

My original plans for this assignment were to create a new schema for data, utilizing BeautifulSoup and <https://interactive.taxfoundation.org/propertytax/> (<https://interactive.taxfoundation.org/propertytax/>) and <https://www.zillow.com/howto/api/GetSearchResults.htm> (<https://www.zillow.com/howto/api/GetSearchResults.htm>) to create a collection of properties that have tax info and associated price. Then, using the free <https://developer.mapquest.com/documentation/open/geocoding-api/batch/get/> (<https://developer.mapquest.com/documentation/open/geocoding-api/batch/get/>) to get the lat, long coordinates and calculate distance to the metro. However, due to an error with the taxfoundation website and Zillow's developer portal having issues, I crafted my own dataset from information found online by googling. Note, it is not accurate, as the distance was done manually for now. This is merely an experiment with plotly and how we can display the data to the user.

```
In [2]: import plotly.express as px
import pandas as pd
import csv
# TODO: utilize public tax info to generate this, use zillowHelper.py to es
timate/display realestate/rent costs
entries = {
    'address': [],
    'citystatezip': [],
    'price': [],
    'distance': [],
}
with open('addresses.csv', mode='r') as infile:
    reader = csv.reader(infile)
    for row in reader:
        entries['address'].append(row[0])
        entries['citystatezip'].append(row[1])
        entries['price'].append(row[2])
        entries['distance'].append(row[3])

n_entries = len(entries['address'])

df = pd.DataFrame(dict(distance=entries['distance'], price=entries['price'],
                        citystatezip=entries['citystatezip']))
# Use column names of df for the different parameters x, y, color, ...
fig = px.scatter(df, x="distance", y="price",
                 title="Distance From Metro vs. Price",
                 hover_name="citystatezip", hover_data=['citystatezip'],
                 )

fig.show()
```