Assignment 10: Data Scraping

Asreeta Ushasri

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 Load packages

#install.packages('rvest')
#install.packages("dataRetrieval")
#install.packages("tidycensus")


library(tidyverse)
library(rvest)
library(lubridate)
library(viridis)
library(dataRetrieval)
library(tidycensus)
library(here)
library(cowplot)
library(purrr)

getwd()
```

```
## [1] "/home/guest/AU_EDE_Fall2024"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2023 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=2023

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

```
#2 Set the URL as the website for the data

URL <-
  read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2023')

URL
```

```
## {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)“.

```r
#3 Collect system information data

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

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

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

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

Water_System_Name
```

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

```r
PWSID
```

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

```r
Ownership
```

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

```r
MGD
```

```
##  [1] "28.9000" "33.3000" "43.7000" "30.0000" "40.0000" "37.2300" "34.2000"
##  [8] "44.9000" "40.3500" "30.9000" "56.7000" "33.3000"
```

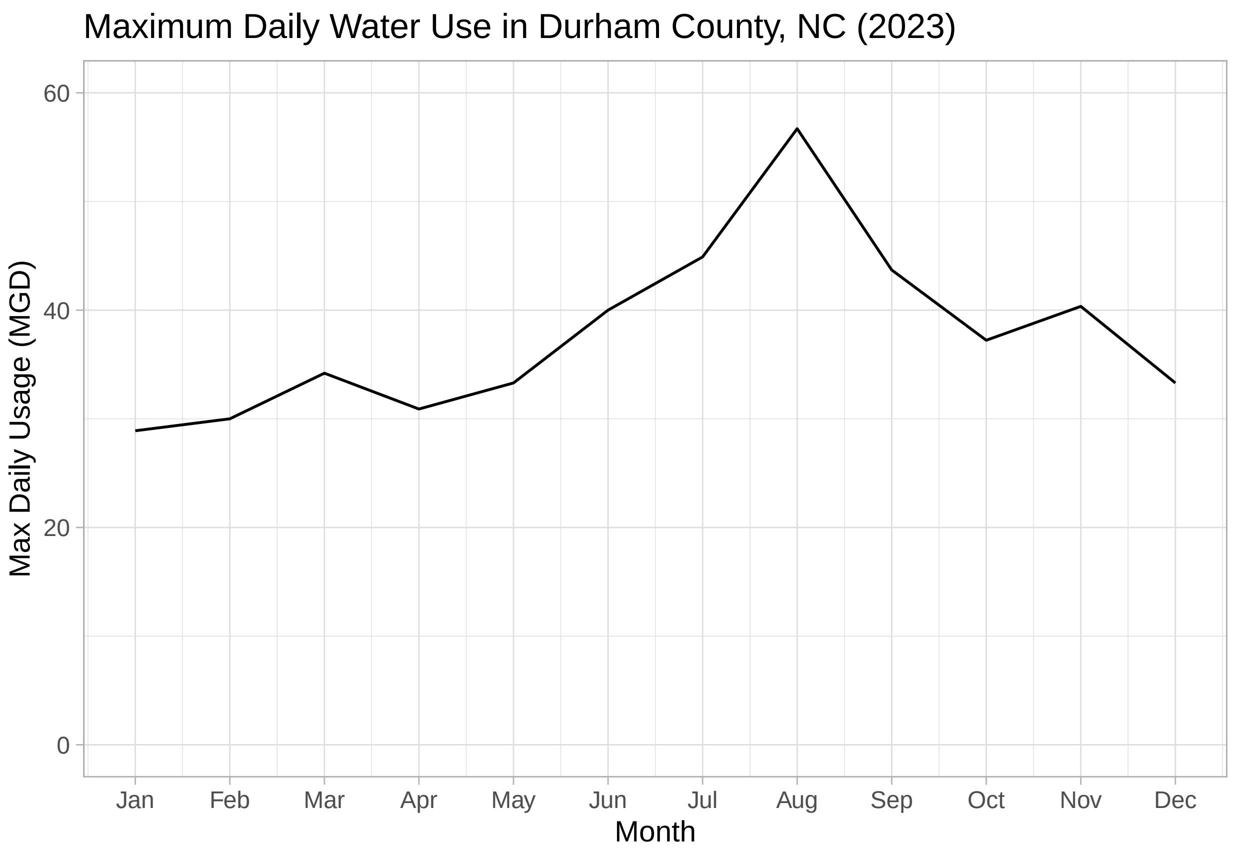
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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…

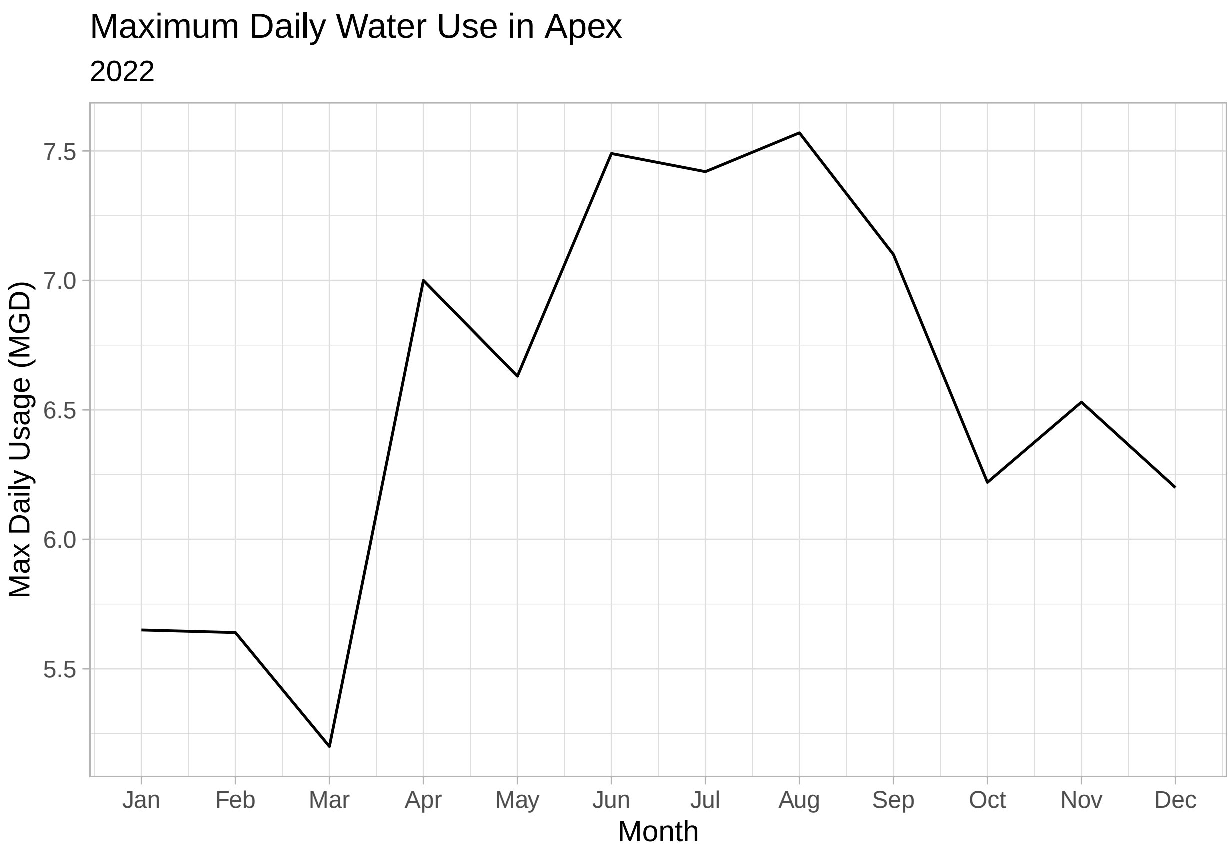
5. Create a line plot of the maximum daily withdrawals across the months for 2023, making sure, the months are presented in proper sequence.

```r
#4 Create a dataframe

Water_Months <- c("Jan", "May", "Sept", "Feb", "June", "Oct",
          "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")

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

Durham_Water_DF <- data.frame("Water_System" = rep(Water_System_Name),
                             "Max_Day_Use" = as.numeric(MGD),
                             "PWSID" = rep(PWSID),
                             "Ownership" = rep(Ownership),
                             "Month_Name" = Water_Months,
                             "Month_Date" = as.numeric(Date))

#view(Durham_Water_DF)
#class(Durham_Water_DF$Month_Date)
#class(Durham_Water_DF$Max_Day_Use)
#str(Durham_Water_DF)

#5 Line Plot

Water_Plot <- ggplot(Durham_Water_DF, aes(x = Month_Date, y = Max_Day_Use)) +
  geom_line() +
  labs(title = "Maximum Daily Water Use in Durham County, NC (2023)",
                     x = "Month",
                     y = "Max Daily Usage (MGD)") + theme_light() +
  scale_x_continuous(breaks = seq(1, 12, by = 1),
    labels = month.abb) +
  ylim(0, 60)

print(Water_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, returning a dataframe. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

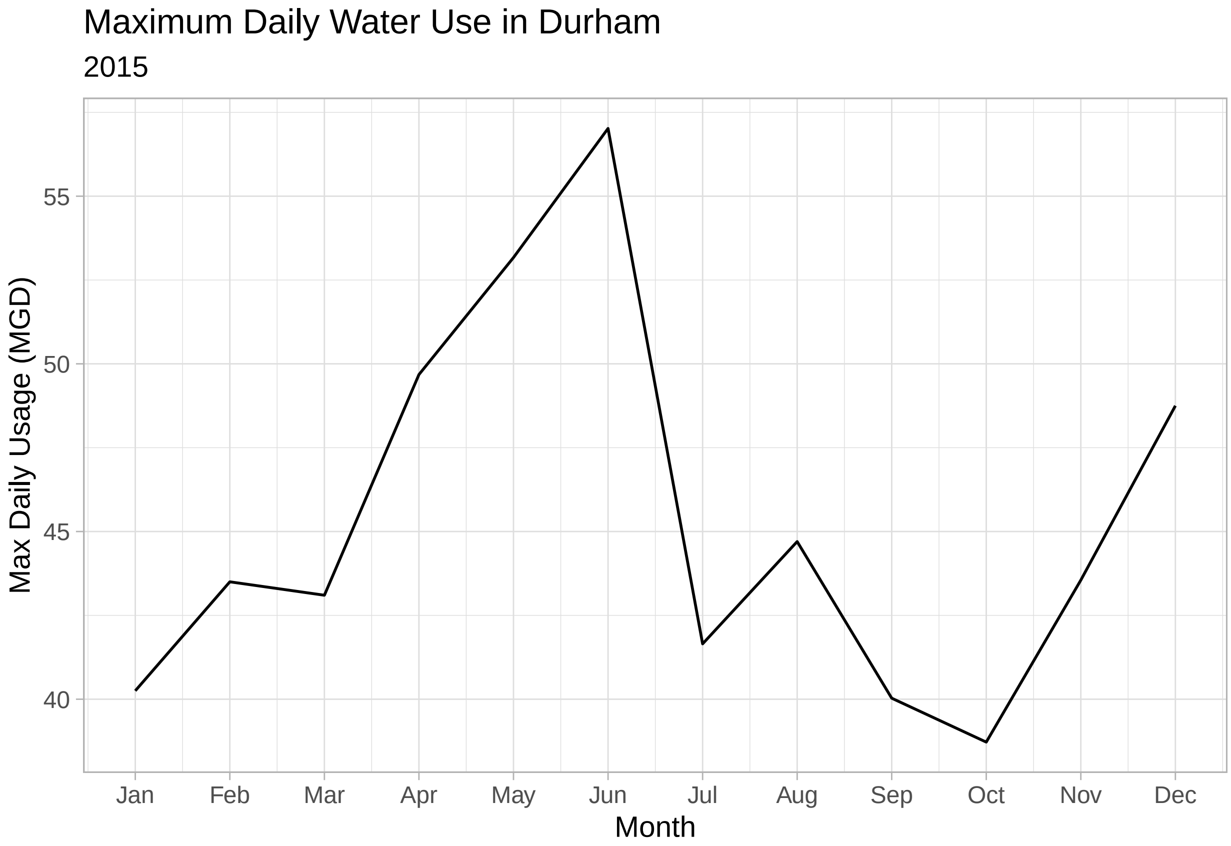
```
#6. First, start a function with two inputs

Water_Usage_Function <- function(Input_PWSID, Input_Year){

  #Next construct the scrape URL code

  Base_URL <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?'

  Scrape_URL <- paste0(Base_URL, 'pwsid=', Input_PWSID, '&year=', Input_Year)

  #print(Scrape_URL)

  #Assign the html nodes of the desired columns for the data frame

  Function_URL <- read_html(Scrape_URL)

  Function_Water_System_Name <- Function_URL %>%
    html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
    html_text()
  Function_PWSID <- Function_URL %>%
    html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
    html_text()

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

  Function_MGD <- Function_URL %>%
    html_nodes("th~ td+ td") %>%
    html_text()

  #Assign the month labels for to match the order scraped from website

  Water_Months <- c("Jan", "May", "Sept", "Feb", "June", "Oct",
            "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")

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

  #Set up data frame with desired columns

  Function_WaterUsage_DF <- data.frame(
    "Water_System" = rep(Function_Water_System_Name),
    "Max_Day_Use" = as.numeric(Function_MGD),
    "PWSID" = rep(Function_PWSID),
    "Ownership" = rep(Function_Ownership),
    "Month_Name" = Water_Months,
    "Month_Date" = as.numeric(Date))

  #Assign a plot title based on the variable water system name

  Plot_Title <- paste("Maximum Daily Water Use in", Function_Water_System_Name)

  #Plot the findings from the function

  Function_Plot <- ggplot(Function_WaterUsage_DF,
                          aes(x = Month_Date, y = Max_Day_Use)) +
    geom_line() + labs(title = Plot_Title,
                       subtitle = Input_Year,
                       x = "Month",
                       y = "Max Daily Usage (MGD)") + theme_light() +
    scale_x_continuous(breaks = seq(1, 12, by = 1),
      labels = month.abb)

  print(Function_Plot) }

#Test function with PWSID as Apex and Year as 2022

Water_Usage_Function("03-92-045", 2022)
```



```
#Create a separate function which only produces data frame
#This will be used for Part 8 and Part 9
#This function will be helpful for custom graphs

Water_Usage_DF_Function <- function(Input_PWSID, Input_Year){

  Base_URL <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?'

  Scrape_URL <- paste0(Base_URL, 'pwsid=', Input_PWSID, '&year=', Input_Year)

  #print(Scrape_URL)

  Function_URL <- read_html(Scrape_URL)

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

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

  Function_Ownership <- Function_URL %>%
    html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
    html_text()
  Function_MGD <- Function_URL %>%
    html_nodes("th~ td+ td") %>%
    html_text()

  Water_Months <- c("Jan", "May", "Sept", "Feb", "June", "Oct",
            "Mar", "Jul", "Nov", "Apr", "Aug", "Dec")

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

  Function_WaterUsage_DF <- data.frame(
    "Water_System" = rep(Function_Water_System_Name),
    "Max_Day_Use" = as.numeric(Function_MGD),
    "PWSID" = rep(Function_PWSID),
    "Ownership" = rep(Function_Ownership),
    "Month_Name" = Water_Months,
    "Month_Date" = as.numeric(Date)) }
```

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

```
#7 Run the function using Durham 2015 data

Water_Usage_Function("03-32-010", 2015)
```

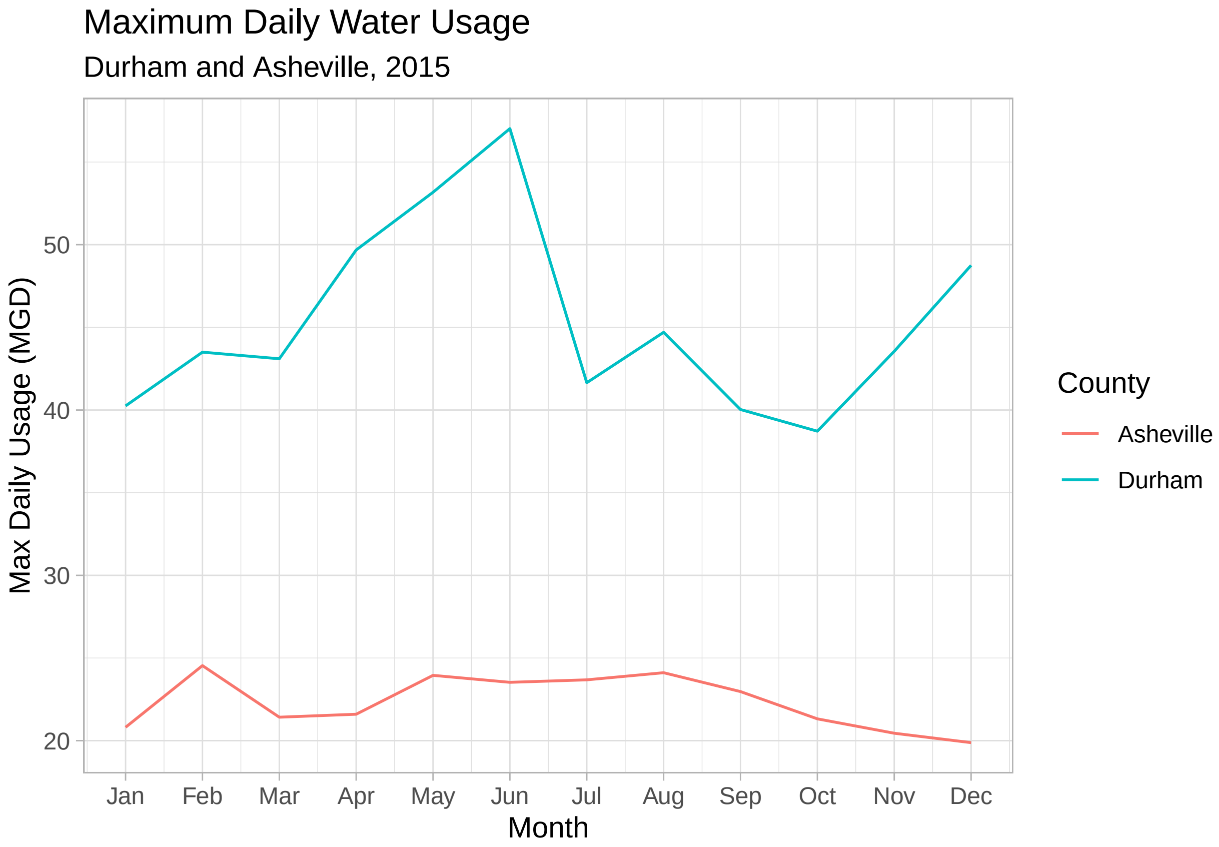


8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8 Extract Durham and Asheville Data from 2015
#Use the Water_Usage_DF_Function, with no automatic plots
#This will allow assignment of data frames from two counties
#From the two data frames, manually plot both data sets on the same graph

Durham_Data <- as.data.frame(Water_Usage_DF_Function("03-32-010", 2015))
Asheville_Data <- as.data.frame(Water_Usage_DF_Function("01-11-010", 2015))

#Combine the data sets into one data frame

County_Water_Use <- rbind(Durham_Data, Asheville_Data)

#view(County_Water_Use)

#Plot the two counties on the same graph with ggplot

class(County_Water_Use$Water_System)
```

```
## [1] "character"
```

```
County_MDG_Graph <- ggplot(data = County_Water_Use,
            aes(x = Month_Date, y = Max_Day_Use,
       color = Water_System)) +
  geom_line() + theme_light() +
  labs(title = "Maximum Daily Water Usage",
                      subtitle = "Durham and Asheville, 2015",
                      x = "Month",
                      y = "Max Daily Usage (MGD)",
       color = "County") +
    scale_x_continuous(breaks = seq(1, 12, by = 1), labels = month.abb)

County_MDG_Graph
```

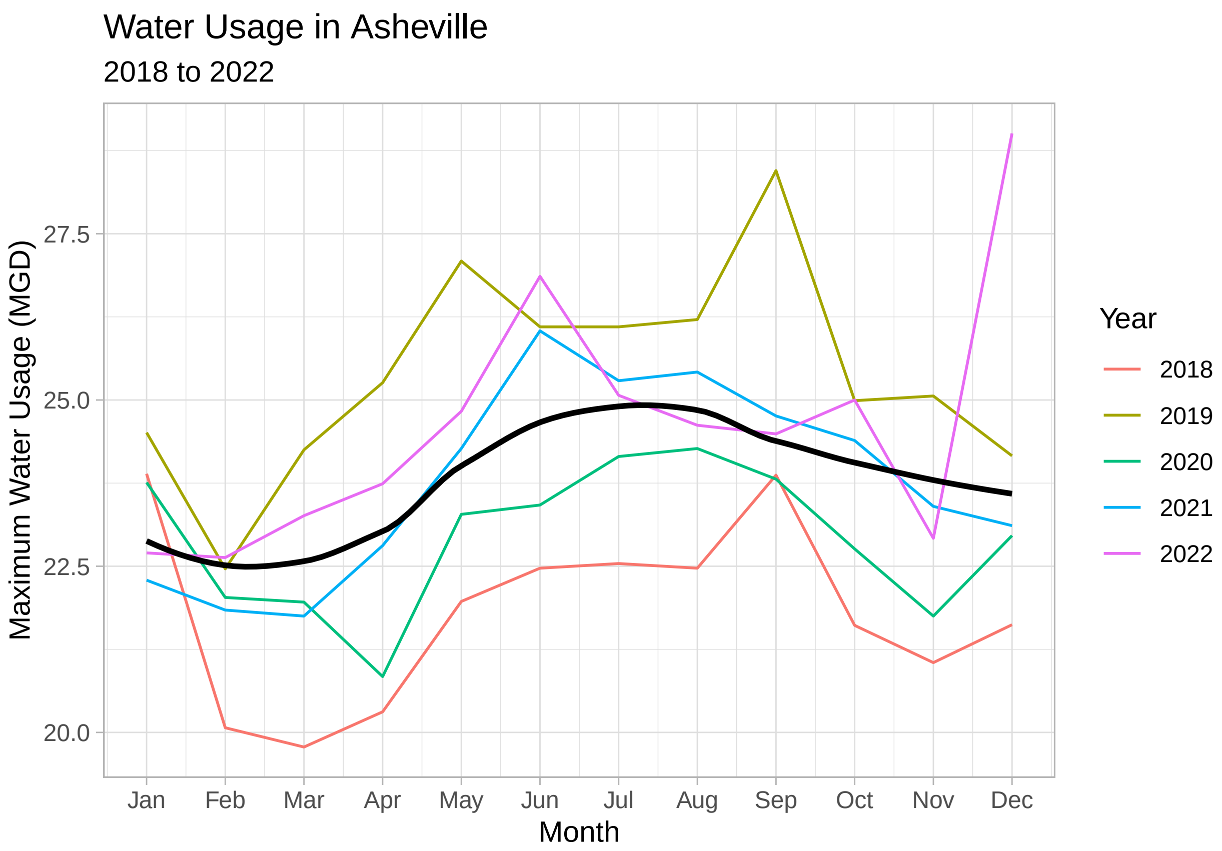


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2022.Add a smoothed line to the plot (method = 'loess').

   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 multiyear graph of Asheville water usage

Asheville_Years <- c(2018,2019,2020,2021,2022)

Asheville_PWSID <- rep.int("01-11-010",length(Asheville_Years))

Asheville_Multiyear_DF <- map2(Asheville_PWSID, Asheville_Years,
       Water_Usage_DF_Function) %>%
  map2(Asheville_Years, ., ~ mutate(.y, Year = .x)) %>%
  bind_rows()

#view(Asheville_Multiyear_DF)

#Plot Multiple Years
ggplot(Asheville_Multiyear_DF) +
  geom_line(aes(x=Month_Date, y=Max_Day_Use, color=as.character(Year))) +
    geom_smooth(aes(x = Month_Date, y = Max_Day_Use),
                method = "loess", se = FALSE, color = "black") +
  labs(title = "Water Usage in Asheville",
       subtitle = "2018 to 2022",
       y="Maximum Water Usage (MGD)",
       x="Month",
       color = "Year") +
  theme_light() +
    scale_x_continuous(breaks = seq(1, 12, by = 1),
      labels = month.abb)
```

```
## `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 has variability between years regarding the water usage. For example, Asheville had a lower water usage in 2018 than in 2019. The water usage in September 2018 was under 25 MGD, whereas the water usage in September 2019 was above 27.5. The water usage in 2020 and 2021 dropped from the 2019 levels.
>
> This data showcases a seasonal trend over time. Water usage in Asheville peaks around the warmer months of August/September. In contrast, the water usage drops in March/April and November. 2022 differs from this trend slightly. Although this year follows a similar general seasonality trend, peak water usage in 2022 occured in May/June, with another sudden peak in December. Overall, Asheville displays a seasonal trend in water usage over time.