Project2

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.2
## v ggplot2   3.5.2     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(tidytuesdayR)
```

Part 1

```
#1A
Exp <- function(x, k) {
  results <- 1 #note to self, you need to initialize something in a for loop in r, without this the fun
    for (i in 1:k) {
      results <- results+(x^i)/factorial(i)
    }
    return(results)
}

Exp(5,2) #Output should be 18.5
```

```
## [1] 18.5
```

```
#1B
#sample mean
sample_mean <- function(x) {
      mean_res <-sum(x)/length(x)
      return(mean_res)
}
x <-c(18,21,22,3,5,15,16)
sample_mean(x)
```

```
## [1] 14.28571
```

```r
#sample sd
sample_sd <- function(x) {
  N <- length(x)
  the_mean <- sample_mean(x)
    total <- 0 #again need to initialize
        for (i in 1:N) {
     total <- (total + (x[i]-the_mean)^2)
        }
  #this is the last part of the equation, separate because there is no need to   iterate over anything
  deno <- (N-1)
  ans <- sqrt(total/deno)
  return(ans)
}

sample_sd(x)
```

```
## [1] 7.476949
```

```r
#1C
calculate_CI <- function(x, conf = 0.95) {
    sd <- sample_sd(x)
    mean <- sample_mean(x)
    length <- length(x)

    alpha <- 1-conf
    se <- sd/sqrt(length)
    df <- length-1
    t_score <- qt(p = alpha / 2, df = df,lower.tail= FALSE)

    me <- t_score*se#to simplify writing the next setp

    upper_bound <- mean+me
    lower_bound <- mean-me

    return(c(lower_bound = lower_bound, upper_bound = upper_bound))#this could be taken out but I wante
}

calculate_CI(x, conf = 0.95)
```

```
## lower_bound upper_bound
##     7.37069    21.20074
```

```r
calculate_CI(x, conf = 0.99)
```

```
## lower_bound upper_bound
##    3.808445   24.762984
```

Part2

```r
library(here)
```

```
## here() starts at C:/Users/topas/OneDrive/Documents/Computational-Statistics-Projects
```

```r
if (!file.exists(here("data", "tuesdata_rainfall.RDS"))) {
    tuesdata <- tidytuesdayR::tt_load("2020-01-07")
    rainfall <- tuesdata$rainfall
    temperature <- tuesdata$temperature

    # save the files to RDS objects
    saveRDS(tuesdata$rainfall, file = here("data", "tuesdata_rainfall.RDS"))
    saveRDS(tuesdata$temperature, file = here("data", "tuesdata_temperature.RDS"))
}
```

```r
rainfall <- readRDS(here("data", "tuesdata_rainfall.RDS"))
temperature <- readRDS(here("data", "tuesdata_temperature.RDS"))
```

```r
#Part 1: Drop any NA values from rainfall
rainfall <- rainfall %>%
  drop_na()

#Part2
library(lubridate)
rainfall$date <-ymd(paste(rainfall$year,rainfall$month,rainfall$day, sep = "-"))

rainfall <- subset(rainfall, select= -c(month,day))

head(rainfall)
```

```
## # A tibble: 6 x 10
##   station_code city_name  year rainfall period quality   lat  long station_name
##   <chr>        <chr>     <dbl>    <dbl>  <dbl> <chr>   <dbl> <dbl> <chr>
## 1 009151       Perth      1967      2.8      1 Y       -32.0  116. Subiaco Wast~
## 2 009151       Perth      1967      4.8      1 Y       -32.0  116. Subiaco Wast~
## 3 009151       Perth      1967      5.8      1 Y       -32.0  116. Subiaco Wast~
## 4 009151       Perth      1967     16        1 Y       -32.0  116. Subiaco Wast~
## 5 009151       Perth      1967      1        1 Y       -32.0  116. Subiaco Wast~
## 6 009151       Perth      1967      1        1 Y       -32.0  116. Subiaco Wast~
## # i 1 more variable: date <date>
```

```r
#Part3
rainfall$city_name <- toupper(rainfall$city_name)

#Part4
join <- inner_join(rainfall, temperature, by = c("date", "city_name"))
```

```
## Warning in inner_join(rainfall, temperature, by = c("date", "city_name")): Detected an unexpected ma
## i Row 1 of `x` matches multiple rows in `y`.
## i Row 138722 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```r
nrow(join)
```

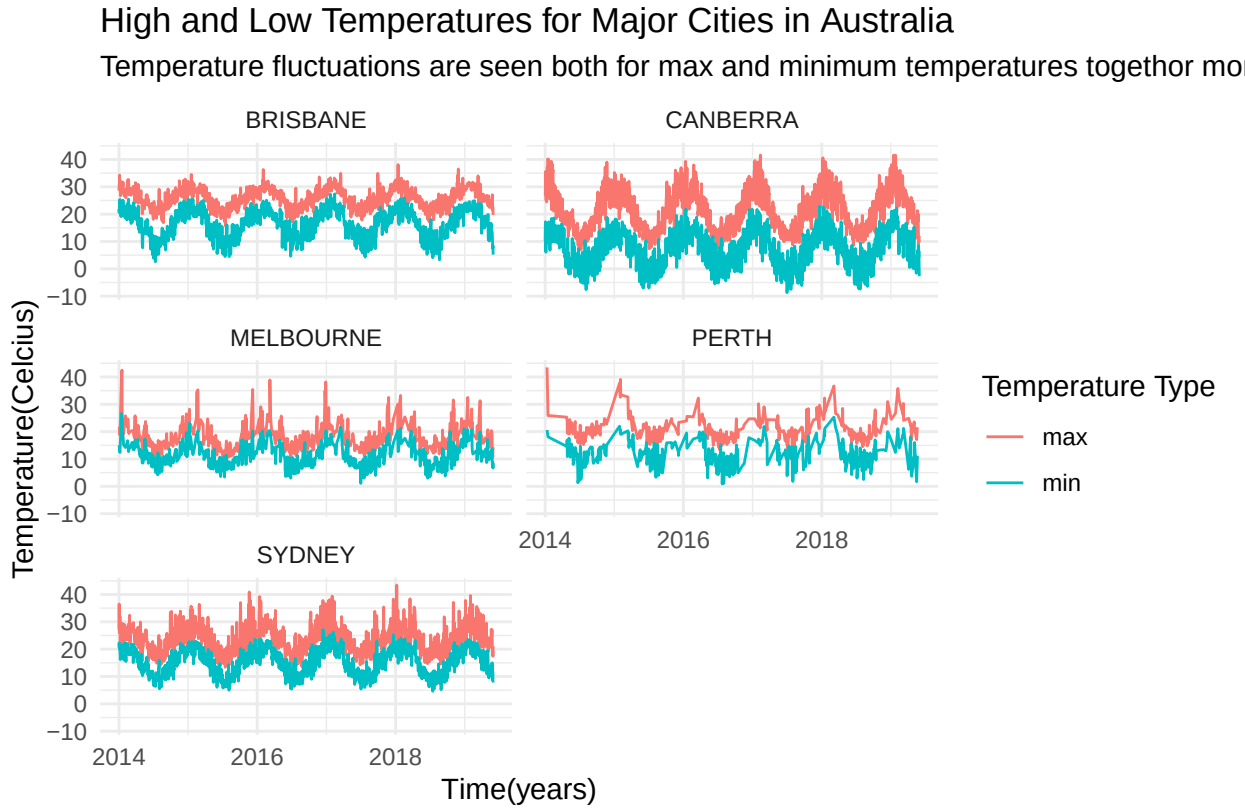
```
## [1] 83964
```

```
ncol(join)
```

```
## [1] 13
```

Part 3

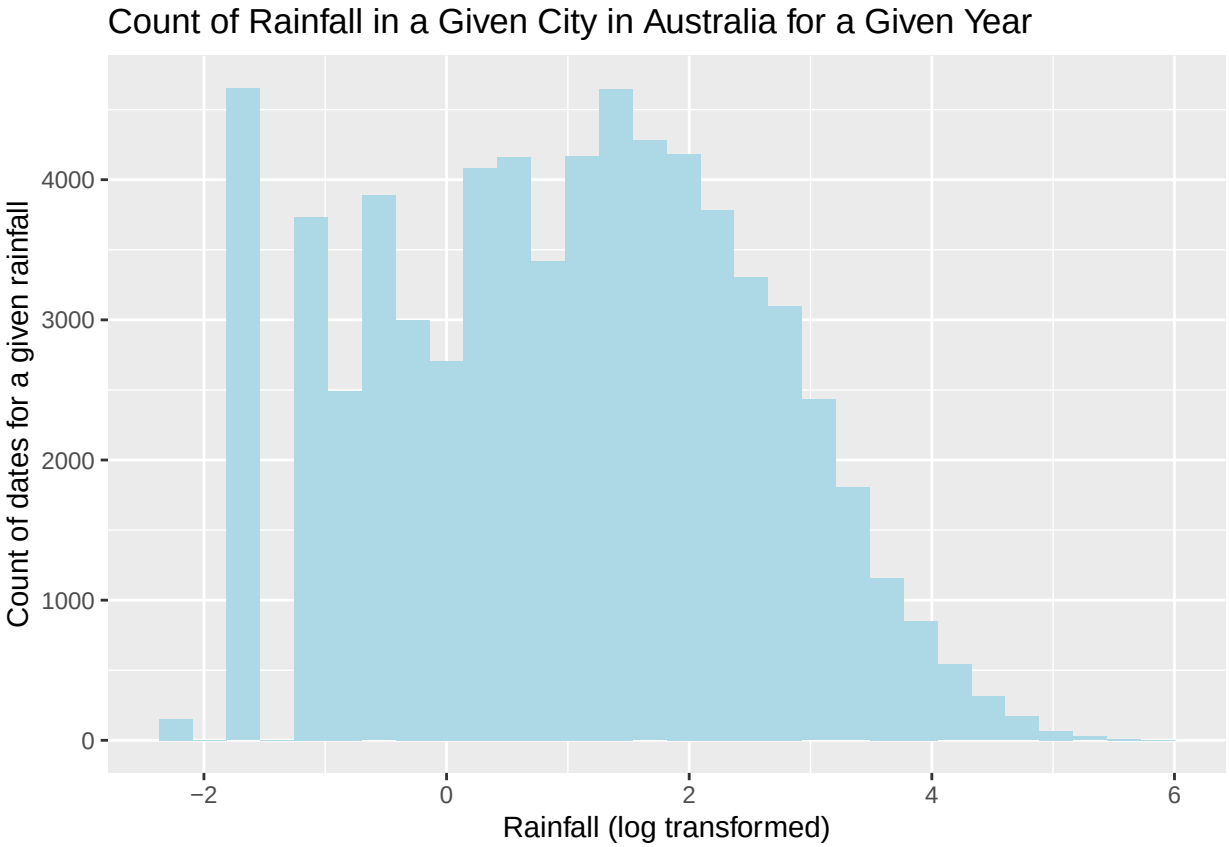
```
#3a
#An overall title for the plot and a subtitle summarizing key trends that you found. Also include a cap
#There should be an informative x-axis and y-axis label.

only_new <- join %>%
  filter(year >= 2014)
graph1 <- ggplot(data = only_new, aes(x=date,y=temperature, color=temp_type))+
  geom_line()+
  facet_wrap(~city_name, ncol =2)+
  labs(x = "Time(years)", y= "Temperature(Celcius)", title = "High and Low Temperatures for Major Cities
  theme_minimal()

graph1
```




```
#3b
#The aim here is to design and implement a function that can be re-used to visualize all of the data in
graph_it <- function(city_name,year){
  if(!(city_name %in% join$city_name)) {
```

```
    stop("Error: This city does not exist")
  }

  if(!(year %in% join$year)) {
    stop("Error: This year does not exist")
  }

  else if(sum(join$city_name == city_name & join$year == year) == 0) {
    stop("Error: This combo of city name and year is invalid") #the logic behind this section relies on
  }

 rel <- join %>%
  filter(city_name == city_name & year == year) %>%
  ggplot(aes(log(rainfall)))+
   geom_histogram(fill = "lightblue")+
   labs(x="Rainfall (log transformed)", y = "Count of dates for a given rainfall", title = "Count of Ra

 return(rel)
}
#change the name of the title here

graph_it("PERTH",1994)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 16830 rows containing non-finite outside the scale range
## (`stat_bin()`).
```



For this section, I first chose to write the error messages. So if the user does not have the correct city name from the join data set or the correct year, it will produce an error. In addition I include a stop/error message for when there is a combination of city and year that may exist separate in the data set, but does not exist together. Then, if the city name and year are correct, I used DPLYR to filter the input city name and the given year, used ggplot2 to return a histogram.

Part 4

```
#4a
rain_df <- join %>%
  filter(year >= 2014) %>%
  group_by(city_name,year) %>%
  summarize(
    mean = sample_mean(rainfall),
    sd = sample_sd(rainfall),
    lower_bound = calculate_CI(rainfall)[1],
    upper_bound = calculate_CI(rainfall)[2],

  )
```

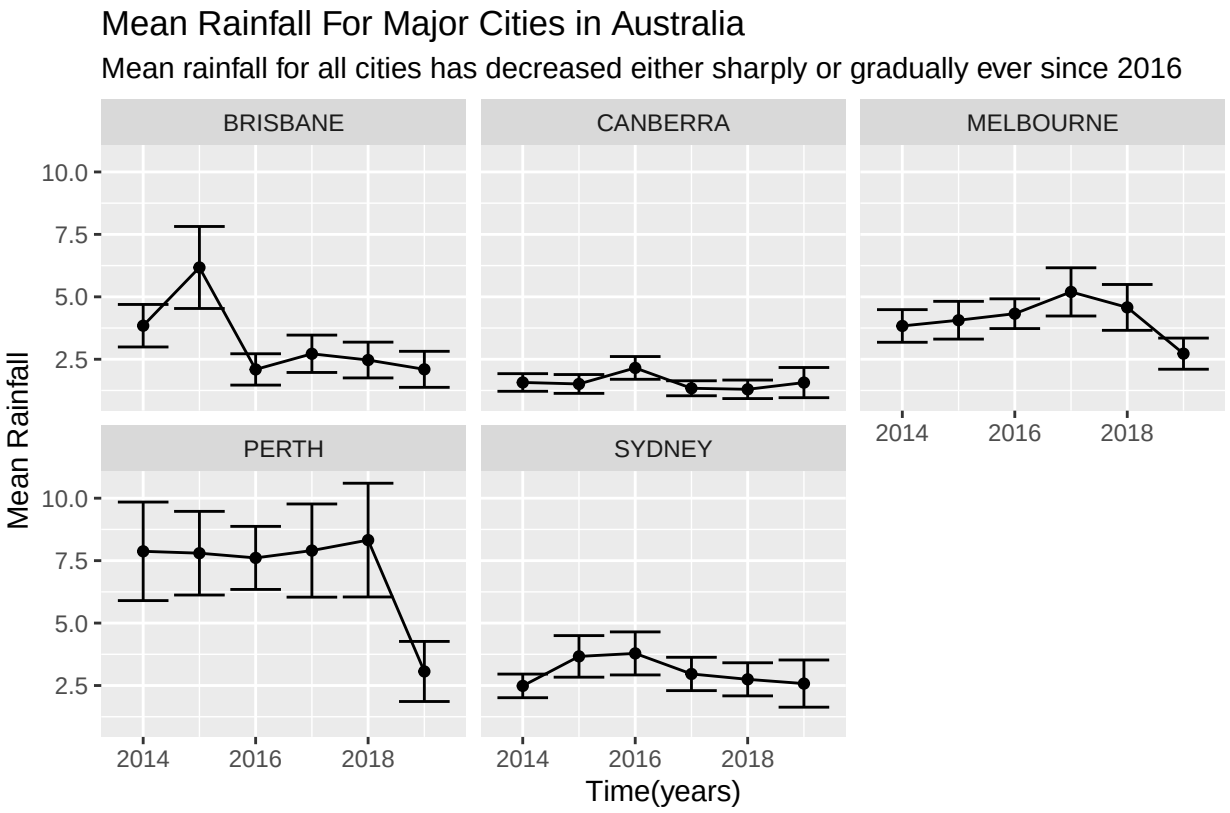
```
## `summarise()` has grouped output by 'city_name'. You can override using the
## `.groups` argument.
```

```
glimpse(rain_df)
```

```
## Rows: 30
```

```
## Columns: 6
## Groups: city_name [5]
## $ city_name   <chr> "BRISBANE", "BRISBANE", "BRISBANE", "BRISBANE", "BRISBANE"~
## $ year        <dbl> 2014, 2015, 2016, 2017, 2018, 2019, 2014, 2015, 2016, 2017~
## $ mean        <dbl> 3.843641, 6.174595, 2.092562, 2.719101, 2.468966, 2.097333~
## $ sd          <dbl> 12.277074, 22.745373, 8.607206, 10.164961, 9.626574, 6.348~
## $ lower_bound <dbl> 2.9926743, 4.5331087, 1.4654175, 1.9711835, 1.7525381, 1.3~
## $ upper_bound <dbl> 4.694608, 7.816081, 2.719706, 3.467019, 3.185393, 2.818605~
```

```
#4b
ggplot(data = rain_df, aes(x=year,y = mean))+
  geom_point()+
  facet_wrap(~city_name)+
  geom_line()+
  geom_errorbar(aes(ymin = lower_bound, ymax = upper_bound))+
  labs(x = "Time(years)", y="Mean Rainfall", title = "Mean Rainfall For Major Cities in Australia", sub
```




```
#Example from class, good to have as notes
#for (i in names(penguins)) {
 #print(i)
 #var_data <- pull(penguins,i)
 #print(var_data)
 #if(is.numeric(var_data)){
   #print(mean(var_data, na.rm = TRUE))
 #}
```

```
 #else{
   #print("NA")
 #}
  #}
library(sessioninfo)
session_info()
```

```
## Warning in system2("quarto", "-V", stdout = TRUE, env = paste0("TMPDIR=", :
## running command '"quarto" TMPDIR=C:/Users/topas/AppData/Local/Temp/RtmpYBnW3i/file23d605e022171 -V' had
## status 1

## - Session info -----------------------------------------------------------------
##  setting  value
##  version  R version 4.3.3 (2024-02-29 ucrt)
##  os       Windows 11 x64 (build 26100)
##  system   x86_64, mingw32
##  ui       RTerm
##  language (EN)
##  collate  English_United States.utf8
##  ctype    English_United States.utf8
##  tz       America/New_York
##  date     2025-09-27
##  pandoc   3.4 @ C:/Program Files/RStudio/resources/app/bin/quarto/bin/tools/ (via rmarkdown)
##  quarto   NA @ C:\\PROGRA~1\\RStudio\\RESOUR~1\\app\\bin\\quarto\\bin\\quarto.exe
##
## - Packages ---------------------------------------------------------------------
##  package      * version date (UTC) lib source
##  cli            3.6.3   2024-06-21 [1] CRAN (R 4.3.3)
##  digest         0.6.37  2024-08-19 [1] CRAN (R 4.3.3)
##  dplyr        * 1.1.4   2023-11-17 [1] CRAN (R 4.3.3)
##  evaluate       1.0.5   2025-08-27 [1] CRAN (R 4.3.3)
##  farver         2.1.2   2024-05-13 [1] CRAN (R 4.3.3)
##  fastmap        1.2.0   2024-05-15 [1] CRAN (R 4.3.3)
##  forcats      * 1.0.0   2023-01-29 [1] CRAN (R 4.3.3)
##  generics       0.1.4   2025-05-09 [1] CRAN (R 4.3.3)
##  ggplot2      * 3.5.2   2025-04-09 [1] CRAN (R 4.3.3)
##  glue           1.8.0   2024-09-30 [1] CRAN (R 4.3.3)
##  gtable         0.3.6   2024-10-25 [1] CRAN (R 4.3.3)
##  here         * 1.0.2   2025-09-15 [1] CRAN (R 4.3.3)
##  hms            1.1.3   2023-03-21 [1] CRAN (R 4.3.3)
##  htmltools      0.5.8.1 2024-04-04 [1] CRAN (R 4.3.3)
##  knitr          1.50    2025-03-16 [1] CRAN (R 4.3.3)
##  labeling       0.4.3   2023-08-29 [1] CRAN (R 4.3.1)
##  lifecycle      1.0.4   2023-11-07 [1] CRAN (R 4.3.3)
##  lubridate    * 1.9.4   2024-12-08 [1] CRAN (R 4.3.3)
##  magrittr       2.0.3   2022-03-30 [1] CRAN (R 4.3.3)
##  pillar         1.11.1  2025-09-17 [1] CRAN (R 4.3.3)
##  pkgconfig      2.0.3   2019-09-22 [1] CRAN (R 4.3.3)
##  purrr        * 1.0.4   2025-02-05 [1] CRAN (R 4.3.3)
##  R6             2.6.1   2025-02-15 [1] CRAN (R 4.3.3)
##  RColorBrewer   1.1-3   2022-04-03 [1] CRAN (R 4.3.1)
##  readr        * 2.1.5   2024-01-10 [1] CRAN (R 4.3.3)
```

```
##  rlang          1.1.5   2025-01-17 [1] CRAN (R 4.3.3)
##  rmarkdown      2.29    2024-11-04 [1] CRAN (R 4.3.3)
##  rprojroot      2.1.1   2025-08-26 [1] CRAN (R 4.3.3)
##  rstudioapi     0.17.1  2024-10-22 [1] CRAN (R 4.3.3)
##  scales         1.4.0   2025-04-24 [1] CRAN (R 4.3.3)
##  sessioninfo  * 1.2.3   2025-02-05 [1] CRAN (R 4.3.3)
##  stringi        1.8.7   2025-03-27 [1] CRAN (R 4.3.3)
##  stringr      * 1.5.2   2025-09-08 [1] CRAN (R 4.3.3)
##  tibble       * 3.2.1   2023-03-20 [1] CRAN (R 4.3.3)
##  tidyr        * 1.3.1   2024-01-24 [1] CRAN (R 4.3.3)
##  tidyselect     1.2.1   2024-03-11 [1] CRAN (R 4.3.3)
##  tidytuesdayR * 1.2.1   2025-04-29 [1] CRAN (R 4.3.3)
##  tidyverse    * 2.0.0   2023-02-22 [1] CRAN (R 4.3.3)
##  timechange     0.3.0   2024-01-18 [1] CRAN (R 4.3.3)
##  tzdb           0.5.0   2025-03-15 [1] CRAN (R 4.3.3)
##  utf8           1.2.4   2023-10-22 [1] CRAN (R 4.3.3)
##  vctrs          0.6.5   2023-12-01 [1] CRAN (R 4.3.3)
##  withr          3.0.2   2024-10-28 [1] CRAN (R 4.3.3)
##  xfun           0.52    2025-04-02 [1] CRAN (R 4.3.3)
##  yaml           2.3.10  2024-07-26 [1] CRAN (R 4.3.3)
##
##  [1] C:/Users/topas/AppData/Local/R/win-library/4.3
##  [2] C:/Program Files/R/R-4.3.3/library
##  * -- Packages attached to the search path.
##
## ------------------------------------------------------------------------------
```