Assignment 09: Data Scraping

Hanna Bliska

OVERVIEW

This exercise accompanies the lessons in Environmental Data Analytics on data scraping.

Directions

1. Rename this file `<FirstLast>_A09_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 to **answer the questions** in this assignment document.
5. When you have completed the assignment, **Knit** the text and code into a single PDF file.

Set up

1. Set up your session:

- Check your working directory
- Load the packages `tidyverse`, `rvest`, and any others you end up using.
- Set your ggplot theme

```
#1
getwd()
```

```
## [1] "/Users/hbliska/Desktop/EDA-Fall2022"
```

```
library(tidyverse)
library(lubridate)
library(rvest)
library(scales)

mytheme <- theme_classic(base_size = 12) + theme(
  axis.text = element_text(color="black"),
  legend.position = "right") #creating a theme

theme_set(mytheme) #setting my theme
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2021 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=2021

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

```
#2

webpage <- read_html(
  "https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2021")
#fetching contents into webpage object

webpage #viewing object
```

```
## {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
- PSWID
- Ownership
- From the "3. Water Supply Sources" section:
- Maximum Daily 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), with the first value being "27.6400".

```
#3
water.system.name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water.system.name
```

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

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

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

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

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

```
max.withdrawals.mgd <- webpage %>%
  html_nodes("th~ td+ td , th~ td+ td") %>%
  html_text()
max.withdrawals.mgd
```

```
##  [1] "27.6400" "41.7900" "36.7200" "27.9700" "37.9500" "42.2400" "30.5400"
##  [8] "43.6200" "31.2800" "33.7600" "46.0800" "29.7800"
```

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...

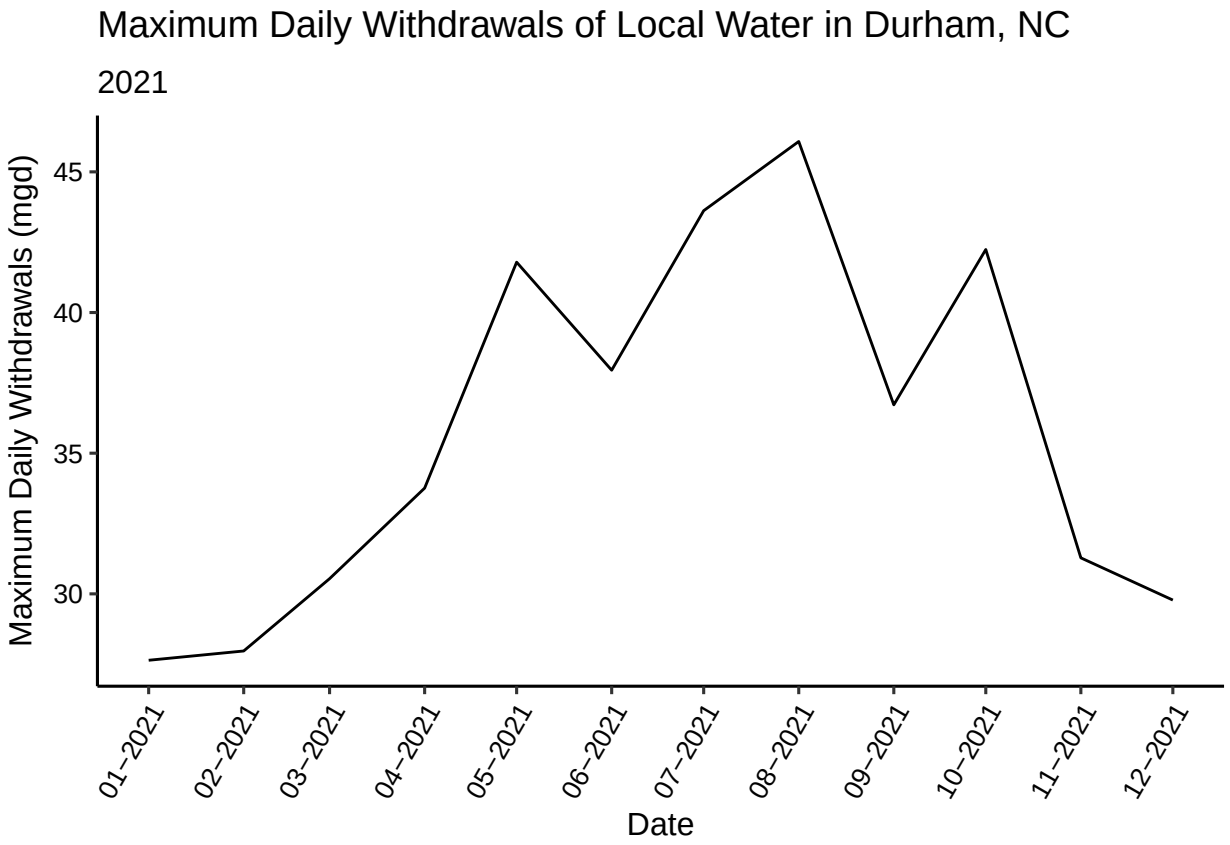
5. Create a line plot of the maximum daily withdrawals across the months for 2021

```
#4
max_withdrawals_df <- data.frame(
  "Month"=as.factor(c("Jan", "May", "Sept", "Feb", "Jun",
                      "Oct", "Mar", "Jul", "Nov",
                      "Apr", "Aug", "Dec")), #creating month vector
  "Year"=as.factor(rep(2021,12)), #repeating 2021 for all 12 values
  "Maximum.Daily.Withdrawals"=as.numeric(max.withdrawals.mgd))

max_withdrawals_df <- max_withdrawals_df %>%
  mutate("Water.System.Name"=!!water.system.name,
         "PSWID"=!!pswid,
         "Ownership"=ownership,
         Date = my(paste(Month,"-",Year))) %>%
  #using mutate to create columns retrieving the scraped variables
  arrange(ymd(Date)) #arranging in chronological order

max_withdrawals_df #viewing data frame
```

```
##    Month Year Maximum.Daily.Withdrawals Water.System.Name     PSWID
## 1    Jan 2021                     27.64            Durham 03-32-010
## 2    Feb 2021                     27.97            Durham 03-32-010
## 3    Mar 2021                     30.54            Durham 03-32-010
```

```
## 4    Apr 2021                    33.76           Durham 03-32-010
## 5    May 2021                    41.79           Durham 03-32-010
## 6    Jun 2021                    37.95           Durham 03-32-010
## 7    Jul 2021                    43.62           Durham 03-32-010
## 8    Aug 2021                    46.08           Durham 03-32-010
## 9   Sept 2021                    36.72           Durham 03-32-010
## 10   Oct 2021                    42.24           Durham 03-32-010
## 11   Nov 2021                    31.28           Durham 03-32-010
## 12   Dec 2021                    29.78           Durham 03-32-010
##       Ownership       Date
## 1  Municipality 2021-01-01
## 2  Municipality 2021-02-01
## 3  Municipality 2021-03-01
## 4  Municipality 2021-04-01
## 5  Municipality 2021-05-01
## 6  Municipality 2021-06-01
## 7  Municipality 2021-07-01
## 8  Municipality 2021-08-01
## 9  Municipality 2021-09-01
## 10 Municipality 2021-10-01
## 11 Municipality 2021-11-01
## 12 Municipality 2021-12-01
```

```r
#5
max_withdrawals_plot <-
  ggplot(max_withdrawals_df, aes(
    x=Date, y=Maximum.Daily.Withdrawals))+
  geom_line() + #creating line plot
  scale_x_date(date_breaks="1 month", labels=date_format("%m-%Y")) +
  #using scale_x_date to make a break for each month in x axis
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  #tilting the angle of the x axis text
  ylab(expression("Maximum Daily Withdrawals (mgd)")) +
  #setting y axis label
  xlab(expression("Date")) + #setting x axis label
   ggtitle(expression(
    "Maximum Daily Withdrawals of Local Water in Durham, NC"),
    subtitle="2021") #adding title and subtitle
max_withdrawals_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.
scrape.it <- function(the_PWSID, the_year){

  #retrieve website contents
  webpage <-read_html(paste0(
    'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=',
    the_PWSID, "&year=", the_year))

  #setting element address variables
  the_PWSID_tag <-'td tr:nth-child(1) td:nth-child(5)'
  the_water_name_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_data_tag <- 'th~ td+ td , th~ td+ td'

  #scraping the data items
  the_PWSID <- webpage %>% html_nodes(
    the_PWSID_tag) %>% html_text()
  the_water_name <- webpage %>% html_nodes(
    the_water_name_tag) %>% html_text()
  the_ownership <- webpage %>% html_nodes(
    the_ownership_tag) %>% html_text()
```

```
  the_daily_withdrawals <- webpage %>% html_nodes(
    the_data_tag) %>% html_text()

  #creating a scraped data frame
  scrape_max_withdrawals_df <- data.frame(
    "Month" = as.factor(c("Jan", "May", "Sept",
                          "Feb", "Jun","Oct",
                          "Mar", "Jul", "Nov",
                          "Apr", "Aug", "Dec")),
    "Year" = as.factor(rep(the_year,12)),
    "Maximum.Daily.Withdrawals"=as.numeric(the_daily_withdrawals))

  scrape_max_withdrawals_df <-scrape_max_withdrawals_df %>%
  mutate("PSWID" =!!the_PWSID,
         "Water.System.Name"=!!the_water_name,
         "Ownership"=!!the_ownership,
         "Date"=my(paste(Month,"-",!!the_year))) %>%
    #using mutate to create columns retrieving the scraped variables
  arrange(ymd(Date)) #arranging in chronological order
}
scrape_max_withdrawals_df <- scrape.it("03-32-010", 2021)
#scraping for Durham PSWID (03-32-010) and 2021 year
scrape_max_withdrawals_df #viewing data frame
```

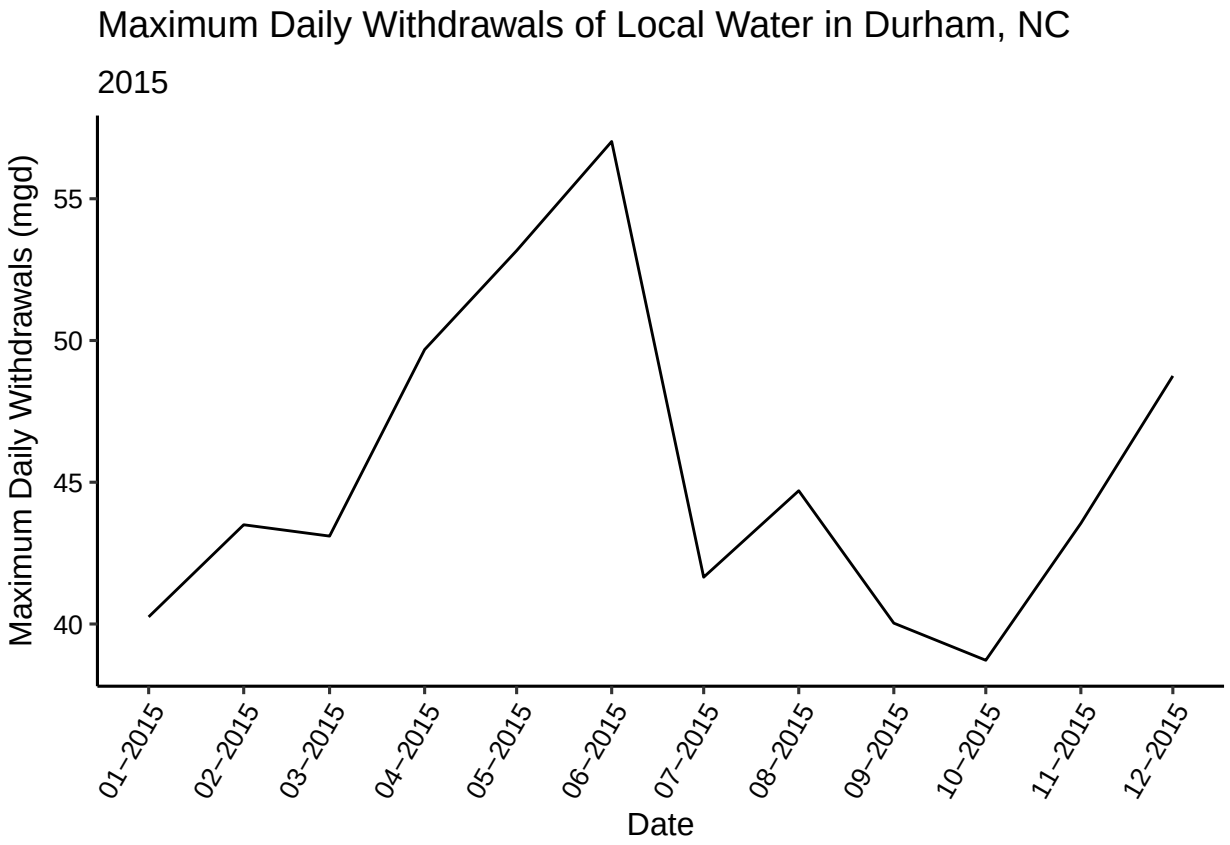
```
##    Month Year Maximum.Daily.Withdrawals     PSWID Water.System.Name
## 1    Jan 2021                     27.64 03-32-010            Durham
## 2    Feb 2021                     27.97 03-32-010            Durham
## 3    Mar 2021                     30.54 03-32-010            Durham
## 4    Apr 2021                     33.76 03-32-010            Durham
## 5    May 2021                     41.79 03-32-010            Durham
## 6    Jun 2021                     37.95 03-32-010            Durham
## 7    Jul 2021                     43.62 03-32-010            Durham
## 8    Aug 2021                     46.08 03-32-010            Durham
## 9   Sept 2021                     36.72 03-32-010            Durham
## 10   Oct 2021                     42.24 03-32-010            Durham
## 11   Nov 2021                     31.28 03-32-010            Durham
## 12   Dec 2021                     29.78 03-32-010            Durham
##       Ownership       Date
## 1  Municipality 2021-01-01
## 2  Municipality 2021-02-01
## 3  Municipality 2021-03-01
## 4  Municipality 2021-04-01
## 5  Municipality 2021-05-01
## 6  Municipality 2021-06-01
## 7  Municipality 2021-07-01
## 8  Municipality 2021-08-01
## 9  Municipality 2021-09-01
## 10 Municipality 2021-10-01
## 11 Municipality 2021-11-01
## 12 Municipality 2021-12-01
```

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

```
#7
#testing the function
dur2015_scrape_max_withdrawals_df <- scrape.it('03-32-010', 2015)
#scraping for Durham PSWID (03-32-010) and 2015 year
dur2015_scrape_max_withdrawals_df #viewing data frame
```

```
##    Month Year Maximum.Daily.Withdrawals     PSWID Water.System.Name
## 1    Jan 2015                     40.25 03-32-010            Durham
## 2    Feb 2015                     43.50 03-32-010            Durham
## 3    Mar 2015                     43.10 03-32-010            Durham
## 4    Apr 2015                     49.68 03-32-010            Durham
## 5    May 2015                     53.17 03-32-010            Durham
## 6    Jun 2015                     57.02 03-32-010            Durham
## 7    Jul 2015                     41.65 03-32-010            Durham
## 8    Aug 2015                     44.70 03-32-010            Durham
## 9   Sept 2015                     40.03 03-32-010            Durham
## 10   Oct 2015                     38.72 03-32-010            Durham
## 11   Nov 2015                     43.55 03-32-010            Durham
## 12   Dec 2015                     48.75 03-32-010            Durham
##       Ownership       Date
## 1  Municipality 2015-01-01
## 2  Municipality 2015-02-01
## 3  Municipality 2015-03-01
## 4  Municipality 2015-04-01
## 5  Municipality 2015-05-01
## 6  Municipality 2015-06-01
## 7  Municipality 2015-07-01
## 8  Municipality 2015-08-01
## 9  Municipality 2015-09-01
## 10 Municipality 2015-10-01
## 11 Municipality 2015-11-01
## 12 Municipality 2015-12-01
```

```
#plotting
max_withdrawals_2015_plot <-
  ggplot(dur2015_scrape_max_withdrawals_df, aes(
    x=Date, y=Maximum.Daily.Withdrawals))+
  geom_line() + #creating line plot
  scale_x_date(date_breaks="1 month", labels=date_format("%m-%Y")) +
  #using scale_x_date to make a break for each month in x axis
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  #tilting the angle of the x axis text
  ylab(expression("Maximum Daily Withdrawals (mgd)")) +
  #setting y axis label
  xlab(expression("Date")) + #setting x axis label
  ggtitle(expression(
    "Maximum Daily Withdrawals of Local Water in Durham, NC"),
    subtitle="2015") #adding title and subtitle
max_withdrawals_2015_plot
```



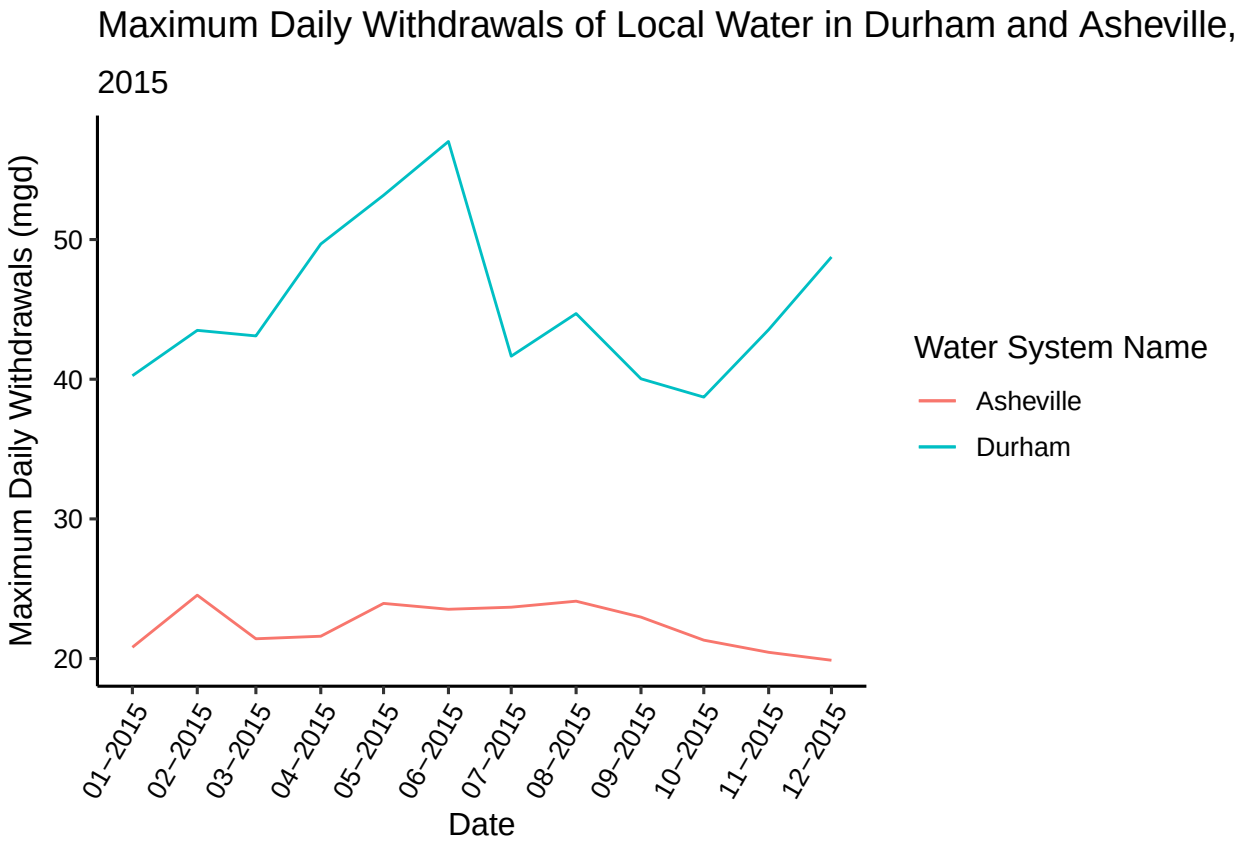
8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares Asheville's to Durham's water withdrawals.

```
#8
ash2015_scrape_max_withdrawals_df <- scrape.it('01-11-010', 2015)
#scraping for Asheville PSWID (01-11-010) and 2015 year
ash2015_scrape_max_withdrawals_df #viewing data frame
```

```
##    Month Year Maximum.Daily.Withdrawals     PSWID Water.System.Name
## 1    Jan 2015                     20.81 01-11-010         Asheville
## 2    Feb 2015                     24.54 01-11-010         Asheville
## 3    Mar 2015                     21.42 01-11-010         Asheville
## 4    Apr 2015                     21.60 01-11-010         Asheville
## 5    May 2015                     23.95 01-11-010         Asheville
## 6    Jun 2015                     23.53 01-11-010         Asheville
## 7    Jul 2015                     23.68 01-11-010         Asheville
## 8    Aug 2015                     24.11 01-11-010         Asheville
## 9   Sept 2015                     22.97 01-11-010         Asheville
## 10   Oct 2015                     21.32 01-11-010         Asheville
## 11   Nov 2015                     20.45 01-11-010         Asheville
## 12   Dec 2015                     19.88 01-11-010         Asheville
##       Ownership       Date
## 1  Municipality 2015-01-01
## 2  Municipality 2015-02-01
```

```
## 3  Municipality 2015-03-01
## 4  Municipality 2015-04-01
## 5  Municipality 2015-05-01
## 6  Municipality 2015-06-01
## 7  Municipality 2015-07-01
## 8  Municipality 2015-08-01
## 9  Municipality 2015-09-01
## 10 Municipality 2015-10-01
## 11 Municipality 2015-11-01
## 12 Municipality 2015-12-01
```

```
ash_durham_2015_plot <- ggplot() +
  geom_line(data=dur2015_scrape_max_withdrawals_df,
            aes(
    x=Date, y=Maximum.Daily.Withdrawals, color="Durham")) +
  #creating line plot
  geom_line(data=ash2015_scrape_max_withdrawals_df,
            aes(
    x=Date, y=Maximum.Daily.Withdrawals, color="Asheville")) +
  scale_x_date(date_breaks="1 month", labels=date_format("%m-%Y")) +
  #using scale_x_date to make a break for each month in x axis
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  #tilting the angle of the x axis text
  ylab(expression("Maximum Daily Withdrawals (mgd)")) +
  #setting y axis label
  xlab(expression("Date")) + #setting x axis label
  labs(color="Water System Name") + #setting legend title
  ggtitle(expression(
    "Maximum Daily Withdrawals of Local Water in Durham and Asheville, NC"),
    subtitle="2015") #adding title and subtitle
ash_durham_2015_plot
```



9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019. Add a smoothed line to the plot.

   TIP: See Section 3.2 in the "09_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
the_years <- seq(2010,2019) #creating a sequence of years

ash_2010_2019_df <- map2("01-11-010",the_years,scrape.it) %>%
  #using map2 to run the function scrape.it with two inputs
  #two inputs are the PSWID and the years
  bind_rows() #binding the data frames to a single one
head(ash_2010_2019_df) #viewing first few rows of the data frame
```

```
##   Month Year Maximum.Daily.Withdrawals     PSWID Water.System.Name    Ownership
## 1   Jan 2010                     21.89 01-11-010         Asheville Municipality
## 2   Feb 2010                     19.95 01-11-010         Asheville Municipality
## 3   Mar 2010                     19.74 01-11-010         Asheville Municipality
## 4   Apr 2010                     21.25 01-11-010         Asheville Municipality
## 5   May 2010                     20.99 01-11-010         Asheville Municipality
## 6   Jun 2010                     22.53 01-11-010         Asheville Municipality
##         Date
## 1 2010-01-01
```

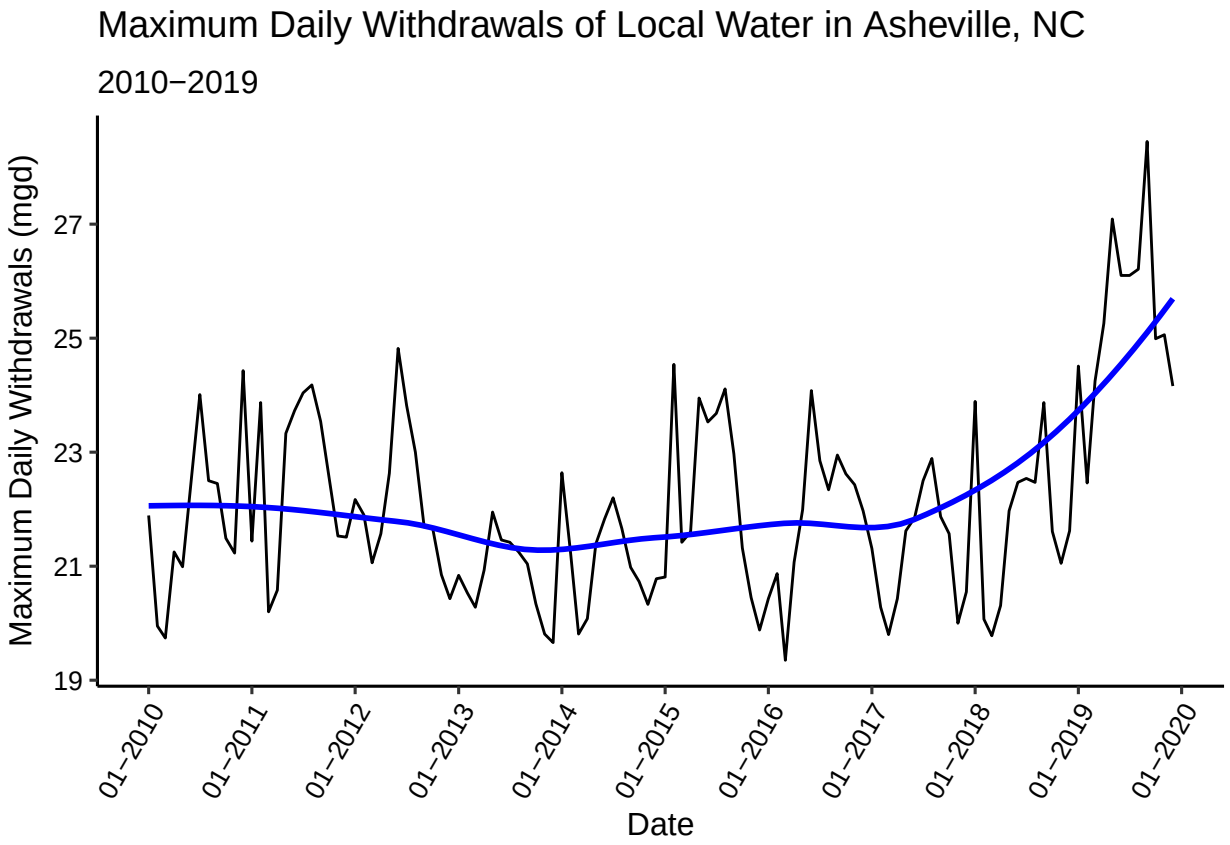
```
## 2 2010-02-01
## 3 2010-03-01
## 4 2010-04-01
## 5 2010-05-01
## 6 2010-06-01
```

```
tail(ash_2010_2019_df) #viewing last few rows of the data frame
```

```
##     Month Year Maximum.Daily.Withdrawals     PSWID Water.System.Name
## 115   Jul 2019                     26.10 01-11-010         Asheville
## 116   Aug 2019                     26.21 01-11-010         Asheville
## 117  Sept 2019                     28.45 01-11-010         Asheville
## 118   Oct 2019                     24.99 01-11-010         Asheville
## 119   Nov 2019                     25.06 01-11-010         Asheville
## 120   Dec 2019                     24.16 01-11-010         Asheville
##        Ownership       Date
## 115 Municipality 2019-07-01
## 116 Municipality 2019-08-01
## 117 Municipality 2019-09-01
## 118 Municipality 2019-10-01
## 119 Municipality 2019-11-01
## 120 Municipality 2019-12-01
```

```
ash_2010_2019_plot <- ggplot(ash_2010_2019_df, aes(
    x=Date, y=Maximum.Daily.Withdrawals)) +
  geom_line() + #creating line plot
  geom_smooth(method=loess, color="blue", se=FALSE) +
  scale_x_date(date_breaks="1 year", labels=date_format("%m-%Y")) +
  #using scale_x_date to make a break for each month in x axis
  theme(axis.text.x=element_text(angle=60, hjust=1)) +
  #tilting the angle of the x axis text
  ylab(expression("Maximum Daily Withdrawals (mgd)")) +
  #setting y axis label
  xlab(expression("Date")) + #setting x axis label
  ggtitle(expression(
    "Maximum Daily Withdrawals of Local Water in Asheville, NC"),
    subtitle="2010-2019") #adding title and subtitle
ash_2010_2019_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: Yes, it appears that Asheville has an increase in water usage over time. Particularly, after 2017, it appears that water usage has drastically increased in Asheville. Before 2017, water usage appeared to be relatively constant.