Downloaded Data from RedFin in CSVs.

Check for duplicates using MLS Listing as Primary Key, since that is unique

**For new construction listings, it didn’t have a MLS # so we deleted those from our dataset.**

<http://www.city-data.com/zips/94102.html>

94102, 94103, 94104, 94105, 94107, 94108, 94109, 94110, 94111, 94112, 94114, 94115, 94116, 94117, 94118, 94121, 94122, 94123, 94124, 94127, 94129, 94130, 94131, 94132, 94133, 94134, 94158.  
  
Read more: <http://www.city-data.com/zipmaps/San-Francisco-California.html>

Ran the data by zip code

**Analyzing Search Engine Results Pages on a Large Scale**

For our project, we reviewed search engine result pages data related to travel. Specifically, our data consists of search terms, like “flights to hong kong” and the various websites, like Expedia & Travel Advisor, that had the quickest search result queries, which is under Search Time.

At first, we attempted to load the data in Postgres but the data included various characters that prevented the import. Therefore, we pulled the data into a Juypter Notebook to analyze the data.

We're going to be sharing a way to get SERP data and have it in a DataFrame (table / csv / excel sheet) for analysis, on a large scale, and in an automated way.

We will be using the programming language Python, so there will be some coding involved.

**Importing the data**

**Handling the data**

We will be using three Python packages for our work:

* [advertools](https://github.com/eliasdabbas/advertools): To connect to the Google CSE API and receive SERPs in a table format.
* [pandas](https://pandas.pydata.org/): For data manipulation, reshaping, merging, sorting, etc.
* [matplotlib](https://matplotlib.org/): For data visualization.

A few notes on the different columns available:

"queryTime" is the time that the query was run (when the request was created). This is different from "searchTime" which is the amount of time it took Google to run the query (usually less than one second). Most of the main columns will always be there, but if you pass different parameters you will have more or less columns. For example, you would have columns describing the images, in case you specify the type of search to be "image".

**The dataset**

We are going to take a look at the airlines tickets industry, and here are the details:

* Destinations: we got the top 100 destinations from Wikipedia and used them as the basis for the queries.
* Keywords: each destination was prepended with two variations, so we will be looking at "trips to destination" and "tickets to destination"
* Countries: Each variation of those was requested for one of two English-speaking countries; The United States, and The United Kingdom
* SERPs: Naturally, each result contains ten links, together with their metadata.

As a result we have 100 destinations x 2 variations x 2 countries x 10 results = 4,000 rows of data.