David Grace and Anthony Giusti

ETL Project – NBA

08/10/2019

***Extract***

Our ETL project revolved around the NBA, specifically, the 2017 season. We used a CSV that contained 2017 Player Data. This included their biography information and statistics for the entire season such as points, assists, rebounds, etc.

Our second source of data was *extracted* from an imported NBA-API Module which was pulled from NBA.com. This dataset gave us access to NBA data that can be broken down by team, season and individual games. Given that there was such a huge amount of information, our *transformation* step was vital to filter relevant information.

***Transform***

To normalize our data, we *transformed* our sources through jupyter notebook. Our first step was to read the 2017 csv and display it as a data frame so we could delete unnecessary columns, filter, and create a player\_id index that would be relevant to our future data frames. From the CSV, we would create 2 data frames. The first being a brief player bio that included player name and position. The second, containing their 2017 statistics. For each table, we had to drop unnecessary columns and rename column headers to join with future tables.

From the NBA-API, our *transformations* required a little more effort to create 2 additional tables. While these datasets came from the same API’s they were contained in different lists that required different methods to clean the data. For our teams table, we had to drop unnecessary columns to provide the bio information of all 30 teams. In our last data frame, we looked at a game list that included not only nba games but also included affiliate leagues (D-League, G-League, WNBA). For this set we first had to filter out to only feature the standard NBA games. For our next filter, we had to filter again just for the 2017 Regular NBA season.

***Load***

We decided to use Postgres as our final target database. Once we were able to make a connection, we could query our tables and analyze players, games, teams etc.

***Quick DB Reference:***

