Assignment 09: Data Scraping

Meilin Chan

Total points:

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on data scraping.

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Fay_09_Data_Scraping.Rmd") prior to submission.

Set up

1. Set up your session:

- Check your working directory
- Load the packages `tidyverse`, `rvest`, and any others you end up using.
- Set your ggplot theme

```
#1
getwd()
```

```
## [1] "C:/Users/meili/OneDrive - Duke University/EDA/Environmental_Data_Analytics_2022"
```

```
library(tidyverse)
library(rvest)
```

```
## Warning: package 'rvest' was built under R version 4.1.3
```

```
library(dataRetrieval)
```

```
## Warning: package 'dataRetrieval' was built under R version 4.1.3
```

```
library(lubridate)
library(viridis)
library(tinytex)

my_theme <- theme_classic() +
  theme(legend.position = "bottom",
        legend.text = element_text(color = "black"))
theme_set(my_theme)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2019 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- Change the date from 2020 to 2019 in the upper right corner.
- Scroll down and select the LWSP link next to Durham Municipality.
- Note the web address: https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020

Indicate this website as the as the URL to be scraped. (In other words, read the contents into an `rvest` webpage object.)

```
#2
website.20 <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')
```

3. The data we want to collect are listed below:

- From the "1. System Information" section:

- Water system name

- PSWID

- Ownership

- From the "3. Water Supply Sources" section:

- MAX Daily Use (MGD) - for each month

In the code chunk below scrape these values, assigning them to three separate variables.

> HINT: The first value should be "Durham", the second "03-32-010", the third "Municipality", and the last should be a vector of 12 numeric values, with the first value being 36.0100.

```
#3
water.system.name <- website.20 %>%
  html_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>%
  html_text()
water.system.name #Durham
```

```
## [1] "Durham"
```

```
pswid <- website.20 %>%
  html_nodes('td tr:nth-child(1) td:nth-child(5)') %>%
  html_text()
pswid #03-32-010
```

```
## [1] "03-32-010"
```

```
ownership <- website.20 %>%
  html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>%
  html_text()
ownership #Municipality
```

```
## [1] "Municipality"
```

```
max.withdrawals.mgd <- website.20 %>%
  html_nodes('th~ td+ td') %>%
  html_text()
max.withdrawals.mgd #dim 1:12; 1st value = 36.0100
```

```
##  [1] "36.0100" "36.9800" "41.6900" "32.0500" "40.6100" "40.5600" "37.2900"
##  [8] "43.6300" "33.3200" "32.3700" "41.9300" "28.0600"
```

4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date

column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

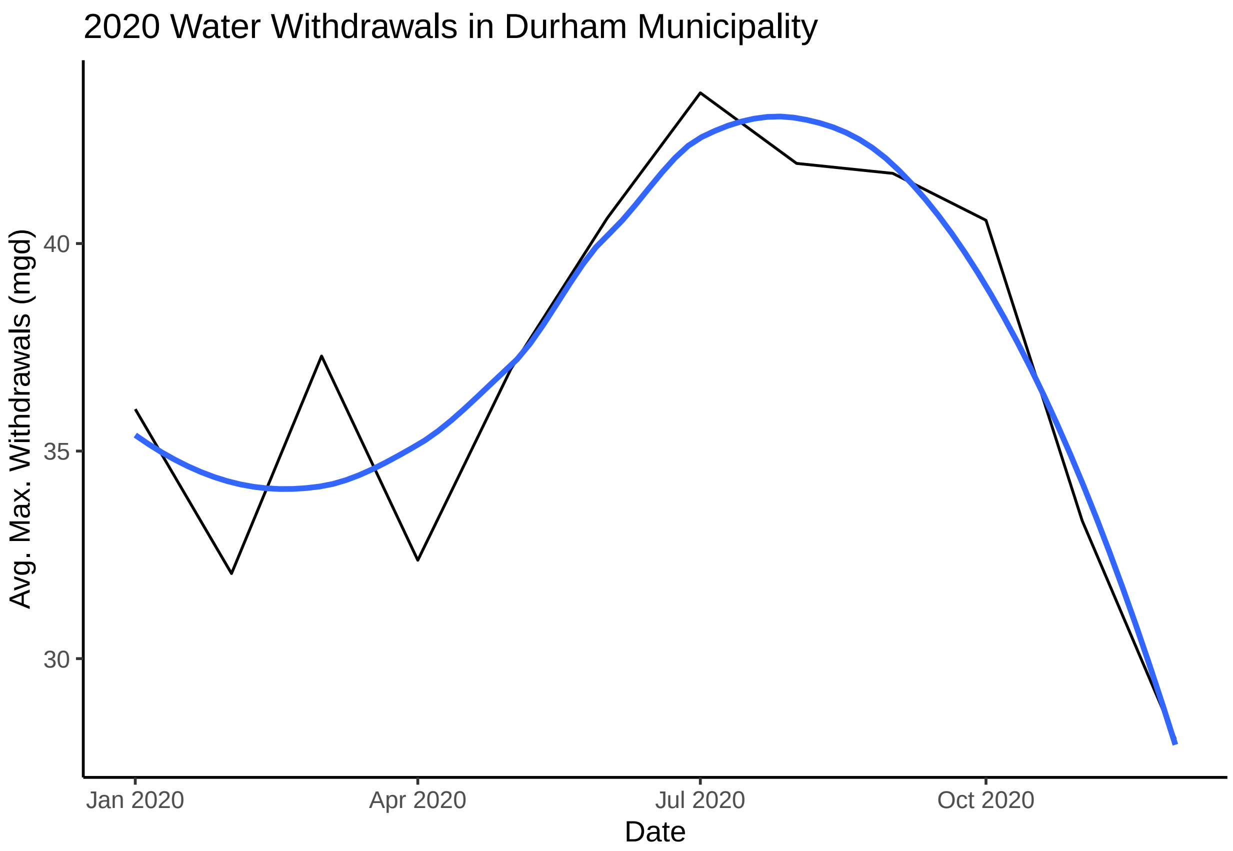
TIP: Use `rep()` to repeat a value when creating a dataframe.

NOTE: It's likely you won't be able to scrape the monthly widthrawal data in order. You can overcome this by creating a month column in the same order the data are scraped: Jan, May, Sept, Feb, etc…

5. Plot the max daily withdrawals across the months for 2020

```
#4
df_maxwith <- data.frame("Month" = c("Jan", "May", "Sep", "Feb",
                                     "June","Oct", "March", "July", "Nov",
                                     "April", "Aug", "Dec"),
                         "Year" = rep(2020, 12),
                         "Avg_Max_Withdrawals" = as.numeric(max.withdrawals.mgd),
                         "PSWID" = pswid,
                         "Water_System" = rep(water.system.name),
                         "Ownership" = rep(ownership)
                         )

df_maxwith <- df_maxwith %>%
  mutate("Date" = my(paste(Month, "-", Year))) %>%
  select(Date, Avg_Max_Withdrawals,
         PSWID, Water_System, Ownership)

df_maxwith <- df_maxwith[order(df_maxwith$Date, decreasing = FALSE),]


#5
maxdaily_plot <- ggplot(df_maxwith, aes(x=Date, y=Avg_Max_Withdrawals)) +
  geom_line()+
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2020 Water Withdrawals in",
                     water.system.name,ownership),
       y="Avg. Max. Withdrawals (mgd)",
       x="Date")

plot(maxdaily_plot)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



6. Note that the PSWID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PSWID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site scraped**.

```
#6.

scrape.NCDEQ <- function(the_year,pswid_num){
  the_url <- paste0("https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=",
                    pswid_num,"&year=",the_year)
  #link website
  the_website <- read_html(the_url)

  #locating elements
  system.name.fun <- the_website %>%
  html_nodes('div+ table tr:nth-child(1) td:nth-child(2)') %>%
  html_text()

  pswid.fun <- the_website %>%
  html_nodes('td tr:nth-child(1) td:nth-child(5)') %>%
  html_text()

  ownership.fun <- the_website %>%
  html_nodes('div+ table tr:nth-child(2) td:nth-child(4)') %>%
  html_text()

  max.withdrawals.fun <- the_website %>%
  html_nodes('th~ td+ td') %>%
  html_text()

  month <- c("Jan", "May", "Sep", "Feb", "June",
             "Oct", "March", "July", "Nov", "April", "Aug", "Dec")

  the_df <- data.frame("Month" = month,
                       "Year" = rep(the_year, 12),
                       "Avg_Max_Withdrawals" = as.numeric(max.withdrawals.fun),
                       "PSWID" = pswid.fun,
                       "Water_System" = rep(system.name.fun),
                       "Ownership" = rep(ownership.fun)
                       )

  the_df <- the_df %>%
  mutate("Date" = my(paste(Month, "-", Year))) %>%
  select(Date, Avg_Max_Withdrawals, PSWID, Water_System, Ownership)

  the_df <- the_df[order(the_df$Date, decreasing = FALSE),]

  return(the_df)

}
```

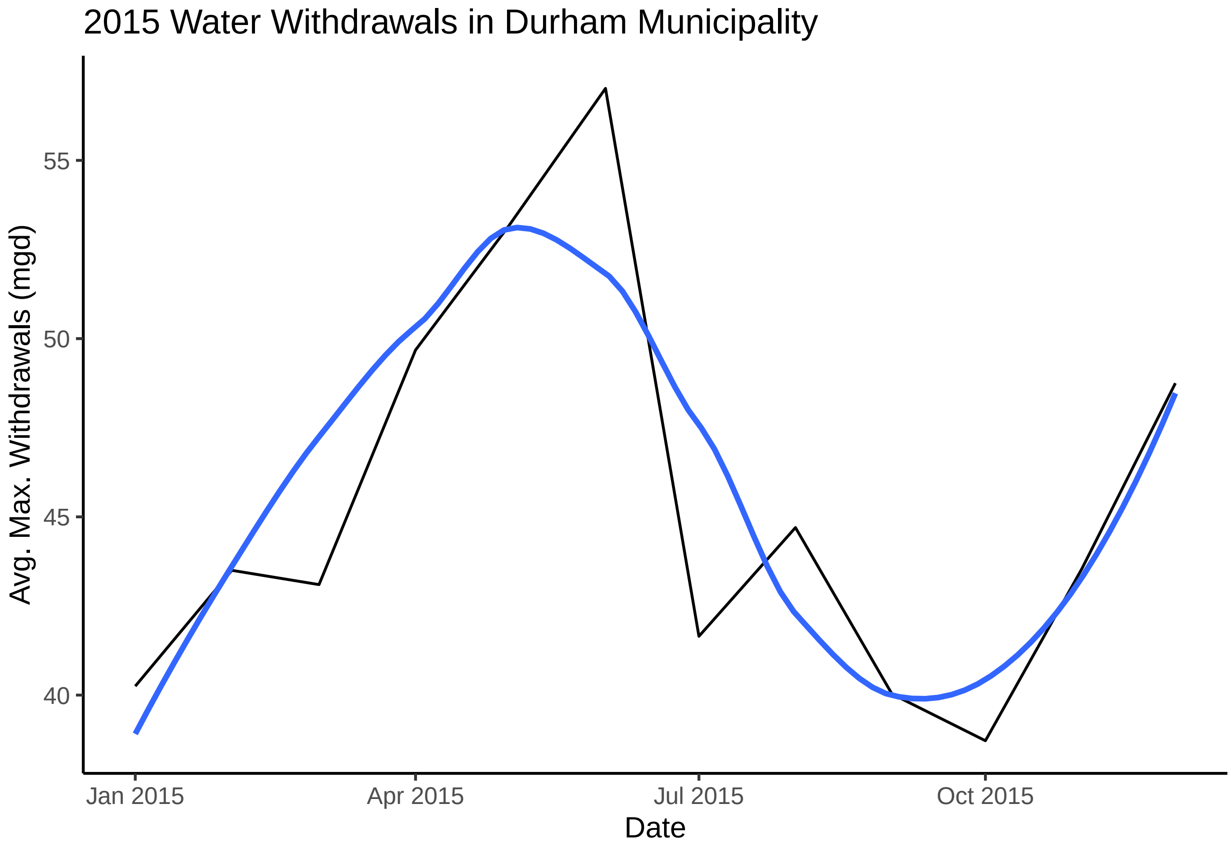
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID='03-32-010') for each month in 2015

```
#7
NDEQ_15 <- scrape.NCDEQ('2015','03-32-010')

summary(NDEQ_15)
```

```
##       Date            Avg_Max_Withdrawals    PSWID           Water_System
##  Min.   :2015-01-01   Min.   :38.72       Length:12          Length:12
##  1st Qu.:2015-03-24   1st Qu.:41.30       Class :character   Class :character
##  Median :2015-06-16   Median :43.52       Mode  :character   Mode  :character
##  Mean   :2015-06-16   Mean   :45.34
##  3rd Qu.:2015-09-08   3rd Qu.:48.98
##  Max.   :2015-12-01   Max.   :57.02
##   Ownership
##  Length:12
##  Class :character
##  Mode  :character
##
##
##
```

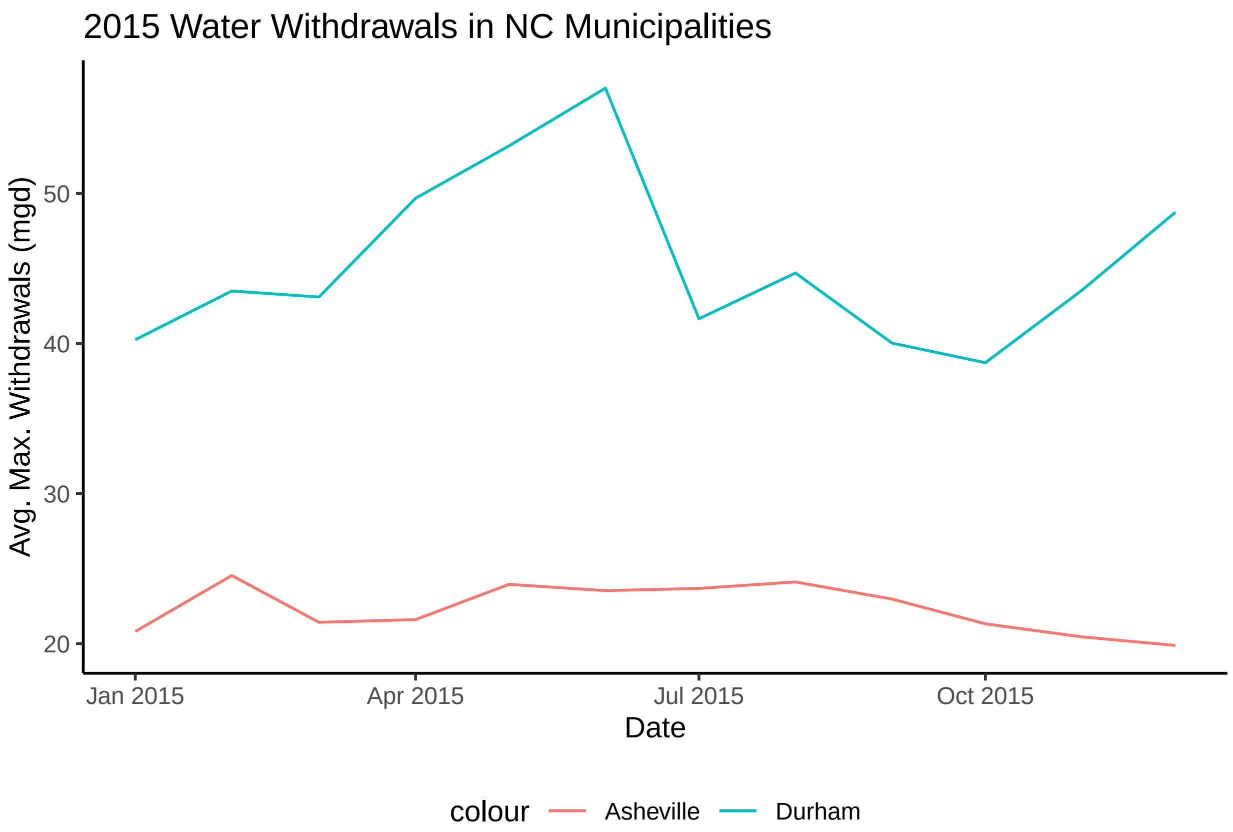
```
ggplot(NDEQ_15, aes(x=Date, y =Avg_Max_Withdrawals))+
  geom_line()+
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2015 Water Withdrawals in",
                     water.system.name,ownership),
       y="Avg. Max. Withdrawals (mgd)",
       x="Date")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares the Asheville to Durham's water withdrawals.

```
#8
ashe.2015 <- scrape.NCDEQ('2015','01-11-010')

ggplot() +
  geom_line(data=ashe.2015,
            aes(x = Date, y =Avg_Max_Withdrawals, color = "Asheville")) +
  geom_line(data=NDEQ_15,
            aes(x = Date, y =Avg_Max_Withdrawals, color = "Durham")) +
  labs(title = "2015 Water Withdrawals in NC Municipalities",
       y="Avg. Max. Withdrawals (mgd)",
       x="Date") +
  theme(legend.position = "bottom")
```



9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

```
#9

ashe.2015 <- scrape.NCDEQ('2015','01-11-010')
ashe.2016 <- scrape.NCDEQ('2016','01-11-010')
ashe.2017 <- scrape.NCDEQ('2017','01-11-010')
ashe.2018 <- scrape.NCDEQ('2018','01-11-010')
ashe.2019 <- scrape.NCDEQ('2019','01-11-010')

ashe.15.16 <- full_join(ashe.2015, ashe.2016)
```

```
## Joining, by = c("Date", "Avg_Max_Withdrawals", "PSWID", "Water_System", "Ownership")
```

```
ashe.15.17 <- full_join(ashe.15.16, ashe.2017)
```

```
## Joining, by = c("Date", "Avg_Max_Withdrawals", "PSWID", "Water_System", "Ownership")
```

```
ashe.15.18 <- full_join(ashe.15.17, ashe.2018)
```

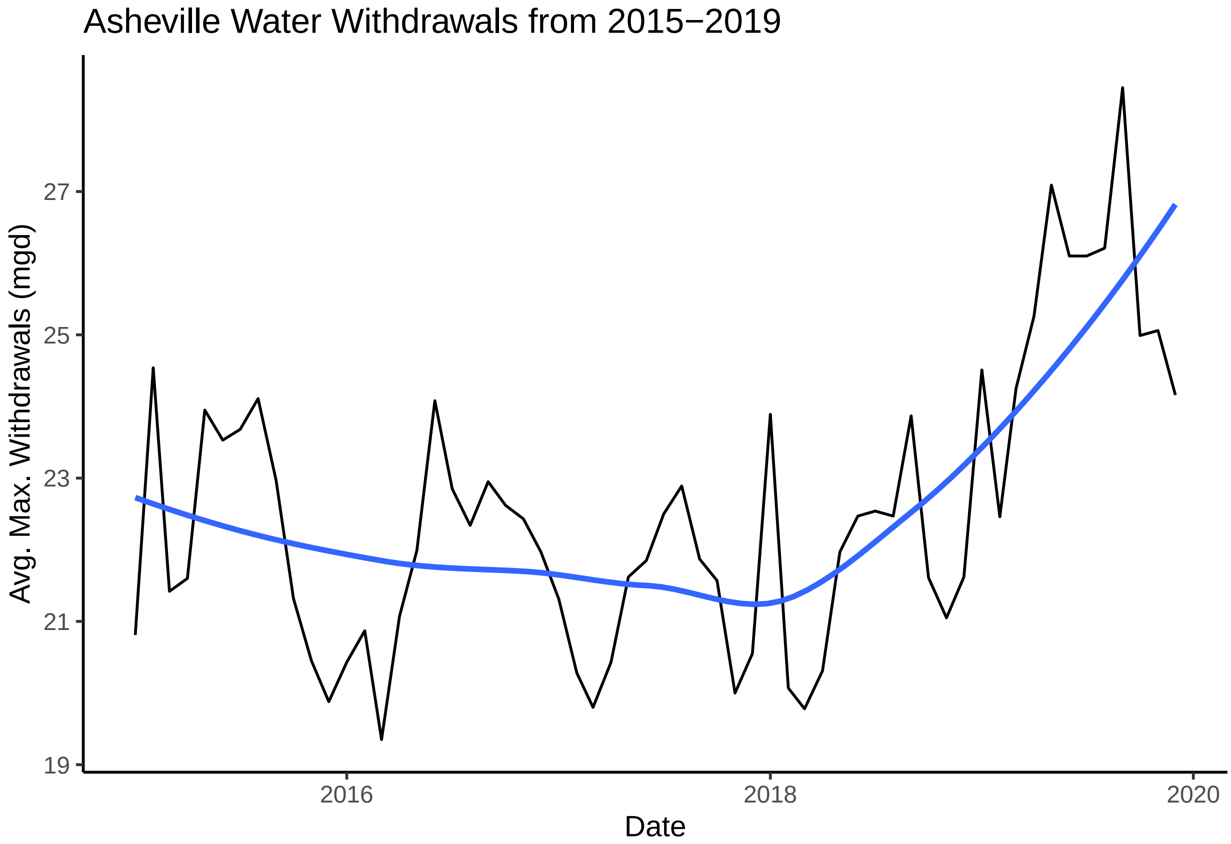
```
## Joining, by = c("Date", "Avg_Max_Withdrawals", "PSWID", "Water_System", "Ownership")
```

```
ashe.15.19 <- full_join(ashe.15.18, ashe.2019)
```

```
## Joining, by = c("Date", "Avg_Max_Withdrawals", "PSWID", "Water_System", "Ownership")
```

```
ggplot(ashe.15.19, aes(x=Date, y =Avg_Max_Withdrawals))+
  geom_line()+
  geom_smooth(method="loess",se=FALSE) +
  labs(title = "Asheville Water Withdrawals from 2015-2019",
       y="Avg. Max. Withdrawals (mgd)",
       x="Date")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Question: Just by looking at the plot (i.e. not running statistics), does Asheville have a trend in water usage over time?

Yes, Asheville's water usage over time has generally increased from 2015 - 2019