In [5]:

```
#Importing required libraries
import numpy as np
import pandas as pd

!pip install geopy

from geopy.geocoders import Nominatim
try:
    import geocoder
except:
    !pip install geocoder
    import geocoder

!pip install beautifulsoup4


import requests
from bs4 import BeautifulSoup

try:
    import folium
except:
    !pip install folium
    import folium

from sklearn.cluster import KMeans
```

```
Collecting geopy
  Downloading https://files.pythonhosted.org/packages/07/e1/9c72de674d5c2b8fcb0738a5ceeb5
424941fefa080bfe4e240d0bacb5a38/geopy-2.0.0-py3-none-any.whl (111kB)
     |████████████████████████████████| 112kB 18.3MB/s eta 0:00:01
Collecting geographiclib<2,>=1.49 (from geopy)
  Downloading https://files.pythonhosted.org/packages/8b/62/26ec95a98ba64299163199e95ad1b
0e34ad3f4e176e221c40245f211e425/geographiclib-1.50-py3-none-any.whl
Installing collected packages: geographiclib, geopy
Successfully installed geographiclib-1.50 geopy-2.0.0
Collecting geocoder
  Downloading https://files.pythonhosted.org/packages/4f/6b/13166c909ad2f2d76b929a4227c95
2630ebaf0d729f6317eb09cbceccbab/geocoder-1.38.1-py2.py3-none-any.whl (98kB)
     |████████████████████████████████| 102kB 4.8MB/s ta 0:00:011
Collecting ratelim (from geocoder)
  Downloading https://files.pythonhosted.org/packages/f2/98/7e6d147fd16a10a5f821db6e25f19
2265d6ecca3d82957a4fdd592cad49c/ratelim-0.1.6-py2.py3-none-any.whl
Requirement already satisfied: six in /home/jupyterlab/conda/envs/python/lib/python3.6/si
te-packages (from geocoder) (1.15.0)
Collecting future (from geocoder)
  Downloading https://files.pythonhosted.org/packages/45/0b/38b06fd9b92dc2b68d58b75f900e9
7884c45bedd2ff83203d933cf5851c9/future-0.18.2.tar.gz (829kB)
     |████████████████████████████████| 829kB 5.8MB/s eta 0:00:01
Collecting click (from geocoder)
  Using cached https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4f8d338c2fd15c8
d33fdd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3
.6/site-packages (from geocoder) (2.24.0)
Requirement already satisfied: decorator in /home/jupyterlab/conda/envs/python/lib/python
3.6/site-packages (from ratelim->geocoder) (4.4.2)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/pyt
hon3.6/site-packages (from requests->geocoder) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/l
ib/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/jupyterla
b/conda/envs/python/lib/python3.6/site-packages (from requests->geocoder) (1.25.11)
Requirement already satisfied: chardet<4,>=3.0.2 in /home/jupyterlab/conda/envs/python/li
b/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/a0/81daf51dcd359a9377b110
```

```
a8a886b3895921802d2fc1b2397e
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
Collecting beautifulsoup4
  Downloading https://files.pythonhosted.org/packages/d1/41/e6495bd7d3781cee623ce23ea6ac7
3282a373088fcd0ddc809a047b18eae/beautifulsoup4-4.9.3-py3-none-any.whl (115kB)
     |████████████████████████████████| 122kB 6.1MB/s eta 0:00:01
Collecting soupsieve>1.2; python_version >= "3.0" (from beautifulsoup4)
  Downloading https://files.pythonhosted.org/packages/6f/8f/457f4a5390eeae1cc3aeab89deb77
24c965be841ffca6cfca9197482e470/soupsieve-2.0.1-py3-none-any.whl
Installing collected packages: soupsieve, beautifulsoup4
Successfully installed beautifulsoup4-4.9.3 soupsieve-2.0.1
```

In [6]:

```
g = geocoder.arcgis('Copenhagen, Denmark')
blr_lat = g.latlng[0]
blr_lng = g.latlng[1]
print("The Latitude and Longitude of Copenhagen is {} and {}".format(blr_lat, blr_lng))
```

```
The Latitude and Longitude of Copenhagen is 55.67567000000008 and 12.567560000000071
```

In [7]:

```
#Getting the list of localities from Wikipedia site; Suburbs of Copenhagen
neig = requests.get("https://commons.wikimedia.org/wiki/Category:Suburbs_of_Copenhagen").
text
```

In [8]:

```
soup = BeautifulSoup(neig, 'html.parser')
#Creating a new list in order to store the neighborhood data
neighborhoodlist = []
for i in soup.find_all('div', class_='mw-category')[0].find_all('a'):
    neighborhoodlist.append(i.text)

#Creating a dataframe from the list
neig_df = pd.DataFrame({"Locality": neighborhoodlist})
neig_df.head()
```

Out[8]:

	Locality
0	Albertslund
1	Avedøre
2	Bagsværd
3	Ballerup
4	Brøndby

In [9]:

```
#Shape of dataframe neig_df
neig_df.shape
```

Out[9]:

```
(39, 1)
```

In [10]:

```
#Defining a function to get the location of the identified localities from above
def get_location(localities):
    g = geocoder.arcgis('{}, Copenhagen, Denmark'.format(localities))
    get_latlng = g.latlng
    return get_latlng
```

In [11]:

```
co_ordinates = []
for i in neig_df["Locality"].tolist():
    co_ordinates.append(get_location(i))
print(co_ordinates)
```

```
[[55.6827106498026, 12.485215158980598], [55.67567000000008, 12.567560000000071], [55.675
67000000008, 12.567560000000071], [55.67567000000008, 12.567560000000071], [55.6756700000
0008, 12.567560000000071], [55.67567000000008, 12.567560000000071], [55.75425000000007, 1
2.568710000000067], [55.705865370235756, 12.465741984268632], [55.67567000000008, 12.5675
600000000071], [55.67567000000008, 12.567560000000071], [55.67567000000008, 12.567560000000
0071], [55.681250000000034, 12.588890000000049], [55.722668222308464, 12.490995026644642]
, [55.79500000000007, 12.536620000000028], [55.74588000000006, 12.425380000000075], [55.6
9522569398116, 12.545710217862117], [55.67567000000008, 12.567560000000071], [55.66339000
000005, 12.467910000000074], [55.681250000000034, 12.588890000000049], [55.69354504611975
, 12.542825044385538], [55.650352550056816, 12.620930140317434], [55.77833000000004, 12.5
90260000000058], [55.77209000000005, 12.505550000000028], [55.696428772349506, 12.5381919
15314966], [55.67567000000008, 12.567560000000071], [55.71589787024535, 12.46935908694819
5], [55.693789810736, 12.539641296857791], [55.67567000000008, 12.567560000000071], [55.6
7567000000008, 12.567560000000071], [55.697153104547354, 12.54366010969018], [55.67567000
000008, 12.567560000000071], [55.67567000000008, 12.567560000000071], [55.70021616422939,
12.477099543068103], [55.69581004590533, 12.545415052466751], [55.67567000000008, 12.5675
600000000071], [55.67567000000008, 12.567560000000071], [55.67567000000008, 12.567560000000
0071], [55.67567000000008, 12.567560000000071], [55.67567000000008, 12.567560000000071]]
```

In [12]:

```
#Creating a dataframe from the list of location
co_ordinates_df = pd.DataFrame(co_ordinates, columns=['Latitudes', 'Longitudes'])
```

In [13]:

```
#Adding co-ordinates to the neig_df dataframe
neig_df["Latitudes"] = co_ordinates_df["Latitudes"]
neig_df["Longitudes"] = co_ordinates_df["Longitudes"]
```

In [14]:

```
neig_df.head()
```

Out[14]:

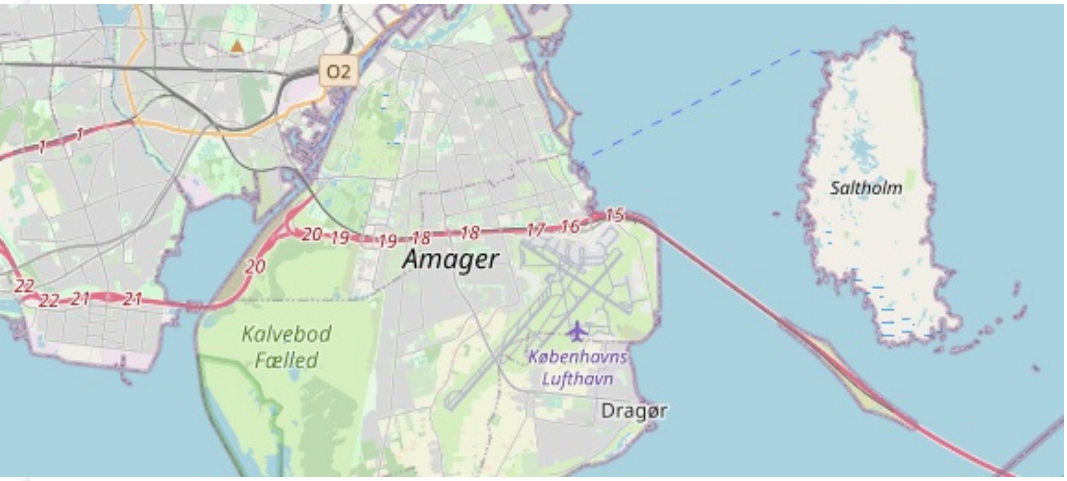
	Locality	Latitudes	Longitudes
0	Albertslund	55.682711	12.485215
1	Avedøre	55.675670	12.567560
2	Bagsværd	55.675670	12.567560
3	Ballerup	55.675670	12.567560
4	Brøndby	55.675670	12.567560

In [44]:

```
#Creating a map
blr_map = folium.Map(location=[blr_lat, blr_lng],zoom_start=11)

#adding markers to the map and #marker for Copenhagen
folium.Marker([blr_lat, blr_lng], popup='<i>Copenhagen</i>').add_to(blr_map)

#markers for localities
for latitude,longitude,name in zip(neig_df["Latitudes"], neig_df["Longitudes"], neig_df[
"Locality"]):
    folium.CircleMarker(
        [latitude, longitude],
        radius=6,
        color='blue',
        popup=name,
        fill=True,
        fill_color='#3186ff'
```

```
    ).add_to(blr_map)

blr_map
```

Out[44]:



In [45]:

```
#Foursquare Credentials
# @hidden_cell
CLIENT_ID = 'V5JVEGFG4DBXKZHZ2VQT44SGEAT2SHI30DVYQHBCAWC30UXD'
CLIENT_SECRET = 'T4KYVGROLUQZD2401UCIJOYWYKRG0L4Z54YA4LHVZ0VG0YRM'
VERSION = '20180605' # Foursquare API version

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

```
Your credentails:
CLIENT_ID: CLIENT_ID
CLIENT_SECRET:CLIENT_SECRET
```

In [46]:

```
#Getting the top 100 venues in each locality
radius = 2000
LIMIT = 100

venues = []

for lat, lng, locality in zip(neig_df["Latitudes"], neig_df["Longitudes"], neig_df["Locality"]):
    url = 'https://api.foursquare.com/v2/venues/explore?client_id={}&client_secret={}&ll={},{}&v={}&radius={}&limit={}'.format(CLIENT_ID, CLIENT_SECRET, lat, lng, VERSION, radius, LIMIT)
    results = requests.get(url).json()['response']['groups'][0]['items']

    for venue in results:
        venues.append((locality, lat, lng, venue['venue']['name'], venue['venue']['location']['lat'], venue['venue']['location']['lng'], venue['venue']['categories'][0]['name']))
```

In [47]:

```
venues[0]
```

Out[47]:

```
('Albertslund',
 55.6827106498026,
 12.485215158980598,
 'Que Pasa',
 55.68415300860668,
 12.485997478267718,
 'Mexican Restaurant')
```

In [48]:

```
#Converting the venue list into dataframe
venues_df = pd.DataFrame(venues)
venues_df.columns = ['Locality', 'Latitude', 'Longitude', 'Venue name', 'Venue Lat', 'Venue Lng', 'Venue Category']
venues_df.head()
```

Out[48]:

	Locality	Latitude	Longitude	Venue name	Venue Lat	Venue Lng	Venue Category
0	Albertslund	55.682711	12.485215	Que Pasa	55.684153	12.485997	Mexican Restaurant
1	Albertslund	55.682711	12.485215	Miki's Pizzeria	55.681376	12.484521	Pizza Place
2	Albertslund	55.682711	12.485215	Crossfit Vanløse - Værkstedet	55.684073	12.490756	Gym
3	Albertslund	55.682711	12.485215	Armando's	55.692925	12.489601	Pizza Place
4	Albertslund	55.682711	12.485215	Lagkagehuset	55.687747	12.490913	Bakery

In [49]:

```
#Number of venues per Locality
venues_df.groupby(['Locality']).count()
```

Out[49]:

Locality	Latitude	Longitude	Venue name	Venue Lat	Venue Lng	Venue Category
Albertslund	96	96	96	96	96	96
Avedøre	100	100	100	100	100	100
Bagsværd	100	100	100	100	100	100
Ballerup	100	100	100	100	100	100
Brøndby	100	100	100	100	100	100
Buddinge	100	100	100	100	100	100
Charlottenlund	80	80	80	80	80	80
Dyssegård	53	53	53	53	53	53
Ejby (Storkøbenhavn)	100	100	100	100	100	100
Gentofte	100	100	100	100	100	100
Glostrup	100	100	100	100	100	100
Greve Strand	100	100	100	100	100	100
Hareskovby	49	49	49	49	49	49
Hjortekær (Lyngby-Taarbæk Kommune)	12	12	12	12	12	12
Hjortespring	20	20	20	20	20	20
Holte	100	100	100	100	100	100
Hundige	100	100	100	100	100	100

Locality	Latitude	Longitude	Venue name	Venue Lat	Venue Lng	Venue Category
Hvidovre	46	46	46	46	46	46
Ishøj Strand	100	100	100	100	100	100
Jægersborg	100	100	100	100	100	100
Kastrup	100	100	100	100	100	100
Klampenborg	37	37	37	37	37	37
Kongens Lyngby	60	60	60	60	60	60
Lundtofte	100	100	100	100	100	100
Måløv	100	100	100	100	100	100
Mørkhøj (Gladsaxe)	43	43	43	43	43	43
Nærum	100	100	100	100	100	100
Ordrup	100	100	100	100	100	100
Rødovre	100	100	100	100	100	100
Skodsborg	100	100	100	100	100	100
Skovlunde	100	100	100	100	100	100
Skovshoved	100	100	100	100	100	100
Solrød Strand	72	72	72	72	72	72
Sorgenfri	100	100	100	100	100	100
Søborg	100	100	100	100	100	100
Taarbæk	100	100	100	100	100	100
Tårnby	100	100	100	100	100	100
Vangede	100	100	100	100	100	100
Virum	100	100	100	100	100	100

In [50]:

```
#Unique categories of the above venues
print('There are {} unique categries.'.format(len(venues_df['Venue Category'])))
```

```
There are 3368 unique categries.
```

In [51]:

```
#List of categories
print('Total number of unique catefories are {}'.format(len(venues_df['Venue Category'].
unique().tolist())))
#First 10 categories
venues_df['Venue Category'].unique().tolist()#[:10]
```

```
Total number of unique catefories are 172
```

Out[51]:

```
['Mexican Restaurant',
 'Pizza Place',
 'Gym',
 'Bakery',
 'Park',
 'Supermarket',
 'Grocery Store',
 'Gym / Fitness Center',
 'Korean Restaurant',
 'Soccer Field',
 'Sports Bar',
 'Shopping Mall',
 'Multiplex',
 'Gastropub',
 'Turkish Restaurant',
 'Concert Hall',
 'Convenience Store',
 'Bar',
```

'Ice Cream Shop',
'Gaming Cafe',
'Discount Store',
'Asian Restaurant',
'Café',
'Flower Shop',
'Fish & Chips Shop',
'Scandinavian Restaurant',
'Train Station',
'Thai Restaurant',
'Fast Food Restaurant',
'Indian Restaurant',
'Juice Bar',
'Plaza',
'Clothing Store',
'American Restaurant',
'Department Store',
'Light Rail Station',
'Lake',
'Burger Joint',
'Pharmacy',
'Food & Drink Shop',
'Hookah Bar',
'Sporting Goods Shop',
'Martial Arts School',
'Smoke Shop',
'Metro Station',
'Bus Station',
'Electronics Store',
'Apres Ski Bar',
'Gas Station',
'Bookstore',
'Theme Park',
'Cocktail Bar',
'Beer Bar',
'Music Venue',
'Theme Park Ride / Attraction',
'Breakfast Spot',
'Movie Theater',
'Art Museum',
'Hotel',
'Hostel',
'History Museum',
'French Restaurant',
'Camera Store',
'Coffee Shop',
'Comic Shop',
'Dessert Shop',
'Theater',
'Gift Shop',
'Library',
'Pie Shop',
'Furniture / Home Store',
'Australian Restaurant',
'Restaurant',
'Health & Beauty Service',
'Toy / Game Store',
'Monument / Landmark',
'Food Court',
'Wine Bar',
'Pier',
'Capitol Building',
'Tapas Restaurant',
'Seafood Restaurant',
'Sandwich Place',
'Indie Movie Theater',
'Beer Store',
'Hot Dog Joint',
'Roof Deck',
'BBQ Joint',
'Playground',
'Pool',

'Taco Place',
'Kitchen Supply Store',
'Salad Place',
'Wine Shop',
'Historic Site',
'Other Great Outdoors',
'Sushi Restaurant',
'Beach',
'Steakhouse',
'Flea Market',
'Italian Restaurant',
'Food Truck',
'Track',
'Racetrack',
'Stadium',
'Art Gallery',
'Kebab Restaurant',
'Tea Room',
'Garden',
'Campground',
'Sports Club',
'Scenic Lookout',
'Falafel Restaurant',
'Chinese Restaurant',
'Buffet',
'Trail',
'Harbor / Marina',
'South American Restaurant',
'Vietnamese Restaurant',
'Palace',
'Opera House',
'Street Food Gathering',
'Pub',
'Vegetarian / Vegan Restaurant',
'Church',
'Botanical Garden',
'Fountain',
'Boat Rental',
'Design Studio',
'Badminton Court',
'Food Stand',
'Hockey Arena',
'Arts & Crafts Store',
'Jazz Club',
'Lounge',
'Antique Shop',
'College Cafeteria',
'Rest Area',
'Farm',
'Gym Pool',
'Hockey Rink',
'Athletics & Sports',
'Intersection',
'Advertising Agency',
'Cemetery',
'Greek Restaurant',
'Middle Eastern Restaurant',
'Deli / Bodega',
'African Restaurant',
'Record Shop',
'Skate Park',
'North Indian Restaurant',
'Yoga Studio',
'Ramen Restaurant',
'Molecular Gastronomy Restaurant',
'Water Park',
'South Indian Restaurant',
'Nature Preserve',
'Rock Club',
'Mediterranean Restaurant',
'Performing Arts Venue',
'Candy Store',

```
 'Fish Market',
 'General Entertainment',
 'Dog Run',
 'Soccer Stadium',
 'Diner',
 'Museum',
 'Garden Center',
 'Japanese Restaurant',
 'Moving Target',
 'Bus Line']
```

In [52]:

```
#encoding onehot
blr_onehot = pd.get_dummies(venues_df[['Venue Category']], prefix="", prefix_sep="")

blr_onehot['Locality'] = venues_df['Locality']

#moving the column, locality to the front
blr_onehot = blr_onehot[ [ 'Locality' ] + [ col for col in blr_onehot.columns if col!='L
ocality' ] ]
blr_onehot.head()
```

Out[52]:

	Locality	Advertising Agency	African Restaurant	American Restaurant	Antique Shop	Apres Ski Bar	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	...	Track	Trail	St
0	Albertslund	0	0	0	0	0	0	0	0	0	...	0	0	
1	Albertslund	0	0	0	0	0	0	0	0	0	...	0	0	
2	Albertslund	0	0	0	0	0	0	0	0	0	...	0	0	
3	Albertslund	0	0	0	0	0	0	0	0	0	...	0	0	
4	Albertslund	0	0	0	0	0	0	0	0	0	...	0	0	

5 rows × 173 columns

In [53]:

```
blr_grouped = blr_onehot.groupby(['Locality']).mean().reset_index()
print(blr_grouped.shape)
blr_grouped.head()
```

```
(39, 173)
```

Out[53]:

	Locality	Advertising Agency	African Restaurant	American Restaurant	Antique Shop	Apres Ski Bar	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	...	Track	Trail
0	Albertslund	0.0	0.0	0.010417	0.0	0.010417	0.0	0.00	0.0	0.010417	...	0.0	0.0
1	Avedøre	0.0	0.0	0.000000	0.0	0.000000	0.0	0.01	0.0	0.000000	...	0.0	0.0
2	Bagsværd	0.0	0.0	0.000000	0.0	0.000000	0.0	0.01	0.0	0.000000	...	0.0	0.0
3	Ballerup	0.0	0.0	0.000000	0.0	0.000000	0.0	0.01	0.0	0.000000	...	0.0	0.0
4	Brøndby	0.0	0.0	0.000000	0.0	0.000000	0.0	0.01	0.0	0.000000	...	0.0	0.0

5 rows × 173 columns

In [54]:

```
#numbers of localities having Italian Restaurants
```

```
len(blr_grouped[blr_grouped['Mexican Restaurant'] > 0])
```

Out[54]:

2

In [55]:

```
blr_mexican = blr_grouped[['Locality', 'Mexican Restaurant']]
blr_mexican.head()
```

Out[55]:

	Locality	Mexican Restaurant
0	Albertslund	0.010417
1	Avedøre	0.000000
2	Bagsværd	0.000000
3	Ballerup	0.000000
4	Brøndby	0.000000

In [56]:

```
#clustering K-means
cluster = 3

#Dataframe for clustering
blr_clustering = blr_mexican.drop(['Locality'], 1)

#running K-means clustering
k_means = KMeans(init="k-means++", n_clusters=cluster, n_init=12).fit(blr_clustering)

#getting the labels for first 10 locality
print(k_means.labels_[0:10])
```

```
[2 0 0 0 0 0 0 0 0 0]
```

In [57]:

```
#Creating a blr_mexican dataframe
blr_labels = blr_mexican.copy()

#adding label to blr_labels
blr_labels["Cluster Label"] = k_means.labels_

blr_labels.head()
```

Out[57]:

	Locality	Mexican Restaurant	Cluster Label
0	Albertslund	0.010417	2
1	Avedøre	0.000000	0
2	Bagsværd	0.000000	0
3	Ballerup	0.000000	0
4	Brøndby	0.000000	0

In [58]:

```
#Merging blr_labels and neig_df dataframes, in order to get the latitude and longitudes f
or each locality
blr_labels = blr_labels.join(neig_df.set_index('Locality'), on='Locality')
blr_labels.head()
```

Out[58]:

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
0	Albertslund	0.010417	2	55.682711	12.485215
1	Avedøre	0.000000	0	55.675670	12.567560
2	Bagsværd	0.000000	0	55.675670	12.567560
3	Ballerup	0.000000	0	55.675670	12.567560
4	Brøndby	0.000000	0	55.675670	12.567560

In [59]:

```
#Grouping the localities per their Cluster Labels
blr_labels.sort_values(["Cluster Label"], inplace=True)
blr_labels.head()
```

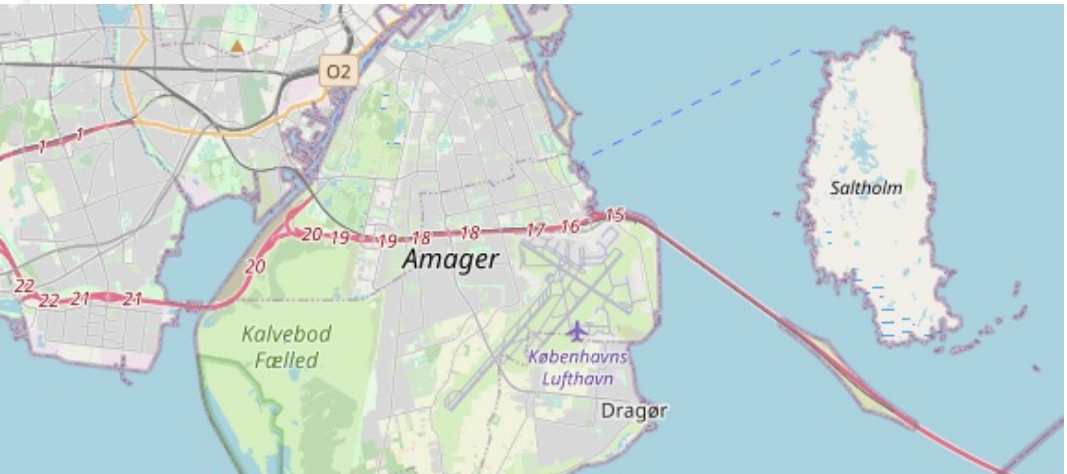
Out[59]:

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
19	Jægersborg	0.0	0	55.693545	12.542825
21	Klampenborg	0.0	0	55.778330	12.590260
22	Kongens Lyngby	0.0	0	55.772090	12.505550
23	Lundtofte	0.0	0	55.696429	12.538192
24	Måløv	0.0	0	55.675670	12.567560

In [61]:

```
#Plot the cluster on map
cluster_map = folium.Map(location=[blr_lat, blr_lng],zoom_start=11)

#marker for Bangalore
folium.Marker([blr_lat, blr_lng], popup='<i>Copenhagen</i>').add_to(cluster_map)

#Getting the colors for the clusters
col = ['red', 'green', 'blue']

#markers for localities
for latitude,longitude,name,clus in zip(blr_labels["Latitudes"], blr_labels["Longitudes"]
, blr_labels["Locality"], blr_labels["Cluster Label"]):
    label = folium.Popup(name + ' - Cluster ' + str(clus))
    folium.CircleMarker(
        [latitude, longitude],
        radius=6,
        color=col[clus],
        popup=label,
        fill=False,
        fill_color=col[clus],
        fill_opacity=0.3
    ).add_to(cluster_map)

cluster_map
```

Out[61]:



In [62]:

```
#Cluster 1
cluster_1 = blr_labels[blr_labels['Cluster Label'] == 0]
print("There are {} localities in cluster-1".format(cluster_1.shape[0]))
mean_presence_1 = cluster_1['Mexican Restaurant'].mean()
print("The mean occurence of Mexican restaurant in cluster-1 is {0:.2f}".format(mean_pres
ence_1))
cluster_1
```

```
There are 37 localities in cluster-1
The mean occurence of Mexican restaurant in cluster-1 is 0.00
```

Out[62]:

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
19	Jægersborg	0.0	0	55.693545	12.542825
21	Klampenborg	0.0	0	55.778330	12.590260
22	Kongens Lyngby	0.0	0	55.772090	12.505550
23	Lundtofte	0.0	0	55.696429	12.538192
24	Måløv	0.0	0	55.675670	12.567560
25	Mørkhøj (Gladsaxe)	0.0	0	55.715898	12.469359
26	Nærum	0.0	0	55.693790	12.539641
27	Ordrup	0.0	0	55.675670	12.567560
28	Rødovre	0.0	0	55.675670	12.567560
29	Skodsborg	0.0	0	55.697153	12.543660
30	Skovlunde	0.0	0	55.675670	12.567560
31	Skovshoved	0.0	0	55.675670	12.567560
33	Sorgenfri	0.0	0	55.695810	12.545415
34	Søborg	0.0	0	55.675670	12.567560
35	Taarbæk	0.0	0	55.675670	12.567560
36	Tårnby	0.0	0	55.675670	12.567560
20	Kastrup	0.0	0	55.650353	12.620930
37	Vangede	0.0	0	55.675670	12.567560
38	Virum	0.0	0	55.675670	12.567560
17	Hvidovre	0.0	0	55.663390	12.467910
1	Avedøre	0.0	0	55.675670	12.567560
2	Bagsværd	0.0	0	55.675670	12.567560
3	Ballerup	0.0	0	55.675670	12.567560
4	Brøndby	0.0	0	55.675670	12.567560
5	Buddinge	0.0	0	55.675670	12.567560
6	Charlottenlund	0.0	0	55.754250	12.568710

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
7	Dyssegård	0.0	0	55.705865	12.465742
18	Ishøj Strand	0.0	0	55.681250	12.588890
8	Ejby (Storkøbenhavn)	0.0	0	55.675670	12.567560
10	Glostrup	0.0	0	55.675670	12.567560
11	Greve Strand	0.0	0	55.681250	12.588890
12	Hareskovby	0.0	0	55.722668	12.490995
13	Hjortekær (Lyngby-Taarbæk Kommune)	0.0	0	55.795000	12.536620
14	Hjortespring	0.0	0	55.745880	12.425380
15	Holte	0.0	0	55.695226	12.545710
16	Hundige	0.0	0	55.675670	12.567560
9	Gentofte	0.0	0	55.675670	12.567560

In [64]:

```
#Cluster 2
cluster_2 = blr_labels[blr_labels['Cluster Label'] == 1]
print("There are {} localities in cluster-2".format(cluster_2.shape[0]))
mean_presence_2 = cluster_2['Mexican Restaurant'].mean()
print("The mean occurence of Mexican restaurant in cluster-2 is {0:.2f}".format(mean_pres
ence_2))
cluster_2
```

```
There are 1 localities in cluster-2
The mean occurence of Mexican restaurant in cluster-2 is 0.01
```

Out[64]:

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
32	Solrød Strand	0.013889	1	55.700216	12.4771

In [65]:

```
#Cluster 3
cluster_3 = blr_labels[blr_labels['Cluster Label'] == 2]
print("There are {} localities in cluster-3".format(cluster_3.shape[0]))
mean_presence_3 = cluster_3['Mexican Restaurant'].mean()
print("The mean occurence of Mexican restaurant in cluster-3 is {0:.2f}".format(mean_pres
ence_3))
cluster_3
```

```
There are 1 localities in cluster-3
The mean occurence of Mexican restaurant in cluster-3 is 0.01
```

Out[65]:

	Locality	Mexican Restaurant	Cluster Label	Latitudes	Longitudes
0	Albertslund	0.010417	2	55.682711	12.485215

Conclusion: The above analysis shows that Cluster 1 has many (37) restaurants, while Cluster 2 and 3 has almost no (1) Mexican restaurants. This shows that there is a great potential to open a Mexican restaurant, with limited competition, in 2 clusters of the suburbs of Copenhagen Additionally, it also shows that Cluster 1 is highly competitive with many venues categorized as Mexican Restaurants, thus not recommended to open a new restaurant here.

In [ ]: