As its been pointed out to me on that it would be handy if within [fitzRoy](https://jimmyday12.github.io/fitzRoy/) that it should contain past players data from footywire.

So here is roughly how to do that.

**Step 1 – get all the packages you need**

library(rvest)

## Loading required package: xml2

library(tidyverse)

## ── Attaching packages ──────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.1.0 ✔ purrr 0.3.0

## ✔ tibble 2.0.1 ✔ dplyr 0.8.0.1

## ✔ tidyr 0.8.3 ✔ stringr 1.4.0

## ✔ readr 1.3.1 ✔ forcats 0.4.0

## ── Conflicts ─────────────────── tidyverse\_conflicts() ──

## ✖ dplyr::filter() masks stats::filter()

## ✖ readr::guess\_encoding() masks rvest::guess\_encoding()

## ✖ dplyr::lag() masks stats::lag()

## ✖ purrr::pluck() masks rvest::pluck()

library(naniar)

[naniar](https://cran.r-project.org/web/packages/naniar/vignettes/getting-started-w-naniar.html) isn’t something I have used on the blog before but I find its pretty handy. What we are doing here is we are web-scraping and we have an issue in our scrape. Basically the row in which we pluck our height, weight and position from sometimes contains height and weight but not position.

In that case when we make our data tidy, when we go to find a ‘’position’’ for a player who doesn’t have one, we get the whole row. Hopefully this will be a bit more clear with the screenshots and alike below.

**Step 2 – Pick a team to scrape (this case I’m going to do** [**West Coast**](https://www.footywire.com/afl/footy/ti-west-coast-eagles)**)**

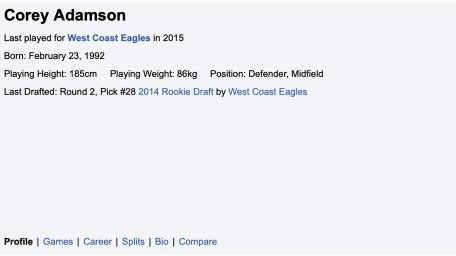
Keeping in mind we want to scrape the webpages lets just look at a few eagles players and see how their pages are structured. To do this I don’t think we need to really dig into the html but lets just look at the pages.

First player I decided to click on was [John Annear](https://www.footywire.com/afl/footy/pp-west-coast-eagles--john-annear) we can see we can get his date of birth, his height and weight.



John Annear screenshot footywire

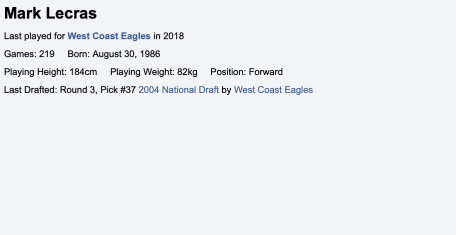
Second player I decided to look at was [Corey Adamson](https://www.footywire.com/afl/footy/pp-west-coast-eagles--corey-adamson) here we can see we can get his date of birth, his height, his weight, his position and his draft position.



Corey Adamson screenshot footywire

Thirdy player I decided to look at was [Mark Lecras](https://www.footywire.com/afl/footy/pp-west-coast-eagles--mark-lecras)

Here we can see we get his date of birth, games played, height, weight, position and draft position.



Mark Lecras screenshot footywire

Then after randomly clicking on a few other players I am reasonble confident that these 3 cover the variety of different bits of information provided for all past players.

So now lets get scraping

**Step 3 – Scrape a single player**

For this example lets do [Mark Lecras](https://www.footywire.com/afl/footy/pp-west-coast-eagles--mark-lecras), the reason is he has the most amount of information for past players on his page.

cbind.fill <- function(...){

nm <- list(...)

nm <- lapply(nm, as.matrix)

n <- max(sapply(nm, nrow))

do.call(cbind, lapply(nm, function (x)

rbind(x, matrix(, n-nrow(x), ncol(x)))))

}

page<-read\_html("https://www.footywire.com/afl/footy/pp-west-coast-eagles--mark-lecras")

player<- page%>%

html\_nodes("#playerProfileName")%>%

html\_text()

player

## [1] "Mark Lecras"

playing.for<- page%>%

html\_nodes("#playerProfileTeamDiv a b")%>%

html\_text() %>% as.tibble()

## Warning: `as.tibble()` is deprecated, use `as\_tibble()` (but mind the new semantics).

## This warning is displayed once per session.

playing.for

## # A tibble: 1 x 1

## value

##

## 1 West Coast Eagles

games<-page%>%

html\_nodes("#playerProfileData1")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Games: ).\*(?=Born:)"))%>%as.tibble()

games

## # A tibble: 1 x 1

## value

##

## 1 "219 "

born<- page%>%

html\_nodes("#playerProfileData1")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_remove(".\*Born: ")%>%

str\_squish() %>% as.tibble()

born

## # A tibble: 1 x 1

## value

##

## 1 August 30, 1986

weight<-page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Weight:).\*(?=kg)"))%>%as.tibble()

weight

## # A tibble: 1 x 1

## value

##

## 1 " 82"

height<-page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Height:).\*(?=cm)"))%>%as.tibble()

height

## # A tibble: 1 x 1

## value

##

## 1 " 184"

draft\_position <- page%>%

html\_nodes("#playerProfileDraftInfo")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Drafted: ).\*(?=by)"))%>%as.tibble()

draft\_position

## # A tibble: 1 x 1

## value

##

## 1 Round 3, Pick #37 2004 National Draft

club\_drafted <- page%>%

html\_nodes("#playerProfileDraftInfo a+ a")%>%

html\_text()%>%str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_remove(".\*by") %>% as.tibble()

club\_drafted

## # A tibble: 1 x 1

## value

##

## 1 West Coast Eagles

position <- page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_remove(".\*Position: ")%>%

str\_squish() %>% as.tibble()

position

## # A tibble: 1 x 1

## value

##

## 1 Forward

player\_information <- cbind.fill(player, playing.for, games,born, weight, height,draft\_position, club\_drafted, position)

player\_information <- as.tibble(player\_information)

player\_information

## # A tibble: 1 x 9

## V1 value V3 V4 V5 V6 V7 V8 V9

##

## 1 Mark L… West Coa… "219 " August… " 82" " 18… Round 3, Pic… West Co… Forw…

Selecter gadget is doing a great job, but what we can notice here is that the html\_nodes do not change for height, weight, position. So in our earlier case with [John Annear](https://www.footywire.com/afl/footy/pp-west-coast-eagles--john-annear) what this means is that in his position column it will contain the whole row as we were not able to remove words post ‘position’ like we did for [Mark Lecras](https://www.footywire.com/afl/footy/pp-west-coast-eagles--mark-lecras).

This is where [naniar](https://cran.r-project.org/web/packages/naniar/vignettes/getting-started-w-naniar.html) will come in handy, for all those rows that do not have a valid position (duplicated text values we have already in other columns) we can just use naniar and specifically its replace\_with\_na function to make those specific data entries na

Lets see how this would work below.

**Step 4 – Scrape a whole retired team list.**

url<-"https://www.footywire.com/afl/footy/ti-west-coast-eagles"

link<-read\_html(url)%>%

html\_nodes(".lnormtop a")%>%

html\_attr("href")

# david-brown #nwws or ewmocw

url\_players<-str\_c("https://www.footywire.com/afl/footy/",link)

# url\_players<-head(url\_players,19)

#need to get rid of david brown

# url\_players<-url\_players[-21]

player\_info <- function(x){

# page <- read\_html(x)

page<-read\_html(x)

player<- page%>%

html\_nodes("#playerProfileName")%>%

html\_text()

player

playing.for<- page%>%

html\_nodes("#playerProfileTeamDiv a b")%>%

html\_text() %>% as.tibble()

playing.for

games<-page%>%

html\_nodes("#playerProfileData1")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Games: ).\*(?=Born:)"))%>%as.tibble()

games

born<-page%>%

html\_nodes("#playerProfileData1")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_remove(".\*Born: ")%>%

str\_squish() %>% as.tibble()

born

weight<-page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Weight:).\*(?=kg)"))%>%as.tibble()

weight

height<-page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Height:).\*(?=cm)"))%>%as.tibble()

height

draft\_position <- page%>%

html\_nodes("#playerProfileDraftInfo")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_extract(pattern =("(?<=Drafted: ).\*(?=by)"))%>%as.tibble()

draft\_position

club\_drafted <- page%>%

html\_nodes("#playerProfileDraftInfo a+ a")%>%

html\_text()%>%str\_replace\_all("[\r\n]" , "")%>%

str\_squish()%>%

str\_remove(".\*by") %>% as.tibble()

club\_drafted

position <- page%>%

html\_nodes("#playerProfileData2")%>%

html\_text()%>%

str\_replace\_all("[\r\n]" , "")%>%

str\_remove(".\*Position: ")%>%

str\_squish() %>% as.tibble()

position

#combine, name, and make it a tibble

player\_information <- cbind.fill(player, playing.for, games,born, weight, height,draft\_position, club\_drafted, position)

player\_information <- as.tibble(player\_information)

# print(x)

# return(x)

return(player\_information)

}

footywire <- purrr::map\_df(url\_players, player\_info)

footywire

## # A tibble: 238 x 9

## V1 value V3 V4 V5 V6 V7 V8 V9

##

## 1 Corey … West C… Februa… " 86" " 18… Round 2, … West C… Defender,…

## 2 Damien… West C… "54 " March … " 75" " 18… Round 4, … Collin… Midfield

## 3 John A… West C… June 1… " 80" " 17… Playing H…

## 4 David … West C… "3 " June 1… " 93" " 19… Round 3, … West C… Playing H…

## 5 Steven… West C… "79 " Januar… " 83" " 18… Round 1, … West C… Forward

## 6 Ashley… West C… April … " 86" " 18… Round 2, … West C… Midfield

## 7 Jason … Sydney… "193 " Novemb… " 10… " 20… Pick #1 1… West C… Ruck

## 8 Drew B… West C… "265 " Februa… " 89" " 18… Pick #1 1… West C… Defender,…

## 9 Adrian… West C… Decemb… " 85" " 17… Playing H…

## 10 Glen B… West C… June 1… " 95" " 19… Playing H…

## # … with 228 more rows

names(footywire) <- c("player", "club", "games","born","weight","height", "draft\_position", "club\_drafted", "position")

df\_replace<-footywire%>%filter(!position %in% c("Midfield", "Defender", "Defender, Forward", "Defender, Midfield",

"Forward", "Forward, Ruck", "Midfield, Forward", "Ruck"))%>%

select(position)

footywire\_eagles<-footywire%>%naniar::replace\_with\_na(replace=list(position=df\_replace))

footywire\_eagles

## # A tibble: 238 x 9

## player club games born weight height draft\_position club\_drafted

##

## 1 Corey… West… Febr… " 86" " 185" Round 2, Pick… West Coast …

## 2 Damie… West… "54 " Marc… " 75" " 181" Round 4, Pick… Collingwood…

## 3 John … West… June… " 80" " 178"

## 4 David… West… "3 " June… " 93" " 194" Round 3, Pick… West Coast …

## 5 Steve… West… "79 " Janu… " 83" " 180" Round 1, Pick… West Coast …

## 6 Ashle… West… Apri… " 86" " 188" Round 2, Pick… West Coast …

## 7 Jason… Sydn… "193… Nove… " 104" " 201" Pick #1 1991 … West Coast …

## 8 Drew … West… "265… Febr… " 89" " 184" Pick #1 1992 … West Coast …

## 9 Adria… West… Dece… " 85" " 177"

## 10 Glen … West… June… " 95" " 195"

## # … with 228 more rows, and 1 more variable: position

Bingo there you have it, how to scrape a whole teams worth of data.