**ETL Project Report**

**Project Overview**

* The main purpose of the project was to examine different statistics in MLS (Major League Soccer) teams and players, while trying to draw comparisons and observations between club salaries and how much the players earn in the year 2019.

**E(Extract)**

* We decided to use web scraping as our main source of data retrieval. The links to for our data are as follows:
  + <https://mlsplayers.org/resources/salary-guide>
  + <https://fbref.com/en/comps/22/2798/2019-Major-League-Soccer-Stats>
* The data from these websites were uploaded through their URL’s while also being converted into CSV files that were uploaded into our new repository in Python/Jupyter Lab.
* We also queried the tables created from Python in postgres to create views to get the useful information that was required to answer our questions.
* **A screenshot of a cell phone

  Description automatically generated**

**T(Transform)**

* To clean our data, Jupyter Lab was used.
* The data transformation dependencies that were used were:
  + Pandas
  + Matplotlib
  + Numpy
  + Sys
  + os
  + Psycopg2
  + Sqlalchemy
  + BeautfiulSoup
  + Requests
* There were some columns within the data sets that needed to be deleted as they were not important.
  + Ex. Notes
* New columns within the data sets needed to be created so that the Clubs column would match up with the Squad column.
* Due to the MLS being separated by conferences (Eastern and Western), the data was split up. In order for it to be transformed, it had to be combined.
* Club ID’s were created in order to more easily maneuver and identify teams within SQL.
  + We intended these ID’s to be primary keys so that we would be able to reference the other clubs.
* Base Salaries and Guaranteed Compensation needed to be converted so that there was an aggregate.
* There have been teams that have been added to the league since 2019 and showed up in the data, so those teams needed to be identified and dropped from the table.
* Last names needed to be checked to see if there were any unique players.
* For the purpose of creating a relational database, the data was normalized furth with a table for clubs, while also creating Clubs as a primary key.
  + We created a separate table to normalize the data for Club ID’s. In the future, we would want to do the same for Player ID’s.
* Teams, Individual Players, and Salary Classes were created for the tables in SQL.
  + We created views from these tables in SQL to make the data more useful for the end user.
* **A screenshot of a video game

  Description automatically generated**

**L(Load)**

* The data was that was transformed through Jupyter Lab was loaded into the PostgreSQL database.
  + This was done by creating a database connection.
  + Team names table was created within the database
  + A Session Object was created to connect to the database
* Once the transformed data was loaded within the SQL database, different queries were created, such as:
  + Top 10 highest paid players in the MLS
  + Highest Salary within each club, sorted in descending order
  + Salaries table ordered by total salary amount spent on players by club in descending order
  + Wins percentage by club (Which teams underachieved/overachieved?)
  + Players with the most goals
  + Average goals scored per game by club
* Visualizations were created to help better understand the queries.
* We chose to load our final data into PostgreSQL, a relational databsase, because our final collection of tables relate to one another because of Club ID’s and Player ID’s. After the data normalization phase, we had a total of 8 tables and created views of them for final production of the representation of the data.
* **A screenshot of a computer screen

  Description automatically generated**