All of the datasets in this project were loaded to a mysql database utilizing a connection available in the sqlalchemy library. After loading to the database each dataset was queried to confirm the load occurred correctly. Below are the descriptions of the processes followed for the 3 data pulls.

**Census API**

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

An API from <https://api.census.gov/data/timeseries/qwi/se> was used with an API key. The data pulled was in a JSON file format. When the JSON file format was pulled, it pulled back the data using a list. To test the data, I pulled 2017 data and created a variable named “data2017q1”. Once that was created, I then pulled in the index data for the row, and then for the column. The scope of the project was to pull data for years, and I chose 2010 to 2017, and since the data variables were set up in quarters, I chose Q1 to Q4. In addition, I created two variables; years, quarters and put them in a list. Once that was completed, I needed to use a for loop in order get all of the years and quarters from the census data to bring back (2010 – 2017), and (Q1-Q4) data. The next step was to create a data frame using the url + api key, and it brought back the list in a pandas dataframe.

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

The data brought back state and county data in code format. In order to bring back the counties in scope which were; Winchester, Staunton, Blacksburg, Virginia Beach, Charlottesville, Lynchburg, Richmond, Roanoke, Harrisonburg I had to go to another section of the Census Bureau site and look up the county codes that aligned to the cities that matched the cities from Bureau of Labor & Statistics. The county codes had to be converted; therefore, I first printed the data frame showing the codes, and only pulled the codes in scope. The code then had to be converted into the city names and then printed into a data frame. In order to get the codes, I used a .loc to locate the county codes and then converted and reprinted them into a new dataframe.

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

I chose to use an API key so that our project would prove our abilities to extract codes in various ways we’ve been trained to. My team member’s responsibilities were to convert and read an Excel .csv file, Webscrape a database; therefore, we wanted to add another example of how to collect API data, and it is NOT EASY!!!

**Indeed Web Scrape**

Indeed.com Web Scraping

Extract: Available jobs in Virginia were scraped from indeed.com from a series of

cities identified by the Bureau of Labor and Statistics site of relevant cities. These

cities were looped through with the requests library and BeautifulSoup library for 50

pages, totaling 500 record per city. Only records with both a job title and location

were used.

Transform: Both the job title and location where stripped of return characters from

the respective parts of the div locations. The job title was cleaned with two

functions. The first function cleaned emojis and other odd special characters. The

second function cleaned numbers and other special characters. The location was

split at the ‘,’ character into two pieces and the first piece was then reassigned to

the location variable.

Load: Each complete entry with a job title and a location were pulled into a list,

identifying the job. That job is then pulled into a list, creating a list of lists. This was

then converted into a pandas dataframe with columns identified and matching the

Bureau of Labor and Statistics sheets

**BLS CSV**

Extract:

The data was taken from the Bureau of Labor Statistics website, specifically

employment data for specific cities/regions in Virginia from the years 2009 to 2019.

The data came in as .xlsx files.

Transform:

The 9 separate .xlsx for each city had to be combined for easy comparison. With

.xlsx files, the extraneous headings had to be skipped and display only the data. To

differentiate the data when they are combined, the column ‘cities’ was added.