ETL Project

**Objective:**

Exploratory data analysis to understand if the income level of the country to which the artist belongs to has any relation to the probability of them winning an Oscar.

**Extract Data:**

The data was obtained from 3 different sources

1. Oscars data from Kaggle in the form of csv. - [link](https://www.kaggle.com/fmejia21/demographics-of-academy-awards-oscars-winners)
2. Country iso code data from Wikipedia page through web scraping as a Pandas Dataframe - [link](https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2)
3. Country economic details data was extracted from the world bank data API as a Json object in Pandas. - [link](https://datahelpdesk.worldbank.org/knowledgebase/articles/898590-country-api-queries)

**Transform Data:**

1. Oscar Data from Kaggle. The following transformations were conducted on the data obtained from the above sources.
2. All the transformations were completed using python in a Jupyter Notebook using Pandas.
3. Drop unwanted columns. Keep only relevant columns that make sense.
4. Reordered the columns as Movie, Person, Award, Year\_of\_award, Birthplace, Data\_of\_birth and Biourl.
5. Using the replace method we removed the ‘\_’ in the column names.
6. Split the full name of the actors/actresses/directors into two separate columns of first name and last name using the split method.
7. Split the date of birth for the person and extract only the year of birth using the split method.
8. We observed some of the years were only two digits, hence we added a prefix of ’19’ to make all the years in the format YYYY. This was completed using the lambda function in the apply method.
9. The place of birth column which contained the city and country was split to extract only the country of birth as a separate column using split and then the apply method to get the last element of the list as the country.
10. We observed that for United states, the name of the state was mentioned instead of country hence we changed it to reflect as United States wherever the country column had only 2 letters. This was done using function lambda.
11. Countries Scotland, England and Wales do not have a specific Iso code hence we had to change them to United Kingdom.
12. Similarly, Czechoslovakia was changed to Czechia as that is current name of the country with an Iso code.
13. Country code Data from Wiki.
14. Extracted the scraped data as a html table from the Wikipedia page.
15. Select the relevant table which contains the country iso codes.
16. Drop unwanted columns and rename the columns as required.
17. The country names Moldova, Taiwan, Russia, United Kingdom were not consistent hence it was transformed to reflect uniform country names.
18. Since the Oscar dataframe had Canary Islands as a country we added the country and corresponding Iso code to the data as it was missing.
19. Country API data
20. Obtained the base URL for the API query.
21. Ran a for loop through the previous wiki data frame to obtain the Region, Income level and Capital city for each country from the Json returned by the API.
22. Stored the info obtained in a dataframe.

**Load Data:**

1. Created a new database in postgres SQL.
2. Created 3 table schemas to match the structure with the dataframes from pandas.
3. Loaded the dataframes from pandas into postgres SQL by establishing a connection and ‘to\_sql’ method from Pandas.

**SQL Analysis:**

1. Joined (left) the Oscar, Country codes and Country API tables to display for each award winner the movie, the award they won and the income level of the country they belong to.
2. Obtained the number of awards per income level using the Group by function. We observed that majority of the awards were won by artists from High Income level countries.