Team members: Mahtab Karimipour

Molly Eskelson

ETL Project

Datasets:

* + [European Soccer Database](https://www.kaggle.com/hugomathien/soccer)-sqlite
  + [European Soccer Database Supplementary](https://www.kaggle.com/jiezi2004/soccer)-csv
  + [Football.data.org/API](https://www.football-data.org/documentation/quickstart/)-api
* What useful investigation could be done with the final database?
  + Predicting outcomes of future games
  + Create own sports fantasy league
  + In house league/player/team analysis tool. (Think Moneyball (book/movie)
* Whether final DB will be relational or nonrelational, and why?
  + Relational Database as both api and sqlite datasets fields are known
  + In both API ,csv, and sqlite datasets, player, teams and matches typically have same type of attributes.

What we did-

-

Summary of project and files in it.

Ww took the [European Soccer Database Supplementary](https://www.kaggle.com/jiezi2004/soccer) csv file and adding it to the [European Soccer Database](https://www.kaggle.com/hugomathien/soccer) sqlite3 (database). Because the database was so large (with every pass and play from every team/player even when the focus is just on one league) we Sqlite3 database to just ten rows.

The Flask Api call so the Sqlite database and the added csv table.

We are trying to have the Flask api call on requests to the Original Api ([Football.data.org/API](https://www.football-data.org/documentation/quickstart/)).

To get and api key go to <https://www.football-data.org/client/register> .