library(tidyverse)  #helps wrangle data

library(lubridate)  #helps wrangle date attributes

library(ggplot2)  #helps visualize data

> setwd("C:/Users/scott/Documents/BikeShare/CSV") #Sets working directory

> getwd() #Check to make sure working directory is correct

#Upload data sets (CSV files)

> Jul\_2022 <- read.csv("202207-divvy-tripdata.csv")

> Aug\_2022 <- read.csv("202208-divvy-tripdata.csv")

> Sep\_2022 <- read.csv("202209-divvy-publictripdata.csv")

> Oct\_2022 <- read.csv("202210-divvy-tripdata.csv")

> Nov\_2022 <- read.csv("202211-divvy-tripdata.csv")

> Dec\_2022 <- read.csv("202212-divvy-tripdata.csv")

> Jan\_2023 <- read.csv("202301-divvy-tripdata.csv")

> Feb\_2023 <- read.csv("202302-divvy-tripdata.csv")

> Mar\_2023 <- read.csv("202303-divvy-tripdata.csv")

> Apr\_2023 <- read.csv("202304-divvy-tripdata.csv")

> May\_2023 <- read.csv("202305-divvy-tripdata.csv")

> Jun\_2023 <- read.csv("202301-divvy-tripdata.csv")

#Combine data into one data frame

> fullyear1 <- bind\_rows(Jul\_2022, Aug\_2022, Sep\_2022, Oct\_2022, Nov\_2022, Dec\_2022, Jan\_2023, Feb\_2023, Mar\_2023, Apr\_2023, May\_2023, Jun\_2023)

#Remove columns that are not relevant to this case

> fullyear2 <- subset(fullyear1, select = -c(5, 6, 7, 8))

#Find the time of each ride (in seconds) that we will call ride\_length

> fullyear2$ride\_length <- difftime(fullyear2$ended\_at, fullyear2$started\_at)

#Create new columns date, month, day, year and weekday

> fullyear2$date <- as.Date(fullyear2$started\_at)

> fullyear2$month <- format(as.Date(fullyear2$date), "%m")

> fullyear2$day <- format(as.Date(fullyear2$date), "%d")

> fullyear2$year <- format(as.Date(fullyear2$date), "%y")

> fullyear2$weekday <- format(as.Date(fullyear2$date), "%A")

#Convert class for ride length to be able to perform calculations

> fullyear3$ride\_length <- type.convert(fullyear3$ride\_length, as.is = TRUE)

#Remove bad data with rows that have a negative ride length

> fullyear3 <- fullyear2[!(fullyear2$ride\_length<0),]

#Compare how casual and member riders differ by weekday and ride length

> aggregate(fullyear3$ride\_length ~ fullyear3$member\_casual + fullyear3$weekday, FUN = mean)

#Weekdays were listed out of order so we fix that issue

> fullyear3$weekday <- ordered(fullyear3$weekday, levels = c("Sunday" , "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"))

#Check code again to insure it is properly ordered

> aggregate(fullyear3$ride\_length ~ fullyear3$member\_casual + fullyear3$weekday, FUN = mean)

#Compare the total number of rides between casual and members for each day

fullyear3 %>%

group\_by(member\_casual, weekday) %>%

summarise(numberofrides = n())

#Compare how casual and member riders differ between bike choice

fullyear3 %>%

group\_by(member\_casual, rideable\_type) %>%

summarise(numberofrides = n())

#Create visual to compare total rides per day between customers

fullyear3 %>%

group\_by(member\_casual, weekday) %>%

summarise(number\_of\_rides = n()) %>%

arrange(member\_casual, weekday) %>%

ggplot(aes(x = weekday, y = number\_of\_rides, fill = member\_casual,)) +

geom\_col(position = "dodge") +

labs(title = "Cyclistic Bike-Share Rides from July 2022 - June 2023", subtitle = "Total Number of Rides Grouped by Customer Type and Weekday", x = "Weekday", y = "Number of Rides"), fill = “Customer Type”))

#Create visual to compare average ride length per day between customers

fullyear3 %>%

group\_by(member\_casual, weekday) %>%

summarise(average\_duration = mean(ride\_length)) %>%

arrange(member\_casual, weekday) %>%

ggplot(aes(x = weekday, y = average\_duration, fill = member\_casual,)) +

geom\_col(position = "dodge") +

labs(title = "Cyclistic Bike-Share Rides from July 2022 - June 2023", subtitle = "Average Time of Rides Grouped by Customer Type and Weekday", x = "Weekday", y = "Average Duration(in seconds)", fill = “Customer Type”)

#Create visual to compare bike choice between customers

> fullyear3 %>%

group\_by(member\_casual, rideable\_type) %>%

summarise(number\_of\_rides = n()) %>%

arrange(member\_casual, rideable\_type) %>%

ggplot(aes(x = rideable\_type, y = number\_of\_rides, fill = member\_casual,)) +

geom\_col(position = "dodge") +

labs(title = "Cyclistic Bike-Share Rides from July 2022 - June 2023", subtitle = "Number of Rides Grouped by Bike and Customer Type", x = "Bike Type", y = "Number of Rides", fill = "Customer Type")