Assignment 10: Data Scraping

Tori Newton

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

#Check working directory
getwd()
```

```
## [1] "/home/guest/EDA_Spring2025"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2024 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=2024

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

```
#2:Fetch the web resources from the URL
webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2024')
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: Scraping the data
water_system_name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water_system_name
```

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

```
pwsid <- webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
pwsid
```

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

```
ownership <- webpage %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
ownership
```

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

```
maximum_day_use <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()
maximum_day_use
```

```
##  [1] "34.5000" "36.0600" "37.3300" "32.1000" "46.6500" "37.3600" "38.2000"
##  [8] "41.9000" "36.5800" "36.7300" "42.9600" "34.4500"
```

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...

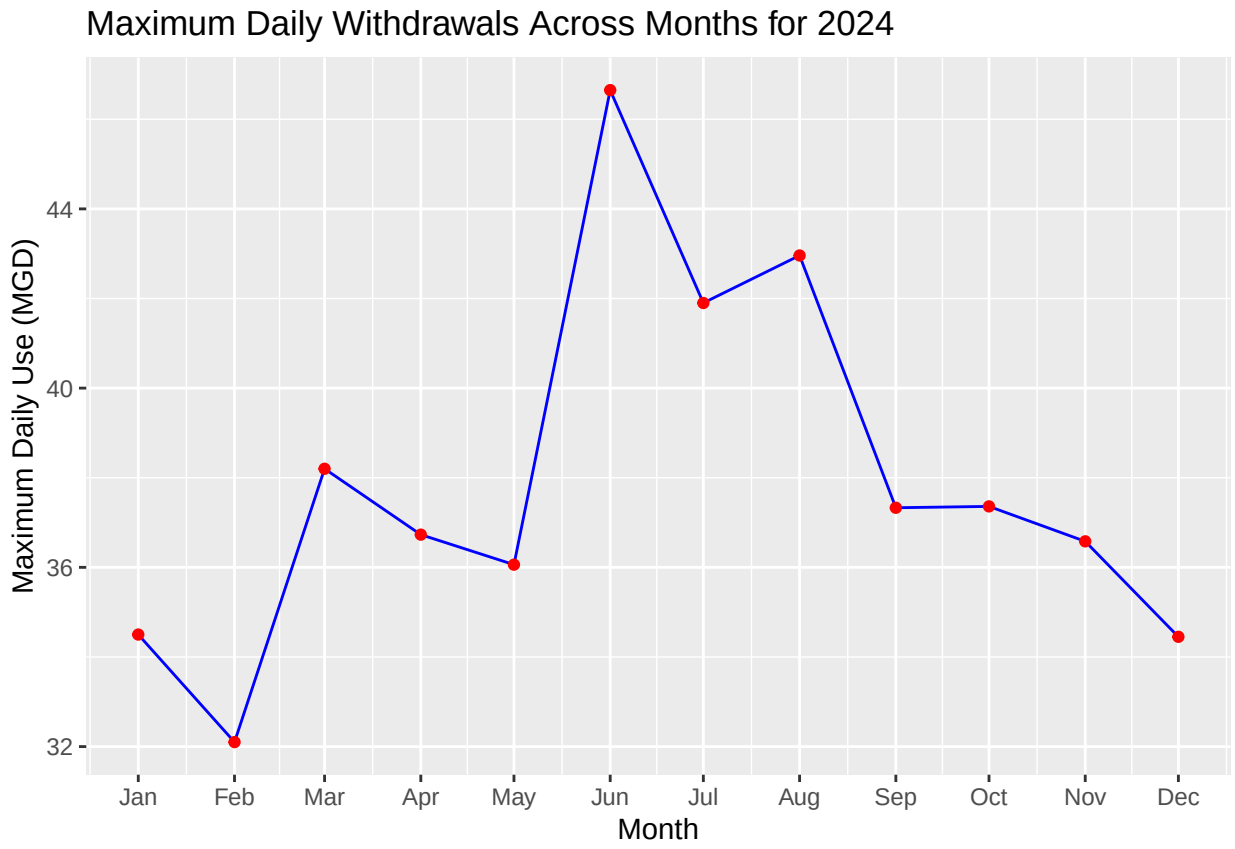
5. Create a line plot of the maximum daily withdrawals across the months for 2024, making sure, the months are presented in proper sequence.

```
#4: Convert scraped data into a dataframe
months <- c("Jan", "May", "Sept", "Feb", "June", "Oct", "Mar", "July", "Nov", "April", "Aug", "Dec")

df <- data.frame(
  "Month" = months,
  "Year" = rep(2024,12),
  "Max_Day_Use" = as.numeric(maximum_day_use)) %>%
    mutate(
    Water_System_Name = rep(water_system_name, 12),
    PWSID = rep(pwsid, 12),
    Ownership = rep(ownership, 12),
    Date = my(paste(Month, "-", Year)))
print(df)
```

```
##    Month Year Max_Day_Use Water_System_Name     PWSID    Ownership       Date
## 1    Jan 2024       34.50            Durham 03-32-010 Municipality 2024-01-01
## 2    May 2024       36.06            Durham 03-32-010 Municipality 2024-05-01
## 3   Sept 2024       37.33            Durham 03-32-010 Municipality 2024-09-01
## 4    Feb 2024       32.10            Durham 03-32-010 Municipality 2024-02-01
## 5   June 2024       46.65            Durham 03-32-010 Municipality 2024-06-01
## 6    Oct 2024       37.36            Durham 03-32-010 Municipality 2024-10-01
## 7    Mar 2024       38.20            Durham 03-32-010 Municipality 2024-03-01
## 8   July 2024       41.90            Durham 03-32-010 Municipality 2024-07-01
## 9    Nov 2024       36.58            Durham 03-32-010 Municipality 2024-11-01
## 10 April 2024       36.73            Durham 03-32-010 Municipality 2024-04-01
## 11   Aug 2024       42.96            Durham 03-32-010 Municipality 2024-08-01
## 12   Dec 2024       34.45            Durham 03-32-010 Municipality 2024-12-01
```

```r
#5: Line plot
df %>%
  ggplot(aes(x = Date, y = Max_Day_Use)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
  scale_x_date(
    breaks = seq(from = as.Date("2024-01-01"), to = as.Date("2024-12-01"), by = "month"),
    labels = format(seq(from = as.Date("2024-01-01"), to = as.Date("2024-12-01"), by = "month"), "%b")
  ) +
  labs(
    title = "Maximum Daily Withdrawals Across Months for 2024",
    x = "Month",
    y = "Maximum Daily Use (MGD)")
```



6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function with two input - "PWSID" and "year" - that:

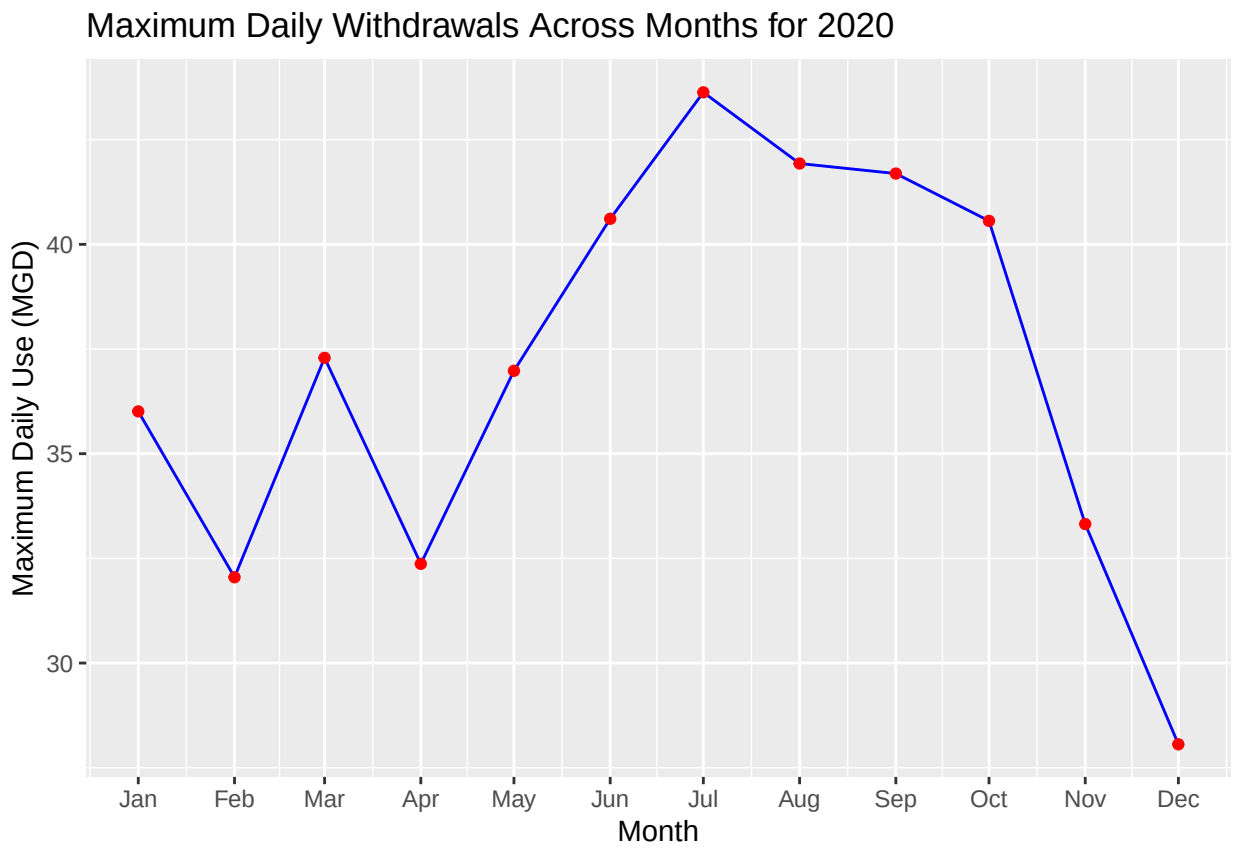
- Creates a URL pointing to the LWSP for that PWSID for the given year
- Creates a website object and scrapes the data from that object (just as you did above)
- Constructs a dataframe from the scraped data, mostly as you did above, but includes the PWSID and year provided as function inputs in the dataframe.
- Returns the dataframe as the function's output

```r
#6: Create scraping function
scrape_data <- function(PWSID, the_year) {
#Retrieve the website contents
  the_website <-
    read_html(paste0
              ('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=', PWSID, "&year=", the_year))
  #Scrape the data items
  water_system_name <- the_website %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
pwsid <- the_website %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
ownership <- the_website %>%
  html_nodes("div+ table tr:nth-child(2) td:nth-child(4)") %>%
  html_text()
maximum_day_use <- the_website %>%
  html_nodes("th~ td+ td") %>%
  html_text()
#Convert to a dataframe
df_new <- data.frame(
  "Month" = months,
  "Year" = rep(the_year,12),
  "Max_Day_Use" = as.numeric(maximum_day_use)) %>%
    mutate(Water_System_Name = rep(water_system_name, 12),
    PWSID = rep(pwsid, 12),
    Ownership = rep(ownership, 12),
    Date = ymd(paste(Year, Month, "01", sep = "-")))
#Return the dataframe
 return(df_new)
}
```

7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2020

```r
#7: Scrape data for 2020
df_2020 <- scrape_data(PWSID= "03-32-010", the_year=2020)

#Line plot
df_2020 %>%
  ggplot(aes(x = Date, y = Max_Day_Use)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
  scale_x_date(
    breaks = seq(from = as.Date("2020-01-01"), to = as.Date("2020-12-01"), by = "month"),
    labels = format(seq(from = as.Date("2020-01-01"), to = as.Date("2020-12-01"), by = "month"), "%b")
  ) +
  labs(
    title = "Maximum Daily Withdrawals Across Months for 2020",
    x = "Month",
    y = "Maximum Daily Use (MGD)")
```



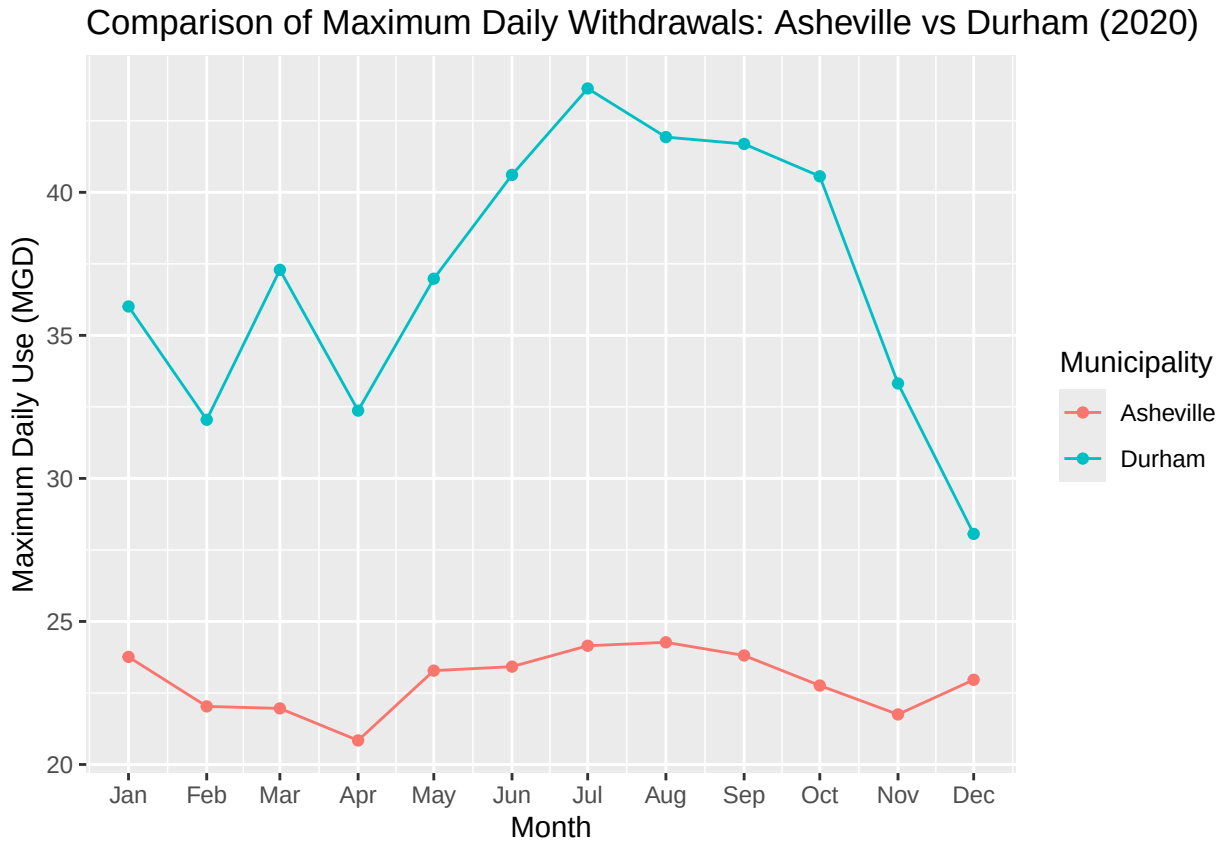
8. Use the function above to extract data for Asheville (PWSID = '01-11-010') in 2020. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8: Scrape data for Asheville in 2020
df_asheville_2020 <- scrape_data(PWSID = "01-11-010", the_year = 2020)

#Combine data for Durham and Asheville
df_combined <- bind_rows(
  df_2020 %>% mutate(Municipality = "Durham"),
  df_asheville_2020 %>% mutate(Municipality = "Asheville")
)

#Plot comparing Asheville and Durham's water withdrawals
df_combined %>%
  ggplot(aes(x = Date, y = Max_Day_Use, color = Municipality)) +
  geom_line() +
  geom_point() +
  scale_x_date(
    breaks = seq(from = as.Date("2020-01-01"), to = as.Date("2020-12-01"), by = "month"),
    labels = format(seq(from = as.Date("2020-01-01"), to = as.Date("2020-12-01"), by = "month"), "%b")
  ) +
  labs(
    title = "Comparison of Maximum Daily Withdrawals: Asheville vs Durham (2020)",
    x = "Month",
```

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



9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2023.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, and use that to construct your plot.

```
#9: Asheville 2018-2023
Asheville_PWSID <- "01-11-010"
the_years <- 2018:2023

asheville_list <- rep(Asheville_PWSID, length(the_years))

df_asheville_2018_2023 <- map2(asheville_list, the_years, scrape_data) %>%
  bind_rows()
print(df_asheville_2018_2023)
```

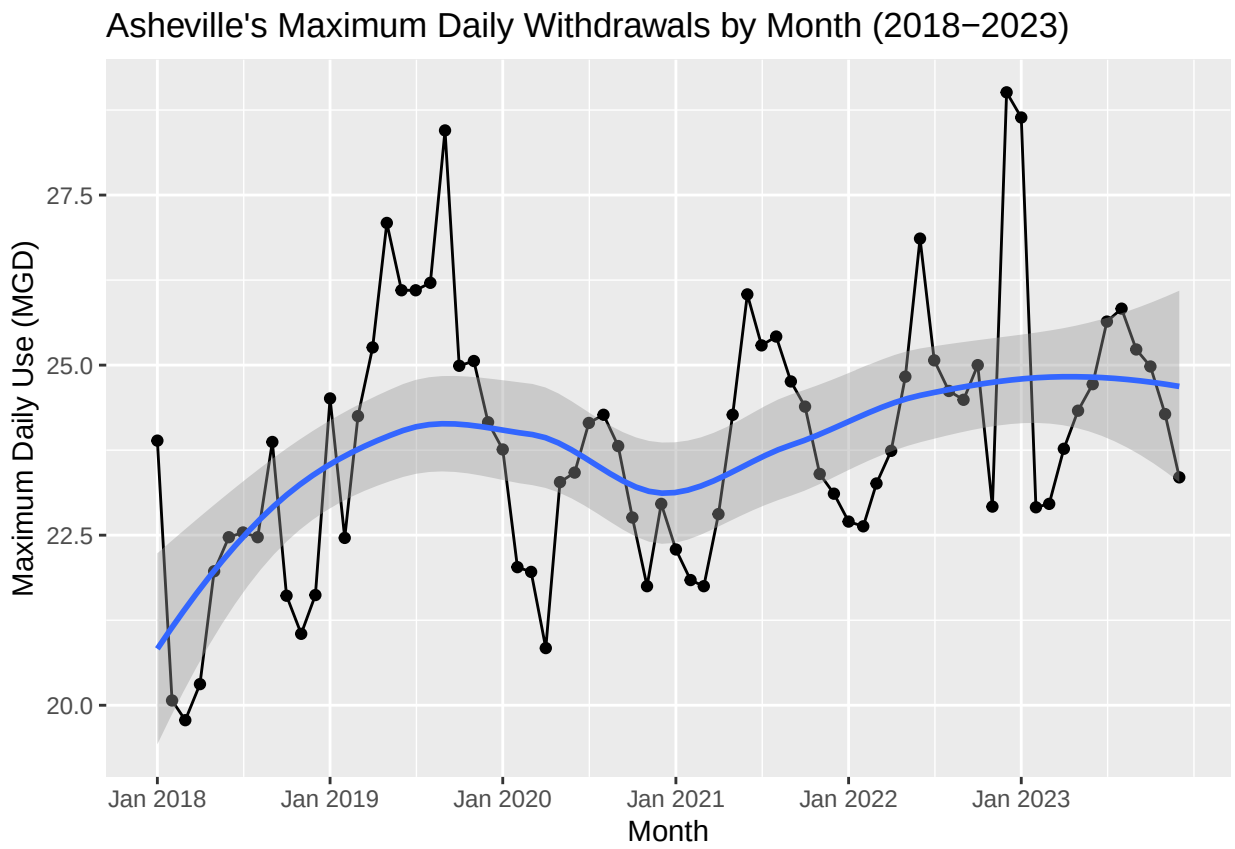
```
##    Month Year Max_Day_Use Water_System_Name     PWSID    Ownership       Date
## 1    Jan 2018       23.89         Asheville 01-11-010 Municipality 2018-01-01
## 2    May 2018       21.97         Asheville 01-11-010 Municipality 2018-05-01
## 3   Sept 2018       23.87         Asheville 01-11-010 Municipality 2018-09-01
```

```
## 4    Feb 2018       20.07        Asheville 01-11-010 Municipality 2018-02-01
## 5   June 2018       22.47        Asheville 01-11-010 Municipality 2018-06-01
## 6    Oct 2018       21.61        Asheville 01-11-010 Municipality 2018-10-01
## 7    Mar 2018       19.78        Asheville 01-11-010 Municipality 2018-03-01
## 8   July 2018       22.54        Asheville 01-11-010 Municipality 2018-07-01
## 9    Nov 2018       21.05        Asheville 01-11-010 Municipality 2018-11-01
## 10 April 2018       20.31        Asheville 01-11-010 Municipality 2018-04-01
## 11   Aug 2018       22.47        Asheville 01-11-010 Municipality 2018-08-01
## 12   Dec 2018       21.62        Asheville 01-11-010 Municipality 2018-12-01
## 13   Jan 2019       24.51        Asheville 01-11-010 Municipality 2019-01-01
## 14   May 2019       27.09        Asheville 01-11-010 Municipality 2019-05-01
## 15  Sept 2019       28.45        Asheville 01-11-010 Municipality 2019-09-01
## 16   Feb 2019       22.46        Asheville 01-11-010 Municipality 2019-02-01
## 17  June 2019       26.10        Asheville 01-11-010 Municipality 2019-06-01
## 18   Oct 2019       24.99        Asheville 01-11-010 Municipality 2019-10-01
## 19   Mar 2019       24.25        Asheville 01-11-010 Municipality 2019-03-01
## 20  July 2019       26.10        Asheville 01-11-010 Municipality 2019-07-01
## 21   Nov 2019       25.06        Asheville 01-11-010 Municipality 2019-11-01
## 22 April 2019       25.26        Asheville 01-11-010 Municipality 2019-04-01
## 23   Aug 2019       26.21        Asheville 01-11-010 Municipality 2019-08-01
## 24   Dec 2019       24.16        Asheville 01-11-010 Municipality 2019-12-01
## 25   Jan 2020       23.76        Asheville 01-11-010 Municipality 2020-01-01
## 26   May 2020       23.28        Asheville 01-11-010 Municipality 2020-05-01
## 27  Sept 2020       23.81        Asheville 01-11-010 Municipality 2020-09-01
## 28   Feb 2020       22.03        Asheville 01-11-010 Municipality 2020-02-01
## 29  June 2020       23.42        Asheville 01-11-010 Municipality 2020-06-01
## 30   Oct 2020       22.76        Asheville 01-11-010 Municipality 2020-10-01
## 31   Mar 2020       21.96        Asheville 01-11-010 Municipality 2020-03-01
## 32  July 2020       24.15        Asheville 01-11-010 Municipality 2020-07-01
## 33   Nov 2020       21.75        Asheville 01-11-010 Municipality 2020-11-01
## 34 April 2020       20.84        Asheville 01-11-010 Municipality 2020-04-01
## 35   Aug 2020       24.27        Asheville 01-11-010 Municipality 2020-08-01
## 36   Dec 2020       22.96        Asheville 01-11-010 Municipality 2020-12-01
## 37   Jan 2021       22.29        Asheville 01-11-010 Municipality 2021-01-01
## 38   May 2021       24.27        Asheville 01-11-010 Municipality 2021-05-01
## 39  Sept 2021       24.76        Asheville 01-11-010 Municipality 2021-09-01
## 40   Feb 2021       21.84        Asheville 01-11-010 Municipality 2021-02-01
## 41  June 2021       26.04        Asheville 01-11-010 Municipality 2021-06-01
## 42   Oct 2021       24.39        Asheville 01-11-010 Municipality 2021-10-01
## 43   Mar 2021       21.75        Asheville 01-11-010 Municipality 2021-03-01
## 44  July 2021       25.29        Asheville 01-11-010 Municipality 2021-07-01
## 45   Nov 2021       23.40        Asheville 01-11-010 Municipality 2021-11-01
## 46 April 2021       22.81        Asheville 01-11-010 Municipality 2021-04-01
## 47   Aug 2021       25.42        Asheville 01-11-010 Municipality 2021-08-01
## 48   Dec 2021       23.11        Asheville 01-11-010 Municipality 2021-12-01
## 49   Jan 2022       22.70        Asheville 01-11-010 Municipality 2022-01-01
## 50   May 2022       24.83        Asheville 01-11-010 Municipality 2022-05-01
## 51  Sept 2022       24.49        Asheville 01-11-010 Municipality 2022-09-01
## 52   Feb 2022       22.63        Asheville 01-11-010 Municipality 2022-02-01
## 53  June 2022       26.86        Asheville 01-11-010 Municipality 2022-06-01
## 54   Oct 2022       25.00        Asheville 01-11-010 Municipality 2022-10-01
## 55   Mar 2022       23.26        Asheville 01-11-010 Municipality 2022-03-01
## 56  July 2022       25.07        Asheville 01-11-010 Municipality 2022-07-01
## 57   Nov 2022       22.92        Asheville 01-11-010 Municipality 2022-11-01
```

```
## 58 April 2022       23.74         Asheville 01-11-010 Municipality 2022-04-01
## 59   Aug 2022       24.62         Asheville 01-11-010 Municipality 2022-08-01
## 60   Dec 2022       29.01         Asheville 01-11-010 Municipality 2022-12-01
## 61   Jan 2023       28.64         Asheville 01-11-010 Municipality 2023-01-01
## 62   May 2023       24.33         Asheville 01-11-010 Municipality 2023-05-01
## 63  Sept 2023       25.23         Asheville 01-11-010 Municipality 2023-09-01
## 64   Feb 2023       22.91         Asheville 01-11-010 Municipality 2023-02-01
## 65  June 2023       24.72         Asheville 01-11-010 Municipality 2023-06-01
## 66   Oct 2023       24.98         Asheville 01-11-010 Municipality 2023-10-01
## 67   Mar 2023       22.96         Asheville 01-11-010 Municipality 2023-03-01
## 68  July 2023       25.64         Asheville 01-11-010 Municipality 2023-07-01
## 69   Nov 2023       24.28         Asheville 01-11-010 Municipality 2023-11-01
## 70 April 2023       23.77         Asheville 01-11-010 Municipality 2023-04-01
## 71   Aug 2023       25.83         Asheville 01-11-010 Municipality 2023-08-01
## 72   Dec 2023       23.35         Asheville 01-11-010 Municipality 2023-12-01
```

```r
#Plot of Asheville's max daily withdrawal for 2018-2023
df_asheville_2018_2023 %>%
  ggplot(aes(x=Date, y=Max_Day_Use)) +
  geom_line() +
  geom_point() +
  geom_smooth(method = "loess") +
  scale_x_date(
    breaks = seq(from = as.Date("2018-01-01"), to = as.Date("2023-12-01"), by = "year"),
    labels = format(seq(from = as.Date("2018-01-01"), to = as.Date("2023-12-01"),
                        by = "year"), "%b %Y")
  ) +
  labs(
    title = "Asheville's Maximum Daily Withdrawals by Month (2018-2023)",
    x = "Month",
    y = "Maximum Daily Use (MGD)",
    color = "Year"
  )
```

```
## `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: From just looking at the plot, it appears that overall Asheville has increased water usage over time. Although, the trend does fluctuate up and down with a dip in 2021 but overall the water usage in 2023 is much higher than in 2018. >