Amber Moran

Sam Lin

Katherine Huynh

Team ASK ETL Report

Extract:

All of our data came from WorldBank.org. We used 2 CSV files and we also retrieved some data using the web API of WorldBank.org:

* CSV files: <https://data.worldbank.org/indicator/>
* API: <https://datahelpdesk.worldbank.org/knowledgebase/articles/889392-about-the-indicators-api-documentation>

The API gave us information on temperature and rainfall for a couple of different time periods (1960-1979 and 1980-1999). A limitation of this API is that it can only pull data for one country at a time. Therefore we chose to use five countries: United States, China, Brazil, Germany, and India.

The first of our two CSV files gave us information on greenhouse gas emissions (by kilotons of C02) for all countries from the year 1960 to 2018.

The second CSV file gave us information on the population of all countries from the year 1960 to 2018.

Our Jupyter Notebook file has some data about income, which was part of a metadata file that came with the population CSV. Although we transformed this data, we did not end up loading it because we realized that we did not have enough information on how the data was gathered and it would not be appropriate to use in an analysis.

Transform:

The temperature data that came to us from the API was initially separated by country and year range (1960-1979 and 1980-1999). For each year range in each country, we had the average temperature for each month. All of this temperature data was put into one data frame, and from here we narrowed down the data by taking the mean of the average temperatures each month from 1980-1999. From here, we took these means and placed them into a new frame with the countries as the columns, we then transposed the columns and reset the index so that the country names appeared in their own column called “country\_name”. We repeated this same process for the rainfall data that also came from the API.

The greenhouse gas data from the CSV initially had some unnecessary columns that were dropped, such as the country code, the indicator name, and the “Unnamed” column, all of which were dropped. We then located the names of the five countries that we gathered data from in the API call. There were many ‘NaN’ columns, since no data was entered from 1960-1969 and from 2013-2018, so these columns were dropped. Because our API data only gives us information from the years 1980-1999, we also dropped the columns for any years that fell outside of this range. The “Country Name” column was renamed “country\_name” to match the columns for the temperature and rainfall data frames, and the year columns were renamed “year1980”, “year1981”, etc. A similar process was followed for the population data from the other CSV, though there was a lot more data entered into this CSV, so dropping the ‘NaN’ columns was not necessary.

Load:

After the transformation was complete, a database was created in MySQL called climatechange\_db. Next, the schema was completed by creating four tables to represent the transformed data frames. The table names are: population\_final, green\_house\_gas\_final, rain\_averages\_final, and temp\_averages\_final. The index was removed from the data frames so the values would load into the tables. Finally, after the data successfully loaded, queries were written to see the contents of the tables and a general query was added to show how all the tables join together.