NFL Player Profiles with Arrest records (ETL-Project)

The purpose of this project is to take data from different sources otherwise known as extract, then transform the data by dealing with missing values, choosing data to include in the final table, renaming columns, and dropping duplicate data, and finally loading the transformed data into a relational database. The project also identified NFL player arrest records and their demographic player records pulled from player profile records. As a bonus, we are also asking: "Which college did the most players with arrest records during the period 2000 -2017 come from?"

**Extract:**

We analyzed all the available arrest records from Kaggle NFL Arrests 2000 - 2017. A record of reported NFL arrests with details about crime, team and player.

* Web Scraping: 'html' from https://www.kaggle.com/patrickmurphy/nfl-arrests
* pull: 'json' from https://www.kaggle.com/zynicide/nfl-football-player-stats#profiles\_1512362725.022629.json
* Create data frame from NFL Arrests csv file
* Choose columns wanted from original data frame
* Rename columns
* Create data frame from .json file NFL profiles

The data frames created from the .json file and .csv file are shown below in figures 1A and 1B.

Fig 1A:

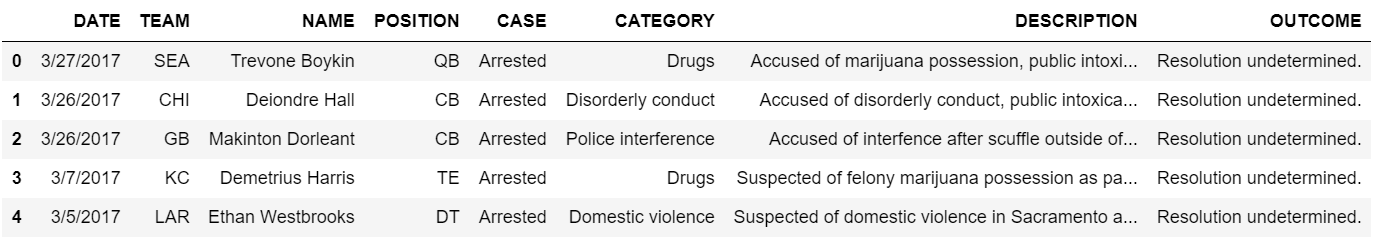
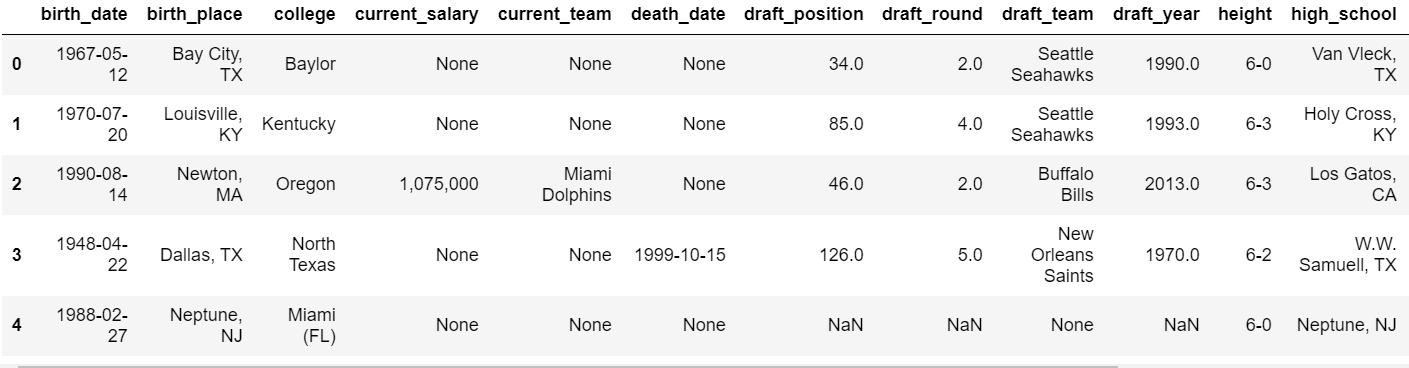


Fig 1B:



**Transform:**

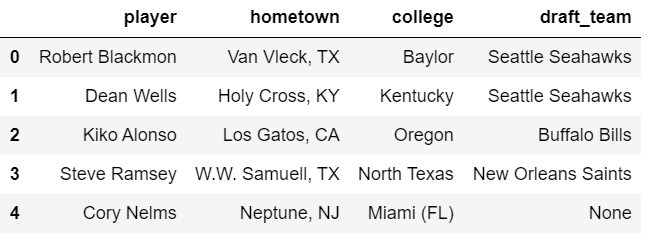
As part of the transformation of the data, the columns of each new created data frame were evaluated and columns that were relevant to the project was selected from the data frames. Next, some columns were renamed to give them more descriptive labeling. After manipulating columns, each data frame was evaluated to find what type of data was included in the database, the shape of the data meaning the number of rows and columns in the database, missing data, and duplicate data.

### After evaluating all the aspects mentioned above, the NFL\_profiles\_df data frame was found to have missing data in the "college","draft\_team", and "hometown" columns. The missing data in those columns would be filled with "not available", "not drafted", and "not available" respectively due to the desire to not eliminate any rows of data from the data frame. Each data frame was found to have duplicate data in the player column. The duplicates were dropped except for the first instance of the player name. The reason for this choice is because we were addressing only one instance of an NFL player's arrests and not considering multiple arrests for the same individual. The transformed tables can be found below in figures 2A and 2B.

Figure 2A:



Figure 2B:



Load:

After the data was cleaned, the next step in the ETL process is loading. For this project a relational database was chosen; specifically, PostgreSQL. The reason for choosing a relational database is due to each table being related to players in the NFL. One table is named "NFL\_arrests" and the other "nfl\_profiles". Each table can be queried separately; however as will be explaind later, the tables can be joined where the player's names match in both tables. Joining the tables in this way gives the most robust data than either of the tables do separately. After the tables were loaded into PostgreSQL a query was performed using SQL Alchemy to combine the data of each table in a format that would give much more interesting data regarding the NFL players in the data frame than either table could independently. The final data frame created by querying the database can be found below in figure 3.

Figure 3:

