In [2]:
```
#Question 1, week 1. Introduction:
#Services vs. Safety in San Fransisco
#I inted to investigate if neighborhood crime rates in San Fransisco are linked
to the
#various venues that are located in that neighborhood: as an example, if there a
re more
#bars or nightclubs in the neighborhood, will there be more reported incidents?
#This is relevant information for both landlords, sellers of real estate and fam
ilies
#with young kids, that are comparing different neighborhoods when relocating.
```

In [3]:
```
#Question 2, week 1:
#I will be using Foursquare location data of San Fransisco for the venue informa
tion in combination with the
#San Fransico Police Department data (https://data.sfgov.org/Public-Safety/Map-o
f-Police-Department-Incident-Reports-2018-to-/jq29-s5wp),
#both of these datasets shoud be up-to-date and are interlinked by latitude and
longitude coordinates; both datasets have also a 'neighborhood'-field. As the SF
PD database
#is huge, I will be limiting the dataset to incidents reported between 1st Jan 2
020 and 25th May 2020 or a shorter period if necessary (the Jan-May 2020
#period is already big with 46,382 incidents). The Foursquare data is accessed v
ia their developer site, and the SFPD data is available in CSV-format
#from their home page mentioned earlier.
#I intend to combine the datasets and cluster the data per neighborhood or per l
ocation, depending on which one is more parctical - the data amount is big, whic
h I expect to be
#a problem in practice. Thereafter I will compare the results: are there some ne
ighborhoods with higher amount of crime incidents, and if so, are there more of
some type
#venues in those neighborhoods compared to other neighborhoods?
```

In [4]:
```
import numpy as np  # useful for many scientific computing in Python
import pandas as pd # primary data structure library
import folium
!conda install -c conda-forge folium=0.5.0 --yes
print('Folium installed and imported!')
```

```
Collecting package metadata (current_repodata.json): done
Solving environment: failed with initial frozen solve. Retrying with flexible
solve.
Collecting package metadata (repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: /home/jupyterlab/conda/envs/python

  added / updated specs:
    - folium=0.5.0


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    altair-4.1.0               |             py_1         614 KB  conda-forge
    branca-0.4.1               |             py_0          26 KB  conda-forge
    brotlipy-0.7.0             |py36h8c4c3a4_1000         346 KB  conda-forge
    chardet-3.0.4              |py36h9f0ad1d_1006         188 KB  conda-forge
    cryptography-2.9.2         |   py36h45558ae_0         613 KB  conda-forge
    folium-0.5.0               |             py_0          45 KB  conda-forge
    pandas-1.0.4               |   py36h830a2c2_0        10.1 MB  conda-forge
    pysocks-1.7.1              |   py36h9f0ad1d_1          27 KB  conda-forge
    toolz-0.10.0               |             py_0          46 KB  conda-forge
    vincent-0.4.4              |             py_1          28 KB  conda-forge
    ------------------------------------------------------------
                                           Total:        12.0 MB

The following NEW packages will be INSTALLED:

  altair             conda-forge/noarch::altair-4.1.0-py_1
  attrs              conda-forge/noarch::attrs-19.3.0-py_0
  branca             conda-forge/noarch::branca-0.4.1-py_0
  brotlipy           conda-forge/linux-64::brotlipy-0.7.0-py36h8c4c3a4_1000
  chardet            conda-forge/linux-64::chardet-3.0.4-py36h9f0ad1d_1006
  cryptography       conda-forge/linux-64::cryptography-2.9.2-py36h45558ae_0
  entrypoints        conda-forge/linux-64::entrypoints-0.3-py36h9f0ad1d_1001
  folium             conda-forge/noarch::folium-0.5.0-py_0
  idna               conda-forge/noarch::idna-2.9-py_1
  importlib_metadata conda-forge/noarch::importlib_metadata-1.6.0-0
  jinja2             conda-forge/noarch::jinja2-2.11.2-pyh9f0ad1d_0
  jsonschema         conda-forge/linux-64::jsonschema-3.2.0-py36h9f0ad1d_1
  markupsafe         conda-forge/linux-64::markupsafe-1.1.1-py36h8c4c3a4_1
  pandas             conda-forge/linux-64::pandas-1.0.4-py36h830a2c2_0
  pyopenssl          conda-forge/noarch::pyopenssl-19.1.0-py_1
  pyrsistent         conda-forge/linux-64::pyrsistent-0.16.0-py36h8c4c3a4_0
  pysocks            conda-forge/linux-64::pysocks-1.7.1-py36h9f0ad1d_1
  pytz               conda-forge/noarch::pytz-2020.1-pyh9f0ad1d_0
  requests           conda-forge/noarch::requests-2.23.0-pyh8c360ce_2
  toolz              conda-forge/noarch::toolz-0.10.0-py_0
  urllib3            conda-forge/noarch::urllib3-1.25.9-py_0
  vincent            conda-forge/noarch::vincent-0.4.4-py_1



Downloading and Extracting Packages
pysocks-1.7.1        | 27 KB     | ##################################### | 10
0%
toolz-0.10.0         | 46 KB     | ##################################### | 10
0%
chardet-3.0.4        | 188 KB    | ##################################### | 10
0%
pandas-1.0.4         | 10.1 MB   | ##################################### | 10
0%
folium-0.5.0         | 45 KB     | ##################################### | 10
0%
```

In [5]:
```
import matplotlib.pyplot as plt #Other tools if need be
import pylab as pl
%matplotlib inline
```

In [6]:
```
#San Fransisco Police Department data downloaded from their site
#df_data_0 becomes the mother of all incident data
df_data_0 = pd.read_csv('Police_Department_Incident_Reports__2018_to_Present.csv
')
df_data_0.head()
```

Out[6]:

	Incident Datetime	Incident Date	Incident Time	Incident Year	Incident Day of Week	Report Datetime	Row ID	Incident ID	Incident Number	
0	2020/01/01 05:00:00 AM	2020/01/01	05:00	2020	Wednesday	2020/05/19 02:06:00 PM	92820009035	928200	200304799	20´
1	2020/01/01 05:00:00 AM	2020/01/01	05:00	2020	Wednesday	2020/05/19 02:06:00 PM	92820009027	928200	200304799	20´
2	2020/01/01 12:00:00 AM	2020/01/01	00:00	2020	Wednesday	2020/05/14 12:17:00 PM	92687768020	926877	200294366	20´
3	2020/01/01 08:00:00 AM	2020/01/01	08:00	2020	Wednesday	2020/05/12 08:23:00 AM	92627506304	926275	200289652	20´
4	2020/01/01 12:00:00 AM	2020/01/01	00:00	2020	Wednesday	2020/05/12 03:55:00 PM	92640306304	926403	200290756	20´

5 rows × 26 columns

In [7]:
```
#Cleaning
df_incidents=df_data_0
df_incidents.dropna()
df_incidents.drop(['Incident Datetime', 'Report Datetime', 'Incident Number', 'R
ow ID', 'Incident ID', 'CAD Number', 'Resolution', 'CNN', 'Supervisor District
'], axis=1, inplace=True)
df_incidents.rename(columns={'Latitude':'Y', 'Longitude':'X'}, inplace=True)
df_incidents.rename(columns={'Incident Category':'IncidentCategory'}, inplace=Tr
ue)
```

In [8]:
```
#Further cleaning, dropping incidents that have no coordinates or category
df_incidents.dropna(subset = ["X"], inplace=True)
df_incidents.dropna(subset = ["IncidentCategory"], inplace=True)
df_incidents.describe()
```

Out[8]:

	Incident Year	Incident Code	Y	X
count	44044.0	44044.000000	44044.000000	44044.000000
mean	2020.0	25105.604350	37.768422	-122.423967
std	0.0	25383.343795	0.024027	0.026174
min	2020.0	1001.000000	37.707988	-122.511295
25%	2020.0	6244.000000	37.753837	-122.434203
50%	2020.0	9020.000000	37.775161	-122.418043
75%	2020.0	51040.000000	37.785492	-122.407691
max	2020.0	75030.000000	37.829991	-122.363743

In [9]:
```
df_incidents.describe(include=['object'])
```

Out[9]:

	Incident Date	Incident Time	Incident Day of Week	Report Type Code	Report Type Description	Filed Online	IncidentCategory	Incident Subcategory	Incic Descript
count	44044	44044	44044	44044	44044	7282	44044	44044	44
unique	146	1438	7	4	6	1	50	69	
top	2020/01/10	00:00	Friday	II	Initial	True	Larceny Theft	Other	Theft, F Loc Veh >$
freq	461	1185	6788	35489	28878	7282	12035	6446	4

In [10]:
```
#basic statistics for future use
#looking at which neighborhoods have the most incidents
df_freq=df_incidents['Analysis Neighborhood'].value_counts().to_frame()
df_freq.reset_index()
df_freq.index.names = ['Neighborhood']
df_freq.rename(columns={'Analysis Neighborhood':'Incidents'},inplace=True)
df_freq.reset_index()
```

Out[10]:

	Neighborhood	Incidents
0	Mission	5018
1	Tenderloin	4547
2	South of Market	3539
3	Financial District/South Beach	3353
4	Bayview Hunters Point	3015
5	Western Addition	1606
6	Nob Hill	1492
7	Castro/Upper Market	1385
8	Hayes Valley	1310
9	Sunset/Parkside	1304
10	Marina	1194
11	Outer Richmond	1066
12	West of Twin Peaks	981
13	Bernal Heights	969
14	North Beach	958
15	Pacific Heights	896
16	Russian Hill	859
17	Chinatown	836
18	Potrero Hill	806
19	Excelsior	730
20	Haight Ashbury	710
21	Mission Bay	694
22	Outer Mission	630
23	Lone Mountain/USF	577
24	Visitacion Valley	562
25	Inner Richmond	545
26	Inner Sunset	535
27	Noe Valley	512
28	Portola	504
29	Lakeshore	471
30	Golden Gate Park	428
31	Oceanview/Merced/Ingleside	403
32	Japantown	400
33	Presidio Heights	308
34	Twin Peaks	255
35	Glen Park	235
36	Treasure Island	155
37	Presidio	79
38	Lincoln Park	67
39	McLaren Park	52
40	Seacliff	52

In [ ]:

In [11]:
```
#giving the neighborhoods lat & long coodrinates by taking the mean of the incid
ent coordinates -> more realistic point when added to Foursquare data later on
coordx=df_incidents.groupby('Analysis Neighborhood')['X'].mean()
coordy=df_incidents.groupby('Analysis Neighborhood')['Y'].mean()
coordy.reset_index()
coordy.index.names = ['Neighborhood']
coordy.reset_index()
coordx.reset_index()
coordx.index.names = ['Neighborhood']
coordx.reset_index()
```

Out[11]:

	Neighborhood	X
0	Bayview Hunters Point	-122.391006
1	Bernal Heights	-122.416461
2	Castro/Upper Market	-122.432535
3	Chinatown	-122.407250
4	Excelsior	-122.433301
5	Financial District/South Beach	-122.400700
6	Glen Park	-122.433356
7	Golden Gate Park	-122.468517
8	Haight Ashbury	-122.443592
9	Hayes Valley	-122.427426
10	Inner Richmond	-122.465429
11	Inner Sunset	-122.466270
12	Japantown	-122.433023
13	Lakeshore	-122.479421
14	Lincoln Park	-122.498369
15	Lone Mountain/USF	-122.448680
16	Marina	-122.436185
17	McLaren Park	-122.415694
18	Mission	-122.416759
19	Mission Bay	-122.394390
20	Nob Hill	-122.416173
21	Noe Valley	-122.431790
22	North Beach	-122.410756
23	Oceanview/Merced/Ingleside	-122.460612
24	Outer Mission	-122.442805
25	Outer Richmond	-122.492084
26	Pacific Heights	-122.432822
27	Portola	-122.407082
28	Potrero Hill	-122.396370
29	Presidio	-122.453483
30	Presidio Heights	-122.450340
31	Russian Hill	-122.420404
32	Seacliff	-122.485866
33	South of Market	-122.407608
34	Sunset/Parkside	-122.491852
35	Tenderloin	-122.414575
36	Treasure Island	-122.373184
37	Twin Peaks	-122.445956
38	Visitacion Valley	-122.412119
39	West of Twin Peaks	-122.460424
40	Western Addition	-122.428351

In [12]:
```
dfmerged=pd.merge(coordx, coordy, on='Neighborhood')
dfmerged1=pd.merge(df_freq, dfmerged, on='Neighborhood')
dfmerged1.reset_index()
```

Out[12]:

	Neighborhood	Incidents	X	Y
0	Mission	5018	-122.416759	37.761495
1	Tenderloin	4547	-122.414575	37.783317
2	South of Market	3539	-122.407608	37.778262
3	Financial District/South Beach	3353	-122.400700	37.788987
4	Bayview Hunters Point	3015	-122.391006	37.732513
5	Western Addition	1606	-122.428351	37.782520
6	Nob Hill	1492	-122.416173	37.789882
7	Castro/Upper Market	1385	-122.432535	37.763426
8	Hayes Valley	1310	-122.427426	37.775220
9	Sunset/Parkside	1304	-122.491852	37.750980
10	Marina	1194	-122.436185	37.800495
11	Outer Richmond	1066	-122.492084	37.777656
12	West of Twin Peaks	981	-122.460424	37.734708
13	Bernal Heights	969	-122.416461	37.741175
14	North Beach	958	-122.410756	37.804238
15	Pacific Heights	896	-122.432822	37.790590
16	Russian Hill	859	-122.420404	37.800047
17	Chinatown	836	-122.407250	37.796399
18	Potrero Hill	806	-122.396370	37.758925
19	Excelsior	730	-122.433301	37.719792
20	Haight Ashbury	710	-122.443592	37.769522
21	Mission Bay	694	-122.394390	37.772063
22	Outer Mission	630	-122.442805	37.720567
23	Lone Mountain/USF	577	-122.448680	37.777904
24	Visitacion Valley	562	-122.412119	37.712479
25	Inner Richmond	545	-122.465429	37.780476
26	Inner Sunset	535	-122.466270	37.760717
27	Noe Valley	512	-122.431790	37.749212
28	Portola	504	-122.407082	37.727338
29	Lakeshore	471	-122.479421	37.720122
30	Golden Gate Park	428	-122.468517	37.769677
31	Oceanview/Merced/Ingleside	403	-122.460612	37.717556
32	Japantown	400	-122.433023	37.785364
33	Presidio Heights	308	-122.450340	37.785386
34	Twin Peaks	255	-122.445956	37.751911
35	Glen Park	235	-122.433356	37.738301
36	Treasure Island	155	-122.373184	37.824233
37	Presidio	79	-122.453483	37.803069
38	Lincoln Park	67	-122.498369	37.781397
39	McLaren Park	52	-122.415694	37.719662
40	Seacliff	52	-122.485866	37.786440

In [13]:
```
dfmerged2=dfmerged1.reset_index()
dfmerged2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41 entries, 0 to 40
Data columns (total 4 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   Neighborhood  41 non-null     object
 1   Incidents     41 non-null     int64
 2   X             41 non-null     float64
 3   Y             41 non-null     float64
dtypes: float64(2), int64(1), object(1)
memory usage: 1.4+ KB
```

In [14]:
```
df_incidents['IncidentCategory'].value_counts().to_frame() #what indicent type i
s the most common
```

Out[14]:

	IncidentCategory
Larceny Theft	12035
Other Miscellaneous	3707
Malicious Mischief	3210
Burglary	2847
Non-Criminal	2773
Assault	2671
Motor Vehicle Theft	2360
Warrant	1432
Recovered Vehicle	1301
Robbery	1167
Fraud	1054
Drug Offense	1036
Missing Person	992
Lost Property	949
Offences Against The Family And Children	932
Suspicious Occ	896
Disorderly Conduct	792
Traffic Violation Arrest	512
Miscellaneous Investigation	462
Other	461
Other Offenses	377
Weapons Offense	323
Stolen Property	274
Weapons Carrying Etc	235
Forgery And Counterfeiting	152
Arson	148
Vandalism	119
Case Closure	117
Sex Offense	106
Courtesy Report	97
Traffic Collision	89
Embezzlement	69
Family Offense	64
Fire Report	60
Prostitution	50
Vehicle Impounded	34
Vehicle Misplaced	28
Suicide	26
Civil Sidewalks	25
Drug Violation	18
Suspicious	11
Liquor Laws	10
Rape	7
Human Trafficking (A), Commercial Sex Acts	4

In [15]:
```
#incidents on map
#San Francisco latitude and longitude values
latitude = 37.77
longitude = -122.42
sanfran_map = folium.Map(location=[latitude, longitude], zoom_start=12)
sanfran_map
```

Out[15]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [16]:
```
# instantiate a feature group for the incidents in the dataframe
incidents = folium.map.FeatureGroup()

# loop through the incidents and add each to the incidents feature group
for lat, lng, in zip(df_incidents.Y, df_incidents.X):
    incidents.add_child(
        folium.features.CircleMarker(
            [lat, lng],
            radius=5, # define how big you want the circle markers to be
            color='yellow',
            fill=True,
            fill_color='blue',
            fill_opacity=0.6
        )
    )
sanfran_map.add_child(incidents)
```

Out[16]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [17]:
```
from folium import plugins

#the first map has too many incidents to be of use, clustering necessary for usa
bility
sanfran_map = folium.Map(location = [latitude, longitude], zoom_start = 12)

# instantiate a mark cluster object for the incidents in the dataframe
incidents = plugins.MarkerCluster().add_to(sanfran_map)

# loop through the dataframe and add each data point to the mark cluster
for lat, lng, label, in zip(df_incidents.Y, df_incidents.X, df_incidents.Inciden
tCategory):
    folium.Marker(
        location=[lat, lng],
        icon=None,
        popup=label,
    ).add_to(incidents)

# display map
sanfran_map   #better, to be used in the presentation later on
```

Out[17]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [18]:
```
#moving on to Foursquare stuff - need a large dataset of venues, but if it prove
s too difficul to handle, then a smaller set will have to do.
#more libraries
import json
!conda install -c conda-forge geopy --yes
from geopy.geocoders import Nominatim # convert an address into latitude and lon
gitude values

import requests # library to handle requests
from pandas.io.json import json_normalize # tranform JSON file into a pandas dat
aframe
import matplotlib.cm as cm
import matplotlib.colors as colors
from sklearn.cluster import KMeans
```

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

## Package Plan ##

  environment location: /home/jupyterlab/conda/envs/python

  added / updated specs:
    - geopy


The following packages will be downloaded:

    package                    |            build
    ---------------------------|-----------------
    geographiclib-1.50         |             py_0          34 KB  conda-forge
    geopy-1.22.0               |     pyh9f0ad1d_0          63 KB  conda-forge
    ------------------------------------------------------------
                                           Total:          97 KB

The following NEW packages will be INSTALLED:

  geographiclib      conda-forge/noarch::geographiclib-1.50-py_0
  geopy              conda-forge/noarch::geopy-1.22.0-pyh9f0ad1d_0



Downloading and Extracting Packages
geopy-1.22.0         | 63 KB     | ##################################### | 10
0%
geographiclib-1.50   | 34 KB     | ##################################### | 10
0%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

In [19]:
```
CLIENT_ID = 'F1DSQIYSJRZYVLEU1DDSRFB0QPWDZQHCEQSGNEF3YWMM44SQ' # your Foursquare
ID
CLIENT_SECRET = 'QREISBS33NHTVRGIM3EIEUSPPA11I1K53HSH1HLHVOTFHK0Y' # your Foursq
uare Secret
VERSION = '20180604'
LIMIT = 30
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: F1DSQIYSJRZYVLEU1DDSRFB0QPWDZQHCEQSGNEF3YWMM44SQ
CLIENT_SECRET:QREISBS33NHTVRGIM3EIEUSPPA11I1K53HSH1HLHVOTFHK0Y
```

In [20]:
```
address = '3350 Mission St, San Francisco, USA'     #Safeway at Mission Street, seems to be quite an active place
geolocator = Nominatim(user_agent="SanFran_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
37.743217349999995 -122.42247150547838
```

In [21]:
```
latitude = 37.743217349999995 #might be unnecessary
longitude = -122.42247150547838
SanFrancisco_map = folium.Map(location=[latitude, longitude], zoom_start=12)
SanFrancisco_map
```

Out[21]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [22]:
```
for lat, lng, neighborhood in zip(dfmerged2['Y'], dfmerged2['X'], dfmerged2['Nei
ghborhood']):   #the neighborhoods on a map. Might be unnecessary
    label = '{}'.format(neighborhood)
    label = folium.Popup(label, parse_html=True)
    folium.CircleMarker(
        [lat, lng],
        radius=5,
        popup=label,
        color='blue',
        fill=True,
        fill_color='#3186cc',
        fill_opacity=0.7,
        parse_html=False).add_to(SanFrancisco_map)
SanFrancisco_map
```

Out[22]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [23]:
```
neighborhood_latitude = dfmerged2.loc[0, 'Y'] # neighborhood latitude value
neighborhood_longitude = dfmerged2.loc[0, 'X'] # neighborhood longitude value

neighborhood_name = dfmerged2.loc[0, 'Neighborhood'] # neighborhood name

print('Latitude and longitude values of {} are {}, {}.'.format(neighborhood_nam
e,
                                                               neighborhood_lati
tude,
                                                               neighborhood_long
itude))
```

```
Latitude and longitude values of Mission are 37.76149516319345, -122.416758776
5861.
```

In [28]:
```
LIMIT=200
radius=750
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
```

Out[28]:
```
'https://api.foursquare.com/v2/venues/explore?&client_id=F1DSQIYSJRZYVLEU1DDSRFB0QPWDZQHCEQSGNEF3YWMM44SQ&client_secret=QREISBS33NHTVRGIM3EIEUSPPA11I1K53HSH1HLHVOTFHK0Y&v=20180604&ll=37.76149516319345,-122.4167587765861&radius=750&limit=200'
```

In [29]:
```
results = requests.get(url).json()
results
```

Out[29]:

```
{'meta': {'code': 200, 'requestId': '5ed2481cdf2774001be43ac6'},
 'response': {'suggestedFilters': {'header': 'Tap to show:',
   'filters': [{'name': '$-$$$$', 'key': 'price'}]},
  'headerLocation': 'Mission District',
  'headerFullLocation': 'Mission District, San Francisco',
  'headerLocationGranularity': 'neighborhood',
  'totalResults': 178,
  'suggestedBounds': {'ne': {'lat': 37.768245169943455,
    'lng': -122.40823653113544},
   'sw': {'lat': 37.75474515644345, 'lng': -122.42528102203674}},
  'groups': [{'type': 'Recommended Places',
    'name': 'recommended',
    'items': [{'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '510d68e2e4b0873bc2f5be75',
       'name': 'CrossFit Alinea',
       'location': {'address': '674 S Van Ness Ave',
        'lat': 37.76245764000935,
        'lng': -122.41741647427942,
        'labeledLatLngs': [{'label': 'display',
          'lat': 37.76245764000935,
          'lng': -122.41741647427942},
         {'label': 'entrance', 'lat': 37.762363, 'lng': -122.417369}],
        'distance': 121,
        'postalCode': '94110',
        'cc': 'US',
        'city': 'San Francisco',
        'state': 'CA',
        'country': 'United States',
        'formattedAddress': ['674 S Van Ness Ave',
         'San Francisco, CA 94110',
         'United States']},
       'categories': [{'id': '4bf58dd8d48988d176941735',
         'name': 'Gym',
         'pluralName': 'Gyms',
         'shortName': 'Gym',
         'icon': {'prefix': 'https://ss3.4sqi.net/img/categories_v2/building/g
ym_',
          'suffix': '.png'},
         'primary': True}],
       'photos': {'count': 0, 'groups': []}},
      'referralId': 'e-0-510d68e2e4b0873bc2f5be75-0'},
     {'reasons': {'count': 0,
       'items': [{'summary': 'This spot is popular',
         'type': 'general',
         'reasonName': 'globalInteractionReason'}]},
      'venue': {'id': '4466458af964a5203b331fe3',
       'name': 'ODC Dance Commons',
       'location': {'address': '351 Shotwell St',
        'crossStreet': 'btwn 17th St & 18th St',
        'lat': 37.762705243388325,
        'lng': -122.41597522734932,
        'labeledLatLngs': [{'label': 'display',
          'lat': 37.762705243388325,
          'lng': -122.41597522734932}],
        'distance': 151,
        'postalCode': '94110',
        'cc': 'US',
        'city': 'San Francisco',
        'state': 'CA',
        'country': 'United States',
        'formattedAddress': ['351 Shotwell St (btwn 17th St & 18th St)',
         'San Francisco, CA 94110',
         'United States']},
       'categories': [{'id': '4bf58dd8d48988d134941735',
         'name': 'Dance Studio',
         'pluralName': 'Dance Studios',
```

In [30]:
```
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 [31]:
```
venues = results['response']['groups'][0]['items']

nearby_venues = json_normalize(venues) # flatten JSON

# filter columns
filtered_columns = ['venue.name', 'venue.categories', 'venue.location.lat', 'ven
ue.location.lng']
nearby_venues =nearby_venues.loc[:, filtered_columns]

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

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

nearby_venues.head()
```

```
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel_launc
her.py:3: FutureWarning: pandas.io.json.json_normalize is deprecated, use pand
as.json_normalize instead
  This is separate from the ipykernel package so we can avoid doing imports un
til
```

Out[31]:

	name	categories	lat	lng
0	CrossFit Alinea	Gym	37.762458	-122.417416
1	ODC Dance Commons	Dance Studio	37.762705	-122.415975
2	Royal Cuckoo Market	Market	37.760330	-122.418651
3	ODC Theater	Theater	37.763487	-122.416476
4	Mission: Comics & Art	Comic Shop	37.760978	-122.419408

In [32]:
```
print('{} venues were returned by Foursquare.'.format(nearby_venues.shape[0]))
```

```
100 venues were returned by Foursquare.
```

In [33]:
```
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)

Mission_Venues = getNearbyVenues(names=dfmerged2['Neighborhood'],
                                   latitudes=dfmerged2['Y'],
                                   longitudes=dfmerged2['X']
                                  )
```

```
Mission
Tenderloin
South of Market
Financial District/South Beach
Bayview Hunters Point
Western Addition
Nob Hill
Castro/Upper Market
Hayes Valley
Sunset/Parkside
Marina
Outer Richmond
West of Twin Peaks
Bernal Heights
North Beach
Pacific Heights
Russian Hill
Chinatown
Potrero Hill
Excelsior
Haight Ashbury
Mission Bay
Outer Mission
Lone Mountain/USF
Visitacion Valley
Inner Richmond
Inner Sunset
Noe Valley
Portola
Lakeshore
Golden Gate Park
Oceanview/Merced/Ingleside
Japantown
Presidio Heights
Twin Peaks
Glen Park
Treasure Island
Presidio
Lincoln Park
McLaren Park
Seacliff
```

In [34]:
```
Mission_Venues.groupby('Neighborhood').count()
```

Out[34]:

Neighborhood	Neighborhood Latitude	Neighborhood Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
Bayview Hunters Point	29	29	29	29	29	29
Bernal Heights	45	45	45	45	45	45
Castro/Upper Market	92	92	92	92	92	92
Chinatown	99	99	99	99	99	99
Excelsior	29	29	29	29	29	29
Financial District/South Beach	100	100	100	100	100	100
Glen Park	24	24	24	24	24	24
Golden Gate Park	73	73	73	73	73	73
Haight Ashbury	32	32	32	32	32	32
Hayes Valley	93	93	93	93	93	93
Inner Richmond	75	75	75	75	75	75
Inner Sunset	48	48	48	48	48	48
Japantown	98	98	98	98	98	98
Lakeshore	18	18	18	18	18	18
Lincoln Park	18	18	18	18	18	18
Lone Mountain/USF	31	31	31	31	31	31
Marina	100	100	100	100	100	100
McLaren Park	6	6	6	6	6	6
Mission	86	86	86	86	86	86
Mission Bay	42	42	42	42	42	42
Nob Hill	100	100	100	100	100	100
Noe Valley	82	82	82	82	82	82
North Beach	95	95	95	95	95	95
Oceanview/Merced /Ingleside	4	4	4	4	4	4
Outer Mission	37	37	37	37	37	37
Outer Richmond	45	45	45	45	45	45
Pacific Heights	52	52	52	52	52	52
Portola	36	36	36	36	36	36
Potrero Hill	33	33	33	33	33	33
Presidio	26	26	26	26	26	26
Presidio Heights	47	47	47	47	47	47
Russian Hill	73	73	73	73	73	73
Seacliff	21	21	21	21	21	21
South of Market	56	56	56	56	56	56
Sunset/Parkside	26	26	26	26	26	26
Tenderloin	100	100	100	100	100	100
Treasure Island	13	13	13	13	13	13
Twin Peaks	7	7	7	7	7	7
Visitacion Valley	4	4	4	4	4	4
West of Twin Peaks	5	5	5	5	5	5
Western Addition	53	53	53	53	53	53

In [35]:
```
print('There are {} unique categories.'.format(len(Mission_Venues['Venue Categor
y'].unique())))
```

There are 307 unique categories.

In [36]:
```
# one hot encoding
Mission_onehot = pd.get_dummies(Mission_Venues[['Venue Category']], prefix="", p
refix_sep="")

# add neighborhood column back to dataframe
Mission_onehot['Neighborhood'] = Mission_Venues['Neighborhood']

# define a list of column names
cols = Mission_onehot.columns.tolist()
cols

# move the column name to the beggining
cols.insert(0, cols.pop(cols.index('Neighborhood')))
cols

#then use .reindex() function to reorder
Mission_onehot = Mission_onehot.reindex(columns= cols)

Mission_onehot.head(85)
```

Out[36]:

	Neighborhood	Accessories Store	Adult Boutique	African Restaurant	Alternative Healer	American Restaurant	Amphitheater	Animal Shelter	Antiq Sh
0	Mission	0	0	0	0	0	0	0	
1	Mission	0	0	0	0	0	0	0	
2	Mission	0	0	0	0	0	0	0	
3	Mission	0	0	0	0	0	0	0	
4	Mission	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	
80	Mission	0	0	0	0	1	0	0	
81	Mission	0	0	0	0	0	0	0	
82	Mission	0	0	0	0	0	0	0	
83	Mission	0	0	0	0	0	0	0	
84	Mission	0	0	0	0	0	0	0	

85 rows × 307 columns

In [37]:
```
Sanfran_grouped = Mission_onehot.groupby('Neighborhood').mean().reset_index()
Sanfran_grouped
```

Out[37]:

	Neighborhood	Accessories Store	Adult Boutique	African Restaurant	Alternative Healer	American Restaurant	Amphitheater	Animal Shelter
0	Bayview Hunters Point	0.000000	0.000000	0.034483	0.000000	0.000000	0.000000	0.000000
1	Bernal Heights	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
2	Castro/Upper Market	0.000000	0.010870	0.000000	0.000000	0.010870	0.000000	0.000000
3	Chinatown	0.000000	0.000000	0.000000	0.000000	0.010101	0.000000	0.000000
4	Excelsior	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
5	Financial District/South Beach	0.010000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
6	Glen Park	0.000000	0.000000	0.000000	0.041667	0.000000	0.000000	0.000000
7	Golden Gate Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
8	Haight Ashbury	0.031250	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
9	Hayes Valley	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
10	Inner Richmond	0.000000	0.000000	0.000000	0.000000	0.013333	0.000000	0.000000
11	Inner Sunset	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
12	Japantown	0.000000	0.000000	0.000000	0.000000	0.010204	0.000000	0.000000
13	Lakeshore	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
14	Lincoln Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
15	Lone Mountain/USF	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
16	Marina	0.000000	0.000000	0.000000	0.010000	0.010000	0.000000	0.000000
17	McLaren Park	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
18	Mission	0.000000	0.011628	0.000000	0.000000	0.023256	0.000000	0.000000
19	Mission Bay	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
20	Nob Hill	0.000000	0.010000	0.000000	0.000000	0.010000	0.000000	0.000000
21	Noe Valley	0.000000	0.000000	0.000000	0.000000	0.012195	0.000000	0.000000
22	North Beach	0.010526	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
23	Oceanview/Merced /Ingleside	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
24	Outer Mission	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25	Outer Richmond	0.000000	0.000000	0.000000	0.000000	0.022222	0.000000	0.000000
26	Pacific Heights	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
27	Portola	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
28	Potrero Hill	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
29	Presidio	0.000000	0.000000	0.000000	0.000000	0.000000	0.038462	0.000000
30	Presidio Heights	0.000000	0.000000	0.000000	0.000000	0.042553	0.000000	0.021277
31	Russian Hill	0.000000	0.000000	0.013699	0.000000	0.013699	0.000000	0.000000
32	Seacliff	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
33	South of Market	0.017857	0.000000	0.000000	0.000000	0.035714	0.000000	0.000000
34	Sunset/Parkside	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
35	Tenderloin	0.000000	0.000000	0.000000	0.000000	0.020000	0.000000	0.000000
36	Treasure Island	0.000000	0.000000	0.000000	0.000000	0.076923	0.000000	0.000000
37	Twin Peaks	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
38	Visitacion Valley	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
[illegible]	West of Twin	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

In [ ]:
```
num_top_venues = 20

for hood in Sanfran_grouped['Neighborhood']:
    print("----"+hood+"----")
    temp = Sanfran_grouped[Sanfran_grouped['Neighborhood'] == hood].T.reset_inde
x()
    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')
```

In [38]:
```
Sanfran_groupedcount = Mission_onehot.groupby('Neighborhood').sum().reset_index
()
Sanfran_groupedcount
```

Out[38]:

	Neighborhood	Accessories Store	Adult Boutique	African Restaurant	Alternative Healer	American Restaurant	Amphitheater	Animal Shelter	A
0	Bayview Hunters Point	0	0	1	0	0	0	0	
1	Bernal Heights	0	0	0	0	0	0	0	
2	Castro/Upper Market	0	1	0	0	1	0	0	
3	Chinatown	0	0	0	0	1	0	0	
4	Excelsior	0	0	0	0	0	0	0	
5	Financial District/South Beach	1	0	0	0	0	0	0	
6	Glen Park	0	0	0	1	0	0	0	
7	Golden Gate Park	0	0	0	0	0	0	0	
8	Haight Ashbury	1	0	0	0	0	0	0	
9	Hayes Valley	0	0	0	0	0	0	0	
10	Inner Richmond	0	0	0	0	1	0	0	
11	Inner Sunset	0	0	0	0	0	0	0	
12	Japantown	0	0	0	0	1	0	0	
13	Lakeshore	0	0	0	0	0	0	0	
14	Lincoln Park	0	0	0	0	0	0	0	
15	Lone Mountain/USF	0	0	0	0	0	0	0	
16	Marina	0	0	0	1	1	0	0	
17	McLaren Park	0	0	0	0	0	0	0	
18	Mission	0	1	0	0	2	0	0	
19	Mission Bay	0	0	0	0	0	0	0	
20	Nob Hill	0	1	0	0	1	0	0	
21	Noe Valley	0	0	0	0	1	0	0	
22	North Beach	1	0	0	0	0	0	0	
23	Oceanview/Merced/Ingleside	0	0	0	0	0	0	0	
24	Outer Mission	0	0	0	0	0	0	0	
25	Outer Richmond	0	0	0	0	1	0	0	
26	Pacific Heights	0	0	0	0	0	0	0	
27	Portola	0	0	0	0	0	0	0	
28	Potrero Hill	0	0	0	0	0	0	0	
29	Presidio	0	0	0	0	0	1	0	
30	Presidio Heights	0	0	0	0	2	0	1	
31	Russian Hill	0	0	1	0	1	0	0	
32	Seacliff	0	0	0	0	0	0	0	
33	South of Market	1	0	0	0	2	0	0	
34	Sunset/Parkside	0	0	0	0	0	0	0	
35	Tenderloin	0	0	0	0	2	0	0	
36	Treasure Island	0	0	0	0	1	0	0	
37	Twin Peaks	0	0	0	0	0	0	0	
38	Visitacion Valley	0	0	0	0	0	0	0	
[illegible]	West of Twin	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	

In [39]:
```
num_top_venues = 20

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

```
----Bayview Hunters Point----
                           venue  freq
0   Southern / Soul Food Restaurant   4.0
1              Light Rail Station   3.0
2              Mexican Restaurant   2.0
3                     Bus Station   2.0
4           Thrift / Vintage Store   1.0
5                    Home Service   1.0
6                      Taco Place   1.0
7                          Bakery   1.0
8                            Park   1.0
9                       BBQ Joint   1.0
10             Chinese Restaurant   1.0
11                        Theater   1.0
12         Health & Beauty Service   1.0
13                         Market   1.0
14                          Plaza   1.0
15                         Garden   1.0
16                            Gym   1.0
17            Fried Chicken Joint   1.0
18                           Café   1.0
19            Dumpling Restaurant   1.0


----Bernal Heights----
                   venue  freq
0                  Trail   3.0
1            Coffee Shop   3.0
2             Playground   3.0
3     Italian Restaurant   2.0
4                 Bakery   2.0
5            Pizza Place   2.0
6            Art Gallery   2.0
7           Gourmet Shop   2.0
8              Locksmith   1.0
9        Sushi Restaurant   1.0
10      Indian Restaurant   1.0
11         Cosmetics Shop   1.0
12    Peruvian Restaurant   1.0
13           Cocktail Bar   1.0
14  New American Restaurant   1.0
15                Gay Bar   1.0
16                   Park   1.0
17              Gift Shop   1.0
18           Burger Joint   1.0
19   Caribbean Restaurant   1.0


----Castro/Upper Market----
                      venue  freq
0                   Gay Bar   9.0
1               Coffee Shop   4.0
2           Thai Restaurant   4.0
3         Indian Restaurant   3.0
4   New American Restaurant   3.0
5             Deli / Bodega   2.0
6        Seafood Restaurant   2.0
7                    Bakery   2.0
8              Dessert Shop   2.0
9                 Juice Bar   2.0
10          Asian Restaurant   2.0
11                       Gym   2.0
12           Sandwich Place   2.0
13      Japanese Restaurant   2.0
14                 Wine Bar   2.0
15  Mediterranean Restaurant   2.0
16             Cocktail Bar   1.0
17           History Museum   1.0
18           Ice Cream Shop   1.0
```

In [40]:
```
#looks like the statistics would not point out to a larger amount of bars in Mis
sion or Tenderloin. A closer look at the crime types on those areas might be use
ful
```

In [41]:
```
df_neighborhoods_incidents=df_incidents[['Analysis Neighborhood', 'IncidentCateg
ory']]
df_neighborhoods_incidents.head()
```

Out[41]:

	Analysis Neighborhood	IncidentCategory
0	Financial District/South Beach	Fraud
1	Financial District/South Beach	Fraud
2	Tenderloin	Miscellaneous Investigation
3	Russian Hill	Larceny Theft
4	Marina	Larceny Theft

In [42]:
```
tool=df_neighborhoods_incidents.groupby('Analysis Neighborhood')
tool1=tool.get_group('Mission')
tool1.describe()
```

Out[42]:

	Analysis Neighborhood	IncidentCategory
count	5018	5018
unique	1	46
top	Mission	Larceny Theft
freq	5018	1111

In [43]:
```
tool=df_neighborhoods_incidents.groupby('Analysis Neighborhood')
tool1=tool.get_group('Tenderloin')
tool1.describe()
```

Out[43]:

	Analysis Neighborhood	IncidentCategory
count	4547	4547
unique	1	42
top	Tenderloin	Larceny Theft
freq	4547	713

In [44]:
```
tool=df_neighborhoods_incidents.groupby('Analysis Neighborhood')
tool1=tool.get_group('Western Addition')
tool1.describe()
```

Out[44]:

	Analysis Neighborhood	IncidentCategory
count	1606	1606
unique	1	38
top	Western Addition	Larceny Theft
freq	1606	533

In [ ]:
```
#looking closer at which incidents happens where with onehot
```

In [45]:
```
# one hot encoding
Incident_onehot = pd.get_dummies(df_incidents[['IncidentCategory']], prefix="",
prefix_sep="")

# add neighborhood column back to dataframe
Incident_onehot['Analysis Neighborhood'] = df_incidents['Analysis Neighborhood']

# define a list of column names
cols1 = Incident_onehot.columns.tolist()
cols1

# move the column name to the beggining
cols1.insert(0, cols1.pop(cols1.index('Analysis Neighborhood')))
cols1

#then use .reindex() function to reorder
Incident_onehot = Incident_onehot.reindex(columns= cols1)

Incident_onehot.head(85)
```

Out[45]:

	Analysis Neighborhood	Arson	Assault	Burglary	Case Closure	Civil Sidewalks	Courtesy Report	Disorderly Conduct	Drug Offense	Drug Violation
0	Financial District/South Beach	0	0	0	0	0	0	0	0	0
1	Financial District/South Beach	0	0	0	0	0	0	0	0	0
2	Tenderloin	0	0	0	0	0	0	0	0	0
3	Russian Hill	0	0	0	0	0	0	0	0	0
4	Marina	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...
84	Bayview Hunters Point	0	0	0	0	0	0	0	0	0
85	Marina	0	0	0	0	0	0	0	0	0
86	South of Market	0	0	0	0	0	0	0	0	0
87	Tenderloin	0	0	0	0	0	0	0	0	0
88	Mission Bay	0	0	0	0	0	0	0	0	0

85 rows × 51 columns

In [ ]:

In [46]:
```
Incident_groupedmean = Incident_onehot.groupby('Analysis Neighborhood').mean().r
eset_index()
Incident_groupedmean
```

Out[46]:

	Analysis Neighborhood	Arson	Assault	Burglary	Case Closure	Civil Sidewalks	Courtesy Report	Disorderly Conduct	Drug Offense
0	Bayview Hunters Point	0.003648	0.090879	0.036153	0.001990	0.000000	0.003648	0.032172	0.009950
1	Bernal Heights	0.001032	0.062951	0.115583	0.001032	0.000000	0.000000	0.018576	0.006192
2	Castro/Upper Market	0.004332	0.055596	0.090975	0.004332	0.004332	0.000722	0.010108	0.011552
3	Chinatown	0.008373	0.052632	0.075359	0.004785	0.000000	0.014354	0.014354	0.008373
4	Excelsior	0.001370	0.060274	0.052055	0.000000	0.000000	0.001370	0.019178	0.006849
5	Financial District/South Beach	0.002088	0.064122	0.082016	0.002684	0.000596	0.001789	0.014912	0.007754
6	Glen Park	0.000000	0.068085	0.089362	0.000000	0.000000	0.000000	0.000000	0.004255
7	Golden Gate Park	0.021028	0.035047	0.042056	0.016355	0.000000	0.000000	0.000000	0.030374
8	Haight Ashbury	0.005634	0.035211	0.091549	0.007042	0.014085	0.000000	0.021127	0.030986
9	Hayes Valley	0.008397	0.037405	0.091603	0.002290	0.000000	0.000000	0.011450	0.004580
10	Inner Richmond	0.001835	0.031193	0.064220	0.001835	0.000000	0.012844	0.009174	0.005505
11	Inner Sunset	0.000000	0.059813	0.076636	0.005607	0.000000	0.000000	0.028037	0.000000
12	Japantown	0.002500	0.040000	0.052500	0.000000	0.000000	0.002500	0.007500	0.000000
13	Lakeshore	0.008493	0.033970	0.025478	0.002123	0.000000	0.006369	0.025478	0.002123
14	Lincoln Park	0.000000	0.000000	0.029851	0.000000	0.000000	0.000000	0.000000	0.000000
15	Lone Mountain/USF	0.000000	0.046794	0.057192	0.003466	0.000000	0.000000	0.013865	0.008666
16	Marina	0.000838	0.031826	0.098827	0.001675	0.001675	0.000000	0.014238	0.007538
17	McLaren Park	0.019231	0.038462	0.057692	0.000000	0.000000	0.000000	0.000000	0.000000
18	Mission	0.003189	0.069350	0.055201	0.000598	0.000598	0.002391	0.024512	0.017736
19	Mission Bay	0.002882	0.046110	0.072046	0.002882	0.000000	0.014409	0.008646	0.012968
20	Nob Hill	0.006032	0.052279	0.103217	0.000670	0.000670	0.000000	0.015416	0.012735
21	Noe Valley	0.000000	0.013672	0.121094	0.000000	0.000000	0.000000	0.001953	0.001953
22	North Beach	0.004175	0.049061	0.073069	0.005219	0.000000	0.000000	0.011482	0.019833
23	Oceanview/Merced /Ingleside	0.000000	0.052109	0.029777	0.000000	0.000000	0.000000	0.024814	0.000000
24	Outer Mission	0.004762	0.047619	0.041270	0.000000	0.000000	0.004762	0.025397	0.004762
25	Outer Richmond	0.002814	0.041276	0.047842	0.002814	0.000000	0.002814	0.016886	0.005629
26	Pacific Heights	0.001116	0.021205	0.102679	0.001116	0.000000	0.003348	0.008929	0.012277
27	Portola	0.001984	0.055556	0.055556	0.000000	0.000000	0.005952	0.019841	0.005952
28	Potrero Hill	0.002481	0.033499	0.074442	0.000000	0.000000	0.000000	0.009926	0.012407
29	Presidio	0.000000	0.063291	0.037975	0.012658	0.000000	0.000000	0.025316	0.000000
30	Presidio Heights	0.006494	0.029221	0.071429	0.000000	0.000000	0.000000	0.019481	0.000000
31	Russian Hill	0.002328	0.040745	0.101281	0.002328	0.001164	0.000000	0.010477	0.003492
32	Seacliff	0.000000	0.019231	0.115385	0.000000	0.000000	0.000000	0.000000	0.000000
33	South of Market	0.004238	0.082509	0.061599	0.008194	0.000000	0.001130	0.021192	0.056796
34	Sunset/Parkside	0.003067	0.047546	0.033742	0.003067	0.000000	0.003067	0.019172	0.003067
35	Tenderloin	0.001539	0.093688	0.032549	0.001539	0.000000	0.001759	0.015615	0.102925
36	Treasure Island	0.000000	0.116129	0.070968	0.000000	0.000000	0.000000	0.012903	0.000000
37	Twin Peaks	0.000000	0.011765	0.098039	0.003922	0.000000	0.000000	0.007843	0.011765
38	Visitacion Valley	0.007117	0.096085	0.037367	0.005338	0.000000	0.003559	0.023132	0.001779

In [47]:
```
Incident_groupedcount = Incident_onehot.groupby('Analysis Neighborhood').sum().r
eset_index()
Incident_groupedcount
```

Out[47]:

	Analysis Neighborhood	Arson	Assault	Burglary	Case Closure	Civil Sidewalks	Courtesy Report	Disorderly Conduct	Drug Offense	Dr Violati
0	Bayview Hunters Point	11.0	274.0	109.0	6.0	0.0	11.0	97.0	30.0	[illegible]
1	Bernal Heights	1.0	61.0	112.0	1.0	0.0	0.0	18.0	6.0	[illegible]
2	Castro/Upper Market	6.0	77.0	126.0	6.0	6.0	1.0	14.0	16.0	[illegible]
3	Chinatown	7.0	44.0	63.0	4.0	0.0	12.0	12.0	7.0	[illegible]
4	Excelsior	1.0	44.0	38.0	0.0	0.0	1.0	14.0	5.0	[illegible]
5	Financial District/South Beach	7.0	215.0	275.0	9.0	2.0	6.0	50.0	26.0	[illegible]
6	Glen Park	0.0	16.0	21.0	0.0	0.0	0.0	0.0	1.0	[illegible]
7	Golden Gate Park	9.0	15.0	18.0	7.0	0.0	0.0	0.0	13.0	[illegible]
8	Haight Ashbury	4.0	25.0	65.0	5.0	10.0	0.0	15.0	22.0	[illegible]
9	Hayes Valley	11.0	49.0	120.0	3.0	0.0	0.0	15.0	6.0	[illegible]
10	Inner Richmond	1.0	17.0	35.0	1.0	0.0	7.0	5.0	3.0	[illegible]
11	Inner Sunset	0.0	32.0	41.0	3.0	0.0	0.0	15.0	0.0	[illegible]
12	Japantown	1.0	16.0	21.0	0.0	0.0	1.0	3.0	0.0	[illegible]
13	Lakeshore	4.0	16.0	12.0	1.0	0.0	3.0	12.0	1.0	[illegible]
14	Lincoln Park	0.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0	[illegible]
15	Lone Mountain/USF	0.0	27.0	33.0	2.0	0.0	0.0	8.0	5.0	[illegible]
16	Marina	1.0	38.0	118.0	2.0	2.0	0.0	17.0	9.0	[illegible]
17	McLaren Park	1.0	2.0	3.0	0.0	0.0	0.0	0.0	0.0	[illegible]
18	Mission	16.0	348.0	277.0	3.0	3.0	12.0	123.0	89.0	[illegible]
19	Mission Bay	2.0	32.0	50.0	2.0	0.0	10.0	6.0	9.0	[illegible]
20	Nob Hill	9.0	78.0	154.0	1.0	1.0	0.0	23.0	19.0	[illegible]
21	Noe Valley	0.0	7.0	62.0	0.0	0.0	0.0	1.0	1.0	[illegible]
22	North Beach	4.0	47.0	70.0	5.0	0.0	0.0	11.0	19.0	[illegible]
23	Oceanview/Merced /Ingleside	0.0	21.0	12.0	0.0	0.0	0.0	10.0	0.0	[illegible]
24	Outer Mission	3.0	30.0	26.0	0.0	0.0	3.0	16.0	3.0	[illegible]
25	Outer Richmond	3.0	44.0	51.0	3.0	0.0	3.0	18.0	6.0	[illegible]
26	Pacific Heights	1.0	19.0	92.0	1.0	0.0	3.0	8.0	11.0	[illegible]
27	Portola	1.0	28.0	28.0	0.0	0.0	3.0	10.0	3.0	[illegible]
28	Potrero Hill	2.0	27.0	60.0	0.0	0.0	0.0	8.0	10.0	[illegible]
29	Presidio	0.0	5.0	3.0	1.0	0.0	0.0	2.0	0.0	[illegible]
30	Presidio Heights	2.0	9.0	22.0	0.0	0.0	0.0	6.0	0.0	[illegible]
31	Russian Hill	2.0	35.0	87.0	2.0	1.0	0.0	9.0	3.0	[illegible]
32	Seacliff	0.0	1.0	6.0	0.0	0.0	0.0	0.0	0.0	[illegible]
33	South of Market	15.0	292.0	218.0	29.0	0.0	4.0	75.0	201.0	[illegible]
34	Sunset/Parkside	4.0	62.0	44.0	4.0	0.0	4.0	25.0	4.0	[illegible]
35	Tenderloin	7.0	426.0	148.0	7.0	0.0	8.0	71.0	468.0	[illegible]
36	Treasure Island	0.0	18.0	11.0	0.0	0.0	0.0	2.0	0.0	[illegible]
37	Twin Peaks	0.0	3.0	25.0	1.0	0.0	0.0	2.0	3.0	[illegible]
38	Visitacion Valley	4.0	54.0	21.0	3.0	0.0	2.0	13.0	1.0	[illegible]

In [48]:
```
num_top_incidents = 20

for hood in Incident_groupedmean['Analysis Neighborhood']:
    print("----"+hood+"----")
    temp = Incident_groupedmean[Incident_groupedmean['Analysis Neighborhood'] == 
hood].T.reset_index()
    temp.columns = ['type','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')
```

```
----Bayview Hunters Point----
                                      type  freq
0                            Larceny Theft  0.16
1                      Other Miscellaneous  0.09
2                                  Assault  0.09
3                        Recovered Vehicle  0.08
4                      Motor Vehicle Theft  0.07
5                       Malicious Mischief  0.07
6   Offences Against The Family And Children  0.05
7                                 Burglary  0.04
8                                  Robbery  0.04
9                             Non-Criminal  0.04
10                                 Warrant  0.03
11                      Disorderly Conduct  0.03
12                          Missing Person  0.03
13                          Suspicious Occ  0.02
14                         Weapons Offense  0.02
15                                   Fraud  0.02
16                            Drug Offense  0.01
17                    Weapons Carrying Etc  0.01
18                          Other Offenses  0.01
19                           Lost Property  0.01
```

```
----Bernal Heights----
                                      type  freq
0                            Larceny Theft  0.21
1                                 Burglary  0.12
2                      Other Miscellaneous  0.09
3                       Malicious Mischief  0.08
4                             Non-Criminal  0.07
5                      Motor Vehicle Theft  0.07
6                                  Assault  0.06
7                           Missing Person  0.04
8   Offences Against The Family And Children  0.03
9                        Recovered Vehicle  0.03
10                      Disorderly Conduct  0.02
11                Traffic Violation Arrest  0.02
12                          Suspicious Occ  0.02
13                                 Robbery  0.02
14                                   Fraud  0.02
15                                 Warrant  0.02
16                           Lost Property  0.01
17                               Vandalism  0.01
18           Miscellaneous Investigation  0.01
19                    Weapons Carrying Etc  0.01
```

```
----Castro/Upper Market----
                                      type  freq
0                            Larceny Theft  0.29
1                      Other Miscellaneous  0.09
2                                 Burglary  0.09
3                       Malicious Mischief  0.08
4                                  Assault  0.06
5                             Non-Criminal  0.06
6                      Motor Vehicle Theft  0.05
7                                    Fraud  0.04
8                                  Warrant  0.04
9                        Recovered Vehicle  0.02
10  Offences Against The Family And Children  0.02
11                          Suspicious Occ  0.02
12                          Missing Person  0.02
13                           Lost Property  0.02
14                                 Robbery  0.01
15                         Stolen Property  0.01
16                                   Other  0.01
17                Traffic Violation Arrest  0.01
18           Miscellaneous Investigation  0.01
```

In [49]:
```
num_top_incidents = 20

for hood in Incident_groupedcount['Analysis Neighborhood']:
    print("----"+hood+"----")
    temp = Incident_groupedcount[Incident_groupedcount['Analysis Neighborhood']
== hood].T.reset_index()
    temp.columns = ['type','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_incidents))
    print('\n')
```

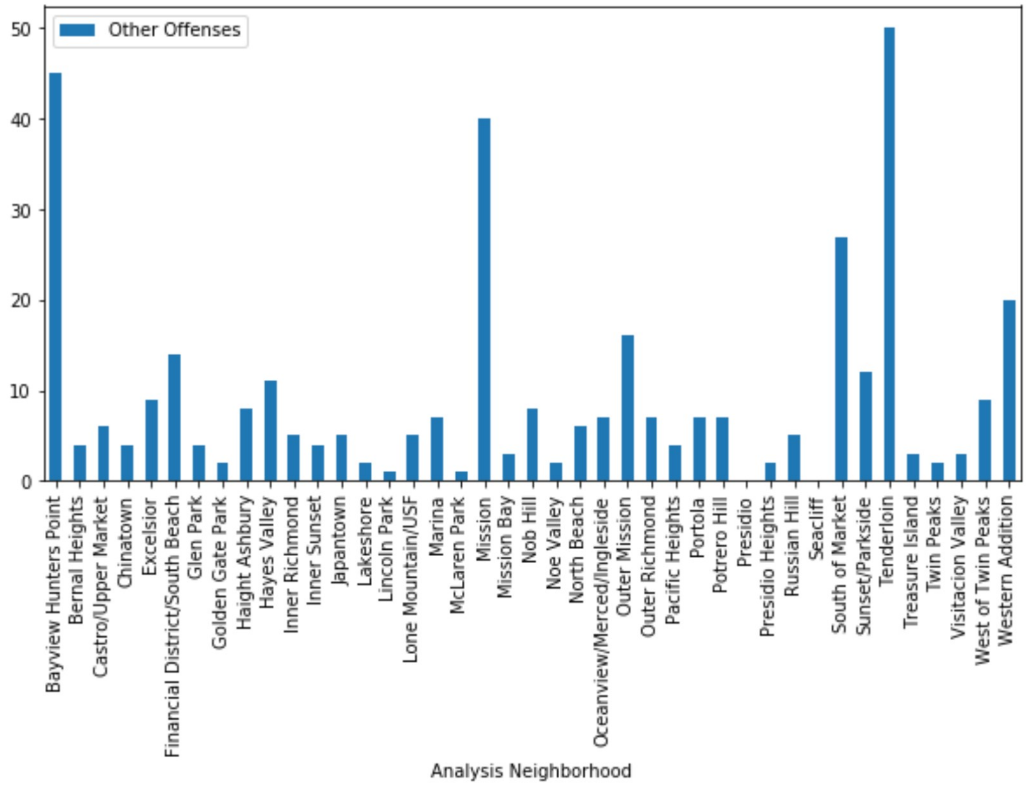
```
----Bayview Hunters Point----
                                     type   freq
0                           Larceny Theft  472.0
1                     Other Miscellaneous  285.0
2                                 Assault  274.0
3                       Recovered Vehicle  234.0
4                     Motor Vehicle Theft  220.0
5                      Malicious Mischief  214.0
6   Offences Against The Family And Children  153.0
7                            Non-Criminal  134.0
8                                Burglary  109.0
9                                 Robbery  106.0
10                     Disorderly Conduct   97.0
11                                Warrant   88.0
12                         Missing Person   84.0
13                         Suspicious Occ   70.0
14                        Weapons Offense   61.0
15                                  Fraud   47.0
16           Miscellaneous Investigation   45.0
17                         Other Offenses   45.0
18                  Weapons Carrying Etc   38.0
19                          Lost Property   37.0


----Bernal Heights----
                                     type   freq
0                           Larceny Theft  202.0
1                                Burglary  112.0
2                     Other Miscellaneous   89.0
3                      Malicious Mischief   75.0
4                            Non-Criminal   69.0
5                     Motor Vehicle Theft   66.0
6                                 Assault   61.0
7                          Missing Person   39.0
8   Offences Against The Family And Children   31.0
9                       Recovered Vehicle   31.0
10                         Suspicious Occ   24.0
11                                  Fraud   24.0
12                                Robbery   23.0
13                     Disorderly Conduct   18.0
14               Traffic Violation Arrest   15.0
15                                Warrant   15.0
16           Miscellaneous Investigation   10.0
17                  Weapons Carrying Etc    9.0
18                        Weapons Offense    9.0
19                          Lost Property    7.0


----Castro/Upper Market----
                                     type   freq
0                           Larceny Theft  400.0
1                                Burglary  126.0
2                     Other Miscellaneous  120.0
3                      Malicious Mischief  114.0
4                            Non-Criminal   85.0
5                                 Assault   77.0
6                     Motor Vehicle Theft   70.0
7                                   Fraud   50.0
8                                 Warrant   49.0
9                       Recovered Vehicle   32.0
10  Offences Against The Family And Children   31.0
11                          Lost Property   30.0
12                         Suspicious Occ   24.0
13                         Missing Person   21.0
14               Traffic Violation Arrest   20.0
15           Miscellaneous Investigation   17.0
16                          Drug Offense   16.0
17                     Disorderly Conduct   14.0
18                                Robbery   13.0
```

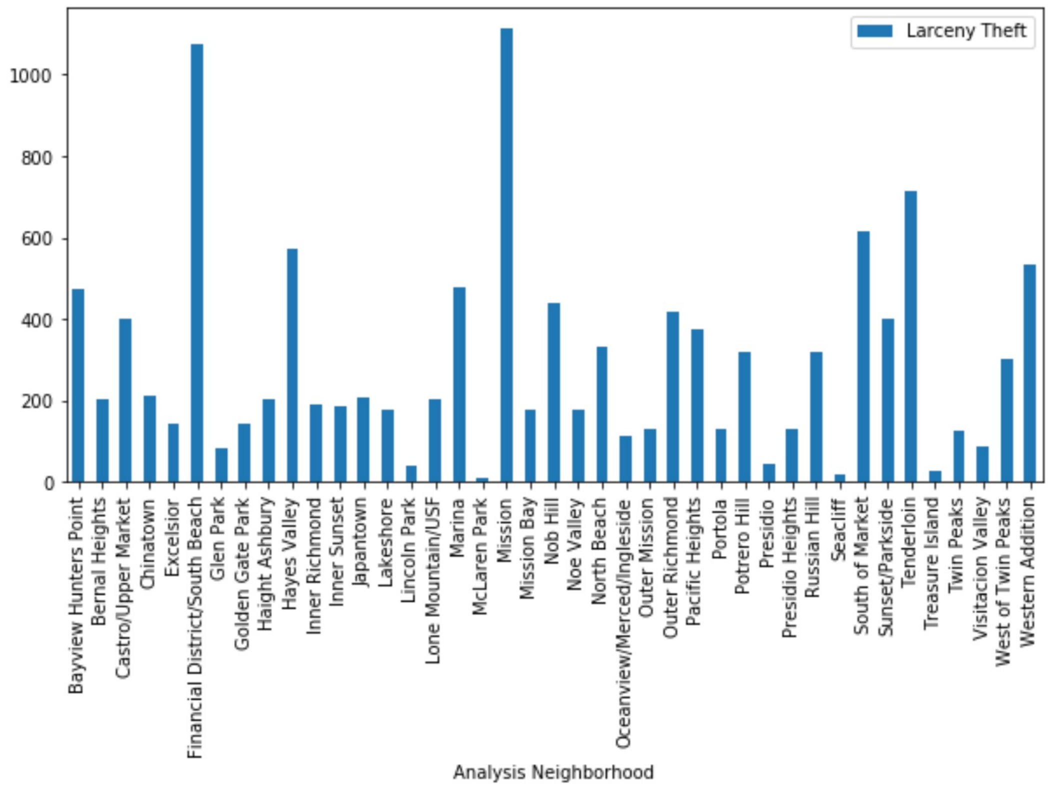
In [51]:
```
Incident_groupedcount.plot(kind='bar',x='Analysis Neighborhood',y='Other Offense
s', figsize=(10,5))
```

Out[51]: <matplotlib.axes._subplots.AxesSubplot at 0x7f5aacf18518>



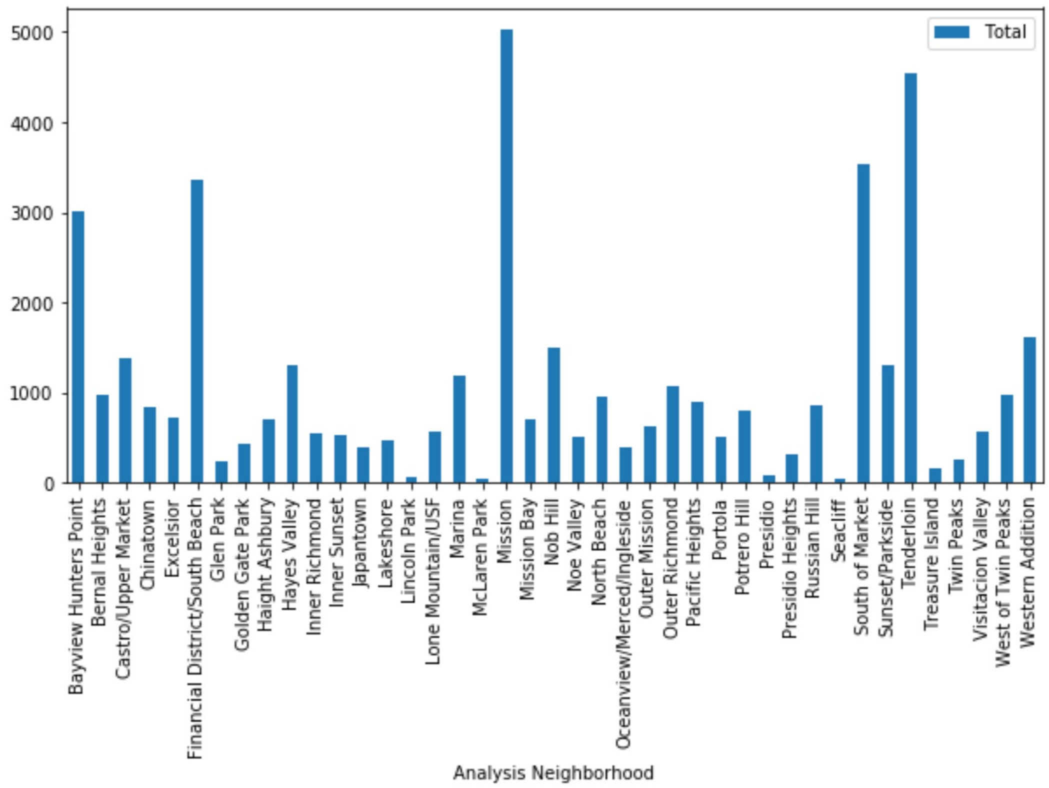
In [52]:
```
Incident_groupedcount.plot(kind='bar',x='Analysis Neighborhood',y='Larceny Theft
', figsize=(10,5))
```

Out[52]: <matplotlib.axes._subplots.AxesSubplot at 0x7f5aacdd7208>



In [53]:
```
Incident_groupedcount.plot(kind='bar',x='Analysis Neighborhood',y='Total', figsize=(10,5))
```

Out[53]: <matplotlib.axes._subplots.AxesSubplot at 0x7f5aacc3f278>



In [ ]:

In [ ]: