# Assignment 09: Data Scraping

Danlei Zou

## 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

#check working directory
getwd()
```

```
## [1] "/Users/danleizou/EDA-Fall2022"
```

```
#loading necessary packages

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

#setting theme
mytheme <- theme_classic(base_size = 14) +
  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 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
LWSP.webpage <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2021')
LWSP.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
- 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

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

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

```
#scraping for PWSID
pwsid <- LWSP.webpage %>%
  html_nodes("td tr:nth-child(1) td:nth-child(5)") %>%
  html_text()
pwsid
```

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

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

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

```
#scraping for max daily use
max.withdrawals.mgd <- LWSP.webpage %>%
  html_nodes("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.

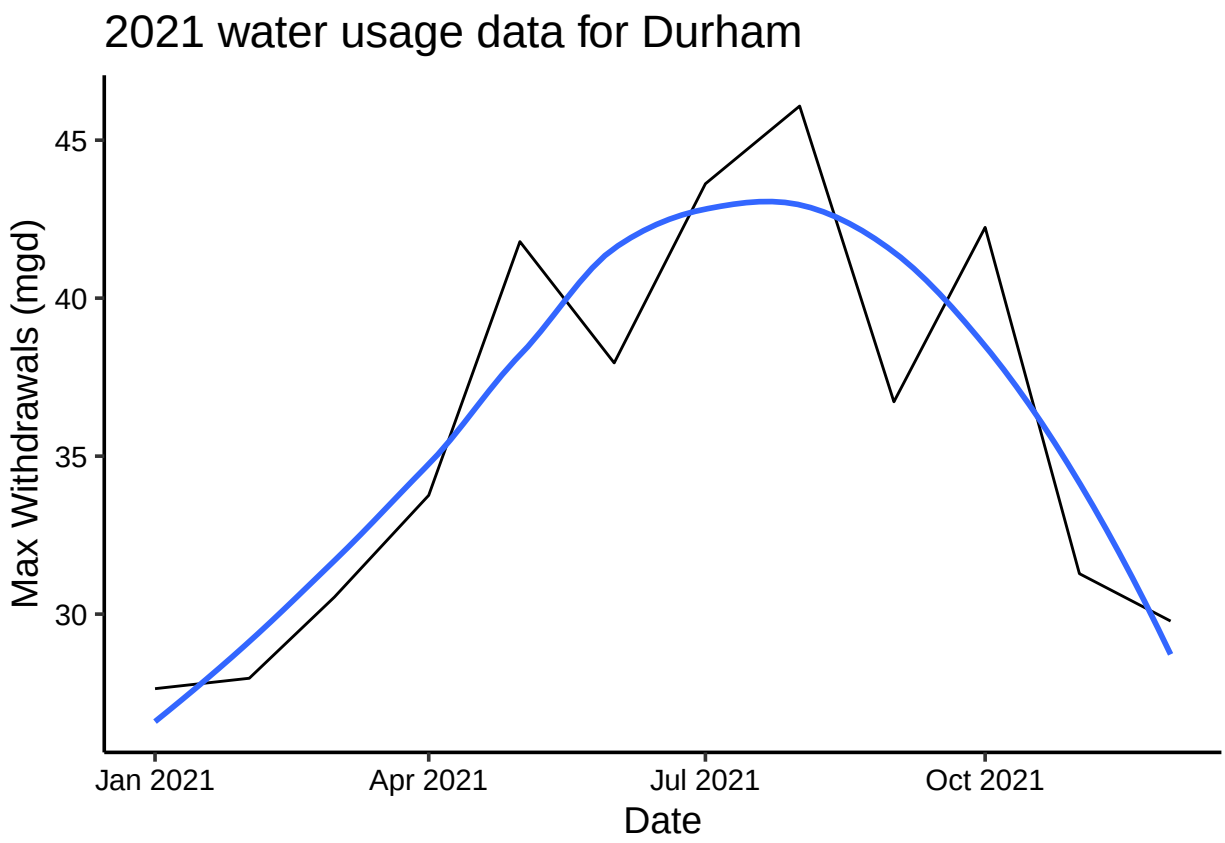
   NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc...

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

```
#4

#creating new month dataframe
month <- c(1,5,9,2,6,10,3,7,11,4,8,12)

#creating dataframe
df.dailywithdrawals <- data.frame("Month" = as.numeric(month),
                                  "Year" = rep(2021,12),
                                  "Ownership" = as.character(ownership),
                                  "Water System Name" = as.character(water.system.name),
                                  "PWSID" = as.character(pwsid),
                                  "Max_Withdrawals_MGD" = as.numeric(max.withdrawals.mgd))
df.dailywithdrawals <- df.dailywithdrawals %>%
  mutate(Date = my(paste(Month, "-", Year)))

#5

#creating line plot of max daily withdrawals across months for 2021 in Durham
ggplot(df.dailywithdrawals, aes(x = Date, y = Max_Withdrawals_MGD)) +
  geom_line() +
```

```
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = paste("2021 water usage data for Durham"),
       y = "Max Withdrawals (mgd)",
       x = "Date")
```

```
## `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 (pwsid) scraped**.

```
#6.

#create scraping function
year <- 2021

scrape.it <- function(the_year, the_pwsid){

  #retrieving website contents
  the_website <- read_html(paste0
                           ('https://www.ncwater.org/WUDC/app/LWSP/report.php?', 'pwsid=', the_pwsid, '&

  #scraping 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()
  max.withdrawals.mgd <- the_website %>%
      html_nodes('th~ td+ td') %>% html_text()

  #converting to dataframe
  df.dailywithdrawals2 <- data.frame("Month" = as.numeric(month),
                                     "Year" = rep(the_year, 12),
                                     "Ownership" = as.character(ownership),
                                     "Water System Name" = as.character(water.system.name),
                                     "PWSID" = as.character(pwsid),
                                     "Max_Withdrawals_MGD" = as.numeric(max.withdrawals.mgd)) %>%
    mutate(Date = my(paste(Month, "-", Year)))

  Sys.sleep(1)

  return(df.dailywithdrawals2)
}
```

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

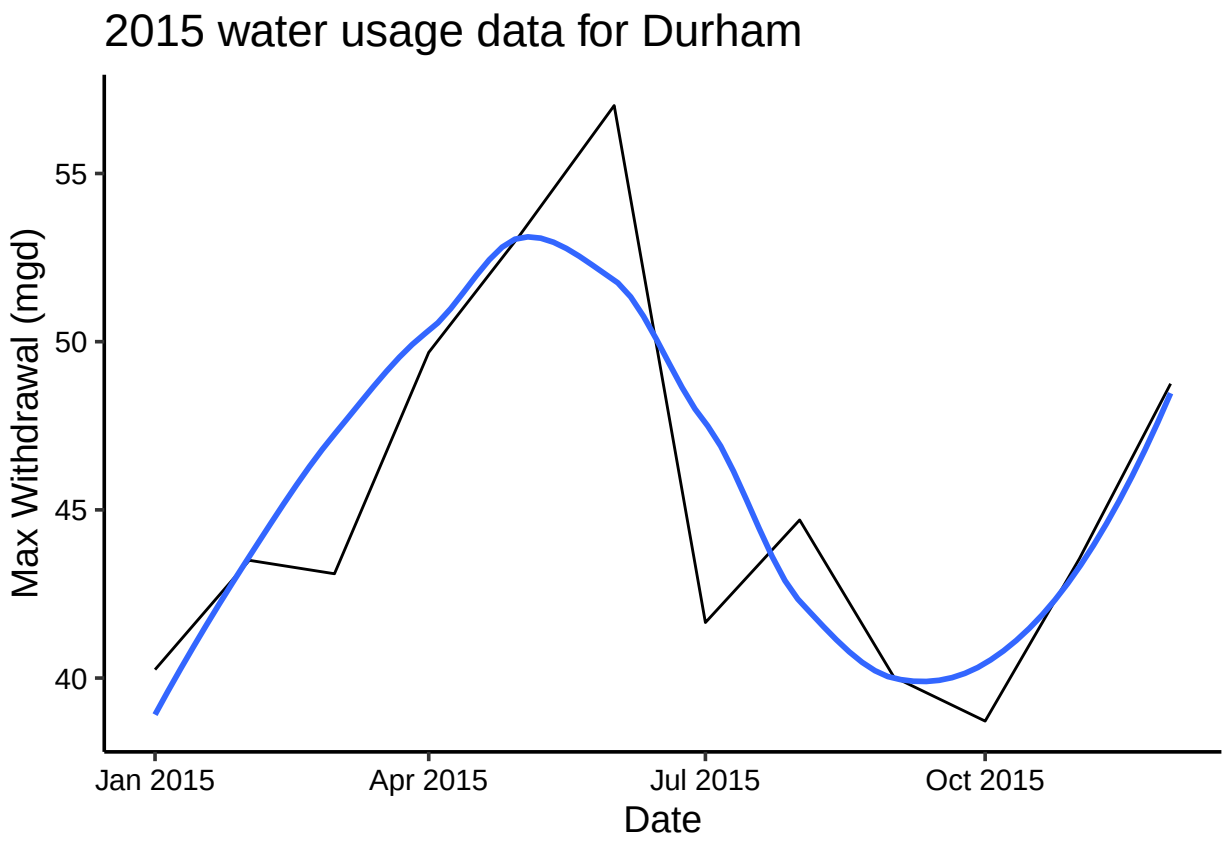
```
#7

#extracting 2015 data for Durham
the_year <- 2015
the_pwsid <- as.character('03-32-010')

#assigning to dataframe
the_dataframe <- data.frame(scrape.it(the_year, the_pwsid))
print(the_dataframe)
```

```
##    Month Year    Ownership Water.System.Name     PWSID Max_Withdrawals_MGD
## 1      1 2015 Municipality            Durham 03-32-010               40.25
## 2      5 2015 Municipality            Durham 03-32-010               53.17
## 3      9 2015 Municipality            Durham 03-32-010               40.03
## 4      2 2015 Municipality            Durham 03-32-010               43.50
## 5      6 2015 Municipality            Durham 03-32-010               57.02
## 6     10 2015 Municipality            Durham 03-32-010               38.72
## 7      3 2015 Municipality            Durham 03-32-010               43.10
## 8      7 2015 Municipality            Durham 03-32-010               41.65
## 9     11 2015 Municipality            Durham 03-32-010               43.55
## 10     4 2015 Municipality            Durham 03-32-010               49.68
## 11     8 2015 Municipality            Durham 03-32-010               44.70
## 12    12 2015 Municipality            Durham 03-32-010               48.75
##          Date
## 1  2015-01-01
## 2  2015-05-01
## 3  2015-09-01
## 4  2015-02-01
```

```
## 5  2015-06-01
## 6  2015-10-01
## 7  2015-03-01
## 8  2015-07-01
## 9  2015-11-01
## 10 2015-04-01
## 11 2015-08-01
## 12 2015-12-01
```

```r
#plotting max daily withdrawals across months for 2015 in Durham
ggplot(the_dataframe, aes(x = Date, y = Max_Withdrawals_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = paste("2015 water usage data for Durham"),
       y = "Max Withdrawal (mgd)",
       x = "Date")
```

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




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.

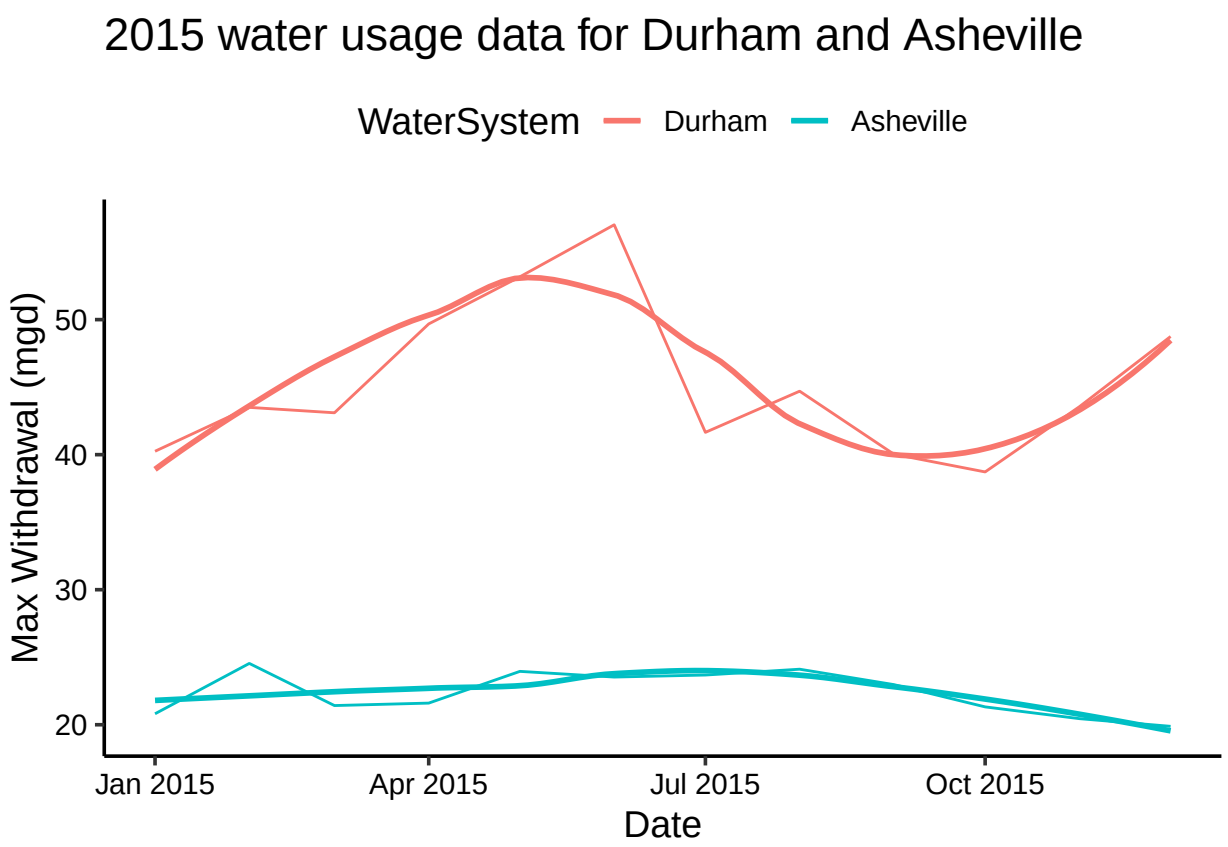
```r
#8

#extracting 2015 data for Asheville
the_year <- 2015
the_pwsid <- as.character('01-11-010')

#assigning to dataframe
the_dataframe_asheville <- data.frame(scrape.it(the_year, the_pwsid))
print(the_dataframe_asheville)
```

```
##    Month Year    Ownership Water.System.Name     PWSID Max_Withdrawals_MGD
## 1      1 2015 Municipality         Asheville 01-11-010               20.81
## 2      5 2015 Municipality         Asheville 01-11-010               23.95
## 3      9 2015 Municipality         Asheville 01-11-010               22.97
## 4      2 2015 Municipality         Asheville 01-11-010               24.54
## 5      6 2015 Municipality         Asheville 01-11-010               23.53
## 6     10 2015 Municipality         Asheville 01-11-010               21.32
## 7      3 2015 Municipality         Asheville 01-11-010               21.42
## 8      7 2015 Municipality         Asheville 01-11-010               23.68
## 9     11 2015 Municipality         Asheville 01-11-010               20.45
## 10     4 2015 Municipality         Asheville 01-11-010               21.60
## 11     8 2015 Municipality         Asheville 01-11-010               24.11
## 12    12 2015 Municipality         Asheville 01-11-010               19.88
##          Date
## 1  2015-01-01
## 2  2015-05-01
## 3  2015-09-01
## 4  2015-02-01
## 5  2015-06-01
## 6  2015-10-01
## 7  2015-03-01
## 8  2015-07-01
## 9  2015-11-01
## 10 2015-04-01
## 11 2015-08-01
## 12 2015-12-01
```

```r
#creating joined dataframe of Asheville and Durham 2015 data
withdrawals_joined <- merge(x = the_dataframe,
                            y = the_dataframe_asheville,
                            by = c("Date", "Month", "Year"))

#creating plot of joined dataframe
ggplot_joined <- withdrawals_joined %>%
  gather(WaterSystem, City, Max_Withdrawals_MGD.x, Max_Withdrawals_MGD.y) %>%
  ggplot(aes(x = Date, y = City, colour = WaterSystem)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  scale_shape_discrete(labels = c("Durham", "Asheville")) +
  scale_colour_discrete(labels = c("Durham", "Asheville")) +
  labs(title = paste("2015 water usage data for Durham and Asheville"),
       y = "Max Withdrawal (mgd)",
       x = "Date")
```

```
print(ggplot_joined)
```

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



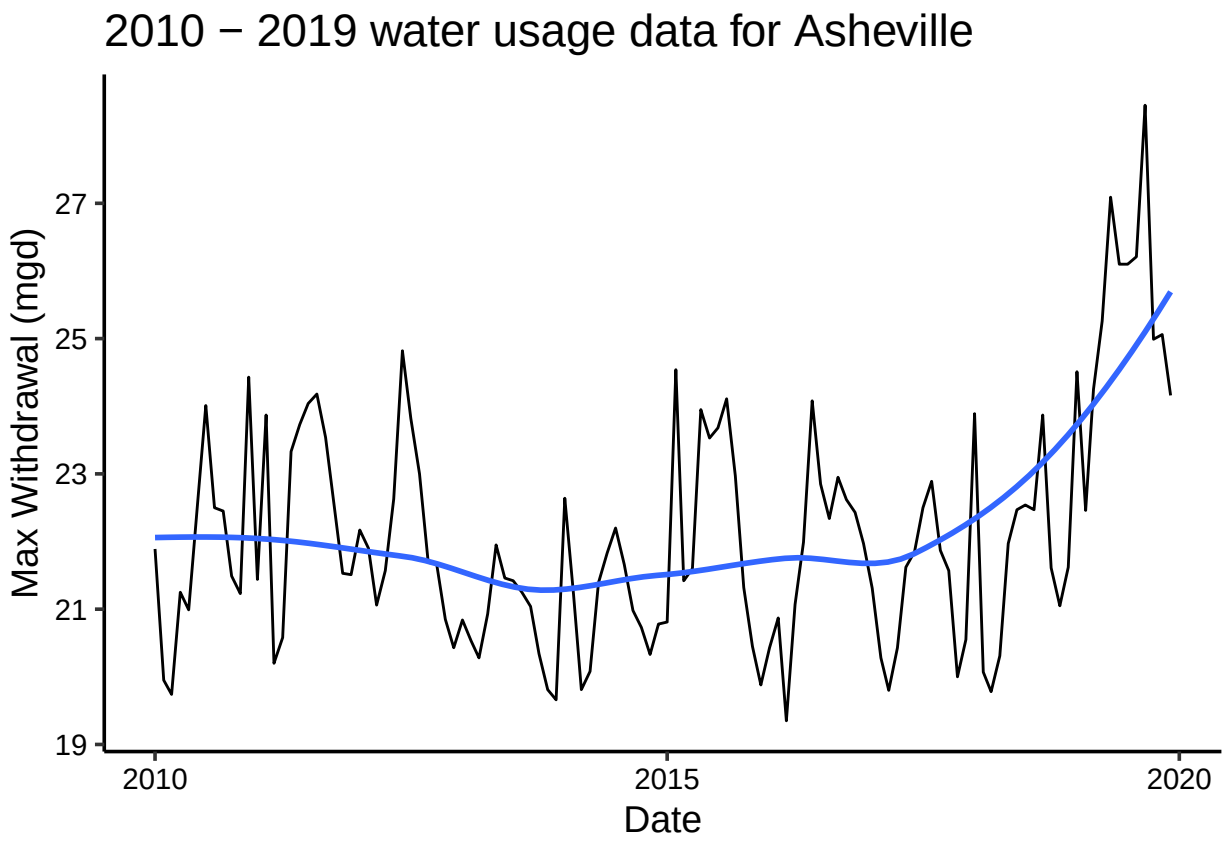

9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2010 thru 2019.Add a smoothed line to the plot.

   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

#extracting 2010-2019 data for Asheville
the_years <- rep(2010:2019)
the_pwsid <- as.character('01-11-010')

#applying scrape function
asheville_df2 <- lapply(X = the_years,
                        FUN = scrape.it,
                        the_pwsid = the_pwsid)

#conflate into single dataframe
```

```r
asheville_df_20102019 <- bind_rows(asheville_df2)

#creating a plot of  daily max withdrawals from 2010-2019 in Asheville
ggplot(asheville_df_20102019, aes(x = Date, y = Max_Withdrawals_MGD)) +
  geom_line() +
  geom_smooth(method = "loess", se = FALSE) +
  labs(title = paste("2010 - 2019 water usage data for Asheville"),
       y = "Max 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?

ANSWER: Yes, Asheville has a trend in water usage over time by looking at the plot. The plot shows an increase in water usage from 2015 to 2019.