Assignment 10: Data Scraping

Jake Whisler

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 Setup
library(tidyverse)
library(rvest)
library(lubridate)
library(here)

getwd()
```

```
## [1] "/home/guest/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 Establishing the scraping URL
LWSP_webpage <- read_html("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 Data scraping
SystemName <- 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()
```

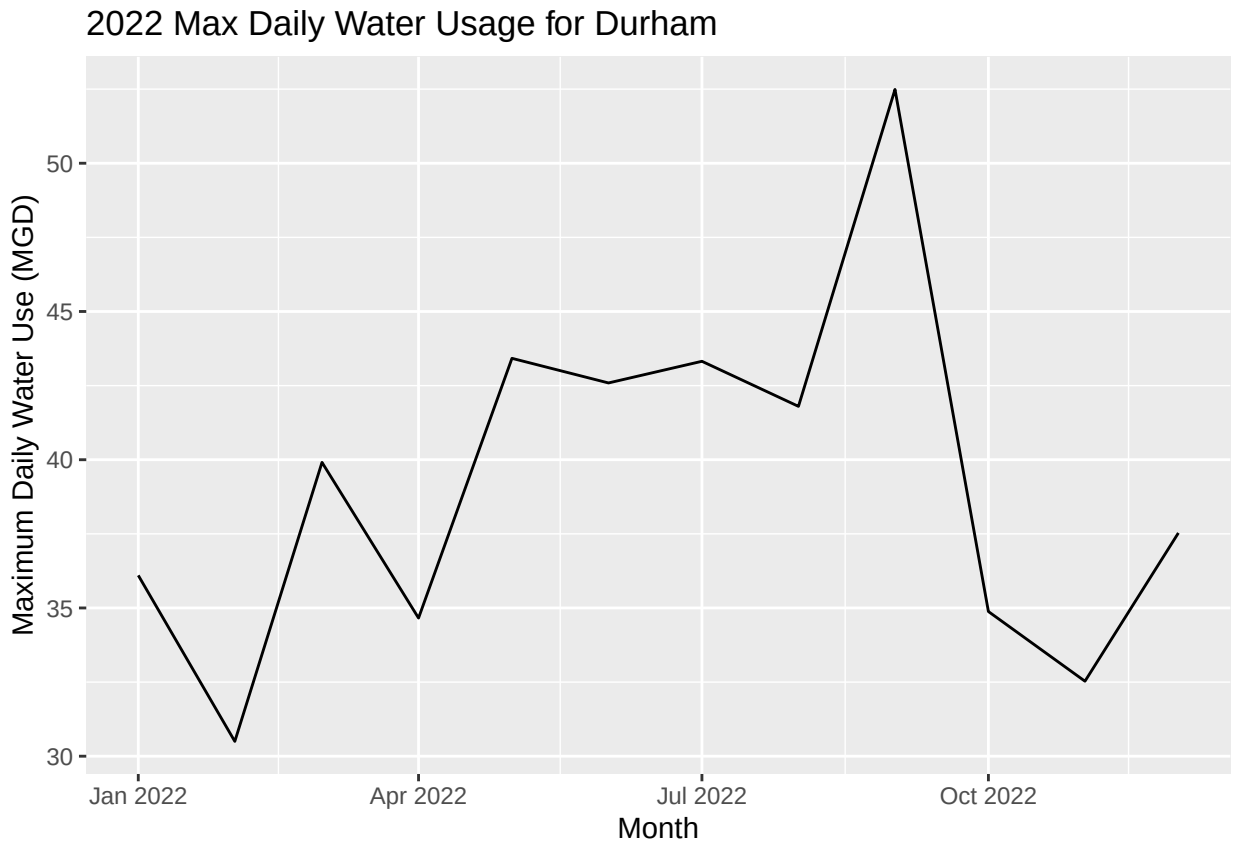
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   > 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

```r
#4 Creating dataframe
Durham_Dataframe <- data.frame("Month" = c("Jan", "May", "Sep",
                                           "Feb", "Jun", "Oct",
                                           "Mar", "Jul", "Nov",
                                           "Apr", "Aug", "Dec"),
                               "Year" = rep("2022"),
                               "DailyMaxUse_MGD" = as.numeric(MGD))
Durham_Dataframe <- Durham_Dataframe %>%
  mutate(SystemName = !!SystemName,
         Ownership = !!Ownership,
         Date = my(paste(Month,"-",Year)))

#5 Creating a line plot
Durham_lineplot <-ggplot(Durham_Dataframe, aes(x=Date, y=DailyMaxUse_MGD)) +
  geom_line() +
  labs(title = "2022 Max Daily Water Usage for Durham",
       x = "Month",
       y = "Maximum Daily Water Use (MGD)")
print(Durham_lineplot)
```

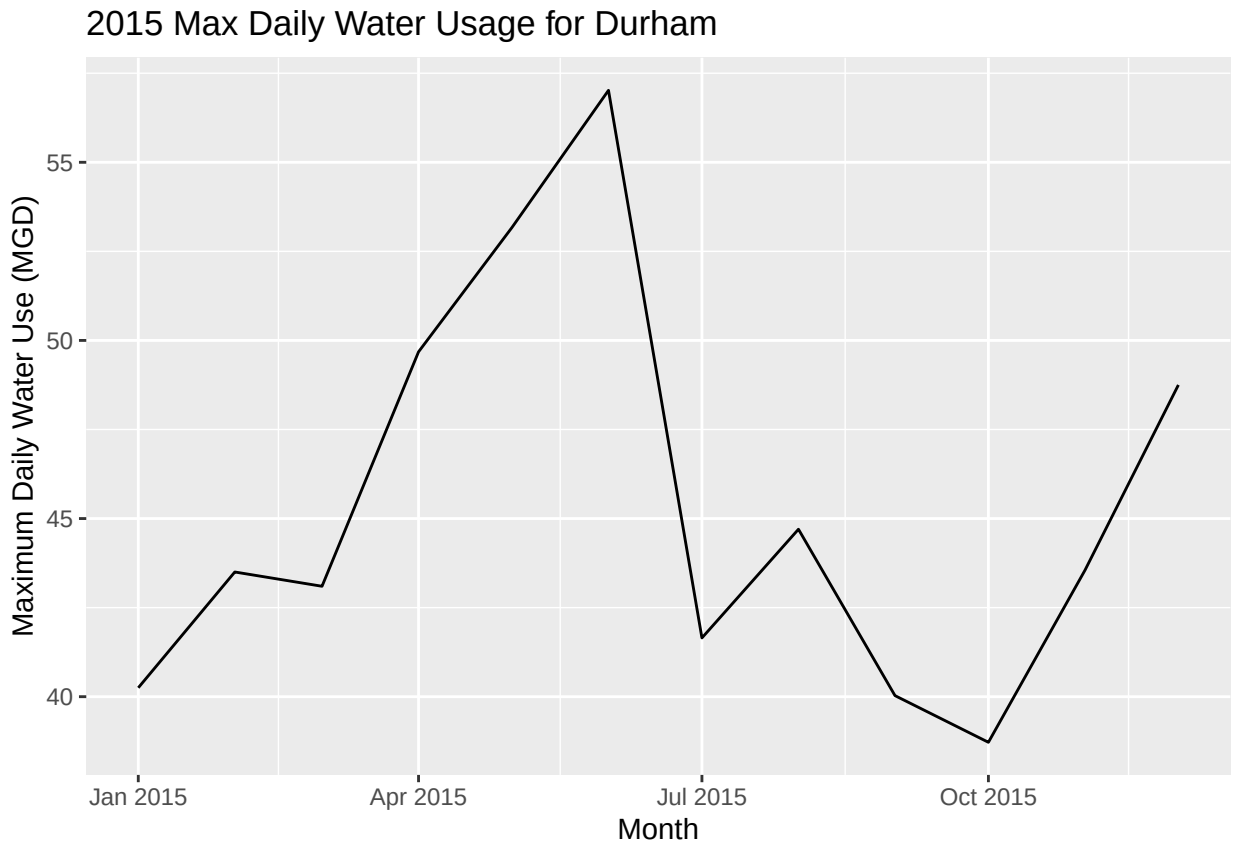


6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6. Constructing my function
scrape.data <- function(the_PWSID, the_year){
  the_website <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',the_PWSID,
                                  '&year=',the_year))
  the_systemname_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
  the_ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
  the_PWSID_tag <- 'td tr:nth-child(1) td:nth-child(5)'
  the_MGD_tag <- 'th~ td+ td'

  the_systemname <- the_website %>% html_nodes(the_systemname_tag) %>% html_text()
  the_ownership <- the_website %>%   html_nodes(the_ownership_tag) %>%  html_text()
  the_PWS_ID <- the_website %>% html_nodes(the_PWSID_tag) %>% html_text()
  the_MGD <- the_website %>% html_nodes(the_MGD_tag) %>% html_text()

  Durham_Function_Dataframe <- data.frame("Month" = c("Jan", "May", "Sep",
                                          "Feb", "Jun", "Oct",
                                          "Mar", "Jul", "Nov",
                                          "Apr", "Aug", "Dec"),
                                "Year" = rep(the_year),
                                "DailyMaxUse_MGD" = as.numeric(the_MGD))
  Durham_Function_Dataframe <- Durham_Function_Dataframe %>%
  mutate(SystemName = !!the_systemname,
         Ownership = !!the_ownership,
         Date = my(paste(Month,"-",Year)))
  return(Durham_Function_Dataframe)
}
```

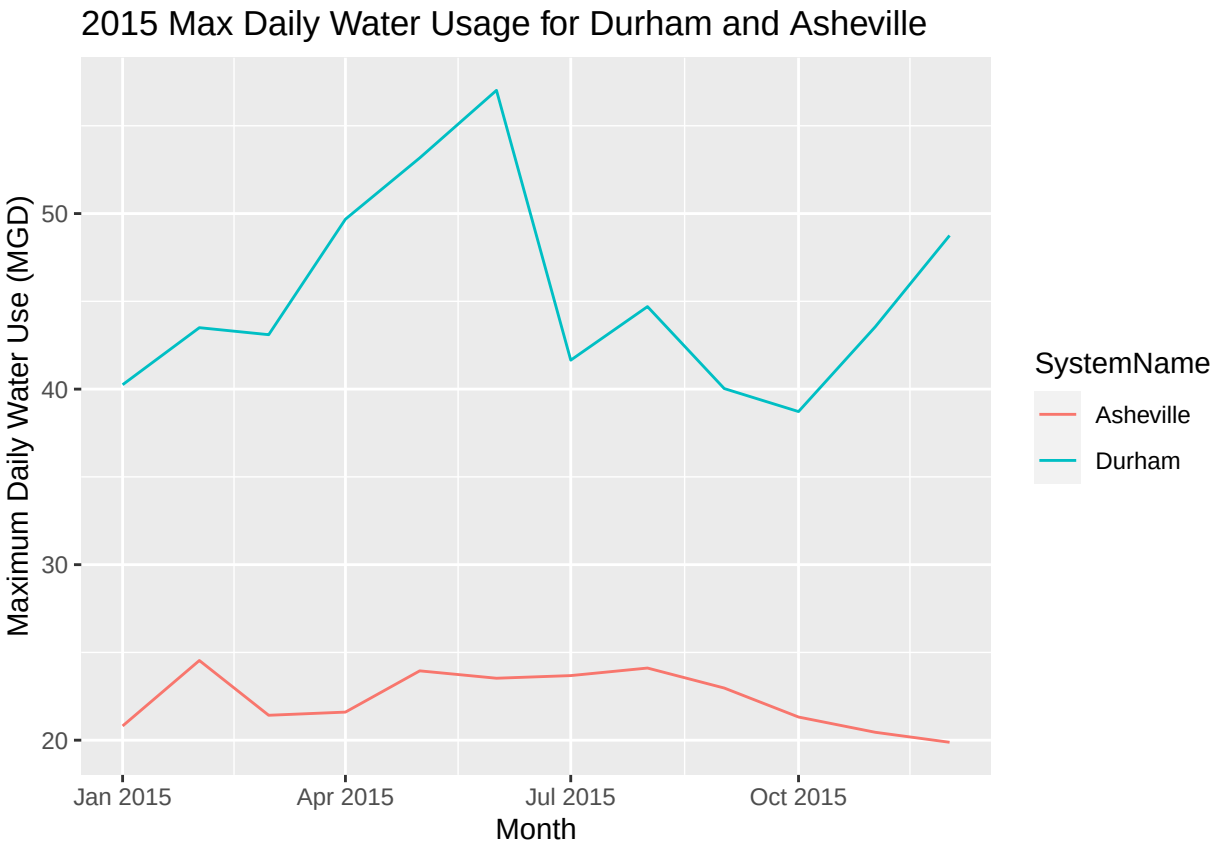
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

```
#7 Using my function to gather Durham 2015 data and plot it
Durham2015 <- scrape.data("03-32-010","2015")

Durham2015_lineplot <-ggplot(Durham2015, aes(x=Date, y=DailyMaxUse_MGD)) +
  geom_line() +
  labs(title = "2015 Max Daily Water Usage for Durham",
       x = "Month",
       y = "Maximum Daily Water Use (MGD)")
print(Durham2015_lineplot)
```



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 Asheville data scraping and merging
Asheville2015 <- scrape.data("01-11-010","2015")

AshevilleDurham <- rbind(Durham2015, Asheville2015)

AshevilleDurham_lineplot <-ggplot(AshevilleDurham, aes(x=Date, y=DailyMaxUse_MGD,
                                                       colour=SystemName)) +
  geom_line() +
  labs(title = "2015 Max Daily Water Usage for Durham and Asheville",
       x = "Month",
       y = "Maximum Daily Water Use (MGD)")
print(AshevilleDurham_lineplot)
```

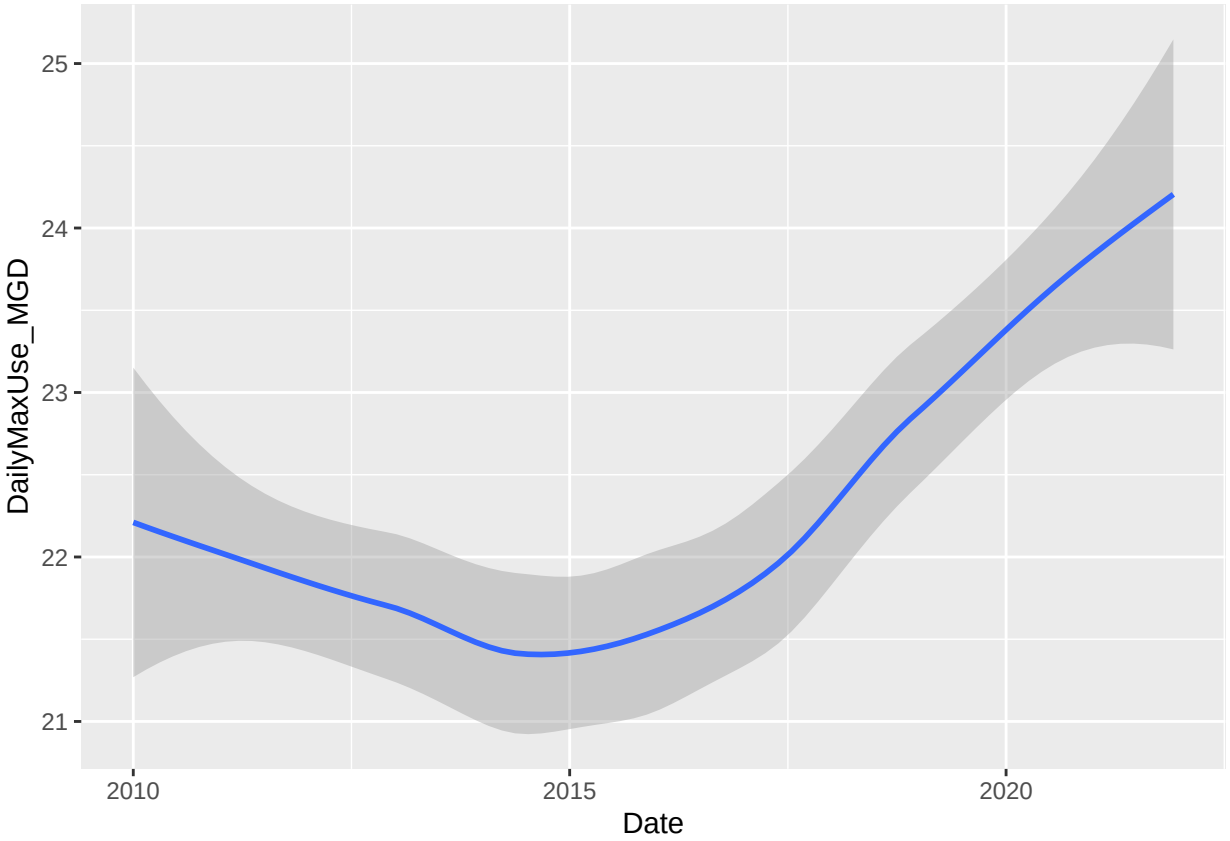


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021.Add a smoothed line to the plot (method = 'loess').

   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 Repeating the function for 2010 through 2021
the_ID <- rep("01-11-010", 12)
the_years = c(2010,2011,2012,2013,2014,2015,2016,2017,2018,2019,2020,2021)

Asheville_dfs <- map2(the_ID, the_years, scrape.data) %>%
  bind_rows()

AshevilleYears <- ggplot(Asheville_dfs, aes(x=Date, y=DailyMaxUse_MGD))+
  geom_smooth(method = "loess")

print(AshevilleYears)
```

```
## `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: > Yes, Asheville has increased steadily in water usage from 2014 to 2021, according to the plot above.