ETL Project Summary

By: Chelby Ryerson & Natalie Crow

Submitted 3/9/2020

EXTRACT

The data used in this project is titles and sourced from the following sources:

- General Hospital Information: https://www.kaggle.com/cms/hospital-general-information
  - Provided a census of hospitals within the United States and southern North American region
  - Includes name, location, type, and ownership characteristics for each hospital
- Hospital Value Based Purchasing Efficiency (HVBP) Scores: https://catalog.data.gov/dataset/hospital-value-based-purchasing-hvbp-efficiency-scores-7b9f0
  - Contained metrics by hospital for Achievement, Improvement, and Measure Scores related to HVBP
- 2010 Census: https://www.kaggle.com/census/us-population-by-zip-code
  - Census data acquired to query population by zip code of hospitals in the above datasets

Each data set was exported in CSV format

TRANSFORM

To review and clean the data sufficiently for querying, the following steps were made:

Hospital Information:

1. Dropped unwanted columns, kept columns A through K
2. Filtered hospitals to only include US based hospitals
   a. Countries such as Guatemala, Virgin Islands, and Puerto Rico were removed, among others
3. Column names were altered to remove spaces and be uniform across each dataset
4. Checked for null values among the remaining data—none found

Hospital Value Based Purchasing (HVBP) Efficiency Scores:

1. Hospital scores provided in this dataset were in the form of strings similar to “6 out of 10” which was converted to the score only (6), result of which was placed in new columns
   a. Conditional for-loop utilized to filter out any “10 out of 10” scores to reflect “10” instead of “1”
   b. Original string score columns then dropped
2. Column names were altered to remove spaces and be uniform across each dataset
3. Checked for null values among the remaining data—none found

Census Data:

1. Reduced the file to zip code and population data columns

2. Dropped rows that contained population less than 100 with suspicion of inaccurate data in those instances
3. Dropped any null/missing values

LOAD

1. Created SQL tables for each individual data set
    a. **Relational** database was utilized due to the consistency of the data available, with no null fields
2. Query executed through python to join the data sets:
    a. Matched the unique Facility_ID within Hospital Information and HVBP Score tables;
    b. then by zip code in Hospital Information and Census tables