Assignment 10: Data Scraping

Sydney Williams

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

```
#1
library(tidyverse); library(rvest); library(lubridate);
library(here);library(ggplot2); library(purrr); library(dplyr)

here()
```

```
## [1] "/home/guest/EDA_Spring2024"
```

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
theURL <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022'
```

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

# scraping the data from 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= ...
```

```
water_system_name <- webpage %>% html_nodes('td tr:nth-child(1) td:nth-child(2)') %>% html_text()
PWSID <- webpage %>% html_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html_text()
Ownership <- webpage %>% html_nodes ('div+ table tr:nth-child(2) td:nth-child(4)') %>% html_text()

MGD <- webpage %>% html_nodes('th~ td+ td')%>% html_text()
MGD <- MGD[1:12]
```

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.)

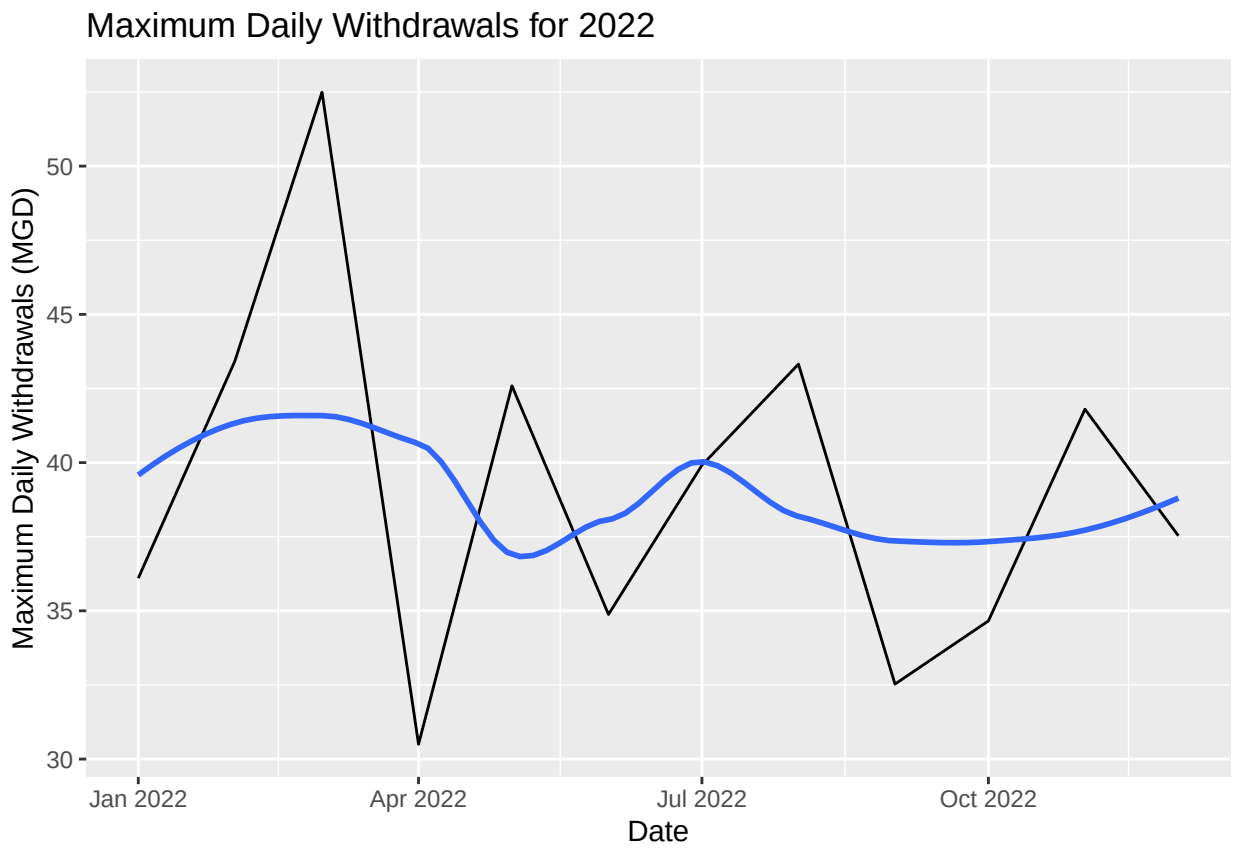
   > TIP: Use `rep()` to repeat a value when creating a dataframe.
   >
   > NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "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

```
#4
# creating dataframe with scraped data
durham_water_df <- data.frame(
  "Water System Name" = rep(12),
  "PWSID" = rep(12),
  "Ownership" = rep(12),
  "MGD" = as.numeric(MGD),
  "Month" = rep(1:12),
  "Year" = rep(2022, 12))

# Adding a Date column by pasting Year and Month
durham_water_df <- durham_water_df  %>%
  mutate(Date = my(paste(Month,"-",Year)))

#5

# Creating a plot with scraped data
```

```
ggplot(durham_water_df, aes(x = Date, y = MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = "Maximum Daily Withdrawals for 2022",
       y = "Maximum Daily Withdrawals (MGD)",
       x = "Date")
```

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



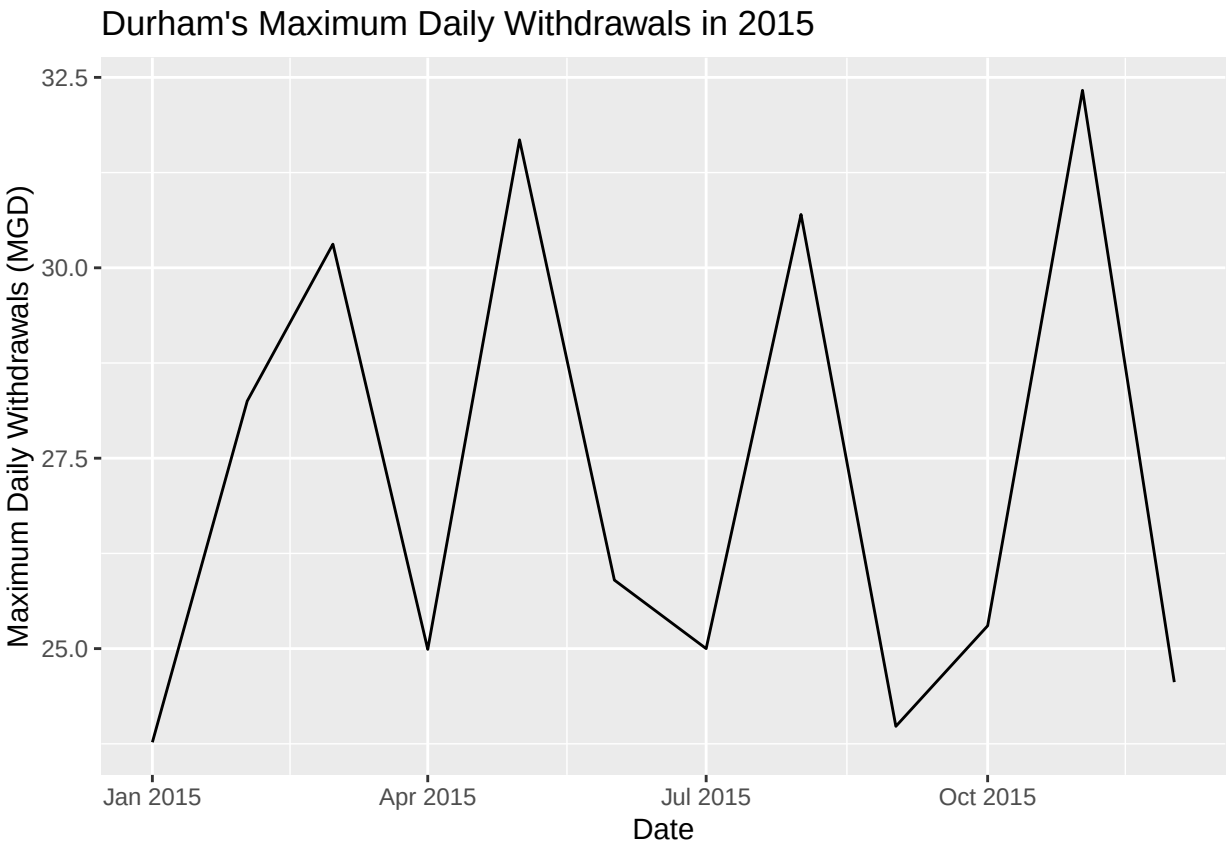
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6.
scrape_water_data <- function(PWSID, year) {
  url <- paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=', PWSID, '&year=', year)}

 # Read the webpage
  webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=')

# checking
scrape_water_data('03-32-010', 2022)
```

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

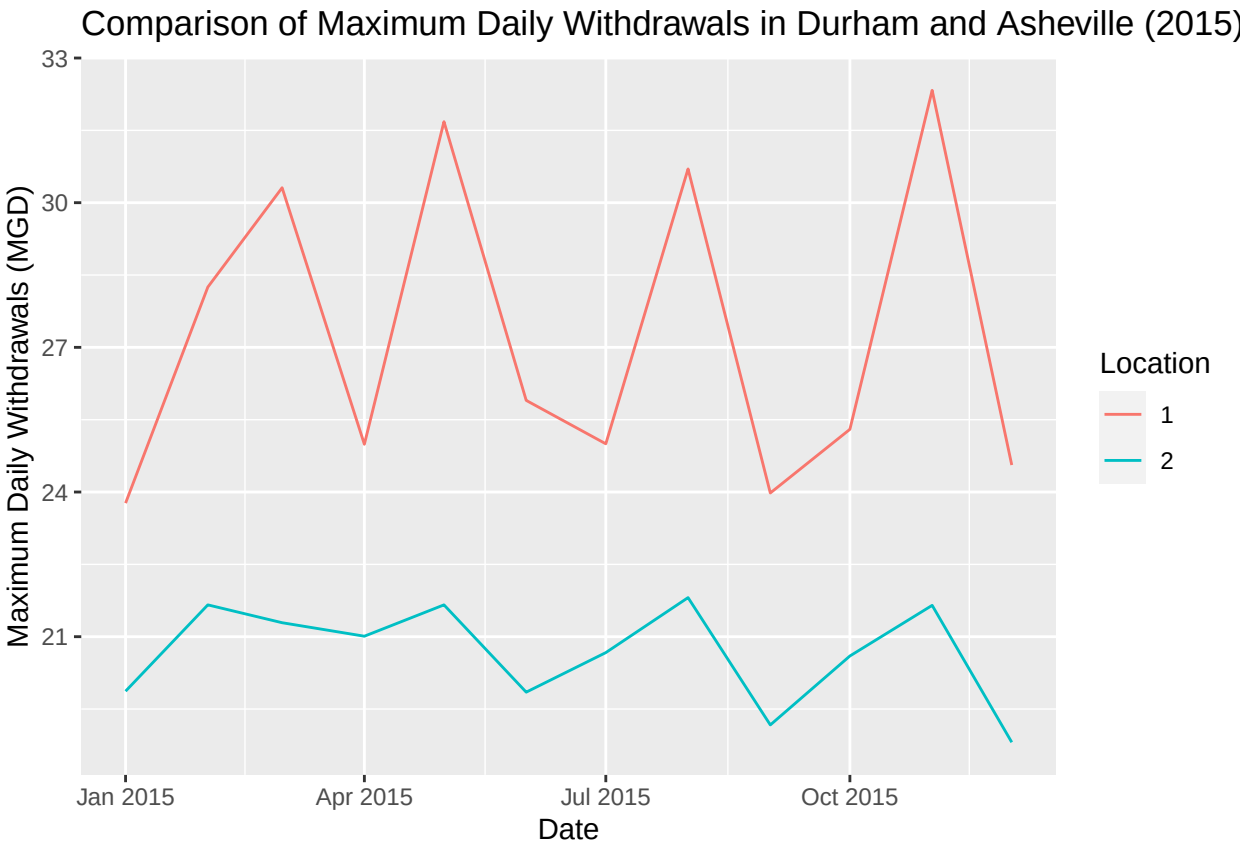
```
scrape_water_data <- function(PWSID, year) {
  url <- paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=', PWSID, '&year=', year)
```

```r
  # Read the webpage
  webpage <- read_html(url)

  # Scraping data from the webpage
  water_system_name <- webpage %>% html_nodes('td tr:nth-child(1) td:nth-child(2)') %>% html_text()
  PWSID <- webpage %>% html_nodes('td tr:nth-child(1) td:nth-child(5)') %>% html_text()
  Ownership <- webpage %>% html_nodes ('div+ table tr:nth-child(2) td:nth-child(4)') %>% html_text()
  MGD <- webpage %>% html_nodes('th+ td') %>% html_text()
  MGD <- MGD[1:12]

  # Manually assigning months
  months <- month.abb

  # Creating a dataframe with scraped data
  water_df <- data.frame(
    'Water System Name' = rep(12),
    'PWSID' = rep(12),
    'Ownership' = rep(12),
    'MGD' = as.numeric(MGD),
    'Month' = months,
    'Year' = rep(year, 12)
  )

  # Adding a Date column by pasting Year and Month
  water_df <- water_df %>%
    mutate(Date = ymd(paste(Year, Month, "01", sep = "-")))

  return(water_df)
}
# Scraping data for Durham in 2015
durham_2015_data <- scrape_water_data('03-32-010', 2015)

# Plotting max daily withdrawal for months for 2015
ggplot(durham_2015_data, aes(x = Date, y = MGD)) +
  geom_line() +
  labs(title = "Durham's Maximum Daily Withdrawals in 2015",
       x = "Date",
       y = "Maximum Daily Withdrawals (MGD)")
```



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
# Scrape data for Asheville in 2015
asheville_2015_data <- scrape_water_data('01-11-010', 2015)

# Combining Durham and Asheville data
binded_data <- bind_rows(durham_2015_data, asheville_2015_data, .id = "Location")

# Plotting comparison of max daily withdrawals
ggplot(binded_data, aes(x = Date, y = MGD, color = Location)) +
  geom_line() +
  labs(title = "Comparison of Maximum Daily Withdrawals in Durham and Asheville (2015)",
       x = "Date",
       y = "Maximum Daily Withdrawals (MGD)",
       color = "Location")
```



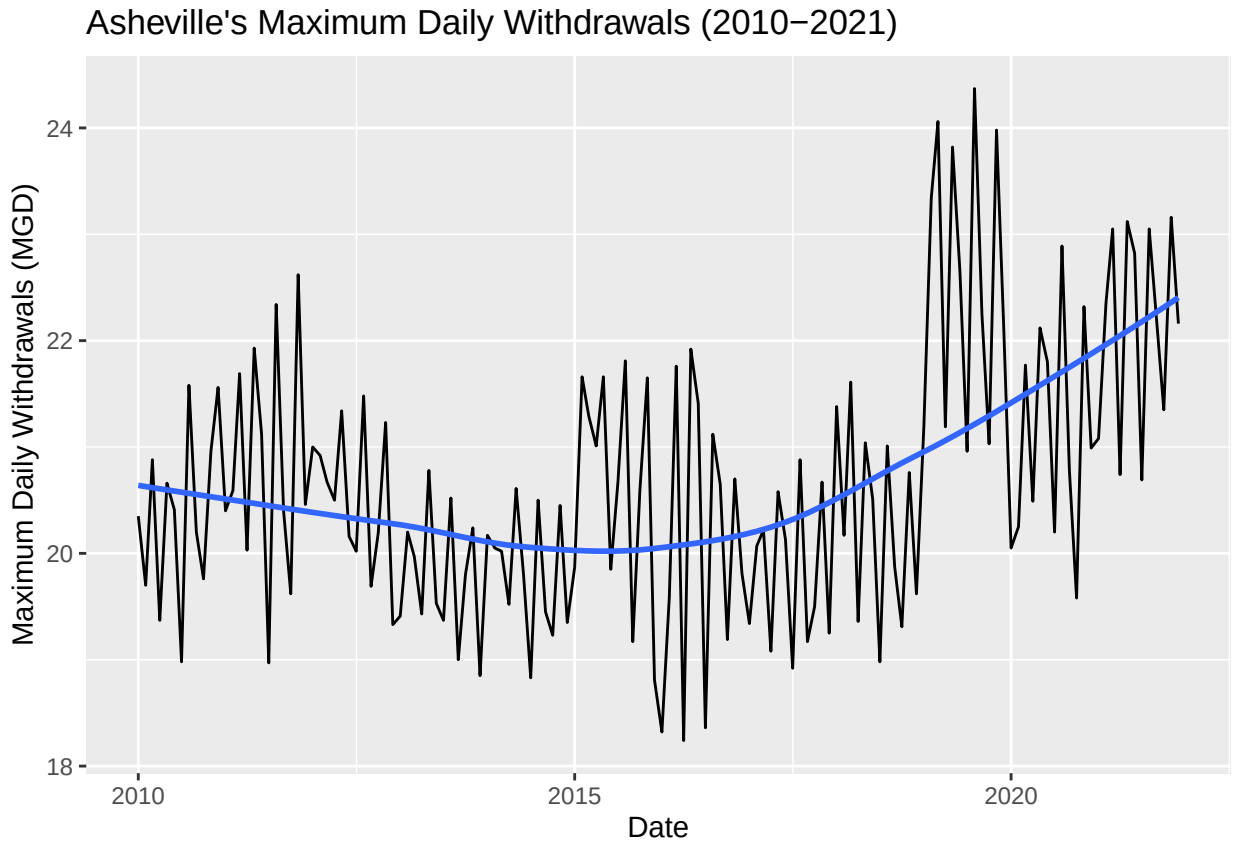
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   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
# Define the function to scrape water data for Asheville for a given year
asheville_data_function <- function(year) {
  # Scraping data for Asheville (PWSID='01-11-010') for the given year
  asheville_data_scrape <- scrape_water_data('01-11-010', year)
  return(asheville_data_scrape)
}

# Apply the function iteratively over the years 2010 to 2021
asheville_data_list <- map(2010:2021, asheville_data_function)

# Combine the resulting data frames into a single one
asheville_combined_data <- bind_rows(asheville_data_list)

# Plot max daily withdrawals across months for the years 2010 to 2021
ggplot(asheville_combined_data, aes(x = Date, y = MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = "Asheville's Maximum Daily Withdrawals (2010-2021)",
       x = "Date",
```

```
       y = "Maximum Daily Withdrawals (MGD)")
```

```
## `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:Asheville's water usage is generally increasing over time, with fluctuations from year to year. The smoothed line suggest there has been a steady increase in maximum daily withdrawals. >