Assignment 09: Data Scraping

Isabel Zungailia

OVERVIEW

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

Directions

1. Rename this file `<FirstLast>_A09_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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

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.
#Check working directory
getwd()
```

```
## [1] "/home/guest/EDA-Fall2022/Assignments"
```

```
#Load necessary packages
library(tidyverse)
library(lubridate)
library(rvest)
library(dplyr)

#Set theme
mytheme <- theme_classic() +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2021 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=2021

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

```
#2.Use rvest's `read_html()` function to bring the contents of the website into our coding environment
webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2021')
```

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:
- Maximum Daily 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), with the first value being "27.6400".

```
#3.
#Set the element address variables
water.system.name_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
pswid_tag <- 'td tr:nth-child(1) td:nth-child(5)'
ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
max.withdrawals.mgd_tag <- 'th~ td+ td'

#Scrape the data items
water.system.name <- webpage %>% html_nodes(water.system.name_tag) %>% html_text()
pswid <- webpage %>%   html_nodes(pswid_tag) %>%  html_text()
ownership <- webpage %>% html_nodes(ownership_tag) %>% html_text()
max.withdrawals.mgd <- webpage %>% html_nodes(max.withdrawals.mgd_tag) %>% html_text()

#Check values
water.system.name
```

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

```
pswid
```

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

```
ownership
```

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

```
max.withdrawals.mgd
```

```
##  [1] "27.6400" "41.7900" "36.7200" "27.9700" "37.9500" "42.2400" "30.5400"
##  [8] "43.6200" "31.2800" "33.7600" "46.0800" "29.7800"
```

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.)

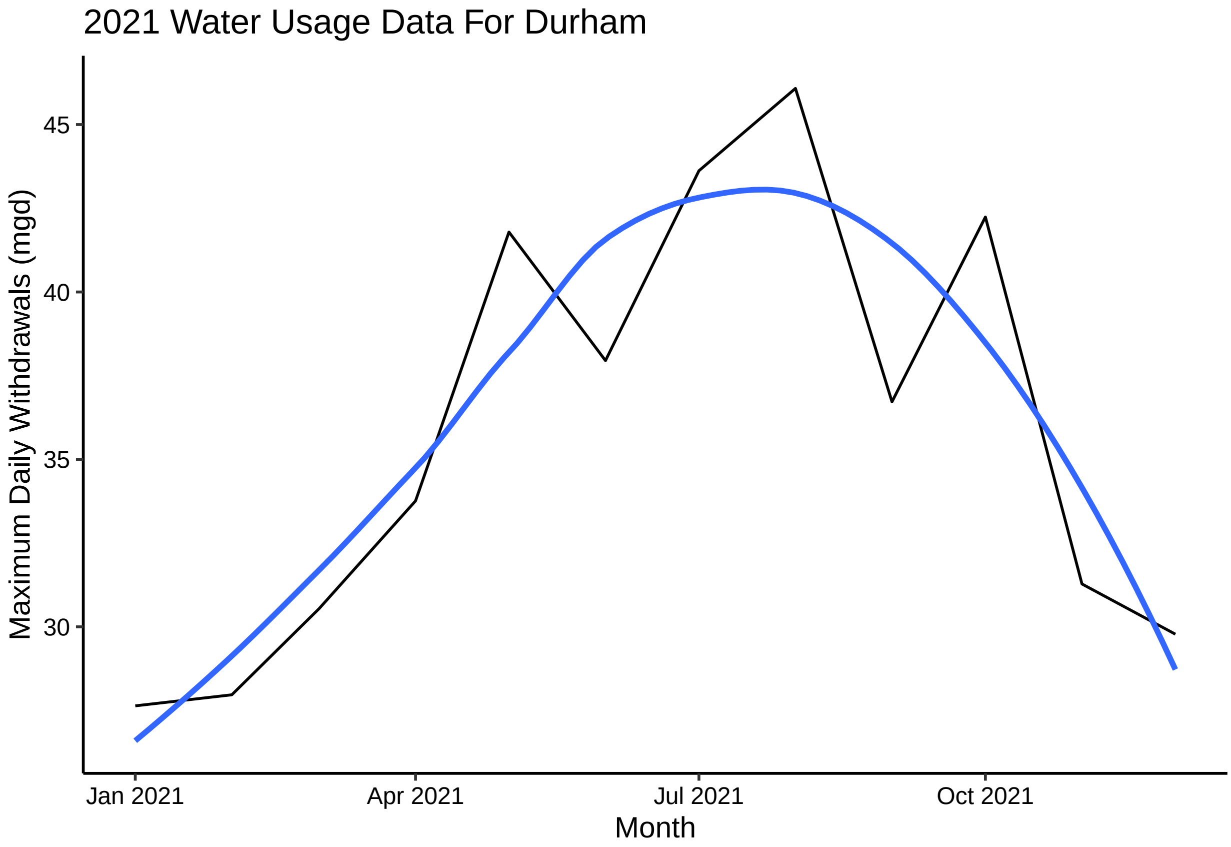
   TIP: Use `rep()` to repeat a value when creating a dataframe.

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…

5. Create a line plot of the maximum daily withdrawals across the months for 2021

```
#4. Convert scraped data into a dataframe
df_withdrawals <- data.frame(Month_abbr = c("Jan", "May", "Sept", "Feb", "June", "Oct",
                                            "Mar", "Jul", "Nov", "Apr", "Aug", "Dec"),
                             Month = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12),
                             Year = rep(2021,12),
                             Max_withdrawals_mgd = as.numeric(max.withdrawals.mgd))

df_withdrawals <- df_withdrawals %>%
  mutate(Water_System_Name = !!water.system.name, #Add necessary columns
         PSWID = !!pswid,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year))) %>% #Create a date column
  arrange(Month) #Arrange by Month

#5. Create a line plot of the maximum daily withdrawals across the months for 2021
ggplot(df_withdrawals,aes(x=Date,y=Max_withdrawals_mgd)) +
  geom_line(aes(group=1)) +
  geom_smooth(method="loess",se=FALSE) +
 labs(title = paste("2021 Water Usage Data For",water.system.name),
       y="Maximum Daily Withdrawals (mgd)",
       x="Month")
```

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



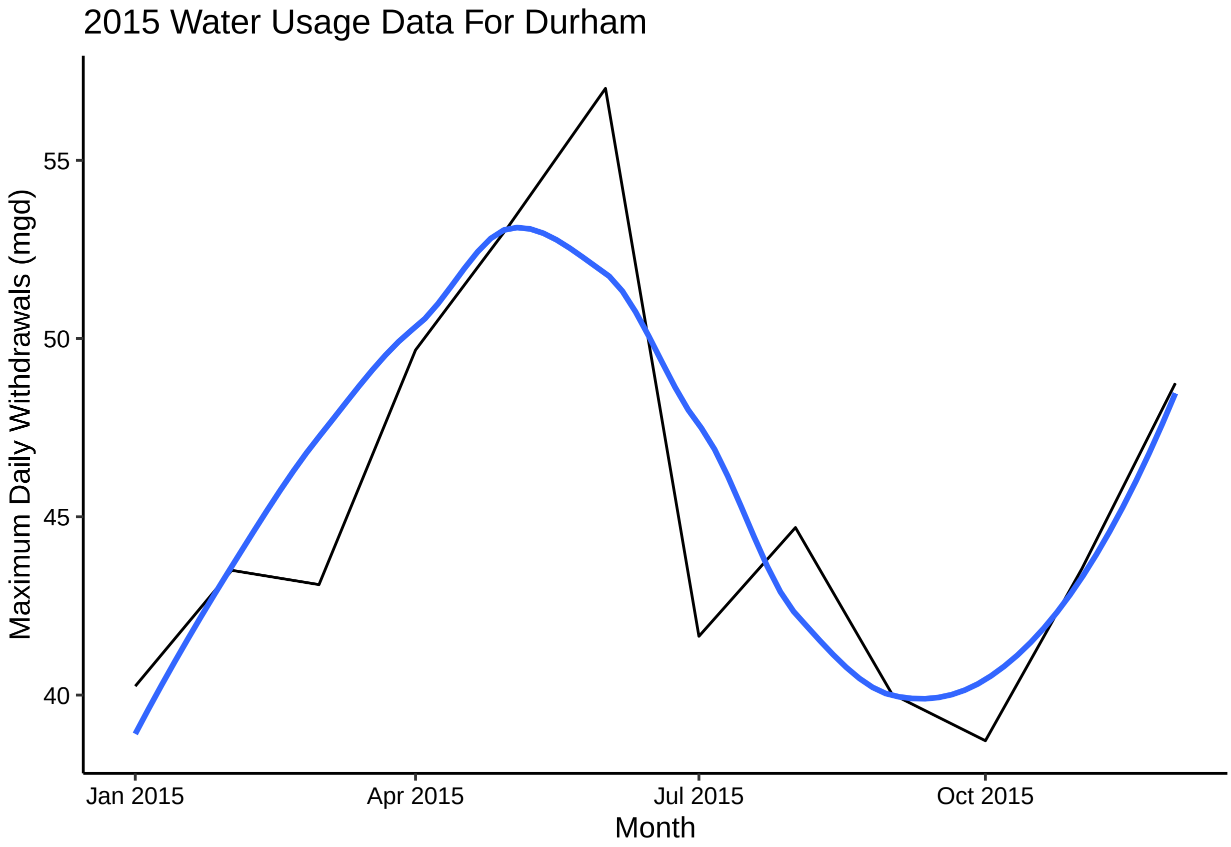
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6.
#Constructing a function to scrape for any PWSID and year for which the NC DEQ has data
scrape.it <- function(the_year, the_pwsid){

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

#Set the element address variables
water.system.name_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
pswid_tag <- 'td tr:nth-child(1) td:nth-child(5)'
ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
max.withdrawals.mgd_tag <- 'th~ td+ td'

#Scrape the data items
water.system.name <- the_website %>% html_nodes(water.system.name_tag) %>% html_text()
pswid <- the_website %>%   html_nodes(pswid_tag) %>%  html_text()
ownership <- the_website %>% html_nodes(ownership_tag) %>% html_text()
max.withdrawals.mgd <- the_website %>% html_nodes(max.withdrawals.mgd_tag) %>% html_text()

#Convert to a dataframe
  df_withdrawals2 <- data.frame(Month_abbr = c("Jan", "May", "Sept", "Feb", "June", "Oct",
                                               "Mar", "Jul", "Nov", "Apr", "Aug", "Dec"),
                                Month = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4, 8, 12))
  df_withdrawals2 <- df_withdrawals2 %>%
  mutate(Water_System_Name = !!water.system.name,
         Ownership = !!ownership,
         Max_Withdrawals_mgd = !!max.withdrawals.mgd,
         PSWID = !!the_pwsid,
         Year = the_year,
        Date = my(paste(Month,"-",Year))) %>%
  arrange(Month)

#Return the dataframe
  return(df_withdrawals2)
}
```

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

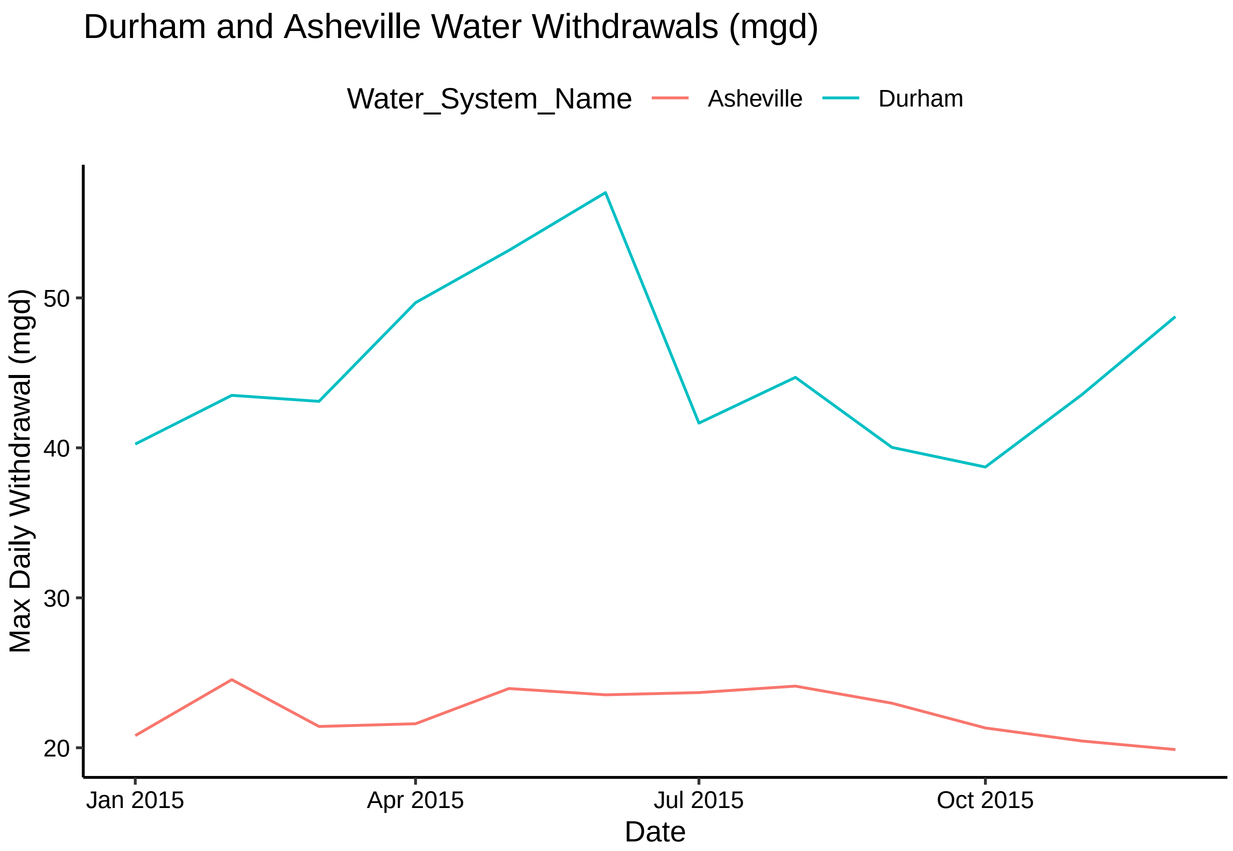
```
#7.
#Use function to extract max daily withdrawals for Durham for each month in 2015
durham_2015<- scrape.it(2015,'03-32-010')
view(durham_2015)

#Plot the results
ggplot(durham_2015,aes(x=Date,y=as.numeric(Max_Withdrawals_mgd))) +
  geom_line(aes(group=1)) +
  geom_smooth(method="loess",se=FALSE) +
 labs(title = paste("2015 Water Usage Data For",water.system.name),
      y="Maximum Daily Withdrawals (mgd)",
      x="Month")
```

```
## `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 Asheville's to Durham's water withdrawals.

```
#8.
#Use function to extract max daily withdrawals for Asheville for each month in 2015
asheville_2015 <- scrape.it(2015,'01-11-010')
view(asheville_2015)

#Combine Durham and Asheville
Durham_Asheville <- rbind(durham_2015, asheville_2015)

#Plot the results
Durham_Asheville_plot <-
  ggplot(Durham_Asheville, aes(x=Date, y=as.numeric(Max_Withdrawals_mgd),color = Water_System_Name)) +
  geom_line() +
  labs(title ="Durham and Asheville Water Withdrawals (mgd)",
    y="Max Daily Withdrawal (mgd)",
       x="Date")

Durham_Asheville_plot
```

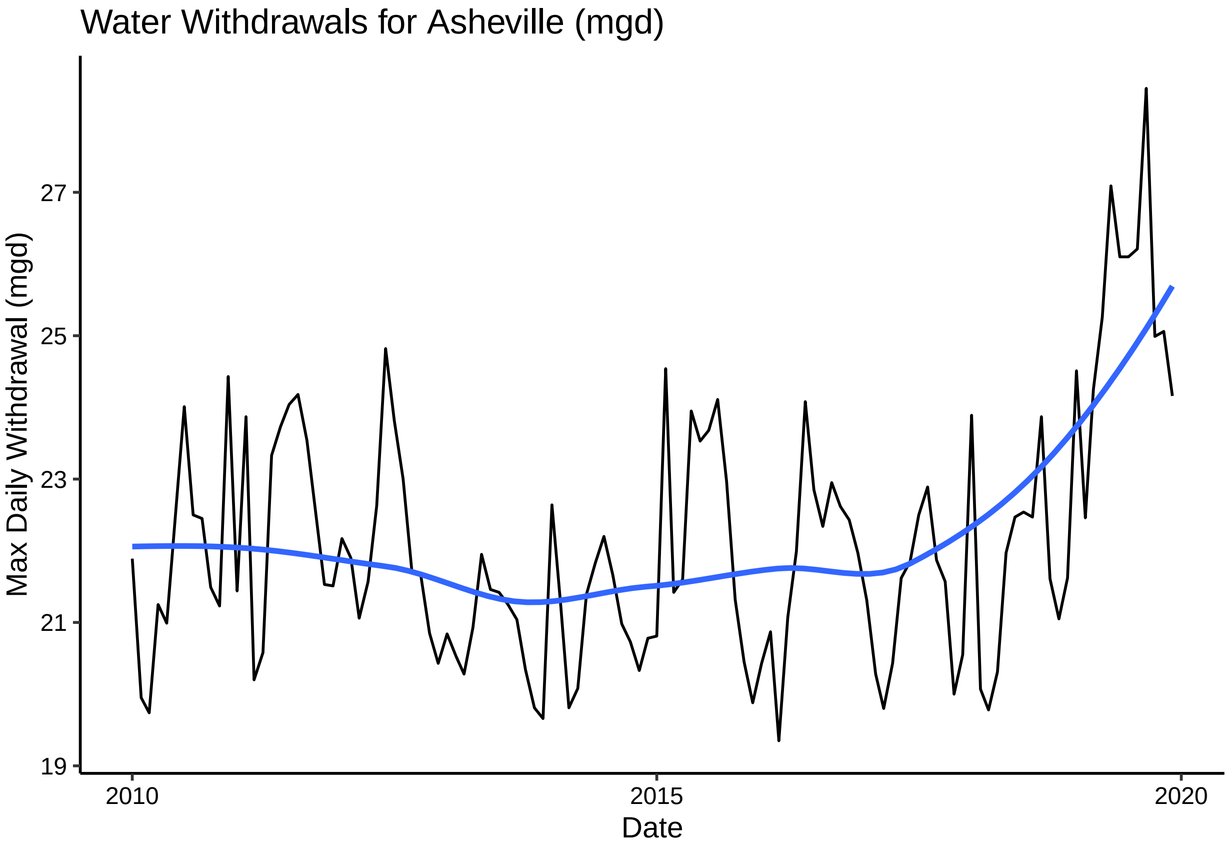


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

   TIP: See Section 3.2 in the "09_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.
#Set desired dates
years <- seq(2010,2019)

#Use map2 function
Asheville_2010_thru_2019 <- map2(years,"01-11-010", scrape.it)

#Combine data frames into a single one (using bind_rows)
Asheville_2010_thru_2019_final <- bind_rows(Asheville_2010_thru_2019)

ggplot(Asheville_2010_thru_2019_final,aes(x=Date,y=as.numeric(Max_Withdrawals_mgd))) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title ="Water Withdrawals for Asheville (mgd) ",
    y="Max Daily Withdrawal (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? Just by looking at the plot, it is clear that Asheville's water usage has fluctuated over time, but the overall trend appears to be somewhat level between the years 2010 up until 2017. There appears to be an increasing trend in the MGD values between 2017 and 2020.