2020

PROJECT 2

CONTRIBUTORS

Ahmed Gurbuz
Blake Skinner
Melissa Morgan
Pedro Flores
Rebekah Vinson

Major League Baseball (MLB) Statistics

WHY MLB???

Because it's **AMERICA'S FAVORITE PASTIME**. Even a non-sports fan can have a fun time at a baseball game! It has so much to offer. It's such a special sport because of its history and its players.

PROJECT GOAL

Create an interactive platform for users to explore the selected MLB data.

PROJECT SUMMARY

The Source:

Sean Lahman's Baseball Database

A sqlite database containing complete batting and pitching statistics from 1871 to 2018, plus fielding statistics, standings, team stats, managerial records, post-season data, and more. The SQL version of the data contains hundreds of thousands of records dating back to the inception of the MLB.

The Data:

For this project, we narrowed our focus to the following:

Player Info (over 19,500 records)

Contains a list of players with an MLB debut date from 1871 to 2018 along with detailed information for each player such as birth country, debut date, height, weight, etc.

Salaries (approximately 26,500 records)

Individual player salary by year and team from 1985 to 2016.

Teams (approximately 2,900 records)

Detailed information about each MLB team's performance, including number of wins and runs scored per season, from 1871 to 2018.

The Execution:

Create a Flask app with interactive visualizations.

PROJECT FINAL PRODUCT

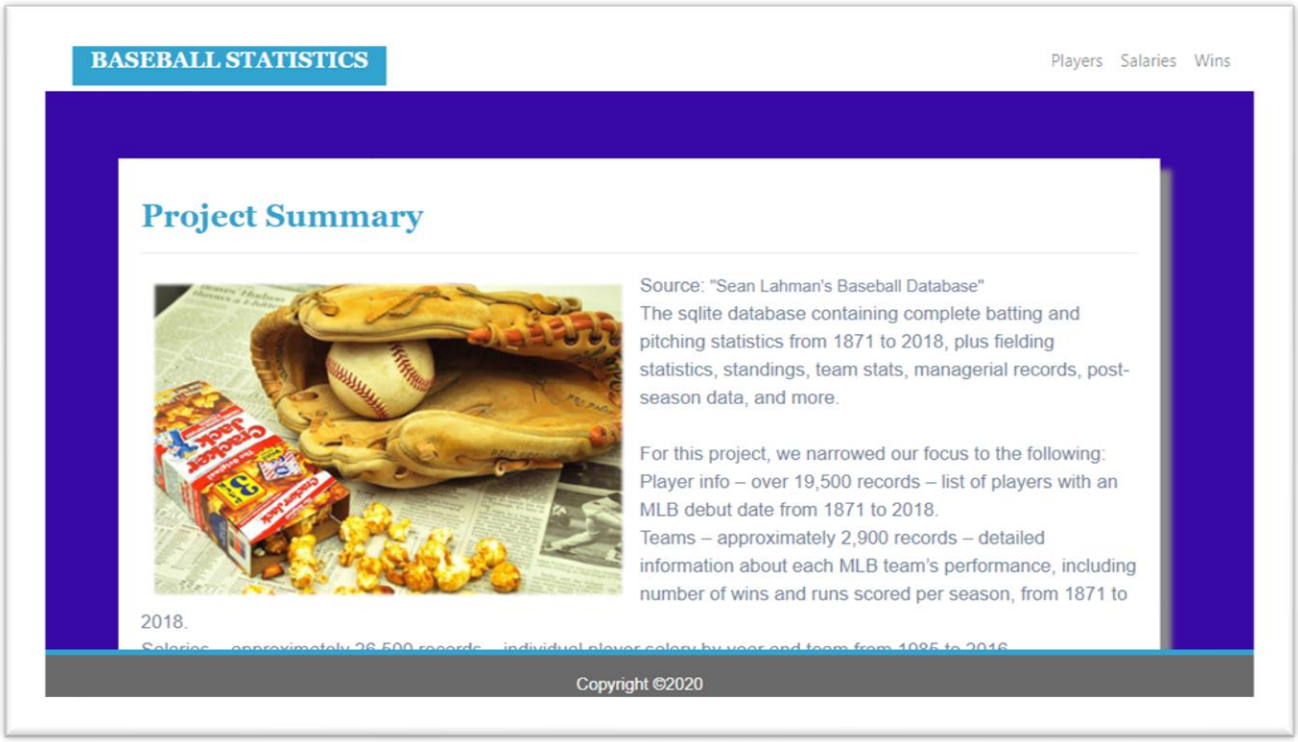
A full-stack web application focusing on visualizing Baseball Statistics. Our baseball statistics app was created using Python (Flask), SQL, JavaScript (Plotly & D3), HTML, CSS, and Bootstrap as programming languages. A link to the app is provided below.

https://baseballsmu.herokuapp.com/

PROJECT SCREENSHOTS – FLASK APP

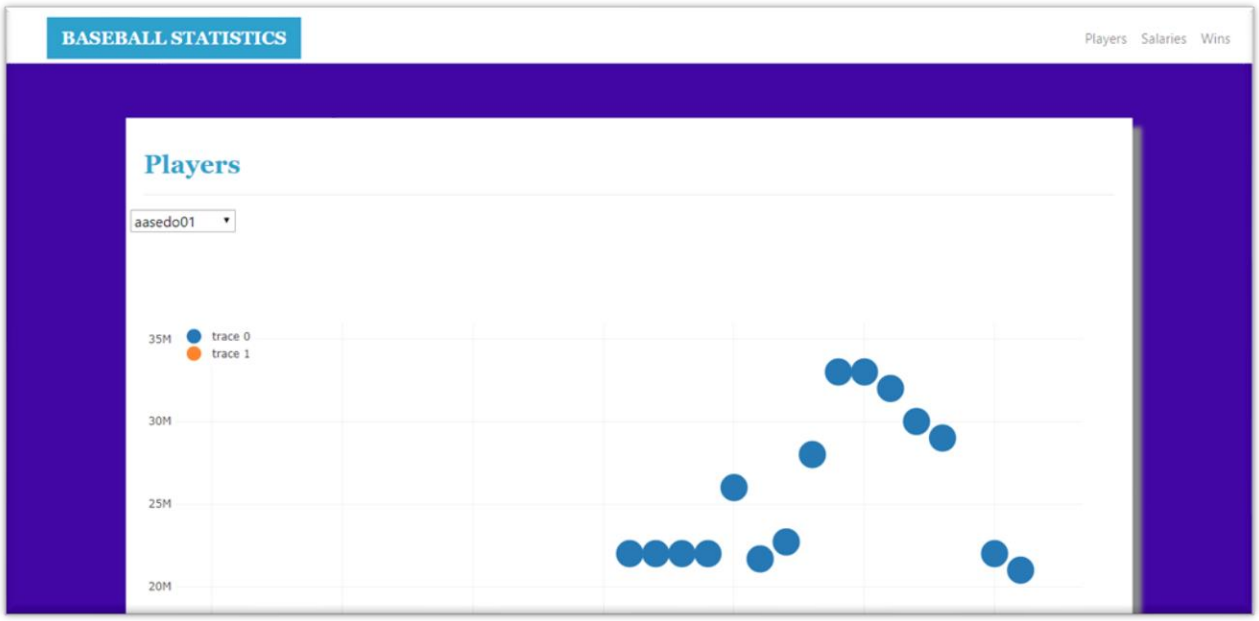
Dashboard/Homepage

Provides an overview of the project along with links to pages containing the interactive visualizations.



Players

Users can filter by individual players and add multiple players to the graph for comparison.

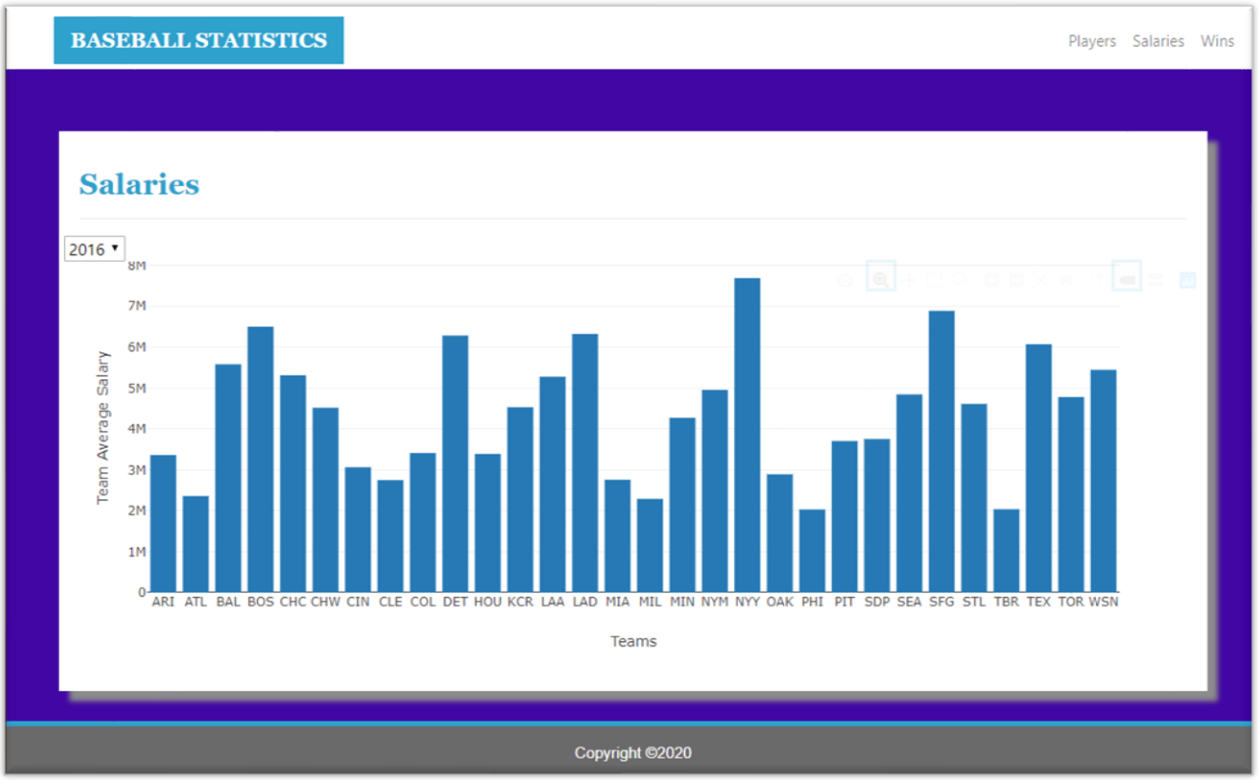


Example of data output when an individual player is selected using the dropdown filer:

Player Alexander Enmanuel (AKA Alex Rodriguez) was born in the USA. He debut his carrier in 1994-07-08. Stats: Weight: 230/ Height: 75/ R batter/ R

Salaries

Users can use the dropdown option to filter by year for a comparison of average team salary per team.



Wins

Page contains an interactive graph displaying team wins versus runs scored that can be filtered by year.

PROJECT NOTES

Biggest Hurdles

Filtering String/Text Data

The biggest issues involved trying to get string or text data to filter properly. Data classified as ints was easy to pull in and filter so the salary and win/runs visualizations did not create problems. Most of the time was spent trying to get the filter to work for player ID (text/varchar). Initially it would not return a json. Once the json issue was taken care of, there were issues passing it to the app.js.

Sticking to filtering by ints instead of txt/varchars would have allowed more time to perfect our current visualizations and add additional graphs.

Restructuring and Condensing the Files

A significant amount of time was spent restructuring the files and condensing them into smaller sizes by removing the extra copies of the database. Additional routes on the app.py file had to be created to get the other index.html files to load.

In turn, those files had to be renamed and moved to the templates folder. We were not successful at calling on the other index.html files when they were not in the same folder as our initial index.html file (our homepage). These files had to also be revised to call on the correct app.js file. It was a tedious and meticulous task.

Additional Considerations

In a perfect world, our app would include the changes below.

- Team names are not clear in our visualization. Currently, each team has a team ID predetermined by our data source. We would create a dashboard explaining names of the teams and their average salary per player explicitly
- Specific player details are currently displayed at the bottom of the page, below the interactive visualization. We would move these details to a clean and neatly displayed box or dashboard.
- Users could filter by player first and last name versus only being able to filter by player ID.