ETL Project

Introduction

Our project focuses on the impact of COVID-19 on the global economy with focus on Canada. We used Pandas, Web Scraping, and SQL to perform our ETL process. The data sources used for extraction included StatsCan, CNN News, Johns Hopkins Github, and Yahoo Finance.

Method - Extract

CNN_COVID_Timeline: A timeline of key events related to COVID-19 was scraped from a CNN website using beautiful soup. Dates and news story headlines were extracted from the website and added to lists using python.

Unemployment: Unemployment data files are imported from Stats Canada in csv format. These data files only show two months of unemployment data. The CSV data files are cleaned for the specific columns containing monthly unemployment rates by geography (Canada and provinces).

CERB: To obtain CERB data by genders, we first created a BeautifulSoup object to retrieve the webscript of the table section, then the CERB dataframe was created with pandas.read_html function.

Method - Transform

CNN_COVID_Timeline: Pandas was used to combine the date and news story lists extracted from the CNN timeline into a dataframe. The data was also formatted to include the respective year for all data for consistency. Pandas was also used to change the date from string format into “datetime” format.

Unemployment - The data files are merged into one dataframe to show 12 months of unemployment data together by province. Transpose was used to have Provinces in columns and GroupBy Month. To extract month & year from the date column, pd.datetime and dt.to_period

Demographics - Stats Canada Provides Demographic data based on total infections between demographics and hospitalizations rates so pulling all this data. Stats Canada data files are merged into one dataframe using the pd.merge function. We then transposed data by dropping the columns we didn't require and were not critical to our table. To further clean the data we renamed columns as per our table and only kept the columns that were to be loaded.

COVID-19 Cases (Global) - COVID case data extracted from Johns Hopkins was formatted in pandas by removing additional columns and renaming columns. Data was formatted into fewer columns using transpose and stack functions. Dates were converted into datetime format and the groupby function was used on the country name to remove duplicates.

COVID-19 Cases(Canada) - COVID case data extracted from Johns Hopkins was formatted in pandas using iloc to isolate for Canada. Additional rows of data were removed as they did not relate to Canadian provinces (i.e. cruise ships). Data was formatted into fewer columns using transpose and stack functions. Dates were converted into datetime format.

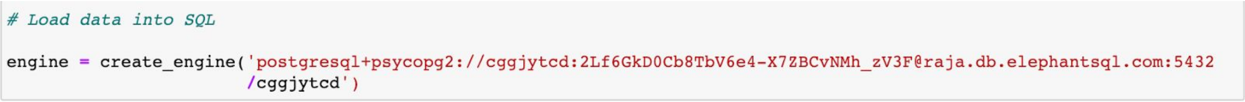
Stock Prices - Stock prices in the past year are downloaded as CSV files from Yahoo Finance. We first created four independent dataframes with Pandas. Some unnecessary columns are dropped before all data frames are combined to the final one. Finally, we merged four data frames to the final stock price data frame that is ready to load to SQL server.

CERB - The original dataframe contains two index levels; we removed multiple index levels in the original dataframe to make sure it complies with the table format in the SQL server. Next, all percentage columns are dropped from the dataframe; only gender number columns are kept in integer format. Finally, all columns are renamed to match the database column names.

Method - Load

First, we created our data tables in PostgreSQL using the “schema.sql “ file.

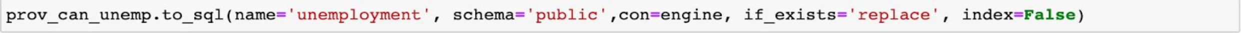
We used PostgreSQL (ElephantSQL) to load the formatted data tables from jupyter notebook (python) into our database. We created an engine using our ElephantSQL server.



```
# Load data into SQL

engine = create_engine('postgresql+psycopg2://cggjytcd:2Lf6GkD0Cb8TbV6e4-X7ZBCvNMh_zV3F@raja.db.elephantsql.com:5432
                       /cggjytcd')
```

After the connection was established, we pushed our data tables to our database.



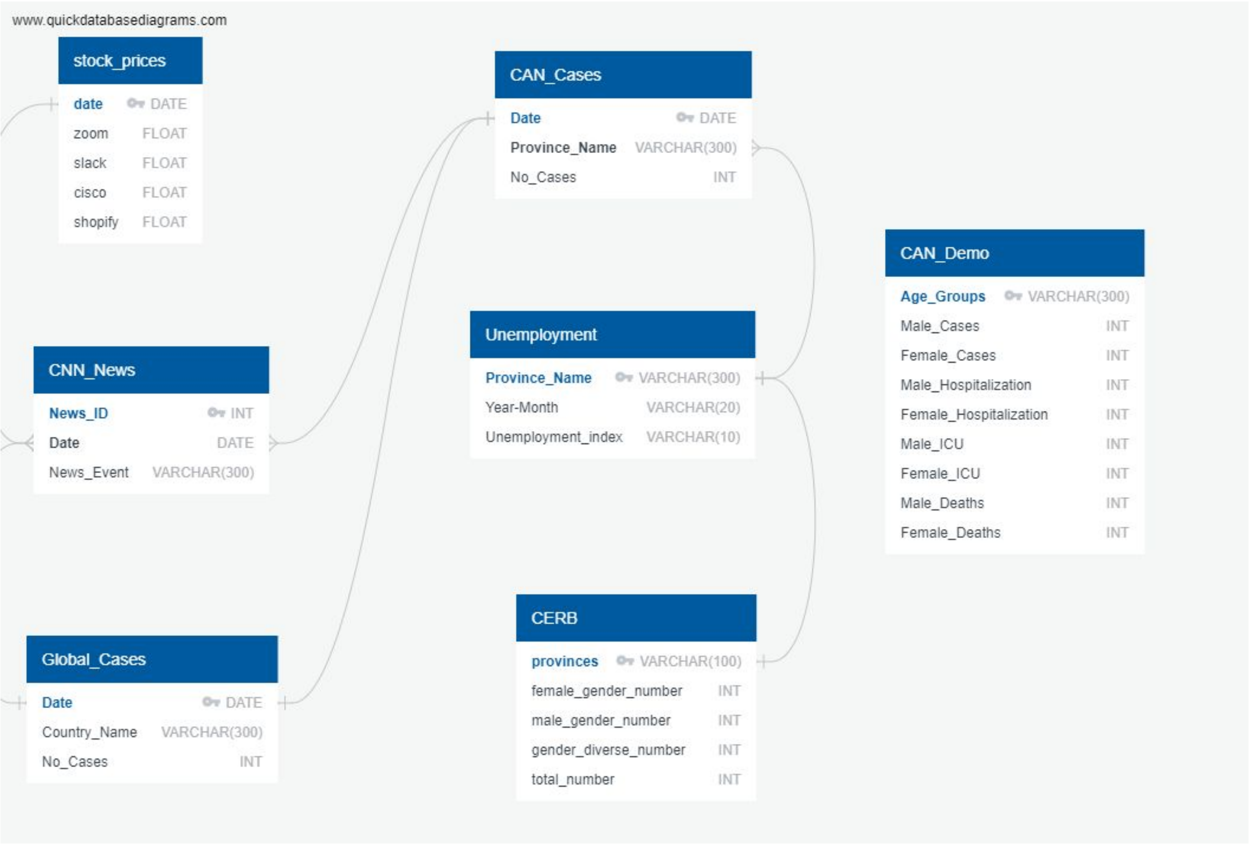
```
prov_can_unemp.to_sql(name='unemployment', schema='public',con=engine, if_exists='replace', index=False)
```

Discussion/Limitations/Next Steps

Connection Limitation (Server): We faced many issues in regards to creating tables and uploading our tables to a public server. This was caused due to the limitation of connections allowed to the public server. We had to take turns in uploading our data or assign one individual to upload our data to avoid this problem of many connections. Moving forward, we have been discussing whether to use a different service or pay to upgrade the number of connections we can make at one time.

Demographic Data (Relationships): We faced issues finding relationships for our demographics table with our other tables. We felt the demographics data was critical information, so for now, we have kept the table without any relationships as useful information. Moving forward, we will try to further clean up the data and find relationships with other tables to make demographics data fit in.

ERD Diagram



stock_prices	
date	🔑 DATE
zoom	FLOAT
slack	FLOAT
cisco	FLOAT
shopify	FLOAT

CAN_Cases	
Date	🔑 DATE
Province_Name	VARCHAR(300)
No_Cases	INT

CAN_Demo	
Age_Groups	🔑 VARCHAR(300)
Male_Cases	INT
Female_Cases	INT
Male_Hospitalization	INT
Female_Hospitalization	INT
Male_ICU	INT
Female_ICU	INT
Male_Deaths	INT
Female_Deaths	INT

CNN_News	
News_ID	🔑 INT
Date	DATE
News_Event	VARCHAR(300)

Unemployment	
Province_Name	🔑 VARCHAR(300)
Year-Month	VARCHAR(20)
Unemployment_index	VARCHAR(10)

Global_Cases	
Date	🔑 DATE
Country_Name	VARCHAR(300)
No_Cases	INT

CERB	
provinces	🔑 VARCHAR(100)
female_gender_number	INT
male_gender_number	INT
gender_diverse_number	INT
total_number	INT