Assignment 10: Data Scraping

Erin Ansbro

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 loading packages
#install.packages("rvest")
library(tidyverse)
library(rvest)
library(lubridate)

#finding wd
getwd()
```

```
## [1] "C:/Users/eka19/OneDrive - Duke University/Documents/872/EDE_Fall2023"
```

```
mytheme <- theme_classic(base_size = 12) +
  theme(axis.text = element_text(color = "navy"),
        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 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 Fetching the web resources from the URL
webpage <- read_html(
  'https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022')
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
- 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
water_system_name <- webpage %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
water_system_name
```

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

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

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

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

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

```
maximum_day_use <- webpage %>%
  html_nodes("th~ td+ td") %>%
  html_text()
maximum_day_use
```

```
##  [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

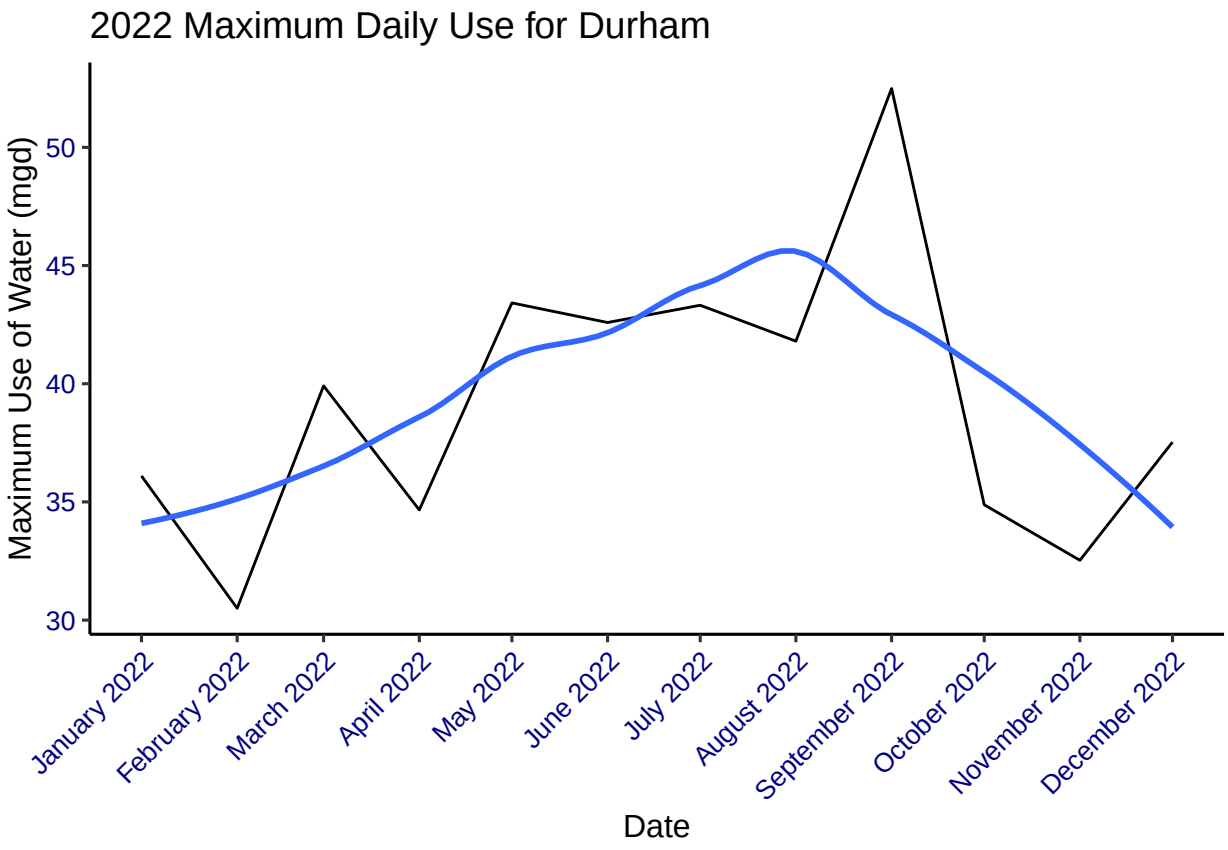
```
#4
water_df <- data.frame("Year"=rep(2022, 12),
                 "Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                 "Maximum_Day_Use" = as.numeric(maximum_day_use))

water_df <- water_df %>%
  mutate(Ownership = !!ownership,
         Water_System_Name = !!water_system_name,
         PWSID = !!pwsid,
         Date = my(paste(Month,"-",Year)))

#5 Plot
dfplot<- ggplot(water_df, aes(x=Date, y=Maximum_Day_Use))+
  geom_line()+
  geom_smooth(method="loess", se=FALSE)+
  labs(title=paste("2022 Maximum Daily Use for",water_system_name),
       y="Maximum Use of Water (mgd)",
       x="Date")+
  scale_x_date(date_breaks="1 month",date_labels = "%B %Y")+
  theme(axis.text.x=element_text(angle = 45, hjust = 1))

print(dfplot)
```

```
## `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. Constructing the scraping web address
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)
#print(the_scrape_url)

#pasting here to make sure url is correct
#https://www.ncwater.org/WUDC/app/LWSP/report.php?pwsid=03-32-010&year=2022


#Creating scraping function
scrapefunction <- function(the_year, the_pwsid){

#Retrieving the website contents
the_website <- read_html(paste0(the_base_url, 'pwsid=', the_pwsid, '&year=', the_year))

print(the_website)
```

```
#Setting the element address variables
the_pwsid_tag <- 'td tr:nth-child(1) td:nth-child(5)'
the_water_system_name_tag <- 'div+ table tr:nth-child(1) td:nth-child(2)'
the_maximum_day_usage_tag <- 'th~ td+ td'
the_ownership_tag <- 'div+ table tr:nth-child(2) td:nth-child(4)'

#Scraping the data items
the_pwsid_scrape <-
  the_website %>% html_nodes(the_pwsid_tag) %>% html_text()
the_watersystem_scrape <-
  the_website %>% html_nodes(the_water_system_name_tag) %>% html_text()
the_maximum_scrape <-
  the_website %>% html_nodes(the_maximum_day_usage_tag) %>% html_text()
the_owner_scrape <-
  the_website %>% html_nodes(the_ownership_tag) %>% html_text()

#Constructing a dataframe from the scraped data
df_scrape2 <- data.frame("Month" = c(1,5,9,2,6,10,3,7,11,4,8,12),
                         "Year" = rep(the_year,12),
                         "Maximum_Day_Usage" = as.numeric(the_maximum_scrape))%>%
  mutate(Water_System_Name = !!the_watersystem_scrape,
         Owner_Name = !!the_owner_scrape,
         PWSID = !!the_pwsid_scrape,
         Date = my(paste(Month,"-",Year)))

  #Pause for a moment
  Sys.sleep(1) #uncomment this if you are doing bulk scraping!

  #Return the dataframe
  return(df_scrape2)
}
```

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

```
#7
final_df <- scrapefunction(2015,'03-32-010')
```

```
## {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= ...
```

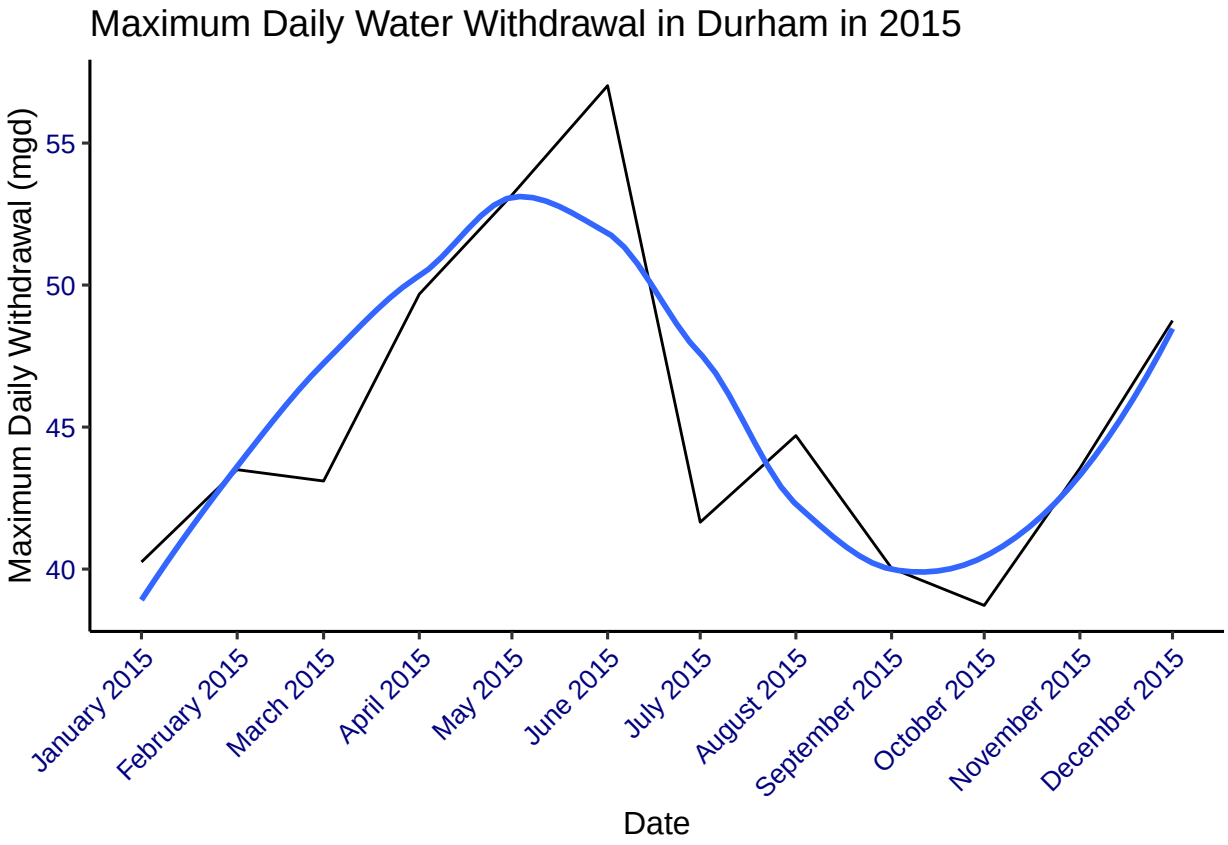
```
view(final_df)

final_df_plot <- ggplot(final_df, aes(x=Date, y=Maximum_Day_Usage))+
                          geom_line()+
                          geom_smooth(method="loess", se=FALSE)+
                          labs(title="Maximum Daily Water Withdrawal in Durham in 2015",
                               y="Maximum Daily Withdrawal (mgd)",
                               x="Date")+
  scale_x_date(date_breaks="1 month",date_labels = "%B %Y")+
```

```
  theme(axis.text.x=element_text(angle = 45, hjust = 1))

print(final_df_plot)
```

```
## `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.

```
#8
Asheville_df <- scrapefunction(2015,'01-11-010')
```

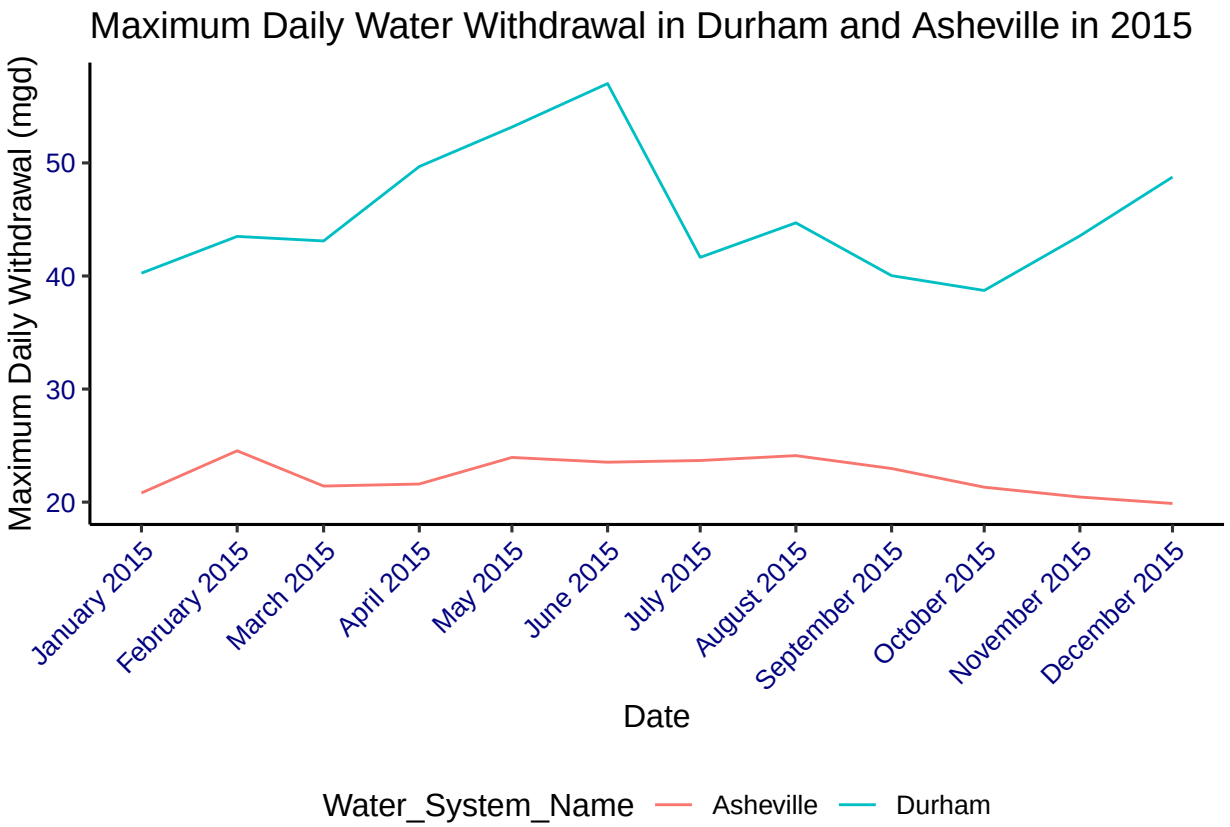
```
## {html_document}
## <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
## [1] <head>\n<title>DWR :: Local Water Supply Planning</title>\n<meta http-equ ...
## [2] <body id="plan">\r\n<!--<div id="division-header">\r\n<a name="top" href= ...
```

```
view(Asheville_df)

join_the_df <- bind_rows(final_df,Asheville_df)

ggplot(join_the_df, aes(x=Date, y=Maximum_Day_Usage, color=Water_System_Name))+
```

```
          geom_line()+
          labs(title="Maximum Daily Water Withdrawal in Durham and Asheville in 2015",
                    y="Maximum Daily Withdrawal (mgd)",
                   x="Date")+
 scale_x_date(date_breaks="1 month",date_labels = "%B %Y")+
 theme(axis.text.x=element_text(angle = 45, hjust = 1))
```



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 a list of the year we want, the same length as the vector above
the_years <- c(2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021)

#"Map" the scrapefunction function to retrieve data for all these
Asheville_df_years <- map2(the_years,'01-11-010', scrapefunction)
```

```
## {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 ...
```

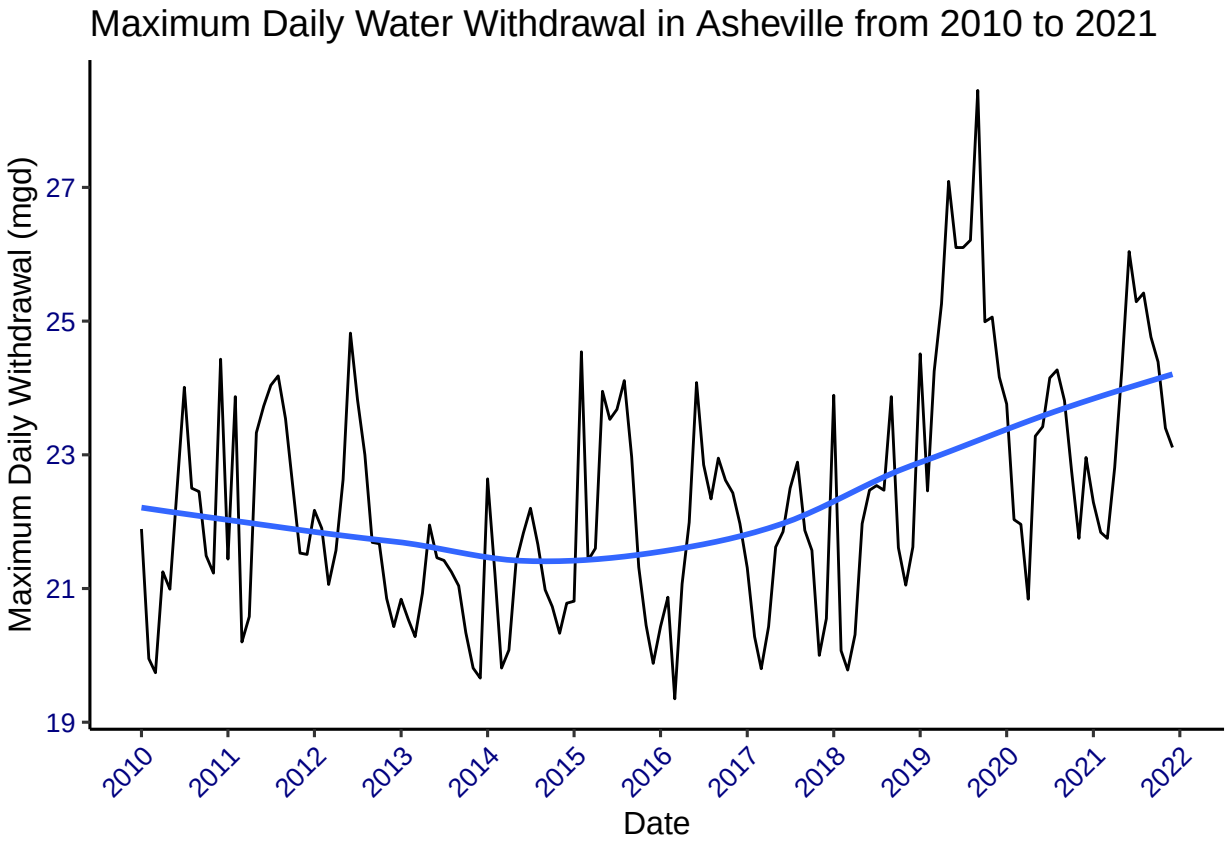
```
## [2] <body id="plan">\r\n<!--<div id="division-header">\r\n<a name="top" href= ...
## {html_document}
## <html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
## [1] <head>\n<title>DWR :: Local Water Supply Planning</title>\n<meta http-equ ...
## [2] <body id="plan">\r\n<!--<div id="division-header">\r\n<a name="top" href= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
## {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= ...
```

```r
#Conflate the returned list of dataframes into a single one
Asheville_df_year <- bind_rows(Asheville_df_years)

#Plot
ggplot(Asheville_df_year, aes(x=Date, y=Maximum_Day_Usage))+
          geom_line()+
          geom_smooth(method="loess", se=FALSE)+
          labs(title="Maximum Daily Water Withdrawal in Asheville from 2010 to 2021",
```

```
                    y="Maximum Daily Withdrawal (mgd)", x="Date")+
  scale_x_date(date_breaks="1 year",date_labels = "%Y")+
 theme(axis.text.x=element_text(angle = 45, hjust = 1))
```

```
## `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: Water usage decreased slightly from 2010 to 2015, and from 2015 onward has steadily increased. >