, Silverstone, Humbergate, Jamest...
M9W	Etobicoke	Northwest, West Humber - Clairville

103 rows × 2 columns

In [6]:
```
geo_data=pd.read_csv("https://cocl.us/Geospatial_data")
geo_data
```

Out[6]:

	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 [7]:
```
df1['Latitude']=geo_data['Latitude'].values
df1['Longitude']=geo_data['Longitude'].values

df1
```

Out[7]:

Postcode	Borough	Neighbourhood	Latitude	Longitude
M1B	Scarborough	Malvern, Rouge	43.806686	-79.194353
M1C	Scarborough	Rouge Hill, Port Union, Highland Creek	43.784535	-79.160497
M1E	Scarborough	Guildwood, Morningside, West Hill	43.763573	-79.188711
M1G	Scarborough	Woburn	43.770992	-79.216917
M1H	Scarborough	Cedarbrae	43.773136	-79.239476
...	...	...	...	...
M9N	York	Weston	43.706876	-79.518188
M9P	Etobicoke	Westmount	43.696319	-79.532242
M9R	Etobicoke	Kingsview Village, St. Phillips, Martin Grove ...	43.688905	-79.554724
M9V	Etobicoke	South Steeles, Silverstone, Humbergate, Jamest...	43.739416	-79.588437
M9W	Etobicoke	Northwest, West Humber - Clairville	43.706748	-79.594054

103 rows × 4 columns

In [8]:
```
df1.shape
```

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

In [ ]:
```

```