Assignment 10: Data Scraping

Ana Andino

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)
```

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.)

```r
#2
url <-
  'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2023'

webpage<- read_html(url)

print(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)".

```r
#3
#System information#
water_system_name <- webpage %>%
  html_nodes('table:nth-child(7) 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('table:nth-child(7) tr:nth-child(2) td:nth-child(4)') %>%
  html_text()


#Water Supply Sources#
max_day_use <- webpage %>%
  html_nodes(':nth-child(31) td:nth-child(9) , tr:nth-child(2) :nth-child(9),
             :nth-child(31) td:nth-child(6), :nth-child(31) td:nth-child(3)') %>%
  html_text()


class(max_day_use)
```

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

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.)

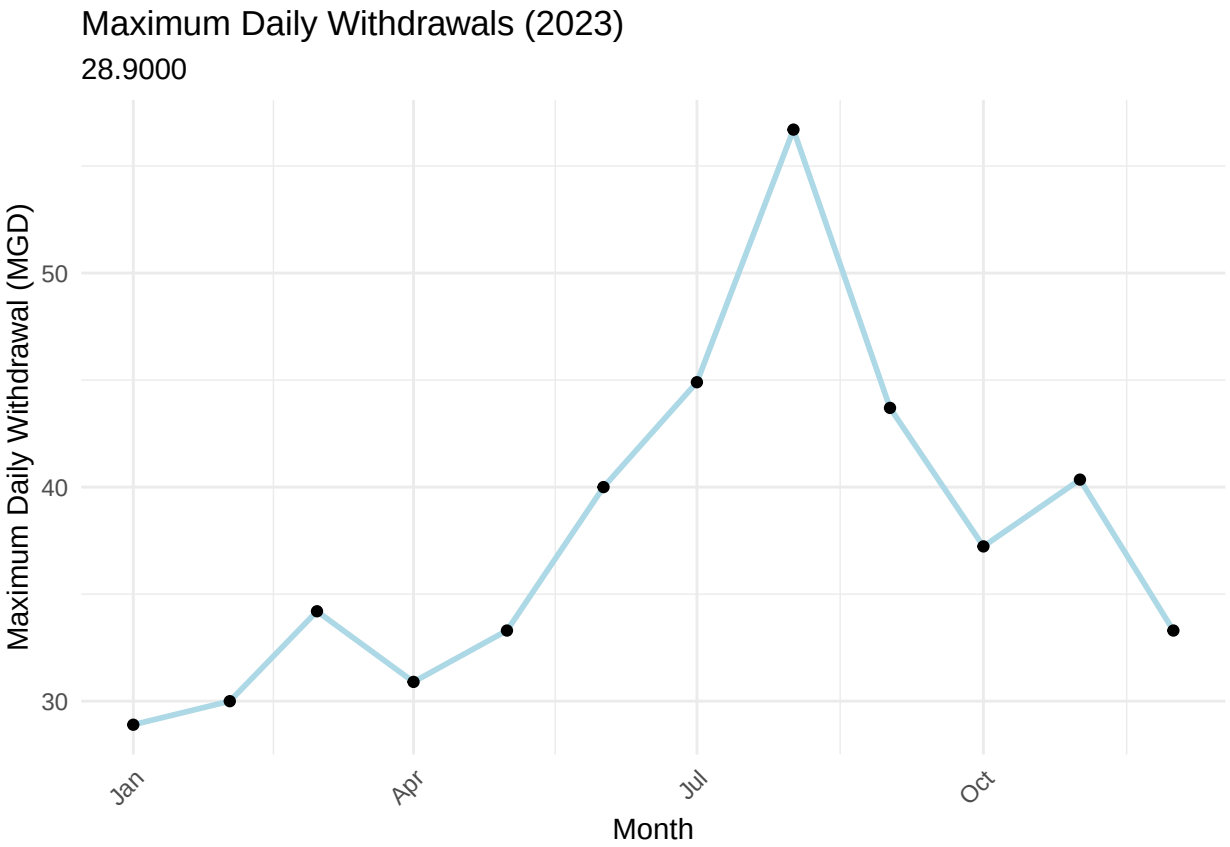
   TIP: Use `rep()` to repeat a value when creating a dataframe.

   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.

```
#4
months <- c("Jan", "May", "Sept", "Feb", "Jun", "Oct", "Mar", "Jul", "Nov",
            "Apr", "Aug", "Dec")

#months <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sept",
#"Oct", "Nov", "Dec")

 df_water <- data.frame(
    Month = months,
    water_system_name = rep(water_system_name, length(months)),
    PWSID = rep(PWSID, length(months)),
    ownership = rep(ownership, length(months)),
    max_day_use = as.numeric(max_day_use),
    Year = as.numeric(2023)) %>%
  mutate(
    date = my(paste(Month,"-",Year)))


#5

ggplot(df_water, aes(x= date, y = max_day_use, group = 1)) +
  geom_line(group = 1, color = "lightblue", size = 1) +
  geom_point() +
     scale_x_date(date_labels = "%b") +
  labs(title = paste("Maximum Daily Withdrawals (2023)"),
       subtitle =max_day_use,
       y = "Maximum Daily Withdrawal (MGD)",
       x = "Month") +
      theme_minimal() +
      theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.
```



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**.

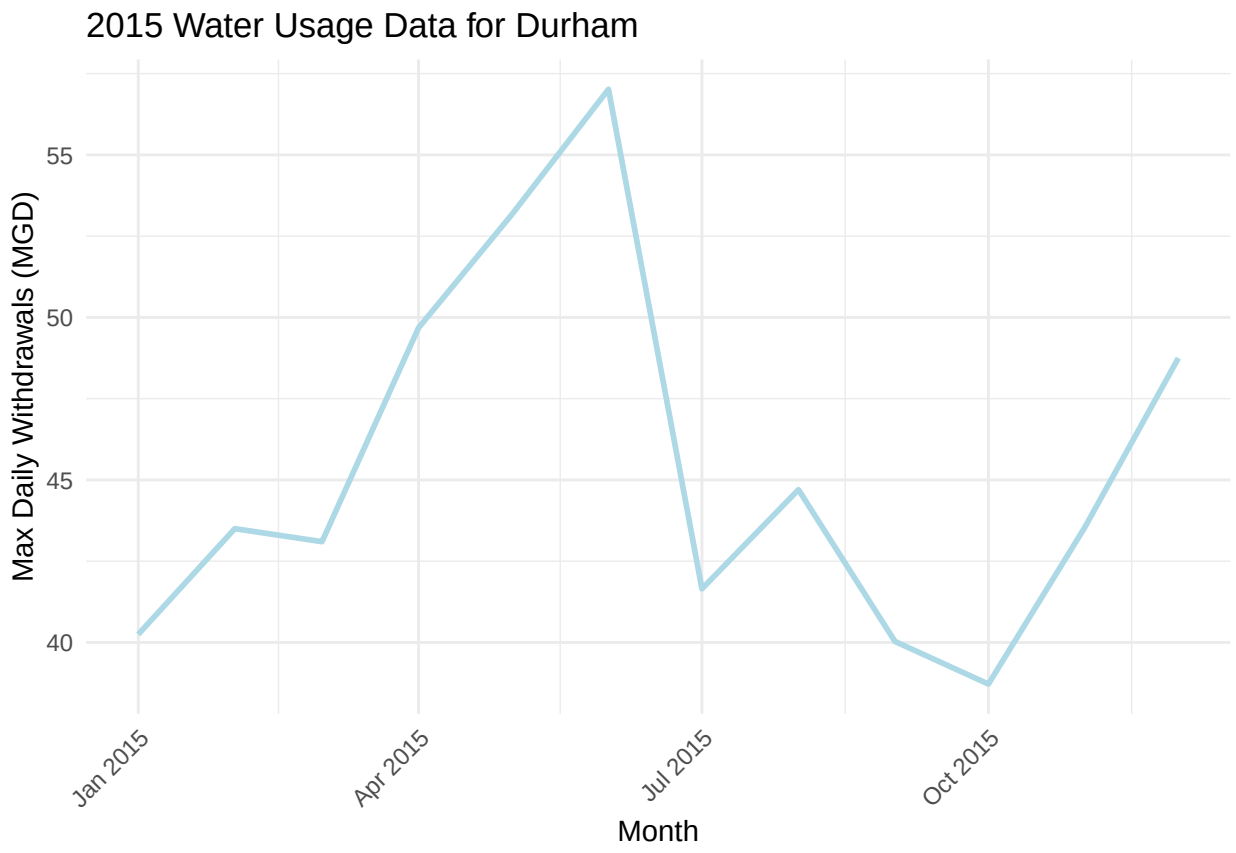
```
#6.


# Define the scraping function
scrape.it <- function(pwsid, year) {

  # Construct the URL using the provided pwsid and year
  the_url <- paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',
                    pwsid, '&year=', year)

  # Fetch the website content
  the_website <- read_html(the_url)

  # Scrape the relevant data
  water_system_name <- the_website %>%
    html_nodes('table:nth-child(7) 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('table:nth-child(7) tr:nth-child(2) td:nth-child(4)') %>%
    html_text()
  max_day_use<- the_website %>% html_nodes('th~ td+ td') %>% html_text()

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

  # Create a dataframe with the scraped data and the month column
  water_supply_df2 <- data.frame(
    Month = months,
    water_system_name = rep(water_system_name, length(months)),
    PWSID = rep(PWSID, length(months)),
    ownership = rep(ownership, length(months)),
    max_day_use = as.numeric(max_day_use),
    Year = as.numeric(year)) %>%
  mutate(
    date = my(paste(Month,"-",Year)))

  #OPTION2:#
   water_supply_dftry <- data.frame(
    Month = months,
    WaterSystemName = rep(water_system_name, length(months)),
    PWSID = rep(PWSID, length(months)),
    Ownership = rep(ownership, length(months)),
    MaxDayUse = max_day_use,
    Year = rep(year, length(months))
  ) %>%
    mutate(Date = parse_date_time(paste(Month, Year), orders = "my"))
  ##

  # Return the dataframe
  return(water_supply_df2)
}
```

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

```
#7

water_data_2015 <- scrape.it("03-32-010",2015)
view(water_data_2015)


# Plot max daily withdrawals for each month
ggplot(water_data_2015, aes(x = date, y = max_day_use)) +
  geom_line(group = 1, color = "lightblue", size = 1) +
  labs(
    title = "2015 Water Usage Data for Durham",
    x = "Month",
    y = "Max Daily Withdrawals (MGD)"
  ) +
  theme_minimal() +
  theme(
```

```
    axis.text.x = element_text(angle = 45, hjust = 1))
```



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.

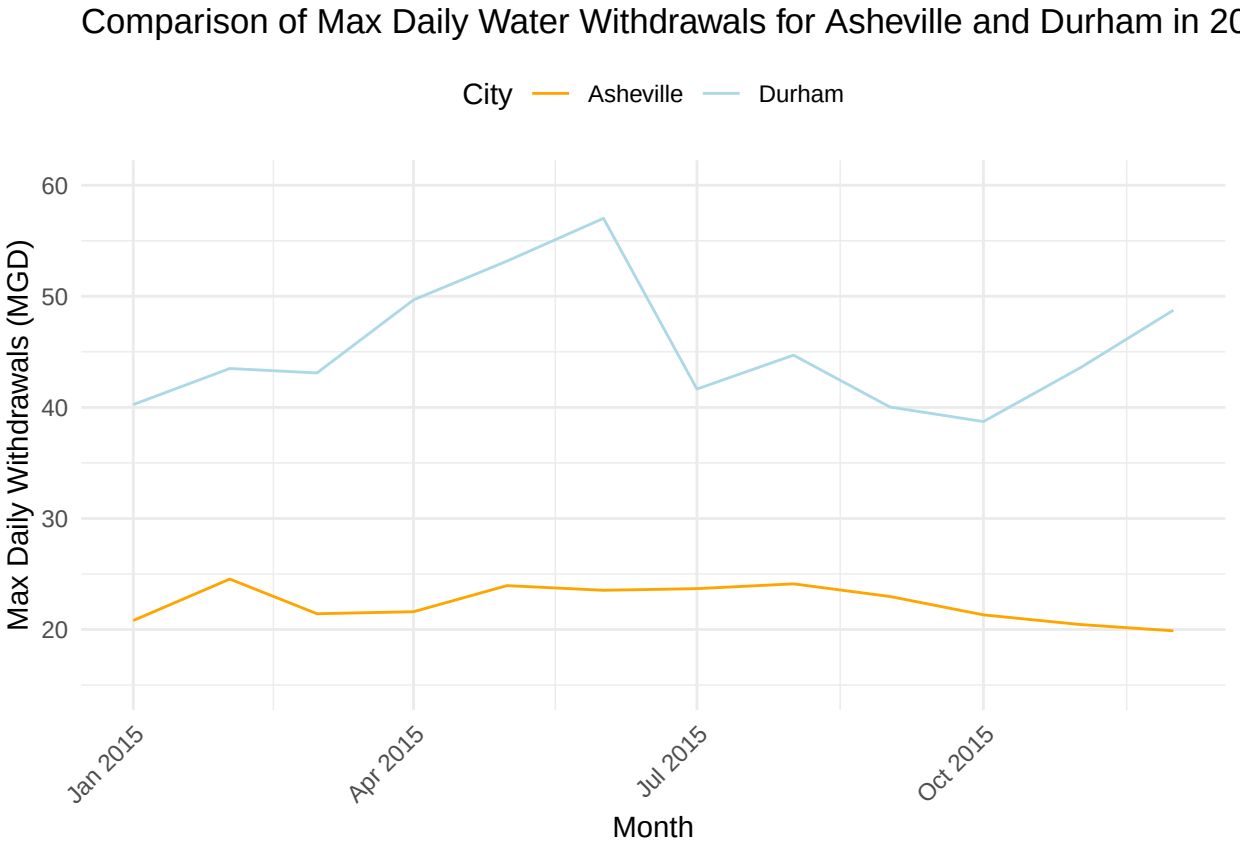
```
#8

durham_2015 <- scrape.it("03-32-010", 2015)
print(durham_2015)
```

```
##    Month water_system_name     PWSID    ownership max_day_use Year       date
## 1    Jan            Durham 03-32-010 Municipality       40.25 2015 2015-01-01
## 2    May            Durham 03-32-010 Municipality       53.17 2015 2015-05-01
## 3   Sept            Durham 03-32-010 Municipality       40.03 2015 2015-09-01
## 4    Feb            Durham 03-32-010 Municipality       43.50 2015 2015-02-01
## 5    Jun            Durham 03-32-010 Municipality       57.02 2015 2015-06-01
## 6    Oct            Durham 03-32-010 Municipality       38.72 2015 2015-10-01
## 7    Mar            Durham 03-32-010 Municipality       43.10 2015 2015-03-01
## 8    Jul            Durham 03-32-010 Municipality       41.65 2015 2015-07-01
## 9    Nov            Durham 03-32-010 Municipality       43.55 2015 2015-11-01
## 10   Apr            Durham 03-32-010 Municipality       49.68 2015 2015-04-01
## 11   Aug            Durham 03-32-010 Municipality       44.70 2015 2015-08-01
## 12   Dec            Durham 03-32-010 Municipality       48.75 2015 2015-12-01
```

```r
asheville_2015 <- scrape.it("01-11-010", 2015)
print(asheville_2015)
```

```
##    Month water_system_name     PWSID    ownership max_day_use Year       date
## 1    Jan         Asheville 01-11-010 Municipality       20.81 2015 2015-01-01
## 2    May         Asheville 01-11-010 Municipality       23.95 2015 2015-05-01
## 3   Sept         Asheville 01-11-010 Municipality       22.97 2015 2015-09-01
## 4    Feb         Asheville 01-11-010 Municipality       24.54 2015 2015-02-01
## 5    Jun         Asheville 01-11-010 Municipality       23.53 2015 2015-06-01
## 6    Oct         Asheville 01-11-010 Municipality       21.32 2015 2015-10-01
## 7    Mar         Asheville 01-11-010 Municipality       21.42 2015 2015-03-01
## 8    Jul         Asheville 01-11-010 Municipality       23.68 2015 2015-07-01
## 9    Nov         Asheville 01-11-010 Municipality       20.45 2015 2015-11-01
## 10   Apr         Asheville 01-11-010 Municipality       21.60 2015 2015-04-01
## 11   Aug         Asheville 01-11-010 Municipality       24.11 2015 2015-08-01
## 12   Dec         Asheville 01-11-010 Municipality       19.88 2015 2015-12-01
```

```r
durham_2015 <- durham_2015 %>%
  mutate(City = "Durham")

asheville_2015 <- asheville_2015 %>%
  mutate(City = "Asheville")

# Combine the data for both cities
durham_asheville_combined <- bind_rows(durham_2015, asheville_2015)


# Plot the comparison of water withdrawals for Asheville and Durham
ggplot(durham_asheville_combined, aes(x = date,
                                      y = max_day_use, color = City,
                                      group = City)) +
  geom_line() +
  scale_y_continuous(limits = c(15, 60))+
    labs(
    title = "Comparison of Max Daily Water Withdrawals for Asheville and Durham in 2015",
    x = "Month",
    y = "Max Daily Withdrawals (MGD)",
    color = "City"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "top"
  ) +
  scale_color_manual(values = c("Durham" = "lightblue", "Asheville" = "orange"))
```



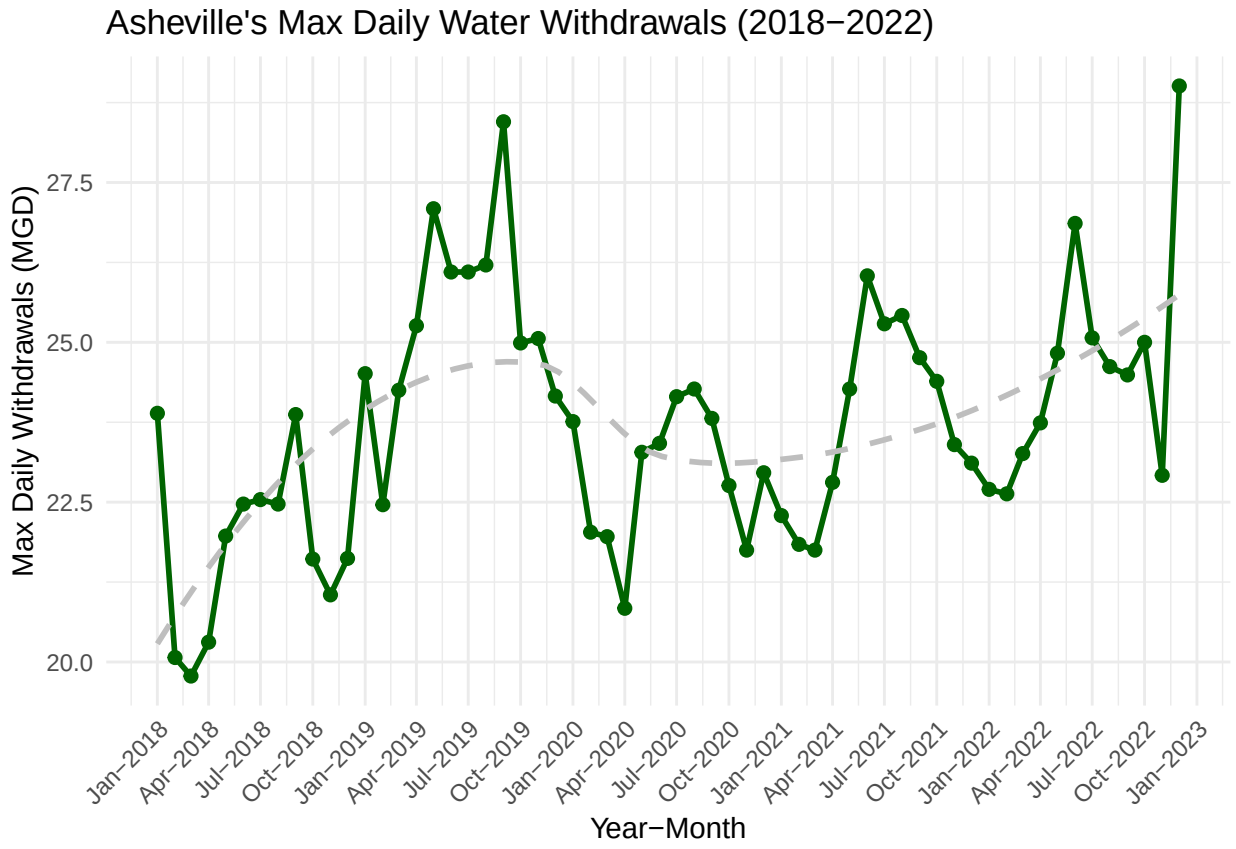
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2022.Add a smoothed line to the plot (method = 'loess').

   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

# Load required libraries
library(ggplot2)
library(purrr)
library(dplyr)

years <- 2018:2022

asheville_all_years <- bind_rows(lapply(years, function(year) scrape.it("01-11-010", year)))


ggplot(asheville_all_years, aes(x = date, y = max_day_use)) +
  geom_line(color = "darkgreen", size = 1) +
  geom_point(color = "darkgreen", size = 2) +
  geom_smooth(method = "loess", color = "grey", linetype = "dashed", se = FALSE) +
   scale_x_date(date_labels = "%b-%Y", date_breaks = "3 months") +
  labs(
```

```
    title = "Asheville's Max Daily Water Withdrawals (2018-2022)",
    x = "Year-Month",
    y = "Max Daily Withdrawals (MGD)"
  ) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    legend.position = "none"
  )
```

```
## `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, there exists a gradual upward trend in Asheville's maximum daily water withdrawals over the years 2018-2022. It does not go upward constantly as there are some seasonal flunctions but the grey line shows the overall increasing trend. >