ETL Project Final Report

6/8/2021

**Team members:**

Chuck Youngman, Felix Ogbodu, Sindhura Surapaneni, Al Burroughs, and Supraja Koppisetty

**Project Description/Outline:**

We will combine two datasets to create a summary that shows all the movies in theatres and the streaming platforms. The final database will be relational. The final tables for production will be ‘online’ and ‘theater’. The transformations used on the project are Cleaning, filtering, and joining. The title for the final database is ‘movies’.

**ETL Project steps:**

**Extract data**

1. Downloaded two csv files from Kaggle.com website:

[Movies on Netflix, Prime Video, Hulu and Disney+ | Kaggle](https://www.kaggle.com/ruchi798/movies-on-netflix-prime-video-hulu-and-disney)

1. <https://drive.google.com/file/d/1IFDKr1Dv4iIhaAEQb29tQ8CmOWyVenQT/view>

(The original data was formatted in csv)

1. Load files into git hub repository

**Transform data**

1. Import requirements:

import pandas as pd

from sqlalchemy import create\_engine

import numpy as np

(using jupyter notebook)

1. Read ‘Movies On Streaming Platforms updated 2.csv’ file into pandas df
2. Create df column headers
3. Replace spaces in columns headers (Rotten Tomatoes , Prime\_Video) with underscores to ensure dataset import into postgres table
4. Replacing infinite datapoints with nan.
5. Dropping all the rows with nan values in the given pandas data frame
6. Remove % from Rotten Tomatoes column
7. Printing df (confirmation checkpoint)
8. Read ‘36238\_IMDd-Movie-Database.csv’ file into pandas df
9. Convert the Release Date to datetime
10. Add a column for Year
11. Remove day and month from release date and place year data in newly create column for the release Year
12. Print the data frame (confirmation checkpoint)
13. Replace spaces in column headers (Lead Actor, Director Name, Lead Actor FB Likes, Cast FB Likes, Director FB Likes, Movie FB Likes, IMDb Score (1-10), Total Reviews, Duration (min), Gross Revenue) with underscores to ensure dataset import into postgres table
14. Replacing infinite datapoints with nan.
15. Dropping all the rows with nan values in the given pandas data frame
16. Drop duplicates
17. Printing df (confirmation checkpoint)
18. Confirm ‘Online’ dataset
19. Confirm ‘Theater’ dataset

**Load data**

1. Connect to local postgres database
2. Check for database postgres database tables
3. Use pandas to load online csv converted Data Frame into ‘theater’ database
4. Use pandas to load theater csv converted Data Frame into ‘online’ database