

STANDING ON THIN ICE

Scraping, manipulating, and organizing the 2015-2016 National Hockey League
Player Standings using R, MongoDB, and gWidgets



DECEMBER 17, 2015

DSCS 6020 COLLECTING, STORING, AND RETRIEVING DATA
FALL 2015

Table of Contents

| | |
|--|----------|
| Proposal | 2 |
| Original Proposal – As taken directly from Blackboard | 2 |
| Updated Proposal – Redefined based on feedback from classmates | 3 |
| Some Problems and Solutions | 3 |
| Using the correct webpages | 3 |
| Relearning MongoDB | 4 |
| Having to reinstall R, RStudio, and several packages | 4 |
| Working with gWidgets | 4 |
| R code | 4 |
| Output | 4 |
| The first GUI | 4 |
| The Second GUI | 7 |
| Conclusion | 8 |
| Websites Used for Scraping | 9 |
| Acknowledgement | 9 |
| Figure 1: The Serach GUI with default parameters | 5 |
| Figure 2: Drop down list of all players by full name | 5 |
| Figure 3: Drop down list of all possible first names in the current season | 6 |
| Figure 4: Drop down list of all possible last names in the current season | 6 |
| Figure 5: Drop down list of all possible positions | 7 |
| Figure 6: Drop down list of all NHL teams | 7 |
| Figure 7: Second GUI with all results | 8 |

Proposal

Original Proposal – As taken directly from Blackboard

So my past job had nothing to do with databases and I'm not entirely sure what I will be doing in the future so I decided that I would do a project based on an interest of mine: Hockey.

For a past assignment, I scraped specific player pages and printed out their stats. I want to expand on that assignment and combine it with some of our newer topics. My idea is that I will scrape all of the player information from the NHL website (meaning every player in the NHL) *. Then I will store the information in a NoSQL database**. Lastly I will create some user functions so that a user can search specific things in the database like team roster, players by position, specific players, and stats.

I will also have to manipulate some of the data before I move it into the database. For example, players are listed as John Doe (G) where G signifies goalie. I'll have to simplify the name and create a column for position. Also depending on their position, the information on the players'

pages is different, so parsing through information will have to be done in a certain order and different rules will be used based on the outcomes.

I'm not entirely sure how I will go about updating the information, if I plan to do so... or if it's a one-time grab. If I choose to use last season's roster, then I won't have to deal with trying to update information but it would be nice to be able to update this season's information after games.

*I have done some examination of the NHL's source code and I've seen that on the search player's page, each player is assigned a unique identifier which is what is used for each player's web page.

** Although the amount of players is pretty small, I think the amount of columns would be really annoying to translate into tables for SQL.

Updated Proposal – Redefined based on feedback from classmates

In assignment six, I scraped specific player pages and printed out their stats. For my term project, I want to expand on that assignment and combine it with some of our newer topics. My idea is that I will scrape all of the player information from the NHL website (meaning every player in the NHL). Then I will store the information using MongoDB. Lastly I will create a GUI so that a user can search the database for specific player information such as all the players on the Boston Bruins, every goalie in this current season, or search for players by their first or last name.

My steps for this assignment will include

- Scraping the current season roster
- Parsing and reformatting the data to use for scraping the individual player pages
- Scraping the individual pages
- Reformatting all of the data and entering it into MongoDB
- Extracting information from MongoDB to use for the GUI
- Building a GUI to search for requested player information and displaying the results

I decided for this project I will not continuously scrape information and update the database; my code will only be doing a one-time grab every time it is run. The reason for this is that I want to focus on learning how to use a GUI in R. I've never had the opportunity to build one for a class so I thought why not now.

Some Problems and Solutions

Using the correct webpages

One of the first problems arose when I didn't realize that the information I scraped for assignment six came from the Boston Bruins website rather than the NHL website. The individual player pages are almost identical on both sites but their source code is very different.

I am a little embarrassed to say that it took much longer than it should have to realize the mistake. Once caught, I had to spend some time looking through the NHL source code to be able to parse the information properly.

Relearning MongoDB

Enough time has passed since I turned in Assignment 12 on MongoDB that I needed to refresh my memory. Most of my confusion came from the fact that I had to wipe my computer after I turned Assignment 12 and forgot to download MongoDB again. All these problems had a simple solution: go over the MongoDB slides which walk you through installation.

Having to reinstall R, RStudio, and several packages

As I mentioned earlier, I had to wipe my computer because it was having some software issues. As this happened close to final exams and projects, I tried to set it up as quickly as I could by reinstalling all of the programs I needed. Unfortunately, I did not reinstall R correctly and was missing the tcl package. After perusing the internet for an hour and several unhelpful attempts, I found a forum which suggested I uninstall and reinstall everything. This was the best solution but obviously took some time.

Working with gWidgets

As I mentioned earlier, I have no experience making GUIs in any of the languages that I know. This has always been on my bucket list of things I should know how to do but haven't been important enough to spend some time on. That being said, since we didn't cover this in class I had to spend about a day or 2 searching the internet for tutorials on gWidgets. This was definitely one of the most aggravating parts of my project because of my unfamiliarity, I ended up having to design the GUIs 6 or 7 different ways before I got to a display that was visually pleasing and user friendly.

R code

Also included in my turned in submission is the pretty-R.pdf and the code if you are interested in running it.

Output

When running the code, two GUI's will pop up, one after the other. The first is the prompt for the user to search for the relevant data. The second is the output from the given search criteria.

The first GUI

If you would like to search for a specific player, you can search either by full name, first name, or last name (if you don't know the player's full name). If you'd like to compare players, you can compare players by their position or by their team. You can also compare players on the same team who play the same position. If you would just like to see all players in the current season, you can click either search button with the default parameters and a list of all players will be generated.

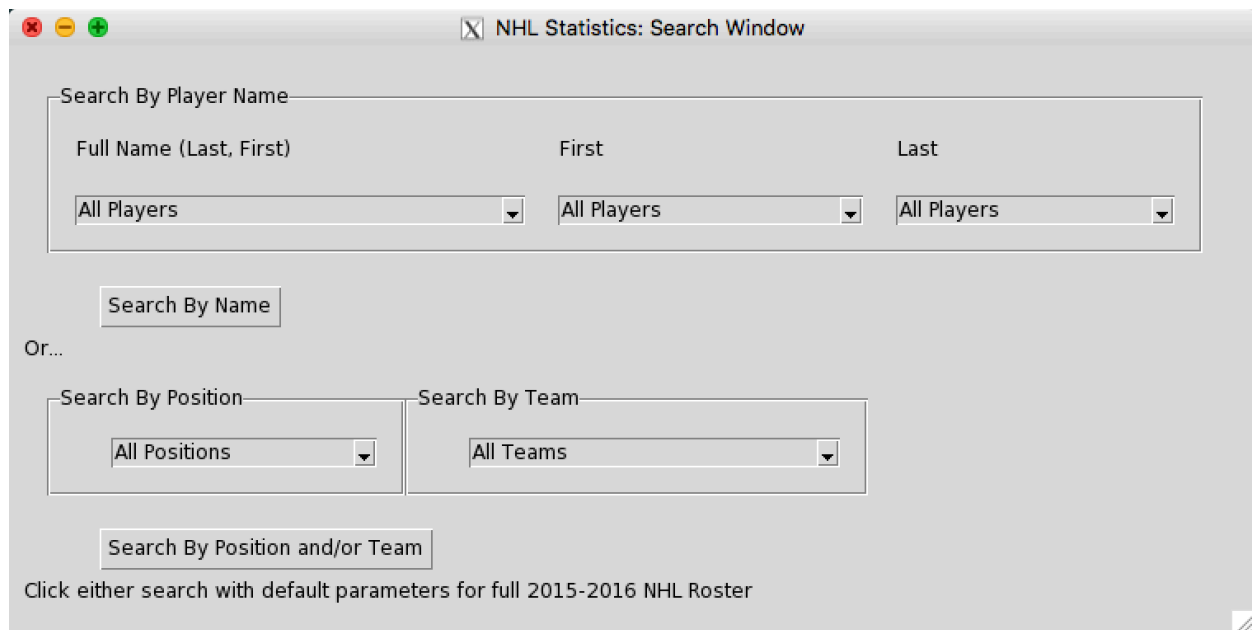


Figure 1: The Search GUI with default parameters

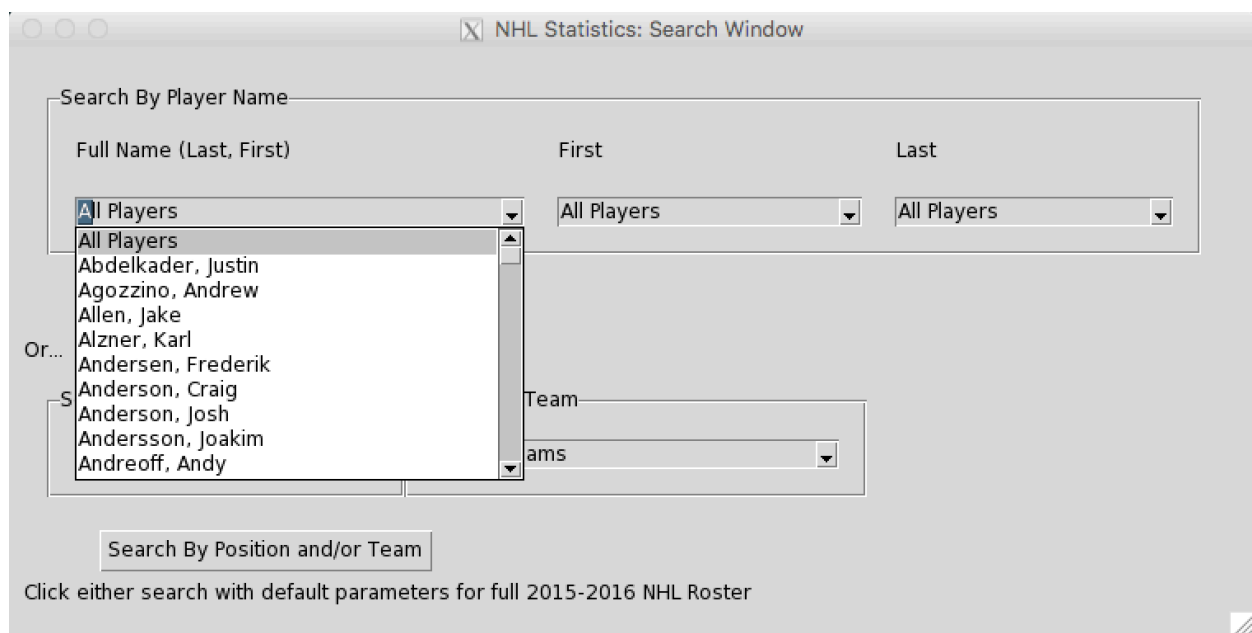


Figure 2: Drop down list of all players by full name

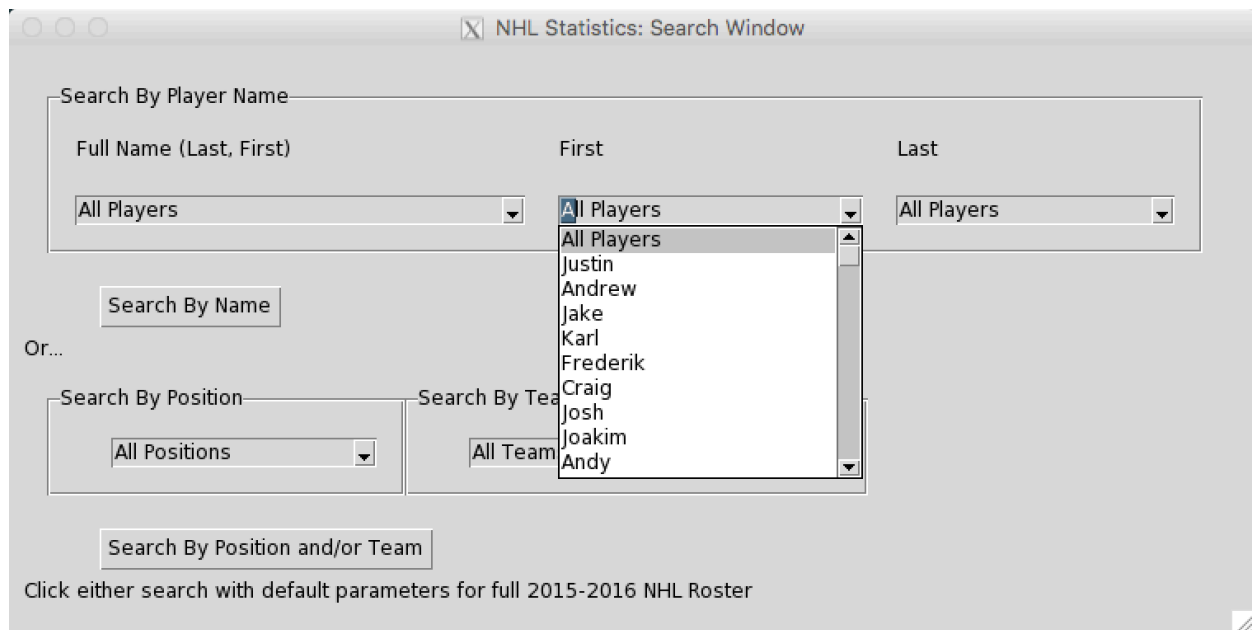


Figure 3: Drop down list of all possible first names in the current season

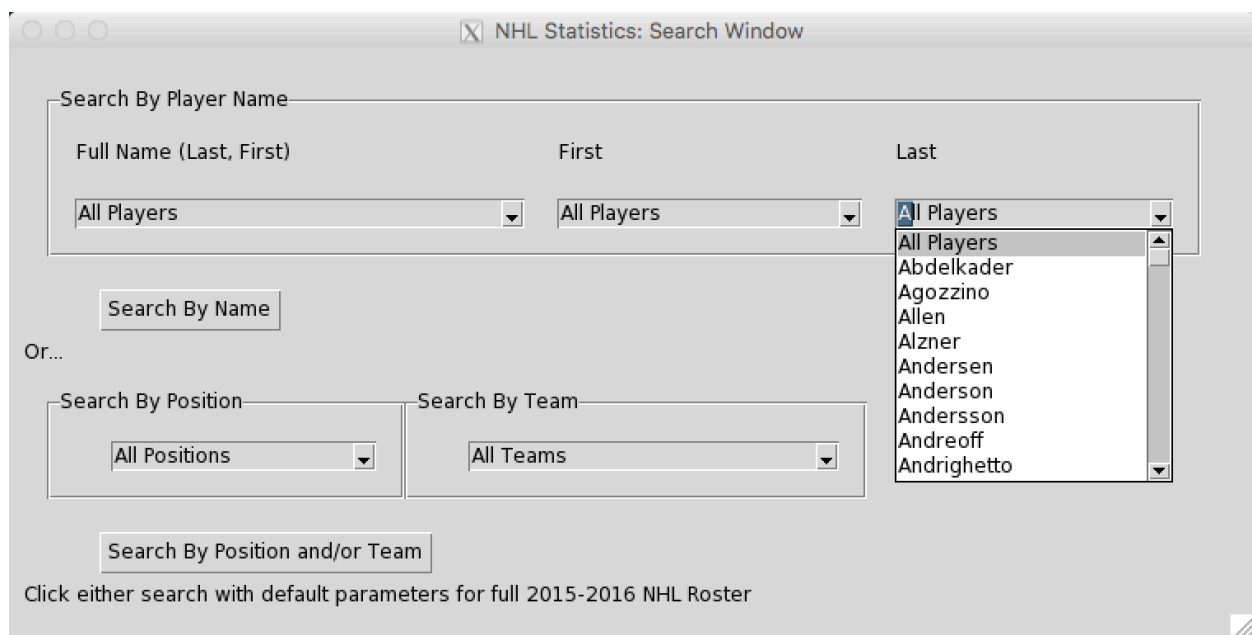


Figure 4: Drop down list of all possible last names in the current season

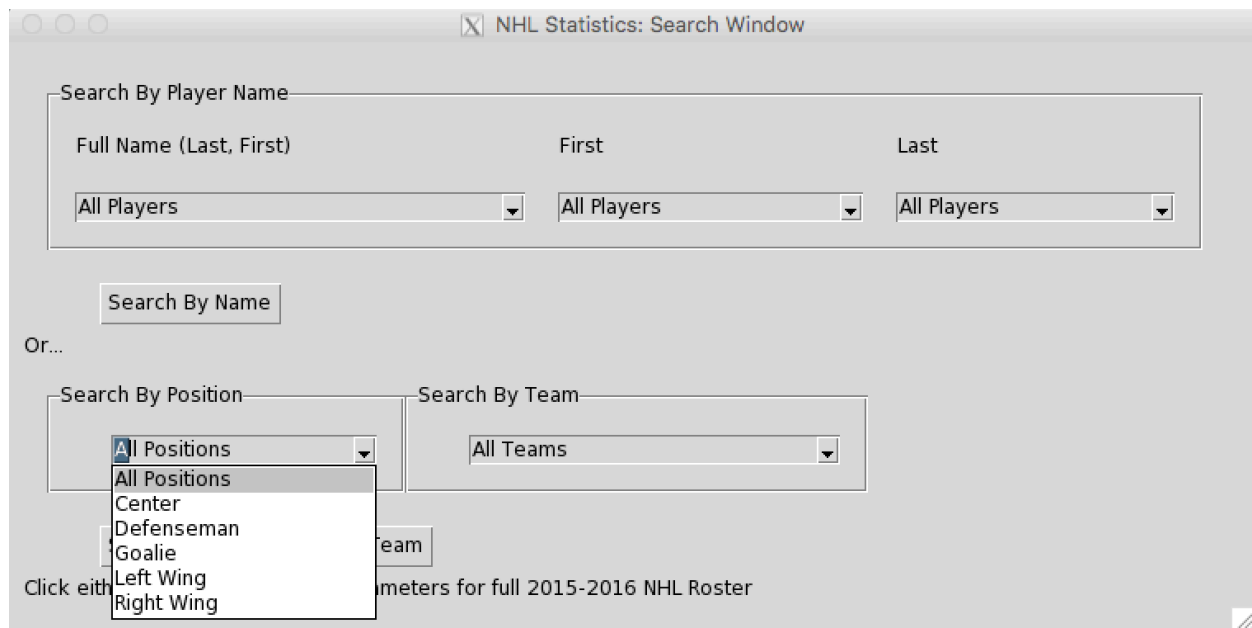


Figure 5: Drop down list of all possible positions

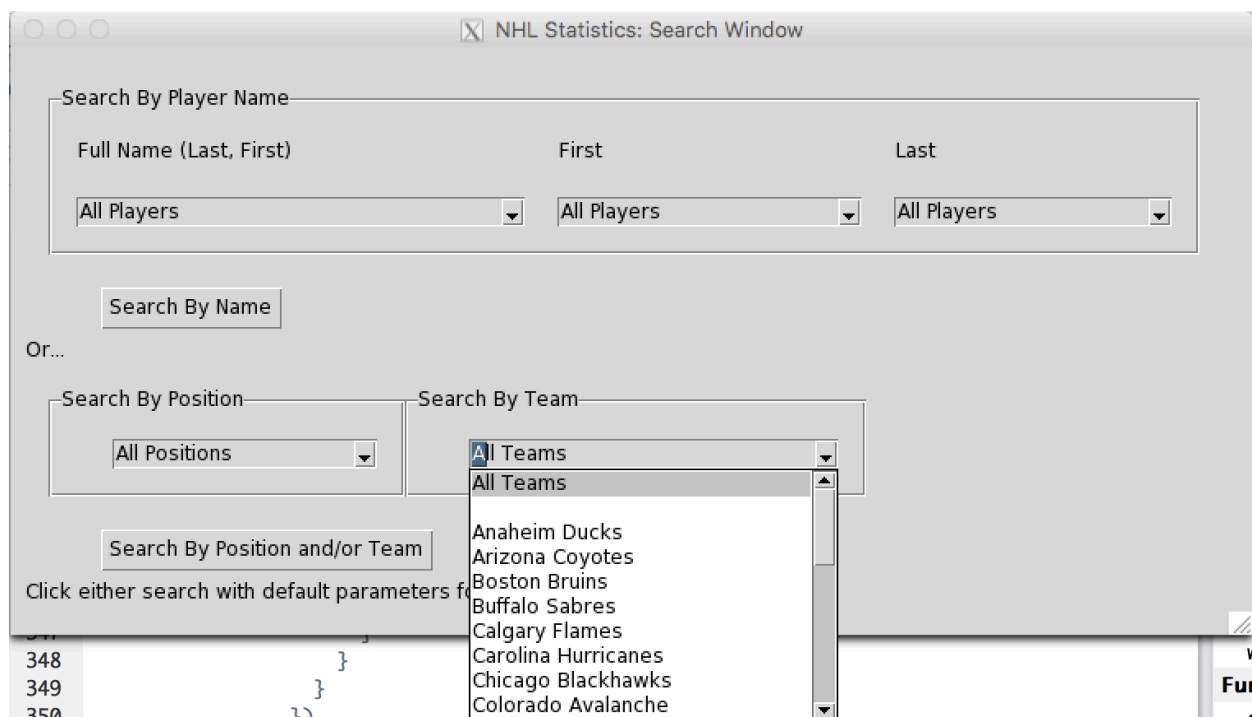
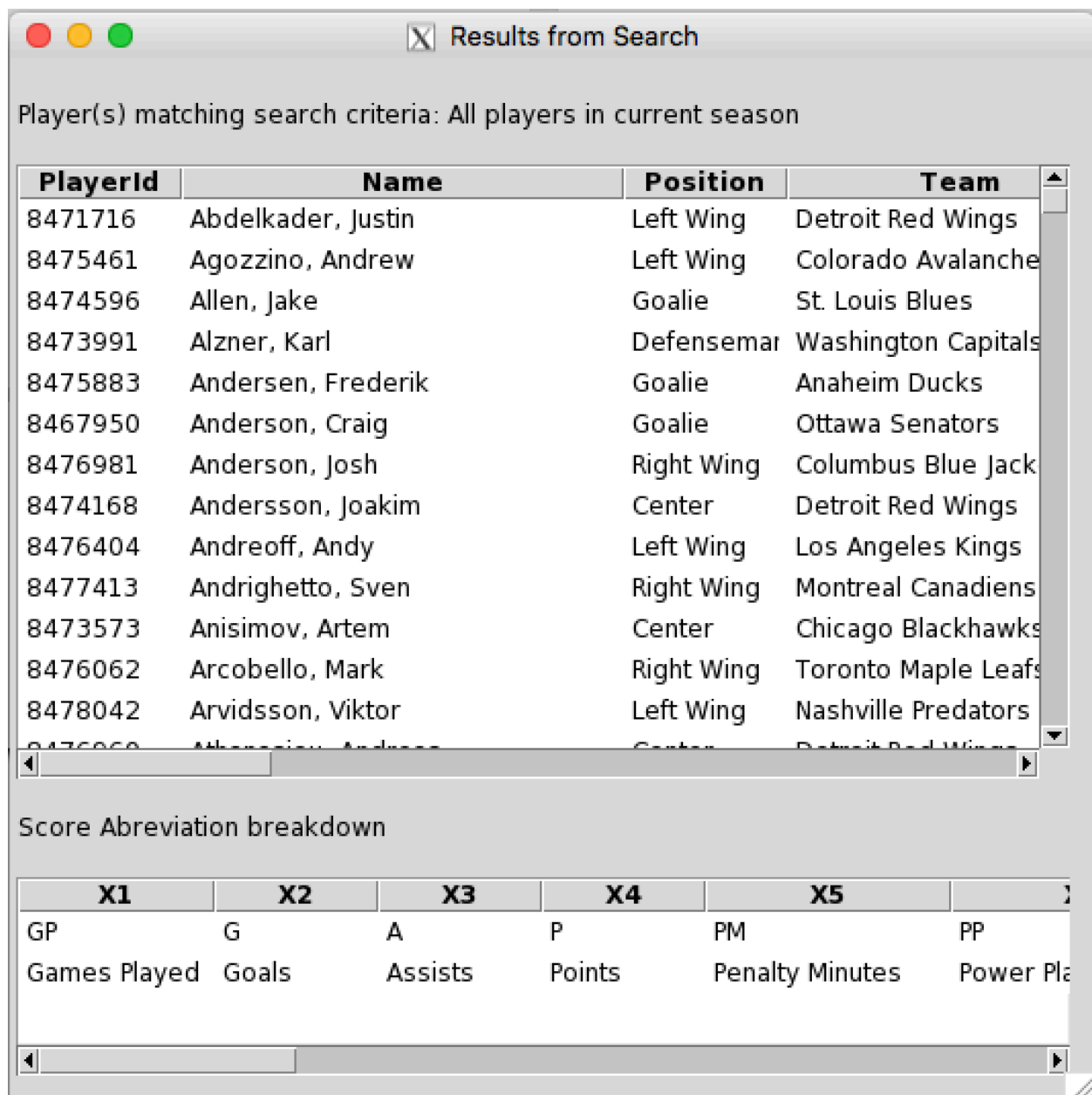


Figure 6: Drop down list of all NHL teams

Note that there is a purposeful blank in Figure 6 because there are some people in the current season who are currently not on any team.

The Second GUI

When clicking the search button(s), the first GUI will close and the second GUI will open with the results displayed. The GUI will consist of two tables: the first is the results, the second breaks down the abbreviations. At the top of the GUI, your search criteria will be displayed.



Player(s) matching search criteria: All players in current season

| PlayerId | Name | Position | Team |
|----------|---------------------|------------|---------------------|
| 8471716 | Abdelkader, Justin | Left Wing | Detroit Red Wings |
| 8475461 | Agozzino, Andrew | Left Wing | Colorado Avalanche |
| 8474596 | Allen, Jake | Goalie | St. Louis Blues |
| 8473991 | Alzner, Karl | Defenseman | Washington Capitals |
| 8475883 | Andersen, Frederik | Goalie | Anaheim Ducks |
| 8467950 | Anderson, Craig | Goalie | Ottawa Senators |
| 8476981 | Anderson, Josh | Right Wing | Columbus Blue Jack |
| 8474168 | Andersson, Joakim | Center | Detroit Red Wings |
| 8476404 | Andreoff, Andy | Left Wing | Los Angeles Kings |
| 8477413 | Andrighetto, Sven | Right Wing | Montreal Canadiens |
| 8473573 | Anisimov, Artem | Center | Chicago Blackhawks |
| 8476062 | Arcobello, Mark | Right Wing | Toronto Maple Leafs |
| 8478042 | Arvidsson, Viktor | Left Wing | Nashville Predators |
| 8476060 | Athanasios, Andreas | Center | Detroit Red Wings |

Score Abreviation breakdown

| X1 | X2 | X3 | X4 | X5 | X6 |
|--------------|-------|---------|--------|-----------------|------------|
| GP | G | A | P | PM | PP |
| Games Played | Goals | Assists | Points | Penalty Minutes | Power Play |

Figure 7: Second GUI with all results

Conclusion

I learned a lot about scraping, MongoDB, and GUIs in this assignment. I also learned that it is important to take a step back for a moment when necessary

Websites Used for Scraping

Website used to extract all players in the current season. There were 17 pages of information, so the webpages looked like this but with the page count adjusted accordingly.

<http://www.nhl.com/ice/playersearch.htm?position=S&season=20152016&pg=1>

Individual player pages. All urls had the same characteristics other than the change in player id.

<http://www.nhl.com/ice/player.htm?id=8471716>

Acknowledgement

I would like to thank my roommate Sarah Janiszewski for constantly talking about her fantasy league. Her complaints about the organization of the NHL's website and phone app made a good source of inspiration for this project.

I would also like to acknowledge Yatish Jain. I don't usually like to ask for help unless I have searched the internet for hours and can't find an answer. Yatish has been very helpful in answering my very confusing questions.