**ETL-project for Group 3**

Group members: Dainty, Mabel, Sonny and Arthur

Group 3 Proposal - updated Wednesday, Jan. 27, 2021

Overall, we have finished the project.

DATA SETS:

We’re using this population data from Kaggle: https://www.kaggle.com/utkarshxy/who-worldhealth-statistics-2020-complete?select=alcoholSubstanceAbuse.csv

And this US Real Estate Listings data from Kaggle: https://en.wikipedia.org/wiki/List\_of\_countries\_by\_GDP\_(nominal)\_per\_capita

EXTRACT:

We are going to download the first file as CSV (Arthur, 10 minutes)

Then we'll scrape the second data source and extract country and per capita GDP. (group, 45minutes)

CHALLENGE:

We'll use pandas (read\_csv) to import the first CSV file (Sonny, 20 minutes)

## We downloaded the csv file and loaded into pandas as dataframe in less than 10 minutes.

And use MongoDB to scrape the second datasource

## we used pandas (instead of MongoDB) to scrape the per capita GDP from wikipedia, also in much less than 45 minutes.

TRANSFORM

Once imported, we will join the tables on country (Dainty: 15 min)

## the table join didn't take much time, although we decided to clean the data first before joining them.

Rename all columns based on the indication from the first datasource and drop the columns that we didn't find meaningful.

## it turned out that this part took the most of the time, much longer than we thought. We first tried to extract the first instance of any country (for whatever reason, each country has 6 records) using .loc where we got stuck and Sean helped us figured out. We then tried to group the country and average the three-year per capita GDP and found that there are string values in the supposedly numbers/floats column. We spent quite bit of time trying to drop the two values that are not numbers and finally did it.

## after that the joining tables part became really easy.

LOAD

We'll do this as a group activity, and load the cleansed dataframe into Postgres database. (Mabel and group assists, 45 minutes)

## among different data types of the pandas dataframe and tables we created in postgres, there were some struggles, but we were able to change data types and successfully loaded the dataframes into postgres, roughtly with the time we anticipated.