Capstone Project — The Battle of Neighbourhoods¶

Introduction

Bangalore is the IT capital of India, where we find most of fresh graduates starting their carrier in IT(Software Industry).
With it's diverse culture, comes diverse food items. There are many restaurants in Bangalore, each belonging to different categories like Chinese , Italian , French etc. So as part of this project , we will list and visualize all major parts of Bangalore

Questions that can be asked using the above mentioned datasets

```
What is best location in Bangalore for Chinese Cuisine ?
Which areas have large number of Chinese Restaurant?
Which all areas have less number of restaurant ?
Which is the best place to stay if I prefer Chinese Cuisine ?
What places are have best restaurant in Bangalore?
```

Data

For this project we need the following data :

Bangalore Resturants data that contains list Locality, Resturant name,Rating along with their latitude and longitude.
Data source : Zomato kaggel dataset (https://www.kaggle.com/shrutimehta/zomato-restaurants-data)
Description : This data set contains the required information. And we will use this data set to explore various locality of Bangalore and near by places
Data source : Fousquare API
Description : By using this api we will get all the venues in each neighborhood.

Approach

Collect the Bangalore data from Zomato kaggel dataset
Using FourSquare API we will find all venues for each neighborhood.
Filter out all venues that are nearby by locality.
Using aggregative rating for each resturant to find the best places.
Visualize the Ranking of neighborhoods using folium library(python)

In [5]:

```
import pandas as pd
import numpy as np
import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dataframe
# Matplotlib and associated plotting modules
import matplotlib.cm as cm
import matplotlib.colors as colors
# import k-means from clustering stage
from sklearn.cluster import KMeans

!conda install -c conda-forge folium=0.5.0 --yes # uncomment this line if you haven't c
ompleted the Foursquare API lab
import folium # map rendering library
! pip install geocoder
import geocoder
```

```
Collecting package metadata (current_repodata.json): done
Solving environment: done

# All requested packages already installed.

Requirement already satisfied: geocoder in /opt/conda/envs/Python-3.7-mai
n/lib/python3.7/site-packages (1.38.1)
Requirement already satisfied: click in /opt/conda/envs/Python-3.7-main/li
b/python3.7/site-packages (from geocoder) (7.1.2)
Requirement already satisfied: six in /opt/conda/envs/Python-3.7-main/lib/
python3.7/site-packages (from geocoder) (1.15.0)
Requirement already satisfied: future in /opt/conda/envs/Python-3.7-main/l
ib/python3.7/site-packages (from geocoder) (0.18.2)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.7-mai
n/lib/python3.7/site-packages (from geocoder) (2.25.1)
Requirement already satisfied: ratelim in /opt/conda/envs/Python-3.7-main/
lib/python3.7/site-packages (from geocoder) (0.1.6)
Requirement already satisfied: decorator in /opt/conda/envs/Python-3.7-mai
n/lib/python3.7/site-packages (from ratelim->geocoder) (4.4.2)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/envs/Pytho
n-3.7-main/lib/python3.7/site-packages (from requests->geocoder) (2020.12.
5)
Requirement already satisfied: chardet<5,>=3.0.2 in /opt/conda/envs/Python
-3.7-main/lib/python3.7/site-packages (from requests->geocoder) (3.0.4)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/envs/Python-3.7-
main/lib/python3.7/site-packages (from requests->geocoder) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /opt/conda/envs/Py
thon-3.7-main/lib/python3.7/site-packages (from requests->geocoder) (1.26.
3)
```

Read zomata data from csv

In [26]:

```
pd.set_option('max_colwidth',40)
df = pd.read_csv('https://raw.githubusercontent.com/SuryaBoddu-1250/Capstone/main/zomato.csv',encoding='ISO-8859-1')

df.head(15)
```

Out[26]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	I
0	6317637	Le Petit Souffle	162	Makati City	Third Floor, Century City Mall, Kala...	Century City Mall, Poblacion, Makati...	Century City Mall, Poblacion, Makati...	1
1	6304287	Izakaya Kikufuji	162	Makati City	Little Tokyo, 2277 Chino Roces Avenu...	Little Tokyo, Legaspi Village, Makat...	Little Tokyo, Legaspi Village, Makat...	1
2	6300002	Heat - Edsa Shangri-La	162	Mandaluyong City	Edsa Shangri-La, 1 Garden Way, Ortig...	Edsa Shangri-La, Ortigas, Mandaluyon...	Edsa Shangri-La, Ortigas, Mandaluyon...	1
3	6318506	Ooma	162	Mandaluyong City	Third Floor, Mega Fashion Hall, SM M...	SM Megamall, Ortigas, Mandaluyong City	SM Megamall, Ortigas, Mandaluyong Ci...	1
4	6314302	Sambo Kojin	162	Mandaluyong City	Third Floor, Mega Atrium, SM Megamal...	SM Megamall, Ortigas, Mandaluyong City	SM Megamall, Ortigas, Mandaluyong Ci...	1
5	18189371	Din Tai Fung	162	Mandaluyong City	Ground Floor, Mega Fashion Hall, SM ...	SM Megamall, Ortigas, Mandaluyong City	SM Megamall, Ortigas, Mandaluyong Ci...	1
6	6300781	Buffet 101	162	Pasay City	Building K, SM By The Bay, Sunset Bo...	SM by the Bay, Mall of Asia Complex,...	SM by the Bay, Mall of Asia Complex,...	1
7	6301290	Vikings	162	Pasay City	Building B, By The Bay, Seaside Boul...	SM by the Bay, Mall of Asia Complex,...	SM by the Bay, Mall of Asia Complex,...	1
8	6300010	Spiral - Sofitel Philippine Plaza Ma...	162	Pasay City	Plaza Level, Sofitel Philippine Plaz...	Sofitel Philippine Plaza Manila, Pas...	Sofitel Philippine Plaza Manila, Pas...	1
9	6314987	Locavore	162	Pasig City	Brixton Technology Center, 10 Brixto...	Kapitolyo	Kapitolyo, Pasig City	1
10	6309903	Silantro Fil-Mex	162	Pasig City	75 East Capitol Drive, Kapitolyo, Pa...	Kapitolyo	Kapitolyo, Pasig City	1

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	I
11	6309455	Mad Mark's Creamery & Good Eats	162	Pasig City	23 East Capitol Drive, Kapitolyo, Pa...	Kapitolyo	Kapitolyo, Pasig City	1
12	6318433	Silantro Fil-Mex	162	Quezon City	Second Floor, UP Town Center, Katipu...	UP Town Center, Diliman, Quezon City	UP Town Center, Diliman, Quezon City...	1
13	6310470	Guevarra's	162	San Juan City	387 P. Guevarra Corner Argonne Stree...	Addition Hills	Addition Hills, San Juan City	1
14	6314605	Sodam Korean Restaurant	162	San Juan City	17 J. Abad Santos Drive, Little Bagu...	Little Baguio	Little Baguio, San Juan City	1

15 rows × 21 columns

In [7]:

```
df_india = df[df['Country Code'] == 1]
df_BLR = df_india[df_india['City'] == 'Bangalore']
df_BLR.reset_index(drop=True, inplace=True)
df_BLR.head()
```

Out[7]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitu
0	50943	Sultans of Spice	1	Bangalore	BluPetal Hotel, 60 Jyoti Nivas Colle...	BluPetal Hotel, Koramangala	BluPetal Hotel, Koramangala, Bangalore	77.6154
1	58268	The Fatty Bao - Asian Gastro Bar	1	Bangalore	610, 3rd Floor, 12th Main, Off 80 Fe...	Indiranagar	Indiranagar, Bangalore	77.6453
2	51705	Toit	1	Bangalore	298, Namma Metro Pillar 62, 100 Feet...	Indiranagar	Indiranagar, Bangalore	77.6407
3	18162866	Three Dots & A Dash	1	Bangalore	840/1,100 Feet Road, Metro Pillar 56...	Indiranagar	Indiranagar, Bangalore	77.6404
4	18407918	Bombay Brasserie	1	Bangalore	2989/B, 12th Main Road, HAL 2nd Stag...	Indiranagar	Indiranagar, Bangalore	77.6457

5 rows × 21 columns

Data cleaning - remove unwanted data

In [10]:

```
df_Res= df_BLR[df_BLR.Longitude !=0.000000][['Restaurant Name','Locality','Longitude',
'Latitude','Cuisines','Aggregate rating','Rating text','Votes']]
```

In [11]:

```
df_Res = df_Res[df_Res['Aggregate rating'] !=0.0]
```

In [12]:

```
df_Res.head()
```

Out[12]:

	Restaurant Name	Locality	Longitude	Latitude	Cuisines	Aggregate rating	Rating text	Votes
0	Sultans of Spice	BluPetal Hotel, Koramangala	77.615428	12.933284	North Indian, Mughlai	4.1	Very Good	2416
1	The Fatty Bao - Asian Gastro Bar	Indiranagar	77.645396	12.970221	Asian	4.7	Excellent	2369
2	Toit	Indiranagar	77.640709	12.979166	Italian, American, Pizza	4.8	Excellent	10934
3	Three Dots & A Dash	Indiranagar	77.640489	12.980410	European, Continental	3.9	Good	1354
4	Bombay Brasserie	Indiranagar	77.645748	12.970324	Modern Indian	4.2	Very Good	231

In [13]:

```
Bangalore_Rst = folium.Map(location=[12.97, 77.58], zoom_start=12)

X = df_Res['Latitude']
Y = df_Res['Longitude']
Z = np.stack((X, Y), axis=1)

kmeans = KMeans(n_clusters=5, random_state=0).fit(Z)

clusters = kmeans.labels_
colors = ['red', 'green', 'blue', 'yellow','orange']
df_Res ['Cluster'] = clusters

for latitude, longitude, Locality, cluster in zip(df_Res['Latitude'], df_Res['Longitud
e'], df_Res['Locality'], df_Res['Cluster']):
    label = folium.Popup(Locality, parse_html=True)
    folium.CircleMarker(
        [latitude, longitude],
        radius=5,
        popup=label,
        color='black',
        fill=True,
        fill_color=colors[cluster],
        fill_opacity=0.7).add_to(Bangalore_Rst)

Bangalore_Rst
```

Out[13]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [30]:

```
df_Res.head(15          )
```

Out[30]:

	Restaurant Name	Locality	Longitude	Latitude	Cuisines	Aggregate rating	Rating text	Vo
0	Sultans of Spice	BluPetal Hotel, Koramangala	77.615428	12.933284	North Indian, Mughlai	4.1	Very Good	2
1	The Fatty Bao - Asian Gastro Bar	Indiranagar	77.645396	12.970221	Asian	4.7	Excellent	2
2	Toit	Indiranagar	77.640709	12.979166	Italian, American, Pizza	4.8	Excellent	10
3	Three Dots & A Dash	Indiranagar	77.640489	12.980410	European, Continental	3.9	Good	1
4	Bombay Brasserie	Indiranagar	77.645748	12.970324	Modern Indian	4.2	Very Good	
5	Glen's Bakehouse	Indiranagar	77.640625	12.979096	Bakery, Desserts, Cafe	4.0	Very Good	3
6	Onesta	Indiranagar	77.643685	12.978453	Pizza, Cafe, Italian	4.3	Very Good	1
7	Onesta	JP Nagar	77.596791	12.906229	Pizza, Cafe, Italian	4.6	Excellent	
8	ECHOES Koramangala	Koramangala 5th Block	77.615797	12.934179	Continental, American, Italian, Nort...	4.7	Excellent	
9	Truffles	Koramangala 5th Block	77.614293	12.933298	American, Burger, Cafe	4.7	Excellent	9
10	The Black Pearl	Koramangala 5th Block	77.616155	12.934365	North Indian, European, Mediterranean	4.1	Very Good	5
11	Eat Street	Koramangala 6th Block	77.625999	12.939496	North Indian, Chinese, Italian, Stre...	4.3	Very Good	
12	Koramangala Social	Koramangala 7th Block	77.614130	12.935662	Continental, American	4.5	Excellent	1
13	AB's - Absolute Barbecues	Marathahalli	77.699386	12.949934	European, Mediterranean, North Indian	4.6	Excellent	6
14	Flechazo	Marathahalli	77.696664	12.975377	Asian, Mediterranean, North Indian	4.4	Very Good	

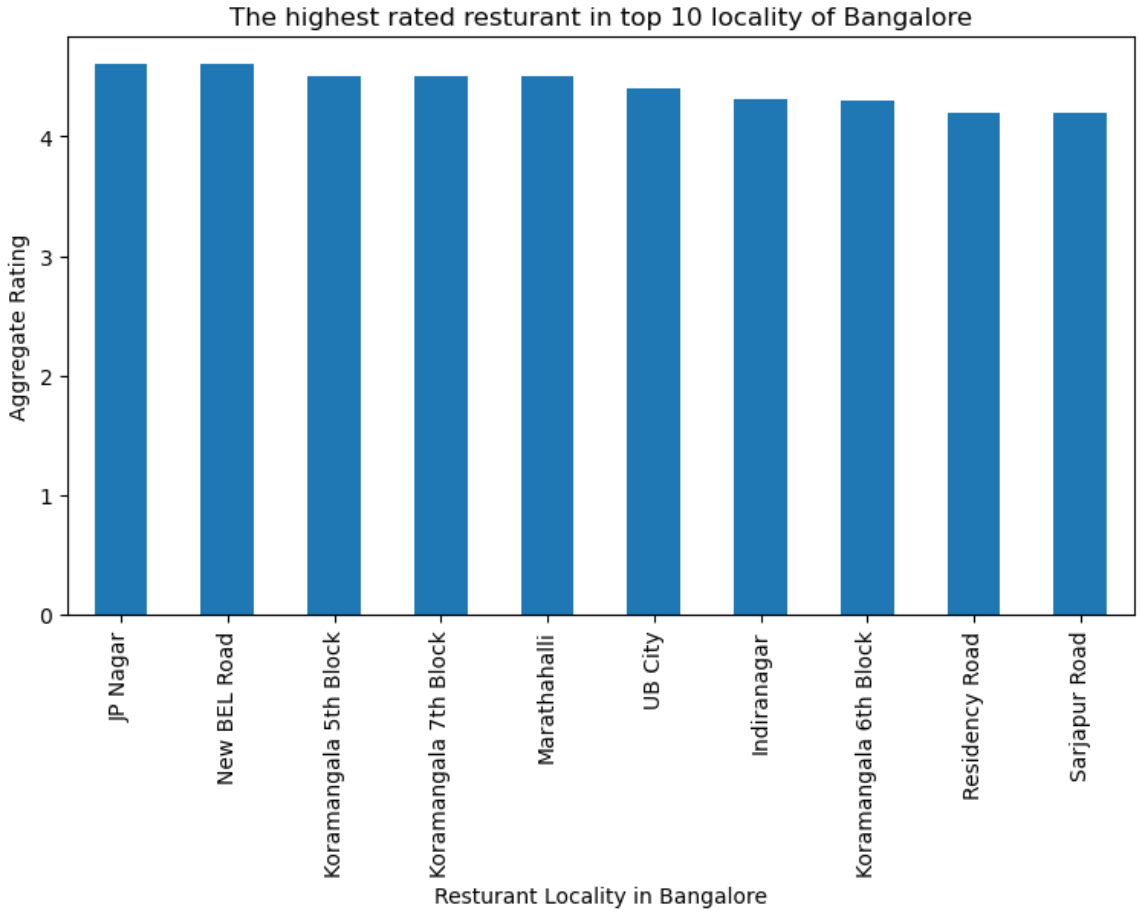
What places are have best restaurant in Bangalore?¶

In [15]:

```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('The highest rated resturant in top 10 locality of Bangalore')
#On x-axis

#giving a bar plot
df_Res.groupby('Locality')['Aggregate rating'].mean().nlargest(10).plot(kind='bar')

plt.xlabel('Resturant Locality in Bangalore')
#On y-axis
plt.ylabel('Aggregate Rating')
#displays the plot
plt.show()
```



what places are have worst restaurants in Bangalore?

In [18]:

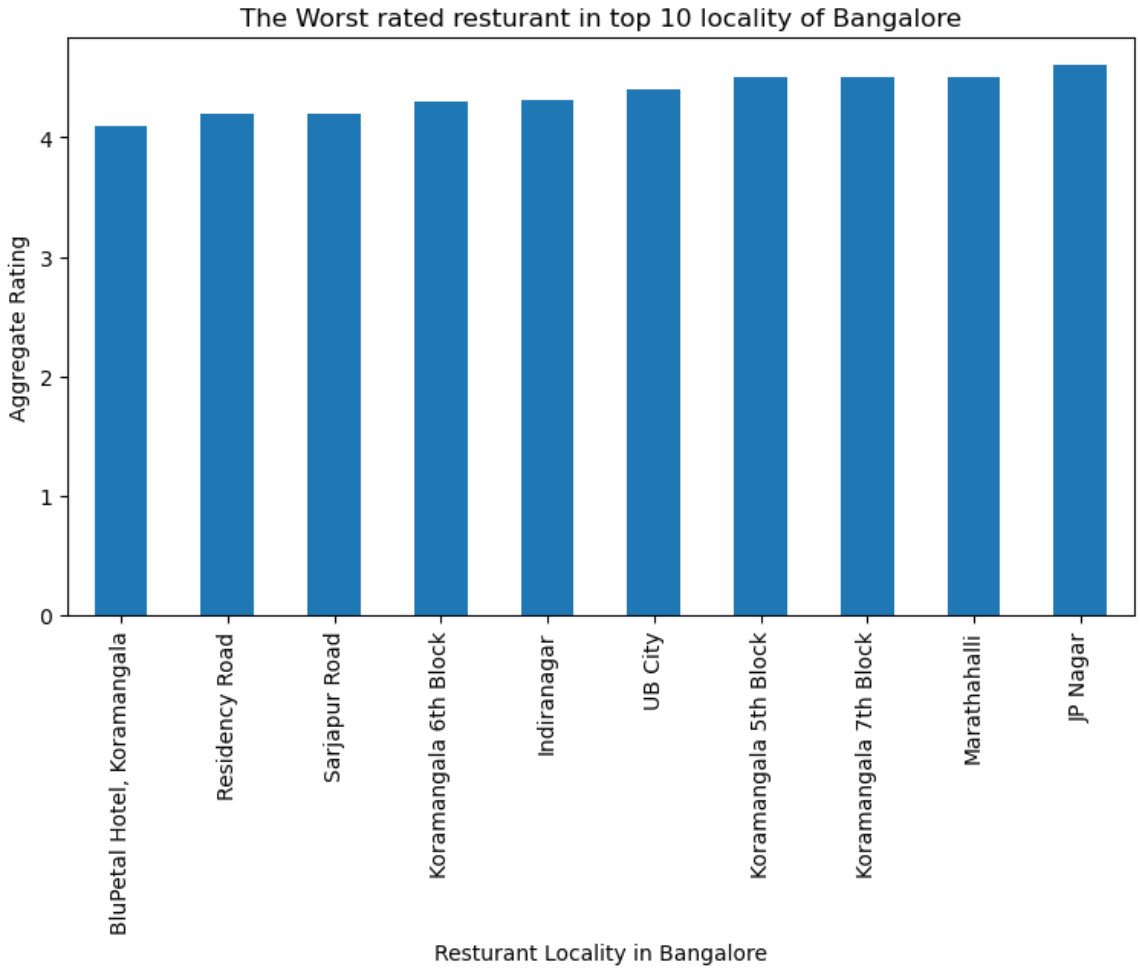
```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('The Worst rated resturant in top 10 locality of Bangalore')
#On x-axis

#giving a bar plot

df_Res.groupby('Locality')['Aggregate rating'].mean().nsmallest(10).plot(kind='bar')

plt.xlabel('Resturant Locality in Bangalore')
#On y-axis
plt.ylabel('Aggregate Rating')

#displays the plot
plt.show()
```



Which place are suitable for edible person in Bangalore?

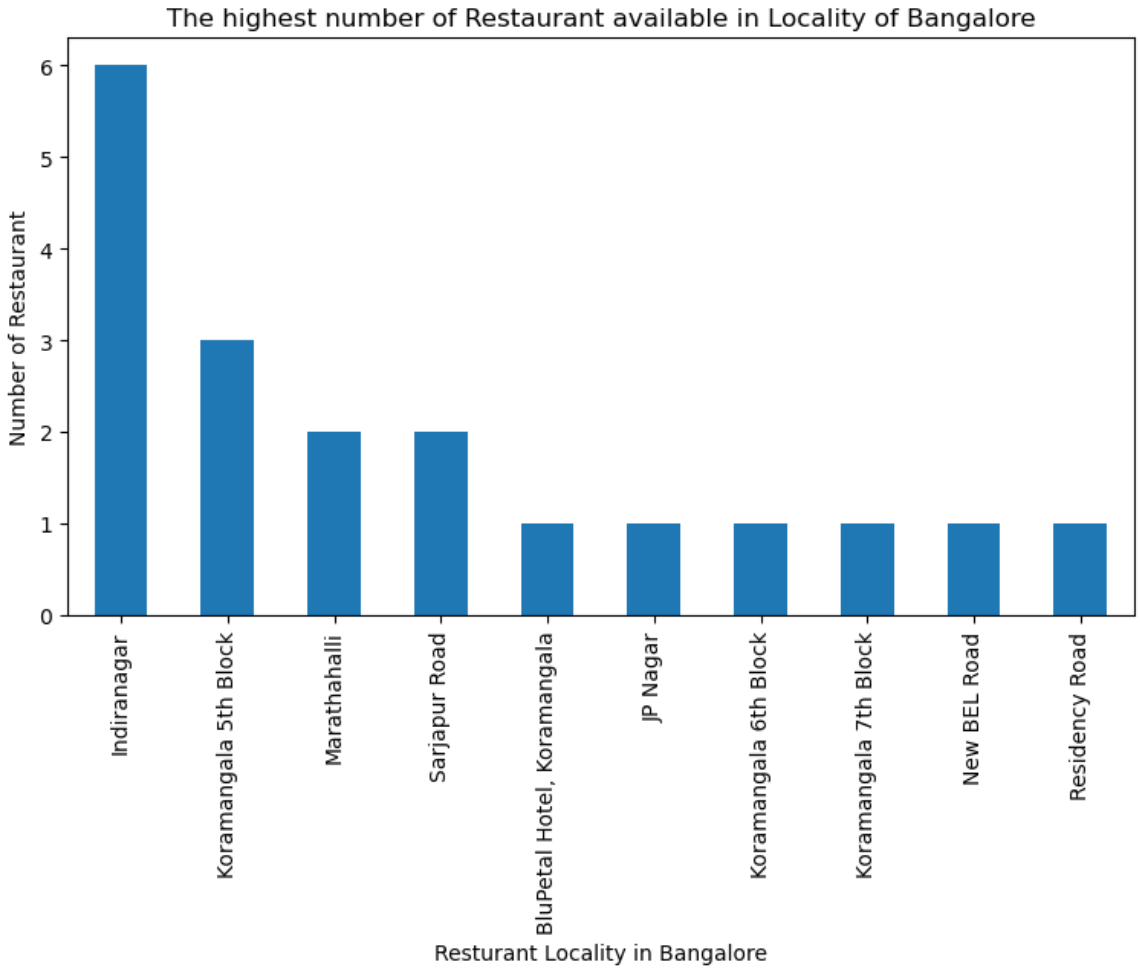
In [20]:

```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('The highest number of Restaurant available in Locality of Bangalore')
#On x-axis

#giving a bar plot
df_Res.groupby('Locality')['Restaurant Name'].count().nlargest(10).plot(kind='bar')

plt.xlabel('Resturant Locality in Bangalore')
#On y-axis
plt.ylabel('Number of Restaurant')

#displays the plot
plt.show()
```



Which place are not suitable for edible person in Bangalore?

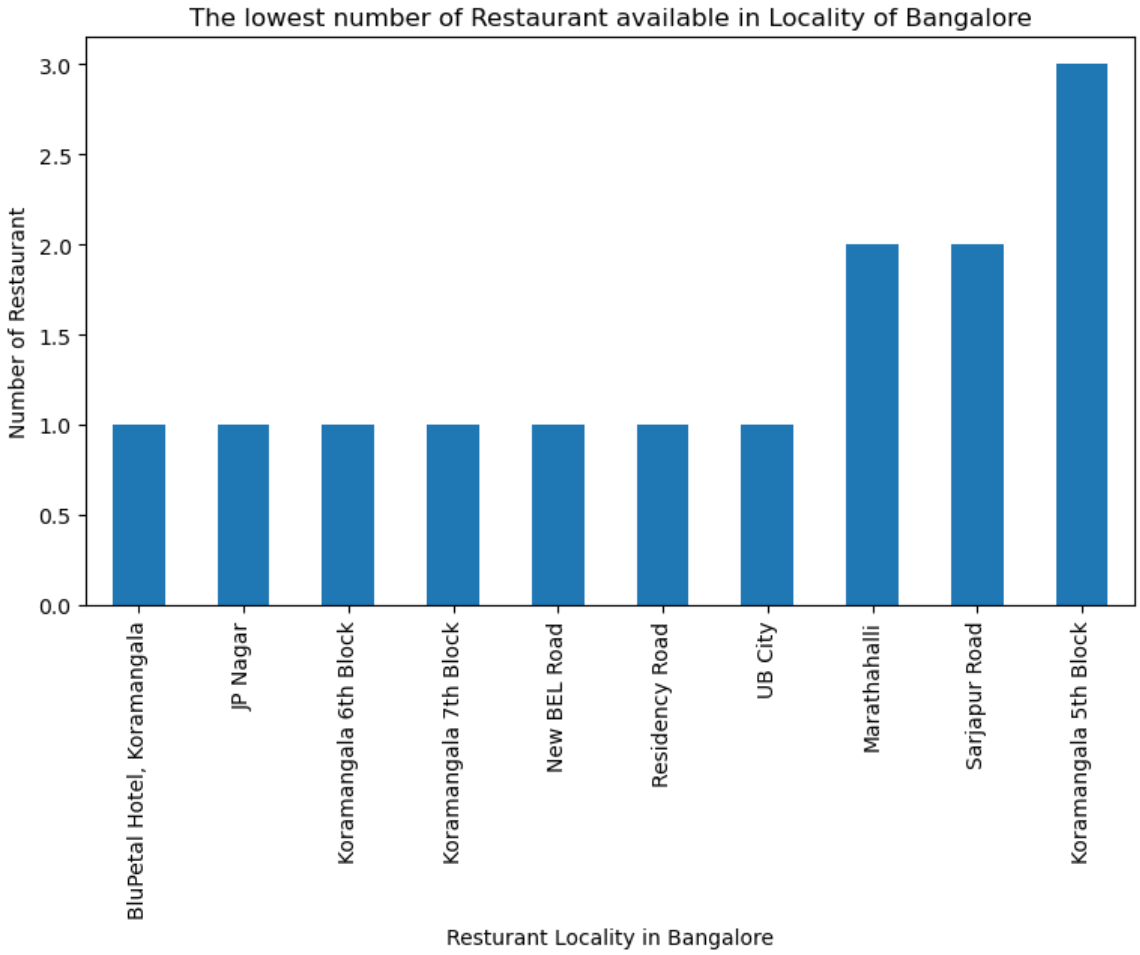
In [22]:

```
import matplotlib.pyplot as plt
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('The lowest number of Restaurant available in Locality of Bangalore')
#On x-axis

#giving a bar plot
df_Res.groupby('Locality')['Restaurant Name'].count().nsmallest(10).plot(kind='bar')

plt.xlabel('Resturant Locality in Bangalore')
#On y-axis
plt.ylabel('Number of Restaurant')

#displays the plot
plt.show()
```



which places are the best Italian resturants in Bangalore?

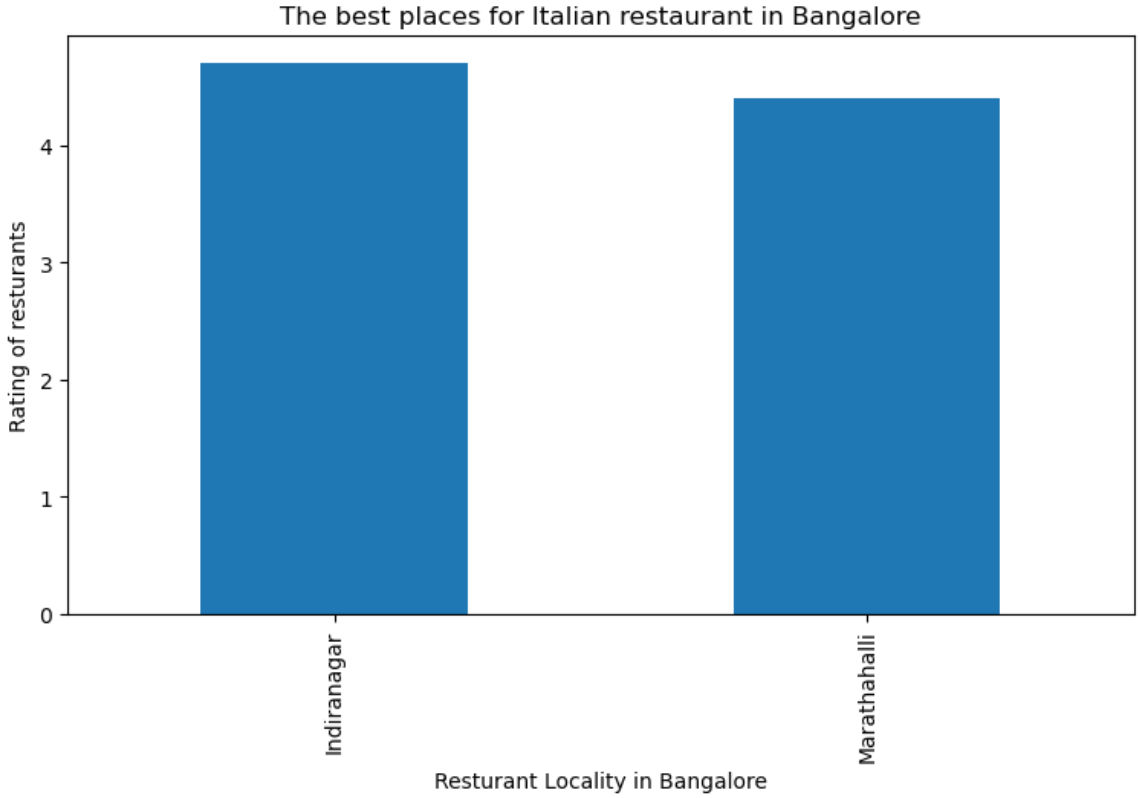
In [34]:

```
plt.figure(figsize=(9,5), dpi = 100)
# title
plt.title('The best places for Italian restaurant in Bangalore')
#On x-axis

#giving a bar plot
df_Res[df_Res['Cuisines'].str.startswith('Asian')].groupby('Locality')['Aggregate ratin
g'].mean().nlargest(5).plot(kind='bar')

plt.xlabel('Resturant Locality in Bangalore')
#On y-axis
plt.ylabel('Rating of resturants')

#displays the plot
plt.show()
```



Data transformation¶

In [35]:

```
df_Res_Loc =  df_Res.groupby('Locality').count()['Restaurant Name'].to_frame()
df_Res_rating= df_Res.groupby('Locality')['Aggregate rating'].mean().to_frame()
d_Cuisines = df_Res.groupby(['Locality'])['Cuisines'].agg(', '.join).reset_index()
d_R = df_Res.groupby(['Locality'])['Rating text'].unique().agg(', '.join).reset_index()
d_V = df_Res.groupby(['Locality'])['Votes'].sum().to_frame()
d_Lat = df_Res.groupby('Locality').mean()['Latitude'].to_frame()
d_Lng = df_Res.groupby('Locality').mean()['Longitude'].to_frame()
df_final = pd.merge(d_Lat,d_Lng,on='Locality').merge(df_Res_Loc, on='Locality').merge(d
_Cuisines, on='Locality').merge(df_Res_rating,on ='Locality').merge(d_R, on ='Locality'
).merge(d_V, on ='Locality')
```

In [36]:

```
df_final = df_final[df_final['Aggregate rating'] != 0.000000]
df_final.columns =['Locality','Lat','Lng', 'No_of_Restaurant','Cusines', 'Agg_Rating',
'Comments' ,'No_of_Votes']
df_final.head()
```

Out[36]:

	Locality	Lat	Lng	No_of_Restaurant	Cusines	Agg_Rating	Comments
0	BluPetal Hotel, Koramangala	12.933284	77.615428	1	North Indian, Mughlai	4.100000	Very Good
1	Indiranagar	12.976278	77.642775	6	Asian, Italian, American, Pizza, Eur...	4.316667	Excellent, Good, Very Good
2	JP Nagar	12.906229	77.596791	1	Pizza, Cafe, Italian	4.600000	Excellent
3	Koramangala 5th Block	12.933947	77.615415	3	Continental, American, Italian, Nort...	4.500000	Excellent, Very Good
4	Koramangala 6th Block	12.939496	77.625999	1	North Indian, Chinese, Italian, Stre...	4.300000	Very Good

In [37]:

```
df_final.shape
```

Out[37]:

```
(11, 8)
```

Define Foursquare Credentials and Version

In [38]:

```
CLIENT_ID = 'VCT1LRCNXEXYMFTSYADSYDGWW2VBBTNVRKQERVWPHS24HXFH' # Foursquare  - Boddu
CLIENT_SECRET = 'GJCFPKHJJKP545PDHNAQLMF0L5QHPEVEUFDWOXBG0T2ECHEW' # Foursquare Secret
VERSION = '20180605' # Foursquare API version

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

```
Your credentails:
CLIENT_ID: VCT1LRCNXEXYMFTSYADSYDGWW2VBBTNVRKQERVWPHS24HXFH
CLIENT_SECRET:GJCFPKHJJKP545PDHNAQLMF0L5QHPEVEUFDWOXBG0T2ECHEW
```

create a function to repeat the same process to all the Locality in Bangalore

In [39]:

```
def getNearbyVenues(names, latitudes, longitudes, radius=500,LIMIT = 100):

    venues_list=[]
    for name, lat, lng in zip(names, latitudes, longitudes):
        print(name)

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret
={}&v={}&ll={},{}&radius={}&limit={}'.format(
            CLIENT_ID,
            CLIENT_SECRET,
            VERSION,
            lat,
            lng,
            radius,
            LIMIT)

        # make the GET request
        results = requests.get(url).json()["response"]['groups'][0]['items']

        # return only relevant information for each nearby venue
        venues_list.append([(
            name,
            lat,
            lng,
            v['venue']['name'],
            v['venue']['location']['lat'],
            v['venue']['location']['lng'],
            v['venue']['categories'][0]['name']) for v in results])

    nearby_venues = pd.DataFrame([item for venue_list in venues_list for item in venue_
list])
    nearby_venues.columns = ['Locality',
                  'Locality Latitude',
                  'Locality Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

In [40]:

```
Bangalore_venues = getNearbyVenues(names=df_final['Locality'],
                                   latitudes=df_final['Lat'],
                                   longitudes=df_final['Lng']
                                  )
```

```
BluPetal Hotel, Koramangala
Indiranagar
JP Nagar
Koramangala 5th Block
Koramangala 6th Block
Koramangala 7th Block
Marathahalli
New BEL Road
Residency Road
Sarjapur Road
UB City
```

In [41]:

```
Bangalore_venues.head()
```

Out[41]:

	Locality	Locality Latitude	Locality Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	BluPetal Hotel, Koramangala	12.933284	77.615428	Truffles Ice & Spice	12.933443	77.614265	Burger Joint
1	BluPetal Hotel, Koramangala	12.933284	77.615428	Gilly's Rest-O-Bar	12.932987	77.614755	Bar
2	BluPetal Hotel, Koramangala	12.933284	77.615428	Stoner	12.932759	77.614132	Ice Cream Shop
3	BluPetal Hotel, Koramangala	12.933284	77.615428	XOOX Brewmill	12.935507	77.614982	Brewery
4	BluPetal Hotel, Koramangala	12.933284	77.615428	Meghana Foods	12.934543	77.616194	Indian Restaurant

In [42]:

```
Bangalore_venues.groupby('Locality').count()
```

Out[42]:

Locality	Locality Latitude	Locality Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
BluPetal Hotel, Koramangala	100	100	100	100	100	100
Indiranagar	60	60	60	60	60	60
JP Nagar	30	30	30	30	30	30
Koramangala 5th Block	100	100	100	100	100	100
Koramangala 6th Block	24	24	24	24	24	24
Koramangala 7th Block	81	81	81	81	81	81
Marathahalli	6	6	6	6	6	6
New BEL Road	34	34	34	34	34	34
Residency Road	100	100	100	100	100	100
Sarjapur Road	15	15	15	15	15	15
UB City	59	59	59	59	59	59

In [43]:

```
print('There are {} uniques categories.'.format(len(Bangalore_venues['Venue Category'].
unique())))
```

```
There are 109 uniques categories.
```

In [44]:

```
## Analyze Each Locality

# one hot encoding
Bangalore_onehot = pd.get_dummies(Bangalore_venues[['Venue Category']], prefix="", pref
ix_sep="")

# add Locality column back to dataframe
Bangalore_onehot['Locality'] = Bangalore_venues['Locality']

# move Locality column to the first column
column_list = Bangalore_onehot.columns.tolist()
column_number = int(column_list.index('Locality'))
column_list = [column_list[column_number]] + column_list[:column_number] + column_list[
column_number+1:]
Bangalore_onehot = Bangalore_onehot[column_list]

Bangalore_onehot.head()
```

Out[44]:

	Locality	Afghan Restaurant	American Restaurant	Andhra Restaurant	Arcade	Art Gallery	Arts & Crafts Store	Asian Restaurant	Bal
0	BluPetal Hotel, Koramangala	0	0	0	0	0	0	0	
1	BluPetal Hotel, Koramangala	0	0	0	0	0	0	0	
2	BluPetal Hotel, Koramangala	0	0	0	0	0	0	0	
3	BluPetal Hotel, Koramangala	0	0	0	0	0	0	0	
4	BluPetal Hotel, Koramangala	0	0	0	0	0	0	0	

5 rows × 110 columns

In [45]:

```
Bangalore_grouped = Bangalore_onehot.groupby('Locality').mean().reset_index()
Bangalore_grouped
```

Out[45]:

	Locality	Afghan Restaurant	American Restaurant	Andhra Restaurant	Arcade	Art Gallery	Arts & Crafts Store	Asia Restaura
0	BluPetal Hotel, Koramangala	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0200(
1	Indiranagar	0.000000	0.00	0.00	0.016667	0.000000	0.000000	0.0000(
2	JP Nagar	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0000(
3	Koramangala 5th Block	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0200(
4	Koramangala 6th Block	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0000(
5	Koramangala 7th Block	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0123
6	Marathahalli	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0000(
7	New BEL Road	0.000000	0.00	0.00	0.000000	0.000000	0.000000	0.0294
8	Residency Road	0.000000	0.01	0.01	0.000000	0.000000	0.010000	0.0000(
9	Sarjapur Road	0.066667	0.00	0.00	0.000000	0.000000	0.000000	0.0000(
10	UB City	0.000000	0.00	0.00	0.000000	0.016949	0.016949	0.0169

11 rows × 110 columns

In [46]:

```
Bangalore_grouped.shape
```

Out[46]:

```
(11, 110)
```

In [47]:

```
## print each Locality along with the top 5 most common venues

num_top_venues = 5

for hood in Bangalore_grouped['Locality']:
    print("----"+hood+"----")
    temp = Bangalore_grouped[Bangalore_grouped['Locality'] == hood].T.reset_index()
    temp.columns = ['venue','freq']
    temp = temp.iloc[1:]
    temp['freq'] = temp['freq'].astype(float)
    temp = temp.round({'freq': 2})
    print(temp.sort_values('freq', ascending=False).reset_index(drop=True).head(num_top
_venues))
    print('\n')
```

```
----BluPetal Hotel, Koramangala----
                 venue  freq
0     Indian Restaurant  0.17
1                 Café  0.06
2  Fast Food Restaurant  0.05
3    Chinese Restaurant  0.05
4          Dessert Shop  0.04


----Indiranagar----
               venue  freq
0               Café  0.15
1  Indian Restaurant  0.13
2                Pub  0.07
3       Dessert Shop  0.05
4        Pizza Place  0.05


----JP Nagar----
         venue  freq
0  Pizza Place  0.07
1       Bakery  0.07
2         Café  0.07
3  Coffee Shop  0.07
4          Bar  0.07


----Koramangala 5th Block----
                 venue  freq
0     Indian Restaurant  0.18
1    Chinese Restaurant  0.05
2                 Café  0.05
3  Fast Food Restaurant  0.04
4          Dessert Shop  0.04


----Koramangala 6th Block----
                   venue  freq
0     Seafood Restaurant  0.08
1         Clothing Store  0.08
2            Pizza Place  0.08
3                    Pub  0.04
4  General Entertainment  0.04


----Koramangala 7th Block----
                venue  freq
0   Indian Restaurant  0.17
1        Dessert Shop  0.05
2  Chinese Restaurant  0.05
3           Bookstore  0.04
4              Lounge  0.04


----Marathahalli----
                 venue  freq
0        Dessert Shop  0.50
1   Indian Restaurant  0.17
2         Pizza Place  0.17
3  Chinese Restaurant  0.17
4                 Pub  0.00
```

```
----New BEL Road----
                 venue  freq
0        Ice Cream Shop  0.21
1     Indian Restaurant  0.15
2    Chinese Restaurant  0.12
3  Fast Food Restaurant  0.12
4           Pizza Place  0.06


----Residency Road----
               venue  freq
0              Café  0.15
1  Indian Restaurant  0.10
2               Pub  0.08
3               Bar  0.06
4            Lounge  0.04


----Sarjapur Road----
               venue  freq
0              Café  0.20
1  Afghan Restaurant  0.07
2           Brewery  0.07
3     Ice Cream Shop  0.07
4       Liquor Store  0.07


----UB City----
                venue  freq
0  Italian Restaurant  0.08
1             Lounge  0.05
2  Mexican Restaurant  0.05
3               Café  0.05
4         Coffee Shop  0.03
```

In [48]:

```
## put that into a pandas dataframe
## First, write a function to sort the venues in descending order.

def return_most_common_venues(row, num_top_venues):
    row_categories = row.iloc[1:]
    row_categories_sorted = row_categories.sort_values(ascending=False)

    return row_categories_sorted.index.values[0:num_top_venues]
```

In [49]:

```
num_top_venues = 10

indicators = ['st', 'nd', 'rd']

# create columns according to number of top venues
columns = ['Locality']
for ind in np.arange(num_top_venues):
    try:
        columns.append('{}{} Most Common Venue'.format(ind+1, indicators[ind]))
    except:
        columns.append('{}th Most Common Venue'.format(ind+1))

# create a new dataframe
Locality_venues_sorted = pd.DataFrame(columns=columns)
Locality_venues_sorted['Locality'] = Bangalore_grouped['Locality']

for ind in np.arange(Bangalore_grouped.shape[0]):
    Locality_venues_sorted.iloc[ind, 1:] = return_most_common_venues(Bangalore_grouped.
iloc[ind, :], num_top_venues)

Locality_venues_sorted
```

Out[49]:

	Locality	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6th Most Common Venue	7th Co
0	BluPetal Hotel, Koramangala	Indian Restaurant	Café	Fast Food Restaurant	Chinese Restaurant	Dessert Shop	Restaurant	L
1	Indiranagar	Café	Indian Restaurant	Pub	Dessert Shop	Pizza Place	Lounge	Bc
2	JP Nagar	Pizza Place	Bakery	Café	Coffee Shop	Bar	Clothing Store	
3	Koramangala 5th Block	Indian Restaurant	Chinese Restaurant	Café	Fast Food Restaurant	Dessert Shop	Restaurant	L
4	Koramangala 6th Block	Seafood Restaurant	Clothing Store	Pizza Place	Pub	General Entertainment	Chinese Restaurant	Rest
5	Koramangala 7th Block	Indian Restaurant	Dessert Shop	Chinese Restaurant	Bookstore	Lounge	Café	
6	Marathahalli	Dessert Shop	Indian Restaurant	Pizza Place	Chinese Restaurant	Pub	Persian Restaurant	Rest
7	New BEL Road	Ice Cream Shop	Indian Restaurant	Chinese Restaurant	Fast Food Restaurant	Pizza Place	Paper / Office Supplies Store	
8	Residency Road	Café	Indian Restaurant	Pub	Bar	Lounge	Coffee Shop	C Rest
9	Sarjapur Road	Café	Afghan Restaurant	Brewery	Ice Cream Shop	Liquor Store	Lounge	E Eur Rest
10	UB City	Italian Restaurant	Lounge	Mexican Restaurant	Café	Coffee Shop	Restaurant	

In [50]:

```
## Cluster Locality
## Run k-means to cluster the Locality into 5 clusters.

# set number of clusters
kclusters = 5

Bangalore_clustering = Bangalore_grouped.drop('Locality', 1)

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(Bangalore_clustering)

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:10]
kmeans.labels_.shape
```

Out[50]:

```
(11,)
```

In [51]:

```
Bangalore_merged = df_final.head(240)
Bangalore_merged['Cluster Labels'] = kmeans.labels_

# merge Bangalore_grouped with df_Chinese to add latitude/longitude for each Locality
Bangalore_merged = Bangalore_merged.join(Locality_venues_sorted.set_index('Locality'),
on='Locality')

Bangalore_merged.head()
```

```
/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/ipykernel/__ma
in__.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  from ipykernel import kernelapp as app
```

Out[51]:

	Locality	Lat	Lng	No_of_Restaurant	Cusines	Agg_Rating	Comments
0	BluPetal Hotel, Koramangala	12.933284	77.615428	1	North Indian, Mughlai	4.100000	Very Good
1	Indiranagar	12.976278	77.642775	6	Asian, Italian, American, Pizza, Eur...	4.316667	Excellent, Good, Very Good
2	JP Nagar	12.906229	77.596791	1	Pizza, Cafe, Italian	4.600000	Excellent
3	Koramangala 5th Block	12.933947	77.615415	3	Continental, American, Italian, Nort...	4.500000	Excellent, Very Good
4	Koramangala 6th Block	12.939496	77.625999	1	North Indian, Chinese, Italian, Stre...	4.300000	Very Good

In [52]:

```
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=10)

# set color scheme for the clusters
x = np.arange(kclusters)
ys = [i+x+(i*x)**2 for i in range(kclusters)]
#colors_array = cm.rainbow(np.linspace(0, 1, len(ys)))
#rainbow = [colors.rgb2hex(i) for i in colors_array]
colors = ['red', 'green', 'blue', 'yellow','orange']

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(Bangalore_merged['Lat'], Bangalore_merged['Lng'], Ban
galore_merged['Locality'], Bangalore_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color='black',
        fill=True,
        fill_color=colors[cluster],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[52]:

Make this Notebook Trusted to load map: File -> Trust Notebook

In [55]:

```
## Examine Clusters

## Cluster 1
Bangalore_merged.loc[Bangalore_merged['Cluster Labels'] == 0, Bangalore_merged.columns
[[1] + list(range(5, Bangalore_merged.shape[1]))]]
```

Out[55]:

	Lat	Agg_Rating	Comments	No_of_Votes	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Mos Commo Venu
2	12.906229	4.6	Excellent	781	0	Pizza Place	Bakery	Caf
4	12.939496	4.3	Very Good	753	0	Seafood Restaurant	Clothing Store	Pizz Plac
10	12.972161	4.4	Very Good	754	0	Italian Restaurant	Lounge	Mexica Restaurar

In [56]:

```
## Examine Clusters

## Cluster 2
Bangalore_merged.loc[Bangalore_merged['Cluster Labels'] == 1, Bangalore_merged.columns
[[1] + list(range(5, Bangalore_merged.shape[1]))]]
```

Out[56]:

	Lat	Agg_Rating	Comments	No_of_Votes	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
6	12.962655	4.5	Excellent, Very Good	7890	1	Dessert Shop	Indian Restaurant	Pizza Place

In [57]:

```
## Examine Clusters

## Cluster 3
Bangalore_merged.loc[Bangalore_merged['Cluster Labels'] == 2, Bangalore_merged.columns
[[1] + list(range(5, Bangalore_merged.shape[1]))]]
```

Out[57]:

	Lat	Agg_Rating	Comments	No_of_Votes	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
1	12.976278	4.316667	Excellent, Good, Very Good	19834	2	Café	Indian Restaurant	Pub
8	12.972532	4.200000	Very Good	334	2	Café	Indian Restaurant	Pub
9	12.913652	4.200000	Excellent, Good	6110	2	Café	Afghan Restaurant	Brewery

In [58]:

```
## Examine Clusters

## Cluster 4
Bangalore_merged.loc[Bangalore_merged['Cluster Labels'] ==3 , Bangalore_merged.columns
[[1] + list(range(5, Bangalore_merged.shape[1]))]]
```

Out[58]:

	Lat	Agg_Rating	Comments	No_of_Votes	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue
7	13.029198	4.6	Excellent	627	3	Ice Cream Shop	Indian Restaurant	Chinese Restaurant

In [59]:

```
## Examine Clusters

## Cluster 5
Bangalore_merged.loc[Bangalore_merged['Cluster Labels'] == 4, Bangalore_merged.columns
[[1] + list(range(5, Bangalore_merged.shape[1]))]]
```

Out[59]:

	Lat	Agg_Rating	Comments	No_of_Votes	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Mos Commo Venu
0	12.933284	4.1	Very Good	2416	4	Indian Restaurant	Café	Fast Foo Restaurar
3	12.933947	4.5	Excellent, Very Good	15328	4	Indian Restaurant	Chinese Restaurant	Caf
5	12.935662	4.5	Excellent	1288	4	Indian Restaurant	Dessert Shop	Chines Restaurar

COnclusion

In [ ]:

```
1. Due to data size limitation, have not considered full set of restaurants in Bangalor
e
2. Based on above findings, one can easily identify goodrestaurances of choice in Banga
lore
```