# Assignment 10: Data Scraping

Meg Manning

## OVERVIEW

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

## Directions

1. Rename this file `<FirstLast>_A10_DataScraping.Rmd` (replacing `<FirstLast>` with your first and last name).
2. Change "Student Name" on line 3 (above) with your name.
3. Work through the steps, **creating code and output** that fulfill each instruction.
4. Be sure your code is tidy; use line breaks to ensure your code fits in the knitted output.
5. Be sure to **answer the questions** in this assignment document.
6. When you have completed the assignment, **Knit** the text and code into a single PDF file.

## Set up

1. Set up your session:

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

```r
#1
library(tidyverse)
library(rvest)
library(lubridate)
library(here); here()
```

```
## [1] "/Users/margaretmanning/Desktop/GitHub/ENVIRON 872/EDE_Fall2023"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2022 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- 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=2022

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

```
#2 Read in the contents of the webpage
webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
webpage
```

```
## {html_document}
## <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
## [1] <head>\n<title>DWR :: Local Water Supply Planning</title>\n<meta http-equ ...
## [2] <body id="plan">\r\n<!--<div id="division-header">\r\n<a name="top" href= ...
```

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

- From the "1. System Information" section:
- Water system name
- PWSID
- Ownership
- From the "3. Water Supply Sources" section:
- Maximum Day Use (MGD) - for each month

In the code chunk below scrape these values, assigning them to four 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 (represented as strings)".

```
#3 Scrape each variable we want
#Water System Name
water_system <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water_system
```

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

```
#PWSID
PWSID <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
PWSID
```

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

```
#Ownership
ownership <- webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
ownership
```

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

```r
#Maximum day use (MGD) for each month
MGD_bymonth <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()
MGD_bymonth
```

```
##  [1] "36.1000" "43.4200" "52.4900" "30.5000" "42.5900" "34.8800" "39.9100"
##  [8] "43.3200" "32.5300" "34.6600" "41.8000" "37.5300"
```

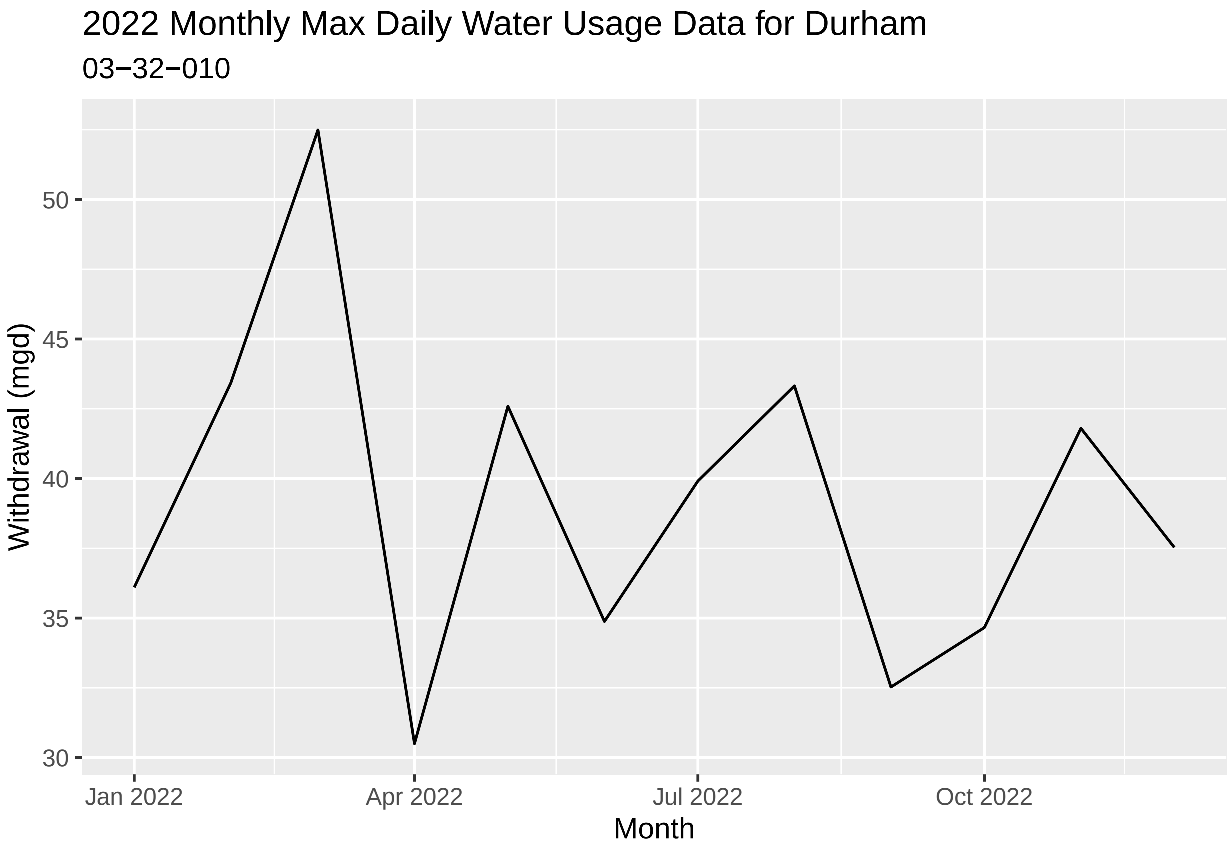
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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 chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc… Or, you could scrape month values from the web page…

5. Create a line plot of the maximum daily withdrawals across the months for 2022

```r
#4 convert to dataframe
df_watersupply <- data.frame("Month" = rep(1:12),
                             "Year" = rep(2022,12),
                             "Max_Day_Use" = as.numeric(MGD_bymonth))

# Tidy up dataframe and add date, ownership, and PWSID
df_watersupply <- df_watersupply %>%
  mutate(water_system = !!water_system,
         PWSID = !!PWSID,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year)))

#5 create line plot of max daily withdrawals across 2022
ggplot(df_watersupply,aes(x=Date,y=Max_Day_Use)) +
  geom_line() +
  labs(title = paste("2022 Monthly Max Daily Water Usage Data for",water_system),
       subtitle = PWSID,
       y="Withdrawal (mgd)",
       x="Month")
```

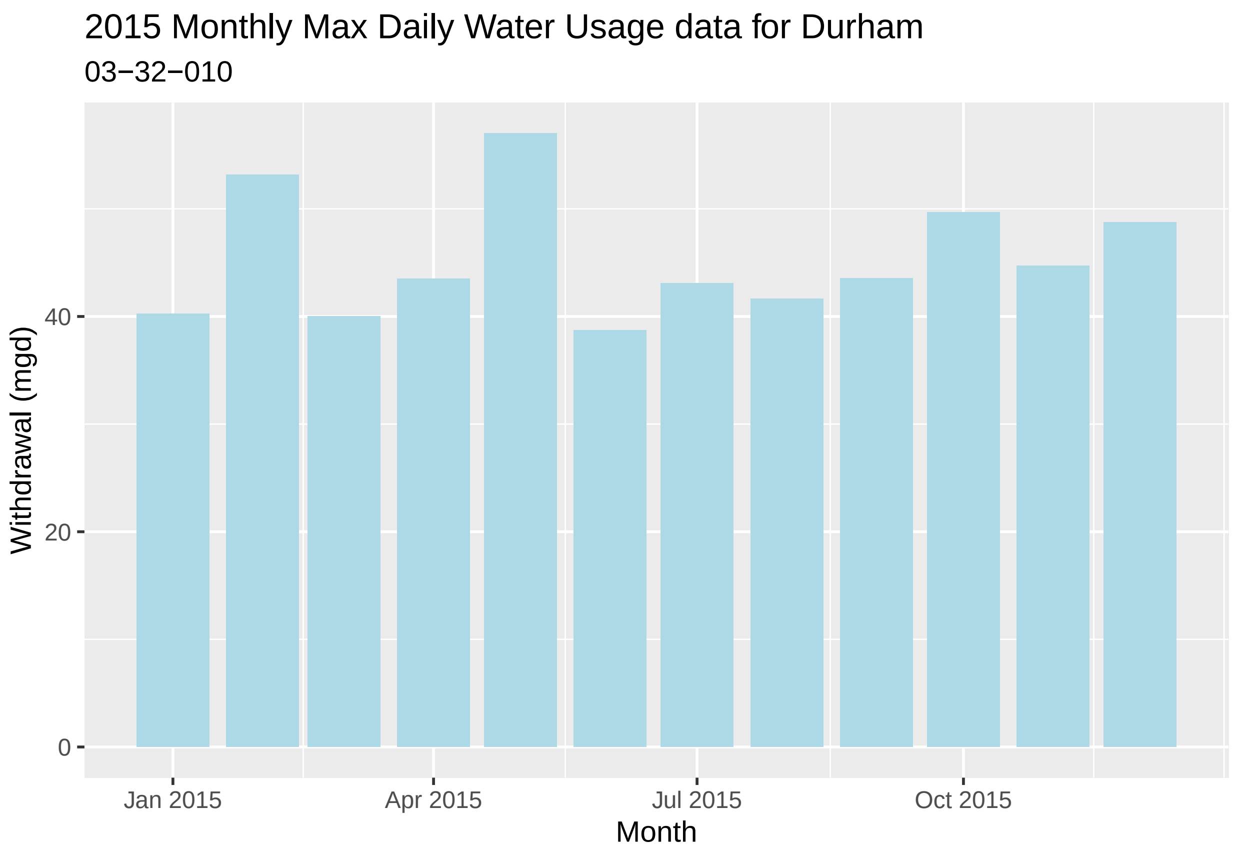


6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6. Construct a function using code above for any PWSID

PWSID <- "03-32-010"
the_year <- 2015

scrape.it <- function(PWSID, the_year){

  the_website <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',
                                  PWSID, "&year=", the_year))

  water_system_tag <- "div+ table tr:nth-child(1) td:nth-child(2)"
  PWSID_tag <- "td tr:nth-child(1) td:nth-child(5)"
  ownership_tag <- "div+ table tr:nth-child(2) td:nth-child(4)"
  MGD_bymonth_tag <- "th~ td+ td"

  water_system <- the_website %>% html_nodes(water_system_tag) %>% html_text()
  PWSID <- the_website %>% html_nodes(PWSID_tag) %>% html_text()
  ownership <- the_website %>% html_nodes(ownership_tag) %>% html_text()
  MGD_bymonth <- the_website %>% html_nodes(MGD_bymonth_tag) %>% html_text()

  df_watersupply <- data.frame("Month" = rep(1:12),
                               "Year" = rep(the_year,12),
                               "Max_Day_Use" = as.numeric(MGD_bymonth)) %>%
    mutate(water_system = !!water_system,
           PWSID = !!PWSID,
           Ownership = !!ownership,
           Date = my(paste(Month,"-",Year)))

  return(df_watersupply)
}
```

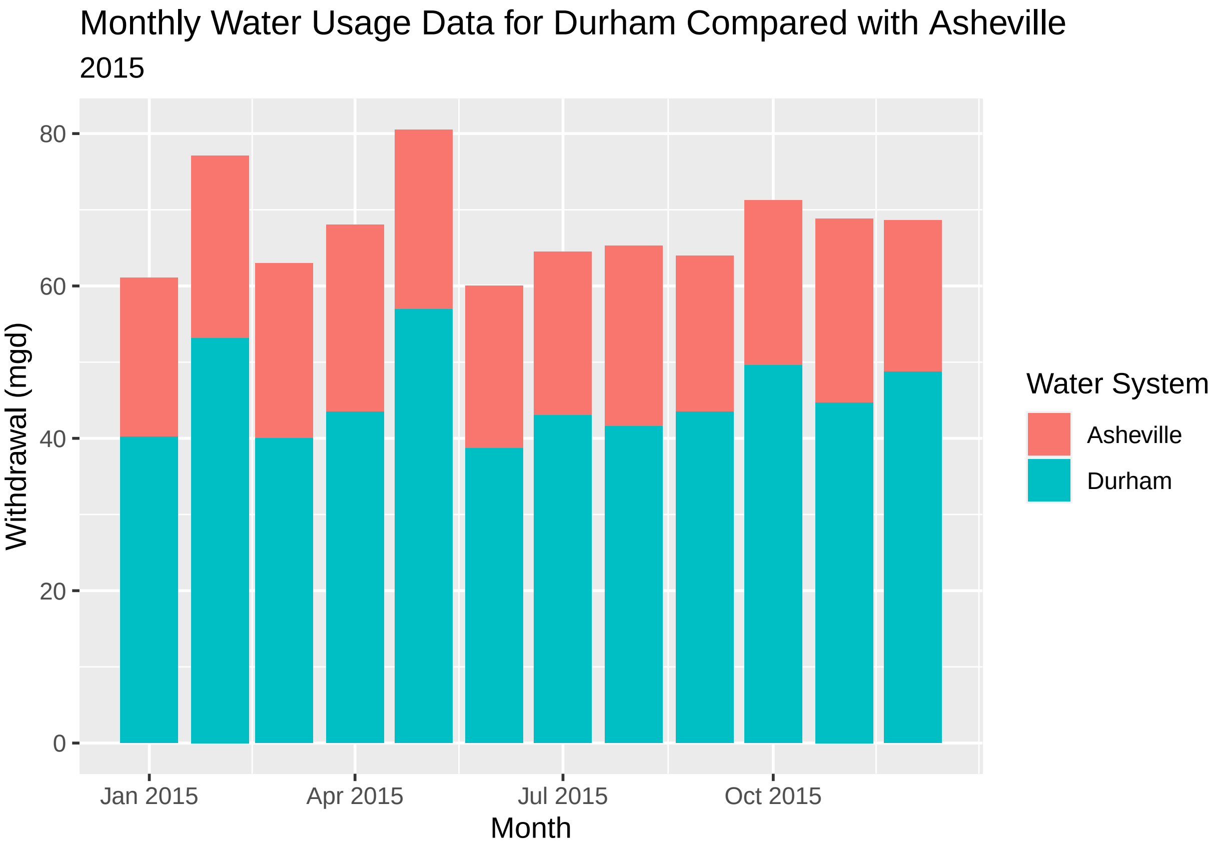
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID='03-32-010') for each month in 2015

```
#7 Extract and plot max daily withdrawals for Durham
durham_df <- scrape.it("03-32-010", 2015)
view(durham_df)

#plot the max daily withdrawals for Durham
ggplot(durham_df,aes(x=Date,y=Max_Day_Use)) +
  geom_col(fill = "lightblue") +
  labs(title = paste(the_year, "Monthly Max Daily Water Usage data for", water_system),
       subtitle = PWSID,
       y="Withdrawal (mgd)",
       x="Month")
```



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 Asheville's to Durham's water withdrawals.

```
#8 Extract data for Asheville in 2015
asheville_df <- scrape.it('01-11-010', 2015)
view(asheville_df)

combined_data <- rbind(durham_df, asheville_df)
view(combined_data)

ggplot(combined_data,aes(y = Max_Day_Use, x=Date, fill = water_system)) +
  geom_col()+
  labs(title = paste("Monthly Water Usage Data for Durham Compared with Asheville"),
       subtitle = "2015",
       y="Withdrawal (mgd)",
       x="Month",
       fill = "Water System")
```



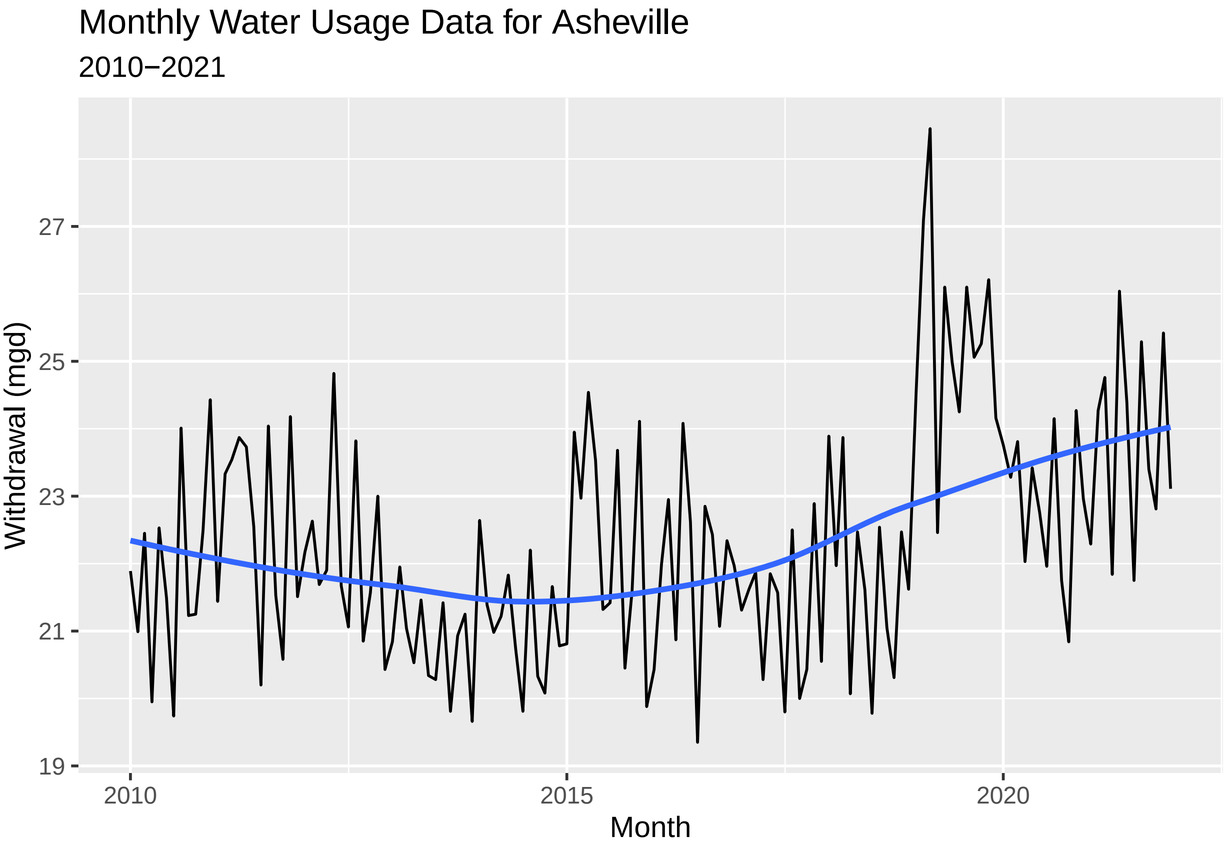
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   TIP: See Section 3.2 in the "10_Data_Scraping.Rmd" where we apply "map2()" to iteratively run a function over two inputs. Pipe the output of the map2() function to `bindrows()` to combine the dataframes into a single one.

```
#9 "Map" the "scrape.it" function to retrieve data for 2010 through 2021
the_years <- seq(2010,2021)
PWSID <- rep("01-11-010", length(the_years))
```

```
df_2010_2021 <- map2(PWSID, the_years, scrape.it) %>%
  bind_rows()

#Plot Asheville's Max Daily Withdrawal by Months
ggplot(df_2010_2021,aes(y = Max_Day_Use, x=Date)) +
  geom_line()+
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = paste("Monthly Water Usage Data for Asheville"),
       subtitle = "2010-2021",
       y="Withdrawal (mgd)",
       x="Month")
```

```
## `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? > Answer: Based on the curved line, it appears that Asheville's daily water withdrawal decreased slightly from around 2010 to 2015, but since 2015 it seems to have a positive trend, and overall it appears that Asheville is increasing their daily water usage. >