Assignment 10: Data Scraping

Nargis Taraki

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

```r
#1

library(tidyverse)
#install.packages("rvest")
library(rvest)
library(lubridate)
library(ggplot2)
library(dplyr)
library(stringr)
library(purrr)
getwd()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

2. We will be scraping data from the NC DEQs Local Water Supply Planning website, specifically the Durham's 2023 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=2023

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

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

```
## {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 Scraping the water system name
the_water_system <- NCDEQ_Web %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)") %>%
  html_text()
the_water_system
```

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

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

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

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

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

```r
maximum_day_use <- NCDEQ_Web %>%
  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.

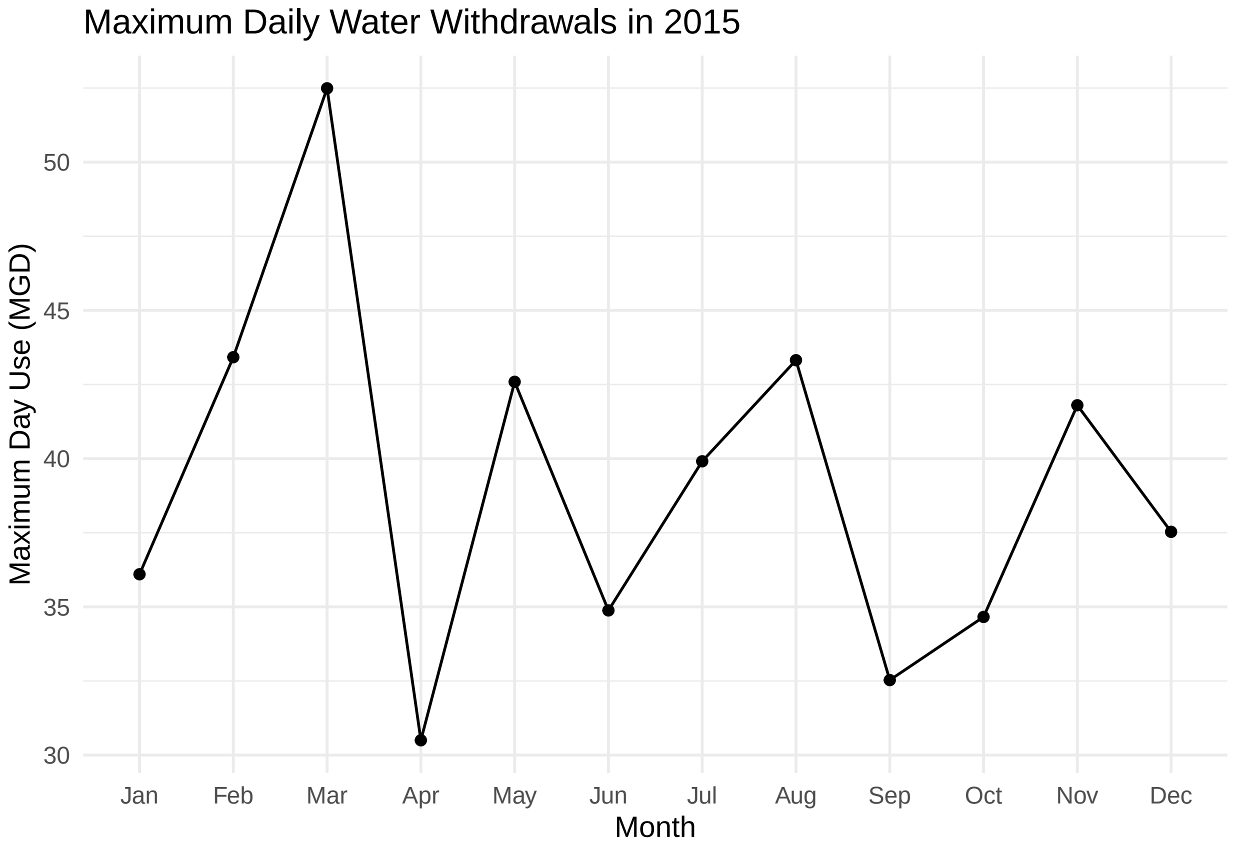
   NOTE: It's likely you won't be able to scrape the monthly widthrawal data in chronological order. You can overcome this by creating a month column manually assigning values in the order the data are scraped: "Jan", "May", "Sept", "Feb", etc… Or, you could scrape month values from the web page…

5. Create a line plot of the maximum daily withdrawals across the months for 2023, making sure, the months are presented in proper sequence.

```r
#4 # Converting maximum daily use values into numeric format

months <- c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec")
years <- rep(2015, times = 12)  # Change the year here if needed

# Creating vectors matching the length of months for all water system attributes
water_system_name_vector <- rep(the_water_system, times = 12)
PWSID_vector <- rep(PWSID, times = 12)
ownership_vector <- rep(ownership, times = 12)

maximum_day_use_vector <- as.numeric(maximum_day_use)

# Creating the dataframe
water_data <- data.frame(
 the_water_system = water_system_name_vector,
  PWSID = PWSID_vector,
  Ownership = ownership_vector,
  Month = months,
  Year = years,
  MaxDayUseMGD = maximum_day_use_vector
)

water_data$Date <- make_date(year = water_data$Year, month = match(water_data$Month, months), day = 1)

# Checking the resulting dataframe
print(water_data)
```

```
##    the_water_system     PWSID    Ownership Month Year MaxDayUseMGD       Date
## 1            Durham 03-32-010 Municipality   Jan 2015        36.10 2015-01-01
## 2            Durham 03-32-010 Municipality   Feb 2015        43.42 2015-02-01
## 3            Durham 03-32-010 Municipality   Mar 2015        52.49 2015-03-01
## 4            Durham 03-32-010 Municipality   Apr 2015        30.50 2015-04-01
## 5            Durham 03-32-010 Municipality   May 2015        42.59 2015-05-01
## 6             Durham 03-32-010 Municipality   Jun 2015        34.88 2015-06-01
## 7             Durham 03-32-010 Municipality   Jul 2015        39.91 2015-07-01
## 8             Durham 03-32-010 Municipality   Aug 2015        43.32 2015-08-01
## 9             Durham 03-32-010 Municipality   Sep 2015        32.53 2015-09-01
## 10            Durham 03-32-010 Municipality   Oct 2015        34.66 2015-10-01
## 11            Durham 03-32-010 Municipality   Nov 2015        41.80 2015-11-01
## 12            Durham 03-32-010 Municipality   Dec 2015        37.53 2015-12-01
```

```
#5 # Creating a line plot

water_data$Month <- factor(water_data$Month, levels = c("Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul"

# Plotting maximum daily withdrawals for each month
ggplot(water_data, aes(x = Month, y = MaxDayUseMGD, group = 1)) +
  geom_line() +
  geom_point() +
  labs(title = "Maximum Daily Water Withdrawals in 2015",
       x = "Month",
       y = "Maximum Day Use (MGD)") +
  theme_minimal()
```



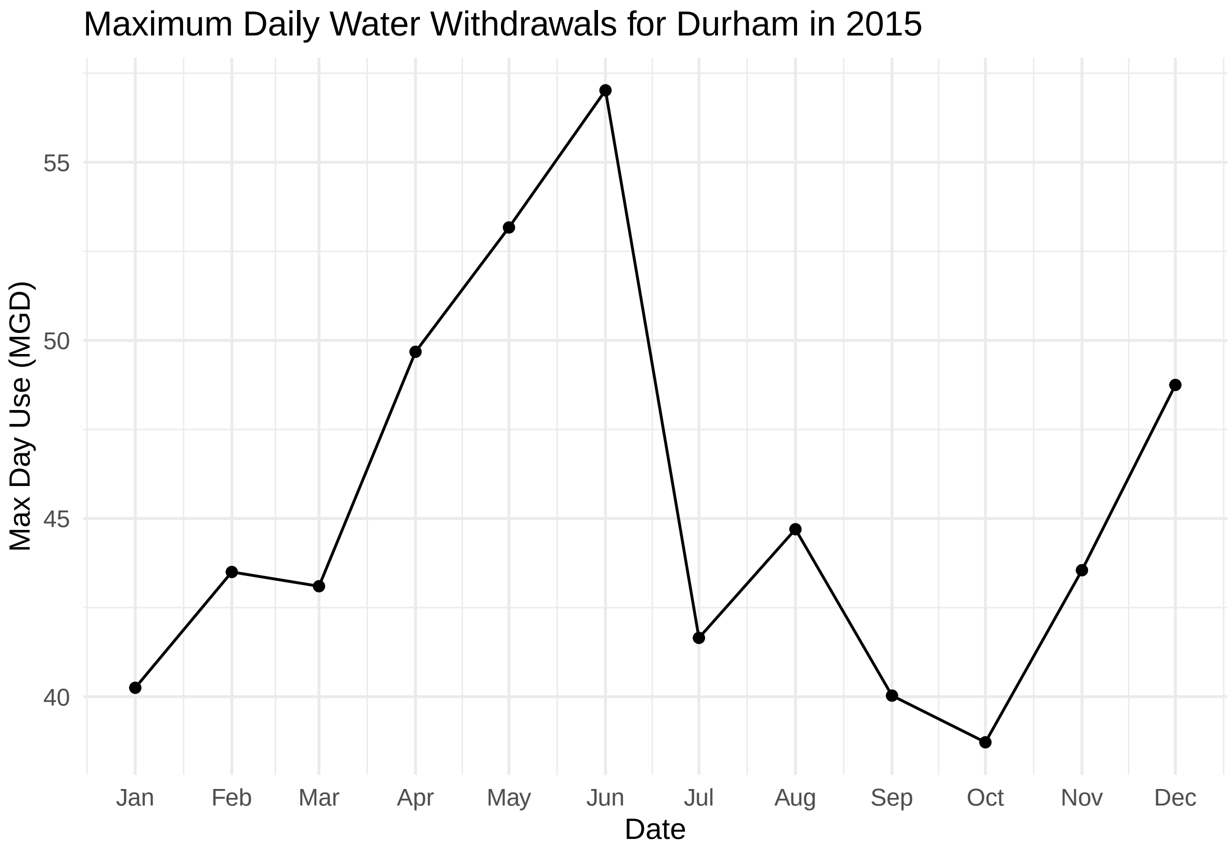
6. Note that the PWSID and the year appear in the web address for the page we scraped. Construct a function using your code above that can scrape data for any PWSID and year for which the NC DEQ has data, returning a dataframe. **Be sure to modify the code to reflect the year and site (pwsid) scraped**.

```r
# 6. Creating a Scrape Data Function
scrape_data_function <- function(PWSID, the_year) {

  the_base_url <- 'https://www.ncwater.org/WUDC/app/LWSP'

  the_scrape_url <- paste0(the_base_url, '/report.php?pwsid=', PWSID, '&year=', the_year)
  page <- read_html(the_scrape_url)

  # Scrape the water system details
  water_system_name <- page %>%
  html_nodes("div+ table tr:nth-child(1) td:nth-child(2)")%>%
  html_text()

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

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

  # Scrape the maximum daily usage for each month (assuming 12 values, one for each month)
  maximum_day_use <-page %>%
  html_nodes("th~ td+ td")%>%
  html_text

  if(length(maximum_day_use) != 12) {
    stop("The number of monthly data points for MaxDayUsage is not correct.")
  }

  # Create the dataframe with months, year, and max day usage
 df_withdrawals_func <- data.frame("Month" = c(1, 5, 9, 2, 6, 10, 3, 7, 11, 4,8,12),
                             "Year" = rep(the_year,12),
                             "MaxDayUsage" = as.numeric(maximum_day_use)) %>%
                    mutate(WaterSystemName= !!water_system_name,
                           PWSID=!!PWSID,
                           Ownership=!!ownership,
                      Date=lubridate::my(paste(Month, "-", Year))) %>%
                    dplyr::arrange(Date)


  #Return the dataframe
  return(df_withdrawals_func)

}
```

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

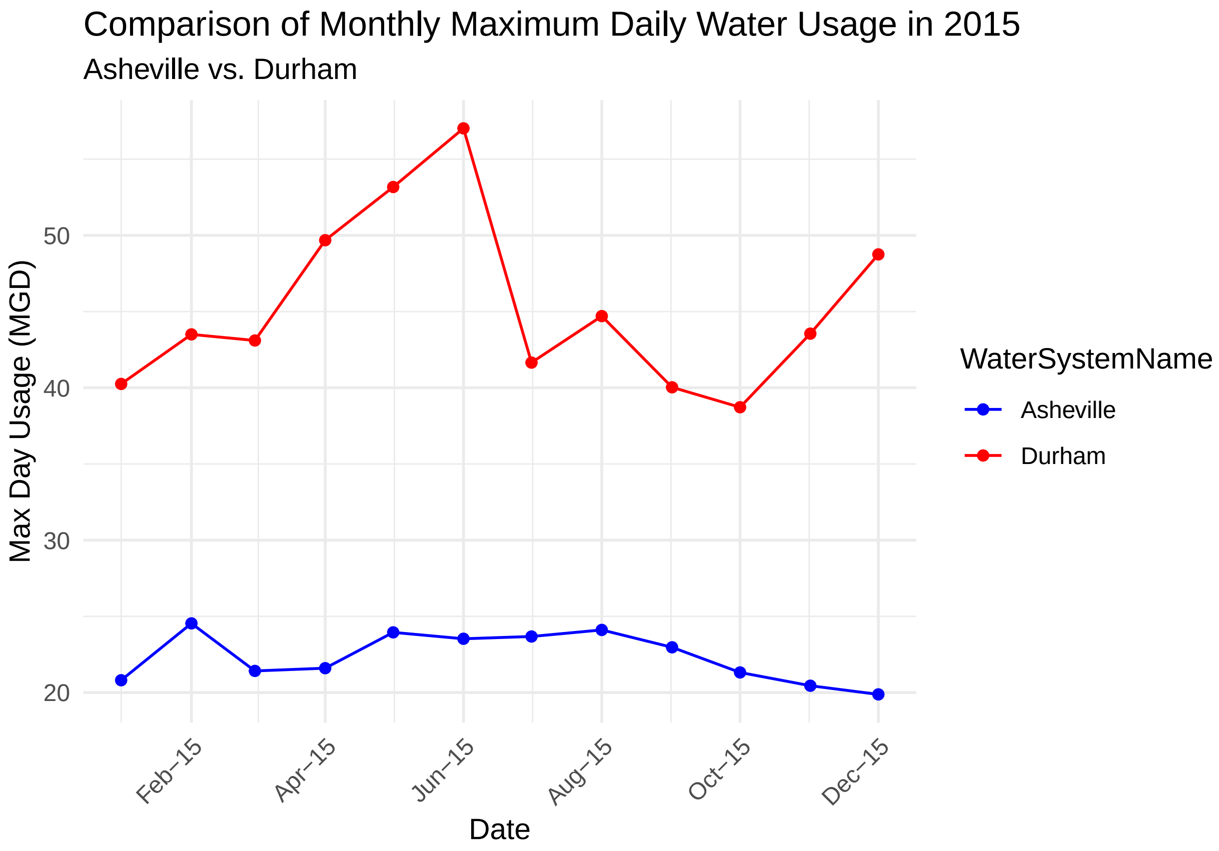
```r
# 7. Fetch Data for Durham (PWSID '03-32-010') in 2015
durham_data_2015 <- scrape_data_function('03-32-010', 2015)
```

```
ggplot(durham_data_2015, aes(x = Date, y = MaxDayUsage)) +
  geom_line() +
  geom_point() +
  labs(title = "Maximum Daily Water Withdrawals for Durham in 2015",
       x = "Date",
       y = "Max Day Use (MGD)") +
  theme_minimal() +
  scale_x_date(date_breaks = "1 month", date_labels = "%b")
```



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
# Fetch data for Asheville (PWSID '01-11-010') in 2015
asheville_data_2015 <- scrape_data_function('01-11-010', 2015)

# Combine the Durham data and Asheville data into one dataframe
combined_data <- bind_rows(asheville_data_2015, durham_data_2015)

# Ensure that the 'Date' column is correctly ordered and in Date format
combined_data$Date <- as.Date(combined_data$Date)

# Create the plot comparing Asheville and Durham's water withdrawals
ggplot(combined_data, aes(x = Date, y = MaxDayUsage, color = WaterSystemName)) +
  geom_line() +
  geom_point() +
  labs(
    title = "Comparison of Monthly Maximum Daily Water Usage in 2015",
    subtitle = "Asheville vs. Durham",
    x = "Date",
    y = "Max Day Usage (MGD)"
  ) +
  theme_minimal() +
  scale_color_manual(values = c("Asheville" = "blue", "Durham" = "red")) +
  scale_x_date(
    date_breaks = "2 months",
    date_labels = "%b-%y"
  ) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1)
  )
```

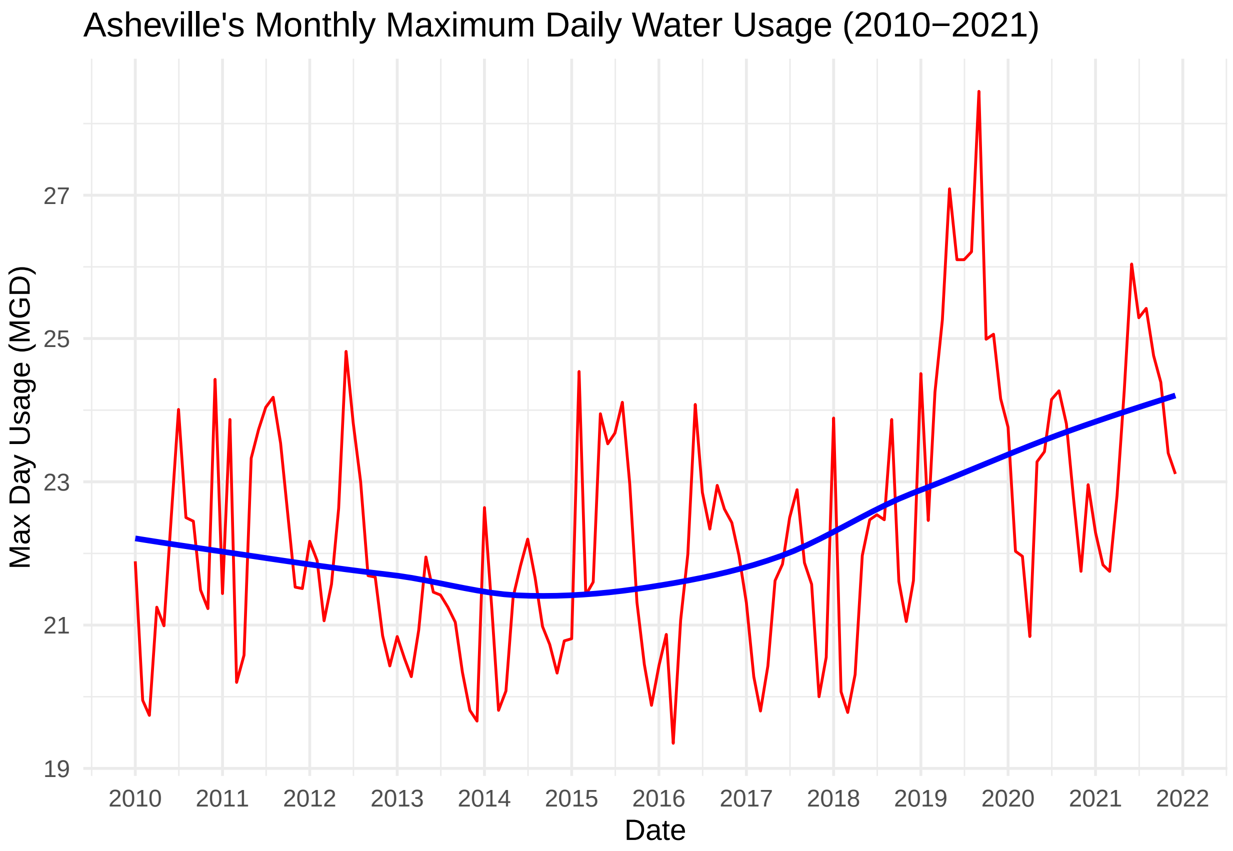


9. Use the code & function you created above to plot Asheville's max daily withdrawal by months for the years 2018 thru 2022.Add a smoothed line to the plot (method = 'loess').

   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.

```r
#9 # Fetch and plot Asheville's maximum daily withdrawal data for 2010 to 2021
library(tidyverse)
library(lubridate)
library(purrr)

# Define the years to be scraped
years <- 2010:2021

# Scrape data for Asheville (PWSID = '01-11-010') for each year
asheville_data_list <- map(years, ~ scrape_data_function('01-11-010', .x))

# Combine the data into a single dataframe
asheville_data <- bind_rows(asheville_data_list)

# Plot the data
ggplot(asheville_data, aes(x = Date, y = MaxDayUsage)) +
  geom_line(color = "red") +
  geom_smooth(method = 'loess', se = FALSE, color = "blue") +
  labs(title = "Asheville's Monthly Maximum Daily Water Usage (2010-2021)",
       x = "Date",
       y = "Max Day Usage (MGD)") +
  theme_minimal() +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y")
```

```
## ‘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: >Based on the plot, Asheville's water usage shows a slight upward trend from 2010 to 2021, indicating a gradual increase in demand over time.