# 100 Days of Code Tracking Log

*Eric Fastner*

*January 1, 2019*

***Template***

**Day 0: January 1, 2019**

**Time:**

**Today's Progress:**

**Thoughts:**

**Link to work:**

---

## 100 Days Of Code - Log

**Day 1: January 1, 2019**

**Time:** 120 Minutes (8:30 PM - 10:30 PM)

**Today's Progress:** Forked and Modified the 100 Days of Code resources to fit my needs. Split my previous Lock Line Analysis into it's own repository via this page

**Thoughts:** Lots of misc items completed that I've been putting off for awhile, I almost pushed the start of this until tomorrow. Ultimately, it was still a really positive 2 hours!

**Link to work:** New Lock Line Analysis Repo, 100 Days of Code Repo

**Day 2: January 2, 2019**

**Time:** 120 Minutes (10:00 PM - 12:00 AM)

**Today's Progress:** Converted lock line vizualization script into a function, branched the file via these instructions, and re-merged with the master branch

**Thoughts:** I need to get more comfortable with branching, but we'll get there. The foundations for this function may be fun to ultimately load into some sort of Shiny app. The code is becoming a lot more flexible

**Link to work:** New Lock Line Viz Function

**Day 3: January 3, 2019**

**Time:** 90 Minutes (8:15 PM - 9:45 PM)

**Today's Progress:** Experimented with the RPostgreSQL package in R to learn how to interact with PostgreSQL. Utilized an r-bloggers tutorial to get started.

**Thoughts:** This was a big confidence boost. I've been a little intimidated about getting this set up, but it was WAY easier than I thought it would be. Glad I have this challenge to make me sit down and figure some of this out

**Link to work:** New RPostgreSQL Functions

**Day 4: January 4, 2019**

**Time:** 60 Minutes (7:30 PM - 8:30 PM)

**Today's Progress:** Kind of a light day. Didn't get a lot done. Played around with the RPostgreSQL commands to begin creating and populating tables via RStudio

**Thoughts:** Mostly exploratory, should come in handy for Day 5

**Link to work:** New RPostgreSQL Functions

**Day 5: January 5, 2019**

**Time:** >150 Minutes (Off and on from 10 AM - 4:00 PM)

**Today's Progress:** Created and populated tables with game results by season from 2007 - 2018, a table with the same data organized by team, and a table with each individual skater's stats going back to 2007. Updated some old summary functions as well

**Thoughts:** Lots of work to do! I need to stop reusing the table creation and populating functions for specific tables. Either I need to make them more flexible, or I need to start creating separate functions for each new table. The game summary and skater summary functions could also use some extra documentation/streamlining. Maybe this would be a good time to start inplementing some unit testing?

**Link to work:** New RPostgreSQL Functions, Skater Summary

**Day 6: January 6, 2019**

**Time:** >90 Minutes (8:30pm to 10pm)

**Today's Progress:** Played around with twitteR package on CRAN. Used instructions from here to get started

**Thoughts:** Frustrating day. I started out by just experimenting with the twitteR package and ultimately pulled all off Lin Manuel Miranda's "Gmorning" and "Gnight" tweets, however the function seems to only pull the truncated version of tweets when they are over 140 characters. I spent a lot of time trying to find a solution, but up until this point I've come up short. Other options would be to switch over to the rtweet package, or to just pull the API myself. I sort of like the latter, as I don't have much experience with APIs.

**Link to work:** Twitter API Functions

**Day 7: January 7, 2019**

**Time:** >90 Minutes (7:30pm to 9:00pm)

**Today's Progress:** Experimented more with RPostgreSQL, created some new tables, and set up some functions to make basic queries. These should come in handy for pulling data across seasons. One of the issues that I used to have was reading multiple seasons into memory from multiple .csv files and then filtering out what I needed. Now I should be able to just query the filtered data straight from the SQL tables as needed.

**Thoughts:** I'm really excited about where this is going. I added in the tables with draft years, which means I can hopefully use this code to query data by game for each player

**Link to work:** Basic RPostgreSQL Queries