**## Project Report**

At the end of the week, your team will submit a Final Report that describes the following:

\* **\*\*E\*\***xtract: your original data sources and how the data was formatted (CSV, JSON, pgAdmin 4, etc).

\* **\*\*T\*\***ransform: what data cleaning or transformation was required.

\* **\*\*L\*\***oad: the final database, tables/collections, and why this was chosen.

**ETL Project: Economic Freedom**

**Group members:** Sagi, Jake, Frances

**Project purpose**

The purpose of our project was to extract data that would enable us to answer the following questions:

How does the population density of a country affect the economic growth over the years?

How does the marginal tax rate impact a country’s economic freedom score?

**Extract**

Our data sources were:

[Countries of the World](https://www.kaggle.com/fernandol/countries-of-the-world) (<https://www.kaggle.com/fernandol/countries-of-the-world>) and

[Economic Freedom of the World](https://www.kaggle.com/gsutters/economic-freedom): (<https://www.kaggle.com/gsutters/economic-freedom>)

Both of the files were csv files. These files were read into our jupyter notebook.

**Transform**

The following steps were taken to clean the data:

One csv file had a trailing space after the name in cell; this prevented the merging of the two data frames on the Country key. The following code resolved this issue:

countrydf['Country'] = pd.core.strings.str\_strip(countrydf['Country'])

Several columns within the dataframe were renamed allowing the files to be merged and making the column headers more legible. The following code was used:

ecodf.rename(columns={'countries':'Country','ECONOMIC FREEDOM':'Economic freedom', '1d\_top\_marg\_tax\_rate':'Top marginal tax rate', '4c\_black\_market': 'Black market'}, inplace=True)

Several rows had missing data. To resolve this we selected the columns to be viewed for missing data. The following code was used:

ecodf\_copy = ecodf\_copy.dropna(subset=['year', 'Country', 'Economic freedom', 'rank','Top marginal tax rate'])

**Load**