ETL Project Report

Alejandra Gomez, Raj Kartha, 02/15/2020

**I. Data Subject:**

City of Austin bike share data.

**II. Extract:**

**Data Sources:**

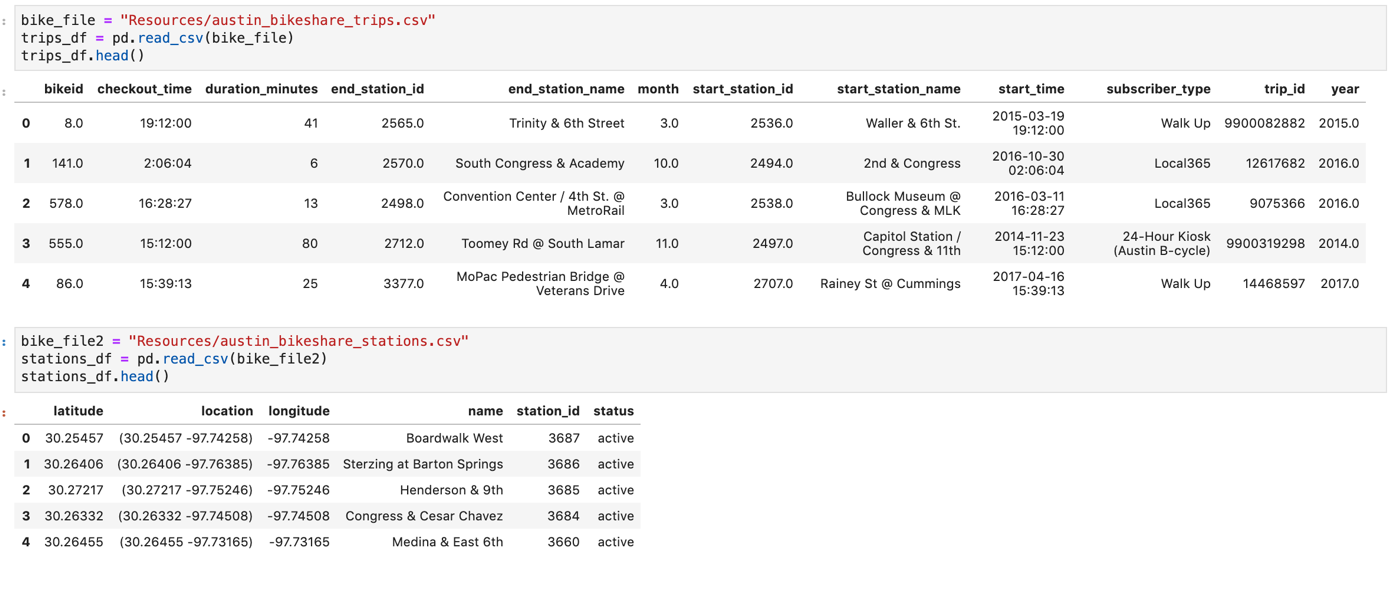
**Kaggle.com** – Austin bike share station data providing status of stations.

**Data.world** – Austin bike share usage data 2013-2017

We used Jupyter Lab and Pandas read\_csv function to bring in the csv files from the two data sources for transformation.

* Although one of our data sources, csv files, provided us with a vast quantity of information on the bike drop-off stations, we lacked information on the status of each station. We wished to observe only the “active” stations for the purpose of our study. Thus, we chose a second data source, csv file, that provided us with this information.

Figure . Image of two data frames (‘trips\_df’, ‘stations\_df’) after initial import process.



**III. Transform:**

With the help of pandas functions in python to transform the data.

* Dropped columns and rows to remove inconsistencies and data not relevant to the topic from both data frames.
* Selected only “active” stations for the purpose of our search, removed “moved” and “closed” stations from column status from stations\_df data frame.
* Rearranged columns to make the data frames less cluttered.
* Removed trailing .0 values in the “year” and “end\_station\_id” columns from new\_trips\_df, as means to facilitate the merge process of the two data frames.
* Merged the two data frames using the pandas merge function, and merged on columns end\_station\_id (from first data frame) and station\_id (from second data frame) field.
* Referenced below (Figure. 1, Figure. 2, Figure. 3) are the data frame images of our project work.

Figure . Image of 1st data frame after clean-up process, ‘new\_trips\_df’.

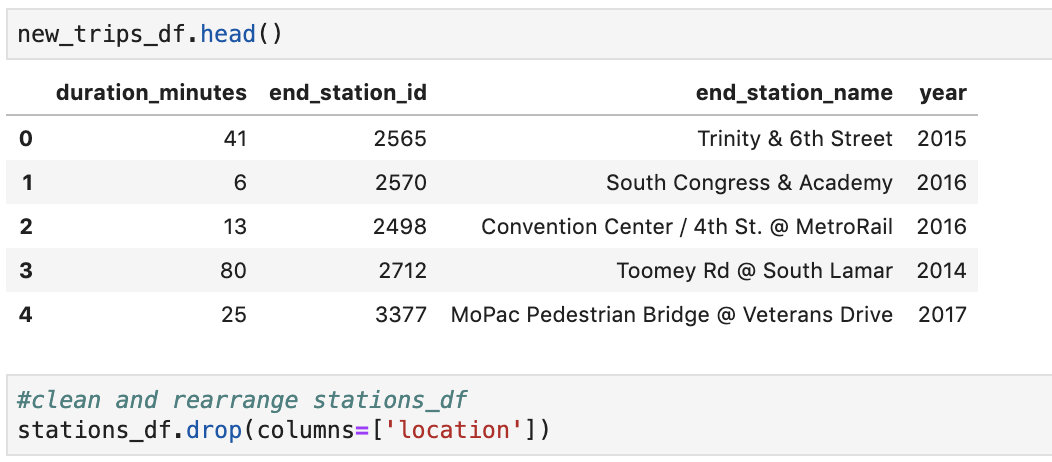


Figure . Image of 2nd data frame after clean-up process, ‘stations\_df’.

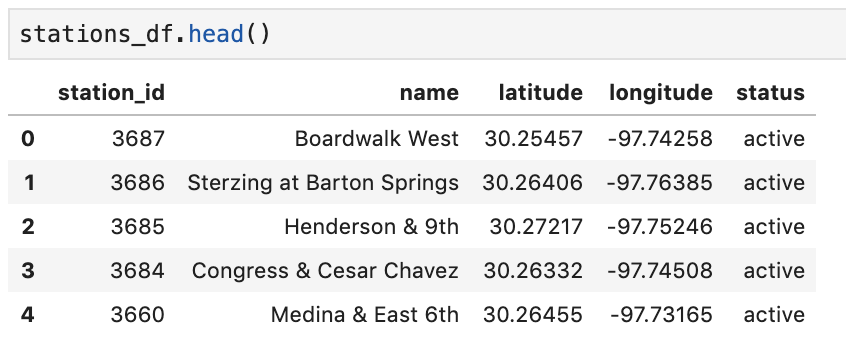
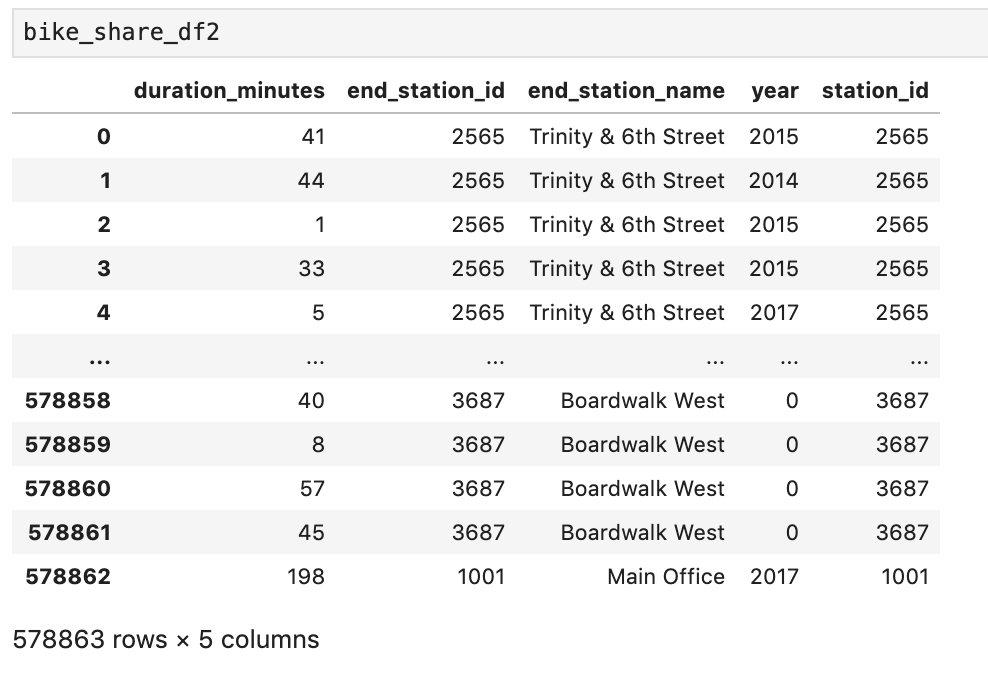


Figure 4. Image of final data frame, after merge process, 'bike\_share\_df2'.



**IV. Load:**

* Created the bike\_share\_df2 database in Postgres with a table to match the columns in the data frame.
* Used the to\_sql function in pandas and the database connection code to transfer and load the merged data frame values into the postgres database.
* Ran a select \* from “tablename” in Postgres to verify the successful load.
* As observed below, from Figure 5, our final table displays in a concise and clean manner, the information relevant only for the “active” stations, which was the purpose of our research study.

Figure 5. Image of table after running "Select \* From bike\_share\_df2" query.

