Assignment 10: Data Scraping

Fiona Kelley

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 Loading the necessary packages.

library(tidyverse)
#install.packages("rvest")
library(rvest)

#Check working directory.
getwd()
```

```
## [1] "/home/guest/R/EDA-Spring2023"
```

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 Reading and setting the website as an object to be scraped.

LWSP_webpage <- read_html(
  'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
LWSP_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 Gathering the data from the webpage and assigning them to four separate variables.

WaterSystem_name <- LWSP_webpage %>%
  html_nodes("div+ table tr:nth-child(1)
             td:nth-child(2)") %>%
  html_text()

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

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

MGD <- LWSP_webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()
```

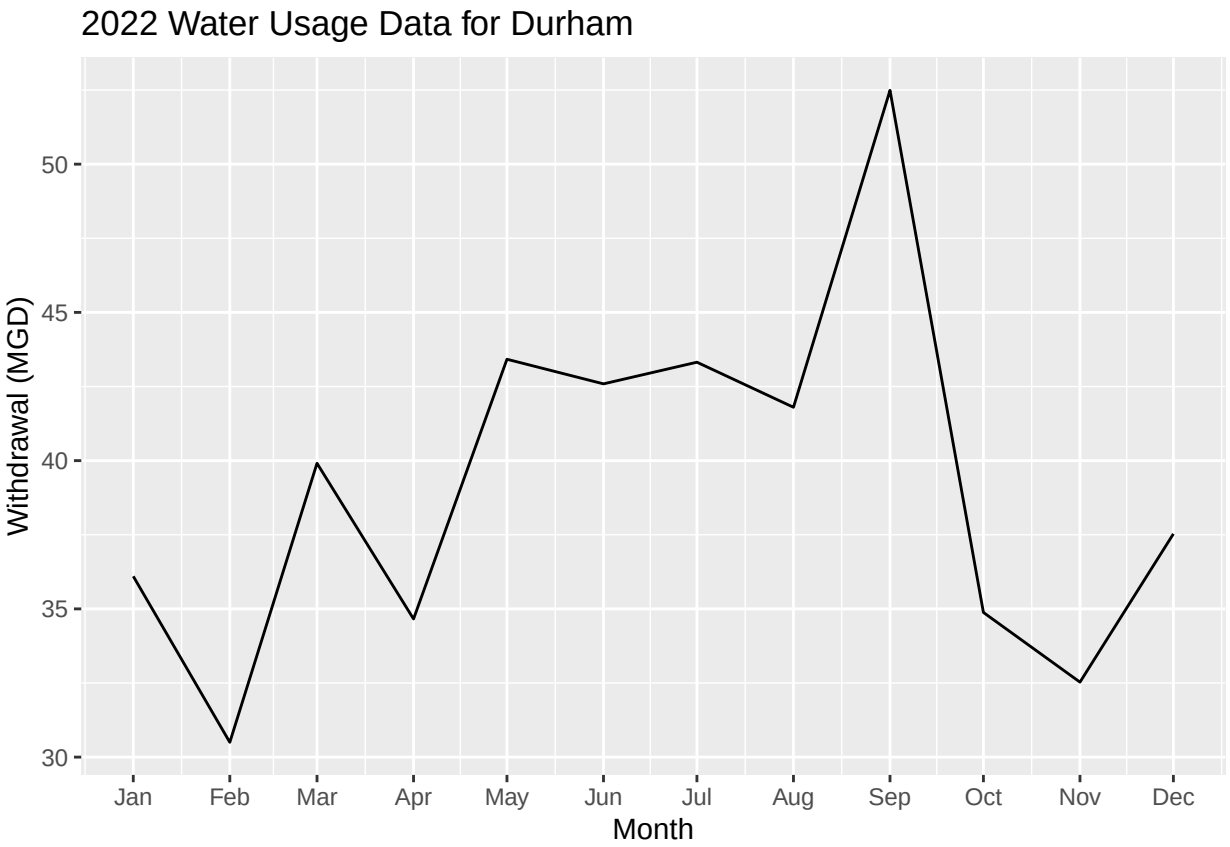
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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

#Create a dataframe of data.

df_data <- data.frame(
  WaterSystem_name = WaterSystem_name,
  PWSID = PWSID,
  Ownership = Ownership,
  "Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
  "Year"=2022,
  MGD = as.numeric(MGD))

#Making a date column read as a date variable.
df_data$Date <- as.Date(my(paste(df_data$Month,"-",df_data$Year)))

#5
#Plotting the data.
Withdrawal_plot <- ggplot(df_data,aes(x=Date,y=MGD)) + geom_line() +
  labs(title = paste("2022 Water Usage Data for",WaterSystem_name),
       y="Withdrawal (MGD)",
       x="Month") +
  scale_x_date(date_breaks = "1 month", date_labels = "%b")

Withdrawal_plot
```



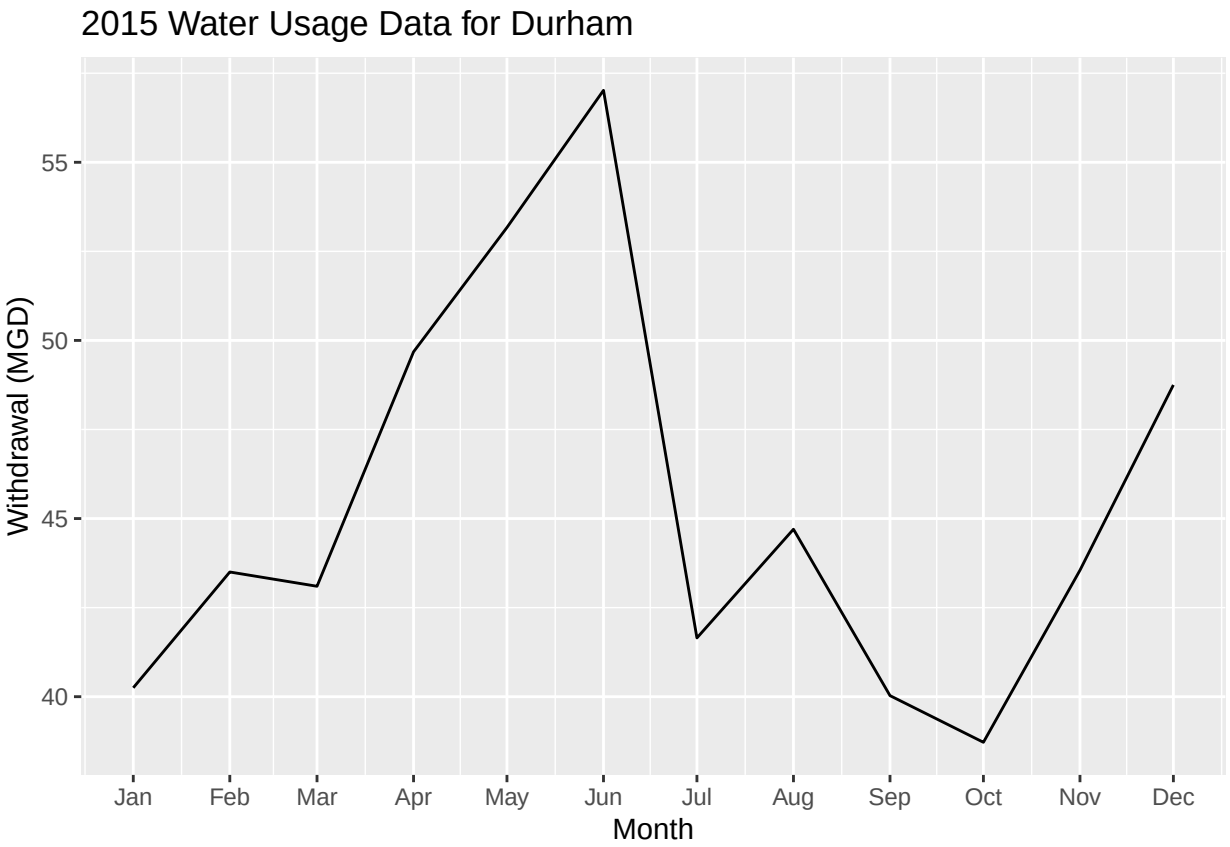
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6.
#Constructing the scraping web address.
the_base_url <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid='
the_site <- '03-32-010'
the_year <- 2015

#Retrieve the website contents.
scrape.it <- function(the_site, the_year){
  the_website <- read_html(paste0(the_base_url, the_site, '&year=', the_year))

#Set variables.
WaterSystem_name_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_tag <- 'th~ td+ td'

#Scrape the data items.
the_WaterSystemName <- the_website %>%html_nodes(WaterSystem_name_tag) %>% html_text()
the_PWSID <- the_website %>%   html_nodes(PWSID_tag) %>%  html_text()
the_Ownership <- the_website %>% html_nodes(Ownership_tag) %>% html_text()
the_MGD <- the_website %>% html_nodes(MGD_tag) %>% html_text()
```

```
the_Month <- c(1,5,9,2,6,10,3,7,11,4,8,12)

#Constructing a dataframe from the scraped data.
df_mgd_data <- data.frame(
  "Month" = the_Month,
  "Year" = rep(the_year, length(the_Month)),
  "Withdrawals" = as.numeric(the_MGD)) %>%
  mutate(
    WaterSystem = !!the_WaterSystemName,
    PWSID = !!the_PWSID,
    Ownership = !!the_Ownership,
    Date = my(paste(Month,"-",Year)))

return(df_mgd_data)
}
```

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

```
#7
#Fetch data for Durham 2015.
DurhamPWSID_2015 <- scrape.it('03-32-010', 2015)

#Plotting the data.
Durham2015_plot <- ggplot(DurhamPWSID_2015,aes(x=Date,y=Withdrawals)) + geom_line() +
  labs(title = paste("2015 Water Usage Data for Durham"),
       y="Withdrawal (MGD)",
       x="Month") +
  scale_x_date(date_breaks = "1 month", date_labels = "%b")

Durham2015_plot
```



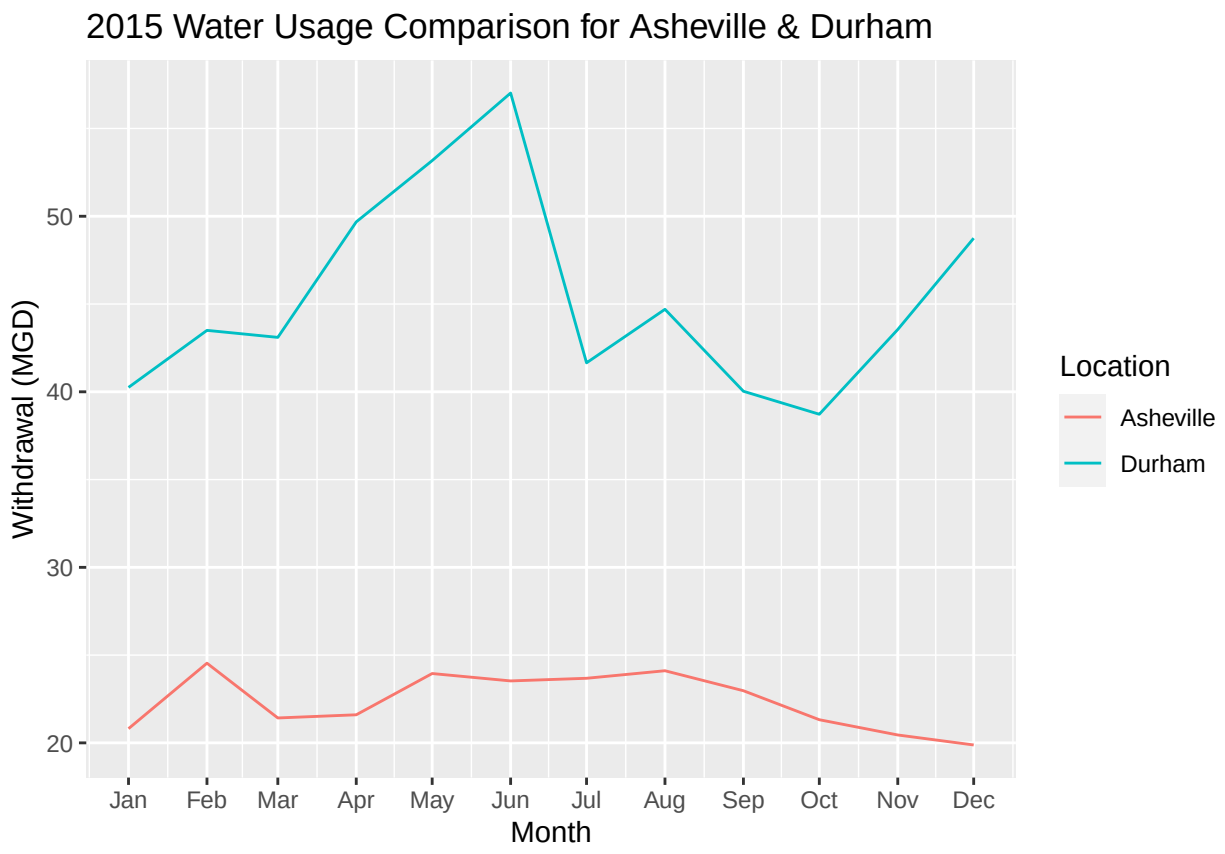
8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8
#Fetch data for Asheville 2015.
AshevillePWSID_2015 <- scrape.it('01-11-010', 2015)

#Combining the datasets for Asheville and Durham.
Combined_PSWID2015 <- full_join(DurhamPWSID_2015, AshevillePWSID_2015)
```

```
## Joining with `by = join_by(Month, Year, Withdrawals, WaterSystem, PWSID,
## Ownership, Date)`
```

```
#Plotting the data.
Combined2015_plot <- ggplot(Combined_PSWID2015,
                            aes(x=Date,
                                y=Withdrawals,
                                color = WaterSystem)) +
  geom_line() +
  labs(title = paste("2015 Water Usage Comparison for Asheville & Durham"),
       y="Withdrawal (MGD)",
       x="Month",
       color = "Location") +
  scale_x_date(date_breaks = "1 month", date_labels = "%b")
```

```
Combined2015_plot
```



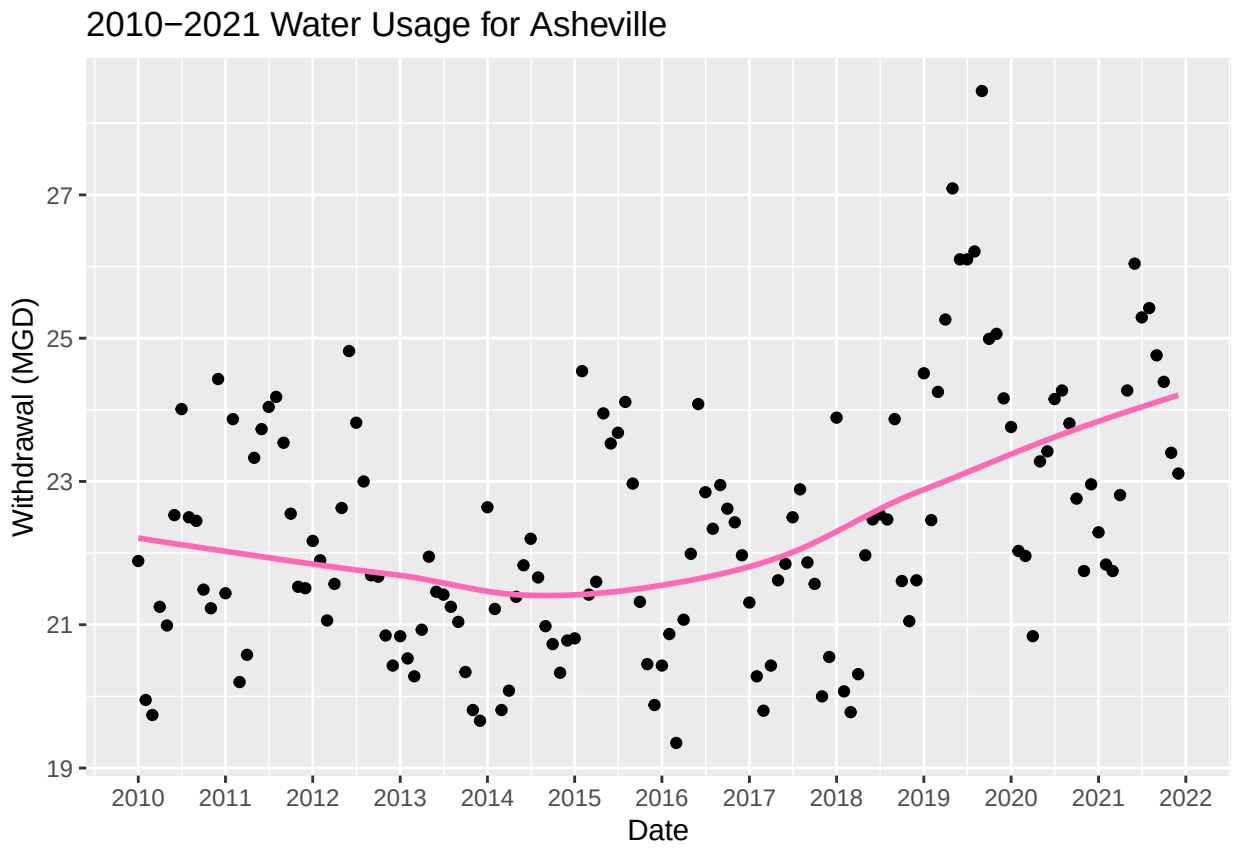
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   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

#Create a list of the years we want.
the_years <- 2010:2021
Asheville_site <- '01-11-010'

#"Map" the "scrape.it" function to retrieve data for all the years
#and combine the data.
df_Asheville_2010_2021 <- map2(Asheville_site, the_years, scrape.it) %>%
  bind_rows()

#Plotting the data.
AshevilleData_plot <- ggplot(df_Asheville_2010_2021,
                             aes(x=Date, y=Withdrawals)) +
  geom_point() +
```

```
  geom_smooth(method = 'loess', se=FALSE, color = "hotpink") +
  labs(title = paste("2010-2021 Water Usage for Asheville"),
       y="Withdrawal (MGD)",
       x="Date") +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")

AshevilleData_plot
```

```
## `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: Looking at the plot, it is apparent that water usage in Asheville is following an increasing trend from 2015 to 2021. There was a decrease in water use from 2010 to 2015, but since then, the trend consistly reflects increasing use.