ETL Project Report

Extract:

We downloaded a CSV file of the top Spotify Tracks of 2018 from Kaggle.com. Then we searched on data.world for another “top music” dataset that would correlate with the Spotify data. We happened upon a CSV of Billboard’s Top 100 for every week from 1958 – 2018.

<https://www.kaggle.com/nadintamer/top-spotify-tracks-of-2018>

<https://data.world/kcmillersean/billboard-hot-100-1958-2017/workspace/file?filename=Hot+Stuff.csv>

Transform:

After reading the Spotify CSV into pandas, we created a data frame using only the columns “name”, “artists”, and “duration\_ms”. We had to strip the song names to remove the featured artists information. We wanted to only include the song name, so that it would match the data in the Billboard data set. We added the column “spotify\_position”, which used the index value + 1 to reflect the song position on the top 100.

For the billboard dataset, we used the “WeekID” column to drop all data before 2018. We created a new data frame which included the columns “Song”, “Performer”, “WeekID”, “Week Position”, “Peak Position”, and “Weeks on Chart”. Using this data frame, we created a groupby data frame using “Song” as the index. Here we captured variables for the number of weeks, average position on chart, and peak position on chart, by using max, mean, and min functions. Using these variables, we create a new data frame, which we joined with the previous data frame to add the “Performer” column. We then had to drop duplicates using the “Song” column.

We renamed the columns in both data frames to match the columns in our MySQL database.

Load:

Using MYSQL Workbench, we created a relational database that contains two tables, one for each data frame. We decided to use a relational database because we were able to transform the data in both data frames to contain the column “song\_name” with correlating datasets. This will allow us to join the tables easily, while maintain data integrity.

Back in Jupyter Notebooks, we created a connection to MySQL using sqlalchemy. We then loaded the data frames into their tables in MYSQL.

In MYSQL Workbench, we joined tables using the “song” column. Our thought process was that it would be interesting to see how the popularity on Spotify compares to that on Billboard. Are there songs that are more popular on streaming than on radio, or artists who get more radio play than streams? Perhaps there is opportunity for radio stations to find new music to draw in a more diverse audience and grow their listener base.