# Final Project Milestone1

# Dataset selection

## Dataset from file

### Data Source

https://www.kaggle.com/c/zillow-prize-1

#### Description

There are two data sets with over 1 million records each and 58 columns. properties_2016 and properties_2017 datasets contain data for each year. The data we will use for this project will be a small sample of the master data.

The two datasets are linked by parcleid.

I transactions dataset, the trabsaction date shows the date the property was sold and logerror is the log10( estimated price - price sold).

Properties dataset has the physical information about the properties. The columns on the properties dataset will have to be renamed. Subsets of data can be used to group by region, and other features such as number of bedrooms, square footage, etc.

In [33]:

```
# Load Libraries
import pandas as pd
import matplotlib.pyplot as plt
import xlrd

# Load Data
transactions_2016 = "Data/transactions_2016.json"
transactions_2017 = "Data/transactions_2017.json"

properties_2016  =  "Data/properties_2016.csv"
properties_2017  =  "Data/properties_2017.csv"
data_dictionary = "Data/data_dictionary.xlsx"

transactions_2016 = pd.read_json(transactions_2016)
transactions_2017 = pd.read_json(transactions_2017)
properties_2016 = pd.read_csv(properties_2016)
properties_2017 = pd.read_csv(properties_2017)
data_dictionary = pd.read_excel(data_dictionary)
```

In [34]:

```
transactions_2016.head()
```

Out[34]:

| | parcelid | logerror | transactiondate |
|---|---|---|---|
| **0** | 11016594 | 0.0276 | 2016-01-01 |
| **1** | 14366692 | -0.1684 | 2016-01-01 |
| **2** | 12098116 | -0.0040 | 2016-01-01 |
| **3** | 12643413 | 0.0218 | 2016-01-02 |
| **4** | 14432541 | -0.0050 | 2016-01-02 |

In [23]:

```
properties_2016.head()
```

Out[23]:

| | Unnamed: 0 | parcelid | airconditioningtypeid | architecturalstyletypeid | basementsqft | bathroomcnt |
|---|---|---|---|---|---|---|
| **0** | 0 | 10754147 | NaN | NaN | NaN | 0.0 |
| **1** | 1 | 10759547 | NaN | NaN | NaN | 0.0 |
| **2** | 2 | 10843547 | NaN | NaN | NaN | 0.0 |
| **3** | 3 | 10859147 | NaN | NaN | NaN | 0.0 |
| **4** | 4 | 10879947 | NaN | NaN | NaN | 0.0 |

5 rows × 59 columns

In [31]:

```
properties_2016.columns
```

Out[31]:

```
Index(['Unnamed: 0', 'parcelid', 'airconditioningtypeid',
       'architecturalstyletypeid', 'basementsqft', 'bathroomcnt', 'bedro
omcnt',
       'buildingclasstypeid', 'buildingqualitytypeid', 'calculatedbathnb
r',
       'decktypeid', 'finishedfloor1squarefeet',
       'calculatedfinishedsquarefeet', 'finishedsquarefeet12',
       'finishedsquarefeet13', 'finishedsquarefeet15', 'finishedsquarefe
et50',
       'finishedsquarefeet6', 'fips', 'fireplacecnt', 'fullbathcnt',
       'garagecarcnt', 'garagetotalsqft', 'hashottuborspa',
       'heatingorsystemtypeid', 'latitude', 'longitude', 'lotsizesquaref
eet',
       'poolcnt', 'poolsizesum', 'pooltypeid10', 'pooltypeid2', 'pooltyp
eid7',
       'propertycountylandusecode', 'propertylandusetypeid',
       'propertyzoningdesc', 'rawcensustractandblock', 'regionidcity',
       'regionidcounty', 'regionidneighborhood', 'regionidzip', 'roomcn
t',
       'storytypeid', 'threequarterbathnbr', 'typeconstructiontypeid',
       'unitcnt', 'yardbuildingsqft17', 'yardbuildingsqft26', 'yearbuil
t',
       'numberofstories', 'fireplaceflag', 'structuretaxvaluedollarcnt',
       'taxvaluedollarcnt', 'assessmentyear', 'landtaxvaluedollarcnt',
       'taxamount', 'taxdelinquencyflag', 'taxdelinquencyyear',
       'censustractandblock'],
      dtype='object')
```

In [24]:

```
data_dictionary.head()
```

Out[24]:

| | Feature | Description |
|---|---|---|
| **0** | 'airconditioningtypeid' | Type of cooling system present in the home (i... |
| **1** | 'architecturalstyletypeid' | Architectural style of the home (i.e. ranch, ... |
| **2** | 'basementsqft' | Finished living area below or partially below... |
| **3** | 'bathroomcnt' | Number of bathrooms in home including fractio... |
| **4** | 'bedroomcnt' | Number of bedrooms in home |

## Webscaraping Data Source

#### Description

Using webscraping techniques, we will use 'latitude', 'longitude' from properties dataset to access properties and get current data for those locations. THe property description of homes in given region will be stored into a dataset with as many features as in properties dataset we can grab. This dataset can then be used to do some price comparision between properties in 2016 and 2017. Getting data from years prior(say 10 years), we will be able to create trend charts and see market fluctuations.

In [29]:

```
# Load Libraries
from selenium import webdriver
from bs4 import BeautifulSoup


from selenium.webdriver import Chrome

driver = Chrome("C:/Users/safar/Downloads/chromedriver_win32/chromedrive
r")

#with Chrome() as driver:
products=[] #List to store name of the product
prices=[] #List to store price of the product
ratings=[] #List to store rating of the product
# This open the chromium web browser. This web browswer will be under th
e control of this application
driver.get("https://www.zillow.com")

# The field "enter an address will be inspected and filled in for the qu
eries"
```

ZillowMainScreen

## data from API

#### Description

Googlemap API and matplotlib or equivalant will be used to locate properties by zipcode and display them on the map of the Unites States. We will convert 'longitude' and 'latitude' columns in properties dataset to zip code and use the zipcode in the API call.We will show the density of homes sold in various regions in the dataset. We will also show the properties we extracted using webscraping techniques.

In [26]:

```
# This is a sample code and does not pertain to this project. We will tr
```