__The Battle of the Neighborhoods - Week 2__

LEUNG CHUNG MING

__Introduction:__

This report is for those planning to start a new restaurant. Provide as many suggestions as possible about the factors to consider when starting a new business.

__Business problem:__

This report begins with a business issue. Focus on the issues you may encounter when opening a new restaurant. There are many factors to consider when setting up a small or large business. For example, if an investor wants to open a fusion Chinese restaurant, the first thing to be determined is the location of the new restaurant. Site selection, vegetable market, school, industrial zone commercial street, this type of shop has a large population of mobile population? If you target the target group to the middle class with high consumption power, you can choose the commercial street route, Choose from high-end department stores, high-end pedestrian streets, or famous commercial streets. Or a community-based route, the restaurant will be opened in the middle and high-end residential areas and scenic spots. This kind of site selection is quiet and heavy.

__Discussion:__

Let's discuss the problem statement mentioned above. First, investors want to open a fusion Chinese restaurant in Manhattan, New York City. New York City Manhattan Wall Street is the financial center of the United States because real estate is very expensive. Now she needs to figure out how many restaurants are in the A, B, C and other neighborhoods. If there are more than two restaurants in the vicinity, then a new restaurant with the same food in the community will be a big risk. Considering the rents in the neighborhood, choosing a place with fewer or no restaurants would be a good choice. This is the effect of reducing profits in order to reduce competition in the industry. Need to find a place that many people visit frequently so that the investor's business is above average. Downtown, cinemas, shopping centers and gas stations will help the restaurant's business operations. Restaurant ratings, customer sign-in may help determine the location of the crowd. I also recommend that the restaurant develop an executive business package to attract executives from the district to come to the business.

__Data Description:__

In [1]:

```
import pandas as pd
import matplotlib
import numpy as np
import math
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator
from collections import namedtuple

df_restaurant = pd.read_csv('https://data.cityofnewyork.us/api/views/43nn-pn8j/rows.cs
v?accessType=DOWNLOAD')
```

In [2]:

```
df_Mana = df_restaurant.loc[df_restaurant['BORO'] == 'Manhattan']

df_Mana.head()
```

Out[2]:

	CAMIS	DBA	BORO	BUILDING	STREET	ZIPCODE	PHONE	DESC
1	50077581	L'ADRESSE AMERICAN BISTRO	Manhattan	5	BRYANT PARK	10018.0	2122212510	
3	50048097	MEE'S NOODLE	Manhattan	930	2ND AVE	10022.0	2128880027	
4	50009874	EL PALACIO SEAFOOD MARKET	Manhattan	1049	SAINT NICHOLAS AVE	10032.0	2125680770	
7	41586126	NEW SPRING BOY CHINESE RESTAURANT	Manhattan	81	ALLEN STREET	10002.0	2126251921	
8	41589545	ZANNY'S CAFE	Manhattan	975	COLUMBUS AVENUE	10025.0	2123166849	

5 rows × 26 columns

In [3]:

```
df_ma_cd=df_Mana['CUISINE DESCRIPTION'].value_counts().to_frame()
df_ma_cd
```

Out[3]:

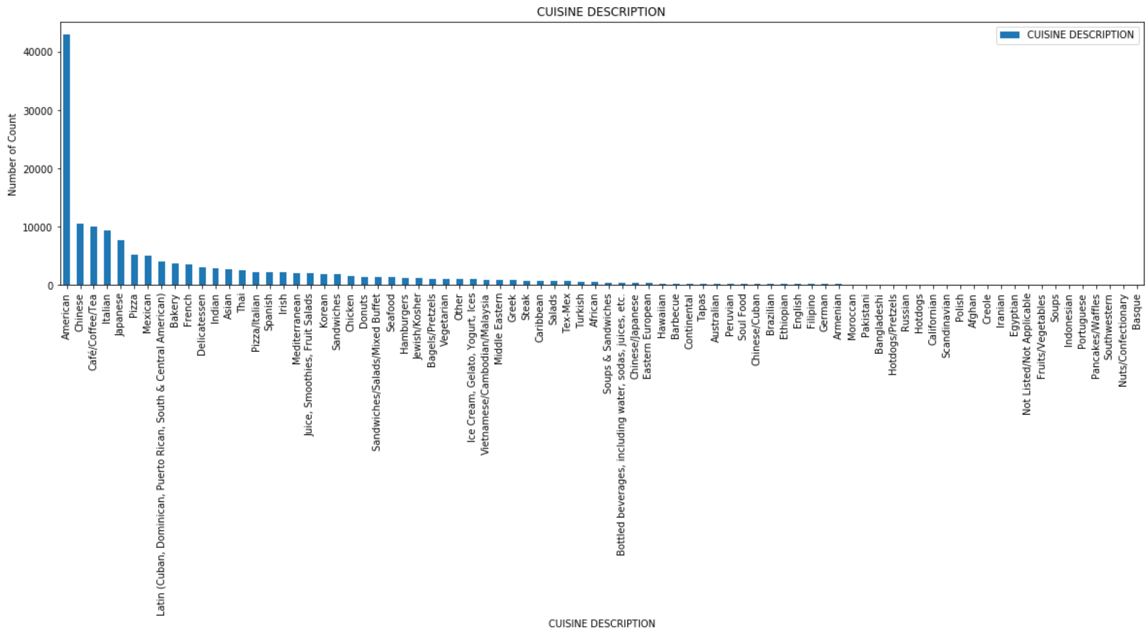
	CUISINE DESCRIPTION
American	43048
Chinese	10485
Café/Coffee/Tea	10016
Italian	9339
Japanese	7721
Pizza	5297
Mexican	5137
Latin (Cuban, Dominican, Puerto Rican, South & Central American)	4001
Bakery	3723
French	3560
Delicatessen	3023
Indian	2889
Asian	2709
Thai	2644
Pizza/Italian	2305
Spanish	2224
Irish	2157
Mediterranean	2145
Juice, Smoothies, Fruit Salads	2065
Korean	1890
Sandwiches	1877
Chicken	1619
Donuts	1439
Sandwiches/Salads/Mixed Buffet	1423
Seafood	1340
Hamburgers	1266
Jewish/Kosher	1195
Bagels/Pretzels	1080
Vegetarian	1061
Other	997
...	...
Soul Food	261
Chinese/Cuban	231
Brazilian	186
Ethiopian	181
English	181
Filipino	171

	CUISINE DESCRIPTION
German	166
Armenian	162
Moroccan	152
Pakistani	145
Bangladeshi	131
Hotdogs/Pretzels	118
Russian	114
Hotdogs	102
Californian	76
Scandinavian	62
Polish	55
Afghan	47
Creole	43
Iranian	43
Egyptian	41
Not Listed/Not Applicable	39
Fruits/Vegetables	37
Soups	36
Indonesian	33
Portuguese	27
Pancakes/Waffles	25
Southwestern	22
Nuts/Confectionary	9
Basque	7

80 rows × 1 columns

In [4]:

```
df_ma_cd.plot(kind='bar', figsize=(20, 5))

plt.title('CUISINE DESCRIPTION') # add a title to the histogram
plt.ylabel('Number of Count') # add y-label
plt.xlabel('CUISINE DESCRIPTION') # add x-label

plt.show()
```



In [5]:

```
df_ma_st=df_Mana['STREET'].value_counts().to_frame()
df_ma_st
```

Out[5]:

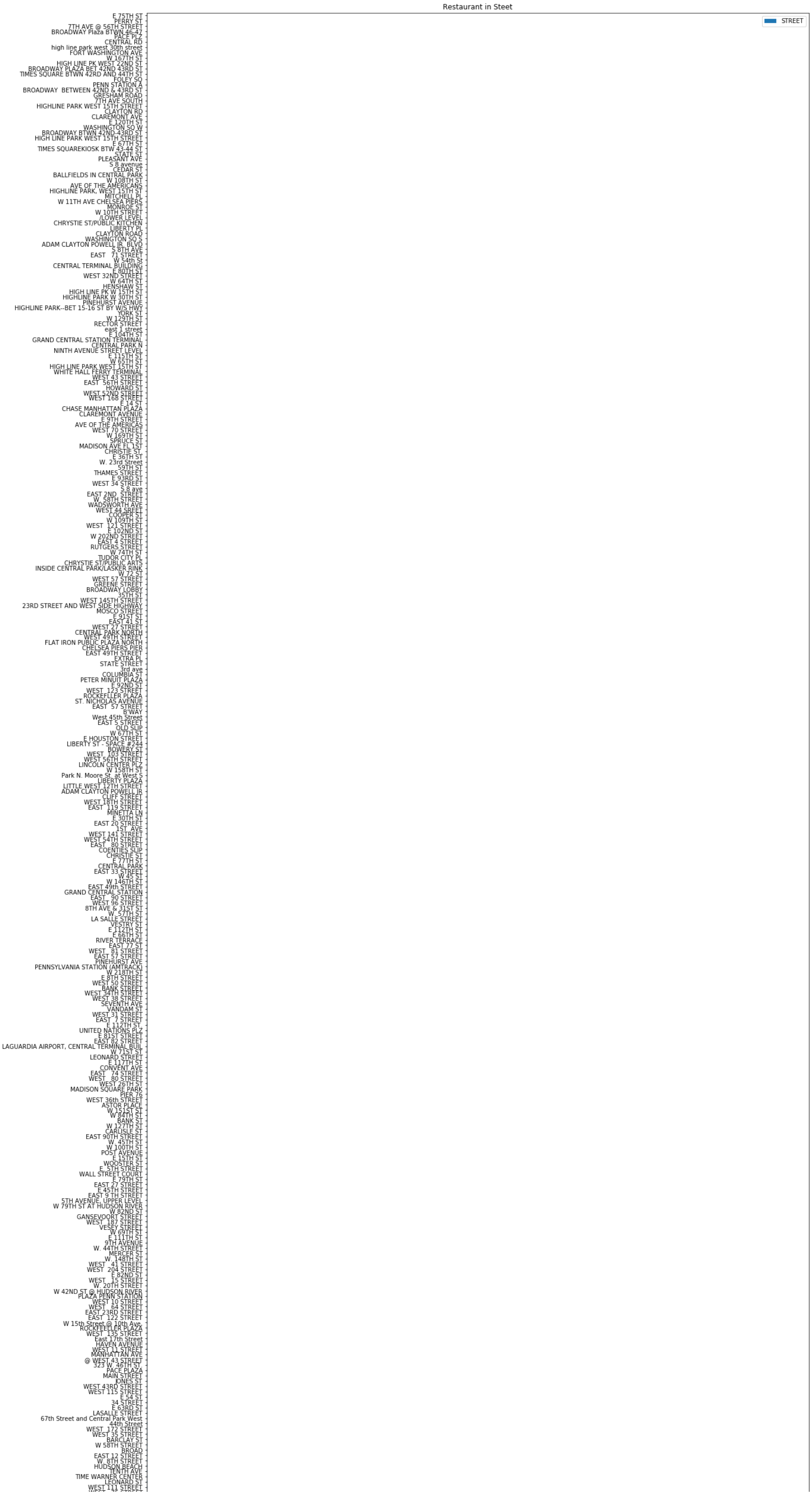
	STREET
BROADWAY	9204
2ND AVE	3678
1ST AVE	2909
2 AVENUE	2875
3RD AVE	2874
3 AVENUE	2300
LEXINGTON AVE	2266
1 AVENUE	2196
9TH AVE	2155
LEXINGTON AVENUE	2148
AMSTERDAM AVE	2130
9 AVENUE	2045
AMSTERDAM AVENUE	2025
8TH AVE	1928
8 AVENUE	1703
7 AVENUE	1303
MADISON AVE	1032
MADISON AVENUE	1030
AVENUE OF THE AMERICAS	998
7TH AVE	990
AVENUE A	957
10 AVENUE	905
5TH AVE	888
MOTT STREET	887
COLUMBUS AVE	882
BOWERY	839
WEST 46 STREET	801
SAINT NICHOLAS AVE	797
10TH AVE	795
COLUMBUS AVENUE	765
...	...
CEDAR ST	2
S 8 avenue	2
PLEASANT AVE	2
STATE ST	2
TIMES SQUAREKIOSK BTW 43-44 ST	2
E 67TH ST	2

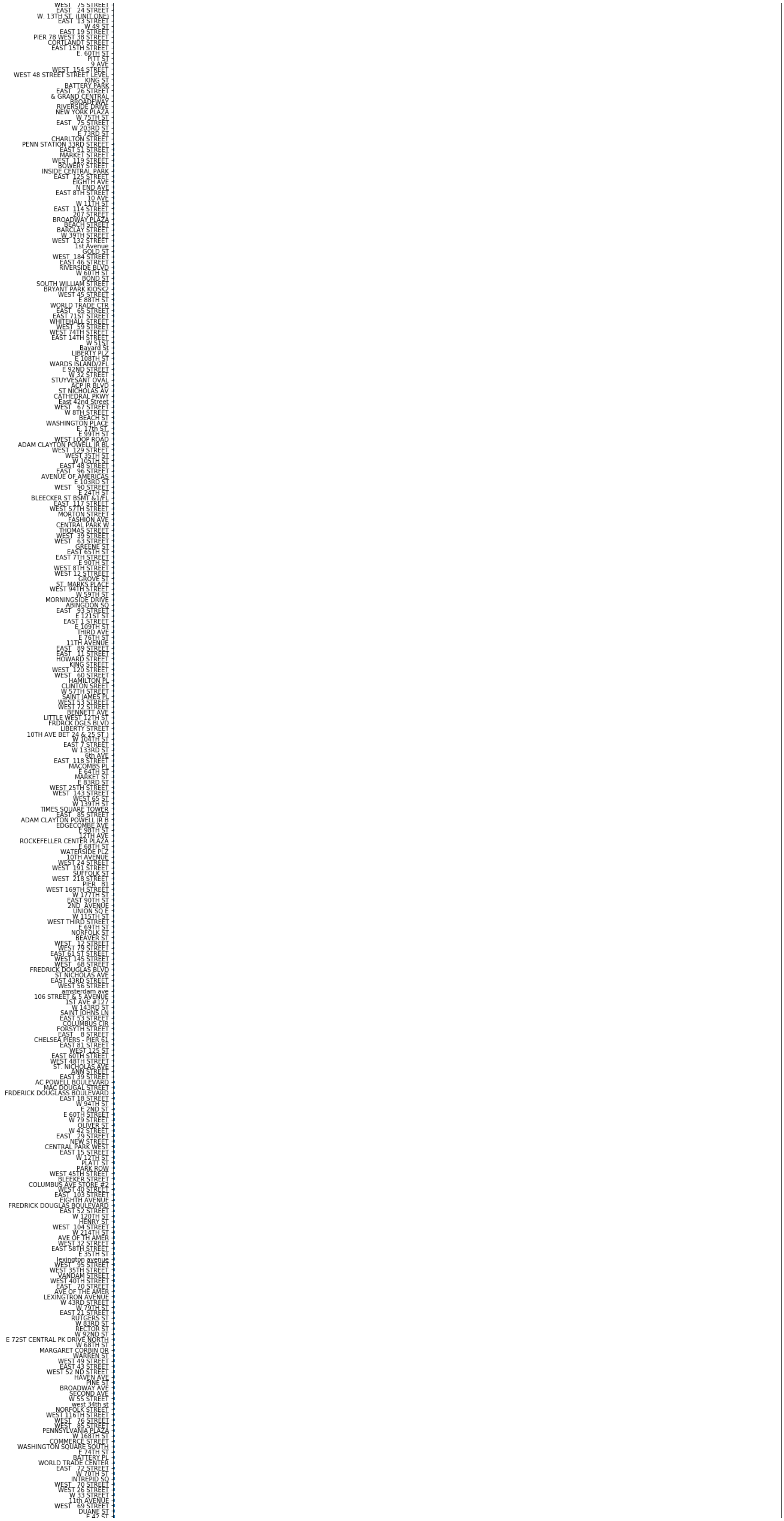
	STREET
HIGH LINE PARK WEST 15TH STREET	1
BROADWAY BTWN 42ND-43RD ST	1
WASHINGTON SQ W	1
E 120TH ST	1
CLAREMONT AVE	1
CLAYTON RD	1
HIGHLINE PARK WEST 15TH STREET	1
7TH AVE SOUTH	1
GRESHAM ROAD	1
BROADWAY BETWEEN 42ND & 43RD ST	1
PENN STATION A	1
FOLEY SQ	1
TIMES SQUARE BTWN 42RD AND 44TH ST	1
BROADWAY PLAZA BET 42ND 43RD ST	1
HIGH LINE PK WEST 22ND ST	1
W 167TH ST	1
FORT WASHINGTON AVE	1
high line park west 30th street	1
CENTRAL RD	1
PACE PLZ	1
BROADWAY Plaza BTWN 46-47	1
7TH AVE @ 56TH STREET	1
PERRY ST	1
E 75TH ST	1

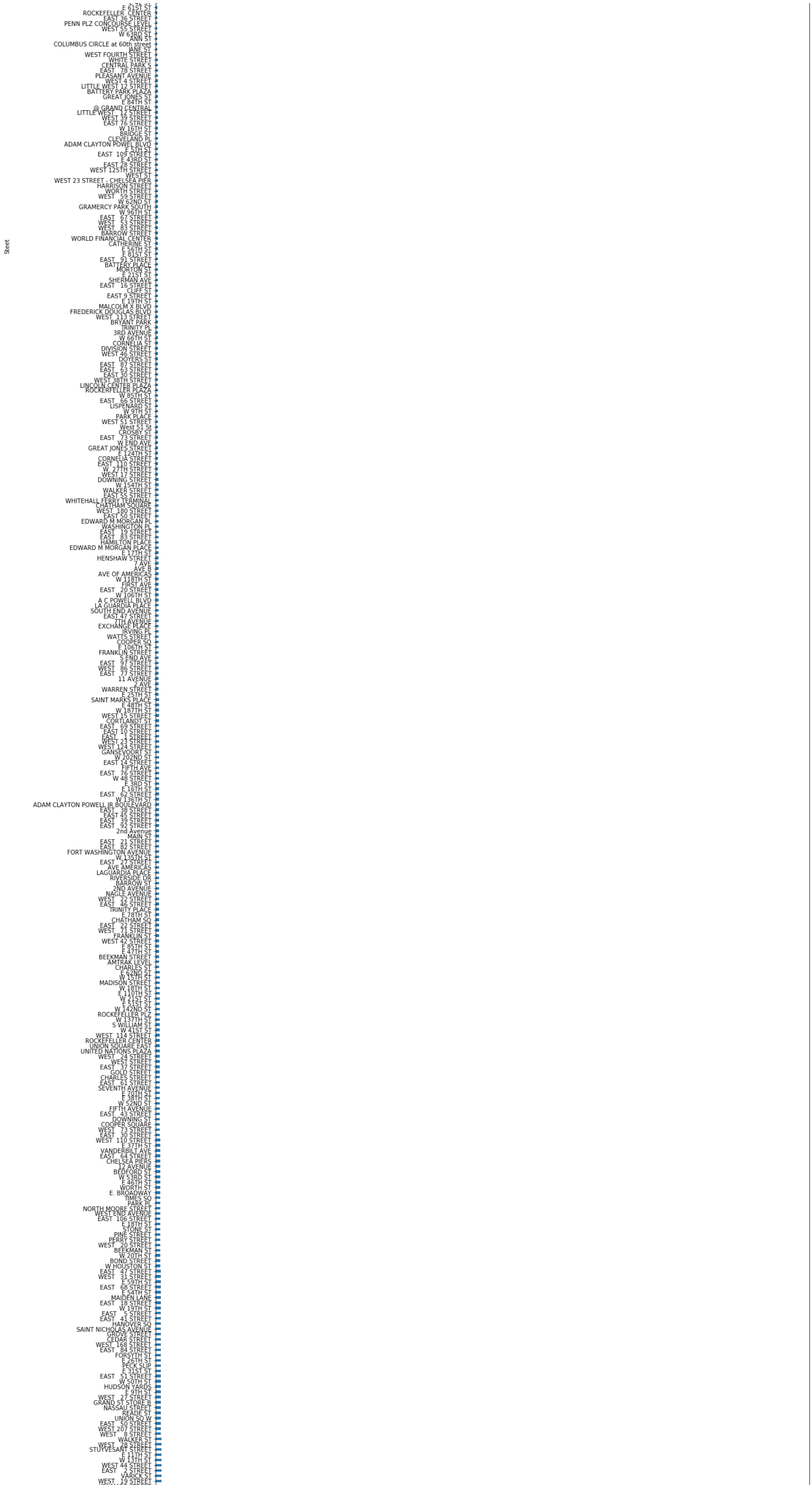
1214 rows × 1 columns

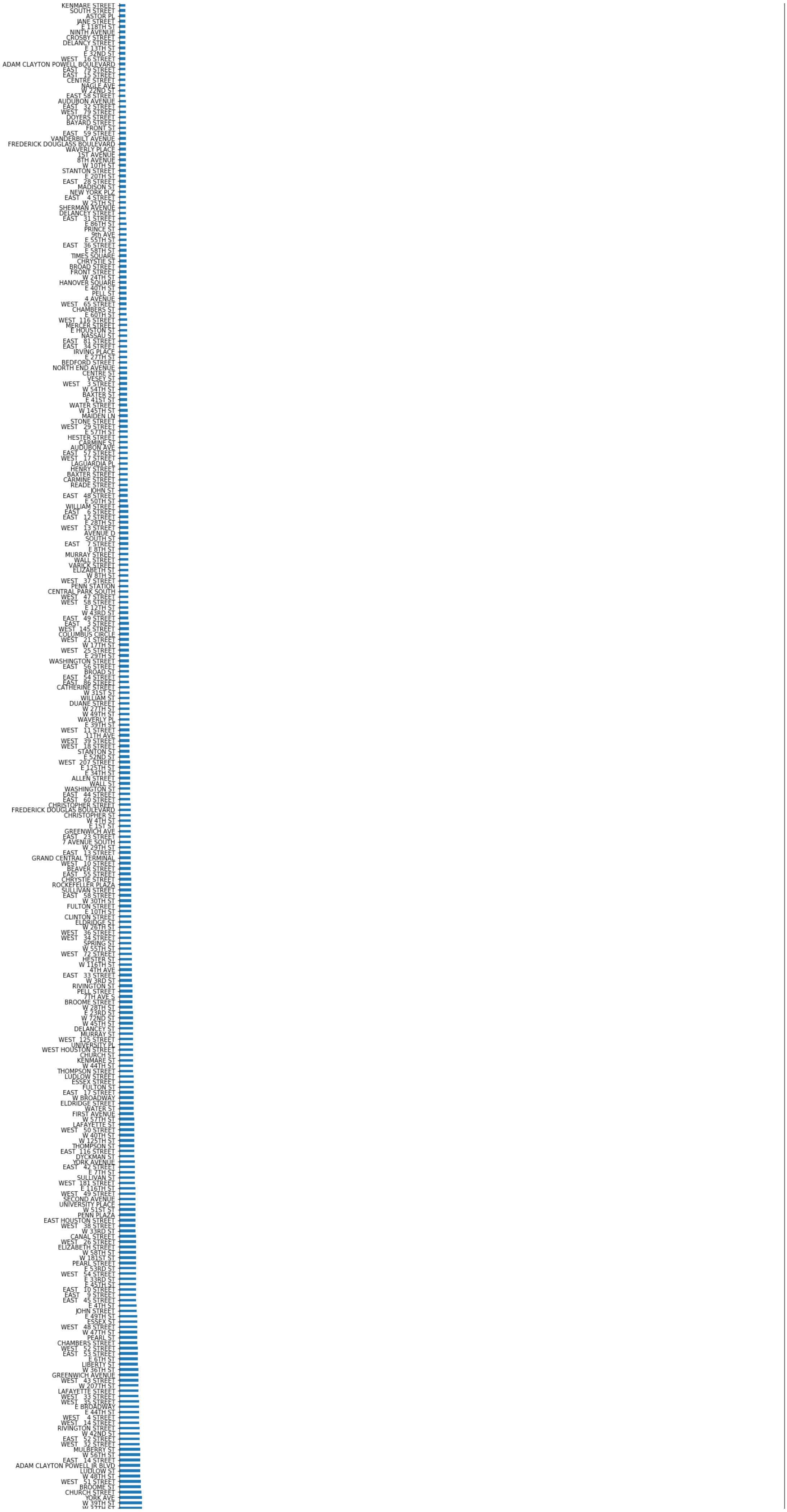
In [6]:

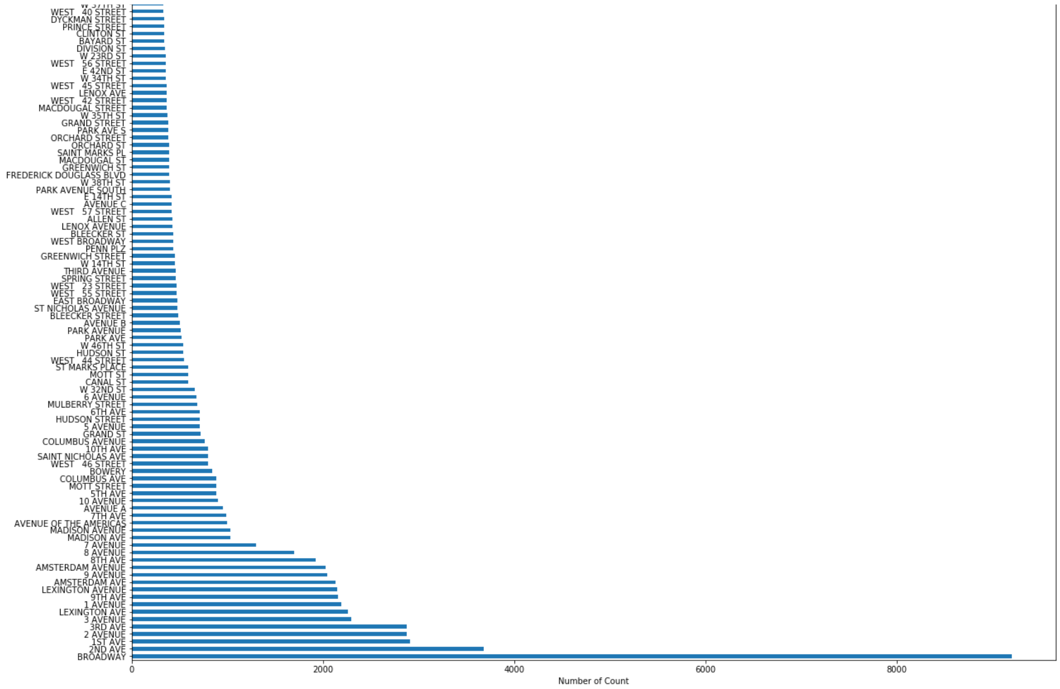
```
df_ma_st.plot(kind='barh', figsize=(20, 200))

plt.title('Restaurant in Steet') # add a title to the histogram
plt.ylabel('Steet') # add y-label
plt.xlabel('Number of Count') # add x-label

plt.show()
```


Restaurant in Steet
STREET




Steet


KENMARE STREET
SOUTH STREET
ASTOR PL
JANE STREET
E 118TH ST
NINTH AVENUE
CROSBY STREET
DELANCY STREET
E 13TH ST
E 32ND ST
WEST 16 STREET
ADAM CLAYTON POWELL BOULEVARD
EAST 79 STREET
EAST 15 STREET
CENTRE STREET
NAGLE AVE
W 22ND ST
EAST 58 STREET
AUDUBON AVENUE
EAST 32 STREET
WEST 79 STREET
DOYERS STREET
BAYARD STREET
FRONT ST
EAST 59 STREET
VANDERBILT AVENUE
FREDERICK DOUGLASS BOULEVARD
WAVERLY PLACE
1ST AVENUE
8TH AVENUE
W 10TH ST
STANTON STREET
E 20TH ST
EAST 28 STREET
MADISON ST
NEW YORK PLZ
EAST 4 STREET
W 25TH ST
SHERMAN AVENUE
DELANCEY STREET
EAST 31 STREET
E 86TH ST
PRINCE ST
9th AVE
E 55TH ST
EAST 36 STREET
E 58TH ST
TIMES SQUARE
CHRYSTIE ST
BROAD STREET
FRONT STREET
W 24TH ST
HANOVER SQUARE
E 40TH ST
PELL ST
4 AVENUE
WEST 65 STREET
CHAMBERS ST
E 60TH ST
WEST 116 STREET
MERCER STREET
E HOUSTON ST
NASSAU ST
EAST 81 STREET
EAST 34 STREET
IRVING PLACE
E 27TH ST
BEDFORD STREET
NORTH END AVENUE
CENTRE ST
VESEY ST
WEST 3 STREET
W 54TH ST
BAXTER ST
E 41ST ST
WATER STREET
W 145TH ST
MAIDEN LN
STONE STREET
WEST 29 STREET
E 57TH ST
HESTER STREET
CARMINE ST
AUDUBON AVE
EAST 57 STREET
WEST 17 STREET
LAGUARDIA PL
HENRY STREET
BAXTER STREET
CARMINE STREET
READE STREET
JOHN ST
EAST 48 STREET
E 50TH ST
WILLIAM STREET
EAST 6 STREET
EAST 12 STREET
E 28TH ST
WEST 13 STREET
AVENUE D
SOUTH ST
EAST 7 STREET
E 8TH ST
MURRAY STREET
WALL STREET
VARICK STREET
ELIZABETH ST
W 8TH ST
WEST 37 STREET
PENN STATION
CENTRAL PARK SOUTH
WEST 47 STREET
WEST 58 STREET
E 12TH ST
W 43RD ST
EAST 49 STREET
EAST 3 STREET
WEST 145 STREET
COLUMBUS CIRCLE
WEST 21 STREET
W 17TH ST
WEST 25 STREET
E 29TH ST
WASHINGTON STREET
EAST 56 STREET
BROAD ST
EAST 54 STREET
EAST 86 STREET
CATHERINE STREET
W 31ST ST
WILLIAM ST
DUANE STREET
W 27TH ST
W 49TH ST
WAVERLY PL
E 39TH ST
WEST 11 STREET
11TH AVE
WEST 39 STREET
WEST 18 STREET
STANTON ST
E 52ND ST
WEST 207 STREET
E 125TH ST
E 34TH ST
ALLEN STREET
WALL ST
WASHINGTON ST
EAST 44 STREET
EAST 60 STREET
CHRISTOPHER STREET
FREDERICK DOUGLAS BOULEVARD
CHRISTOPHER ST
W 4TH ST
E 1ST ST
GREENWICH AVE
EAST 23 STREET
7 AVENUE SOUTH
W 29TH ST
EAST 13 STREET
GRAND CENTRAL TERMINAL
WEST 10 STREET
BEAVER STREET
EAST 55 STREET
CHRYSTIE STREET
ROCKEFELLER PLAZA
SULLIVAN STREET
EAST 58 STREET
W 30TH ST
FULTON STREET
E 10TH ST
CLINTON STREET
ELDRIDGE ST
W 26TH ST
WEST 36 STREET
WEST 34 STREET
SPRING ST
W 55TH ST
WEST 72 STREET
HESTER ST
W 116TH ST
4TH AVE
EAST 33 STREET
W 3RD ST
RIVINGTON ST
PELL STREET
7TH AVE S
BROOME STREET
W 28TH ST
E 23RD ST
W 72ND ST
W 45TH ST
DELANCEY ST
MURRAY ST
WEST 125 STREET
UNIVERSITY PL
WEST HOUSTON STREET
CHURCH ST
KENMARE ST
W 44TH ST
THOMPSON STREET
LUDLOW STREET
ESSEX STREET
FULTON ST
EAST 17 STREET
W BROADWAY
ELDRIDGE STREET
WATER ST
FIRST AVENUE
W 57TH ST
LAFAYETTE ST
WEST 50 STREET
W 40TH ST
W 125TH ST
THOMPSON ST
EAST 116 STREET
DYCKMAN ST
YORK AVENUE
EAST 42 STREET
E 7TH ST
SULLIVAN ST
WEST 181 STREET
E 116TH ST
WEST 49 STREET
SECOND AVENUE
UNIVERSITY PLACE
W 51ST ST
PENN PLAZA
EAST HOUSTON STREET
WEST 38 STREET
W 33RD ST
CANAL STREET
WEST 26 STREET
ELIZABETH STREET
W 58TH ST
W 181ST ST
PEARL STREET
E 53RD ST
WEST 54 STREET
E 33RD ST
E 45TH ST
EAST 10 STREET
EAST 9 STREET
EAST 45 STREET
E 4TH ST
JOHN STREET
E 49TH ST
ESSEX ST
WEST 48 STREET
W 47TH ST
PEARL ST
CHAMBERS STREET
WEST 52 STREET
EAST 53 STREET
E 6TH ST
LIBERTY ST
W 36TH ST
GREENWICH AVENUE
WEST 43 STREET
W 207TH ST
LAFAYETTE STREET
WEST 33 STREET
WEST 35 STREET
E BROADWAY
E 44TH ST
WEST 4 STREET
WEST 14 STREET
RIVINGTON STREET
W 42ND ST
EAST 52 STREET
WEST 23 STREET
MULBERRY ST
W 56TH ST
EAST 14 STREET
ADAM CLAYTON POWELL JR BLVD
LUDLOW ST
W 48TH ST
WEST 51 STREET
BROOME ST
CHURCH STREET
YORK AVE
W 39TH ST



In [16]:

```
from pandas.io.json import json_normalize
!pip install folium==0.5.0
import folium
from geopy.geocoders import Nominatim
import requests
import json
import pandas as pd
```

```
Requirement already satisfied: folium==0.5.0 in /opt/conda/envs/Python36/l
ib/python3.6/site-packages (0.5.0)
Requirement already satisfied: six in /opt/conda/envs/Python36/lib/python
3.6/site-packages (from folium==0.5.0) (1.12.0)
Requirement already satisfied: jinja2 in /opt/conda/envs/Python36/lib/pyth
on3.6/site-packages (from folium==0.5.0) (2.10)
Requirement already satisfied: branca in /opt/conda/envs/Python36/lib/pyth
on3.6/site-packages (from folium==0.5.0) (0.3.1)
Requirement already satisfied: requests in /opt/conda/envs/Python36/lib/py
thon3.6/site-packages (from folium==0.5.0) (2.21.0)
Requirement already satisfied: MarkupSafe>=0.23 in /opt/conda/envs/Python3
6/lib/python3.6/site-packages (from jinja2->folium==0.5.0) (1.1.0)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /opt/conda/envs/Py
thon36/lib/python3.6/site-packages (from requests->folium==0.5.0) (3.0.4)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /opt/conda/envs/Py
thon36/lib/python3.6/site-packages (from requests->folium==0.5.0) (1.24.1)
Requirement already satisfied: idna<2.9,>=2.5 in /opt/conda/envs/Python36/
lib/python3.6/site-packages (from requests->folium==0.5.0) (2.8)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/envs/Pytho
n36/lib/python3.6/site-packages (from requests->folium==0.5.0) (2019.6.16)
```

In [17]:

```
# Define Foursquare Credentials and Version
CLIENT_ID = 'HRMBKZUASN1NWO005IQK4TGG15UVEY5GCLJCYXHXW0VDP00K' # your Foursquare ID
CLIENT_SECRET = 'JSXFO23NR2OMICQSZRFQYDAZG1GMNRALXXACAFVNF5CGAM4C' # your Foursquare Se
cret
VERSION = '20180604'
limit = 100
print('Your credentails:')
print('CLIENT_ID:'+ CLIENT_ID)
print('CLIENT_SECRET:'+ CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID:HRMBKZUASN1NWO005IQK4TGG15UVEY5GCLJCYXHXW0VDP00K
CLIENT_SECRET:JSXFO23NR2OMICQSZRFQYDAZG1GMNRALXXACAFVNF5CGAM4C
```

In [18]:

```
neighborhood_latitude=40.7068769
neighborhood_longitude=-74.0112656
```

In [19]:

```
LIMIT = 200 # limit of number of venues returned by Foursquare API
radius = 400 # define radius

# create URL
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}
&ll={},{}&radius={}&limit={}'.format(
    CLIENT_ID,
    CLIENT_SECRET,
    VERSION,
    neighborhood_latitude,
    neighborhood_longitude,
    radius,
    LIMIT)
url # display URL
```

Out[19]:

```
'https://api.foursquare.com/v2/venues/explore?&client_id=HRMBKZUASN1NWO005
IQK4TGG15UVEY5GCLJCYXHXW0VDP00K&client_secret=JSXFO23NR2OMICQSZRFQYDAZG1GM
NRALXXACAFVNF5CGAM4C&v=20180604&ll=40.7068769,-74.0112656&radius=400&limit
=200'
```

In [20]:

```
results = requests.get(url).json()
#results
```

In [21]:

```
# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']
```

In [22]:

```
venues = results['response']['groups'][0]['items']

SGnearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'venue.loca
tion.lng']
SGnearby_venues =SGnearby_venues.loc[:, filtered_columns]

# filter the category for each row
SGnearby_venues['venue.categories'] = SGnearby_venues.apply(get_category_type, axis=1)

# clean columns
SGnearby_venues.columns = [col.split(".")[-1] for col in SGnearby_venues.columns]

SGnearby_venues
```

Out[22]:

	name	categories	lat	lng
0	Manhatta	New American Restaurant	40.707654	-74.009138
1	Physique 57	Gym / Fitness Center	40.706844	-74.013014
2	One Medical	Doctor's Office	40.706204	-74.011712
3	Greenwich St Jewelers	Jewelry Store	40.708004	-74.012801
4	Equinox Wall Street	Gym	40.707273	-74.010543
5	Fearless Girl	Monument / Landmark	40.706772	-74.010963
6	Bluestone Lane	Café	40.706268	-74.011687
7	Neapolitan Express	Pizza Place	40.706908	-74.009685
8	The Capital Grille	American Restaurant	40.707903	-74.010086
9	Cipriani Wall Street	Event Space	40.706038	-74.009269
10	Crunch - FiDi	Gym / Fitness Center	40.708614	-74.010013
11	Delmonico's	Steakhouse	40.705153	-74.010551
12	Sam's Falafel	Food Truck	40.708744	-74.011458
13	Champs Gourmet Deli	Deli / Bodega	40.706470	-74.011559
14	9/11 Tribute Museum	Museum	40.707995	-74.013701
15	Tiffany & Co.	Jewelry Store	40.706525	-74.010226
16	Federal Hall National Memorial	Historic Site	40.707120	-74.010552
17	Black Fox Coffee Co.	Coffee Shop	40.706573	-74.008155
18	Luke's Lobster	Seafood Restaurant	40.704488	-74.010915
19	William Beaver House	Event Space	40.705217	-74.010045
20	Sophie's Cuban Cuisine	Cuban Restaurant	40.705424	-74.012299
21	La Colombe Torrefaction	Coffee Shop	40.705899	-74.008421
22	The Setai Spa Wall Street	Spa	40.706131	-74.011420
23	sweetgreen	Salad Place	40.705586	-74.008382
24	Rouge Tomate Cart	Food Truck	40.707390	-74.014345
25	Hermès	Accessories Store	40.706505	-74.010993
26	&pizza	Pizza Place	40.706615	-74.009943
27	Crossfit Wall Street	Gym	40.705544	-74.012139
28	Taim	Falafel Restaurant	40.707727	-74.008265
29	Reserve Cut	Steakhouse	40.706132	-74.011354
...	...	...	...	...
70	AKA Wall Street	Hotel	40.708030	-74.007850
71	Chipotle Mexican Grill	Mexican Restaurant	40.704888	-74.012846
72	Black Fox	Coffee Shop	40.703968	-74.011160
73	Il Gelato @ Eataly	Ice Cream Shop	40.709869	-74.011686
74	Dig Inn	American Restaurant	40.706106	-74.007290

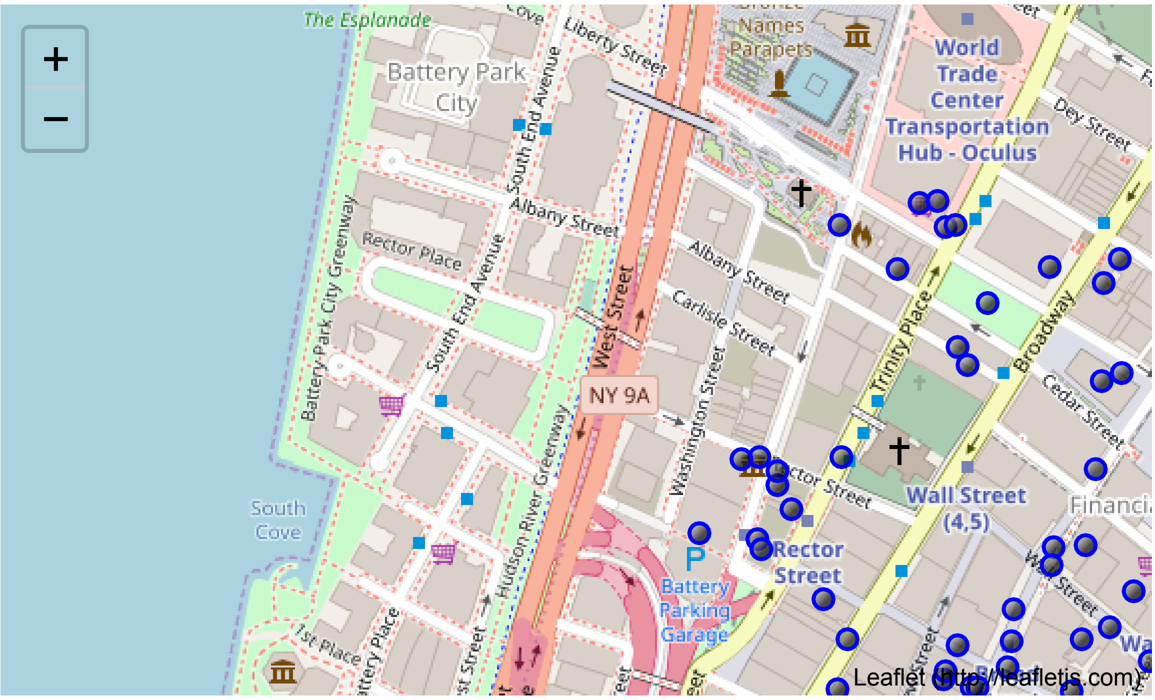
	name	categories	lat	lng
75	Gregorys Coffee	Coffee Shop	40.706110	-74.012850
76	Periscope Coffee On Pearl Street	Food Truck	40.704691	-74.009301
77	For Five Coffee Roasters	Coffee Shop	40.709554	-74.010576
78	La Pasticceria di Eataly Downtown	Bakery	40.709890	-74.011598
79	The Lovelace Cocktail & Gin Bar	Cocktail Bar	40.703585	-74.010600
80	Zuccotti Park	Park	40.709259	-74.011245
81	Fraunces Tavern	Bar	40.703526	-74.011395
82	Louise Nevelson Plaza	Park	40.707680	-74.007965
83	Biryani House (Cart)	Food Truck	40.705050	-74.012313
84	George's New York	Diner	40.707895	-74.013500
85	The Cauldron	Bar	40.704354	-74.010356
86	Juice Press	Juice Bar	40.707281	-74.010204
87	Pure Liquid Wine & Spirits	Wine Shop	40.710083	-74.011791
88	Vintry Wine & Whiskey	Wine Bar	40.704419	-74.010257
89	JOE & THE JUICE	Juice Bar	40.705570	-74.008160
90	LOFT	Women's Store	40.704695	-74.013031
91	O'Hara's Restaurant & Pub	Pub	40.709894	-74.012836
92	Bluestone Lane	Café	40.704599	-74.008748
93	Blue Park Kitchen	American Restaurant	40.706503	-74.008077
94	SoulCycle FiDi	Cycle Studio	40.706904	-74.006717
95	Nish Nush	Falafel Restaurant	40.709418	-74.008056
96	Residence Inn by Marriott New York Downtown Ma...	Hotel	40.709605	-74.009827
97	Just Salad	Salad Place	40.703539	-74.011627
98	Bobby Van's	Steakhouse	40.706291	-74.011038
99	Giardino D'oro	Italian Restaurant	40.707548	-74.007249

100 rows × 4 columns

In [23]:

```
# create map of Singapore place  using latitude and longitude values
map_sg = folium.Map(location=[40.7068769,-74.0112656], zoom_start=16)

# add markers to map
for lat, lng, label in zip(SGnearby_venues['lat'], SGnearby_venues['lng'], SGnearby_ven
ues['name']):
    label = folium.Popup(label, parse_html=True)
    folium.RegularPolygonMarker(
        [lat, lng],
        number_of_sides=16,
        radius=5,
        popup=label,
        color='blue',
        fill_color='#0f0f0f',
        fill_opacity=0.7,
    ).add_to(map_sg)

map_sg
```

Out[23]:



We now have the coordinates of centers (Wall Street) of neighborhoods/areas to be evaluated, equally spaced and within 400m from restaurant.

Investors want to open a fusion Chinese restaurant in Manhattan, New York City, so first I compare the data from the restaurant location, because for this project, choosing a place with fewer or no restaurants would be a good choice. The plan only tests the data I retrieved. I will use a formula to find out which community is suitable for a new restaurant. Let's search for each location in the main place listed by Foursquare. This table finds 100 places. This histogram highlights the proportion of each place. It is very uniform. For Manhattan, it is known to be in a logical view of Manhattan. We can already point us to one place instead of another.

In [24]:

```
import numpy as np

name='16 Maiden Ln','141 Duane St','124 Chambers St','77 Warren St','291 Broadway','45
 Nassau St','41 Nassau St','Equitable Building','125 Maiden Ln'
Area='Financial District, New York City, NY','Tribeca, New York City, NY','Tribeca, New
York City, NY','Tribeca, New York City, NY','Financial District, New York City, NY','Fi
nancial District, New York City, NY','120 Broadway, New York, NY,','Financial District,
New York City, NY'
Level='Floor Ground','Floor Ground','Space','Space','Space','Floor Lower Level','Floor
 Ground','Space','Space'
sqft=1200,3300,4800,5050,1750,6050,1650,16980,5350
lat=40.70927,40.71657,40.71516,40.71503,40.71492,40.70887,40.70879,40.70851,40.70672
lng=-74.01150,-74.00990,-74.01100,-74.01260,-74.00830,-74.01160,-74.01160,-74.01310,-7
4.00870


rents=list(zip(name, Area, Level, sqft, lat, lng))

df_rents = pd.DataFrame(rents, columns = ['name', 'Area', 'Level', 'sqft', 'lat', 'lng'
])

df_rents
```

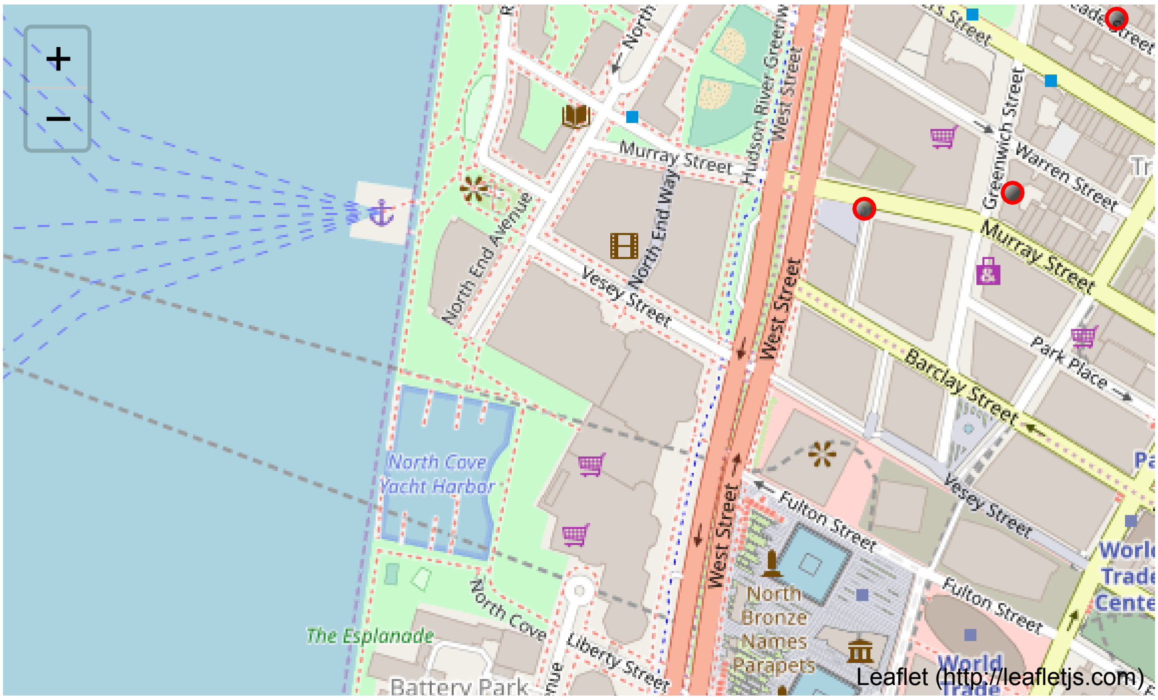
Out[24]:

	name	Area	Level	sqft	lat	lng
0	16 Maiden Ln	Financial District, New York City, NY	Floor Ground	1200	40.70927	-74.0115
1	141 Duane St	Tribeca, New York City, NY	Floor Ground	3300	40.71657	-74.0099
2	124 Chambers St	Tribeca, New York City, NY	Space	4800	40.71516	-74.0110
3	77 Warren St	Tribeca, New York City, NY	Space	5050	40.71503	-74.0126
4	291 Broadway	Financial District, New York City, NY	Space	1750	40.71492	-74.0083
5	45 Nassau St	Financial District, New York City, NY	Floor Lower Level	6050	40.70887	-74.0116
6	41 Nassau St	120 Broadway, New York, NY,	Floor Ground	1650	40.70879	-74.0116
7	Equitable Building	Financial District, New York City, NY	Space	16980	40.70851	-74.0131

In [25]:

```
from folium import plugins

map_rents = folium.Map(location=[40.7118769,-74.0112656], zoom_start=16)

# add markers to map
for lat, lng, label in zip(df_rents['lat'], df_rents['lng'], df_rents['name']):
    label = folium.Popup(label, parse_html=True)
    folium.RegularPolygonMarker(
        [lat, lng],
        number_of_sides=16,
        radius=5,
        popup=label,
        color='red',
        fill_color='#0f0f0f',
        fill_opacity=0.7,
    ).add_to(map_rents)

for lat, lng, label in zip(SGnearby_venues['lat'], SGnearby_venues['lng'], SGnearby_ven
ues['name']):
    label = folium.Popup(label, parse_html=True)
    folium.RegularPolygonMarker(
        [lat, lng],
        number_of_sides=16,
        radius=5,
        popup=label,
        color='blue',
        fill_color='#0f0f0f',
        fill_opacity=0.7,
    ).add_to(map_rents)

map_rents
```

Out[25]:



__Methodology:__

In this case, we finding much open fusion Chinese restaurant in Manhattan, New York City in Manhattan, Nw York City of area.

In this project, we will direct our efforts on detecting areas of Manhattan, New York City that have a Chinese restaurant. We will limit our analysis to area 400m.

In the first step we have collected the required data: location and type (category) of every restaurant within 400m from Manhattan (Wall Street). We have also identified the restaurant (according to Foursquare categorization).

The second step in our analysis will be calculation and exploration near Wall Street can open restaurant rental market and focus our attention on those areas.

In the third and final step, we will focus on most promising areas and within those create clusters of locations that meet some basic requirements established in discussion with stakeholders: we will take into consideration locations with no more than two Chinese restaurants, and we want locations near Wall Street. We will present the map of all such locations but also create clusters of those locations to identify general addresses which should be a starting point for final 'street-level' exploration and search for optimal venue location by stakeholders.

__Analysis:__

In [26]:

```
df_rents.drop(df_rents.index[[1,2,3,4]])
```

Out[26]:

	name	Area	Level	sqft	lat	lng
0	16 Maiden Ln	Financial District, New York City, NY	Floor Ground	1200	40.70927	-74.0115
5	45 Nassau St	Financial District, New York City, NY	Floor Lower Level	6050	40.70887	-74.0116
6	41 Nassau St	120 Broadway, New York, NY,	Floor Ground	1650	40.70879	-74.0116
7	Equitable Building	Financial District, New York City, NY	Space	16980	40.70851	-74.0131

Let's perform some basic interpretive data analysis and get some extra information from the raw data. First let us calculate the number of restaurants in the Manhattan, New York City area to get results for more than 15,000 restaurants. We create a map and try to extract some meaningful information from it. In addition, let us display the 100 places in Manhattan, New York City on the map and mark them with blue dots. Our investors prefer to work for Wall Street, mainly for commercial dinners, so we are looking for a rental market near restaurants. We managed to find 8 places open in the rental market. We created a map and tried to Extract some meaningful information. And marked with a red dot. Let's perform some basic explanatory data analysis. We removed the distance from Wall Street = 141 Duane St, 124 Chambers St, 77 Warren St, 291 Broadway. After excluding the unwanted selections, and from the information in the original data let us know the area of these places such as : 16 Maiden Ln = 1200 sq ft , 45 Nassau St = 6050 sq ft, 41 Nassau St = 1650 sq ft , Equitable Building = 16980 sq ft. Finally, we consider that our investors are the main factor in commercial dinner service. Too big a supply is not available. So Equitable Building is not going to choose. Too few places to supply can not meet the needs of the restaurant. So 16 Maiden L and 41 Nassau St are not going to choose. We are able to find suitable investors in neighboring Wall Street, Manhattan, New York City under the constraints of market supply. The result is 45 Nassau St = 6050 sq ft

In [27]:

```
df_rents.drop(df_rents.index[[0,1,2,3,4,6,7]])
```

Out[27]:

	name	Area	Level	sqft	lat	lng
5	45 Nassau St	Financial District, New York City, NY	Floor Lower Level	6050	40.70887	-74.0116

__Conclusion:__

Our analysis shows that although there are many early Chinese restaurants in Manhattan and New York City, there is no Chinese restaurant near Wall Street. Those location rentals candidates are then clustered to create a region of interest that contains the most candidate locations.

The result of all this is 2 areas, which contain the maximum number of potential Chinese restaurant locations based on the number and distance of existing venues. Of course, this does not mean that these areas are actually the best place for Chinese restaurants! The purpose of this analysis is to provide information only about the area near the Wall Street Center, but not to fill the existing Chinese restaurant. Therefore, the recommended area should only be considered as a starting point for a more detailed analysis, which may ultimately result in a location that not only has no nearby competition, but also considers other factors and satisfies all other relevant conditions.