Assignment 09: Data Scraping

Sara Diamond

Total points:

OVERVIEW

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

Directions

1. Change "Student Name" on line 3 (above) with your name.
2. Work through the steps, **creating code and output** that fulfill each instruction.
3. Be sure to **answer the questions** in this assignment document.
4. When you have completed the assignment, **Knit** the text and code into a single PDF file.
5. After Knitting, submit the completed exercise (PDF file) to the dropbox in Sakai. Add your last name into the file name (e.g., "Fay_09_Data_Scraping.Rmd") prior to submission.

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 working directory and laoding packages
getwd()
```

```
## [1] "/Users/saradiamond/Documents/Environmental_Data_Analytics_2022"
```

```
setwd("/Users/saradiamond/Documents/Environmental_Data_Analytics_2022")
library(tidyverse)
library(rvest)
library(lubridate)

#setting them
mytheme <- theme_classic() +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2019 Municipal Local Water Supply Plan (LWSP):

- Navigate to https://www.ncwater.org/WUDC/app/LWSP/search.php
- Change the date from 2020 to 2019 in the upper right corner.
- 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=2020

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

```
#2 reading
the_url <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')
the_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:
- Average Daily Use (MGD) - for each month

In the code chunk below scrape these values, assigning them to three 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, with the first value being 36.0100.

```
#3 scraping in the system information section
water.system.name <-  the_url %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()

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

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

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

month <- c("Jan", "May", "Sep", "Feb", "Jun", "Oct", "Mar", "Jul",
           "Nov", "Apr", "Aug", "Dec")
```

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 order. You can overcome this by creating a month column in the same order the data are scraped: Jan, May, Sept, Feb, etc...

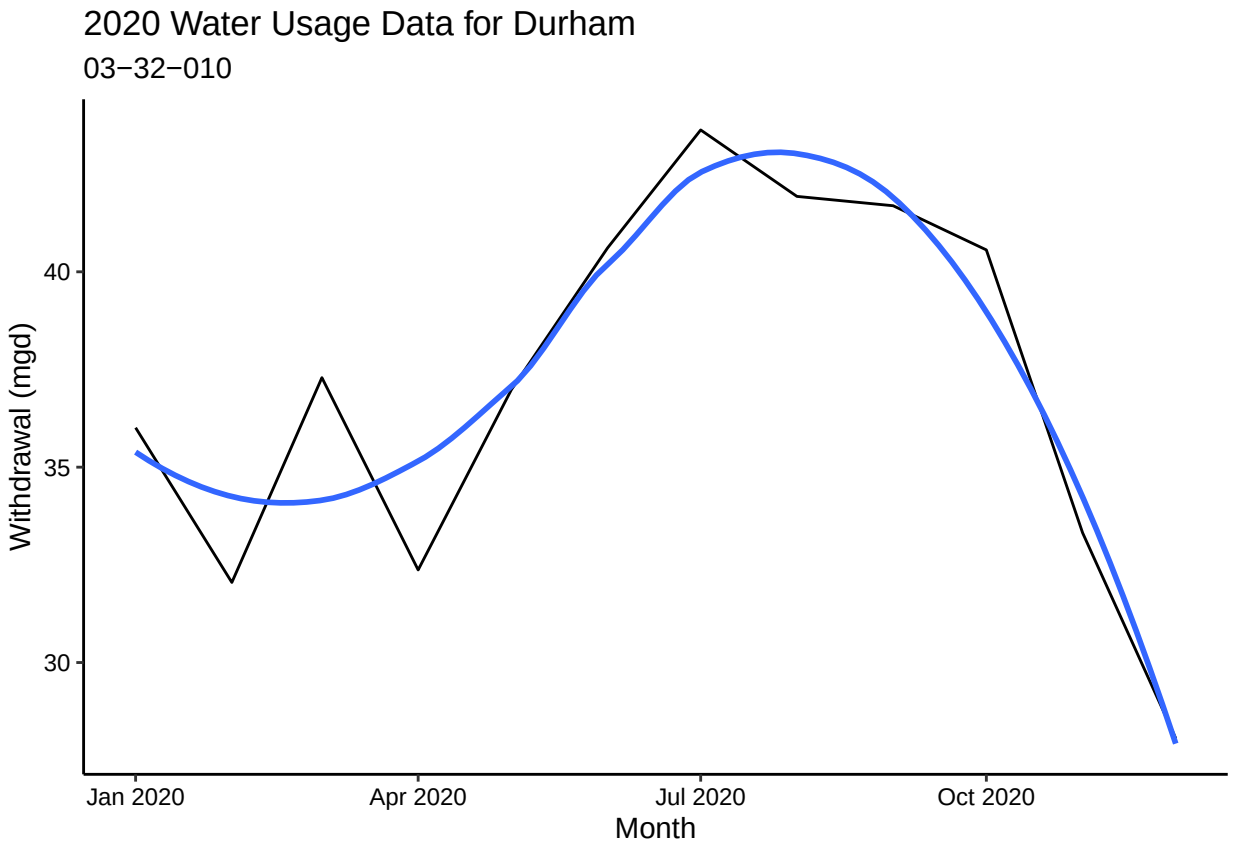
5. Plot the max daily withdrawals across the months for 2020

```
#4 creating a dataframe and naming the columns accordinglh
df_withdrawals <- data.frame("Month" = month,
                             "Year" = rep(2020,12),
                             "Max_Withdrawals_mgd" = as.numeric(max.withdrawals.mgd))

df_withdrawals <- df_withdrawals %>%
  mutate(Water_System = !!water.system.name,
         PWSID = !!pwsid,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year)))

#5 plotting the data
ggplot(df_withdrawals,aes(x=Date,y=Max_Withdrawals_mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title = paste("2020 Water Usage Data for",water.system.name),
       subtitle = pwsid,
       y="Withdrawal (mgd)",
       x="Month")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



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 scraped**.

```
#6.
the_base_url <- "https://www.ncwater.org/WUDC/app/LWSP/report.php?"
pwsid_code <- "03-32-010"
the_year <- 2020
the_scrape_url <- paste0(the_base_url, '/', pwsid_code, '/', the_year)
print(the_scrape_url)
```

```
## [1] "https://www.ncwater.org/WUDC/app/LWSP/report.php?/03-32-010/2020"
```

```
the_website <- read_html(the_scrape_url)

#building the scrape function
water.scrape.it <- function(the_year, pwsid_code){
  the_url <- read_html(paste0
                       ('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid='
                         ,pwsid_code,'&year=',the_year))


  #Setting the tags
  water.system.tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
  pwsid.tag <- 'td tr:nth-child(1) td:nth-child(5)'
  ownership.tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'
  max.withdrawals.mgd.tag <- 'th~ td+ td'

  #Scraping the data items
  the.water.system <- the_url %>% html_nodes(water.system.tag) %>% html_text()
  the.pwsid <- the_url %>%   html_nodes(pwsid.tag) %>%  html_text()
  the.ownership <- the_url %>% html_nodes(ownership.tag) %>% html_text()
  max.withdrawals <- the_url %>% html_nodes(max.withdrawals.mgd.tag) %>% html_text()
  month <- c("Jan", "May", "Sep", "Feb", "Jun", "Oct", "Mar", "Jul",
           "Nov", "Apr", "Aug", "Dec")

  #Convert to a dataframe
  df_withdrawals <- data.frame("Month" = month,
                               "Year" = rep(the_year,12),
                               "Max_Withdrawals.mgd" = as.numeric(max.withdrawals)) %>%
    mutate(Water_System = !!the.water.system,
           PWSID = !!the.pwsid,
           Ownership = !!the.ownership,
           Date = my(paste(Month,"-",Year)))



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

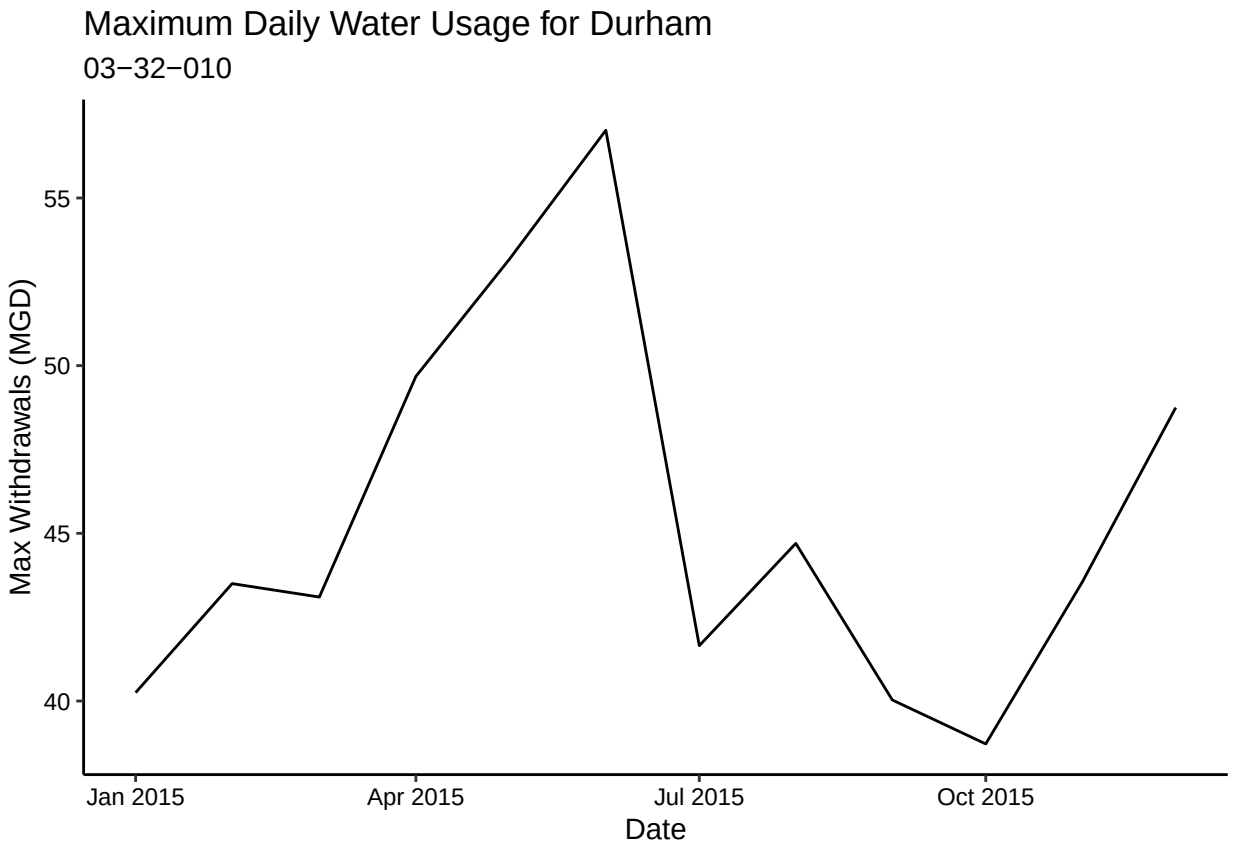
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

```
#7 getting the data for Durham for 2015

water.durham.2015 <- water.scrape.it(2015, "03-32-010")

#plotting the data
```

```
ggplot(water.durham.2015) +
  geom_line(aes(x = Date, y = Max_Withdrawals.mgd)) +
  labs(x = "Date", y = "Max Withdrawals (MGD)",
       title = "Maximum Daily Water Usage for Durham",
       subtitle = pwsid)
```

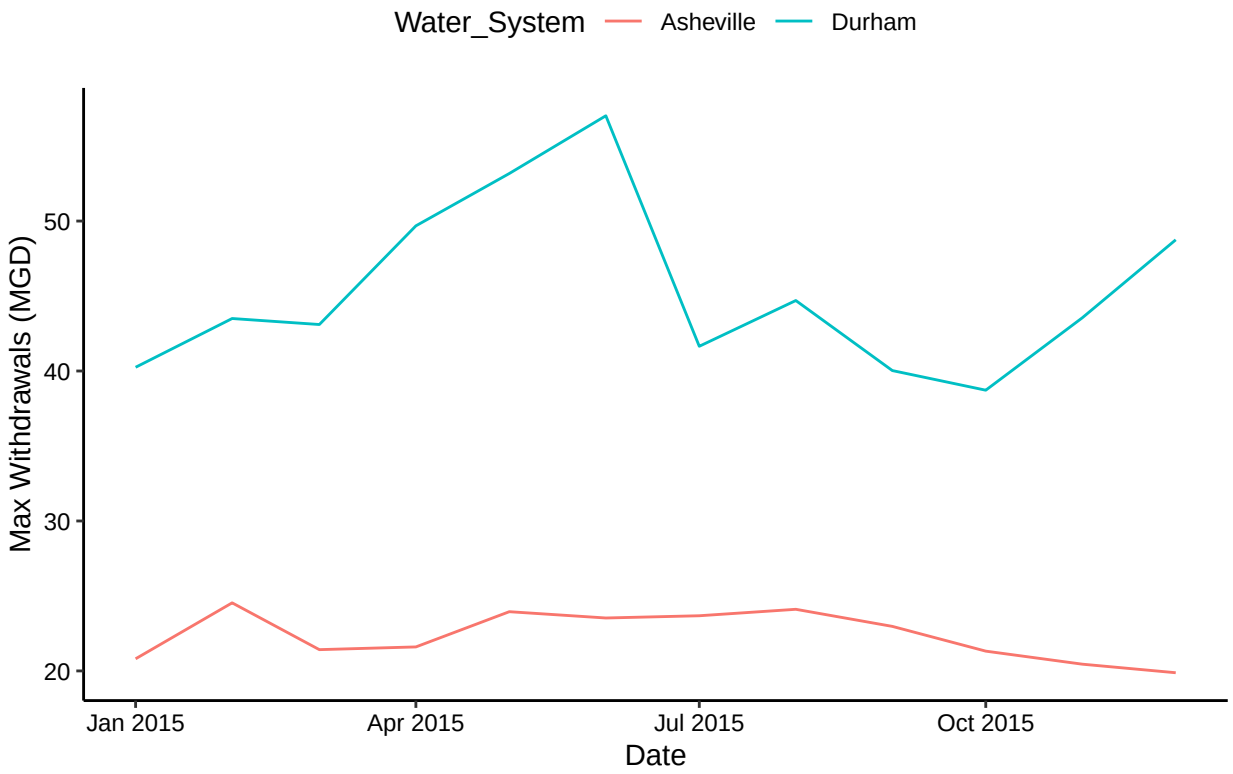


8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares the Asheville to Durham's water withdrawals.

```
#8 getting data from 2015 from asheville
water.asheville.2015 <- water.scrape.it(2015, "01-11-010")

#combining the data
complete.data <- rbind(water.durham.2015, water.asheville.2015)

#plotting combined data
ggplot(complete.data) +
  geom_line(aes(x = Date, y = Max_Withdrawals.mgd, color = Water_System)) +
  labs(x = "Date", y = "Max Withdrawals (MGD)",
       title = "Maximum Daily Water Usage for Durham and Asheville in 2015")
```



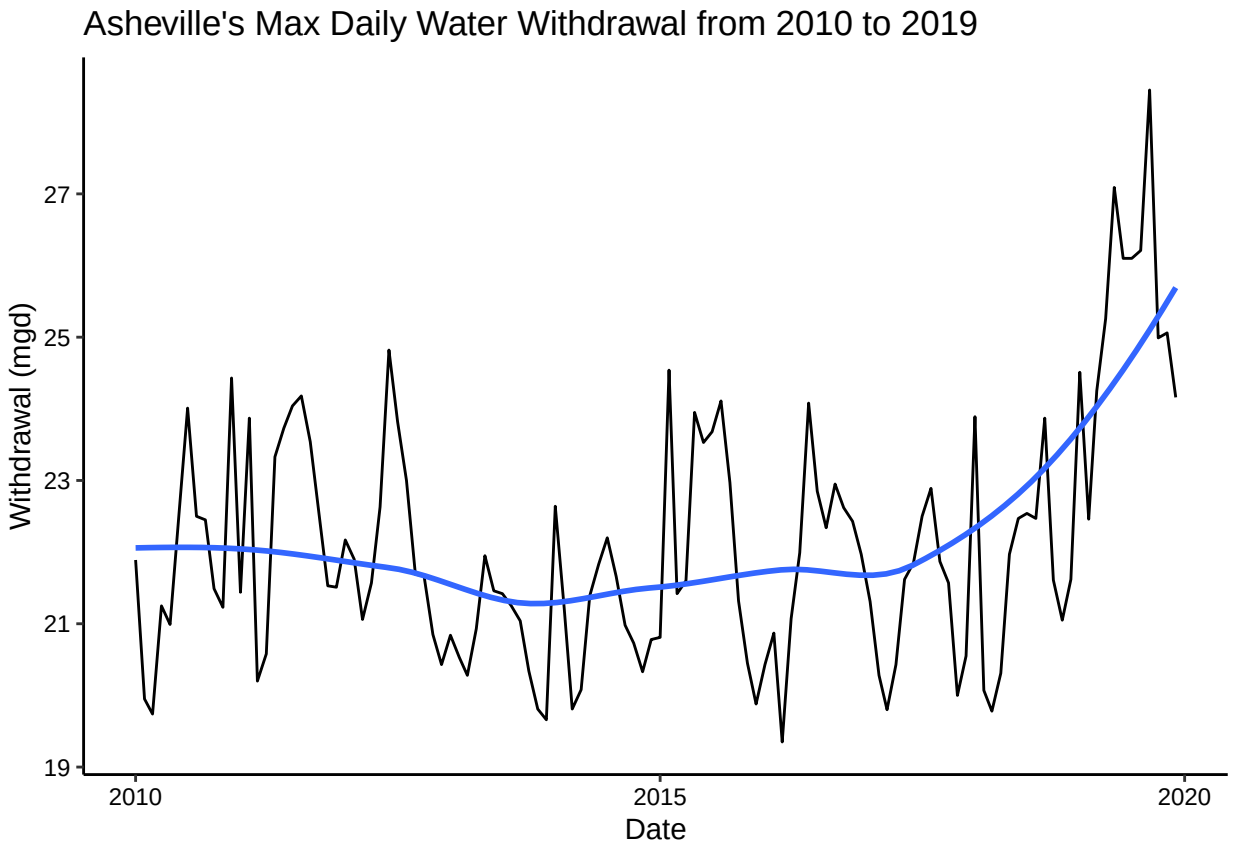
9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

```
#9
the_year = rep(2010:2019)
pwsid_code = '01-11-010'

asheville.data <- lapply(X = the_year,
                  FUN = water.scrape.it,
                  pwsid_code=pwsid_code)
asheville.dataframe <- bind_rows(asheville.data)

ggplot(asheville.dataframe,aes(x=Date,y=Max_Withdrawals.mgd)) +
  geom_line() +
  geom_smooth(method="loess",se=FALSE) +
  labs(title =
         paste("Asheville's Max Daily Water Withdrawal from 2010 to 2019"),
       y="Withdrawal (mgd)",
       x="Date")
```

```
## `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? It seems like there daily max withdrawal has definitely increased over time, but it tends to go up and down throughout the years with no obvious trend.