Harper Hahn

Data Analytics Bootcamp

Date: March 7, 2020

ETL Project Report

This project analyzes the occurrence of SARS (2003), TB (2007), and H1N1 Virus (2009) in different countries around the world.

**Extract:** The data sources for this project are taken from Kaggle.com. All three datasets were extracted in a CSV format.

**Transform:** Basic cleaning took place in Excel (i.e. reformatting dates), but Pandas was the primary tool for data clean up and transformation. After uploading my CSV files into Pandas, I had to start by figuring out which columns I wanted to keep from each CSV file and rename their column headers; ‘country’, ‘year\_(illness)’, ‘number of cases’ and ‘number of deaths.’ After this, each dataset showed the values for the county, year, number of cases and number of deaths where the presence of illnesses were documented. When I configured the columns, I could then merge my CSV files on ‘country’. The result will show the number of cases, deaths a country and the year each country had experienced outbreaks.

**Load:** I connected to PostGres using SQL-Alchemy, a relational database, for the data loading component. I needed to create a table within my database before making a connection between PostGres and Pandas. After the table was generated successfully, I could move on to establishing a connection between Pandas and PostGes. When the connection was made, I made a query within PostGres (SELECT \* FROM pandemics\_data;) which displayed my cleaned Pandas data frame in SQL. I also performed the same query from within Pandas, which is shown within my project notebook. I chose PostGres because I wanted the user to be able access country, disease, or case-specific data, and SQL enables the user to perform queries and retrieve the desired information.

My main challenge was encountering the “case sensitive error” when connecting my final Pandas data frame to PgAdmin. I had to reset my index within Pandas and remove all capital letters, numbers, and parenthesis from my column headers in order for my data to connect properly. After doing this, the connection was successful. I also struggled with whether I should include the year of each occurrence. I ultimately decided it was necessary to include because without including the dates, all the user sees are the vast amounts of what appears to be duplicate data. Including the year of each illness gives the user more information as to when the illnesses produced their number of cases/deaths and in which countries.