Austin Move Decision Making

In [ ]:
```
##Begin by importing libraries, setting up calls for later, etc
```

In [1]:
```
#import libraries
import pandas as pd
import numpy as np
import random
import requests
import json
import codecs
import os
import types
from urllib.request import urlopen
import plotly.express as px
import matplotlib.pylab as plt

!pip install geopy
from geopy.geocoders import Nominatim # module to convert an address into lati
tude and longitude values

# libraries for displaying images
from IPython.display import Image
from IPython.core.display import HTML

# tranforming json file into a pandas dataframe library
from pandas.io.json import json_normalize

! pip install folium==0.5.0
import folium

print('Folium installed')
print('Libraries imported.')
```

```
Requirement already satisfied: geopy in /opt/conda/envs/Python-3.7-main/lib/p
ython3.7/site-packages (2.0.0)
Requirement already satisfied: geographiclib<2,>=1.49 in /opt/conda/envs/Pyth
on-3.7-main/lib/python3.7/site-packages (from geopy) (1.50)
Collecting folium==0.5.0
  Downloading folium-0.5.0.tar.gz (79 kB)
     |████████████████████████████████| 79 kB 8.3 MB/s  eta 0:00:01
Collecting branca
  Downloading branca-0.4.1-py3-none-any.whl (24 kB)
Requirement already satisfied: jinja2 in /opt/conda/envs/Python-3.7-main/lib/
python3.7/site-packages (from folium==0.5.0) (2.11.2)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.7-main/li
b/python3.7/site-packages (from folium==0.5.0) (2.24.0)
Requirement already satisfied: six in /opt/conda/envs/Python-3.7-main/lib/pyt
hon3.7/site-packages (from folium==0.5.0) (1.15.0)
Requirement already satisfied: MarkupSafe>=0.23 in /opt/conda/envs/Python-3.7
-main/lib/python3.7/site-packages (from jinja2->folium==0.5.0) (1.1.1)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/envs/Python-3.7-mai
n/lib/python3.7/site-packages (from requests->folium==0.5.0) (2.9)
Requirement already satisfied: chardet<4,>=3.0.2 in /opt/conda/envs/Python-3.
7-main/lib/python3.7/site-packages (from requests->folium==0.5.0) (3.0.4)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /op
t/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from requests->foli
um==0.5.0) (1.25.9)
Requirement already satisfied: certifi>=2017.4.17 in /opt/conda/envs/Python-
3.7-main/lib/python3.7/site-packages (from requests->folium==0.5.0) (2020.6.2
0)
Building wheels for collected packages: folium
  Building wheel for folium (setup.py) ... done
  Created wheel for folium: filename=folium-0.5.0-py3-none-any.whl size=76240
sha256=0a0b0994eace445e337dde4285051a9603c1078020c5b018c7350d4c2e2ced52
  Stored in directory: /tmp/wsuser/.cache/pip/wheels/b2/2f/2c/109e446b990d663
ea5ce9b078b5e7c1a9c45cca91f377080f8
Successfully built folium
Installing collected packages: branca, folium
Successfully installed branca-0.4.1 folium-0.5.0
Folium installed
Libraries imported.
```

In [2]:

```
#add Foursquare information
CLIENT_ID = 'UOJC0SARIRT4OKOFRU5OHNLUKNS11BLSPUKF2MQSAVVJAL00' # your Foursqua
re ID
CLIENT_SECRET = '0RHBWZQXAAJ2BJGMYWOHMCSWEGCI05CQEBKUUVNBC2B3QRRN' # your Four
square Secret
VERSION = '20180604'
LIMIT = 30
print('Your credentails:')
print('CLIENT_ID: ' + CLIENT_ID)
print('CLIENT_SECRET:' + CLIENT_SECRET)
```

```
Your credentails:
CLIENT_ID: UOJC0SARIRT4OKOFRU5OHNLUKNS11BLSPUKF2MQSAVVJAL00
CLIENT_SECRET:0RHBWZQXAAJ2BJGMYWOHMCSWEGCI05CQEBKUUVNBC2B3QRRN
```

In [3]:
```
#add a location in central Austin to begin

address = '1100 Congress Ave, Austin, TX 78701'

geolocator = Nominatim(user_agent="foursquare_agent")
location = geolocator.geocode(address)
latitude = location.latitude
longitude = location.longitude
print(latitude, longitude)
```

```
30.276485 -97.739768
```

In [37]:
```
#create a statment terrain map to show the terrain around austin
atx_map = folium.Map(location=[latitude, longitude], zoom_start = 12, tiles =
'Stamen Terrain')

atx_map
```

Out[37]: Make this Notebook Trusted to load map: File -> Trust Notebook

Get Austin Data - Population, Crime Rates, Growth Rates

In [22]:
```
#get crime information in Austin areas from https://data.austintexas.gov/Public-Safety/2018-Annual-Crime/pgvh-cpyq/data
#data is available in a json here - https://data.austintexas.gov/resource/pgvh-cpyq.json
#use it to create a dataframe with number of crimes by zip code

r = requests.get('https://data.austintexas.gov/resource/pgvh-cpyq.json')
j = r.json()
df_crimes1 = pd.DataFrame.from_dict(j)

df_crimes1.head()
```

Out[22]:

	go_primary_key_year_plus	council_district	go_highest_offense_desc	crime_type	go_report_da
0	20183440706	9	AUTO THEFT	Auto Theft	10-Dec-
1	20185041473	9	THEFT BY SHOPLIFTING	Theft	16-Oct-
2	20188002409	10	BURGLARY OF VEHICLE	Theft	22-Jun-
3	20181860919	4	BURGLARY OF RESIDENCE	Burglary	5-Jul-
4	20185037796	7	BURGLARY OF VEHICLE	Theft	24-Sep-

In [23]:
```
#clean the rows and columns to get a final dataframe with zip code and number of crimes per zip code

# clean up the dataset to remove unnecessary columns
df_crimes1.drop(['go_primary_key_year_plus','go_highest_offense_desc','crime_type','go_report_date','go_location', 'go_x_coordinate', 'go_y_coordinate', 'clearance_status', 'clearance_date', 'go_district', 'council_district', 'go_census_tract'], axis=1, inplace=True)
df_crimes1.head()
```

Out[23]:

	go_location_zip
0	78705
1	78723
2	78703
3	78751
4	78727

In [24]:
```
# add total column & rename the columns.  Call the zip codes column ZCTA5CE10
 to match the json file later
df_crimes = df_crimes1.apply(pd.Series.value_counts).reset_index()
df_crimes.rename(columns={'index':'ZCTA5CE10', 'go_location_zip':'count'}, inp
lace=True)
df_crimes.head()
```

Out[24]:

	ZCTA5CE10	count
0	78741	84
1	78753	80
2	78704	73
3	78758	67
4	78701	57

In [25]:
```
#the above crimes are counts, not rates so now create a dataframe with populat
ion by zip code using data from here: http://zipatlas.com/us/tx/austin/zip-cod
e-comparison/population-density.htm
#I created both a .csv and xls with this data but Watson Studio was not upload
ing them properly hence this manual step
population_data = [[78705, 26825], [78751, 14005], [78752, 18030], [78741, 406
61], [78723, 30110], [78704, 43249], [78758, 42820], [78702, 22534], [78757, 2
1415], [78756, 7128], [78745, 53044], [78753, 44210], [78703, 19585], [78731,
24059], [78759, 40547], [78727, 22212], [78749, 28497], [78728, 17298], [78721
, 10124], [78722, 6365], [78729, 24561], [78701, 3855], [78748, 25361], [78750
, 23476], [78744, 33706], [78746, 26023], [78717, 8148], [78739, 8643], [78734
, 12925], [78724, 15428], [78726, 6480], [78733, 8716], [78754, 5422], [78735,
9671], [78732, 3629], [78742, 625], [78730, 4885], [78738, 2840], [78747, 5079
], [78736, 6897], [78737, 7100], [78725, 1836], [78719, 1765]]

df_pop = pd.DataFrame(population_data, columns = ['ZCTA5CE10', 'Population'])

df_pop.head()
```

Out[25]:

	ZCTA5CE10	Population
0	78705	26825
1	78751	14005
2	78752	18030
3	78741	40661
4	78723	30110

In [26]:
```
#merge crimes with population to get crime rates rather than number of crimes:
df_pop['ZCTA5CE10']=df_pop['ZCTA5CE10'].astype(int)
df_crimes['ZCTA5CE10']=df_crimes['ZCTA5CE10'].astype(int)
df_crime_rate = df_pop.merge(df_crimes, left_on='ZCTA5CE10', right_on='ZCTA5CE
10')

df_crime_rate['CrimeRate(%)'] = (df_crime_rate.fillna(0)['count'] / df_crime_r
ate.fillna(0)['Population']) *100

df_crime_rate.head()
```

Out[26]:

	ZCTA5CE10	Population	count	CrimeRate(%)
0	78705	26825	24	0.089469
1	78751	14005	22	0.157087
2	78752	18030	34	0.188575
3	78741	40661	84	0.206586
4	78723	30110	48	0.159415

In [40]:
```
#create a choropleth crime map of Austin by zip code
map = folium.Map(location=[30.2672, -97.7431], zoom_start=12)

map.choropleth(geo_data="https://raw.githubusercontent.com/OpenDataDE/State-zip-code-GeoJSON/master/tx_texas_zip_codes_geo.min.json",
             data=df_crime_rate,
             columns=['ZCTA5CE10', 'count'], #ZCTA5CE10 is the name of the zip codes in the json file
             key_on='feature.properties.ZCTA5CE10',
             fill_color='BuPu', fill_opacity=0.7, line_opacity=0.2,
             legend_name='Crimes Committed')
map
```

Out[40]: Make this Notebook Trusted to load map: File -> Trust Notebook

In [27]:
```
#look at recent growth trends by zip code.  Found a chart at the following url with numbers, manually create it into a dataframe https://www.austintexas.gov/sites/default/files/files/Planning/Demographics/ZIPs_Sales_Price_and_Density.pdf
growth_data = [[78754, .28], [78719, .24], [78717, .22], [78722, .23], [78723, .22], [78725, .21], [78737, .18], [78702, .17], [78756, .15], [78726, .13], [78747, .14], [78704, .12], [78705, .11], [78757, .11], [78729, .10], [78751, .10], [78738, .9], [78735, .9], [78739, .9], [78724, .9], [78736, .8], [78703, .8], [78748, .7], [78758, .7], [78721, .7], [78752, .6], [78727, .6], [78728, .6], [78753, .6], [78744, .6],  [78741, .6], [78750, .5], [78745,.5], [78759, .4], [78734, .3], [78731, .3], [78749, .2], [78733, .2], [78746, .2], [78730, .1]]

df_growth = pd.DataFrame(growth_data, columns = ['ZCTA5CE10', 'GrowthRate'])

df_growth.head()
```

Out[27]:

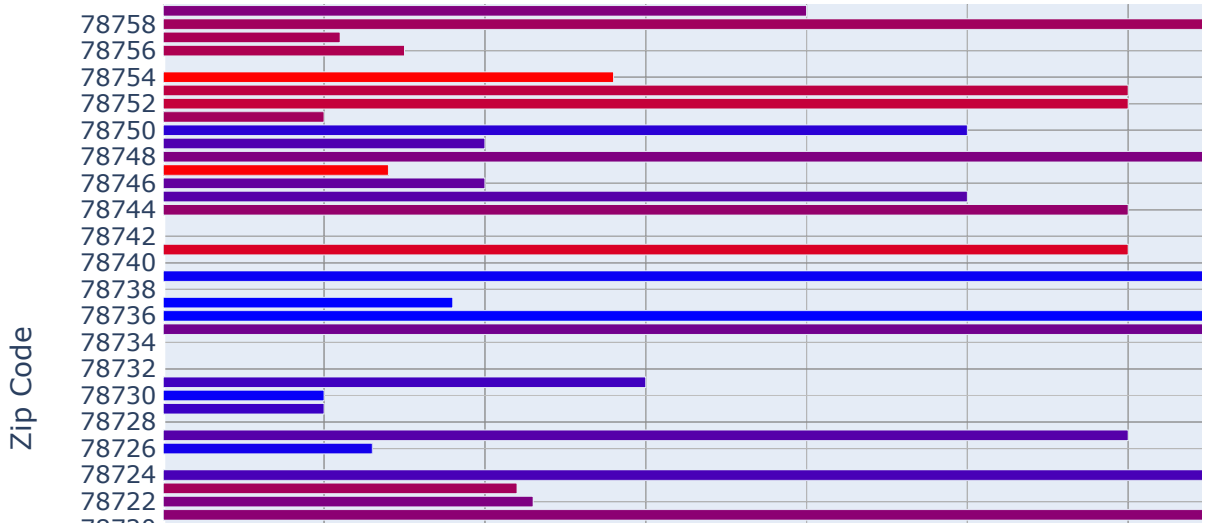
	ZCTA5CE10	GrowthRate
0	78754	0.28
1	78719	0.24
2	78717	0.22
3	78722	0.23
4	78723	0.22

In [28]:
```
#merge crimes with growth to get a cooler chart:
df_growth['ZCTA5CE10']=df_growth['ZCTA5CE10'].astype(str)
df_crime_rate['ZCTA5CE10']=df_crime_rate['ZCTA5CE10'].astype(str)
df_crime_growth = df_growth.merge(df_crime_rate, left_on='ZCTA5CE10', right_on='ZCTA5CE10')

df_crime_growth.head()
```

Out[28]:

	ZCTA5CE10	GrowthRate	Population	count	CrimeRate(%)
0	78754	0.28	5422	13	0.239764
1	78719	0.24	1765	1	0.056657
2	78717	0.22	8148	4	0.049092
3	78722	0.23	6365	8	0.125687
4	78723	0.22	30110	48	0.159415

In [31]:
```
#create a chart showing growth vs crimes by zipcode
fig = px.bar(df_crime_growth, y='ZCTA5CE10', x='GrowthRate', color="CrimeRate
(%)", orientation='h', color_continuous_scale='Bluered',
             labels={ "ZCTA5CE10": "Zip Code",
                      "GrowRate": "GrowthRate",
                      "CrimeRate(%)": "Crime Rate (%)"})
fig.update_layout(
   xaxis = dict (tickformat="%"),
   yaxis = dict (tickformat="##.##",
                 nticks=30))

fig.show()
```



Get Austin Venue Information from Foursquare

In [4]:
```
#pull in the dataset of Austin zip codes and their longitudes and latitudes.
 I took this dataset and deleted all non-Austin entries https://public.opendat
asoft.com/explore/dataset/us-zip-code-latitude-and-longitude/table/?refine.sta
te=TX
#since I am using IBM Watson studio notebook this is how I have to import the
 data
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It incl
udes your credentials.
# You might want to remove those credentials before you share the notebook.
client_4626cec52633482783cead9eb6658497 = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='cKGIXmI8E5JX4WZiOy1ST1QU0bR9K4ijARilZcwXPDT_',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3-api.us-geo.objectstorage.service.networklayer.co
m')

body = client_4626cec52633482783cead9eb6658497.get_object(Bucket='applieddatas
ciencecapstone-donotdelete-pr-pyxclczerfkqqr',Key='us-zip-code-latitude-and-lo
ngitude.csv')['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__,
body )

df_atx_long_lat = pd.read_csv(body)
df_atx_long_lat.head()
```

Out[4]:

	Zip	City	State	Latitude	Longitude
0	78727	Austin	TX	30.425652	-97.714190
1	78762	Austin	TX	30.326374	-97.771258
2	78763	Austin	TX	30.335398	-97.559807
3	78715	Austin	TX	30.450088	-97.486509
4	78741	Austin	TX	30.231252	-97.716000

In [5]:
```
#start getting venue info

LIMIT = 100
def getNearbyVenues(names, latitudes, longitudes, radius=700):

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

        # create the API request URL
        url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&clie
nt_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 = ['ZCTA5CE10',
                  'INTPTLAT10',
                  'INTPTLON10',
                  'Venue',
                  'Venue Latitude',
                  'Venue Longitude',
                  'Venue Category']
    print('done')
    return(nearby_venues)
```

In [6]:
```
df_venues = getNearbyVenues(names=df_atx_long_lat['Zip'],
                                    latitudes=df_atx_long_lat['Latitude'],
                                    longitudes=df_atx_long_lat['Longitude'])
print(df_venues.shape)
```

```
78727, 78762, 78763, 78715, 78741, 78767, 78720, 78704, 78738, 78729, 78748,
73301, 78744, 78783, 78749, 78785, 78735, 78731, 78769, 78747, 78756, 78736,
78765, 78789, 78773, 78709, 78734, 78739, 78725, 78772, 78701, 78705, 78721,
78787, 78716, 78768, 78733, 78723, 78708, 78737, 78728, 78732, 78779, 78761,
78746, 78972, 78730, 73344, 78760, 78782, 78718, 78703, 78753, 78724, 78764,
78758, 78745, 78759, 78774, 78754, 78712, 78751, 78752, 78757, 78778, 78755,
78713, 78766, 78719, 78711, 78726, 78788, 78710, 78750, 78780, 78717, 78771,
78781, 78702, 78742, 78786, 78714, 78722, done
(988, 7)
```

In [7]:
```
#create a table with venue categories across the top and zip codes down the si
de and get counts:
df_venues['COUNTER'] =1       #initially, set that counter to 1.
group_data = df_venues.groupby(['Venue Category','ZCTA5CE10'])['COUNTER'].sum
()
df_venue_by_zip = df_venues.groupby(['Venue Category','ZCTA5CE10'])['COUNTER']
.sum().reset_index()
#df_venue_by_zip.head()

df_venues_pivot = pd.pivot_table(df_venue_by_zip, values = 'COUNTER', index=[
'ZCTA5CE10'], columns = 'Venue Category').reset_index()
df_venues_pivot.head()
```

Out[7]:

Venue Category	ZCTA5CE10	Adult Boutique	American Restaurant	Arcade	Art Gallery	Art Museum	Arts & Crafts Store	Asian Restaurant	Assi Li
0	73301	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	73344	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	78701	NaN	2.0	NaN	NaN	NaN	NaN	NaN	
3	78702	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
4	78703	NaN	NaN	NaN	1.0	NaN	NaN	NaN	

5 rows × 227 columns

In [16]:
```
#get a list of all of the venue categories to determine which are interesting
pd.set_option('display.max_rows', None)
df_ven_cats = df_venues.groupby(['Venue Category']).size()
print(df_ven_cats)
```

Venue Category	
Adult Boutique	1
American Restaurant	8
Arcade	1
Art Gallery	4
Art Museum	3
Arts & Crafts Store	2
Asian Restaurant	2
Assisted Living	2
Athletics & Sports	5
Automotive Shop	2
BBQ Joint	3
Bagel Shop	3
Bakery	10
Bank	7
Bar	15
Basketball Court	1
Bed & Breakfast	1
Beer Bar	1
Beer Garden	4
Beer Store	1
Big Box Store	1
Bookstore	1
Boutique	1
Breakfast Spot	7
Brewery	2
Bubble Tea Shop	1
Burger Joint	11
Burrito Place	2
Bus Station	5
Bus Stop	5
Business Service	9
Butcher	1
Café	6
Cajun / Creole Restaurant	1
Capitol Building	3
Car Wash	1
Casino	1
Chinese Restaurant	7
Chiropractor	1
Clothing Store	7
Cocktail Bar	8
Coffee Shop	29
College Bookstore	1
College Gym	1
College Quad	1
Comedy Club	2
Comfort Food Restaurant	1
Concert Hall	1
Construction & Landscaping	2
Convenience Store	18
Convention Center	1
Cosmetics Shop	11
Cupcake Shop	2
Cycle Studio	1
Dance Studio	2
Deli / Bodega	2

Department Store	2
Dessert Shop	3
Diner	2
Discount Store	8
Dive Bar	4
Doctor's Office	2
Dog Run	1
Donut Shop	9
Dry Cleaner	2
Electronics Store	2
Event Service	1
Fabric Shop	1
Falafel Restaurant	1
Farm	1
Fast Food Restaurant	14
Field	1
Financial or Legal Service	1
Flea Market	1
Flower Shop	1
Food	2
Food & Drink Shop	4
Food Court	1
Food Stand	1
Food Truck	24
Football Stadium	2
Fountain	1
French Restaurant	2
Fried Chicken Joint	8
Frozen Yogurt Shop	1
Furniture / Home Store	14
Garden	2
Gas Station	13
Gastropub	1
Gay Bar	1
General Entertainment	3
Gift Shop	4
Golf Driving Range	1
Gourmet Shop	1
Greek Restaurant	1
Grocery Store	12
Gym	11
Gym / Fitness Center	12
Health & Beauty Service	2
Health Food Store	1
Historic Site	1
History Museum	3
Hobby Shop	3
Home Service	2
Hookah Bar	1
Hostel	1
Hot Dog Joint	1
Hotel	13
Hotel Bar	1
Ice Cream Shop	8
Indian Restaurant	4
Intersection	2
Italian Restaurant	4

Japanese Restaurant	3
Jazz Club	1
Juice Bar	6
Karaoke Bar	1
Kebab Restaurant	1
Kitchen Supply Store	1
Korean Restaurant	5
Lake	1
Latin American Restaurant	1
Lawyer	1
Lighthouse	1
Lingerie Store	1
Liquor Store	7
Lounge	4
Market	2
Martial Arts School	2
Massage Studio	2
Mediterranean Restaurant	5
Men's Store	1
Mexican Restaurant	40
Middle Eastern Restaurant	3
Miscellaneous Shop	1
Mobile Phone Shop	4
Monument / Landmark	1
Moroccan Restaurant	1
Motel	1
Motorcycle Shop	2
Motorsports Shop	1
Mountain	29
Movie Theater	6
Museum	1
Music Store	3
Music Venue	6
Nail Salon	1
New American Restaurant	5
Nightclub	3
Optical Shop	2
Outdoor Sculpture	2
Park	106
Pastry Shop	1
Performing Arts Venue	2
Pet Store	3
Pharmacy	4
Piano Bar	1
Pizza Place	22
Playground	5
Pool	14
Pool Hall	1
Pub	7
Public Art	2
Ramen Restaurant	1
Record Shop	1
Recording Studio	1
Recreation Center	3
Rental Car Location	4
Rental Service	1
Restaurant	6

```
Rock Club                          3
Russian Restaurant                 1
Salad Place                        2
Salon / Barbershop                 1
Sandwich Place                    23
Scenic Lookout                    30
Sculpture Garden                   1
Seafood Restaurant                 2
Shipping Store                     5
Shoe Store                         1
Shop & Service                     2
Shopping Mall                      1
Skating Rink                       1
Ski Shop                           1
Smoke Shop                         8
Smoothie Shop                      1
Snack Place                        1
Soccer Field                       3
Soccer Stadium                     1
Soup Place                         1
South American Restaurant          1
Southern / Soul Food Restaurant    2
Souvenir Shop                      1
Spa                                2
Speakeasy                          1
Sporting Goods Shop                1
Stables                            1
Steakhouse                         1
Storage Facility                   1
Student Center                     3
Supplement Shop                    2
Sushi Restaurant                   5
Taco Place                        16
Tailor Shop                        1
Tanning Salon                      2
Tapas Restaurant                   1
Tennis Court                       3
Tex-Mex Restaurant                 1
Thai Restaurant                    5
Theater                            3
Thrift / Vintage Store            11
Toy / Game Store                   2
Trade School                       1
Trail                              1
Tree                               1
Vegetarian / Vegan Restaurant      4
Video Game Store                   1
Video Store                        8
Vietnamese Restaurant              4
Volleyball Court                   1
Weight Loss Center                 1
Whisky Bar                         1
Wine Shop                          1
Wings Joint                        1
Women's Store                      1
Yoga Studio                        6
dtype: int64
```

In [8]:
```
#drop columns our family is not interested in
df_venues_pivot.drop(['Adult Boutique', 'Arcade', 'Art Gallery', 'Art Museum',
"Arts & Crafts Store", 'Assisted Living', 'Automotive Shop', 'BBQ Joint', 'Bag
el Shop', 'Bakery', 'Bank', 'Basketball Court', 'Bed & Breakfast', 'Beer Bar',
'Beer Garden', 'Beer Store', 'Big Box Store', 'Boutique', 'Brewery', 'Bus Stat
ion', 'Bus Stop', 'Business Service', 'Butcher', 'Cajun / Creole Restaurant',
'Capitol Building', 'Car Wash', 'Casino', 'Chinese Restaurant', 'Chiropractor'
, 'Clothing Store', 'College Bookstore', 'College Gym', 'College Quad', 'Comed
y Club', 'Comfort Food Restaurant', 'Concert Hall', 'Construction & Landscapin
g', 'Convenience Store', 'Convention Center', 'Cosmetics Shop', 'Cupcake Shop'
, 'Cycle Studio', 'Dance Studio', 'Deli / Bodega', 'Department Store', 'Desser
t Shop', 'Diner', 'Discount Store', 'Dive Bar', "Doctor's Office", 'Dog Run',
'Donut Shop', 'Dry Cleaner', 'Electronics Store', 'Event Service', 'Fabric Sho
p', 'Farm', 'Fast Food Restaurant', 'Field', 'Financial or Legal Service', 'Fl
ea Market', 'Flower Shop', 'Food', 'Food & Drink Shop', 'Food Court', 'Food St
and', 'Football Stadium', 'Fountain', 'French Restaurant', 'Fried Chicken Join
t', 'Frozen Yogurt Shop', 'Furniture / Home Store', 'Garden', 'Gas Station',
'Gastropub', 'Gay Bar', 'General Entertainment', 'Gift Shop', 'Golf Driving Ra
nge', 'Gourmet Shop', 'Greek Restaurant', 'Health & Beauty Service', 'Historic
Site', 'History Museum', 'Hobby Shop', 'Home Service', 'Hostel', 'Hot Dog Join
t', 'Hotel', 'Hotel Bar', 'Ice Cream Shop', 'Indian Restaurant', 'Intersectio
n', 'Italian Restaurant', 'Juice Bar', 'Karaoke Bar', 'Kitchen Supply Store',
'Lake', 'Lawyer', 'Lighthouse', 'Lingerie Store', 'Liquor Store', 'Lounge', 'M
arket', 'Martial Arts School', 'Massage Studio', "Men's Store", 'Miscellaneous
Shop', 'Mobile Phone Shop', 'Monument / Landmark', 'Motel', 'Motorcycle Shop',
'Motorsports Shop', 'Mountain', 'Movie Theater', 'Museum', 'Music Store', 'Mus
ic Venue', 'Nail Salon', 'New American Restaurant', 'Nightclub', 'Optical Sho
p', 'Outdoor Sculpture', 'Pastry Shop', 'Performing Arts Venue', 'Pharmacy',
'Pizza Place', 'Playground', 'Pool', 'Public Art', 'Ramen Restaurant', 'Record
Shop', 'Recording Studio', 'Rental Car Location', 'Rental Service', 'Restauran
t', 'Rock Club', 'Russian Restaurant', 'Salad Place', 'Salon / Barbershop', 'S
andwich Place', 'Sculpture Garden', 'Seafood Restaurant', 'Shipping Store', 'S
hoe Store', 'Shop & Service', 'Shopping Mall', 'Skating Rink', 'Ski Shop', 'Sm
oke Shop', 'Smoothie Shop', 'Snack Place', 'Soccer Field', 'Soccer Stadium',
'Soup Place', 'South American Restaurant', 'Southern / Soul Food Restaurant',
'Souvenir Shop', 'Spa', 'Speakeasy', 'Sporting Goods Shop', 'Stables', 'Steakh
ouse', 'Storage Facility', 'Student Center', 'Supplement Shop', 'Tailor Shop',
'Tanning Salon', 'Tapas Restaurant', 'Tennis Court', 'Theater', 'Thrift / Vint
age Store', 'Toy / Game Store', 'Trade School', 'Tree', 'Vegetarian / Vegan Re
staurant', 'Video Game Store', 'Video Store', 'Vietnamese Restaurant', 'Volley
ball Court', 'Weight Loss Center', 'Wine Shop', 'Wings Joint', "Women's Store"
, 'Yoga Studio'], axis=1, inplace=True)
df_venues_pivot.head()
```

Out[8]:

Venue Category	ZCTA5CE10	American Restaurant	Asian Restaurant	Athletics & Sports	Bar	Bookstore	Breakfast Spot	Bubble Tea Shop	Bu
0	73301	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
1	73344	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	78701	2.0	NaN	NaN	3.0	NaN	NaN	NaN	
3	78702	NaN	NaN	NaN	2.0	NaN	1.0	NaN	
4	78703	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

5 rows × 42 columns

In [9]:
```
#create a cleaner final data table
#assume listings are mutually exclusive and create higher level categories:
df_venues_pivot['Restaurants'] =  (df_venues_pivot.fillna(0)['American Restaurant'] + df_venues_pivot.fillna(0)['Asian Restaurant'] + df_venues_pivot.fillna(0)['Breakfast Spot'] + df_venues_pivot.fillna(0)['Burger Joint'] +
                                   df_venues_pivot.fillna(0)['Burrito Place'] + df_venues_pivot.fillna(0)['Taco Place'] + df_venues_pivot.fillna(0)['Tex-Mex Restaurant'] + df_venues_pivot.fillna(0)['Thai Restaurant'] +
                                   df_venues_pivot.fillna(0)['Café'] + df_venues_pivot.fillna(0)['Falafel Restaurant'] + df_venues_pivot.fillna(0)['Food Truck'] + df_venues_pivot.fillna(0)['Mexican Restaurant'] +
                                   df_venues_pivot.fillna(0)['Japanese Restaurant'] + df_venues_pivot.fillna(0)['Latin American Restaurant'] + df_venues_pivot.fillna(0)['Mediterranean Restaurant'] +
                                   df_venues_pivot.fillna(0)['Middle Eastern Restaurant'] + df_venues_pivot.fillna(0)['Moroccan Restaurant'] + df_venues_pivot.fillna(0)['Kebab Restaurant'] +
                                   df_venues_pivot.fillna(0)['Korean Restaurant'] +  df_venues_pivot.fillna(0)['Sushi Restaurant'])

df_venues_pivot['Bars'] = (df_venues_pivot.fillna(0)['Bar'] + df_venues_pivot.fillna(0)['Pub'] + df_venues_pivot.fillna(0)['Whisky Bar'] + df_venues_pivot.fillna(0)['Cocktail Bar'] +
                           df_venues_pivot.fillna(0)['Piano Bar'])

df_venues_pivot['Essentials'] = (df_venues_pivot.fillna(0)['Grocery Store'] + df_venues_pivot.fillna(0)['Pet Store'] + df_venues_pivot.fillna(0)['Health Food Store'])

df_venues_pivot['Gyms'] =  (df_venues_pivot.fillna(0)['Gym'] + df_venues_pivot.fillna(0)['Gym / Fitness Center'] + df_venues_pivot.fillna(0)['Recreation Center'])

df_venues_pivot['Entertainment'] =  (df_venues_pivot.fillna(0)['Bookstore'] + df_venues_pivot.fillna(0)['Bubble Tea Shop'] + df_venues_pivot.fillna(0)['Coffee Shop'] +
                                     df_venues_pivot.fillna(0)['Hookah Bar'] + df_venues_pivot.fillna(0)['Jazz Club'] + df_venues_pivot.fillna(0)['Pool Hall'])

#clean the extra columns:
df_venues_pivot.drop(['American Restaurant', 'Asian Restaurant', 'Breakfast Spot', 'Burger Joint', 'Burrito Place', 'Taco Place', 'Tex-Mex Restaurant', 'Thai Restaurant', 'Café', 'Falafel Restaurant', 'Food Truck'], axis=1, inplace=True)
df_venues_pivot.drop(['Mexican Restaurant', 'Japanese Restaurant', 'Latin American Restaurant', 'Mediterranean Restaurant', 'Middle Eastern Restaurant', 'Moroccan Restaurant', 'Kebab Restaurant', 'Korean Restaurant', 'Sushi Restaurant'], axis=1, inplace=True)

df_venues_pivot.drop(['Bar', 'Pub', 'Whisky Bar', 'Piano Bar', 'Cocktail Bar'], axis=1, inplace=True)

df_venues_pivot.drop(['Grocery Store', 'Pet Store', 'Health Food Store'], axis=1, inplace=True)
```

```
df_venues_pivot.drop(['Gym', 'Gym / Fitness Center', 'Recreation Center'], axis=1, inplace=True)

df_venues_pivot.drop(['Bookstore', 'Bubble Tea Shop', 'Coffee Shop', 'Hookah Bar', 'Jazz Club', 'Pool Hall'], axis=1, inplace=True)

#check the output:
df_venues_pivot.head()
```

Out[9]:

Venue Category	ZCTA5CE10	Athletics & Sports	Park	Scenic Lookout	Trail	Restaurants	Bars	Essentials	Gyms	En
0	73301	NaN	3.0	1.0	NaN	0.0	0.0	0.0	0.0	
1	73344	NaN	3.0	1.0	NaN	0.0	0.0	0.0	0.0	
2	78701	NaN	NaN	NaN	NaN	14.0	9.0	1.0	1.0	
3	78702	NaN	1.0	NaN	NaN	15.0	8.0	1.0	0.0	
4	78703	NaN	2.0	NaN	NaN	0.0	0.0	0.0	0.0	

In [34]:

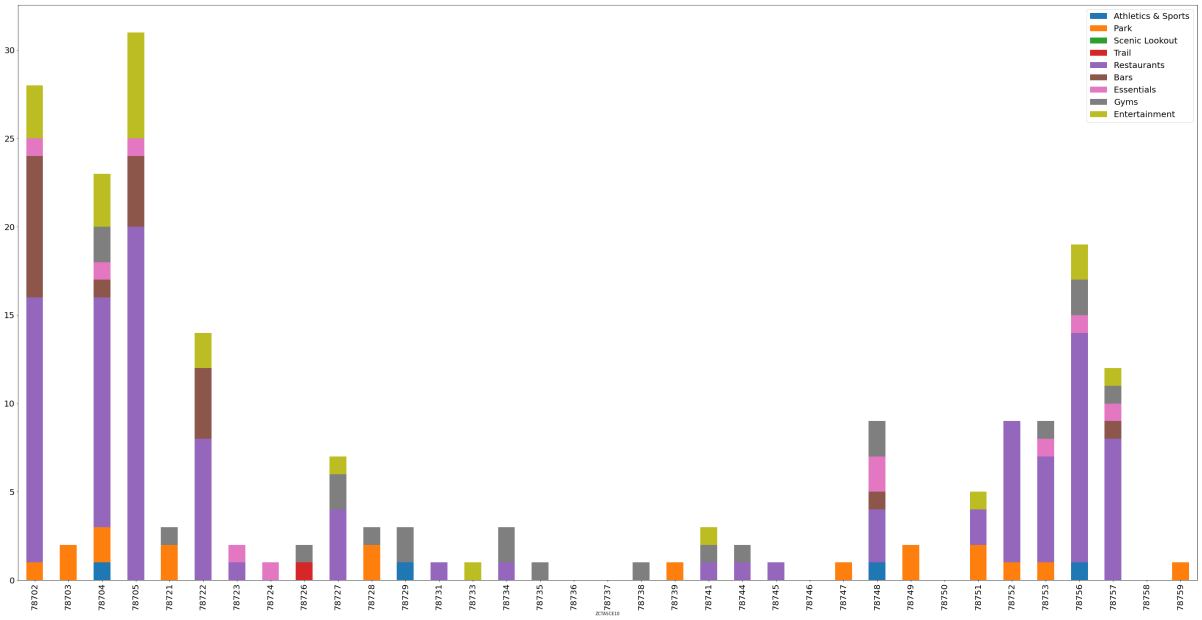
```
#clean the extraneous zip codes out of the dataframe
zips = [[78754], [78719], [78717], [78722], [78723], [78725], [78737], [78702], [78756], [78726], [78747], [78704], [78705], [78757], [78729], [78751], [78738], [78735], [78739], [78724], [78736], [78703], [78748], [78758], [78721], [78752], [78727], [78728], [78753], [78744], [78741], [78750], [78745], [78759], [78734], [78731], [78749], [78733], [78746], [78730]]
df_zips = pd.DataFrame(zips, columns = ['ZCTA5CE10'])

df_venues_pivot2 = df_zips.merge(df_venues_pivot, on='ZCTA5CE10',how='inner', indicator=True)
df_venues_pivot2.drop(['_merge'], axis=1, inplace=True)

df_venues_pivot2.head()
```

Out[34]:

	ZCTA5CE10	Athletics & Sports	Park	Scenic Lookout	Trail	Restaurants	Bars	Essentials	
count	35.000000	4.0	12.000000	0.0	1.0	35.000000	35.000000	35.000000	35
mean	78735.800000	1.0	1.500000	NaN	1.0	3.028571	0.542857	0.285714	0
std	16.118678	0.0	0.522233	NaN	NaN	5.159148	1.615055	0.518563	0
min	78702.000000	1.0	1.000000	NaN	1.0	0.000000	0.000000	0.000000	0
25%	78726.500000	1.0	1.000000	NaN	1.0	0.000000	0.000000	0.000000	0
50%	78737.000000	1.0	1.500000	NaN	1.0	0.000000	0.000000	0.000000	0
75%	78748.500000	1.0	2.000000	NaN	1.0	3.500000	0.000000	0.500000	1
max	78759.000000	1.0	2.000000	NaN	1.0	20.000000	8.000000	2.000000	2

In [35]:
```
#create a stacked bar graph of venue information

ax1 = df_venues_pivot2.groupby(['ZCTA5CE10']).sum().fillna(0)

ax = ax1.plot(kind='bar', stacked=True, figsize=(50,25), fontsize=20, legend=20)

ax.legend(fontsize=20)
```

Out[35]:
```
<matplotlib.legend.Legend at 0x7fdd6eefdfd0>
```



In [36]:
```
#create a cleaner version with total Venues for use later
df_ttl_venue = df_venues_pivot2.copy()
df_ttl_venue['All Venues'] = (df_ttl_venue.fillna(0)['Athletics & Sports'] + df_ttl_venue.fillna(0)['Park'] + df_ttl_venue.fillna(0)['Scenic Lookout'] + df_ttl_venue.fillna(0)['Trail'] +
                              df_ttl_venue.fillna(0)['Restaurants'] + df_ttl_venue.fillna(0)['Bars'] + df_ttl_venue.fillna(0)['Essentials'] +
                              df_ttl_venue.fillna(0)['Gyms'] + df_ttl_venue.fillna(0)['Entertainment'])
df_ttl_venue.drop(['Athletics & Sports', 'Park', 'Scenic Lookout', 'Trail', 'Restaurants', 'Bars', 'Essentials', 'Gyms', 'Entertainment'], axis=1, inplace=True)

df_ttl_venue.head()
```

Out[36]:

	ZCTA5CE10	All Venues
0	78722	14.0
1	78723	2.0
2	78737	0.0
3	78702	28.0
4	78756	19.0

Get Austin Rent Information

In [20]:
```
#now create a dataframe with rent by zip code using data from here: https://www.homecity.com/blog/the-cheapest-and-most-expensive-austin-zip-codes/
rent_data = [[78724, 612], [78721, 612], [78664, 807], [78660, 807], [78728, 807], [78717, 807], [78726, 807], [78750, 807], [78757, 807], [78752, 807], [78723, 807], [78741, 807], [78744, 807], [78747, 807], [78748, 807], [78681, 1006], [78729, 1006], [78727, 1006], [78759, 1006], [78737, 1006], [78756, 1006], [78751, 1006], [78722, 1006], [78732, 1006], [78730, 1006], [78736, 1006], [78749, 1006], [78745, 1006], [78613, 1236], [78735, 1236], [78746, 1236], [78703, 1236], [78705, 1236], [78702, 1236], [78704, 1236], [78737, 1686]]

df_rent = pd.DataFrame(rent_data, columns = ['ZCTA5CE10', 'Avg Rent'])

df_rent.head()
```

Out[20]:

	ZCTA5CE10	Avg Rent
0	78724	612
1	78721	612
2	78664	807
3	78660	807
4	78728	807

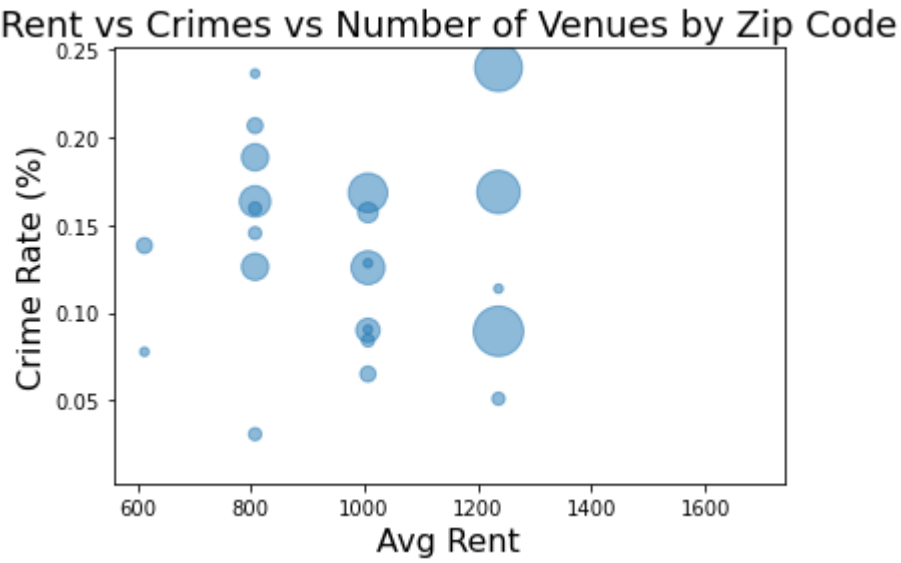
In [29]:
```
#merge the datasets together
df_rent['ZCTA5CE10']=df_rent['ZCTA5CE10'].astype(str)
df_crime_rate['ZCTA5CE10']=df_crime_rate['ZCTA5CE10'].astype(str)

df_rent_crime = df_rent.merge(df_crime_rate, left_on='ZCTA5CE10', right_on='ZCTA5CE10')

df_ttl_venue['ZCTA5CE10']=df_ttl_venue['ZCTA5CE10'].astype(str).reset_index(drop=True)
df_rent_crime['ZCTA5CE10']=df_rent_crime['ZCTA5CE10'].astype(str).reset_index(drop=True)

df_all = df_rent_crime.merge(df_ttl_venue, left_on='ZCTA5CE10', right_on='ZCTA5CE10')
df_all.head()
```

Out[29]:

	ZCTA5CE10	Avg Rent	Population	count	CrimeRate(%)	All Venues
0	78724	612	15428	12	0.077781	1.0
1	78721	612	10124	14	0.138285	3.0
2	78717	807	8148	4	0.049092	0.0
3	78726	807	6480	2	0.030864	2.0
4	78750	807	23476	12	0.051116	0.0

In [30]:
```
#Compare rent to total number of venues to crime rate - couldn't get this to c
olor by zip code so have to compare back to the combined dataset once you find
bubbles of interest

#add a column to make bubbles easier to see:
df_all['Venue_bigger'] = (df_all.fillna(0)['All Venues'] ) *20

plt.scatter('Avg Rent', 'CrimeRate(%)',
            s='Venue_bigger',
            alpha=.5, data=df_all)
plt.xlabel("Avg Rent", size=16)
plt.ylabel("Crime Rate (%)", size=16)
plt.title("Rent vs Crimes vs Number of Venues by Zip Code", size=18)
```

Out[30]:
```
Text(0.5, 1.0, 'Rent vs Crimes vs Number of Venues by Zip Code')
```



In [45]:
```
#get a final dataset with all relevant data by zip code
df_all['ZCTA5CE10']=df_all['ZCTA5CE10'].astype(str)
df_venues_pivot2['ZCTA5CE10']=df_venues_pivot2['ZCTA5CE10'].astype(str)

df_all2 = df_all.merge(df_venues_pivot2, left_on='ZCTA5CE10', right_on='ZCTA5CE10')

df_all2['ZCTA5CE10']=df_ttl_venue['ZCTA5CE10'].astype(str)
df_growth['ZCTA5CE10']=df_growth['ZCTA5CE10'].astype(str)

df_all3 = df_all2.merge(df_growth, left_on='ZCTA5CE10', right_on='ZCTA5CE10')

df_all3.drop(['Venue_bigger'], axis=1, inplace=True)
df_all3.rename(columns={'count':'Total Crimes', 'ZCTA5CE10':'Zip Code'}, inplace=True)

df_all3.head()
```

Out[45]:

	Zip Code	Avg Rent	Population	Total Crimes	CrimeRate(%)	All Venues	Athletics & Sports	Park	Scenic Lookout	Trail	Res
0	78722	612	15428	12	0.077781	1.0	NaN	NaN	NaN	NaN	
1	78723	612	10124	14	0.138285	3.0	NaN	2.0	NaN	NaN	
2	78737	807	6480	2	0.030864	2.0	NaN	NaN	NaN	1.0	
3	78702	807	23476	12	0.051116	0.0	NaN	NaN	NaN	NaN	
4	78756	807	21415	35	0.163437	12.0	NaN	NaN	NaN	NaN	