**#Scrape Webdata from Great Schools**

**#Scrape name data**

**#Define the page the information is coming from**

url <- 'https://www.greatschools.org/virginia/falls-church/fairfax-county-public-schools/schools/'

webpage <- read\_html(url)

**#Scrape name of school**

name.data.html <- html\_nodes(webpage, '.rs-schoolName')

name.data <- html\_text(name.data.html)

name.data <- as.factor(name.data)

**#remove duplicated data**

Name.data1 <- name.data1[!duplicated(name.data1)]

# I did this nine times because there were nine pages of schools on #greatschools.com for Fairfax county.

**#turn all nine rank.datas into data frames.**

Rank.datax <- as.dataframe(rank.datax)

**#combine all the data frames into a single data frame**

name.data.df <-rbind(name.data1.df,name.data2.df,name.data3.df,name.data4.df,name.data5.df,name.data6.df,name.data7.df,name.data8.df,name.data9.df)

#this was tedious and I need to do this all over again for the great #schools rank. This would be impossible on a larger scale project. I #know there is a way to code this so I am now looking at loops.

**#Scrape rank of school**

#This formula extracts the information from all nine pages

Library(purrr)

url\_base <- "https://www.greatschools.org/virginia/falls-church/fairfax-county-public-schools/schools/?page=%d"

map\_df(1:9, function(i) {

# simple but effective progress indicator

cat(".")

pg <- read\_html(sprintf(url\_base, i))

data.frame(rank=html\_text(html\_nodes(pg, "div.dib")),

stringsAsFactors=FALSE)

}) -> rank

dplyr::glimpse(rank)

#Pull address data from greatschools

url\_base <- "https://www.greatschools.org/virginia/falls-church/fairfax-county-public-schools/schools/?page=%d"

map\_df(1:9, function(i) {

# simple but effective progress indicator

cat(".")

pg <- read\_html(sprintf(url\_base, i))

data.frame(address=html\_text(html\_nodes(pg, "div.ptm.notranslate")),

stringsAsFactors=FALSE)

}) -> address

dplyr::glimpse(address)

#Separate name from address

address <- extract(address, "address", c("name", "address"), "([\\D+)(\\d.\*)")](file:///\\D+)(\\d.*)%22))