Assignment 09: Data Scraping

Abhay V Rao

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
getwd()
```

```
## [1] "C:/Users/av241/Documents/Environmental_Data_Analytics_2022/Assignments"
```

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

mytheme <- theme_gray(base_size = 10) +
  theme(axis.text = element_text(color = "darkgrey"),
        legend.position = "bottom")

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
the_website <- read_html('https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2020')
the_website
```

```
## {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:
- 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
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()
```

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.)

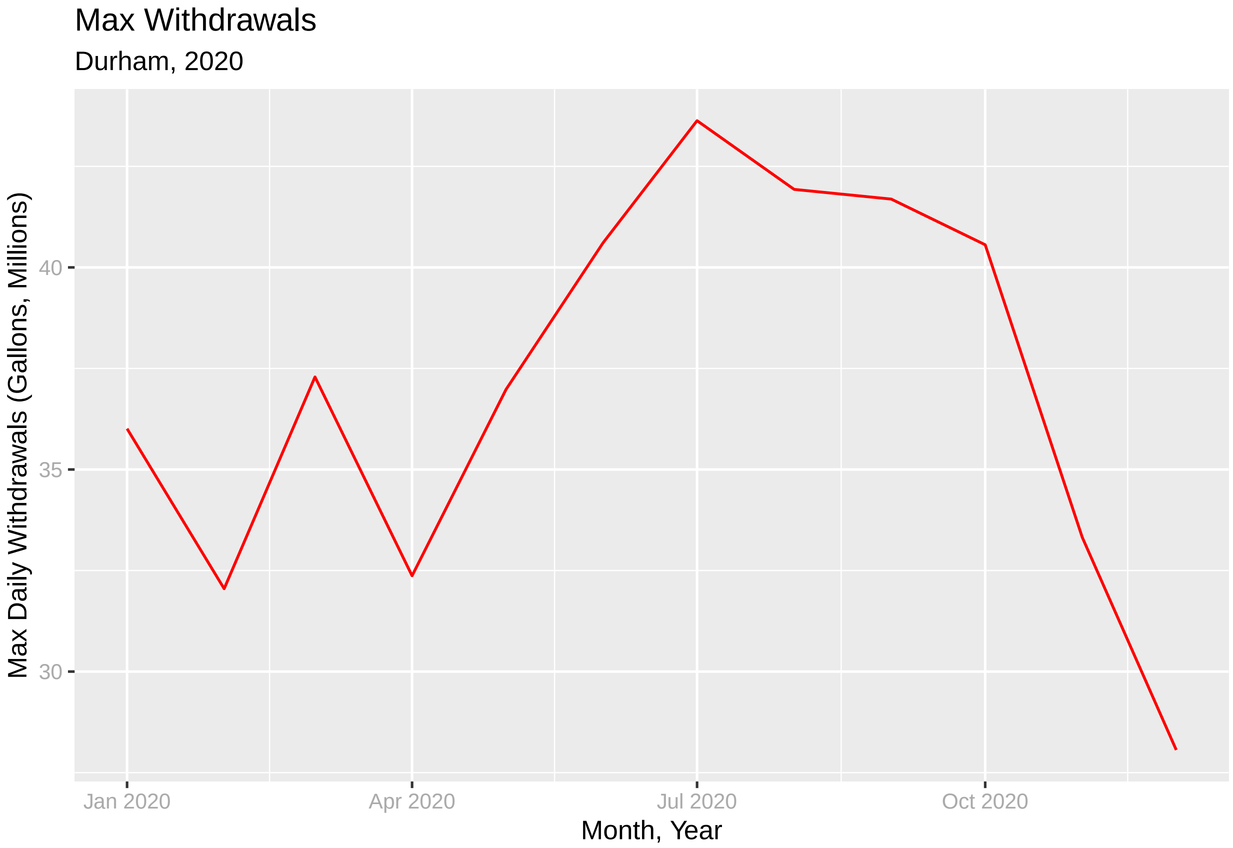
   TIP: Use `rep()` to repeat a value when creating a dataframe.

   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

the_dataframe <- data.frame(
  "Water System Name"= rep(water.system.name),
  "PWSID"=rep(pwsid),
  "Ownership"=rep(ownership),
  "Year"=rep(2020,12),
  "Month"=(month(c(1,5,9,2,6,10,3,7,11,4,8,12))),
  "Max.Withdrawals"=as.numeric(max.withdrawals.mgd))%>%
  mutate(Date=my(paste(Month,"-",Year)))
```

```
#5

ggplot(the_dataframe, aes(x=Date, y=Max.Withdrawals))+
  geom_line(color="red")+
  labs(title='Max Withdrawals', subtitle= "Durham, 2020",
       y="Max Daily Withdrawals (Gallons, Millions)", x="Month, Year")
```



base_url <- ‘https://www.ncwater.org/WUDC/app/LWSP/report.php?’

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.
base_url <- 'https://www.ncwater.org/WUDC/app/LWSP/report.php?'
the_pwsid <- '03-32-010'
the_year <- 2020

url.name <- paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?', 'pwsid=',
                   the_pwsid, '&year=', the_year)
url.name
```

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

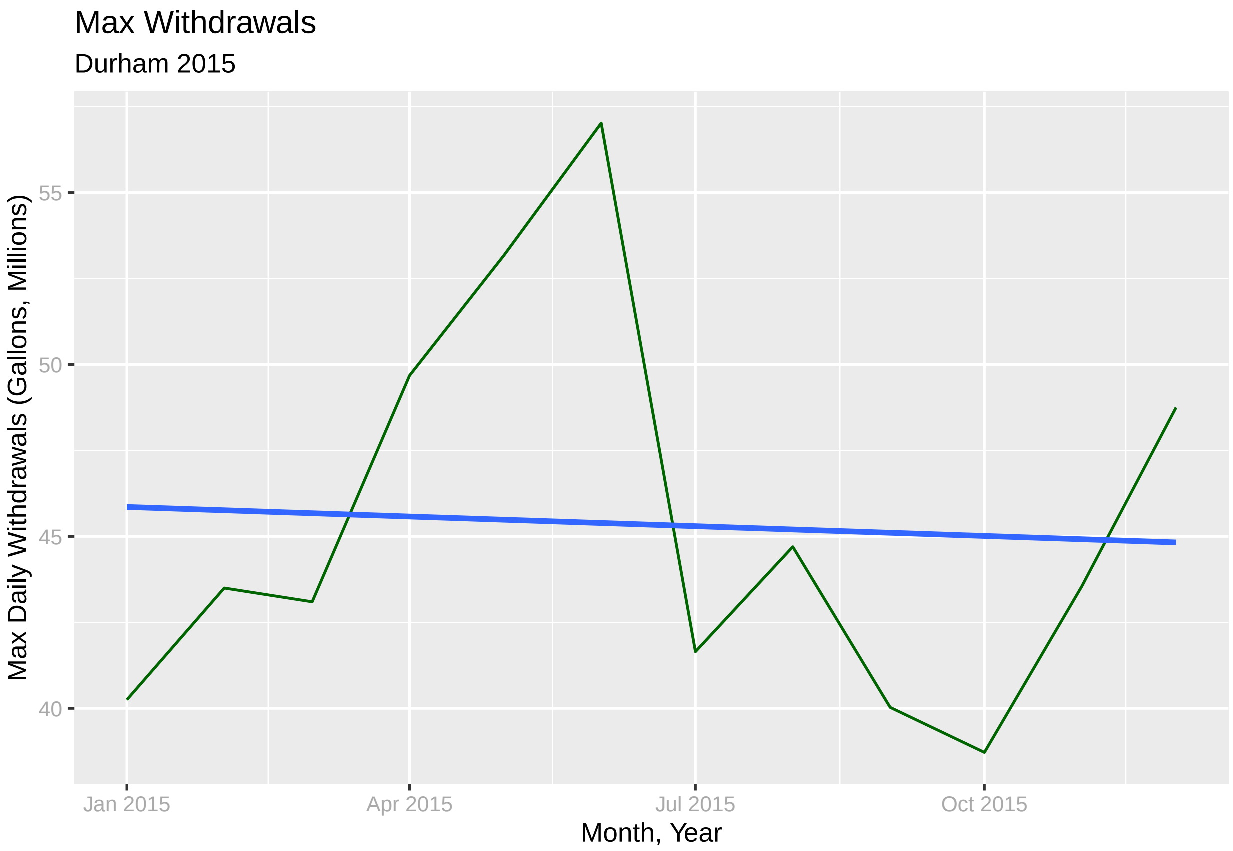
```
the_website <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?',
                                'pwsid=', the_pwsid, '&year=', the_year))

water.system.name.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'

water.system.name <- the_website %>% html_nodes(water.system.name.tag) %>% html_text()
pwsid <- the_website %>% html_nodes(pwsid.tag) %>% html_text()
ownership <- the_website %>% html_nodes(ownership.tag) %>% html_text()
max.withdrawals.mgd <- the_website %>% html_nodes(max.withdrawals.mgd.tag) %>% html_text()

the_dataframe <- data.frame(
  "Year"=rep(the_year,12),
  "Month"=(month(c(1,5,9,2,6,10,3,7,11,4,8,12))),
  "Max.Withdrawals"=as.numeric(max.withdrawals.mgd))%>%
  mutate(Date=my(paste(Month,"-",Year)),
         Water.System.Name = !!water.system.name,
         PWSID =!!pwsid,
         Ownership =!!ownership)
```

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

```
#7
scrape.it <- function(the_year,the_pwsid){

the_website <- read_html(paste0('https://www.ncwater.org/WUDC/app/LWSP/report.php?',
                                'pwsid=', the_pwsid, '&year=', the_year))

water.system.name.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'

water.system.name <- the_website %>% html_nodes(water.system.name.tag) %>% html_text()
pwsid <- the_website %>% html_nodes(pwsid.tag) %>% html_text()
ownership <- the_website %>% html_nodes(ownership.tag) %>% html_text()
max.withdrawals.mgd <- the_website %>% html_nodes(max.withdrawals.mgd.tag) %>% html_text()

the_dataframe <- data.frame(
  "Year"=rep(the_year,12),
  "Month"=(month(c(1,5,9,2,6,10,3,7,11,4,8,12))),
  "Max.Withdrawals"=as.numeric(max.withdrawals.mgd))%>%
  mutate(Date=my(paste(Month,"-",Year)),
         Water.System.Name = !!water.system.name,
         PWSID =!!pwsid,
         Ownership =!!ownership)

}
# Set up a scrape function, then the DF for Durham

the_df1 <- scrape.it(2015, '03-32-010')

ggplot(the_df1, aes(x=Date, y=Max.Withdrawals))+
  geom_line(color="darkgreen")+
  geom_smooth(method="lm" ,se=FALSE) +
  labs(title='Max Withdrawals',
       subtitle= paste(the_df1$Water.System.Name,the_df1$Year),
       y="Max Daily Withdrawals (Gallons, Millions)", x="Month, Year")
```

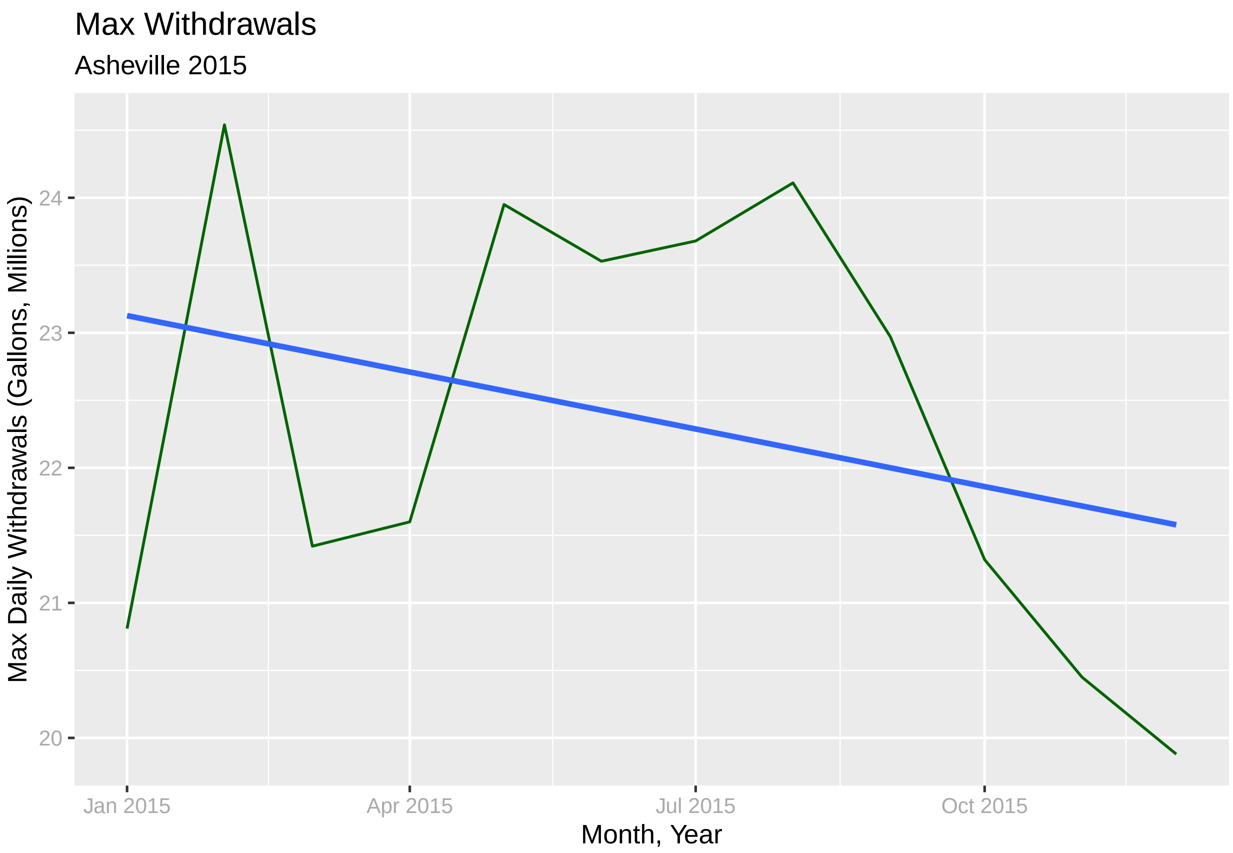
```
## `geom_smooth()` using formula 'y ~ x'
```



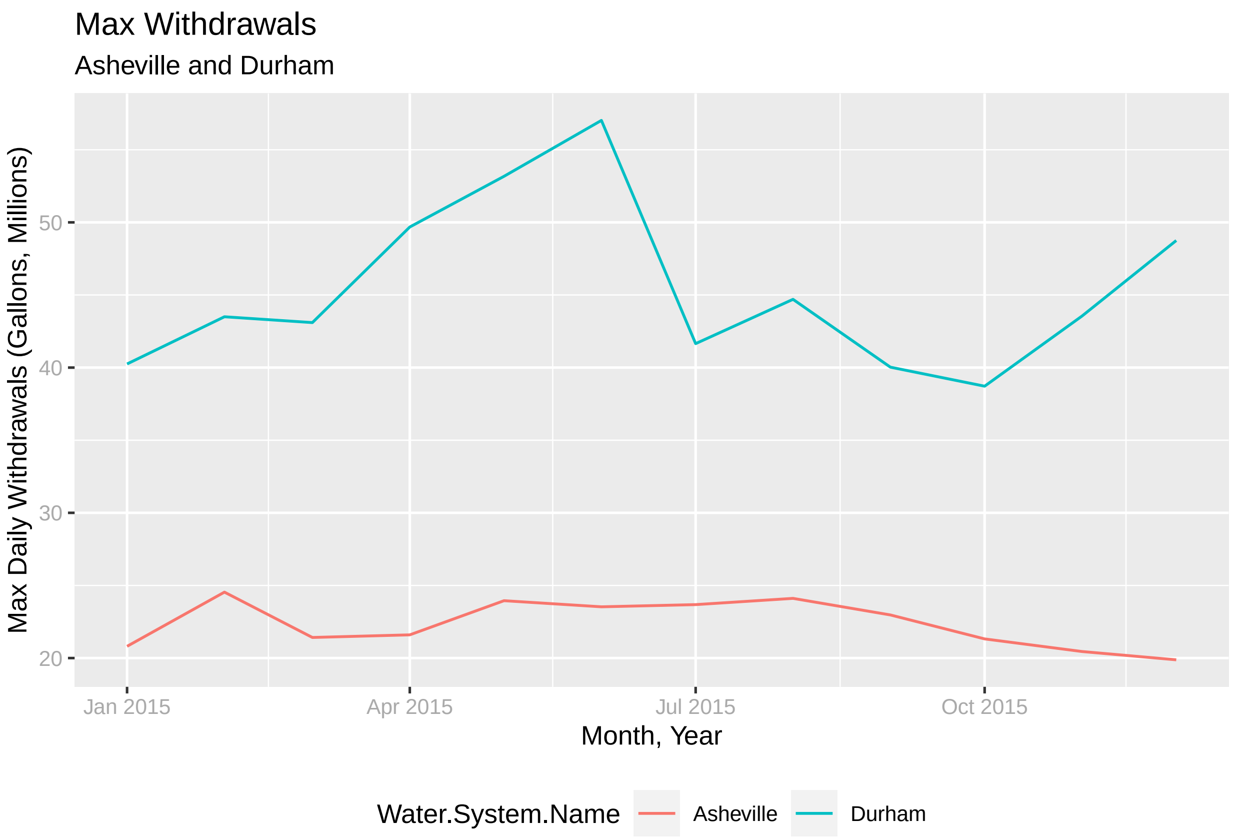
8. Use the function above to extract data for Asheville (PWSID = 01-11-010) in 2015. Combine this data with the Durham data collected above and create a plot that compares the Asheville to Durham's water withdrawals.

```
#8

the_df2 <- scrape.it(2015, '01-11-010')

ggplot(the_df2, aes(x=Date, y=Max.Withdrawals))+
  geom_line(color="darkgreen")+
  geom_smooth(method="lm",se=FALSE) +
  labs(title='Max Withdrawals',
       subtitle= paste(the_df2$Water.System.Name,the_df2$Year),
       y="Max Daily Withdrawals (Gallons, Millions)", x="Month, Year")
```

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



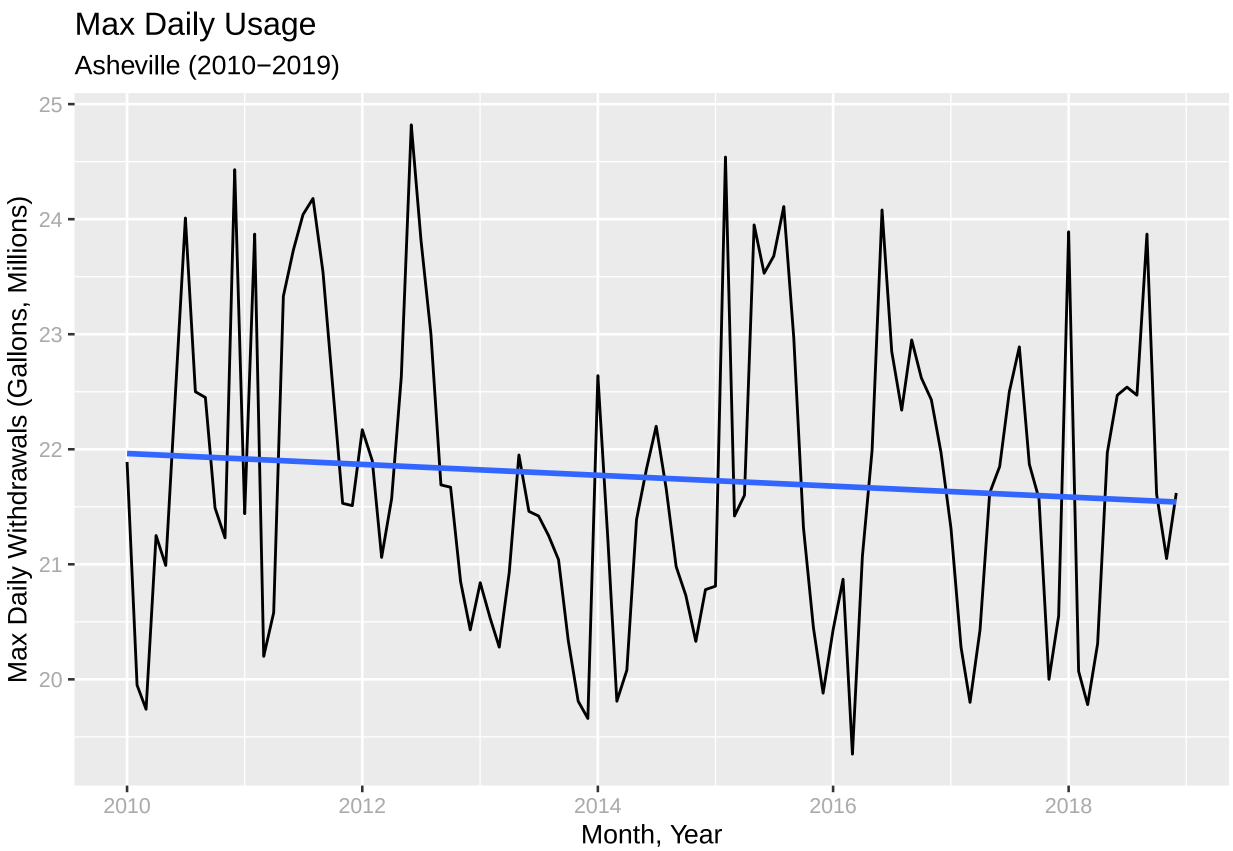
```
df_combined <- bind_rows(the_df1, the_df2) %>%
  group_by(Water.System.Name, Month)

plot_combined <-
  ggplot(df_combined, aes(x = Date, y = Max.Withdrawals, color=Water.System.Name)) +
  geom_line()+
  labs(title='Max Withdrawals', subtitle= "Asheville and Durham",
       y="Max Daily Withdrawals (Gallons, Millions)", x="Month, Year")

print(plot_combined)
```



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.

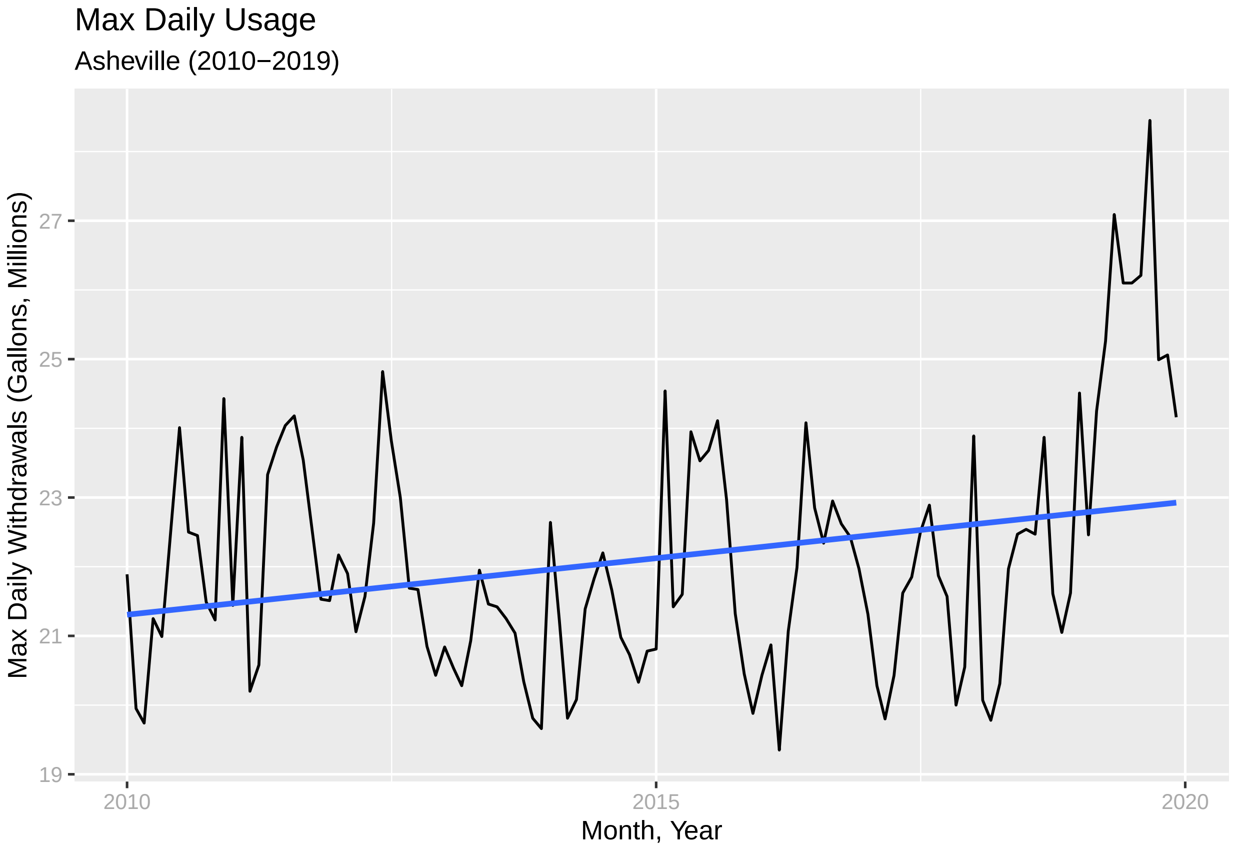
```
#9
the_years = rep(2010:2018)
my_pwsid = '01-11-010'

the_dfs <- map(the_years,scrape.it,the_pwsid=my_pwsid)

#Conflating returned dataframes into a single dataframe
the_df <- bind_rows(the_dfs)

#Plotting
ggplot(the_df,aes(x=Date,y=Max.Withdrawals)) +
  geom_line() +
  geom_smooth(method=lm,se=FALSE)+
  labs(title='Max Daily Usage',
       subtitle= "Asheville (2010-2019)",y="Max Daily Withdrawals (Gallons, Millions)",
       x="Month, Year")
```

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



```
#Adding the 2010:2019 range includes the year 2020.
the_years = rep(2010:2019)
my_pwsid = '01-11-010'

the_dfs <- map(the_years,scrape.it,the_pwsid=my_pwsid)

#Conflating returned dataframes into a single dataframe
the_df <- bind_rows(the_dfs)

#Plotting
ggplot(the_df,aes(x=Date,y=Max.Withdrawals)) +
  geom_line() +
  geom_smooth(method=lm,se=FALSE)+
  labs(title='Max Daily Usage',
       subtitle= "Asheville (2010-2019)",y="Max Daily Withdrawals (Gallons, Millions)",
       x="Month, 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?

An LM line indicates a downward trend in max daily water usage in Asheville until 2019. The trend reverses if the next year is included.