Assignment 10: Data Scraping

Aditi Jackson

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
# installing and loading 'tidyverse' and 'rvest' packages
#install.packages(tidyverse)
#install.packages(rvest)
#install.packages(lubridate)
library(tidyverse)
library(rvest)
library(lubridate)

# checking working directory is set to "ENV872 Setup (local)
getwd()
```

```
## [1] "/Users/aditijackson/ENV872 Setup (local)"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2022 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=2022

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

```
#2
# setting URL to be scraped
URL_scrape <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
```

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

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

```
# scarping PWSID
PWSID <- URL_scrape %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
PWSID
```

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

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

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

```
# scraping Maximum Daily Use (MGD)
maxDailyUse <- URL_scrape %>%
  html_nodes("th~ td+ td") %>%
  html_text()
maxDailyUse
```

```
##  [1] "36.1000" "43.4200" "52.4900" "30.5000" "42.5900" "34.8800" "39.9100"
##  [8] "43.3200" "32.5300" "34.6600" "41.8000" "37.5300"
```

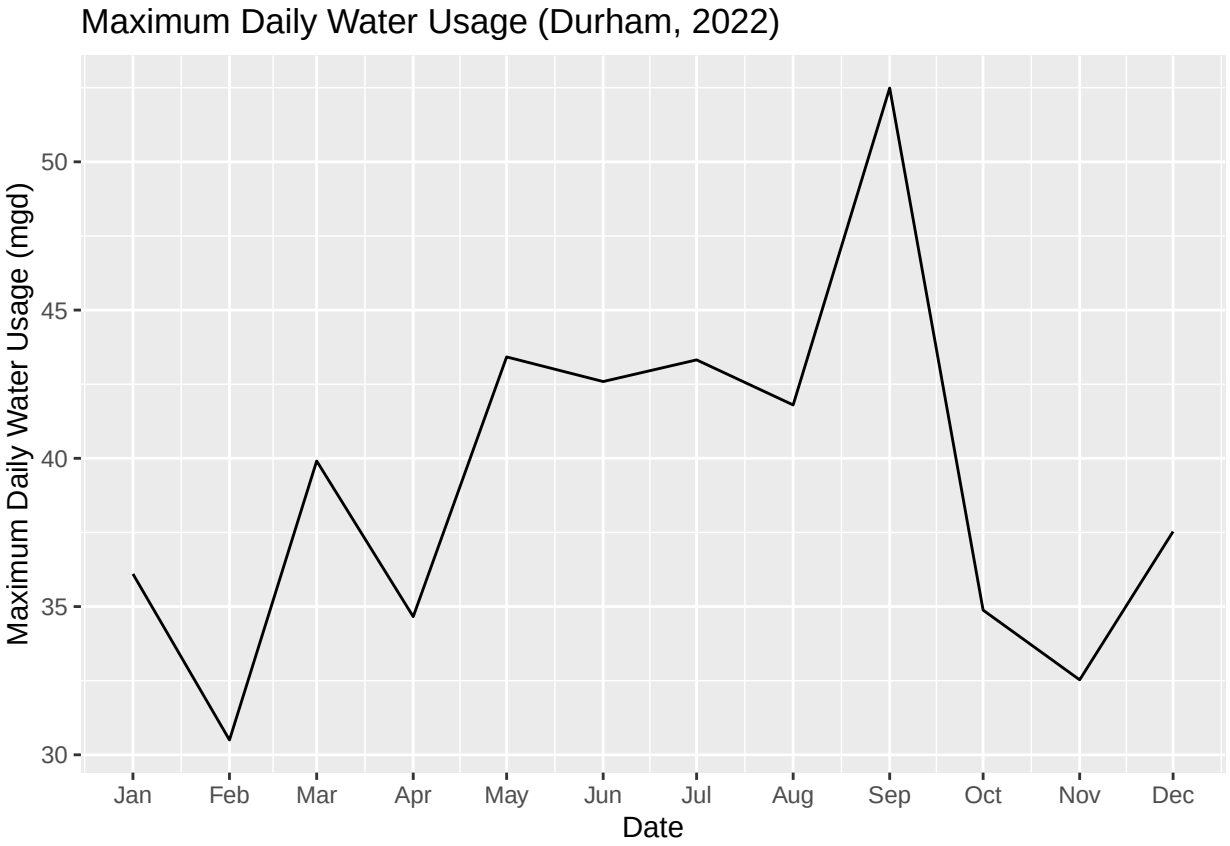
4. Convert your scraped data into a dataframe. This dataframe should have a column for each of the 4 variables scraped and a row for the month corresponding to the withdrawal data. Also add a Date column that includes your month and year in data format. (Feel free to add a Year column too, if you wish.)

   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 2022

```
#4

#creating a dataframe with Month, Year, and Max Daily Value
scraped_durham <- data.frame("Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                         "Year" = rep(2022,12),
                         "Maximum.Daily.Use"=as.numeric(maxDailyUse))  %>%
  # creating pipe, adding remaining scrapped data as columns and creating a date column
  mutate(Water.System.Name = !!waterSystemName,
         PWSID = !!PWSID,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year)))

#5
ggplot(scraped_durham)+
  geom_line(aes(x=Date,y=Maximum.Daily.Use))+
  labs(title=paste0("Maximum Daily Water Usage (Durham, 2022)"),
     y="Maximum Daily Water Usage (mgd)",
      x="Date")+
  scale_x_date(date_labels = "%b", date_breaks = "1 month")
```



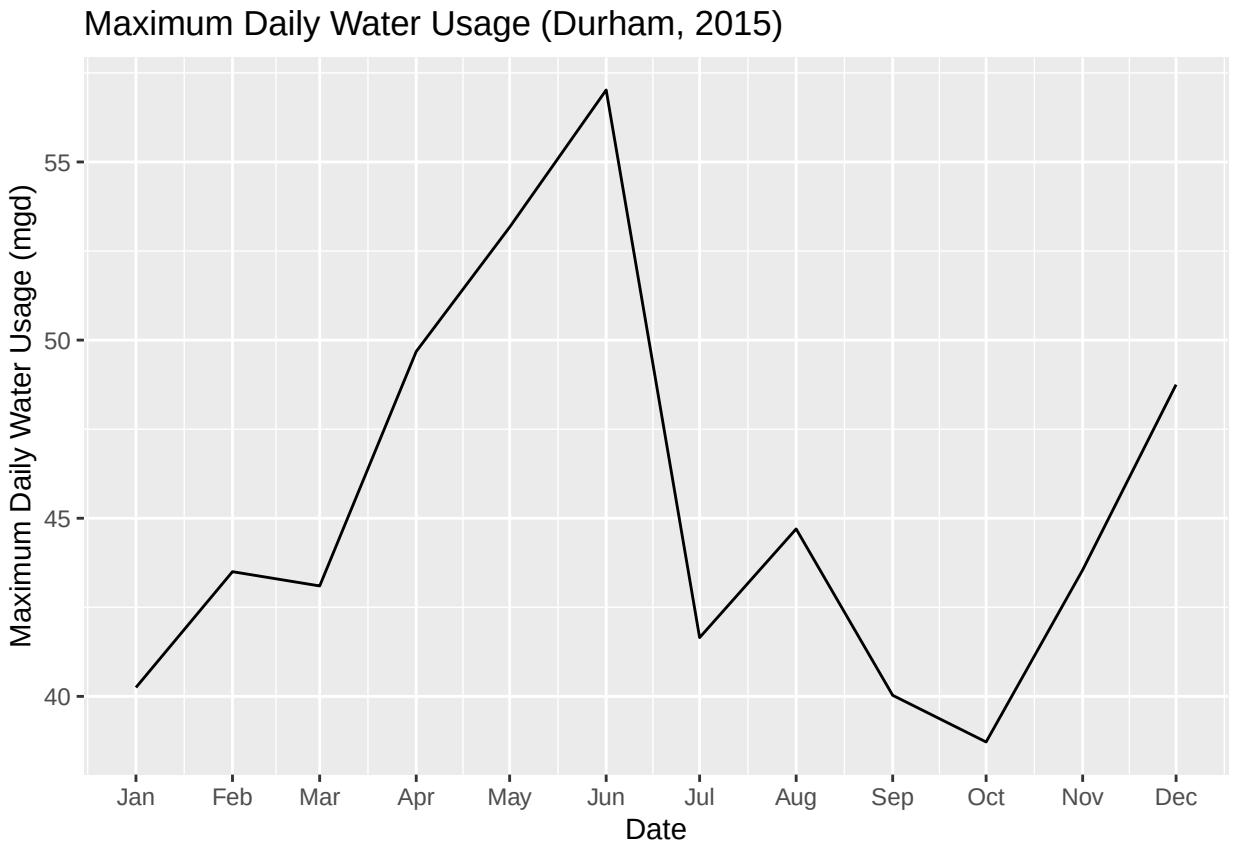
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```
#6.

#Creating scraping function
scraping.function <- function(the_year,the_PWSID){

  # defining variables
  the_base_url <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?'
  #the_PWSID <- '03-32-010'
  #the_year <- 2022
  the_scrape_url <- paste0(the_base_url,'pwsid=',the_PWSID,'&year=',the_year)

  # Getting website contents
  the_website <- read_html(the_scrape_url)

  # setting element address variables
  waterSystemName_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)"
  maxDailyUse_tag <-"th~ td+ td"

  # scrapping data items
  waterSystemName <- the_website %>% html_nodes(waterSystemName_tag) %>% html_text()
```

```
  PSWID <- the_website %>% html_nodes(PWSID_tag) %>% html_text()
  ownership <- the_website %>% html_nodes(ownership_tag) %>% html_text()
  maxDailyUse <- the_website %>% html_nodes(maxDailyUse_tag) %>% html_text()

  # converting to data frame
  df_maxDailyUse <- data.frame("Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                          "Year" = rep(the_year,12),
                          "Maximum.Daily.Use" = as.numeric(maxDailyUse))%>%
      mutate(Water.System.Name = !!waterSystemName,
         PWSID = !!the_PWSID,
         Ownership = !!ownership,
         Date = my(paste(Month,"-",Year)))

  return(df_maxDailyUse)
}
```

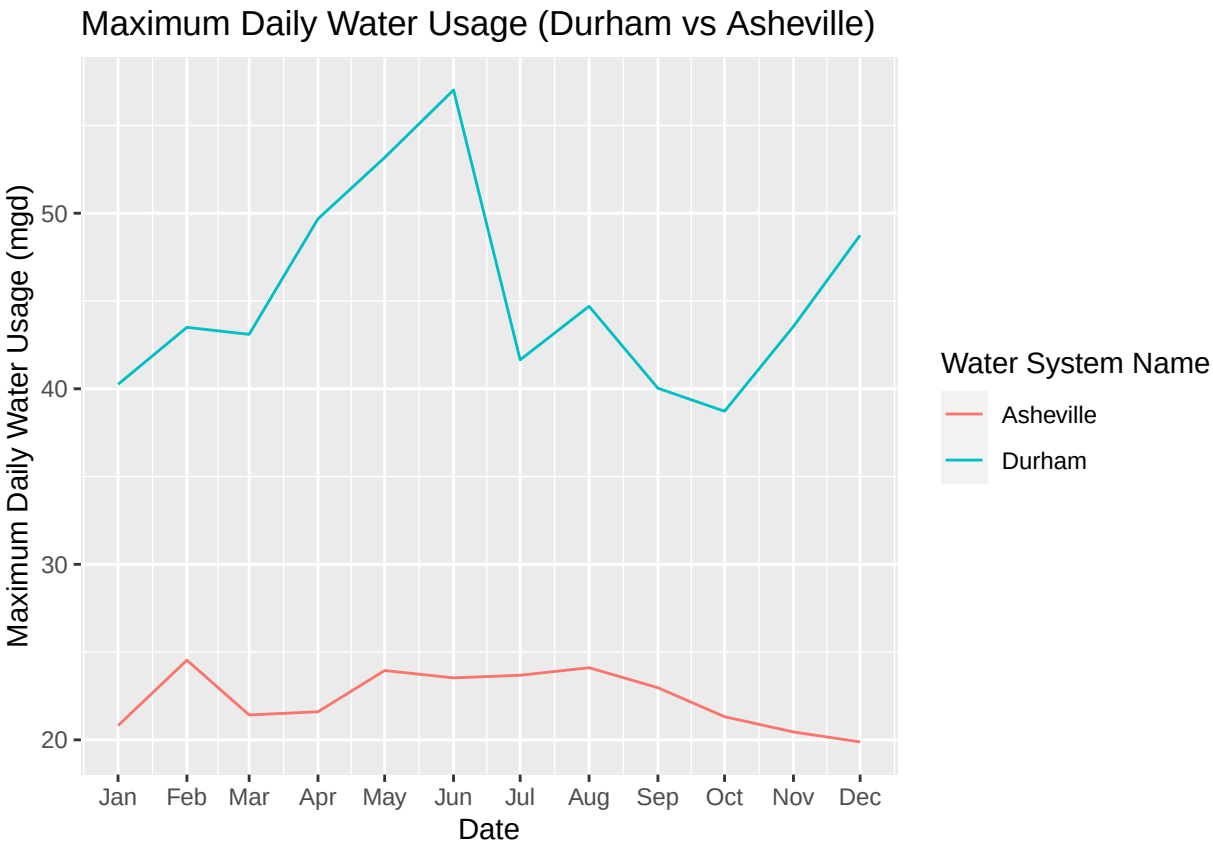
7. Use the function above to extract and plot max daily withdrawals for Durham (PWSID=‘03-32-010’) for each month in 2015

```
#7
# scarping data using function for Durha, 2015
maxDailyUse_Durham_2015_df <- scraping.function(2015,'03-32-010')
view(maxDailyUse_Durham_2015_df)

# plotting data
ggplot(maxDailyUse_Durham_2015_df)+
  geom_line(aes(x=Date,y=Maximum.Daily.Use))+
  labs(title=paste0("Maximum Daily Water Usage (Durham, 2015)"),
     y="Maximum Daily Water Usage (mgd)",
      x="Date")+
  scale_x_date(date_labels = "%b", date_breaks = "1 month")
```



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

# extracting data for Asheville in 2015 using function
maxDailyUse_Asheville_2015_df <- scraping.function(2015,'01-11-010')
view(maxDailyUse_Asheville_2015_df)

# combining data collected with Durham data in #7 - use merge or join functions
maxDailyUSE_Dur_Ash_2015<- rbind(maxDailyUse_Durham_2015_df,maxDailyUse_Asheville_2015_df)

# comparing Asheville and Durham graphically
ggplot(maxDailyUSE_Dur_Ash_2015)+
  geom_line(aes(x=Date,y=Maximum.Daily.Use,color=Water.System.Name))+
  labs(title=paste0("Maximum Daily Water Usage (Durham vs Asheville)"),
     y="Maximum Daily Water Usage (mgd)",
      x="Date",color="Water System Name")+
  scale_x_date(date_labels = "%b", date_breaks = "1 month")
```

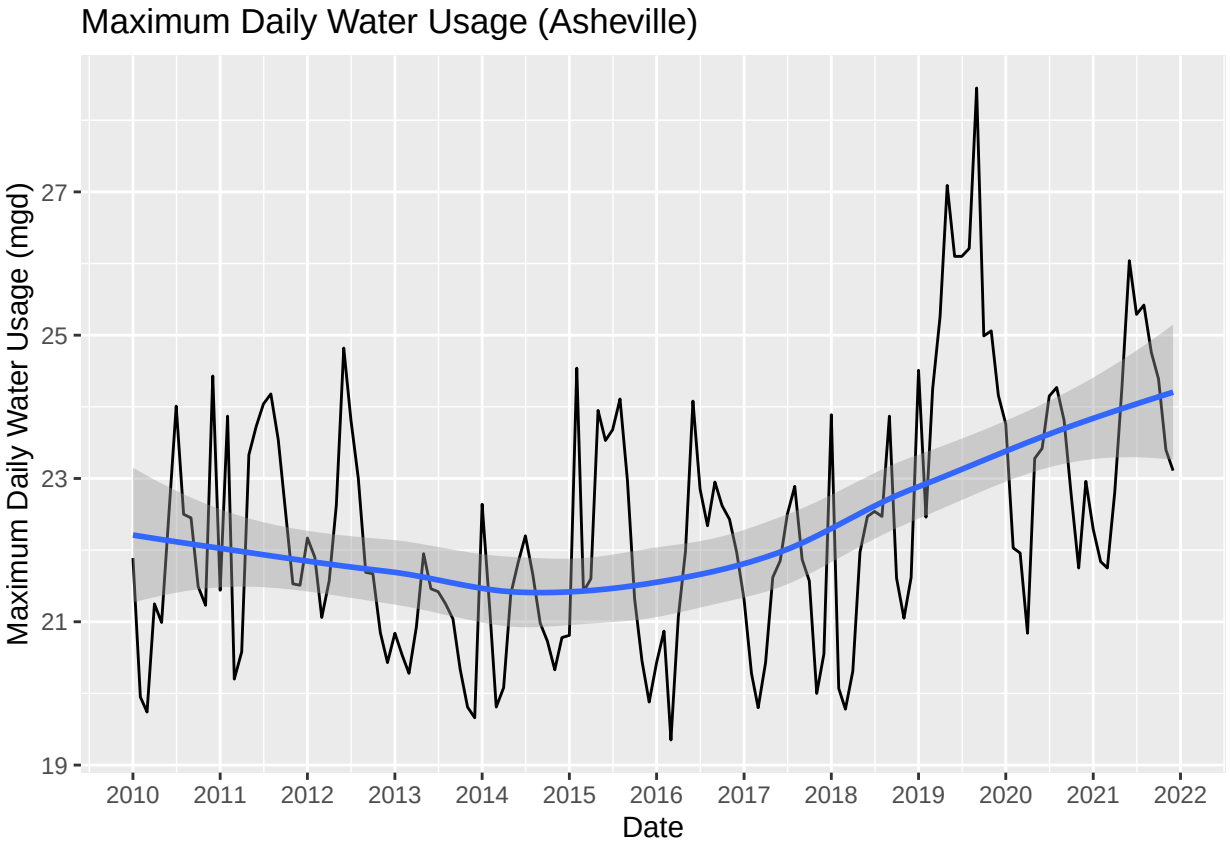


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2021. Add a smoothed line to the plot (method = 'loess').

   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 sequence of years
the_years <- seq(2010,2021)
the_PWSID <- rep("01-11-010",12)

# using map2 with function to retrieve data for desired years
# piping into rbind to bind all data scraped
maxDailyUse_Asheville_2010_2021_df <- map2(the_years,the_PWSID, scraping.function) %>%
  bind_rows()

# plotting Asheville's maximum daily withdrawal for 2010 through 2021
ggplot(maxDailyUse_Asheville_2010_2021_df,aes(x=Date,y=Maximum.Daily.Use))+
  geom_line()+
  geom_smooth(method="loess")+
  labs(title=paste0("Maximum Daily Water Usage (Asheville)"),
     y="Maximum Daily Water Usage (mgd)",
      x="Date")+
  scale_x_date(date_labels = "%Y",date_breaks = "1 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: Yes. Ashville's water usage appears to be trending upwards. We can see from the graph that in 2021 usage was around ~24 mgd, compared to ~21 mgd in 2015 and ~22 mgd in 2010. >