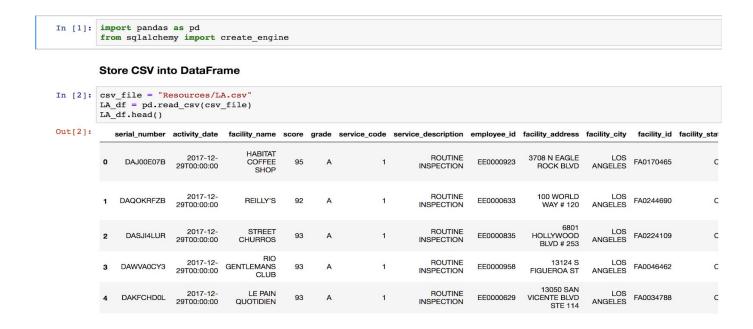
SF & LA Restaurant Inspection Scores

First, we had to decide on what dataset we wanted to use and agreed to do our project on restaurant inspections in San Francisco and Los Angeles. We found our datasets on the Kaggle website.

(https://www.kaggle.com/san-francisco/sf-restaurant-scores-lives-standard and https://www.kaggle.com/citvofLA/la-restaurant-market-health-data)

From there, we extracted the CSV files and used Jupyter Notebook to organize our data by importing pandas, combining both datasets and storing them into a dataframe.



Once we created a dataframe, we created a new dataframe from selecting certain columns we felt were important to keep. The columns we chose are as follows:

	business	city	state	score	date
(The Clift Hotel (Spanish Suite)	San Francisco	CA	100.0	2016-02-16T00:00:00
1	KABABAYAN FAST FOOD	San Francisco	CA	83.0	2016-02-16T00:00:00
2	Extreme Pizza	San Francisco	CA	96.0	2016-02-16T00:00:00
3	Hahn's Hibachi	San Francisco	CA	75.0	2016-02-16T00:00:00
4	Miller's East Coast Deli	San Francisco	CA	75.0	2016-02-16T00:00:00
5	Emporor's Kitchen	San Francisco	CA	NaN	2016-02-16T00:00:00
6	KABABAYAN FAST FOOD	San Francisco	CA	83.0	2016-02-16T00:00:00
7	KABABAYAN FAST FOOD	San Francisco	CA	83.0	2016-02-16T00:00:00
8	China Express Deli	San Francisco	CA	86.0	2016-02-16T00:00:00
•	Miller's Fast Coast Deli	San Francisco	CA	75.0	2016-02-16T00:00:00

business	0	
city	0	
state	0	
score	13935	
date	0	
dtype: int	;4	
combined_d	E=combined_df.dropna(ho	ow='any')
combined_d	f['score'].count()	
98322		
combined_d	f.isna().sum()	
business	0	
city	0	
state	0	
score	0	
date	0	
dtype: int	54	

Business

City State Score

Date

While transforming our data, we had to change column names to keep them uniform. We also ran into the issue of how to drop NaN's in the Scores column. Because the rest of the rows contained data, we could not just use .dropna(how='all'). Instead, we had to use .dropna(how='any'). To make sure there were no more Nan's in our dataset, we used the isna() method.

We then chose to load our data into the relational database, MySQL. Because of the fact that the type of data we have frequently gets updated, a relational database makes sense - it's simple and flexible with organizing data.

We created a MySQL schema to upload our transformed data and proceeded to connect to the local database we created in MySQL called "combined_df." The final table we chose to use for the production database is called "sfla."

We did have difficulty pushing our CSV file to the MySQL table. We had to test the connection to see if it would pull up to the table, but we kept running into not only syntax errors, but operational errors as well.

```
CREATE DATABASE combined db;
 1
         USE combined_db;
                                              2 3 4 • 5
 2
 3

    CREATE TABLE sfla (

                                                   USE combined_db;
 4
          business name TEXT
                                              67
                                               ■ CREATE TABLE sfla (
 56789
          city name TEXT,
                                                    business TEXT,
          state_name TEXT,
                                              8
                                                    city TEXT,
          score INT,
                                              9
                                                    state TEXT,
                                              0
                                                    score INT.
          date DATETIME
                                              11
                                                    date DATETIME
                                              2
10
         );
                                              3
11
                                              4
                                              5 .
                                                   SELECT * FROM sfla;
12 •
         SELECT * FROM sfla;
```

Due to the fact that MySQL defaults to encoding latin-1 we ran into character specific issues from our data. We resolved the error by adding "utf-8" to our connection engine code and were able to complete the project.

Once the data was imported we ran a SQL query to check that all the data was imported successfully.