Introduction

This project is part of the Coursera Capstone project and will utilise Foursquare API for venue information of Tokyo for analysis on the restaurants located in Tokyo. This analysis can be used for people who are looking to explore Tokyo and would like to know which locations have similar restaurants. Restaurant owners can also learn which areas to open their branches.

Data used

A table from https://en.wikipedia.org/wiki/Special_wards_of_Tokyo will be used for the location data. It will be imported into a dataframe using BeautifulSoup.

In [3]:
```
Tokyo_df = pd.read_csv('Tokyo_df_Coord.csv', index_col=[0])
#remove the unnamed column
Tokyo_df
```

Out[3]:

	Ward	Area_SqKm	Population	Major_District	Dist_Latitude	Dist_Longitude
1	Chiyoda	5100	59441	Nagatacho	35.675618	139.743469
2	Chuo	14460	147620	Nihonbashi	35.684058	139.774501
3	Minato	12180	248071	Odaiba	35.626722	139.772101
4	Shinjuku	18620	339211	Shinjuku	35.693763	139.703632
5	Bunkyō	19790	223389	Hongo	35.175386	137.013430

Geopy will be used to get the coordinates of the locations.

```
# use of geopy geocoders
from geopy.geocoders import Nominatim
geolocator = Nominatim(user_agent="Tokyo_explorer")
df['Major_Dist_Coord']= df['Major_District'].apply(geolocator.geocode).apply(lambda x: (x.latitude, x.
longitude))
df[['Latitude', 'Longitude']] = df['Major_Dist_Coord'].apply(pd.Series)
```

Foursquare will be used for venue data, I will query it for top 100 venues within 1km of the districts in Tokyo. The resultant JSON file will be imported into another dataframe.

In [10]:
```
# Create a Data-Frame out of it to Concentrate Only on Restaurants

Tokyo_restaurant = Tokyo_Venues[Tokyo_Venues['Venue_Category']\
                                                        .str.contains('Restaurant')].reset_index(dro
p=True)
Tokyo_restaurant.index = np.arange(1, len(Tokyo_restaurant)+1)
print ("Shape of the Data-Frame with Venue Category only Restaurant: ", Tokyo_restaurant.shape)
Tokyo_restaurant.head(3)
```

```
Shape of the Data-Frame with Venue Category only Restaurant:  (657, 7)
```

Out[10]:

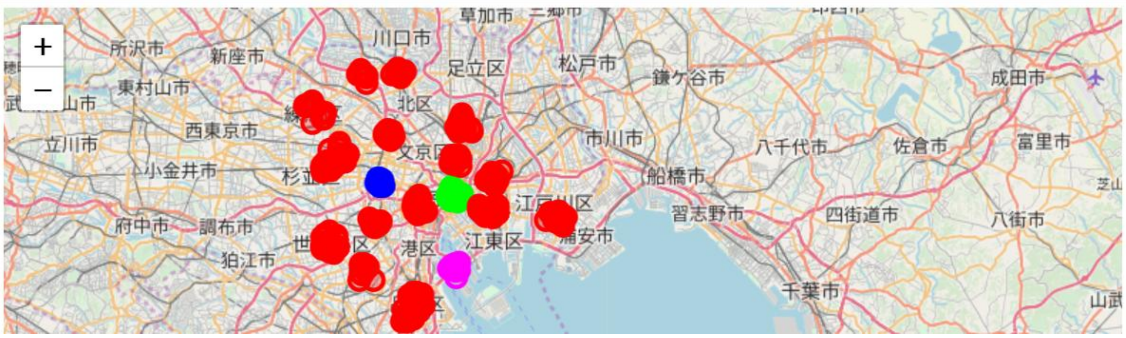
	District	Dist_Latitude	Dist_Longitude	Venue	Venue_Lat	Venue_Long	Venue_Category
1	Nagatacho	35.675618	139.743469	Nagatacho Kurosawa (永田町 黒澤)	35.674699	139.741737	Japanese Restaurant
2	Nagatacho	35.675618	139.743469	Shinamen Hashigo (支那麺 はしご)	35.672184	139.741576	Ramen Restaurant
3	Nagatacho	35.675618	139.743469	Sushi Isshin (鮨 一新)	35.672589	139.739399	Sushi Restaurant

Methodology

Using Folium, I show the venue locations on a map.

```
# add markers to the map
# markers_colors = []
for lat, lon, poi, distr in zip(Tokyo_restaurant['Venue_Lat'],
                                  Tokyo_restaurant['Venue_Long'],
                                  Tokyo_restaurant['Venue_Category'],
                                  Tokyo_restaurant['District']):
    label = folium.Popup(str(poi) + ' ' + str(distr), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=7,
        popup=label,
        color=rainbow[DD.index(distr)-1],
        fill=True,
        fill_color=rainbow[DD.index(distr)-1],
        fill_opacity=0.3).add_to(map_restaurants)

map_restaurants
```

Out[14]:



Then I get the top ten venues and their frequencies.

In [18]:

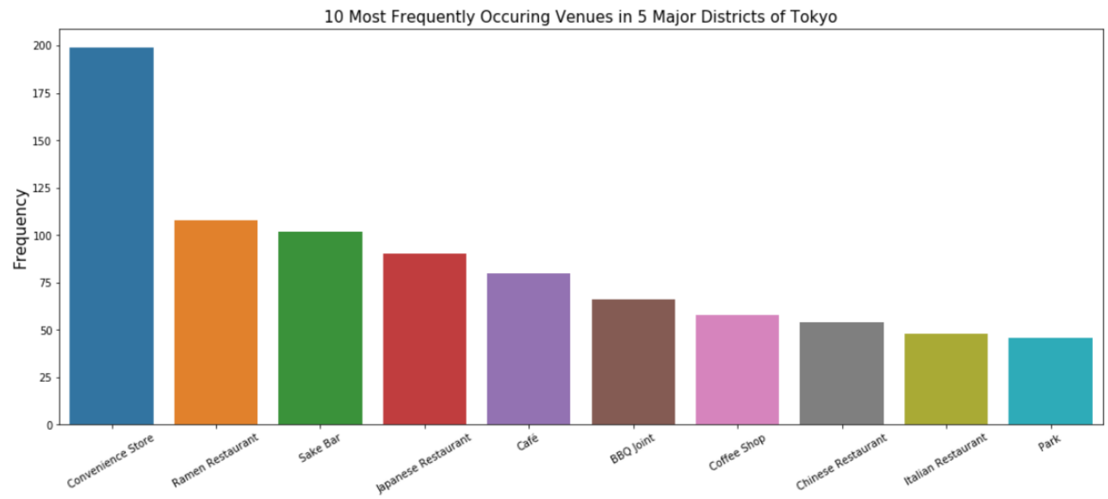
```
# create a dataframe of top 10 categories
Tokyo_Venues_Top10 = Tokyo_Venues['Venue_Category'].value_counts()[0:10].to_frame(name='frequency')
Tokyo_Venues_Top10=Tokyo_Venues_Top10.reset_index()
#Tokyo_5_Dist_Venues_Top10

Tokyo_Venues_Top10.rename(index=str, columns={"index": "Venue_Category", "frequency": "Frequency"}, in
place=True)
Tokyo_Venues_Top10
```

Out[18]:

	Venue_Category	Frequency
0	Convenience Store	199
1	Ramen Restaurant	108
2	Sake Bar	102
3	Japanese Restaurant	90
4	Café	80
5	BBQ Joint	66
6	Coffee Shop	58
7	Chinese Restaurant	54
8	Italian Restaurant	48
9	Park	46

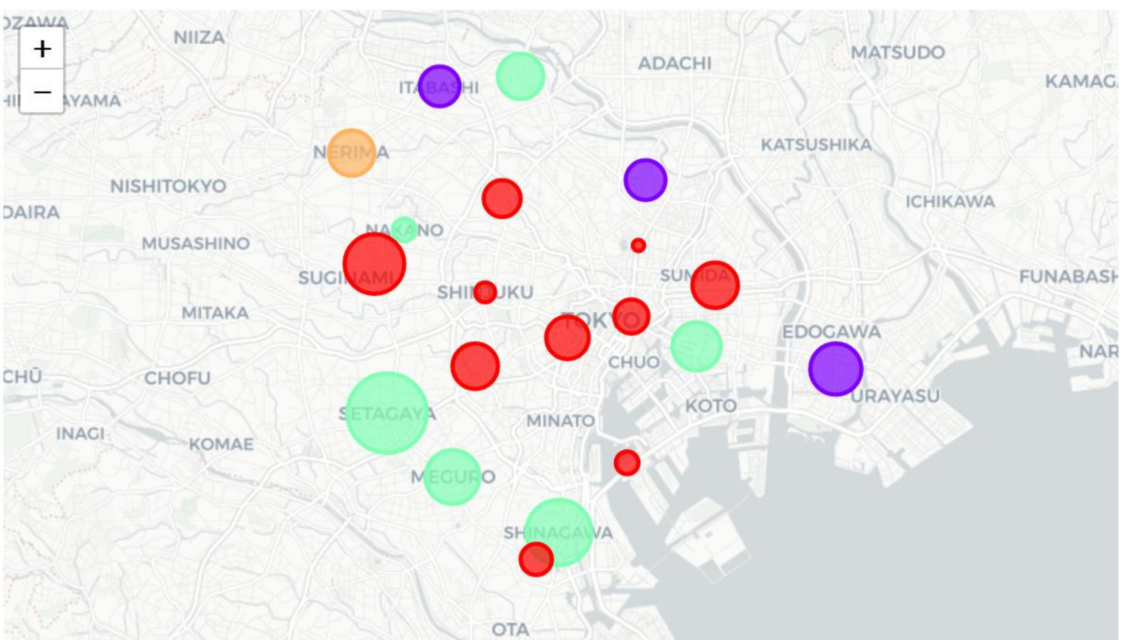
Using seaborn to show the results in a chart

In [19]:
```
import seaborn as sns
fig = plt.figure(figsize=(18,7))
s=sns.barplot(x="Venue_Category", y="Frequency", data=Tokyo_Venues_Top10)
s.set_xticklabels(s.get_xticklabels(), rotation=30)
plt.title('10 Most Frequently Occuring Venues in 5 Major Districts of Tokyo', fontsize=15)
plt.xlabel("Venue Category", fontsize=15)
plt.ylabel ("Frequency", fontsize=15)
plt.savefig("Most_Freq_Venues.png", dpi=300)
plt.show()
```



Using Kmeans 5 clusters

Out[46]:



Results

Cluster 1 is popular for Japanese restaurants

In [36]:
```
Tokyo_Cluster1 = Tokyo_Coordinate_Cluster_merged.loc[Tokyo_Coordinate_Cluster_merged['Cluster Label']
== 0,
                                              Tokyo_Coordinate_Cluster_merged.columns[[3] + list(range(4, Toky
Coordinate_Cluster_merged.shape[1]))]]
print ("No of Neighbourhood in Cluster Label 0: %d" %(Tokyo_Cluster1.shape[0]))
Tokyo_Cluster1
```

No of Neighbourhood in Cluster Label 0: 10

Out[36]:

	Major_District	Dist_Latitude	Dist_Longitude	Cluster Label	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
1	Nagatacho	35.675618	139.743469	0	Japanese Restaurant	BBQ Joint	Coffee Shop	Ramen Restaurant	Hotel
2	Nihonbashi	35.684058	139.774501	0	Japanese Restaurant	BBQ Joint	Soba Restaurant	Café	Chinese Restaurant
3	Odaiba	35.626722	139.772101	0	Italian Restaurant	Plaza	Theme Park	Exhibit	Coffee Shop
4	Shinjuku	35.693763	139.703632	0	Bar	Sake Bar	Ramen Restaurant	Japanese Restaurant	BBQ Joint
6	Ueno	35.711759	139.777645	0	Sake Bar	BBQ Joint	Japanese Restaurant	Chinese Restaurant	Café
7	Kinshicho	35.696312	139.815043	0	Ramen Restaurant	Sake Bar	Coffee Shop	Pub	Chinese Restaurant

Cluster 2 has convenience stores and grocery stores

In [37]:
```
Tokyo_Cluster2 = Tokyo_Coordinate_Cluster_merged.loc[Tokyo_Coordinate_Cluster_merged['Cluster Label']
== 1,
                                              Tokyo_Coordinate_Cluster_merged.columns[[3] + list(range(4, Tokyo_
Coordinate_Cluster_merged.shape[1]))]]
print ("No of Neighbourhood in Cluster Label 0: %d" %(Tokyo_Cluster2.shape[0]))
Tokyo_Cluster2
```

No of Neighbourhood in Cluster Label 0: 3

Out[37]:

	Major_District	Dist_Latitude	Dist_Longitude	Cluster Label	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
18	Arakawa	35.737529	139.781310	1	Convenience Store	Chinese Restaurant	Deli / Bodega	Grocery Store	Tram Station
19	Itabashi	35.774143	139.681209	1	Convenience Store	Bus Stop	Park	Grocery Store	Chinese Restaurant
23	Kasai	35.663400	139.873100	1	Convenience Store	Ramen Restaurant	Grocery Store	Steakhouse	Fast Food Restaurant

Cluster 3 is outlier with intersection and train station

In [38]:
```
Tokyo_Cluster3 = Tokyo_Coordinate_Cluster_merged.loc[Tokyo_Coordinate_Cluster_merged['Cluster Label']
== 2,
                                          Tokyo_Coordinate_Cluster_merged.columns[[3] + list(range(4, Tokyo_
Coordinate_Cluster_merged.shape[1]))]]
print ("No of Neighbourhood in Cluster Label 0: %d" %(Tokyo_Cluster3.shape[0]))
Tokyo_Cluster3
```

```
No of Neighbourhood in Cluster Label 0: 1
```

Out[38]:

	Major_District	Dist_Latitude	Dist_Longitude	Cluster Label	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6tl Co
22	Tateishi	33.481791	131.478154	2	Intersection	Italian Restaurant	Farmers Market	Train Station	Yoshoku Restaurant	Exl

Cluster 4 has convenience stores and ramen restaurants

In [39]:
```
Tokyo_Cluster4 = Tokyo_Coordinate_Cluster_merged.loc[Tokyo_Coordinate_Cluster_merged['Cluster Label']
== 3,
                                          Tokyo_Coordinate_Cluster_merged.columns[[3] + list(range(4, Tokyo_
Coordinate_Cluster_merged.shape[1]))]]
print ("No of Neighbourhood in Cluster Label 0: %d" %(Tokyo_Cluster4.shape[0]))
Tokyo_Cluster4
```

```
No of Neighbourhood in Cluster Label 0: 7
```

Out[39]:

	Major_District	Dist_Latitude	Dist_Longitude	Cluster Label	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th M Comn Ve
8	Kiba	35.672200	139.806100	3	Convenience Store	Ramen Restaurant	Coffee Shop	Japanese Restaurant	Steakhou
9	Shinagawa	35.599252	139.738910	3	Convenience Store	Ramen Restaurant	BBQ Joint	Grocery Store	Sake Bar
10	Meguro	35.621250	139.688014	3	Convenience Store	Café	Coffee Shop	Japanese Restaurant	Grocery Store
12	Setagaya	35.646096	139.656270	3	Convenience Store	Japanese Restaurant	Bakery	Sake Bar	Italian Restaura

Cluster 5 has convenience stores and cafes

In [40]:
```
Tokyo_Cluster5 = Tokyo_Coordinate_Cluster_merged.loc[Tokyo_Coordinate_Cluster_merged['Cluster Label']
== 4,
                                Tokyo_Coordinate_Cluster_merged.columns[[3] + list(range(4, Tokyo_
Coordinate_Cluster_merged.shape[1]))]]
print ("No of Neighbourhood in Cluster Label 0: %d" %(Tokyo_Cluster5.shape[0]))
Tokyo_Cluster5
```

```
No of Neighbourhood in Cluster Label 0: 2
```

Out[40]:

	Major_District	Dist_Latitude	Dist_Longitude	Cluster Label	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue
5	Hongo	35.175386	137.013430	4	Convenience Store	Intersection	Ramen Restaurant	Japanese Restaurant	Café
20	Nerima	35.748360	139.638735	4	Convenience Store	Park	Intersection	Grocery Store	Café

Conclusion

Tokyo is a big city with wide variety of locations and each one is different with some similarities. It is great to be able to use technology to analyze the location data using different techniques to show the similarities and differences of locations which would be otherwise difficult without the use of these technology.