ETL Project: Technical Report

**Extract**

For this project we used rental and sale data on residential properties in the United States. This data was retrieved from Kaggel.com in .CSV format from the following link: <https://www.kaggle.com/paultimothymooney/zillow-house-price-data>. Even though there were many data files to choose from, we chose the files State\_MedianRentalPrice\_AllHomes.csv and Sales\_Prices\_State.csv as we believed it would be a compelling comparison.

**Transform**

During the transform part of the project we loaded the csv files into a dataframe using a jupyter notebook and the pandas library with “pd.re\_csv”. After the files had been transformed into a dataframe, we looked through everything and decided on only keeping the data from 2019 and the state name. This was made possible by adding another library called numpy, and using the “iloc” function combined with “np.r\_” to only get the columns of data we wanted. Then we wanted to do an average of the sales and rental data we found, so we first added new columns to the dataframe then averaged the numbers by rows, and got the results for what we needed. We then merged the two dataframe into one by using the “pd.merge” function and joined them on common columns such as the state name. After we got everything into one dataframe we wanted to clean and format the columns into sales then rental, and remaining everything for easier reading. We accomplished this by using [[]] for renaming and “df.columns” for reordering. The final result is a single data frame containing states and following it by the sales and rental of each month in 2019.

**Load**

For the load portion of the project we decided to load the combined data frame into a SQL database in PostgreSQL. We created a table to load the transformed data into with a schema made from the site “QDB”. We chose PostgreSQL because we were more familiar with the program and the table creating process. We first made a connection to the postgresql local host server. Pulled all tables in the database then using the “.to\_sql” function we pushed the dataframe into the desired table.

Link to “QDB”: <https://www.quickdatabasediagrams.com/>