ETL Project Report – Scott Cerny

I decided to create a database of Major League Baseball player statistics from the 2015 season.

Extract:

My data was extracted from kaggle (https://www.kaggle.com/danielmontilla/baseball-databank). I chose 4 different CSV files from the site that contained different types of MLB player information (Batting.csv, CollegePlaying.csv, People.csv, and Salaries.csv).

I wanted to be able to show more than just the traditional baseball statistics in the database, so in addition to the traditional stats I also added items like salary, which college they attended, their birth country, and more.

Transform:

After importing the 4 CSV files into Pandas dataframes, I had to clean each one. The first thing I did was to filter by the 2015 season for the two CSV files that were specifically tied to individual seasons (batting and salaries). I also had to remove a bunch of columns that I didn't care about for each dataframe. I also had to remove any duplicate entries in the college dataframe as I only wanted one entry per playerid (the CSV had an entry for each separate year of college they attended). I also had to rename some of the columns to match my database schema.

Once I had my dataframes all configured and ready to go, I moved on to loading them into a database.

Load:

I decided to go with a Postgres SQL database as I knew SQL would work better for this data than a non-SQL and I was already most familiar with Postgres.

In Postgres, I created a database called baseball and then entered my schema to create the needed tables (batting, college, salaries, and people). I was able to link the tables using the playerid field found in each one as the primary key.

I then connected to the database via Pandas and sent each of the 4 dataframes to their 4 matching tables in the baseball database.

After the import, I was able to run queries on the database and get the information I wanted using as many of the 4 tables as needed for that particular query.