ETL Project - Data Vis Bootcamp

**Snehitha Soma, Karissa Malseed, Hamish Cocks**

Introduction

This project required us to use the Extract Transform Load strategy to process and import data into a usable database. For our project we chose to analyse information regarding the cast, crew, and ratings of every movie released by the extremely popular Walt Disney Company from its beginnings to the present day.

Our aim was to find varying types of freely available online data sources that were to then be cleaned using the pandas Python library, and subsequently load these cleaned dataframes into a PostgreSQL database ready for exploration.

Extract

The data sources used in the project were extracted from:

1. Disney movie data was extracted from kaggle.com.au as a json file:

<https://www.kaggle.com/sooaaib/walt-disney-movies?select=disney_movies.json>

1. The data with MPAA/Age Rating was downloaded from kaggle.com.au as csv files:

<https://www.kaggle.com/prateekmaj21/disney-movies>

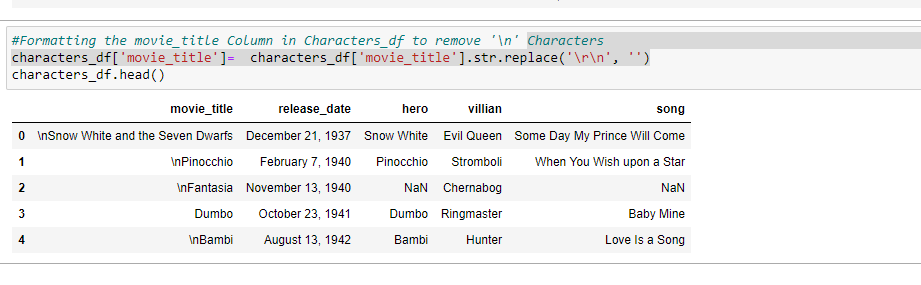
1. The Disney movie characters and voice actors sources were downloaded from dataworld.com.au as csv files:

<https://data.world/kgarrett/disney-character-success-00-16/workspace/file?filename=disney-characters.csv>

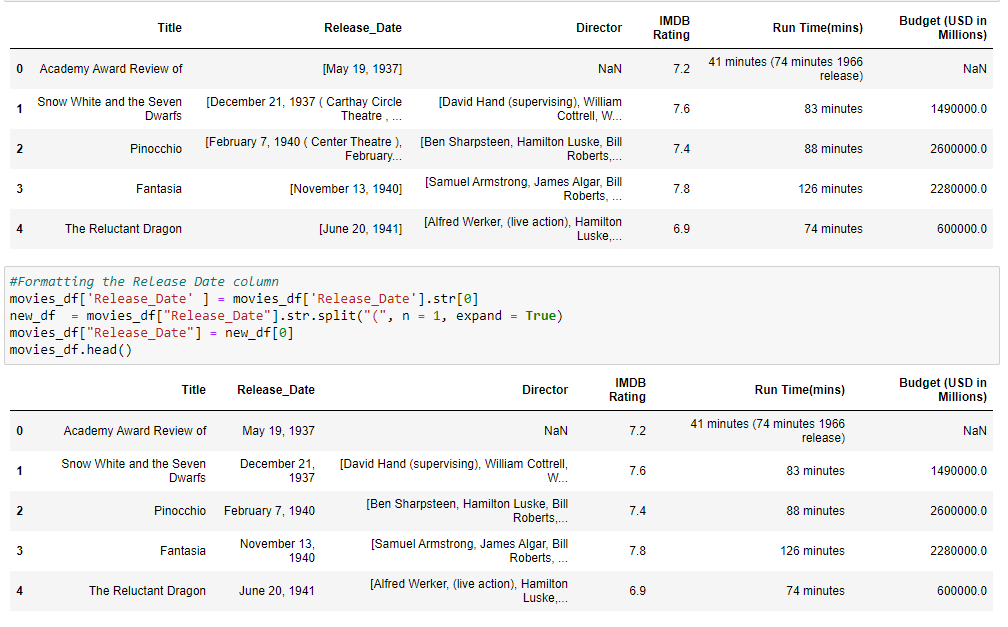
Transform

The extracted data was retrieved into jupyter notebooks for a series of transformations, clean ups.

The specials characters(/n) were removed from the movie title columns.



The Release date column was cleaned by extracting the first index element from the list.

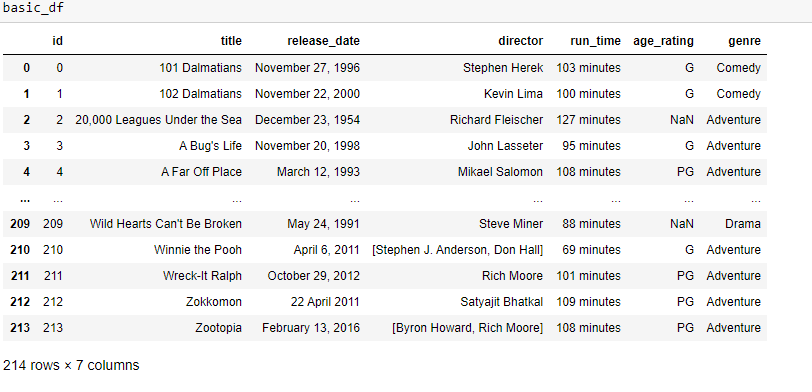


All the movie tiles with improper release dates were dropped.

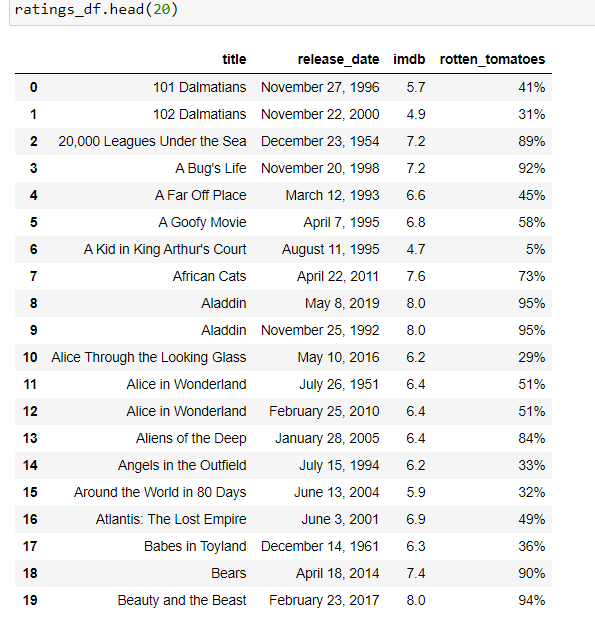


The cleaned data frames were then merged accordingly to create four tables to hold relevant data.

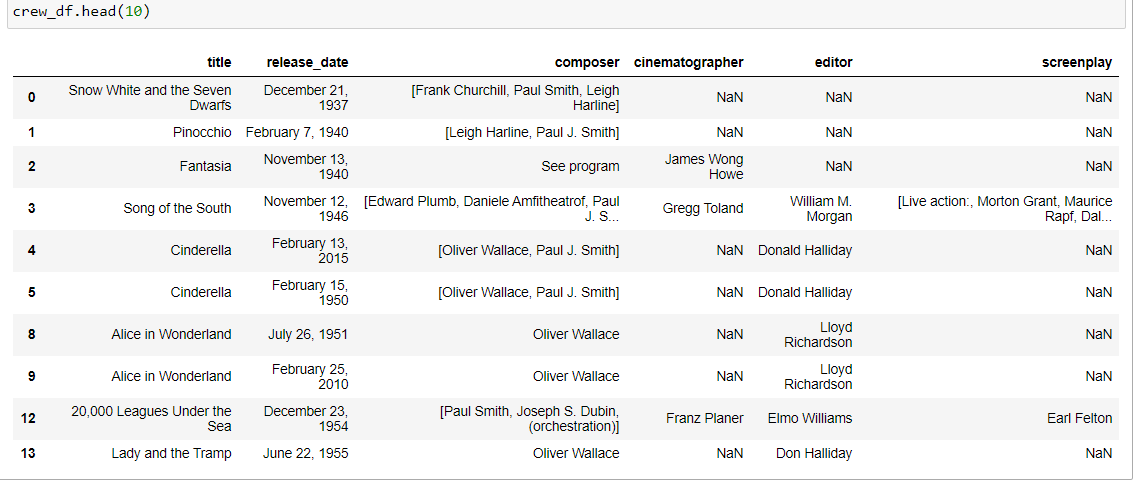
The ‘Basic’ table holds brief information about each movie.



The ‘Ratings’ table holds data about review/rating of each movie.



The ‘Crew’ table holds information about various technicians worked on every movie.

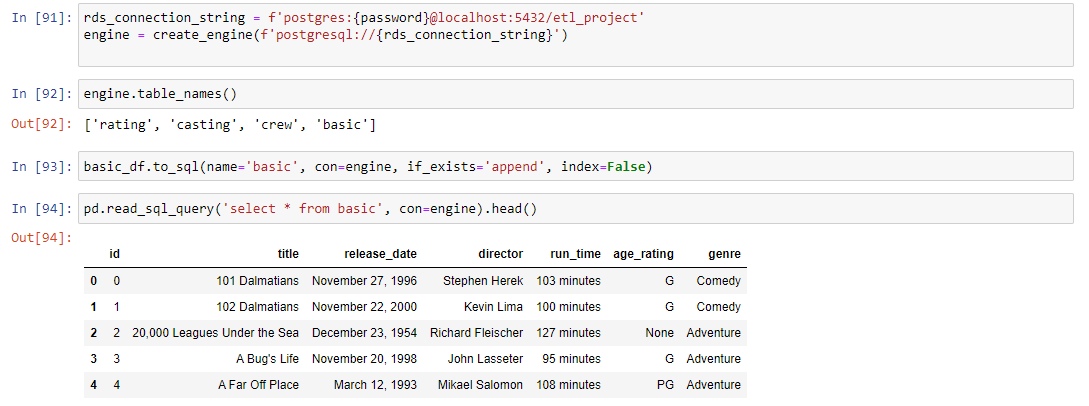


The ‘Cast’ table shows various characters in the movie and voice actors of those characters along with the character names of main leads(Hero and Villain).



Load

The PostgreSQL table schema was defined in PgAdmin using an SQL query file based on the below Entity Relationship Diagram. A connection to PostgreSQL was set up within the Jupyter Notebook using the SQLAlchemy function ‘create\_engine’. The pandas function ‘to\_sql’ was then used to read the dataframes into the afore-mentioned schemas of the SQL database. Another pandas function, ‘read\_sql\_query’, can then be utilised to carry out SQL queries on the schemas.



NOTE: users must alter the ‘config.py’ file within the Github repository with their own PostgreSQL password in order for the connection to be established.

Conclusions

We were successful in our goal of employing ETL to a set of data. We did, however, run into some challenges and roadblocks along the way and can share the following observations:

* Picking a suitable dataset turned out to be more demanding than initially thought. Much of the data surrounding movies in general was often massive, and handling such big datasets seemed infeasible for this project. Furthermore we wanted to prioritise merging datasets from different sources rather than being handed one massive set with everything we need.
* As you can see from the above Transform section it was evident that the data was not in an ideal condition. Many of the columns were not consistent in their formatting, for example we ran into 3 or 4 entries within the ‘release\_date’ column that were not in the same format as the rest of the entries. While it was an easy fix we did not pick it up right away and consequently had to backtrack. It was a good learning experience nonetheless as we were too trusting of the data source.
* During our efforts to load the dataframes into PostgreSQL we struggled with the SQL syntax and realised that SQL automatically renders all column and table names to lowercase which consequently did not match up with the dataframe column names. Similarly we had trouble defining a primary key within the database. Initially we defined Title as the primary key, but as there were remakes of the movies included in our tables we had to switch to Release Date as the primary key.

In conclusion it is evident that ETL is an extremely important method in developing databases. Generating a user-friendly and intuitive database is often the key to many companies' inner workings and therefore gaining competence in ETL should be high on the list of any budding data analyst.