Technical Write-up report for ETL-Project

1. **The sources of data that we extracted:**
   For our project, we used two different datasets; one in the form of a Spotify playlist, and another from the public platform Kaggle which led us to the csv data on Songs from Billboard 1999-2019. Since the Billboard data contained 20 years worth of data, we decided to focus on just 2019. We searched for a Spotify playlist that contained information which may go pretty much hand-in-hand with the types of data available in the Billboard csv. We found the Spotify playlist named "Top Tracks of 2019" to work on.

2. **Extracting the data**
   Spotify: As we were researching, we learned of the Python library 'Spotipy,' which allows full access to all of the music data provided by the Spotify platform. In order to utilize the Python library, we needed the identifiers of our chosen Spotify playlist to import to Jupyter Notebook and work on cleaning. We used the Spotify Developer Platform to extract the CLIENT ID, CLIENT SECRET, and the authorization URL to establish connection(?). With the proper base url of all Spotify API endpoints set up, we imported the chosen playlist to Jupyter Notebook, using Spotipy.
   Billboard: We used the Pandas function to read the csv file and turn it into a data frame.

3. **Transformation**
   In order to transform the data and use it in our study, we performed the following:
   - Reviewed the files and transformed into data frames.
   - Removed unnecessary columns that were not relevant to the focus of this study.
   - Renamed columns from the Jupyter notebook to eliminate reading errors on Postgres.
   - Wrote SQL queries in a Postgres database to create the tables for each dataset to prepare for the later Loading stage.

   The data was inconsistent, and columns had periods in the names, which caused errors with SQL table creation queries. We renamed the columns from the Jupyter notebook, replacing the periods with underscores.
   From the Spotify playlist database, we selected data that pertained to album information, release dates, track information and artist information.

From the Billboard csv file, we selected artist information, track information, release date, ranks, and the genre.
These columns were chosen with the intention of creating a single table that contains all the information relevant to our goal: Find out if the song trends on Spotify reflect the Billboard chart ranks. The two databases were to be joined on the one commonly-occurring column: the Track Name. We chose to use a postgreSQL database to load our final tables.

4. **Loading**
   With sqlalchemy, we established the connection between the postgreSQL database and the Jupyter Notebook. We had previously used sql queries to create tables for the two datasets in the postgreSQL database. Then, from the Jupyter Notebook, we exported the trimmed data frames to the connected postgreSQL database, where it was joined on the Track Name to create a single table that includes the information relevant to the goal of creating a data lake from what was originally two separate datasets.

5. **Challenges we overcame:**
   - Learned how to work with the Spotipy library
   - Creating table in relation to SQL and Jupyter Notebook
   - Dropping unnecessary data and renaming columns in order to input it into SQL database