

Accelerating Growth in the Bike-Share Industry

Downloading necessary libraries

necessary libraries

```
options(repos = c(CRAN = "https://cran.uni-muenster.de/"))
install.packages("tidyverse")
```

```
## Installing package into 'C:/Users/maaza/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
##
```

```
## The downloaded binary packages are in
## C:\Users\maaza\AppData\Local\Temp\RtmpWs6Fh9\downloaded_packages
```

```
install.packages("lubridate")
```

```
## Installing package into 'C:/Users/maaza/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'lubridate' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'lubridate'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\maaza\AppData\Local\R\win-library\4.3\OOLOCK\lubridate\libs\x64\lubridate.dll
## to
## C:\Users\maaza\AppData\Local\R\win-library\4.3\lubridate\libs\x64\lubridate.dll:
## Permission denied
```

```
## Warning: restored 'lubridate'
```

```
##
```

```
## The downloaded binary packages are in
## C:\Users\maaza\AppData\Local\Temp\RtmpWs6Fh9\downloaded_packages
```

```
install.packages("readr")
```

```
## Installing package into 'C:/Users/maaza/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'readr' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'readr'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying  
## C:\Users\maaza\AppData\Local\R\win-library\4.3\00LOCK\readr\libs\x64\readr.dll  
## to C:\Users\maaza\AppData\Local\R\win-library\4.3\readr\libs\x64\readr.dll:  
## Permission denied
```

```
## Warning: restored 'readr'
```

```
##  
## The downloaded binary packages are in  
## C:\Users\maaza\AppData\Local\Temp\RtmpWs6Fh9\downloaded_packages
```

```
install.packages("purrr")
```

```
## Installing package into 'C:/Users/maaza/AppData/Local/R/win-library/4.3'  
## (as 'lib' is unspecified)
```

```
## package 'purrr' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'purrr'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying  
## C:\Users\maaza\AppData\Local\R\win-library\4.3\00LOCK\purrr\libs\x64\purrr.dll  
## to C:\Users\maaza\AppData\Local\R\win-library\4.3\purrr\libs\x64\purrr.dll:  
## Permission denied
```

```
## Warning: restored 'purrr'
```

```
##  
## The downloaded binary packages are in  
## C:\Users\maaza\AppData\Local\Temp\RtmpWs6Fh9\downloaded_packages
```

```
library(purrr) # Load the purrr package for map functions  
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --  
## v dplyr      1.1.2      v readr      2.1.4  
## v forcats    1.0.0      v stringr   1.5.0  
## v ggplot2    3.4.2      v tibble    3.2.1  
## v lubridate  1.9.2      v tidyr     1.3.0
```

```
## -- 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(lubridate)  
library(readr)
```

Combining Data

After installing necessary packages now we will import the data. I have used the past 1 year data from April 2022 to April 2023 and combined them together for better analysis.

```
data <- dir("C:/Users/maaza/OneDrive/Case Study 1/Data/", full.names = TRUE) %>%  
  map_df(read_csv)
```

```
## Rows: 371249 Columns: 13  
## -- Column specification -----  
## Delimiter: ","  
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...  
## dbl (4): start_lat, start_lng, end_lat, end_lng  
## dtm (2): started_at, ended_at  
##  
## i Use 'spec()' to retrieve the full column specification for this data.  
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## Rows: 634858 Columns: 13  
## -- Column specification -----  
## Delimiter: ","  
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...  
## dbl (4): start_lat, start_lng, end_lat, end_lng  
## dtm (2): started_at, ended_at  
##  
## i Use 'spec()' to retrieve the full column specification for this data.  
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## Rows: 769204 Columns: 13  
## -- Column specification -----  
## Delimiter: ","  
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...  
## dbl (4): start_lat, start_lng, end_lat, end_lng  
## dtm (2): started_at, ended_at  
##  
## i Use 'spec()' to retrieve the full column specification for this data.  
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## Rows: 823488 Columns: 13  
## -- Column specification -----  
## Delimiter: ","  
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...  
## dbl (4): start_lat, start_lng, end_lat, end_lng  
## dtm (2): started_at, ended_at  
##  
## i Use 'spec()' to retrieve the full column specification for this data.  
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## Rows: 785932 Columns: 13  
## -- Column specification -----  
## Delimiter: ","  
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...  
## dbl (4): start_lat, start_lng, end_lat, end_lng  
## dtm (2): started_at, ended_at  
##  
## i Use 'spec()' to retrieve the full column specification for this data.  
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## Rows: 701339 Columns: 13
```

```

## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 558685 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 337735 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 181806 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 190301 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 190445 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 258678 Columns: 13

```

```
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## Rows: 426590 Columns: 13
## -- Column specification -----
## Delimiter: ","
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end...
## dbl (4): start_lat, start_lng, end_lat, end_lng
## dtm (2): started_at, ended_at
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
View(data)
#this will import the whole folder
```

Data Manipulation

```
#Adding ride length in our dataframe
data$ride_length <- difftime(data$ended_at, data$started_at, units="mins")

#Adding a column 'Day of the week' to understand the daywise frequency of the rides
data$day_of_the_week <- weekdays(data$started_at)
data$day_of_the_week <-
  factor(data$day_of_the_week,
    levels = c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"))
```

```
casual_data <- subset(data, member_casual == "casual")
mean_value_casual <- mean(casual_data$ride_length, na.rm = TRUE)
max_value_casual <- max(casual_data$ride_length, na.rm = TRUE)
frequency_casual <- table(casual_data$day_of_the_week, useNA = "always")
most_frequent_casual <- names(frequency_casual)[which.max(frequency_casual)]

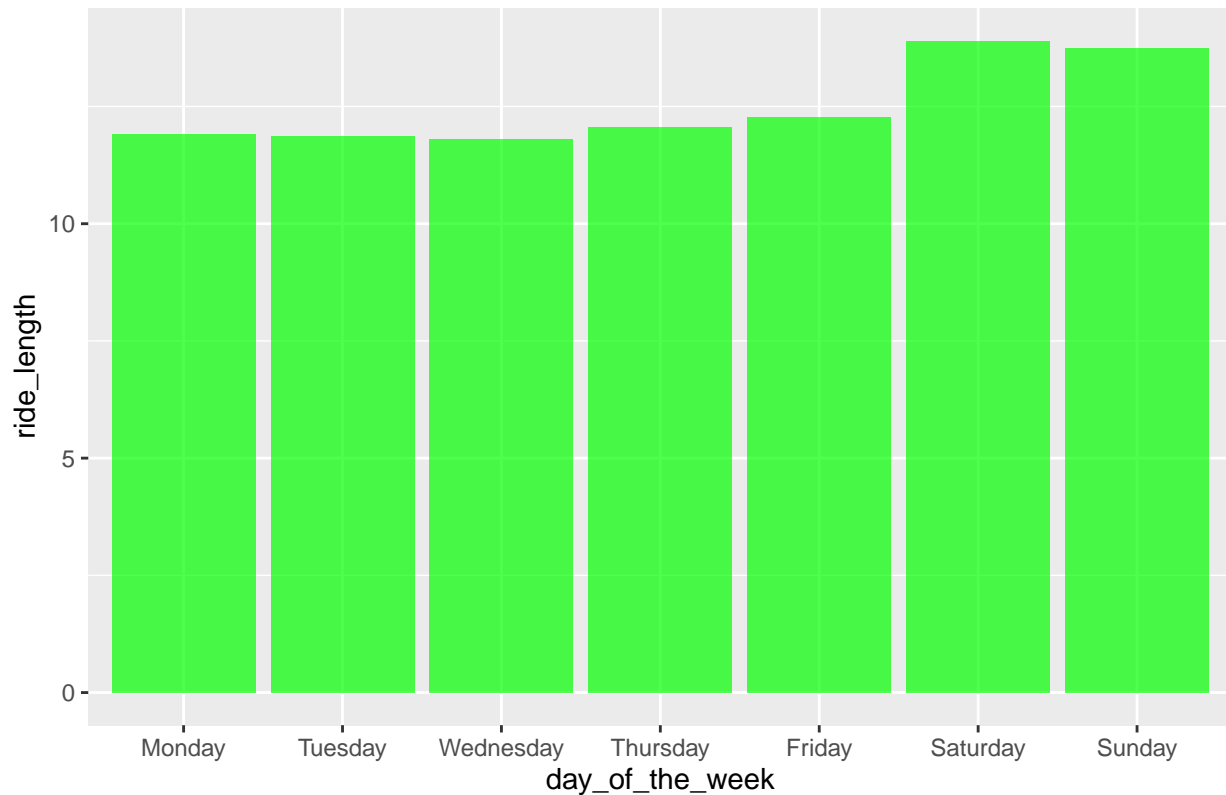
# Create a data frame for members
member_data <- subset(data, member_casual == "member")
mean_value_member <- mean(member_data$ride_length, na.rm = TRUE)
max_value_member <- max(member_data$ride_length, na.rm = TRUE)
frequency_member <- table(member_data$day_of_the_week, useNA = "always")
most_frequent_member <- names(frequency_member)[which.max(frequency_member)]

# Create bar plots to visualize the results
ggplot(member_data, aes(x = day_of_the_week, y = ride_length)) +
  geom_bar(stat = "summary", fun = "mean", fill = "green", alpha = 0.7) +
  ggtitle("Members - Mean Ride Length by Day of the Week")
```

Checking the ride frequency by calculating total number of riders for each group

```
## Don't know how to automatically pick scale for object of type <difftime>.  
## Defaulting to continuous.
```

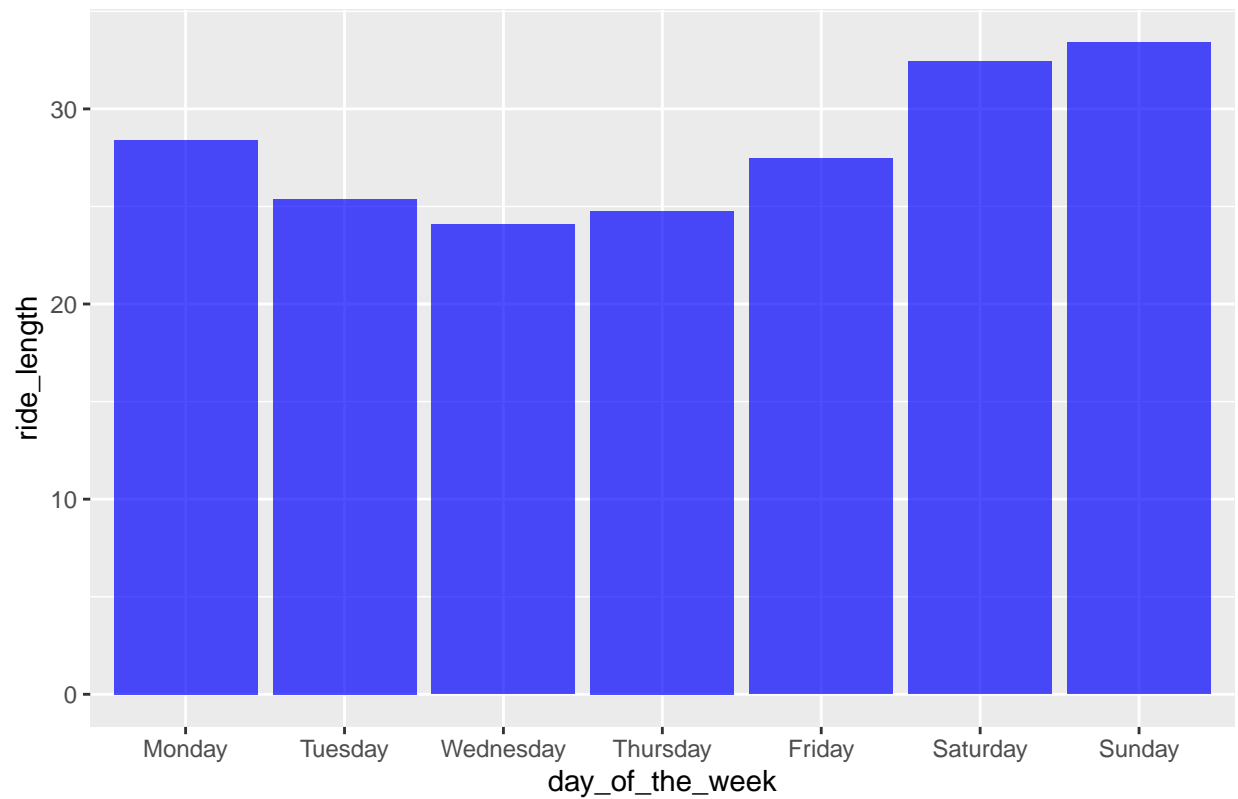
Members – Mean Ride Length by Day of the Week



```
ggplot(casual_data, aes(x = day_of_the_week, y = ride_length)) +  
  geom_bar(stat = "summary", fun = "mean", fill = "blue", alpha = 0.7) +  
  ggtitle("Casual Riders - Mean Ride Length by Day of the Week")
```

```
## Don't know how to automatically pick scale for object of type <difftime>.  
## Defaulting to continuous.
```

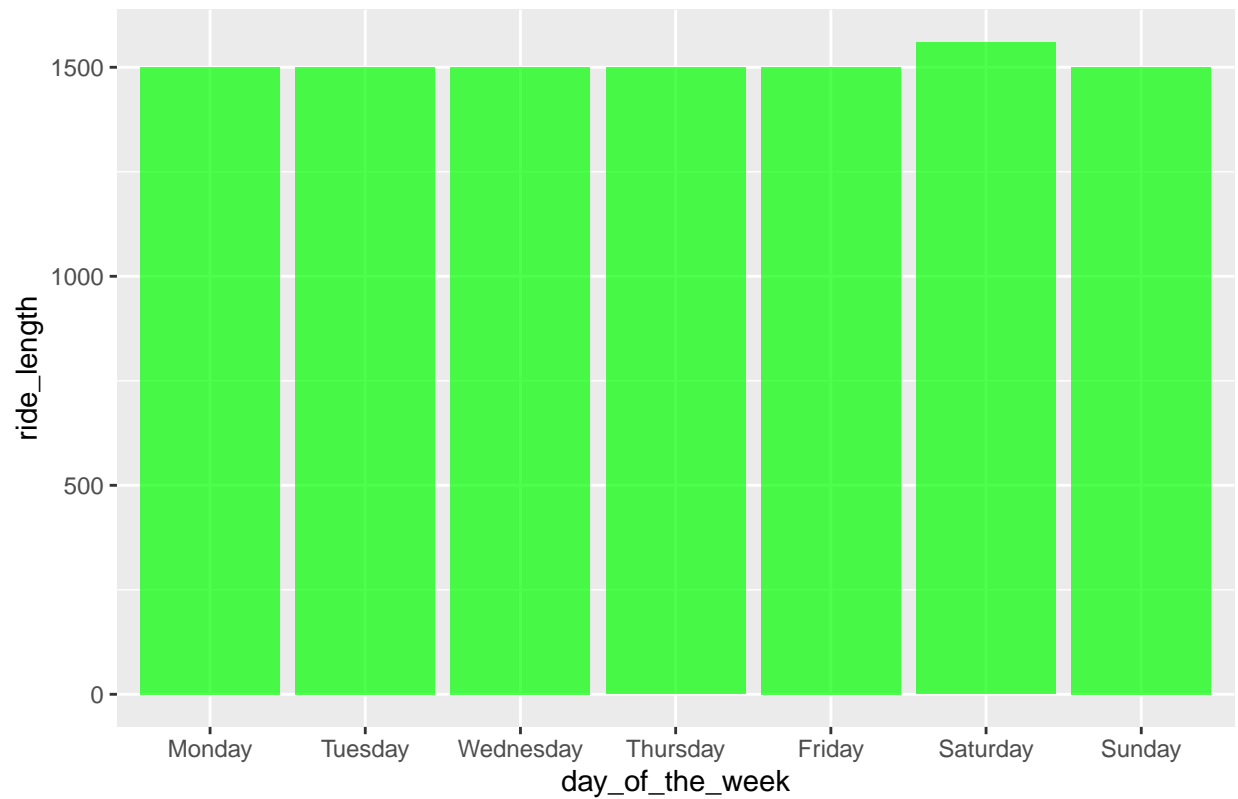
Casual Riders – Mean Ride Length by Day of the Week



```
ggplot(member_data, aes(x = day_of_the_week, y = ride_length)) +  
  geom_bar(stat = "summary", fun = "max", fill = "green", alpha = 0.7) +  
  ggtitle("members - Max Ride Length by Day of the Week")
```

```
## Don't know how to automatically pick scale for object of type <difftime>.  
## Defaulting to continuous.
```

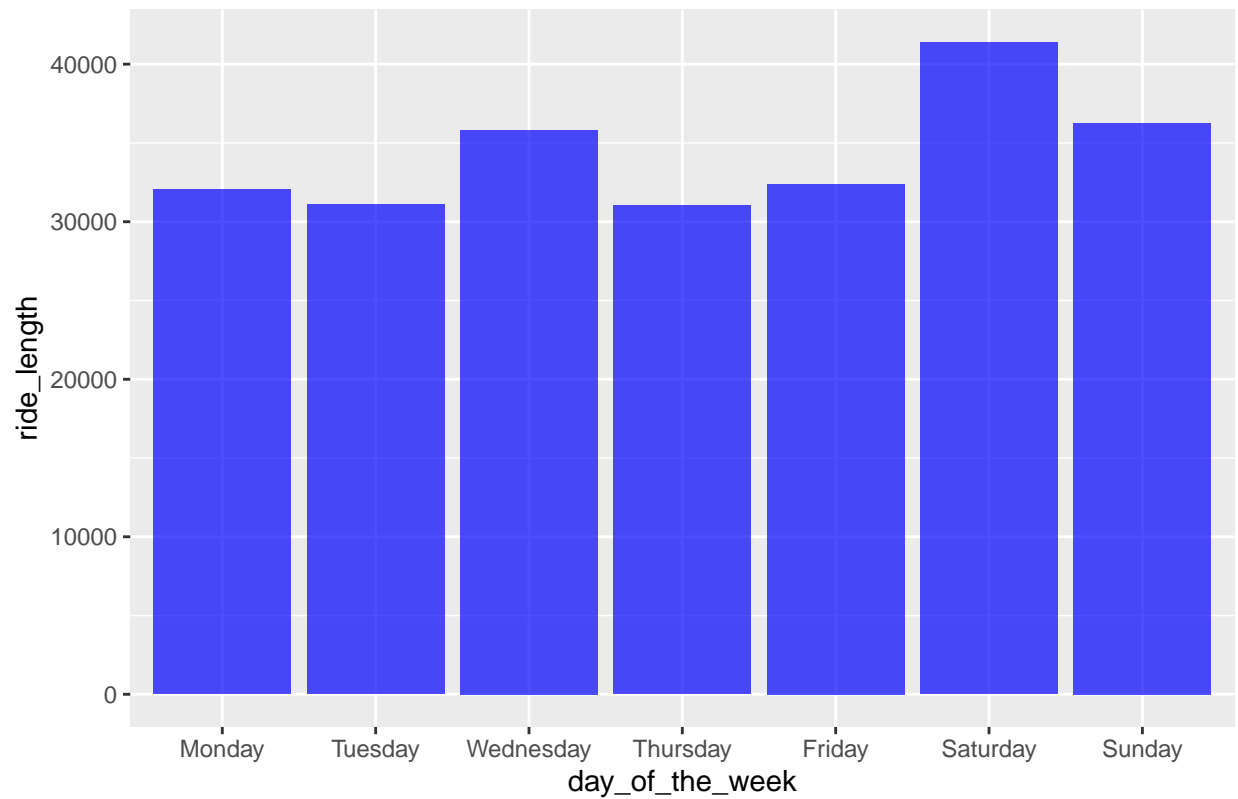
members – Max Ride Length by Day of the Week



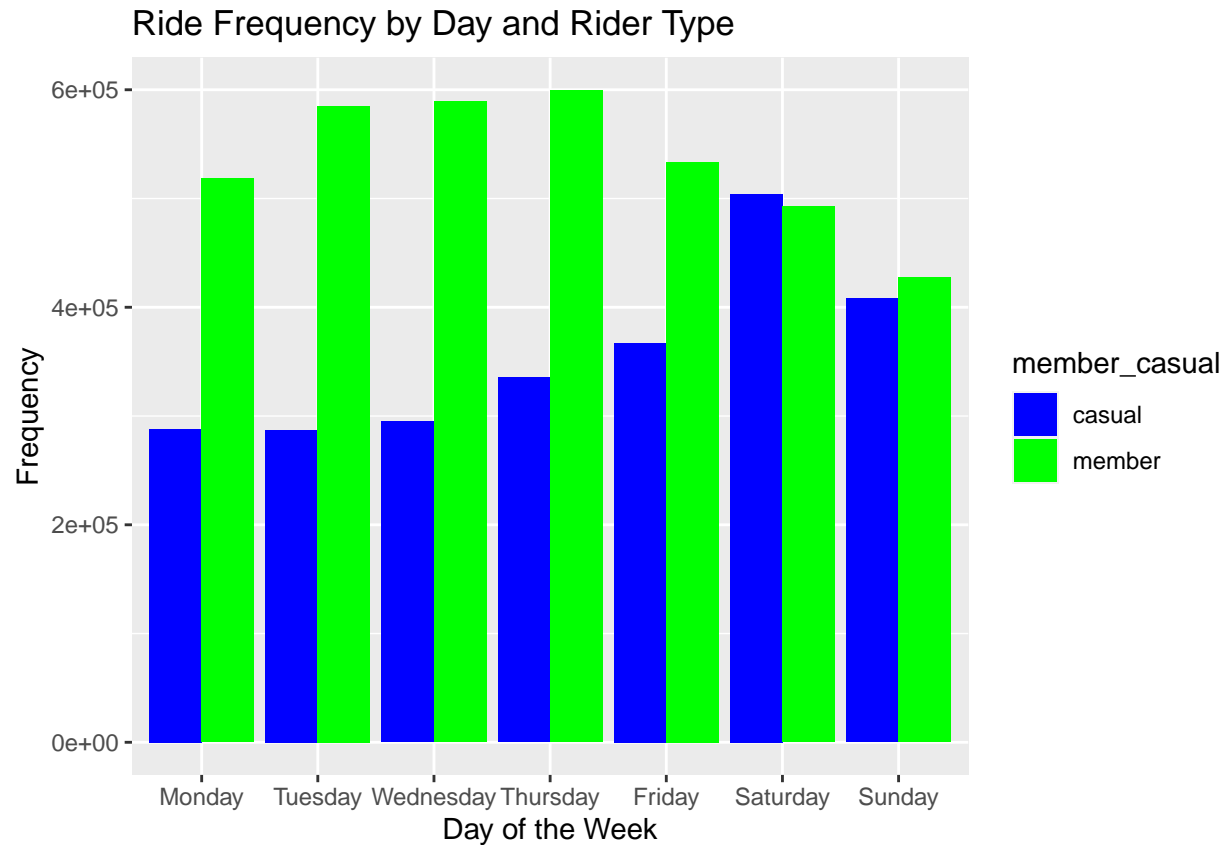
```
ggplot(casual_data, aes(x = day_of_the_week, y = ride_length)) +  
  geom_bar(stat = "summary", fun = "max", fill = "blue", alpha = 0.7) +  
  ggtitle("Casual Riders - Maximum Ride Length by Day of the Week")
```

```
## Don't know how to automatically pick scale for object of type <difftime>.  
## Defaulting to continuous.
```


Casual Riders – Maximum Ride Length by Day of the Week



```
ggplot(data, aes(x = factor(day_of_the_week), fill = member_casual)) +  
  geom_bar(position = "dodge") +  
  labs(x = "Day of the Week", y = "Frequency", title = "Ride Frequency by Day and Rider Type") +  
  scale_fill_manual(values = c("casual" = "blue", "member" = "green"))
```



For better understanding below is the mean ride length, maximum ride length, and most frequent day of the ride for casual riders and members.

```
print(paste("Mean ride length:", mean_value_casual))
```

Casual

```
## [1] "Mean ride length: 28.5461533688785"
```

```
print(paste("Maximum ride length:", max_value_casual))
```

```
## [1] "Maximum ride length: 41387.25"
```

```
print(paste("Most frequent day for ride:", most_frequent_casual))
```

```
## [1] "Most frequent day for ride: Saturday"
```

```
print(paste("Mean ride length:", mean_value_member))
```

Members

```
## [1] "Mean ride length: 12.4312071462249"
```

```
print(paste("Maximum ride length:", max_value_member))
```

```
## [1] "Maximum ride length: 1559.66666666667"
```

```
print(paste("Most frequent day for ride:", most_frequent_member))
```

```
## [1] "Most frequent day for ride: Thursday"
```

Frequently used stations for casual and members

```
station_frequency <- table(subset(data, member_casual == "casual" & !is.na(start_station_name))$start_s  
# Sort the frequency table in descending order  
top_5_stations_casual <- sort(station_frequency, decreasing = TRUE)  
  
#top_5_stations_casual <- head(names(station_frequency)[order(station_frequency, decreasing = TRUE)], n  
print(head(top_5_stations_casual, n=5))
```

Casual

```
##  
##      Streeter Dr & Grand Ave  DuSable Lake Shore Dr & Monroe St  
##                        60469                        33837  
##      Millennium Park      Michigan Ave & Oak St  
##                        26725                        26392  
## DuSable Lake Shore Dr & North Blvd  
##                        24499
```

```
station_frequency <- table(subset(data, member_casual == "member" & !is.na(start_station_name))$start_s  
top_5_stations_member <- sort(station_frequency, decreasing = TRUE)  
print(head(top_5_stations_member, n=5))
```

Members

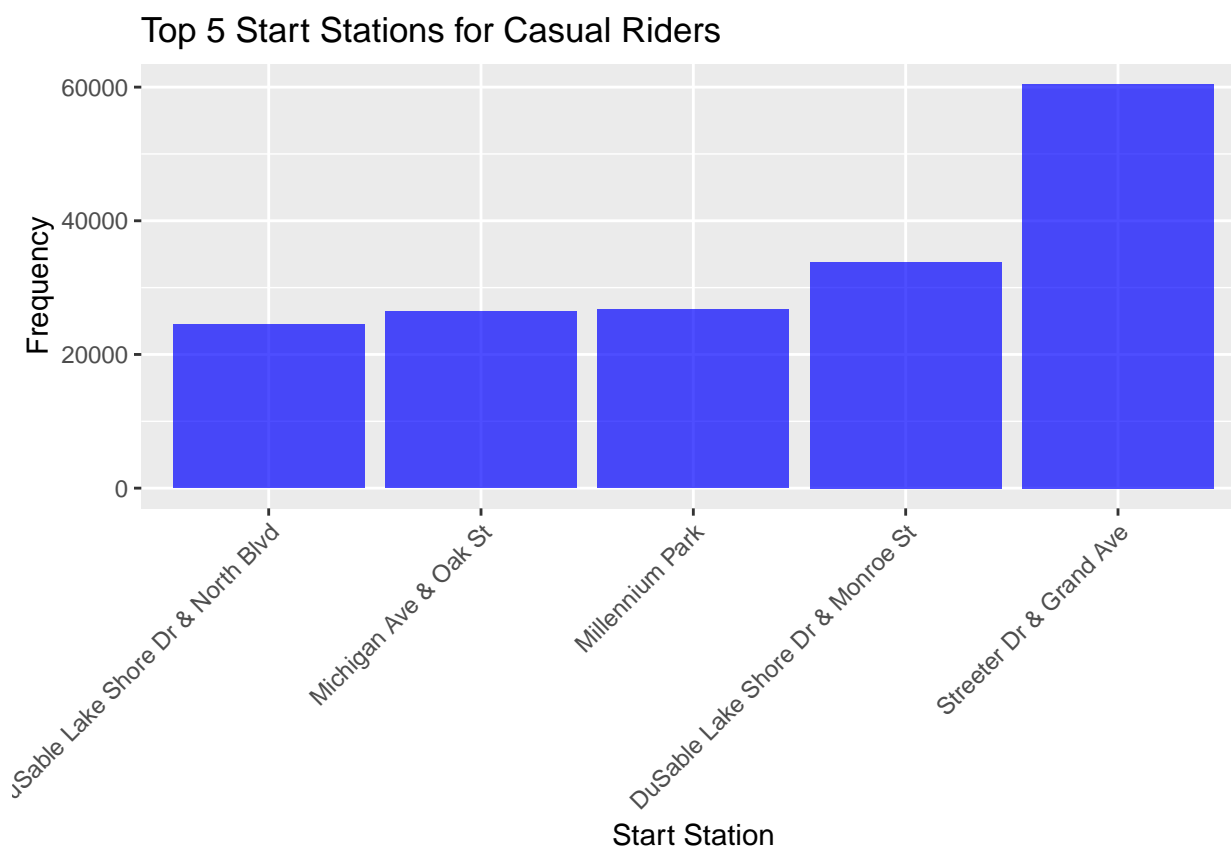
```
##  
##      Kingsbury St & Kinzie St      Clark St & Elm St  
##                        27427                        24937  
##      University Ave & 57th St      Wells St & Concord Ln  
##                        24073                        23598  
## Clinton St & Washington Blvd  
##                        23405
```

Plots for frequently used stations by casual and members

```
casual_data <- subset(data, member_casual == "casual" & !is.na(start_station_name))
station_frequency_casual <- table(casual_data$start_station_name)
top_5_stations_casual <- head(sort(station_frequency_casual, decreasing = TRUE), n = 5)

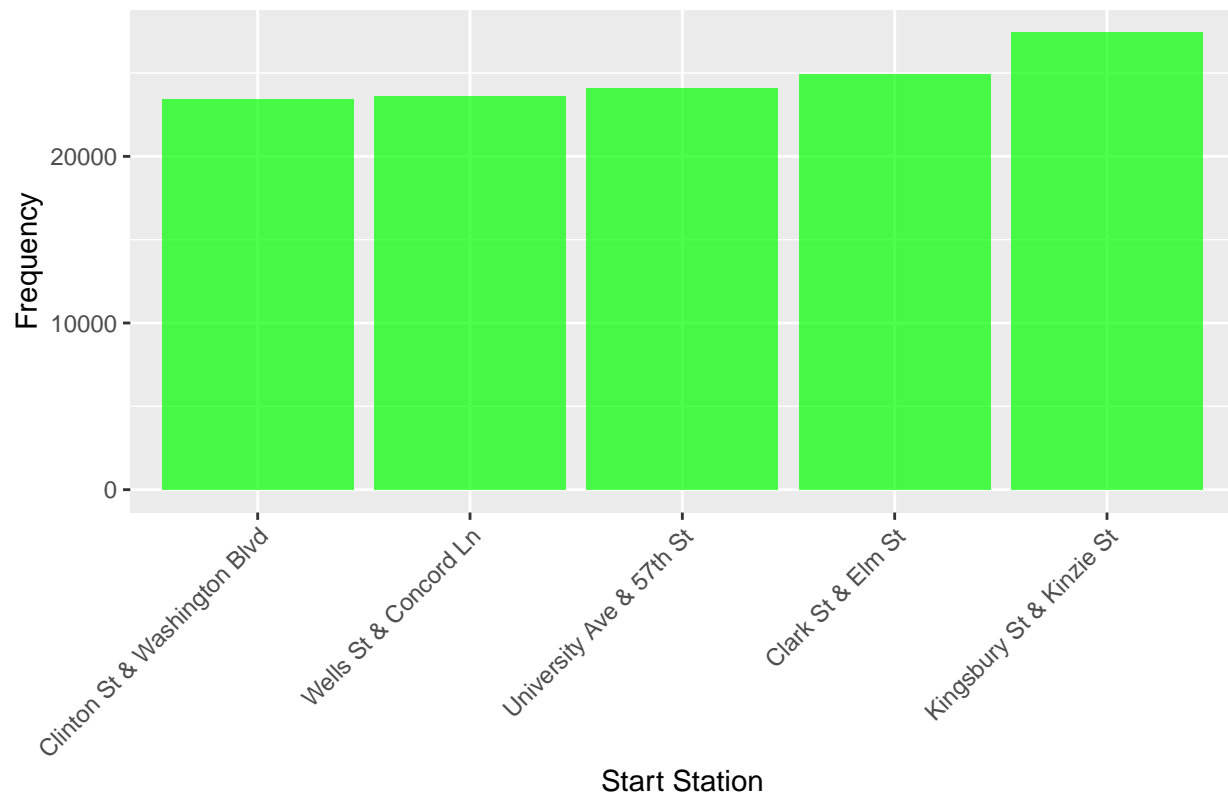
# Create a data frame for members
member_data <- subset(data, member_casual == "member" & !is.na(start_station_name))
station_frequency_member <- table(member_data$start_station_name)
top_5_stations_member <- head(sort(station_frequency_member, decreasing = TRUE), n = 5)

# Plot the top 5 start stations for casual riders
ggplot(data.frame(Station = names(top_5_stations_casual), Frequency = as.numeric(top_5_stations_casual)),
       aes(x = reorder(Station, Frequency), y = Frequency)) +
  geom_bar(stat = "identity", fill = "blue", alpha = 0.7) +
  labs(x = "Start Station", y = "Frequency", title = "Top 5 Start Stations for Casual Riders") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



```
# Plot the top 5 start stations for members
ggplot(data.frame(Station = names(top_5_stations_member), Frequency = as.numeric(top_5_stations_member)),
       aes(x = reorder(Station, Frequency), y = Frequency)) +
  geom_bar(stat = "identity", fill = "green", alpha = 0.7) +
  labs(x = "Start Station", y = "Frequency", title = "Top 5 Start Stations for Members") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Top 5 Start Stations for Members



Distinct start stations for casual and members

```
filtered_data_members <- data %>%
  filter(member_casual == "member")

# Get distinct values of a column from the filtered data
distinct_values_members <- filtered_data_members %>% distinct(start_station_name) %>% nrow()
print(distinct_values_members)
```

Members

```
## [1] 1542
```

```
filtered_data_casual <- data %>%
  filter(member_casual == "casual")

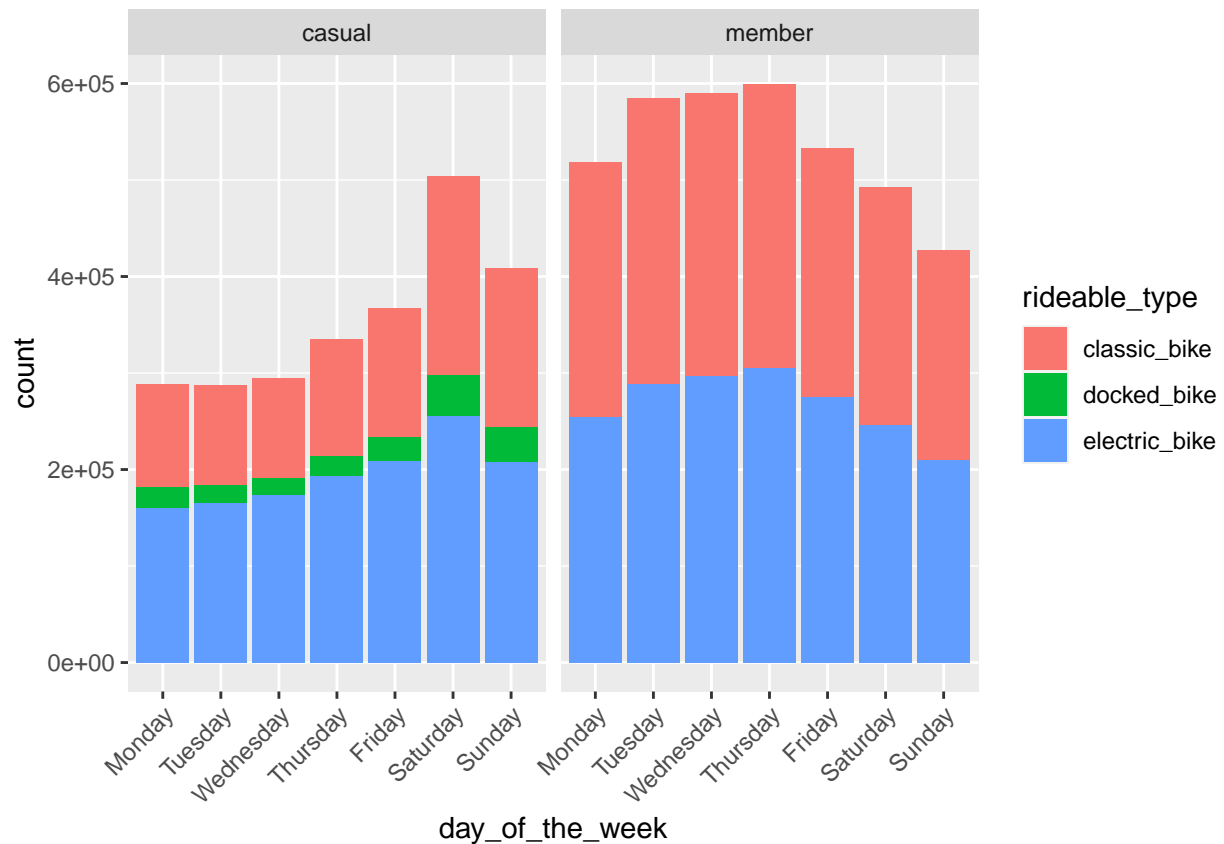
# Get distinct values of a column from the filtered data
distinct_values_casual <- filtered_data_casual %>% distinct(start_station_name) %>% nrow()
print(distinct_values_casual)
```

Casual Riders

```
## [1] 1646
```

So as it can be seen from the above numbers that members have used the bike service from 1542 different stations and for casual bike riders the distinct number of station is 1646.

Stacked column charts for different type of bikes used by members and casual riders



Conclusion

```
summary(data)
```

```
##      ride_id      rideable_type      started_at
## Length:6230310 Length:6230310 Min. :2022-04-01 00:01:48.00
## Class :character Class :character 1st Qu.:2022-06-22 23:26:45.50
## Mode :character Mode :character Median :2022-08-21 12:55:50.50
##                                     Mean :2022-09-10 07:29:08.01
##                                     3rd Qu.:2022-11-02 14:41:48.50
##                                     Max. :2023-04-30 23:59:05.00
##
##      ended_at      start_station_name start_station_id
## Min. :2022-04-01 00:02:15.00 Length:6230310 Length:6230310
```

```

## 1st Qu.:2022-06-22 23:56:19.00 Class :character Class :character
## Median :2022-08-21 13:20:40.50 Mode :character Mode :character
## Mean :2022-09-10 07:47:59.50
## 3rd Qu.:2022-11-02 14:56:04.25
## Max. :2023-05-03 10:37:12.00
##
## end_station_name end_station_id start_lat start_lng
## Length:6230310 Length:6230310 Min. :41.64 Min. : -87.84
## Class :character Class :character 1st Qu.:41.88 1st Qu.: -87.66
## Mode :character Mode :character Median :41.90 Median : -87.64
## Mean :41.90 Mean : -87.65
## 3rd Qu.:41.93 3rd Qu.: -87.63
## Max. :42.07 Max. : -87.52
##
## end_lat end_lng member_casual ride_length
## Min. : 0.00 Min. : -88.14 Length:6230310 Length:6230310
## 1st Qu.:41.88 1st Qu.: -87.66 Class :character Class :difftime
## Median :41.90 Median : -87.64 Mode :character Mode :numeric
## Mean :41.90 Mean : -87.65
## 3rd Qu.:41.93 3rd Qu.: -87.63
## Max. :42.37 Max. : 0.00
## NA's :6290 NA's :6290
## day_of_the_week
## Monday :806310
## Tuesday :872047
## Wednesday:884335
## Thursday :934729
## Friday :900373
## Saturday :996754
## Sunday :835762

```

As it can be seen from the above analysis that the casual riders have frequently used the bike services on weekends and on weekdays there has been a decline while the members tend to use it more on weekdays. This difference might suggest that casual riders use the bike services for leisure and recreational purposes such as going out on weekends or exploring the city. On the other hand the members might be using it for their daily commute like traveling to work. It can also be seen that Streeter Dr & Grand A station has been a frequent starting point for casual riders may be there are student dorms and they usually use the bike service on weekends.

These findings shows a pattern that can be used to help tailor marketing and promotional strategies for different types of riders and to encourage them to become regular members. We can offer special weekend membership plan or maybe offer them a free trial period which allow them to experience the full benefits of being a member without any commitment and during that time offer them perks like discounted rates, priority access to bikes or special offers available only to the members and also may be we can offer student discounts as well. This will attract the casual riders to become full time members.