Capstone Project - The Battle of Neighborhoods (Week 2)

Import Libraries

In [1]:
```
import json, requests
import os
import geopandas
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.cm as cm
import matplotlib.colors as colors
from geopy.geocoders import Nominatim
from pandas.io.json import json_normalize
from sklearn.cluster import KMeans
import folium

pd.options.display.max_rows = None
pd.options.display.max_columns = None
```

We proceed to import the Sandakan neighbourhood csv file which consists the places, names, location, latitude and longtitude.

Sandakan neighbourhood data description:

Number = Index number

Name of neighbourhood = Neighbourhood Names

Area = Area in acres

Residential units = Number of residential homes

Location = Location of neighbourhood

Latitude = Latitude coordinates

Longtitude = Longtitude coordinates

Load data

In [2]:
```
df = pd.read_csv('sandakan.csv', index_col="Number")
```

In [3]:
```
df.head()
```

Out[3]:

Number	Neighbourhood	Area	Residential Units	Location	Latitude	Longtitude
1	Airport	41.630	649	Batu 7, Jalan Lapangan Terbang	5.898035	118.061205
2	Anggerik	15.828	408	Jalan Lintas Sibuga	5.861322	118.037246
3	Astana Height	100.270	483	Batu 1, Jalan Lalang	5.853584	118.116925
4	Berhala Darat	23.200	192	Jalan Sim-Sim	5.850209	118.130763
5	Bukit Permai	270.890	4142	Batu 3 1/2, Jalan Utara	5.864637	118.084975

In [4]:
```
df.tail()
```

Out[4]:

Number	Neighbourhood	Area	Residential Units	Location	Latitude	Longtitude
69	Vista	10.70	172	Batu 7, Jalan Lintas Sibuga	5.858524	118.041216
70	Wijaya	6.03	196	Batu 7, Jalan Labuk	5.884665	118.045531
71	Wira	9.93	312	Jalan Sibuga	5.849188	118.042680
72	Yeng Seng	11.75	125	Batu 2 1/2, Jalan Utara	5.858835	118.098056
73	Yii Villa	1.30	100	Jalan Bulis Sim-Sim	5.854239	118.126795

In [5]:
```
df.shape
```

Out[5]:
```
(73, 6)
```

The dataset consists of 73 rows and 6 columns

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 73 entries, 1 to 73
Data columns (total 6 columns):
Neighbourhood        73 non-null object
Area                 72 non-null float64
Residential Units    73 non-null int64
Location             70 non-null object
Latitude             68 non-null float64
Longtitude           68 non-null float64
dtypes: float64(3), int64(1), object(2)
memory usage: 4.0+ KB
```

In [7]:
```
df.isnull().sum()  #Count NaN values
```

Out[7]:
```
Neighbourhood        0
Area                 1
Residential Units    0
Location             3
Latitude             5
Longtitude           5
dtype: int64
```

Since we need to explore and plot neighbourhoods, I decided to drop NaNs for Latitude and Longtitude

In [8]:
```
df.dropna(inplace=True)
```

In [9]:
```
df.shape
```

Out[9]:
```
(65, 6)
```

In [10]:
```
df.head()
```

Out[10]:

Number	Neighbourhood	Area	Residential Units	Location	Latitude	Longtitude
1	Airport	41.630	649	Batu 7, Jalan Lapangan Terbang	5.898035	118.061205
2	Anggerik	15.828	408	Jalan Lintas Sibuga	5.861322	118.037246
3	Astana Height	100.270	483	Batu 1, Jalan Lalang	5.853584	118.116925
4	Berhala Darat	23.200	192	Jalan Sim-Sim	5.850209	118.130763
5	Bukit Permai	270.890	4142	Batu 3 1/2, Jalan Utara	5.864637	118.084975

In [11]:
```
df.reset_index()
```

Out[11]:

	Number	Neighbourhood	Area	Residential Units	Location	Latitude	Longtitude
0	1	Airport	41.630	649	Batu 7, Jalan Lapangan Terbang	5.898035	118.061205
1	2	Anggerik	15.828	408	Jalan Lintas Sibuga	5.861322	118.037246
2	3	Astana Height	100.270	483	Batu 1, Jalan Lalang	5.853584	118.116925
3	4	Berhala Darat	23.200	192	Jalan Sim-Sim	5.850209	118.130763
4	5	Bukit Permai	270.890	4142	Batu 3 1/2, Jalan Utara	5.864637	118.084975
5	6	Bunga Matahari	11.880	172	Batu 4, Jalan Utara	5.865810	118.075874
6	7	Casa San Uno	38.890	307	Batu 4, Jalan Utara	5.865233	118.072556
7	8	Chrysanthemum	11.400	154	Batu 1 1/2, Jalan Utara	5.857480	118.105876
8	9	Damai & Sri Taman	21.670	123	Batu 4, Jalan Utara	5.858482	118.078921
9	10	Evergreen	23.990	48	Batu 6, Jalan Utara	5.873464	118.057834
10	11	Fajar	55.610	572	Batu 7, Jalan Lapangan Terbang	5.836976	118.098463
11	12	Fajar Perdana	15.030	185	Batu 7, Jalan Lapangan Terbang	5.884339	118.057220
12	13	Fortune	22.400	126	Batu 8, Jalan Labuk	5.885541	118.030970
13	14	Fulliwa	19.660	164	Batu 3 1/2, Jalan Utara	5.862592	118.085900
14	15	Garden Villa	25.760	82	Batu 6, Jalan Utara	5.864005	118.048945
15	16	Grandview	93.000	746	Batu 1 1/2, Jalan Buli Sim-Sim	5.862512	118.119377
16	17	Hap Seng Properties	16.350	74	Jalan Batu Sapi	5.833646	118.092576
17	19	Hing Lee	11.750	227	Batu 3 1/2, Jalan Utara	5.862033	118.090287
18	23	Indah	56.270	356	Batu 4, Jalan Utara	5.842067	118.066095
19	24	Indah Jaya	235.680	2752	Batu 4, Jalan Utara	5.843796	118.067200
20	25	Jade Garden	1154.000	8	Batu 1 1/2, Jalan Utara	5.862170	118.110320
21	26	Kam Jai Yen	8.510	250	Batu 1, Jalan Aman	5.849344	118.110234
22	27	Karamunting Baru	17.670	40	Jalan Karamunting	5.810315	118.072635
23	28	Karamunting Flat	20.870	513	Jalan Batu Sapi	5.813614	118.065127
24	29	Kenari	3.440	590	Batu 7, Jalan Lapangan Terbang	5.895072	118.043659
25	30	Khong Lok (Hillside)	24.809	168	Batu 7, Jalan Lapangan Terbang	5.878357	118.059916
26	31	LPPB	27.210	984	Batu 3 1/2, Jalan Utara	5.862075	118.084777
27	32	Lucky & Wemin	43.544	260	Batu 5, Jalan Utara	5.863112	118.062768
28	33	Mawar	175.386	2396	Jalan Sibuga	5.842216	118.032957
29	34	Megah	44.940	478	Batu 8, Jalan Utara	5.875798	118.042150
30	35	Melanta	9.040	143	Jalan Karamunting	5.810121	118.079529
31	36	Melrose	14.580	44	Batu 3 1/2, Jalan Utara	5.839134	118.115892
32	37	Merpati	154.200	494	Batu 8, Jalan Lapangan Terbang	5.889157	118.042522
33	38	Mesra	23.180	1000	Batu 4, Jalan Utara	5.861271	118.077664
34	40	Mutiara	64.790	836	Batu 3, Jalan Utara	5.854958	118.087570
35	41	Nuri (Oscaraya)	32.650	837	Batu 7, Jalan Lapangan Terbang	5.891094	118.040690
36	42	Pak Tak	29.886	148	Batu 7, Jalan Lapangan Terbang	5.880687	118.056512
37	43	Pecky Valley	36.610	166	Batu 2 1/2, Jalan Utara	5.857141	118.100194

Drop number and location columns from dataframe

In [12]:
```
df = df[['Neighbourhood','Area','Residential Units','Latitude','Longtitude']]
```

In [13]:
```
df.reset_index(drop="Number", inplace=True)
```

In [14]:
```
#save a cleaned csv file for backup

#df.to_csv('skanclean.csv', index=False)
```

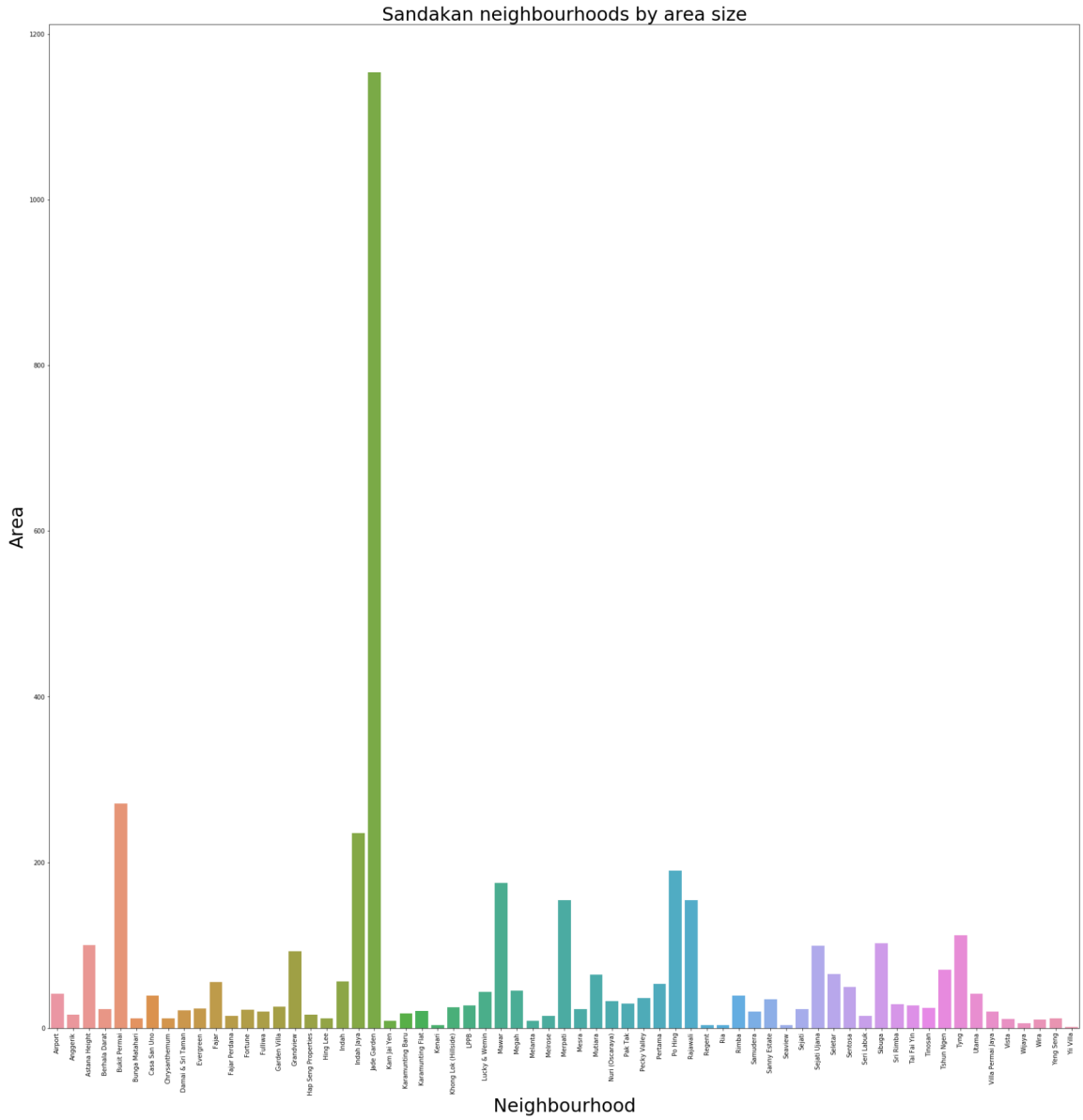
Create visualizations for data exploration

In [15]:
```
df.head()
```

Out[15]:

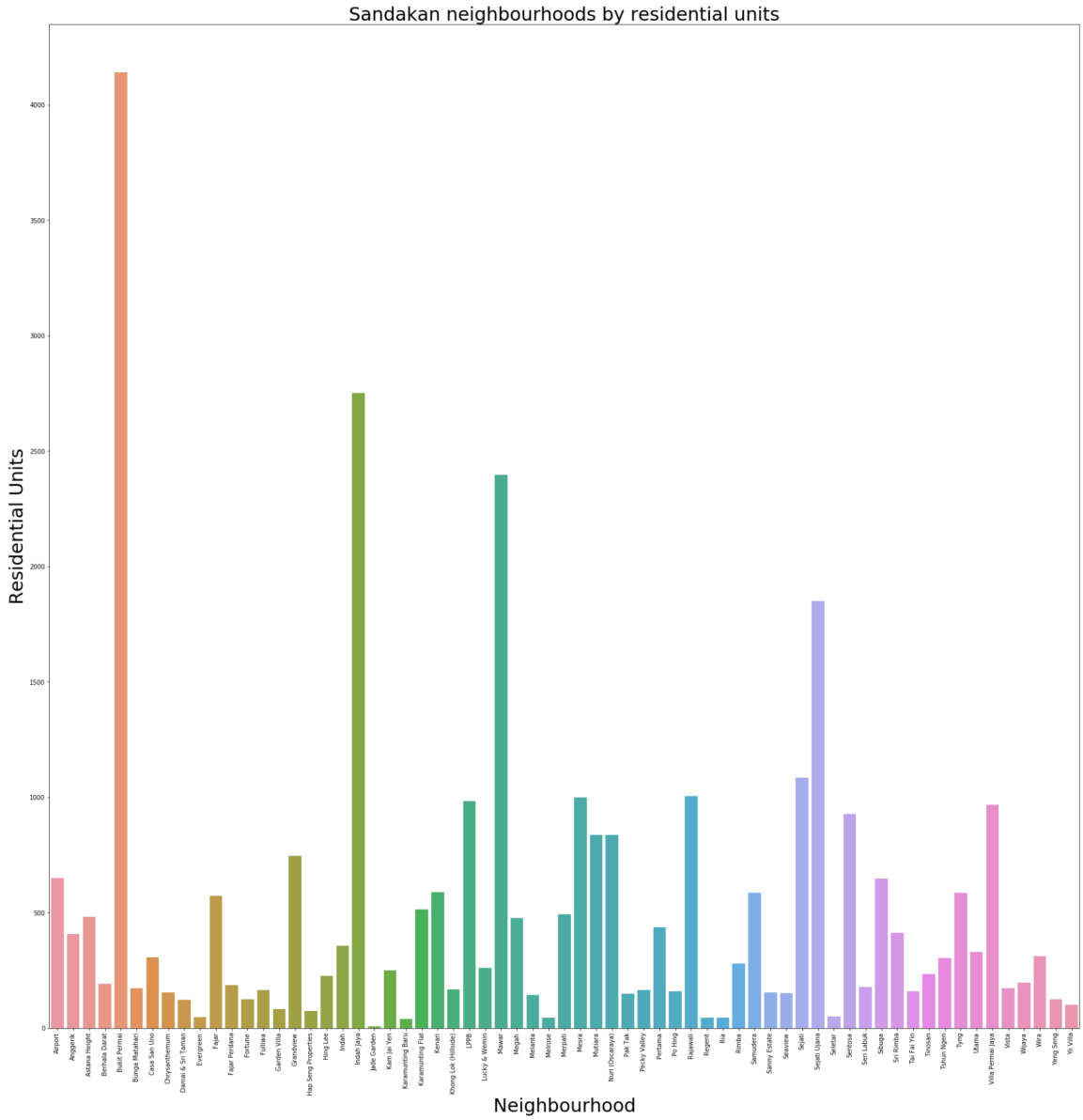
	Neighbourhood	Area	Residential Units	Latitude	Longtitude
0	Airport	41.630	649	5.898035	118.061205
1	Anggerik	15.828	408	5.861322	118.037246
2	Astana Height	100.270	483	5.853584	118.116925
3	Berhala Darat	23.200	192	5.850209	118.130763
4	Bukit Permai	270.890	4142	5.864637	118.084975

In [16]:
```
plt.figure(figsize=(30,30))
plt.title('Sandakan neighbourhoods by area size', fontsize=30)
plt.xlabel('xlabel', fontsize=30)
plt.ylabel('ylabel', fontsize=30)
plt.xticks(rotation='vertical')
sns.barplot(x=df.Neighbourhood,y=df.Area)
plt.show()
```



Observation: Jade Garden has largest area

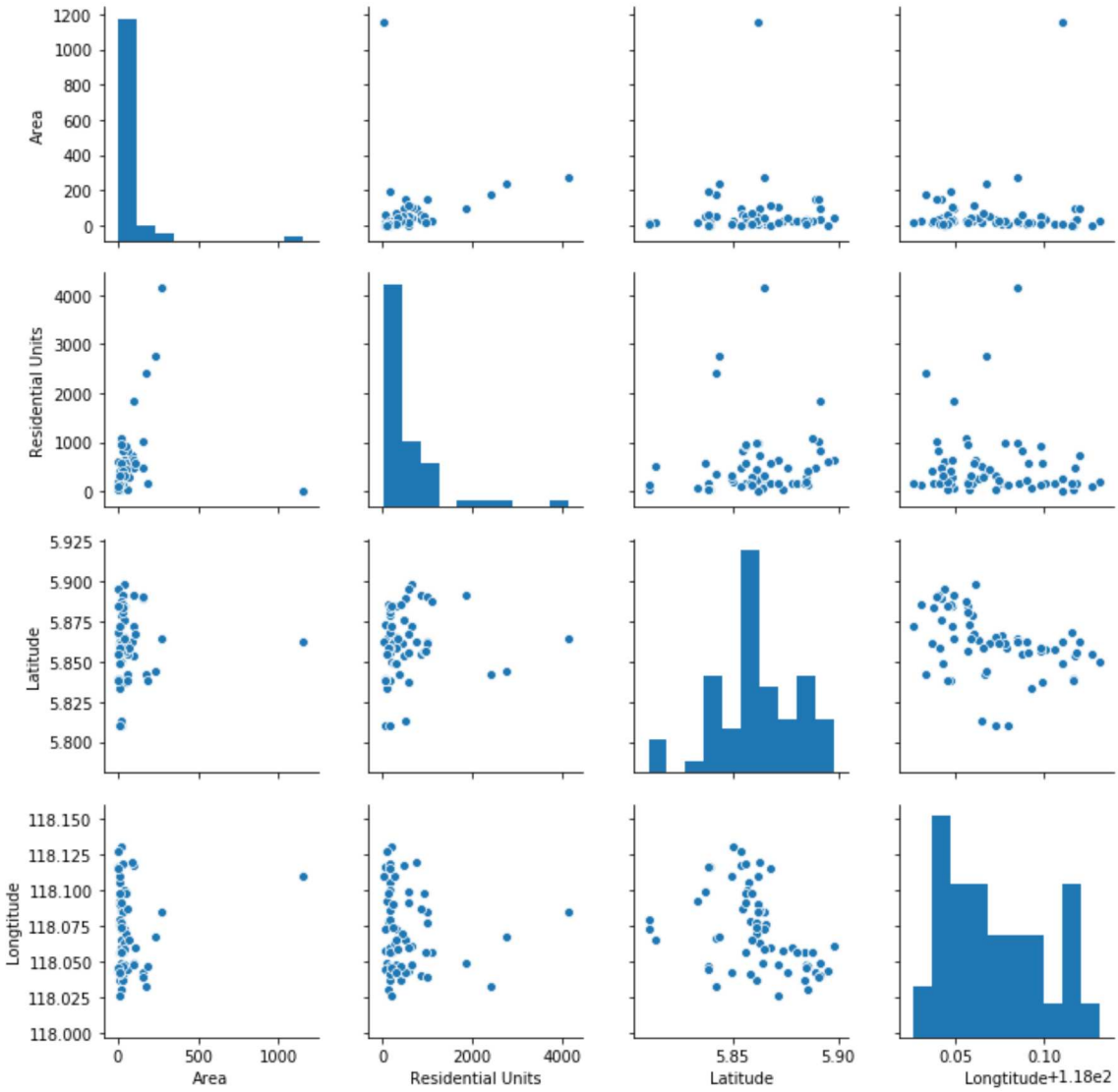
In [17]:
```
plt.figure(figsize=(30,30))
plt.title('Sandakan neighbourhoods by residential units', fontsize=30)
plt.xlabel('xlabel', fontsize=30)
plt.ylabel('ylabel', fontsize=30)
plt.xticks(rotation='vertical')
sns.barplot(x=df.Neighbourhood,y=df['Residential Units'])
plt.show()
```



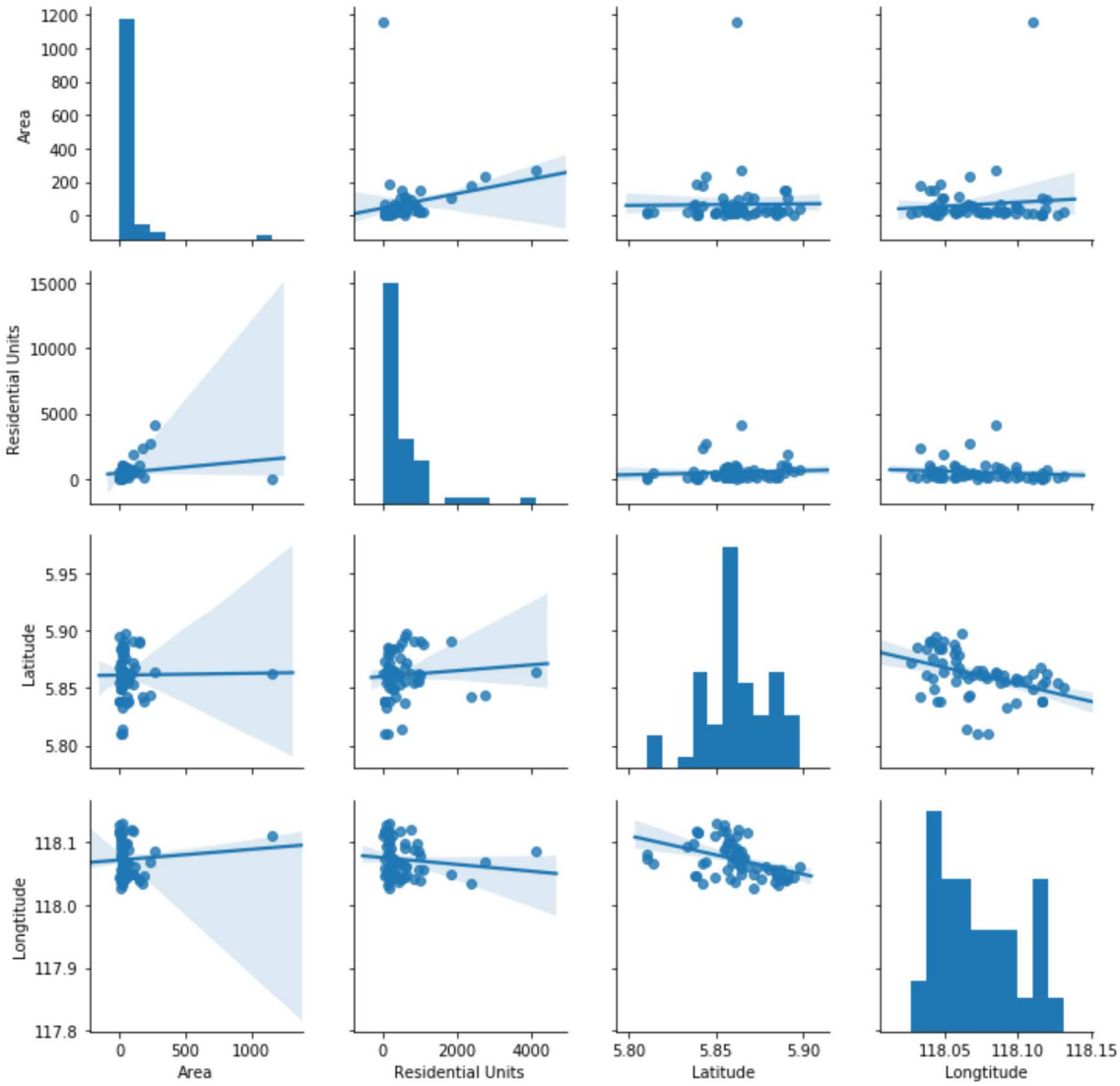
Observation: Bukit Permai has most number of residential units

Plotting pairplots to check for any correlation

In [18]:
```
sns.pairplot(df)
plt.show()
```

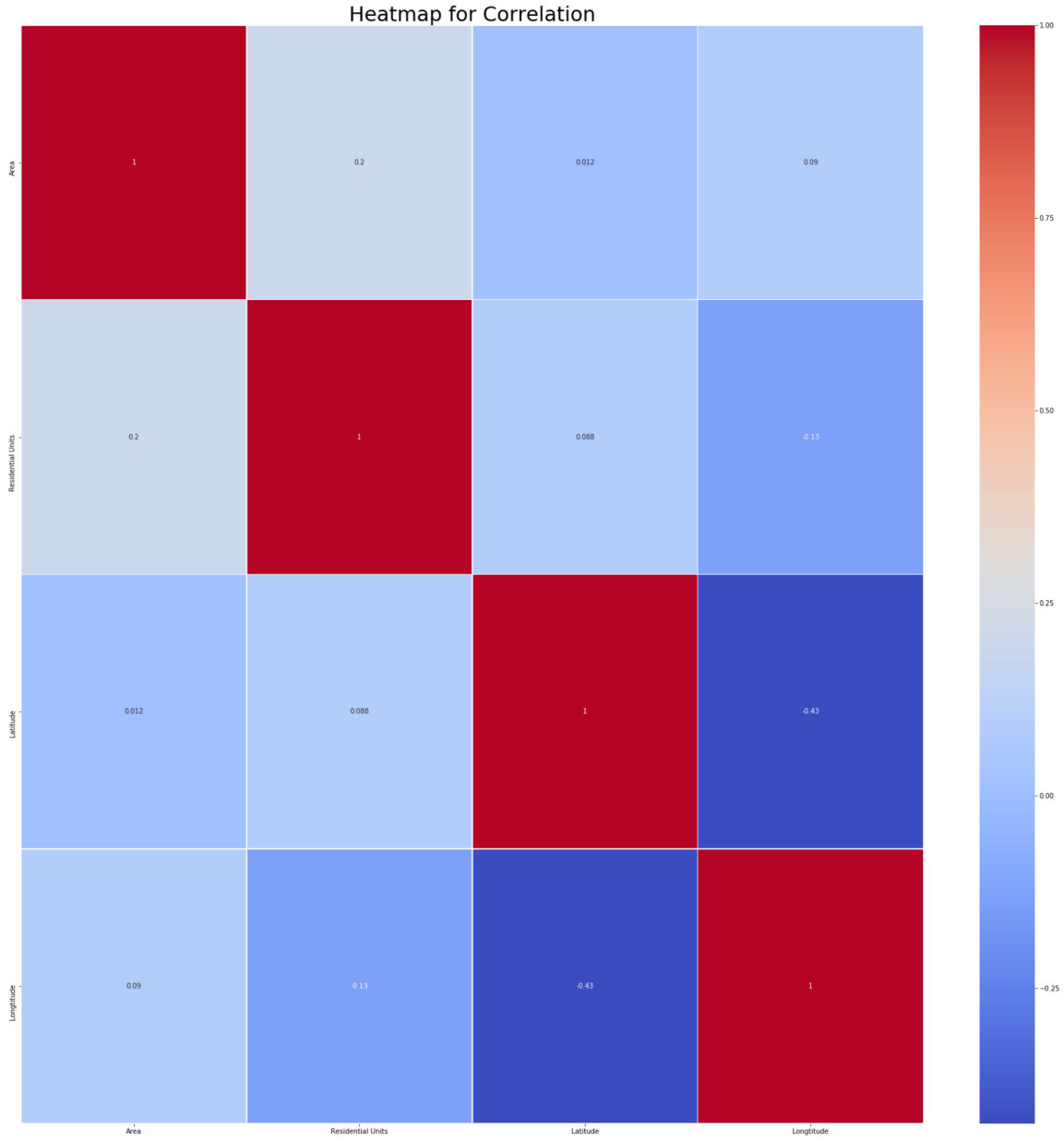


In [19]:
```
sns.pairplot(df, kind='reg')
plt.show()
```



Observation: There seems to be a small linear relationship between Area and Residential Units

In [20]:
```
plt.figure(figsize=(30,30))
plt.title('Heatmap for Correlation', fontsize=30)

sns.heatmap(df.corr(), annot=True, linewidth = 0.5, cmap='coolwarm')
plt.show()
```



Observation: From heatmap diagram only 0.2 correlation coeffient between Area and Residential Units

Create maps to look at all neighbourhoods

In [21]:
```
#Load the cleaned csv file

df = pd.read_csv("skanclean.csv")
```

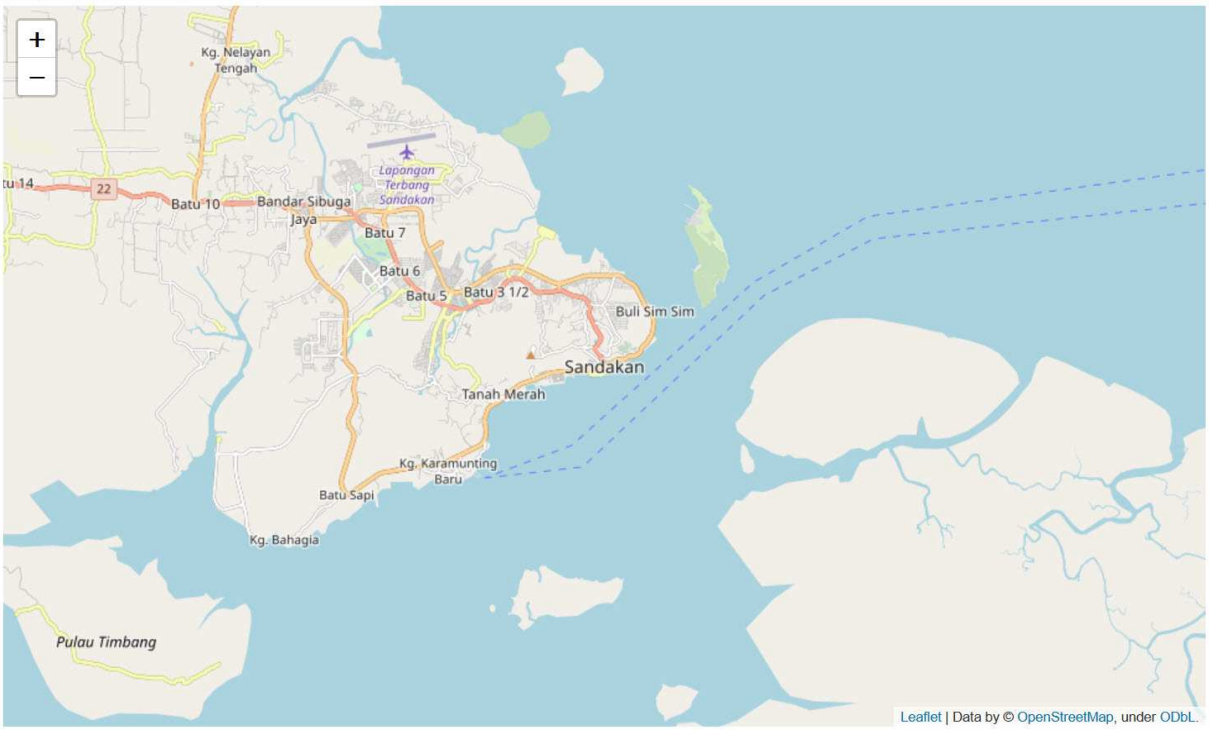
In [22]:
```
#Get the lat and long coordinates for Sandakan
address = 'Sandakan'

geolocator = Nominatim(user_agent="foursquare_agent")

location = geolocator.geocode(address)

latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
5.8391337 118.1158919
```

In [23]:
```
#Sandakan Map
map = folium.Map(location=[latitude,longitude], zoom_start=12)
map
```

Out[23]:

Out[23]:



In [24]:
```
#Segment suburbs coordinates

df_suburbs = df[['Latitude','Longtitude']]
```

In [25]:
```
df_suburbs.head()
```

Out[25]:

	Latitude	Longtitude
0	5.898035	118.061205
1	5.861322	118.037246
2	5.853584	118.116925
3	5.850209	118.130763
4	5.864637	118.084975

In [26]:
```
df_suburbs.shape
```

Out[26]:
```
(65, 2)
```

In [27]:
```
suburbs_list = df_suburbs.values.tolist()
```

In [28]:
```
suburbs_list_size = len(suburbs_list)
```

In [29]:
```
suburbs_list_size
```

Out[29]:
```
65
```

In [30]:
```
#Add Markers
for point in range(0,suburbs_list_size):
    folium.Marker(suburbs_list[point]).add_to(map)
```

In [31]:
```
map
```

Out[31]:

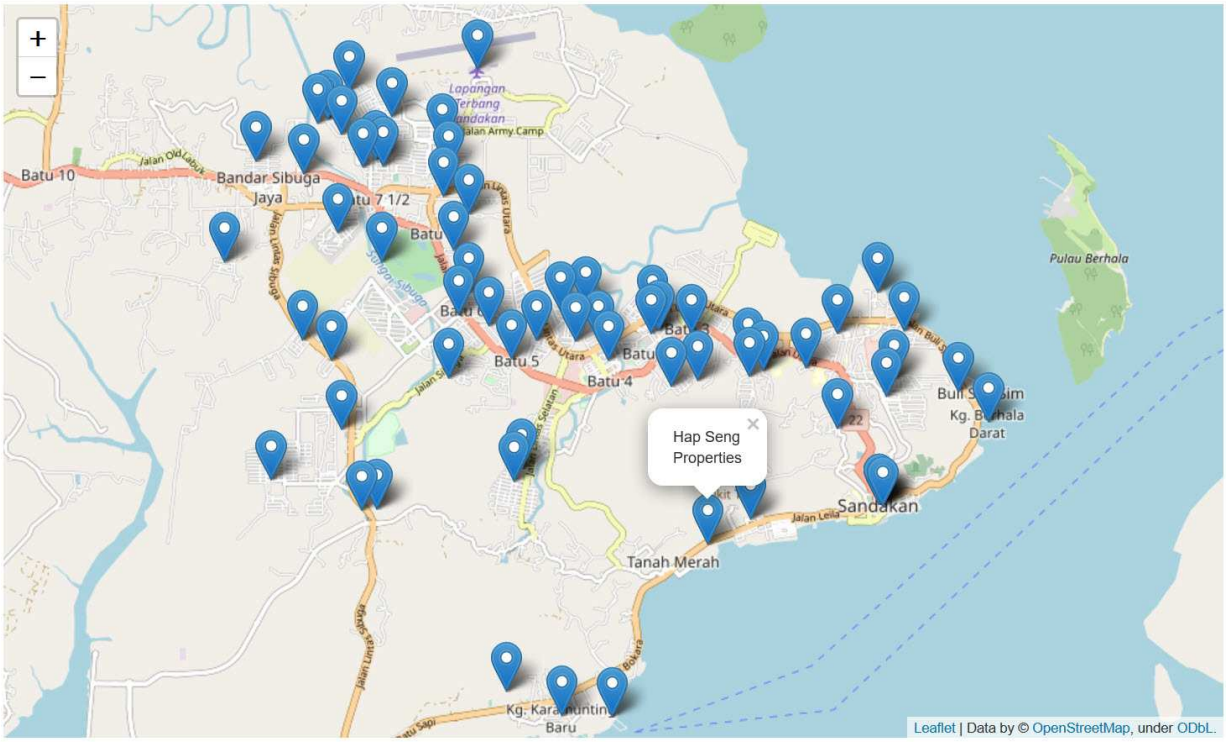
Out[31]:

In [32]:
```
#Add Markers with Popup
for point in range(0,suburbs_list_size):
    folium.Marker(suburbs_list[point], popup=df['Neighbourhood'][point]).add_to
(map)
```

In [33]:
```
map
```

Out[33]:

Out[33]:



Segment and focus Mile 4 to Mile 6 neighbourhoods

In [34]:
```
df1 = pd.read_csv("segment.csv")
```

In [35]:
```
df1
```

Out[35]:

	Neighbourhood	Area	Residential Units	Latitude	Longtitude
0	Bunga Matahari	11.880	172	5.865810	118.075874
1	Casa San Uno	38.890	307	5.865233	118.072556
2	Damai & Sri Taman	21.670	123	5.858482	118.078921
3	Evergreen	23.990	48	5.873464	118.057834
4	Garden Villa	25.760	82	5.863280	118.048945
5	Indah	56.270	356	5.842067	118.066095
6	Indah Jaya	235.680	2752	5.843796	118.067200
7	Lucky & Wemin	43.544	260	5.863112	118.062768
8	Mesra	23.180	1000	5.861271	118.077664
9	Pertama	53.320	438	5.861339	118.069276
10	Tinosan	24.510	235	5.861032	118.074517
11	Tshun Ngen	70.260	304	5.858728	118.065804
12	Tyng	111.700	585	5.867669	118.059997
13	Utama	41.263	329	5.864601	118.058569

In [36]:
```
df1.shape
```

Out[36]:
```
(14, 5)
```

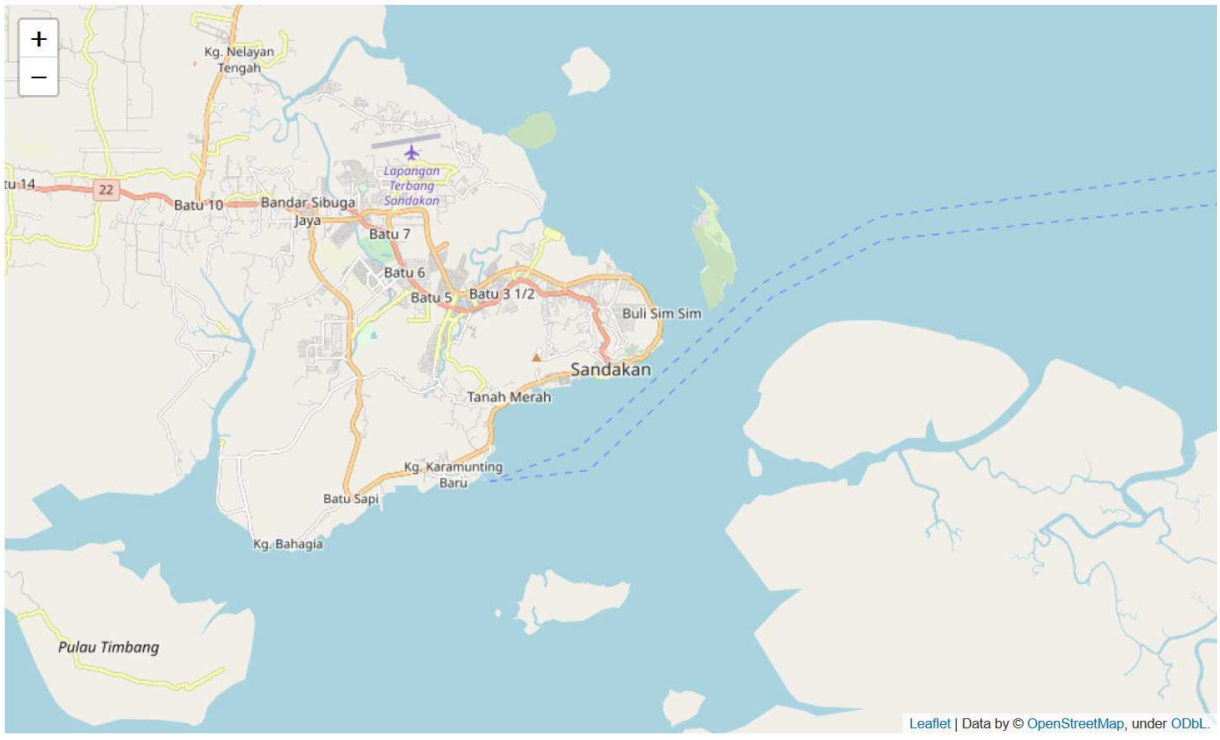
In [37]:
```
address = 'Sandakan'

geolocator = Nominatim(user_agent="foursquare_agent")

location = geolocator.geocode(address)

latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
5.8391337 118.1158919
```

In [38]:
```
#Sandakan Map
map1 = folium.Map(location=[latitude,longitude], zoom_start=12)
map1
```

Out[38]:

Out[38]:



In [39]:
```
#Segment Mile 4 to Mile 6 suburbs coordinates

df1_suburbs = df1[['Latitude','Longtitude']]
```

In [40]:
```
df1_suburbs
```

Out[40]:

	Latitude	Longtitude
0	5.865810	118.075874
1	5.865233	118.072556
2	5.858482	118.078921
3	5.873464	118.057834
4	5.863280	118.048945
5	5.842067	118.066095
6	5.843796	118.067200
7	5.863112	118.062768
8	5.861271	118.077664
9	5.861339	118.069276
10	5.861032	118.074517
11	5.858728	118.065804
12	5.867669	118.059997
13	5.864601	118.058569

In [41]:
```
df1_suburbs.shape
```

Out[41]:
```
(14, 2)
```

In [42]:
```
suburbs1_list = df1_suburbs.values.tolist()
```

In [43]:
```
suburbs1_list
```

Out[43]:
```
[[5.8658095, 118.0758739],
 [5.8652331, 118.0725562],
 [5.8584821, 118.0789209],
 [5.8734638, 118.0578338],
 [5.8632801, 118.0489446],
 [5.8420670999999995, 118.066094999999999],
 [5.843796, 118.06720049999998],
 [5.8631125, 118.0627675],
 [5.8612709, 118.07766389999999],
 [5.8613392, 118.06927649999999],
 [5.8610315, 118.0745175],
 [5.85872755, 118.0658041],
 [5.8676695, 118.0599967],
 [5.864600599999999, 118.058569099999999]]
```

In [44]:
```
suburbs1_list_size = len(suburbs1_list)
```

In [45]:
```
suburbs1_list_size
```

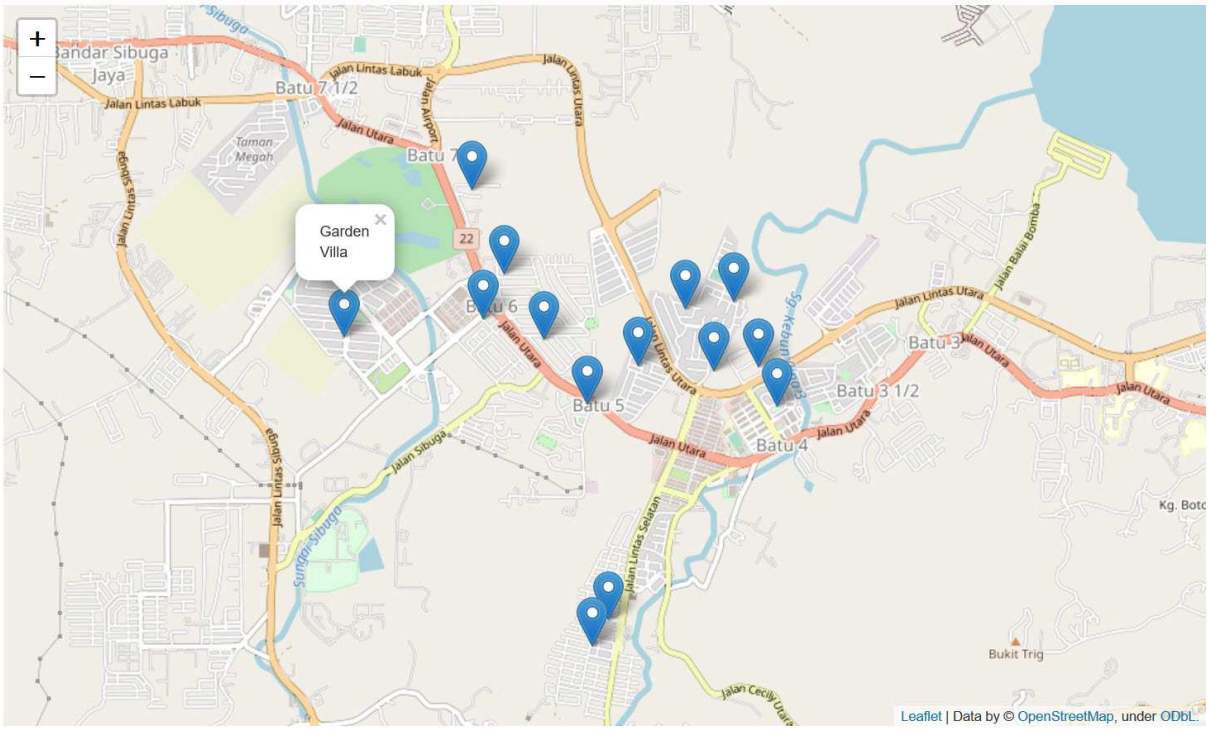
Out[45]:
```
14
```

In [46]:
```
#Add Markers with Popup
for point in range(0,suburbs1_list_size):
    folium.Marker(suburbs1_list[point], popup=df1['Neighbourhood'][point]).add_t
o(map1)
```

In [47]:
```
map1
```

Out[47]:

Out[47]:



Using Foursquare API

Explore Neighborhoods with that focused segment

In [48]:
```
#define our Foursquare credentials and version
CLIENT_ID = 'ZA1DQF403ZFDBZRXJPTGZTZOCFLEFLEKGN0HCDSEZEP4E4WH' # your Foursquare
ID
CLIENT_SECRET = '30UY4KEFYWPITP32JWZIRM1I1NPC42EQ5FVEG2LJV5PISLHY' # your Foursq
uare Secret
VERSION = '20180604'
LIMIT = 15
```

In [49]:
```
neighborhoods_subset = df1[['Neighbourhood','Latitude','Longtitude']]
```

In [50]:
```
neighborhoods_subset
```

Out[50]:

	Neighbourhood	Latitude	Longtitude
0	Bunga Matahari	5.865810	118.075874
1	Casa San Uno	5.865233	118.072556
2	Damai & Sri Taman	5.858482	118.078921
3	Evergreen	5.873464	118.057834
4	Garden Villa	5.863280	118.048945
5	Indah	5.842067	118.066095
6	Indah Jaya	5.843796	118.067200
7	Lucky & Wemin	5.863112	118.062768
8	Mesra	5.861271	118.077664
9	Pertama	5.861339	118.069276
10	Tinosan	5.861032	118.074517
11	Tshun Ngen	5.858728	118.065804
12	Tyng	5.867669	118.059997
13	Utama	5.864601	118.058569

In [51]:
```
def getNearbyVenues(names, latitudes, longitudes, radius=500):

    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 = ['Neighborhood',
                  'Neighborhood Latitude',
                  'Neighborhood Longitude',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']

    return(nearby_venues)
```

In [52]:
```
target_venues = getNearbyVenues(names=neighborhoods_subset['Neighbourhood'],
                                 latitudes=neighborhoods_subset['Latitude'],
                                 longitudes=neighborhoods_subset['Longtitude']
                                )
```

```
Bunga Matahari
Casa San Uno
Damai & Sri Taman
Evergreen
Garden Villa
Indah
Indah Jaya
Lucky & Wemin
Mesra
Pertama
Tinosan
Tshun Ngen
Tyng
Utama
```

In [53]:
```
print(target_venues.shape)
target_venues
```

(129, 7)

Out[53]:

	Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Bunga Matahari	5.865810	118.075874	Sunflower Mini Market Taman Bunga Matahari	5.865544	118.076034	Convenience Store
1	Bunga Matahari	5.865810	118.075874	Starwood Hotels & Resorts	5.866185	118.075382	Hotel
2	Bunga Matahari	5.865810	118.075874	Lee Yuan Chinese Restaurant	5.864253	118.073965	Chinese Restaurant
3	Bunga Matahari	5.865810	118.075874	Taman Tinosan	5.862603	118.073679	Other Great Outdoors
4	Bunga Matahari	5.865810	118.075874	WHITE HORSE CERAMIC, Bdr. Kim Fung, Sandakan, ...	5.870098	118.075628	Furniture / Home Store
5	Bunga Matahari	5.865810	118.075874	Lubuk	5.870108	118.075571	Outdoors & Recreation
6	Damai & Sri Taman	5.858482	118.078921	Servay Supermarket	5.858409	118.078295	Grocery Store
7	Damai & Sri Taman	5.858482	118.078921	Bandar Kim Fung 金凤市	5.856492	118.078332	Town
8	Damai & Sri Taman	5.858482	118.078921	双日Kopitiam茶餐室	5.856717	118.077535	Deli / Bodega
9	Damai & Sri Taman	5.858482	118.078921	Novelty Cafe & Cake House	5.857198	118.079499	Bakery
10	Damai & Sri Taman	5.858482	118.078921	Livingstone Hotel	5.857344	118.081836	Hotel
11	Damai & Sri Taman	5.858482	118.078921	Faces Nasi Kuning Ayam	5.858331	118.080795	Wings Joint
12	Damai & Sri Taman	5.858482	118.078921	Tien Kee Restaurant 田記港式燒腊	5.856493	118.079025	BBQ Joint
13	Damai & Sri Taman	5.858482	118.078921	Kim Fung Market	5.857622	118.079159	Food Court
14	Damai & Sri Taman	5.858482	118.078921	Digital Wise Sdn Bhd	5.857668	118.080302	Electronics Store
15	Damai & Sri Taman	5.858482	118.078921	Kedai Makan Syn Nam Choon Restaurant (2) 新南村雞魚...	5.857652	118.080090	Asian Restaurant
16	Damai & Sri Taman	5.858482	118.078921	7 Eleven	5.857464	118.079115	Convenience Store
17	Damai & Sri Taman	5.858482	118.078921	三點三茶餐厅 Kedai Kopi Kong Fei	5.857155	118.080185	Deli / Bodega
18	Damai & Sri Taman	5.858482	118.078921	2020 Restaurant	5.856228	118.077870	Chinese Restaurant
19	Damai & Sri Taman	5.858482	118.078921	KFC 肯德基	5.857699	118.078737	Fast Food Restaurant
20	Damai & Sri Taman	5.858482	118.078921	Pizza Hut	5.855991	118.078856	Pizza Place
21	Evergreen	5.873464	118.057834	Sandakan Golf & Country Club	5.872148	118.054561	Golf Course
22	Evergreen	5.873464	118.057834	SGCC Gym & Fitness Centre	5.872099	118.054609	Gym
23	Evergreen	5.873464	118.057834	Sandakan Golf And Country Club	5.877885	118.058050	Golf Course
				New Ocean			Seafood

In [54]:
```
#save a copy of csv
#target_venues.to_csv("foursq.csv",index=False)
```

In [55]:
```
target_venues.groupby('Neighborhood').count()
```

Out[55]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Bunga Matahari	6	6	6	6	6	6
Damai & Sri Taman	15	15	15	15	15	15
Evergreen	3	3	3	3	3	3
Garden Villa	15	15	15	15	15	15
Indah	10	10	10	10	10	10
Indah Jaya	7	7	7	7	7	7
Lucky & Wemin	3	3	3	3	3	3
Mesra	15	15	15	15	15	15
Pertama	4	4	4	4	4	4
Tinosan	13	13	13	13	13	13
Tshun Ngen	8	8	8	8	8	8
Tyng	15	15	15	15	15	15
Utama	15	15	15	15	15	15

In [56]:
```
print('There are {} uniques categories.'.format(len(target_venues['Venue Category'].unique())))
```

```
There are 53 uniques categories.
```

Analyze Each Neighborhood

In [57]:
```
# one hot encoding
target_onehot = pd.get_dummies(target_venues[['Venue Category']], prefix="", pre
fix_sep="")

# add neighborhood column back to dataframe
target_onehot['Neighborhood'] = target_venues['Neighborhood']

# move neighborhood column to the first column
fixed_columns = [target_onehot.columns[-1]] + list(target_onehot.columns[:-1])
target_onehot = target_onehot[fixed_columns]

target_onehot.head()
```

Out[57]:

	Neighborhood	American Restaurant	Asian Restaurant	Athletics & Sports	BBQ Joint	Bakery	Baseball Stadium	Bed & Breakfast	Beer Garden	Butcher
0	Bunga Matahari	0	0	0	0	0	0	0	0	0
1	Bunga Matahari	0	0	0	0	0	0	0	0	0
2	Bunga Matahari	0	0	0	0	0	0	0	0	0
3	Bunga Matahari	0	0	0	0	0	0	0	0	0
4	Bunga Matahari	0	0	0	0	0	0	0	0	0

In [58]:
```
target_onehot.shape
```

Out[58]:
```
(129, 54)
```

In [59]:
```
target_grouped = target_onehot.groupby('Neighborhood').mean().reset_index()
target_grouped
```

Out[59]:

	Neighborhood	American Restaurant	Asian Restaurant	Athletics & Sports	BBQ Joint	Bakery	Baseball Stadium	Bed & Breakfast	Beer Garden	B
0	Bunga Matahari	0.00	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
1	Damai & Sri Taman	0.00	0.066667	0.0	0.066667	0.066667	0.0	0.000000	0.000000	0.
2	Evergreen	0.00	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
3	Garden Villa	0.00	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.066667	0.
4	Indah	0.00	0.100000	0.2	0.000000	0.000000	0.1	0.000000	0.000000	0.
5	Indah Jaya	0.00	0.142857	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
6	Lucky & Wemin	0.00	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
7	Mesra	0.00	0.066667	0.0	0.000000	0.066667	0.0	0.000000	0.000000	0.
8	Pertama	0.25	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
9	Tinosan	0.00	0.000000	0.0	0.000000	0.153846	0.0	0.076923	0.000000	0.
10	Tshun Ngen	0.00	0.125000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.
11	Tyng	0.00	0.266667	0.0	0.000000	0.133333	0.0	0.000000	0.000000	0.
12	Utama	0.00	0.000000	0.0	0.000000	0.000000	0.0	0.000000	0.000000	0.

In [60]:
```
target_grouped.shape
```

Out[60]:
```
(13, 54)
```

In [61]:
```
# Print out top 5 venues for each neighbourhood
num_top_venues = 5

for hood in target_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = target_grouped[target_grouped['Neighborhood'] == 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')
```

----Bunga Matahari----

	venue	freq
0	Furniture / Home Store	0.17
1	Chinese Restaurant	0.17
2	Hotel	0.17
3	Other Great Outdoors	0.17
4	Outdoors & Recreation	0.17

----Damai & Sri Taman----

	venue	freq
0	Deli / Bodega	0.13
1	Chinese Restaurant	0.07
2	Grocery Store	0.07
3	Asian Restaurant	0.07
4	Food Court	0.07

----Evergreen----

	venue	freq
0	Golf Course	0.67
1	Gym	0.33
2	American Restaurant	0.00
3	Hakka Restaurant	0.00
4	Halal Restaurant	0.00

----Garden Villa----

	venue	freq
0	Café	0.33
1	Chinese Restaurant	0.20
2	Hakka Restaurant	0.07
3	Halal Restaurant	0.07
4	Football Stadium	0.07

----Indah----

	venue	freq
0	Athletics & Sports	0.2
1	Recreation Center	0.1
2	Food Truck	0.1
3	Department Store	0.1
4	Noodle House	0.1

----Indah Jaya----

	venue	freq
0	Grocery Store	0.14
1	Department Store	0.14
2	Asian Restaurant	0.14
3	Food Truck	0.14
4	Recreation Center	0.14

----Lucky & Wemin----

	venue	freq
0	Park	0.33
1	Chinese Restaurant	0.33
2	Clothing Store	0.33
3	Other Great Outdoors	0.00
4	Hakka Restaurant	0.00

----Mesra----

	venue	freq
0	Convenience Store	0.13
1	Electronics Store	0.13
2	Wings Joint	0.07
3	Grocery Store	0.07

In [62]:
```
#Create a function to return common venues
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 [63]:
```
#Sort each neighbourhood with top 10 venues
num_top_venues = 10

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

# create columns according to number of top venues
columns = ['Neighborhood']
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
neighborhoods_venues_sorted = pd.DataFrame(columns=columns)
neighborhoods_venues_sorted['Neighborhood'] = target_grouped['Neighborhood']

for ind in np.arange(target_grouped.shape[0]):
    neighborhoods_venues_sorted.iloc[ind, 1:] = return_most_common_venues(target
_grouped.iloc[ind, :], num_top_venues)

neighborhoods_venues_sorted
```

Out[63]:

	Neighborhood	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 Most Common Venue	
0	Bunga Matahari	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	Other Great Outdoors	Chinese Restaurant	Food Court	Fis
1	Damai & Sri Taman	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Grocery Store	Electronics Store	Hotel	Co
2	Evergreen	Golf Course	Gym	Wings Joint	Convenience Store	Food Court	Fish & Chips Shop	Fast Food Restaurant	E
3	Garden Villa	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	Indian Restaurant	Halal Restaurant	Beer Garden	Ph
4	Indah	Athletics & Sports	Noodle House	Baseball Stadium	Dim Sum Restaurant	Department Store	Coffee Shop	Food Truck	F
5	Indah Jaya	Food Truck	Asian Restaurant	Grocery Store	Recreation Center	Department Store	Chinese Restaurant	Coffee Shop	
6	Lucky & Wemin	Clothing Store	Park	Chinese Restaurant	Convenience Store	Food Truck	Food Court	Fish & Chips Shop	F
7	Mesra	Electronics Store	Convenience Store	Wings Joint	Other Great Outdoors	Fruit & Vegetable Store	Fast Food Restaurant	Grocery Store	D
8	Pertama	Chinese Restaurant	American Restaurant	Noodle House	Athletics & Sports	BBQ Joint	Food Truck	Food Court	Fis
9	Tinosan	Bakery	Music Venue	Bed & Breakfast	Malay Restaurant	Chinese Restaurant	Noodle House	Café	F
10	Tshun Ngen	Fruit & Vegetable Store	Asian Restaurant	Vegetarian / Vegan Restaurant	Grocery Store	Fish & Chips Shop	Chinese Restaurant	Noodle House	F
11	Tyng	Asian Restaurant	Bakery	Noodle House	Chinese Restaurant	Market	Café	Butcher	
12	Utama	Chinese Restaurant	Malay Restaurant	Fast Food Restaurant	Sushi Restaurant	Restaurant	Lounge	Cupcake Shop	

Clustering Neighborhoods - Using K-means method

In [64]:
```
# set number of clusters
kclusters = 5

target_grouped_clustering = target_grouped.drop('Neighborhood', 1)

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

# check cluster labels generated for each row in the dataframe
kmeans.labels_[0:15]
```

Out[64]:
```
array([0, 0, 3, 4, 0, 0, 2, 0, 1, 0, 0, 0, 0])
```

In [65]:
```
# add clustering labels
neighborhoods_venues_sorted.insert(0, 'Cluster Labels', kmeans.labels_)

target_merged = neighborhoods_subset

# merge toronto_grouped with toronto_data to add latitude/longitude for each nei
ghborhood
target_merged = target_merged.join(neighborhoods_venues_sorted.set_index('Neighb
orhood'), on='Neighbourhood')

target_merged.head() # check the last columns!
```

Out[65]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Mo Commo Venu
0	Bunga Matahari	5.865810	118.075874	0.0	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	Oth Gre Outdoo
1	Casa San Uno	5.865233	118.072556	NaN	NaN	NaN	NaN	NaN	Na
2	Damai & Sri Taman	5.858482	118.078921	0.0	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Groce Sto
3	Evergreen	5.873464	118.057834	3.0	Golf Course	Gym	Wings Joint	Convenience Store	Foo Cou
4	Garden Villa	5.863280	118.048945	4.0	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	Indi Restaura

In [66]:
```
target_merged #Do a check for all
```

Out[66]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Com V
0	Bunga Matahari	5.865810	118.075874	0.0	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	((Outd
1	Casa San Uno	5.865233	118.072556	NaN	NaN	NaN	NaN	NaN	
2	Damai & Sri Taman	5.858482	118.078921	0.0	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Gr
3	Evergreen	5.873464	118.057834	3.0	Golf Course	Gym	Wings Joint	Convenience Store	Food (
4	Garden Villa	5.863280	118.048945	4.0	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	I Resta
5	Indah	5.842067	118.066095	0.0	Athletics & Sports	Noodle House	Baseball Stadium	Dim Sum Restaurant	Depart
6	Indah Jaya	5.843796	118.067200	0.0	Food Truck	Asian Restaurant	Grocery Store	Recreation Center	Depart
7	Lucky & Wemin	5.863112	118.062768	2.0	Clothing Store	Park	Chinese Restaurant	Convenience Store	Food
8	Mesra	5.861271	118.077664	0.0	Electronics Store	Convenience Store	Wings Joint	Other Great Outdoors	F Vege
9	Pertama	5.861339	118.069276	1.0	Chinese Restaurant	American Restaurant	Noodle House	Athletics & Sports	BBQ
10	Tinosan	5.861032	118.074517	0.0	Bakery	Music Venue	Bed & Breakfast	Malay Restaurant	Ch Resta
11	Tshun Ngen	5.858728	118.065804	0.0	Fruit & Vegetable Store	Asian Restaurant	Vegetarian / Vegan Restaurant	Grocery Store	F Chips
12	Tyng	5.867669	118.059997	0.0	Asian Restaurant	Bakery	Noodle House	Chinese Restaurant	M
13	Utama	5.864601	118.058569	0.0	Chinese Restaurant	Malay Restaurant	Fast Food Restaurant	Sushi Restaurant	Resta

In [67]:
```
target_merged.drop(index=1, inplace=True) #Drop Casa San Uno as there are NaNs
```

In [68]:
```
target_merged
```

Out[68]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Com V
0	Bunga Matahari	5.865810	118.075874	0.0	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	((Outd
2	Damai & Sri Taman	5.858482	118.078921	0.0	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Gr
3	Evergreen	5.873464	118.057834	3.0	Golf Course	Gym	Wings Joint	Convenience Store	Food (
4	Garden Villa	5.863280	118.048945	4.0	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	I Resta
5	Indah	5.842067	118.066095	0.0	Athletics & Sports	Noodle House	Baseball Stadium	Dim Sum Restaurant	Depart
6	Indah Jaya	5.843796	118.067200	0.0	Food Truck	Asian Restaurant	Grocery Store	Recreation Center	Depart
7	Lucky & Wemin	5.863112	118.062768	2.0	Clothing Store	Park	Chinese Restaurant	Convenience Store	Food
8	Mesra	5.861271	118.077664	0.0	Electronics Store	Convenience Store	Wings Joint	Other Great Outdoors	F Vege
9	Pertama	5.861339	118.069276	1.0	Chinese Restaurant	American Restaurant	Noodle House	Athletics & Sports	BBQ
10	Tinosan	5.861032	118.074517	0.0	Bakery	Music Venue	Bed & Breakfast	Malay Restaurant	Ch Resta
11	Tshun Ngen	5.858728	118.065804	0.0	Fruit & Vegetable Store	Asian Restaurant	Vegetarian / Vegan Restaurant	Grocery Store	F Chips
12	Tyng	5.867669	118.059997	0.0	Asian Restaurant	Bakery	Noodle House	Chinese Restaurant	M
13	Utama	5.864601	118.058569	0.0	Chinese Restaurant	Malay Restaurant	Fast Food Restaurant	Sushi Restaurant	Resta

In [69]:
```
#Convert float to int for Cluster Labels
target_merged['Cluster Labels'] = target_merged['Cluster Labels'].astype(int)
```

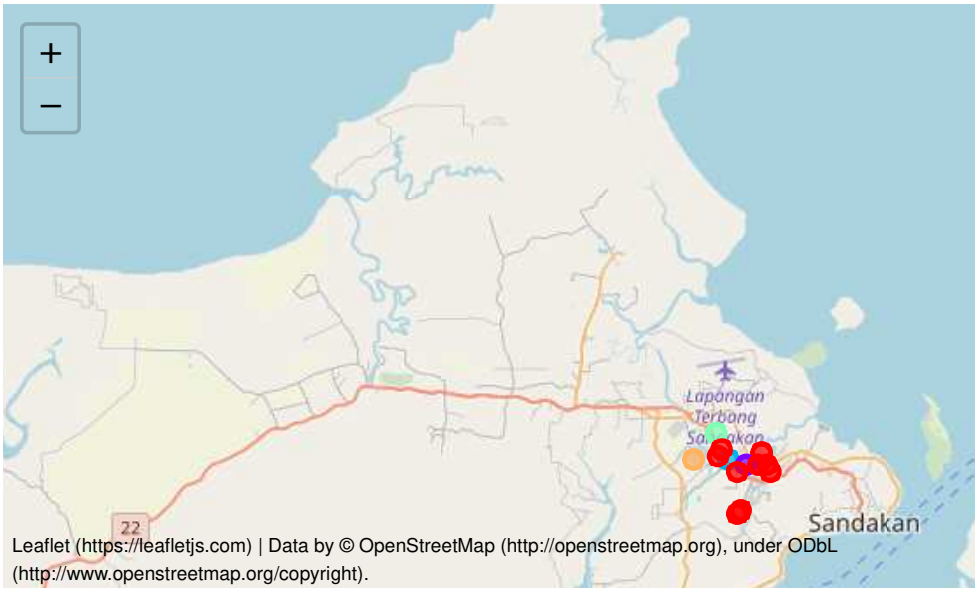
In [70]:
```
target_merged
```

Out[70]:

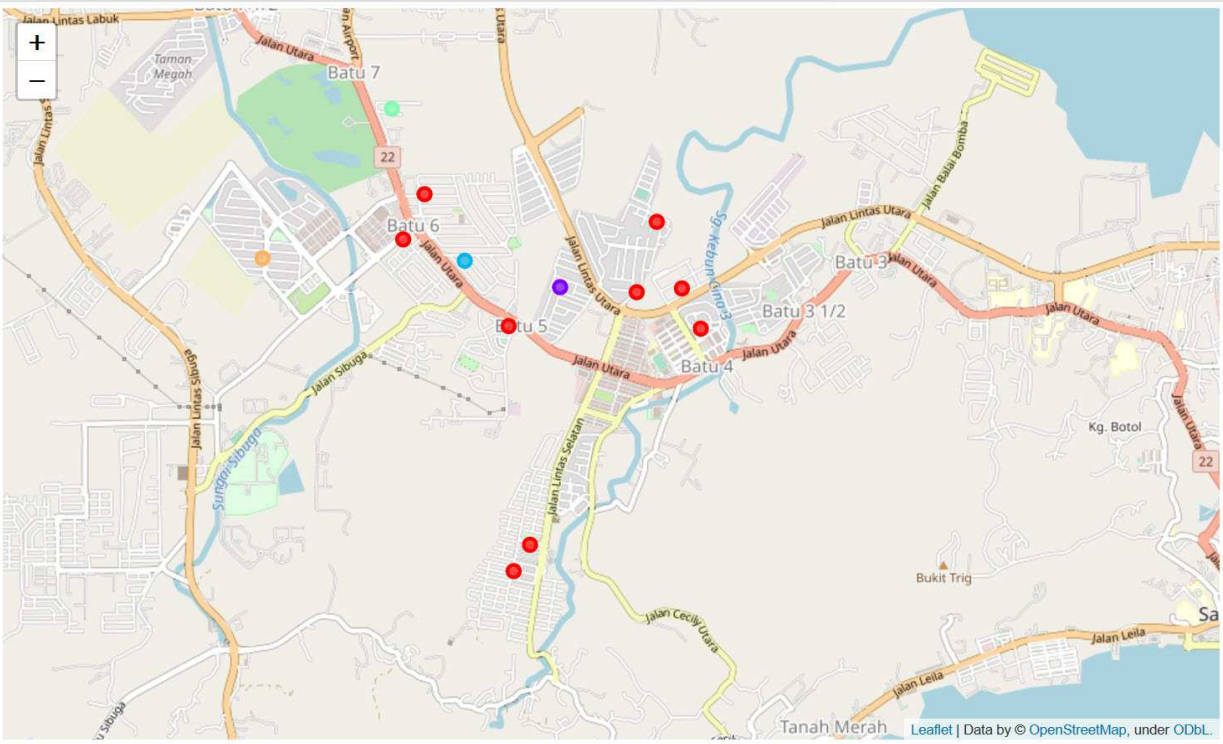
	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Com V
0	Bunga Matahari	5.865810	118.075874	0	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	((Outd
2	Damai & Sri Taman	5.858482	118.078921	0	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Gr
3	Evergreen	5.873464	118.057834	3	Golf Course	Gym	Wings Joint	Convenience Store	Food (
4	Garden Villa	5.863280	118.048945	4	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	I Resta
5	Indah	5.842067	118.066095	0	Athletics & Sports	Noodle House	Baseball Stadium	Dim Sum Restaurant	Depart
6	Indah Jaya	5.843796	118.067200	0	Food Truck	Asian Restaurant	Grocery Store	Recreation Center	Depart
7	Lucky & Wemin	5.863112	118.062768	2	Clothing Store	Park	Chinese Restaurant	Convenience Store	Food
8	Mesra	5.861271	118.077664	0	Electronics Store	Convenience Store	Wings Joint	Other Great Outdoors	F Vege
9	Pertama	5.861339	118.069276	1	Chinese Restaurant	American Restaurant	Noodle House	Athletics & Sports	BBQ
10	Tinosan	5.861032	118.074517	0	Bakery	Music Venue	Bed & Breakfast	Malay Restaurant	Ch Resta
11	Tshun Ngen	5.858728	118.065804	0	Fruit & Vegetable Store	Asian Restaurant	Vegetarian / Vegan Restaurant	Grocery Store	F Chips
12	Tyng	5.867669	118.059997	0	Asian Restaurant	Bakery	Noodle House	Chinese Restaurant	M
13	Utama	5.864601	118.058569	0	Chinese Restaurant	Malay Restaurant	Fast Food Restaurant	Sushi Restaurant	Resta

In [71]:
```
# create map
map_clusters = folium.Map(location=[latitude, longitude], zoom_start=11)

# 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]

# add markers to the map
markers_colors = []
for lat, lon, poi, cluster in zip(target_merged['Latitude'], target_merged['Long
titude'], target_merged['Neighbourhood'], target_merged['Cluster Labels']):
    label = folium.Popup(str(poi) + ' Cluster ' + str(cluster), parse_html=True)
    folium.CircleMarker(
        [lat, lon],
        radius=5,
        popup=label,
        color=rainbow[cluster-1],
        fill=True,
        fill_color=rainbow[cluster-1],
        fill_opacity=0.7).add_to(map_clusters)

map_clusters
```

Out[71]:



Out[71]:



- Red color = Cluster 0
- Violet color = Cluster 1
- Blue color = Cluster 2
- Green color = Cluster 3
- Yellow color = Cluster 4

Display each cluster

In [72]:
```
target_merged.loc[target_merged['Cluster Labels'] == 0]
```

Out[72]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th M Comm Ver
0	Bunga Matahari	5.865810	118.075874	0	Furniture / Home Store	Convenience Store	Hotel	Outdoors & Recreation	Ot Gr Outdo
2	Damai & Sri Taman	5.858482	118.078921	0	Deli / Bodega	Wings Joint	Chinese Restaurant	Fast Food Restaurant	Groc St
5	Indah	5.842067	118.066095	0	Athletics & Sports	Noodle House	Baseball Stadium	Dim Sum Restaurant	Departm St
6	Indah Jaya	5.843796	118.067200	0	Food Truck	Asian Restaurant	Grocery Store	Recreation Center	Departm St
8	Mesra	5.861271	118.077664	0	Electronics Store	Convenience Store	Wings Joint	Other Great Outdoors	Fru Vegeta St
10	Tinosan	5.861032	118.074517	0	Bakery	Music Venue	Bed & Breakfast	Malay Restaurant	Chine Restaur
11	Tshun Ngen	5.858728	118.065804	0	Fruit & Vegetable Store	Asian Restaurant	Vegetarian / Vegan Restaurant	Grocery Store	Fisl Chips Sh
12	Tyng	5.867669	118.059997	0	Asian Restaurant	Bakery	Noodle House	Chinese Restaurant	Mar
13	Utama	5.864601	118.058569	0	Chinese Restaurant	Malay Restaurant	Fast Food Restaurant	Sushi Restaurant	Restaur

In [73]:
```
target_merged.loc[target_merged['Cluster Labels'] == 1]
```

Out[73]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6t Co
9	Pertama	5.861339	118.069276	1	Chinese Restaurant	American Restaurant	Noodle House	Athletics & Sports	BBQ Joint	

In [74]:
```
target_merged.loc[target_merged['Cluster Labels'] == 2]
```

Out[74]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	
7	Lucky & Wemin	5.863112	118.062768	2	Clothing Store	Park	Chinese Restaurant	Convenience Store	Food Truck	

In [75]:
```
target_merged.loc[target_merged['Cluster Labels'] == 3]
```

Out[75]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	6 C
3	Evergreen	5.873464	118.057834	3	Golf Course	Gym	Wings Joint	Convenience Store	Food Court	

In [76]:
```
target_merged.loc[target_merged['Cluster Labels'] == 4]
```

Out[76]:

	Neighbourhood	Latitude	Longtitude	Cluster Labels	1st Most Common Venue	2nd Most Common Venue	3rd Most Common Venue	4th Most Common Venue	5th Most Common Venue	
4	Garden Villa	5.86328	118.048945	4	Café	Chinese Restaurant	Football Stadium	Hakka Restaurant	Indian Restaurant	R

Results and Discussion

The clustering results gave the most number of neighbourhoods are **cluster 0**. Businesses who are keen in setting up any businesses can refer to the clustering results and what sort of businesses are there.

Business people need to factor in costs like rental, utilities, land prices, transportation, labor etc before setting any businesses.

To recap, we collected data from relevant websites and merged them into a single csv file. Some data exploration were performed to look for any patterns amongst the features.

Then we decided to focus on Mile 4 to Mile 6 neighbourhood areas since majority of them are concentrated there.

We mapped these locations using Folium. We used Foursquare API to get the common venues visited by people who live there.

K-Means clustering is applied to cluster these neighbourhoods to five clusters and the result will give new business owners to analyze what sort of opportunities available.

Conclusion

The purpose of this project is to explore business opportunities in Sandakan neighbourhoods. Using clustering methods, we can identify popular venues which can be considered by business people.