GDAPC

Yeoh Jo Ann

2025-08-26

```
install.packages("tidyverse")
```

```
## Installing package into 'C:/Users/yeohj/AppData/Local/R/win-library/4.5'
## (as 'lib' is unspecified)
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\yeohj\AppData\Local\Temp\RtmpueJzqu\downloaded_packages
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(ggplot2)
```

GDA - Capstone Project

Overview

The director of marketing, Lily Moreno, believes the company's future success depends on maximizing the number of annual memberships. The team wants to understand how casual riders and annual members use Cyclistic bikes differently. And rather than creating a marketing campaign that target all-new customers, Moreno believes there is a solid opportunity to convert casual riders into members. The task assigned to you is **how do annual members and casual riders use Cyclistic bikes differently?"**

Business task

Identify the differences in using Cyclistic bikes between casual riders and annual members, including the trend and the factor.

```
df <- read.csv("Divvy_Trips_2019_Q1.csv")
```

Prepare Data Set

The dataset is a public data that people can use to explore how different custoemr types are suing Cyclistic bikes. Note: Cyclistic is a fictional company and the dataset is for the purposes of case study. The data has been made available by Motive International Inc. under this license

The dataset was downloaded in csv file type and the initial exploration was taken to understand the basic structure and identify the possible errors using spreadsheet.

Furthermore, the data set only contain the first quarter of year 2019 which is from 1 January 2019 until 31 March 2019.

Metadata

The following table show the metadata for the data set.

Field	Data Type	Notes
trip_id	number	The unique field for each record
start_time	string	The start time for each trip (DD/MM/YYY HH:MM)
end_time	string	The end time for each trip (DD/MM/YYY HH:MM)
bikeid	number	The bike id for each trip
tripduration	number	The time for each trip (in seconds)
from_station_id	number	The station id the bike start from
from_station_name	string	The station name the bike start from
to_station_id	number	The station id the bike is returned
to_station_name	string	The station name the bike is returned

Field	Data Type	Notes
usertype	string	The user type (Subscriber/Customer) Customer: Non annual member, Subscriber: Annual member.
gender	string	The user gender (Make/Female)
birthyear	number	The user's birthyear

Process Data Set

In spreadsheet, the duplicate records issue, the missing value issue, and incorrect spelling issue were checked. There is no duplicated record, and incorrect spelling for each field. While for the *gender* and *birthyear* field, there are missing values.

The csv file then was loaded into the R Studio Posit Cloud.

```
df <- read.csv("Divvy_Trips_2019_Q1.csv")
head(df)
```

```
##    trip_id     start_time       end_time bikeid tripduration from_station_id
## 1 21742443 1/01/2019 0:04 1/01/2019 0:11   2167          390              199
## 2 21742444 1/01/2019 0:08 1/01/2019 0:15   4386          441               44
## 3 21742445 1/01/2019 0:13 1/01/2019 0:27   1524          829               15
## 4 21742446 1/01/2019 0:13 1/01/2019 0:43    252         1783              123
## 5 21742447 1/01/2019 0:14 1/01/2019 0:20   1170          364              173
## 6 21742448 1/01/2019 0:15 1/01/2019 0:19   2437          216               98
##                     from_station_name to_station_id
## 1              Wabash Ave & Grand Ave            84
## 2              State St & Randolph St           624
## 3                Racine Ave & 18th St           644
## 4      California Ave & Milwaukee Ave           176
## 5 Mies van der Rohe Way & Chicago Ave            35
## 6          LaSalle St & Washington St            49
##                   to_station_name   usertype gender birthyear
## 1      Milwaukee Ave & Grand Ave Subscriber   Male      1989
## 2 Dearborn St & Van Buren St (*) Subscriber Female      1990
## 3  Western Ave & Fillmore St (*) Subscriber Female      1994
## 4              Clark St & Elm St Subscriber   Male      1993
## 5        Streeter Dr & Grand Ave Subscriber   Male      1994
## 6        Dearborn St & Monroe St Subscriber Female      1983
```

The *gender* and *birthyear* fields were removed as both are not suitable for the business task.

```
df_fil <- select(df,-c(gender,birthyear))
head(df_fil, n = 10)
```

```
##     trip_id     start_time       end_time bikeid tripduration from_station_id
## 1  21742443 1/01/2019 0:04 1/01/2019 0:11   2167          390              199
## 2  21742444 1/01/2019 0:08 1/01/2019 0:15   4386          441               44
## 3  21742445 1/01/2019 0:13 1/01/2019 0:27   1524          829               15
## 4  21742446 1/01/2019 0:13 1/01/2019 0:43    252         1783              123
```

```
## 5  21742447 1/01/2019 0:14 1/01/2019 0:20   1170          364             173
## 6  21742448 1/01/2019 0:15 1/01/2019 0:19   2437          216              98
## 7  21742449 1/01/2019 0:16 1/01/2019 0:19   2708          177              98
## 8  21742450 1/01/2019 0:18 1/01/2019 0:20   2796          100             211
## 9  21742451 1/01/2019 0:18 1/01/2019 0:47   6205         1727             150
## 10 21742452 1/01/2019 0:19 1/01/2019 0:24   3939          336             268
##                      from_station_name to_station_id
## 1               Wabash Ave & Grand Ave            84
## 2               State St & Randolph St           624
## 3                 Racine Ave & 18th St           644
## 4       California Ave & Milwaukee Ave           176
## 5  Mies van der Rohe Way & Chicago Ave            35
## 6           LaSalle St & Washington St            49
## 7           LaSalle St & Washington St            49
## 8               St. Clair St & Erie St           142
## 9           Fort Dearborn Dr & 31st St           148
## 10          Lake Shore Dr & North Blvd           141
##                    to_station_name   usertype
## 1       Milwaukee Ave & Grand Ave Subscriber
## 2  Dearborn St & Van Buren St (*) Subscriber
## 3   Western Ave & Fillmore St (*) Subscriber
## 4               Clark St & Elm St Subscriber
## 5         Streeter Dr & Grand Ave Subscriber
## 6         Dearborn St & Monroe St Subscriber
## 7         Dearborn St & Monroe St Subscriber
## 8            McClurg Ct & Erie St Subscriber
## 9              State St & 33rd St Subscriber
## 10         Clark St & Lincoln Ave Subscriber
```

The *start_time* and *end_time* were corrected to the correct data type.

```
df_fil$start_time <- dmy_hm(df_fil$start_time)
df_fil$end_time <- dmy_hm(df_fil$end_time)
```

The new fields were created to extract the information for *start_time* and *end_time*.

```
# start time
df_fil$start_hour <- hour(df_fil$start_time)
df_fil$start_weekday <- weekdays(df_fil$start_time)

# end time
df_fil$end_hour <- hour(df_fil$end_time)
df_fil$end_weekday <- weekdays(df_fil$end_time)
colnames(df_fil)
```

```
##  [1] "trip_id"           "start_time"        "end_time"
##  [4] "bikeid"            "tripduration"      "from_station_id"
##  [7] "from_station_name" "to_station_id"     "to_station_name"
## [10] "usertype"          "start_hour"        "start_weekday"
## [13] "end_hour"          "end_weekday"
```

```r
typeof(df_fil$start_time)
```

```
## [1] "double"
```

A new field which is named as *ride_length* was created to calculate the tripduration in minutes.

```r
df_fil$ride_length <- as.numeric(difftime(df_fil$end_time, df_fil$start_time, units = "mins"))
head(df_fil$ride_length, n =10)
```

```
##  [1]  7  7 14 30  6  4  3  2 29  5
```

Analyse

Basic information

```r
colnames(df_fil)
```

```
##  [1] "trip_id"           "start_time"        "end_time"
##  [4] "bikeid"            "tripduration"      "from_station_id"
##  [7] "from_station_name" "to_station_id"     "to_station_name"
## [10] "usertype"          "start_hour"        "start_weekday"
## [13] "end_hour"          "end_weekday"       "ride_length"
```

```r
str(df_fil)
```

```
## 'data.frame':    365069 obs. of  15 variables:
##  $ trip_id          : int  21742443 21742444 21742445 21742446 21742447 21742448 21742449 21742450 2
##  $ start_time       : POSIXct, format: "2019-01-01 00:04:00" "2019-01-01 00:08:00" ...
##  $ end_time         : POSIXct, format: "2019-01-01 00:11:00" "2019-01-01 00:15:00" ...
##  $ bikeid           : int  2167 4386 1524 252 1170 2437 2708 2796 6205 3939 ...
##  $ tripduration     : num  390 441 829 1783 364 ...
##  $ from_station_id  : int  199 44 15 123 173 98 98 211 150 268 ...
##  $ from_station_name: chr  "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave & 18th St"
##  $ to_station_id    : int  84 624 644 176 35 49 49 142 148 141 ...
##  $ to_station_name  : chr  "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)" "Western Ave
##  $ usertype         : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ start_hour       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ start_weekday    : chr  "Tuesday" "Tuesday" "Tuesday" "Tuesday" ...
##  $ end_hour         : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ end_weekday      : chr  "Tuesday" "Tuesday" "Tuesday" "Tuesday" ...
##  $ ride_length      : num  7 7 14 30 6 4 3 2 29 5 ...
```

```r
dim(df_fil)
```

```
## [1] 365069     15
```

From the above information, we noticed that the data set has a total **365096 records and 15 fields**.

```r
# How many start-station?
length(unique(df_fil$to_station_id))
```

```
## [1] 600
```

```r
length(unique(df_fil$from_station_id))
```

```
## [1] 594
```

From the information above, there is at least 600 stations recorded in this data set.

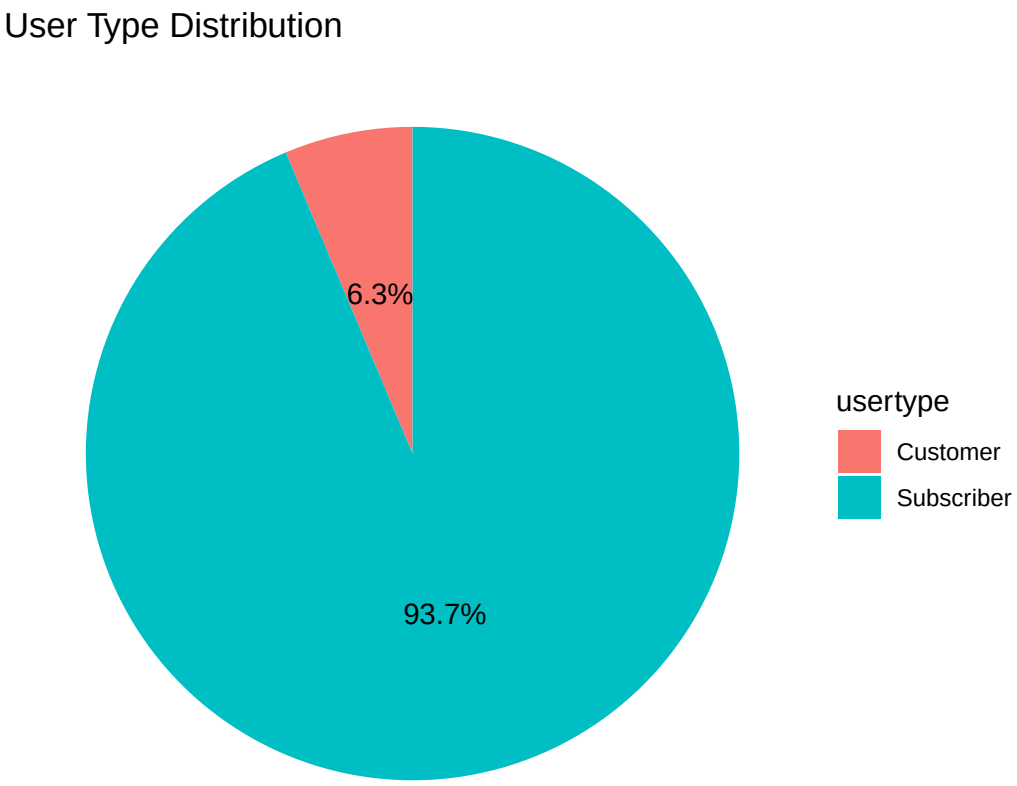
Usertype Distribution

A pie chart was created to understand the propotion of the subscribers and customers.

```r
# User distribution percentage
user_distribution <- table(df_fil$usertype)
prop.table(user_distribution) * 100
```

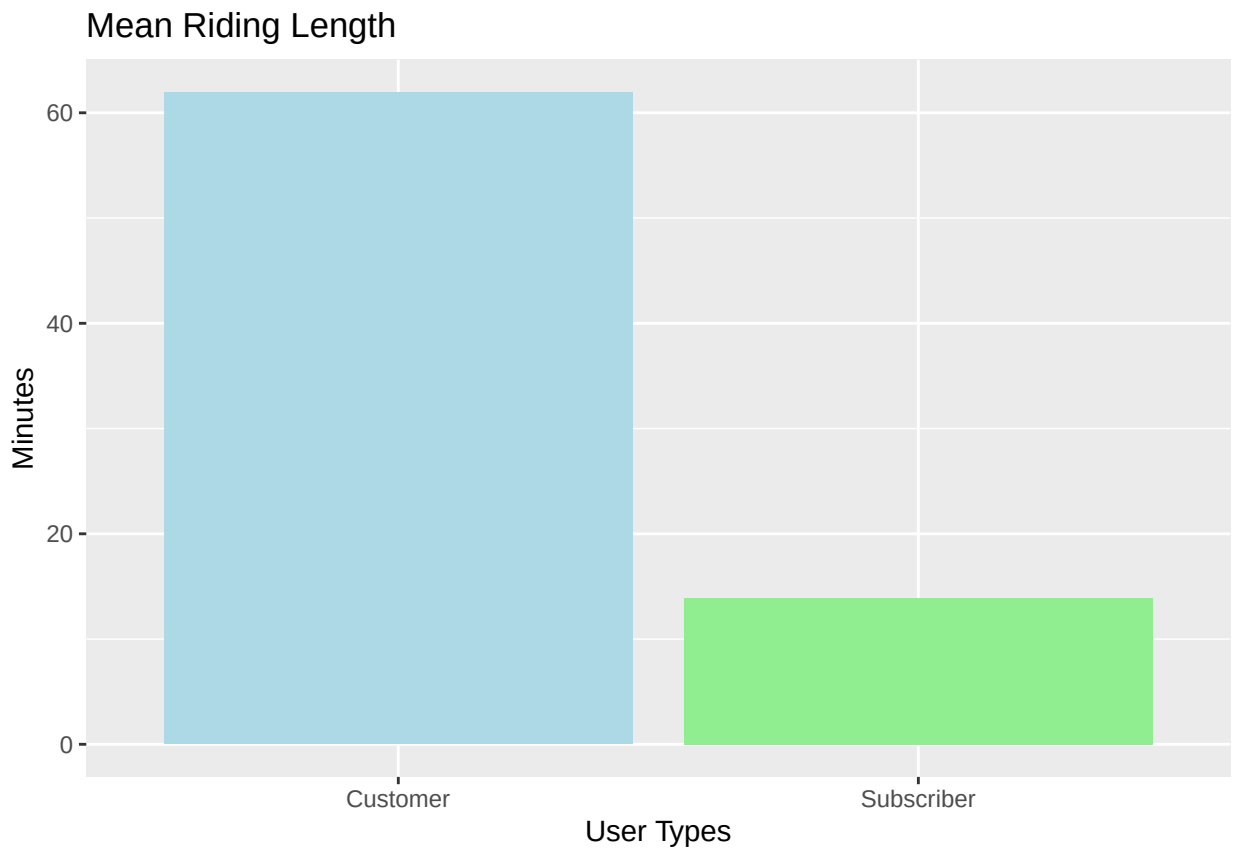
```
##
##   Customer Subscriber
##   6.344828  93.655172
```

```r
user_df <- data.frame(
  usertype = names(user_distribution),
  count = as.numeric(user_distribution)
)
user_df$percentage <- round(user_df$count/sum(user_df$count)*100, 1)

ggplot(user_df, aes(x="", y=count, fill=usertype)) +
  geom_bar(stat="identity", width=1) +
  coord_polar("y", start=0) +
  geom_text(aes(label=paste0(percentage,"%")),
            position=position_stack(vjust=0.5)) +
  labs(title = "User Type Distribution") +
  theme_void()
```

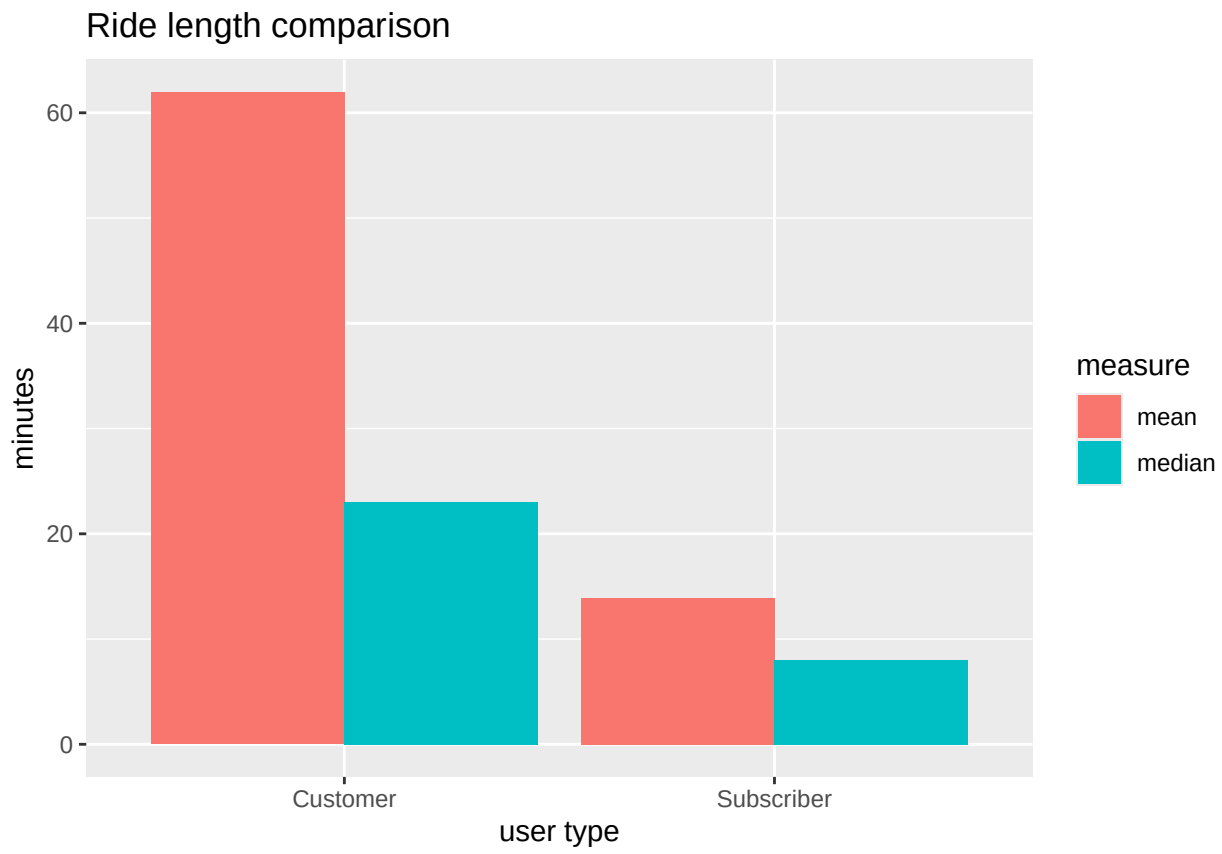


Basic statistic for ride length

```
df_user_sum <- aggregate(ride_length ~ usertype, df_fil, summary)
mean_data <- data.frame(
  usertype = df_user_sum$usertype,
  mean_time = df_user_sum$ride_length[,"Mean"]
)

ggplot(mean_data, aes(x = usertype, y = mean_time)) +
  geom_col(fill = c("lightblue", "lightgreen")) +
  labs(title = "Mean Riding Length", x = "User Types", y = "Minutes")
```

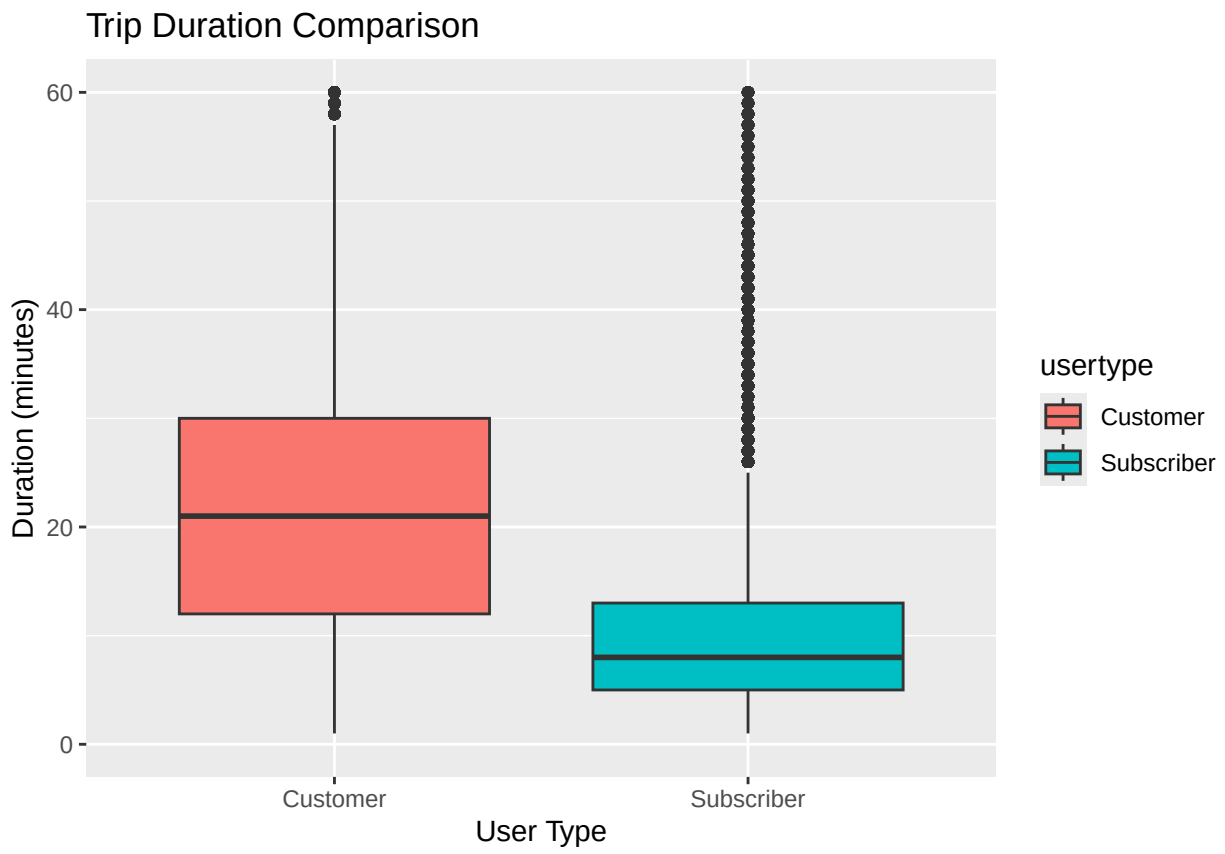



```
# Comparison of mean and median
compare_data <- data.frame(
  usertype = rep(df_user_sum$usertype, 2),
  measure = rep(c("mean", "median"), each = nrow(df_user_sum)),
  time = c(df_user_sum$ride_length[,"Mean"], df_user_sum$ride_length[,"Median"])
)

ggplot(compare_data, aes(x = usertype, y = time, fill = measure)) +
  geom_col(position = "dodge") +
  labs(title = "Ride length comparison", x = "user type", y = "minutes")
```



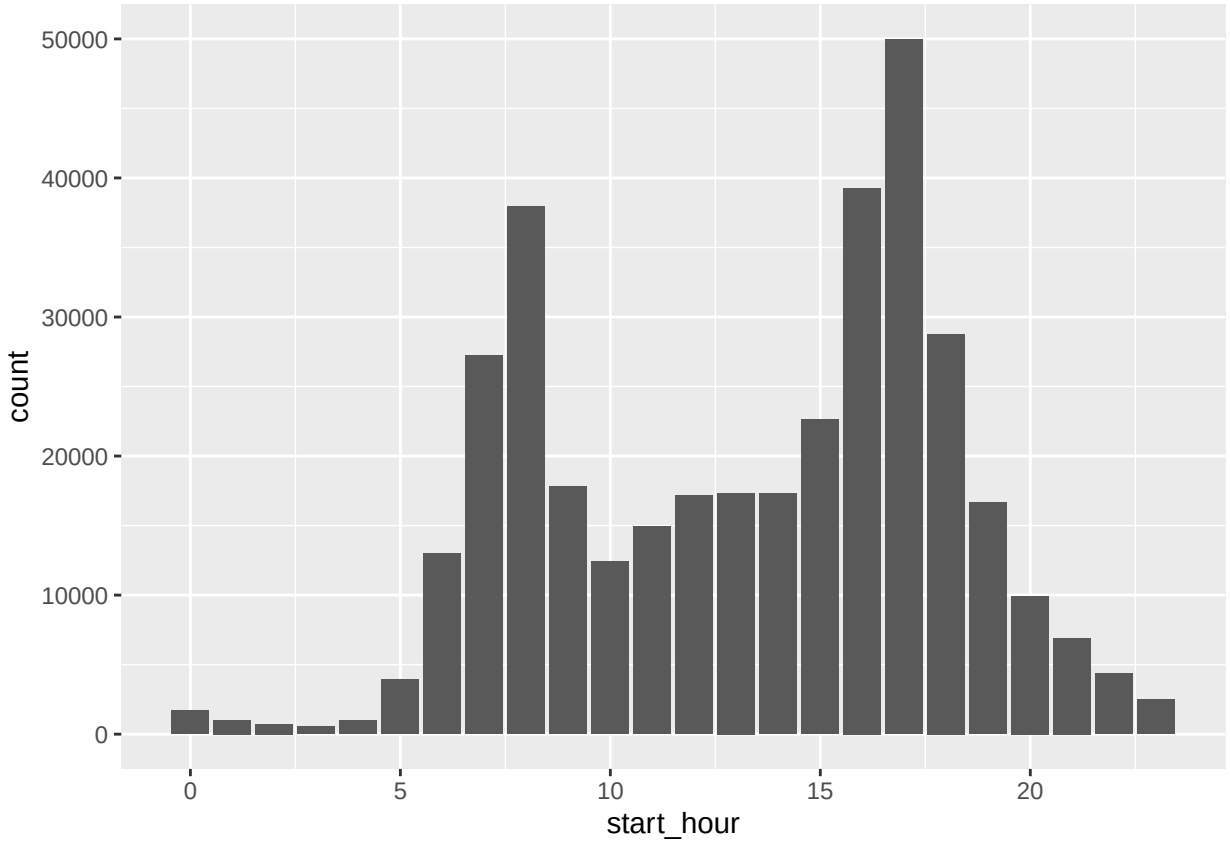

```
# Box plot
ggplot(df_fil, aes(x=usertype, y=ride_length, fill=usertype)) +
  geom_boxplot() +
  ylim(0, 60) +
  labs(title="Trip Duration Comparison",
       x="User Type", y="Duration (minutes)")
```

```
## Warning: Removed 4225 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



Frequency of bike riding for each day (start)

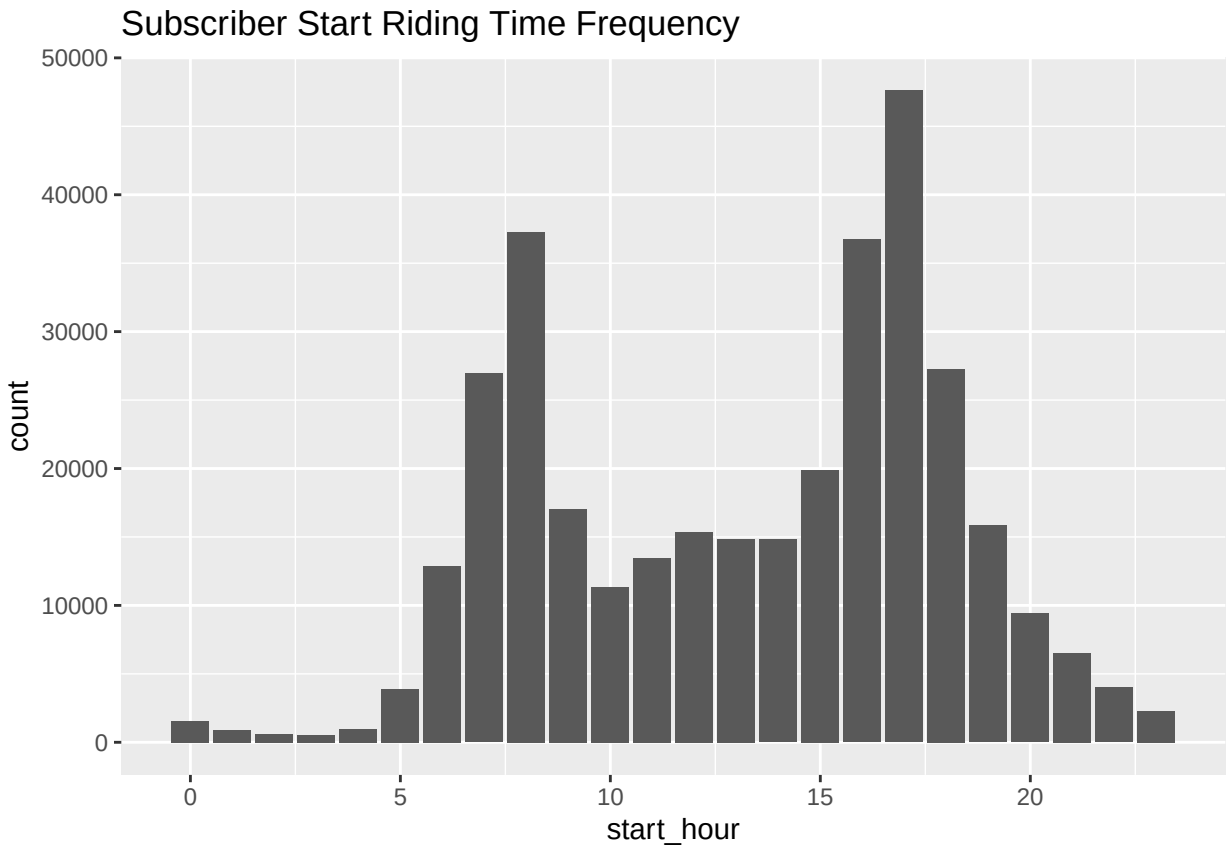
```
hourly_usage <- df_fil %>%
  group_by(start_hour) %>%
  summarise(count = n())
ggplot(data=hourly_usage) +geom_col((mapping = aes(x=start_hour, y=count)))
```




```
hourly_usage
```

```
## # A tibble: 24 x 2
##    start_hour count
##         <int> <int>
##  1          0  1694
##  2          1  1008
##  3          2   725
##  4          3   562
##  5          4   993
##  6          5  3938
##  7          6 13015
##  8          7 27218
##  9          8 37930
## 10          9 17791
## # i 14 more rows
```

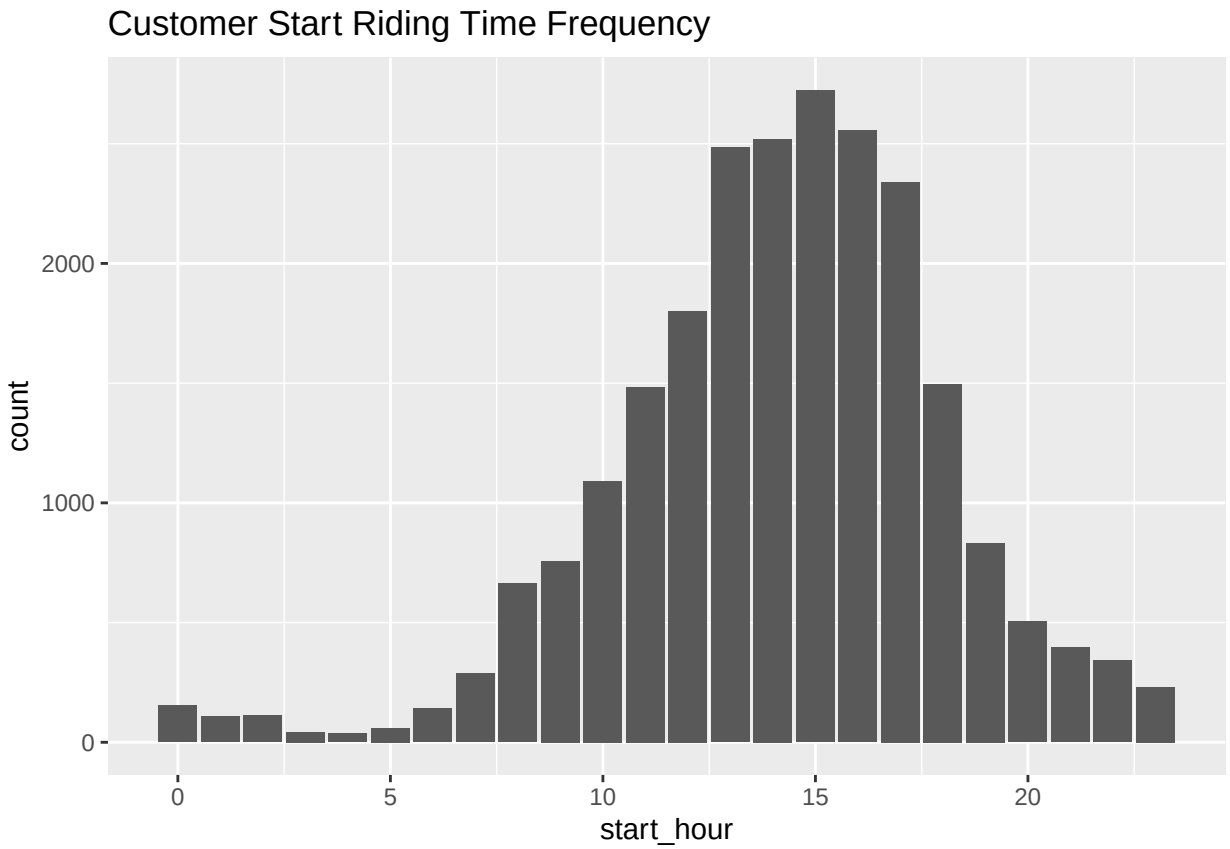
```
# Subscriber hourly usage (start)
sub_hourly_usage <- df_fil %>%
  filter(usertype == "Subscriber") %>%
  group_by(start_hour) %>%
  summarise(count=n())
ggplot(data=sub_hourly_usage) +geom_col((mapping = aes(x=start_hour, y=count)))+
  labs(title = "Subscriber Start Riding Time Frequency")
```



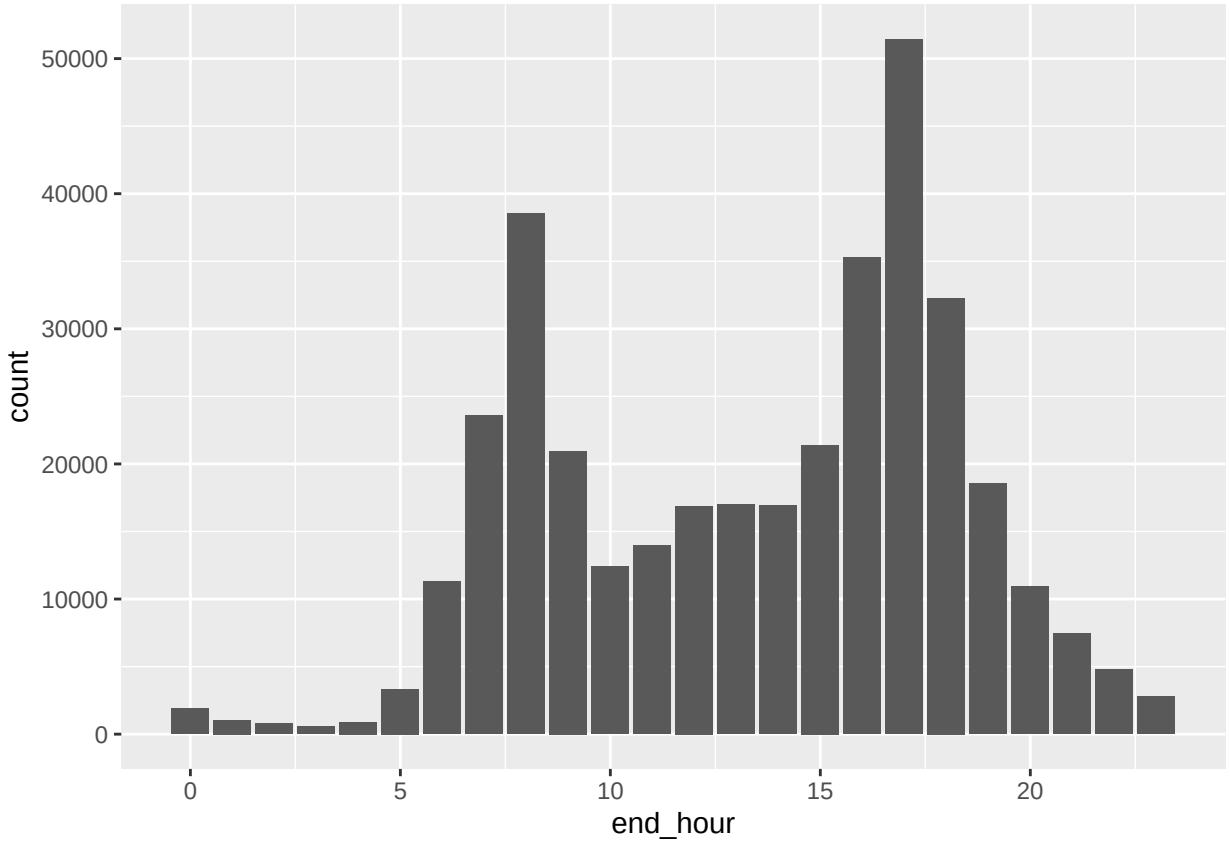

```
sub_hourly_usage
```

```
## # A tibble: 24 x 2
##    start_hour count
##         <int> <int>
##  1          0  1541
##  2          1   901
##  3          2   614
##  4          3   518
##  5          4   956
##  6          5  3878
##  7          6 12874
##  8          7 26930
##  9          8 37266
## 10          9 17033
## # i 14 more rows
```

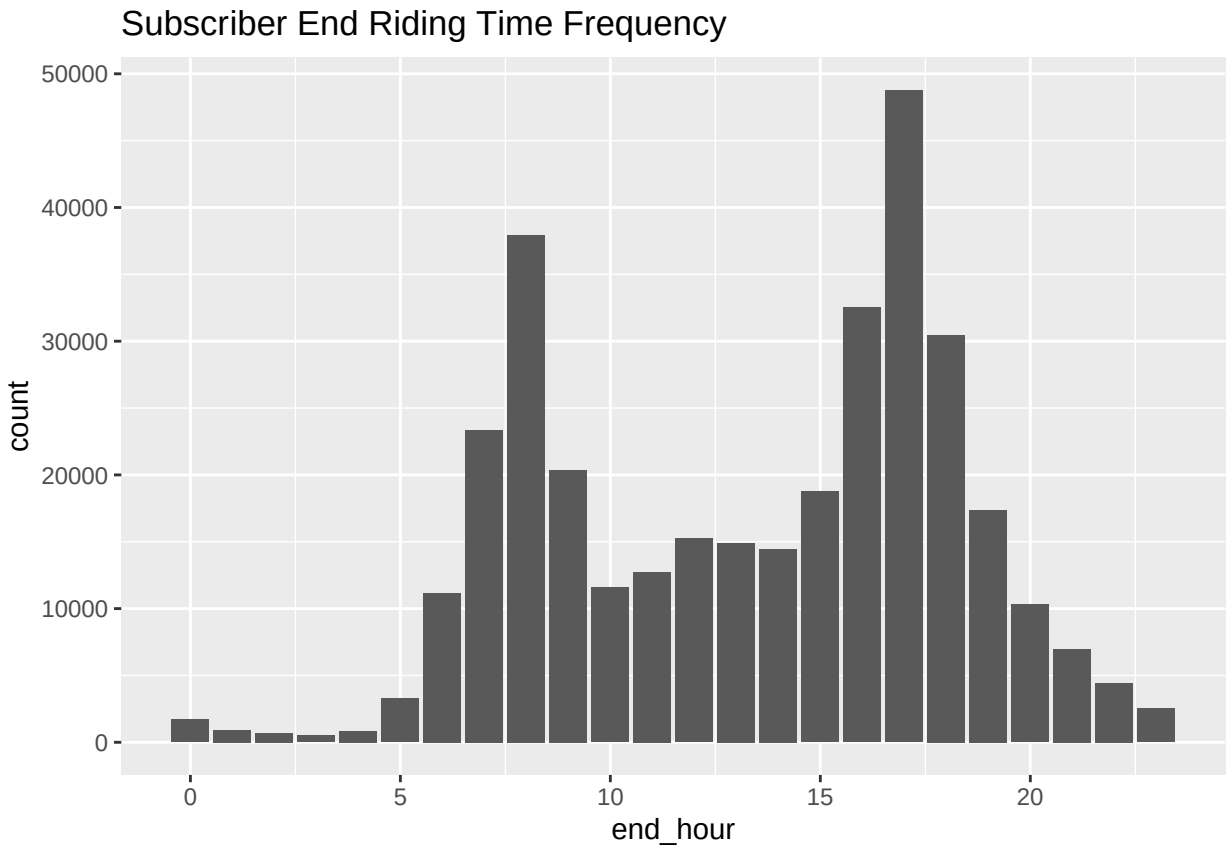
```
# Customer hourly usage (start)
cus_hourly_usage <- df_fil %>%
  filter(usertype == "Customer") %>%
  group_by(start_hour) %>%
  summarise(count=n())
ggplot(data=cus_hourly_usage) +geom_col((mapping = aes(x= start_hour, y=count)))+
  labs(title= "Customer Start Riding Time Frequency")
```



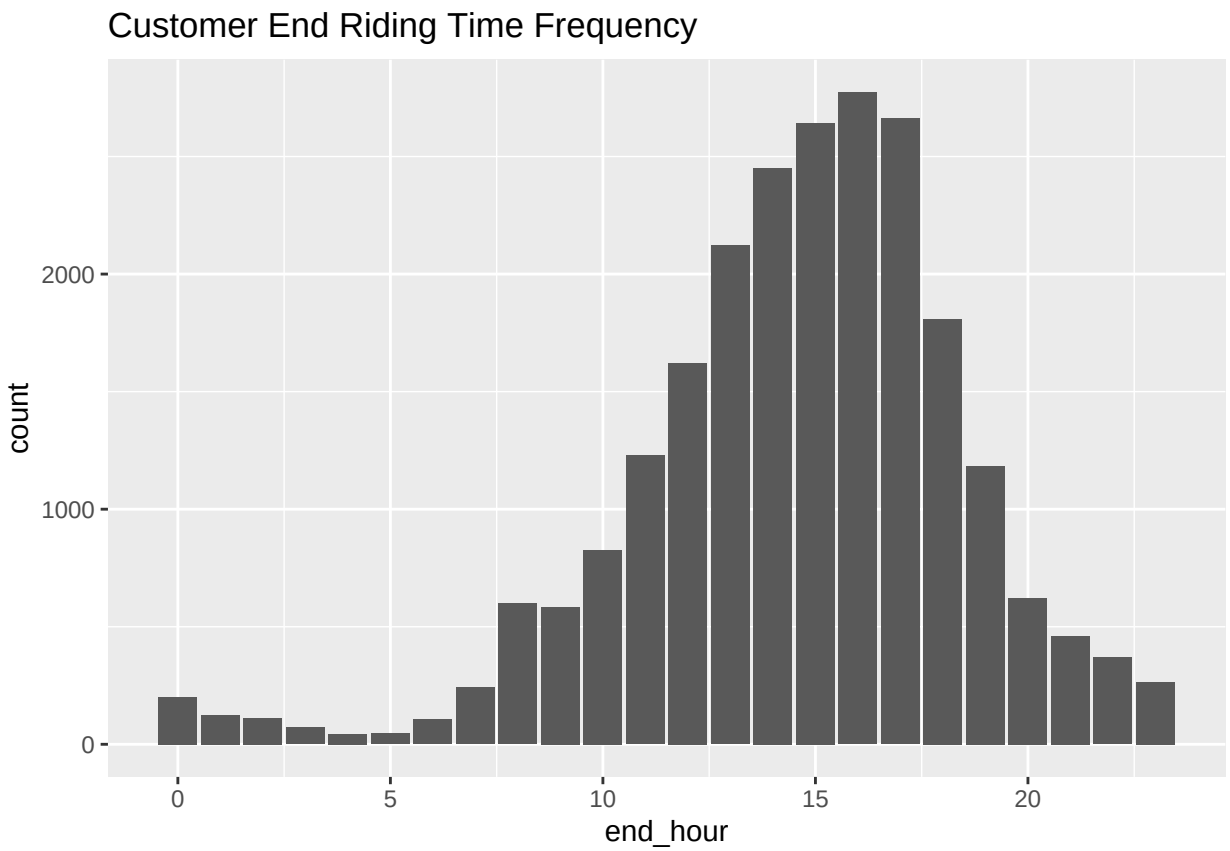

```
# Regardless the usertype, what are the frequency of bike riding for each day (end)
end_hourly_usage <- df_fil %>%
  group_by(end_hour) %>%
  summarise(count = n())
ggplot(data=end_hourly_usage) +geom_col((mapping = aes(x=end_hour, y=count)))
```




```
# Subscriber hourly usage (end)
end_sub_hourly_usage <- df_fil %>%
  filter(usertype == "Subscriber") %>%
  group_by(end_hour) %>%
  summarise(count=n())
ggplot(data=end_sub_hourly_usage) +geom_col((mapping = aes(x=end_hour, y=count)))+
  labs(title = "Subscriber End Riding Time Frequency")
```

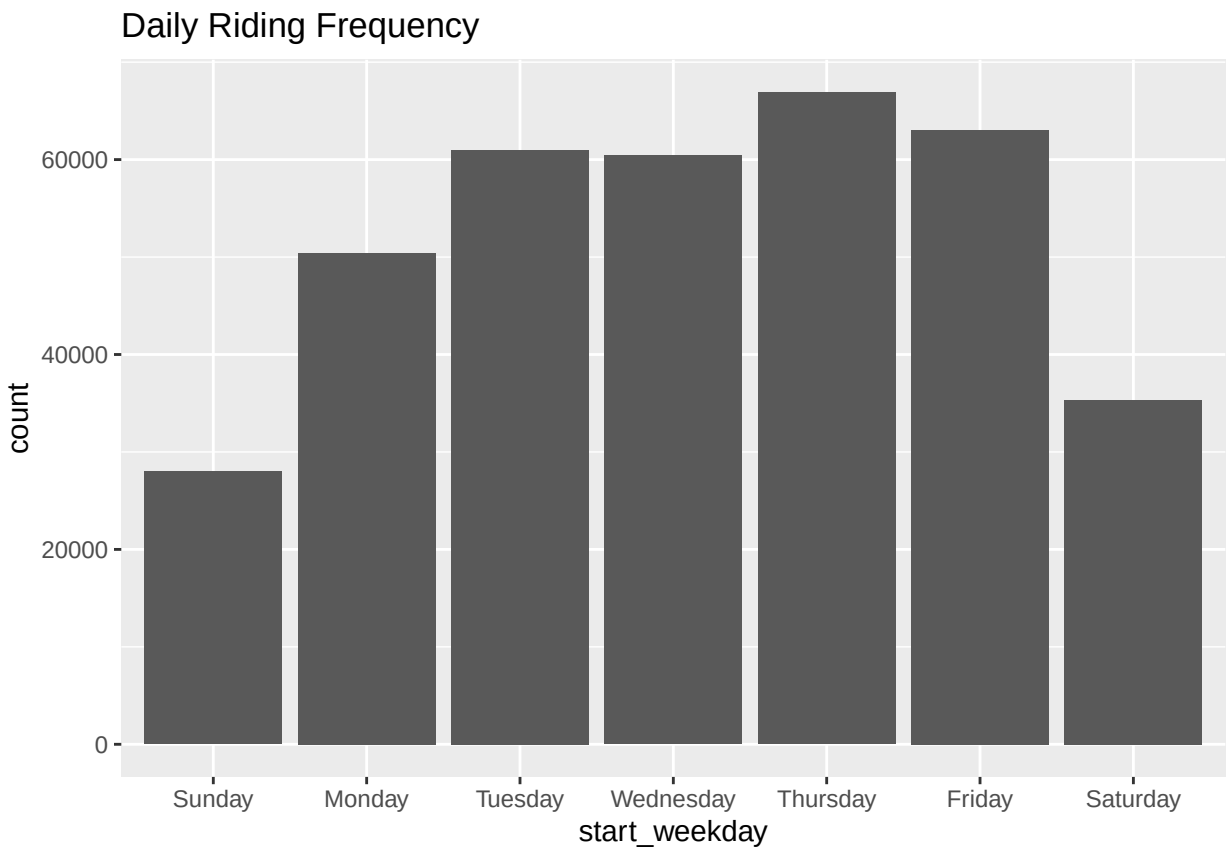



```
# Customer hourly usage (end)
end_cus_hourly_usage <- df_fil %>%
  filter(usertype == "Customer") %>%
  group_by(end_hour) %>%
  summarise(count=n())
ggplot(data=end_cus_hourly_usage) +geom_col((mapping = aes(x=end_hour, y=count)))+
  labs(title= "Customer End Riding Time Frequency")
```

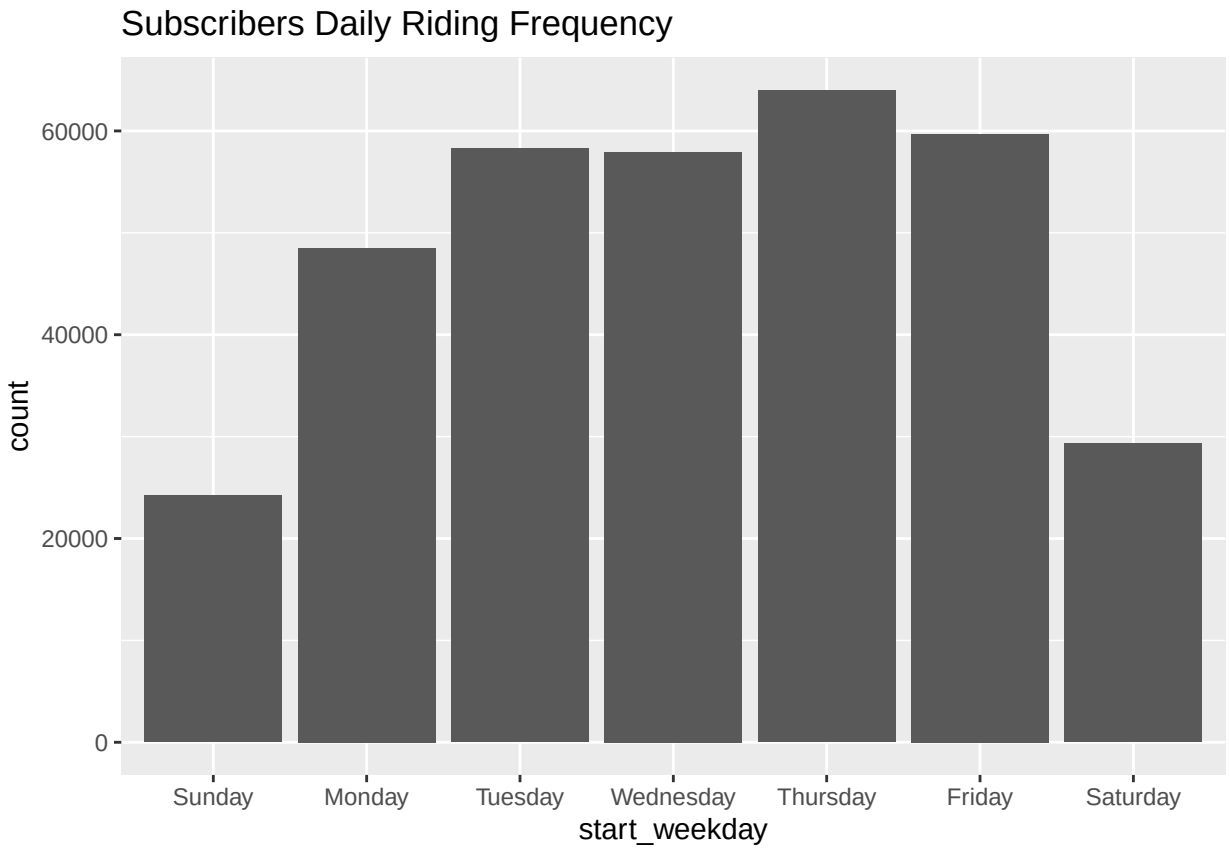


Frequency of bike riding for each week

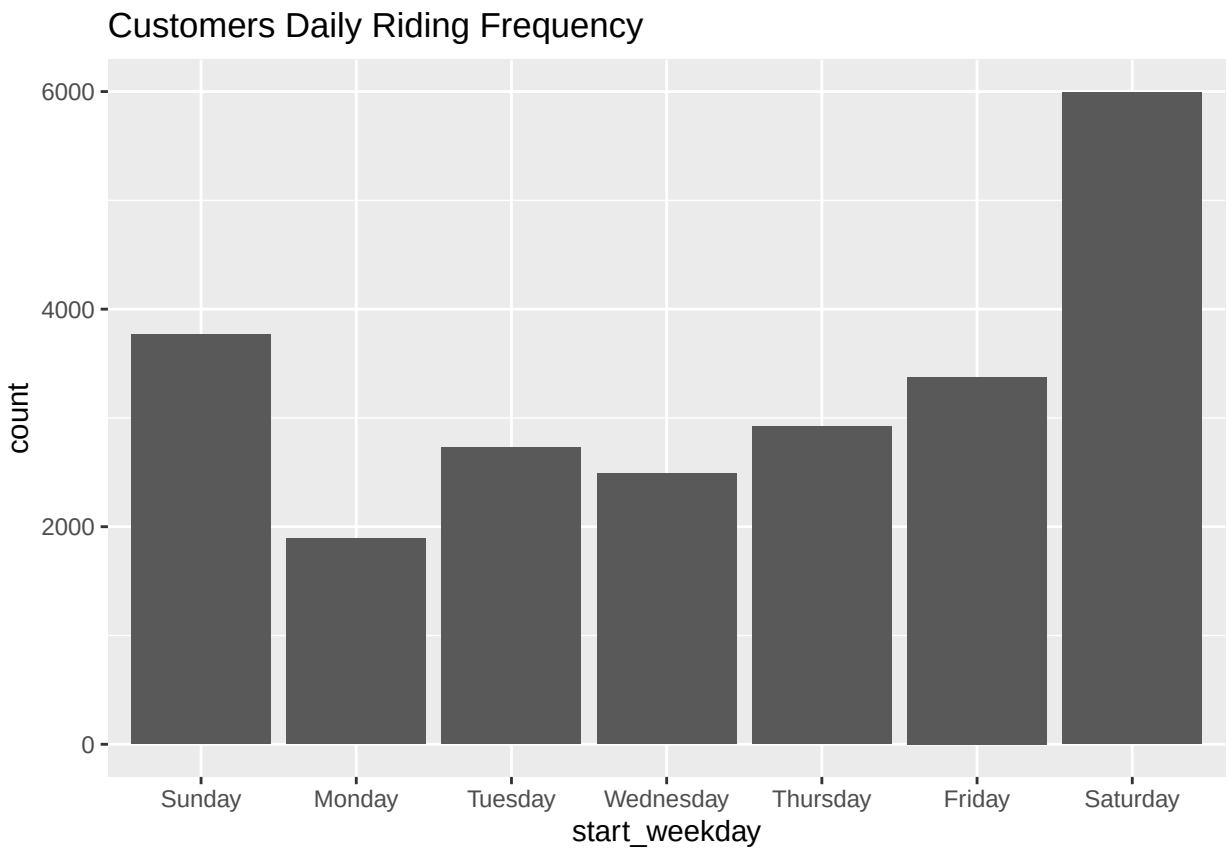
```
# Regardless usertype, what are the riding frequency for each day
daily_usage <-df_fil %>%
  group_by(start_weekday) %>%
  summarise(count=n())
ggplot(data=daily_usage) +geom_col((mapping = aes(x= start_weekday, y=count)))+
  scale_x_discrete(limits = c("Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"))+
  labs(title= "Daily Riding Frequency")
```




```
#  what are the riding frequency for each day for subscriber
sub_daily_usage <-df_fil %>%
  filter (usertype == "Subscriber") %>%
  group_by(start_weekday) %>%
  summarise(count=n())
ggplot(data=sub_daily_usage) +geom_col((mapping = aes(x= start_weekday, y=count)))+
  scale_x_discrete(limits = c("Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"))+
  labs(title= "Subscribers Daily Riding Frequency")
```

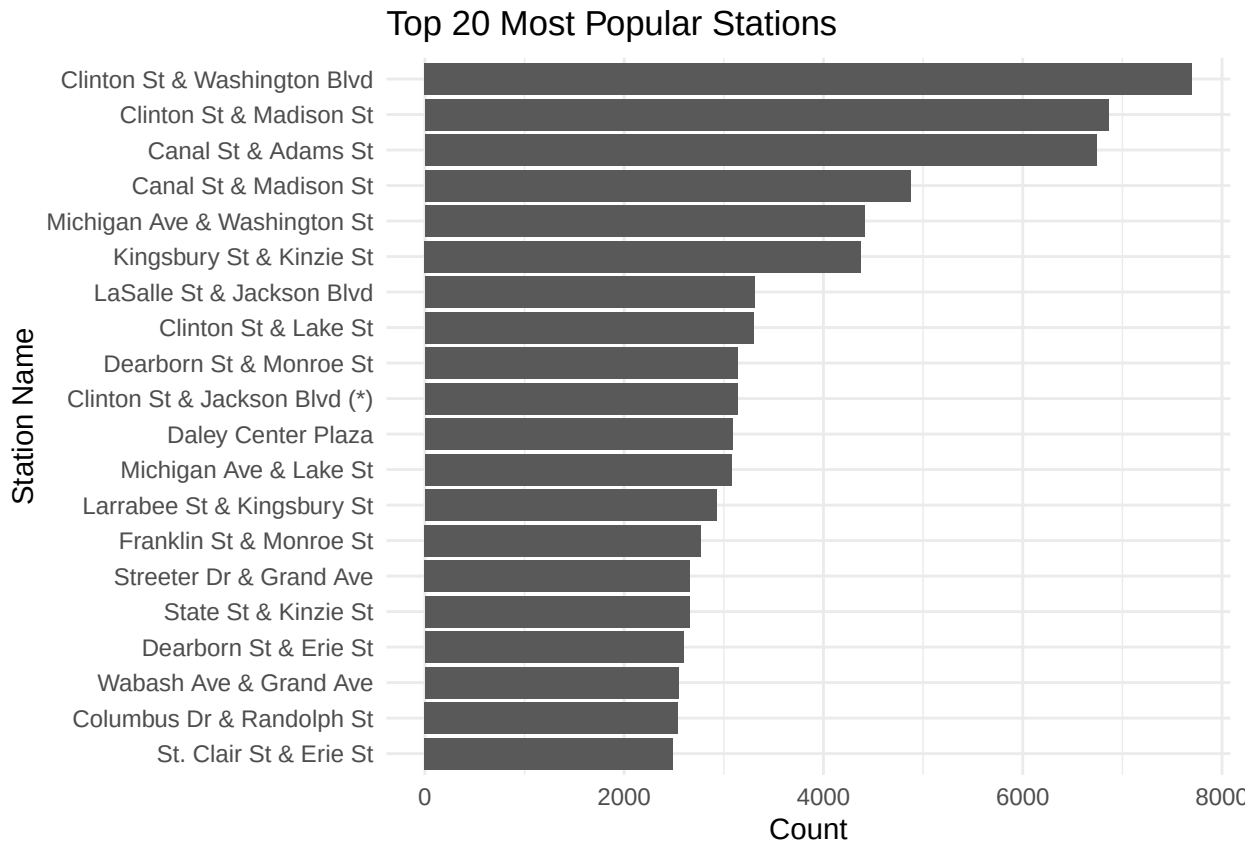



```
#  what are the riding frequency for each day for customer
cus_daily_usage <-df_fil %>%
  filter (usertype == "Customer") %>%
  group_by(start_weekday) %>%
  summarise(count=n())
ggplot(data=cus_daily_usage) +geom_col((mapping = aes(x= start_weekday, y=count)))+
  scale_x_discrete(limits = c("Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"))+
  labs(title= "Customers Daily Riding Frequency")
```

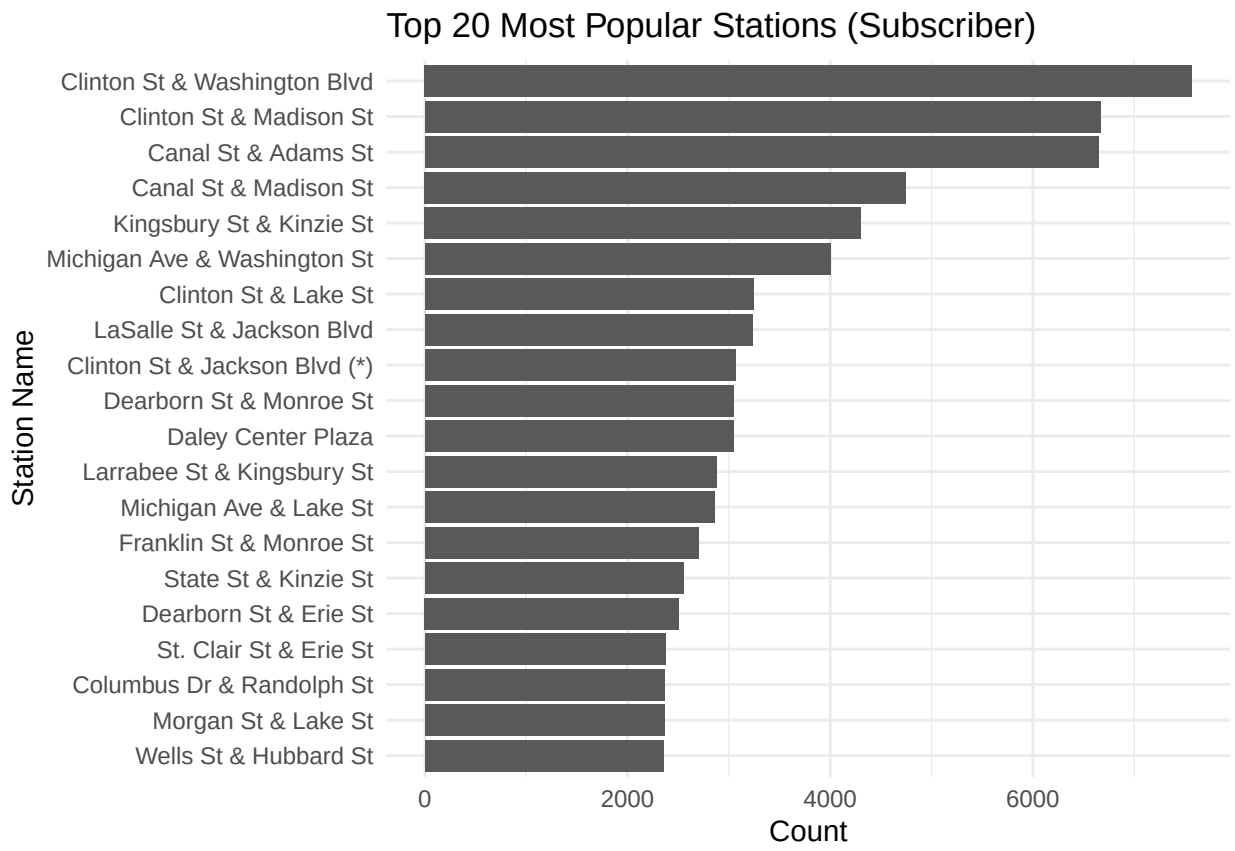


Popularity of each station

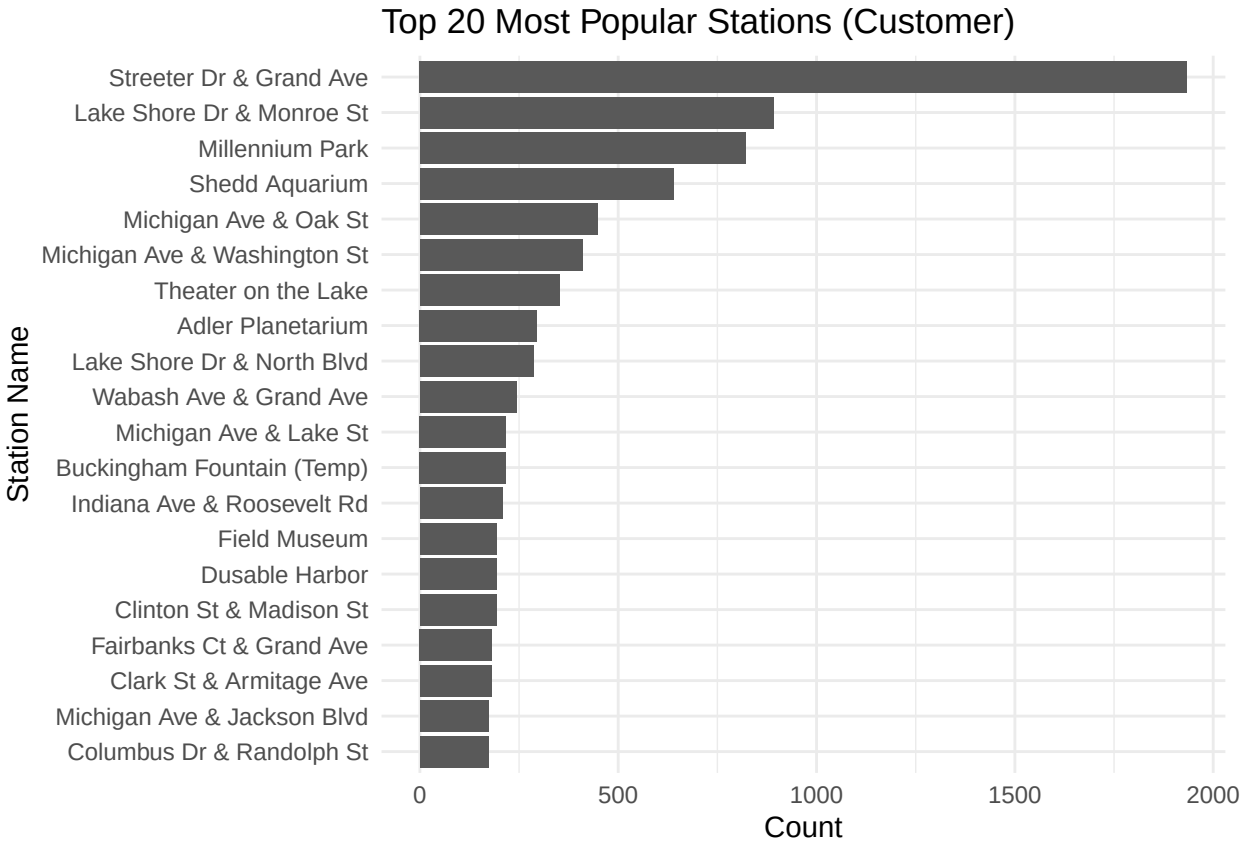
```
top20_station_popularity <- df_fil %>%
  group_by(to_station_id, to_station_name) %>%
  summarise(count = n(), .groups = "drop") %>%
  arrange(desc(count)) %>%
  slice_head(n = 20)
ggplot(data = top20_station_popularity) +
  geom_col(mapping = aes(x = reorder(to_station_name, count), y = count)) +
  coord_flip() +
  labs(title = "Top 20 Most Popular Stations",
       x = "Station Name",
       y = "Count") +
  theme_minimal()
```




```
sub_top20_station_popularity <- df_fil %>%
  filter(usertype =="Subscriber") %>%
  group_by(to_station_id, to_station_name) %>%
  summarise(count = n(), .groups = "drop") %>%
  arrange(desc(count)) %>%
  slice_head(n = 20)
ggplot(data = sub_top20_station_popularity) +
  geom_col(mapping = aes(x = reorder(to_station_name, count), y = count)) +
  coord_flip() +
  labs(title = "Top 20 Most Popular Stations (Subscriber)",
       x = "Station Name",
       y = "Count") +
  theme_minimal()
```




```
cus_top20_station_popularity <- df_fil %>%
  filter(usertype =="Customer") %>%
  group_by(to_station_id, to_station_name) %>%
  summarise(count = n(), .groups = "drop") %>%
  arrange(desc(count)) %>%
  slice_head(n = 20)
ggplot(data = cus_top20_station_popularity) +
  geom_col(mapping = aes(x = reorder(to_station_name, count), y = count)) +
  coord_flip() +
  labs(title = "Top 20 Most Popular Stations (Customer)",
       x = "Station Name",
       y = "Count") +
  theme_minimal()
```



Discussion and Insights

1. Subscribers show more frequent usage on Cyclistic bike With 93.7% of ride records coming from Subscribers, this suggests that **Subscribers either represent the majority of users or use the service much more frequently than Customers**. The significant difference in ride volume indicates distinct usage patterns between these two user types.
   However, considering the data set limitation, a single quarter might provide incomplete information about the overall user distribution patterns.

2. The ride length data reveals interesting differences between user types. **Customers have significantly longer rides**, with a mean of 62 minutes compared to subscribers' 15 minutes. The median value also differ substantially: 22 minutes for customers versus 8 minutes for subscribers.
   The larger gap between the mean and median for customers (62 vs 22 minutes) indicates the presence of some extremely long rides, suggesting varied usage pattern. This could be due to several factors: recreational riding to multiple destination, difficulty locating return station, or simply forgetting to return the bike.
   In contrast, subscribers show more consistent usage patterns, with their mean and median values being relatively close (15 vs 8 minutes), indicating more predictable, likely commute-oriented trips.

3. The box plot reveals that **75% of subscribers have ride length under 14 minutes**, with **half of all subscribers riding for less than 8 minutes**. This pattern indicates that subscribers tend to use the service for **short-distance**, **point-to-point transportation**, likely for commuting.
   The **customer data shows 75% ride for less than 30 minutes and 50% for less than 20 minutes**. Customers not only **ride longer than subscribers but also show greater variability,**

with an IQR of 17 minutes versus only 9 minutes for subscribers. **This suggests more diverse usage patterns among customers compared to the consistent short-trip behavior of subscribers.**

4. The end riding time patterns reveal distinct usage behaviors between user types. **Subscribers show clear commuting patterns with two prominent peaks: around 8 AM and 5 PM**, corresponding to typical work start and end times. The ridership gradually increases leading up to these peak hours and decreases afterward.
   In contrast, customers show a single broad peak around 4-5 PM, with ridership gradually building throughout the afternoon and evening hours. Notably, **customers show minimal morning activity compared to subscribers**.
   This strongly suggests that subscribers primarily use the bikes for work commuting, while **customers appear to use the service more for afternoon and evening recreational activities** rather than structured commute patterns

5. The daily riding frequency shows **clear behavioral differences** between user types.
   Subscribers exhibit a typical weekday commuting pattern, with consistently high usage from Tuesday to Friday (58k-65k rides) and **significantly lower weekend usage** (Sunday: 25k, Saturday: 28k). This suggests **subscribers primarily use bikes for work commuting**.
   Customers show the opposite trend, with **Saturday as the peak day** (6k rides) and Sunday also high (3.7k rides), while **weekday usage remains consistently low** (1.9k-2.7k rides). This indicates **customers are mainly recreational users** who ride during weekends for leisure activities.
   These contrasting patterns confirm that subscribers are commuters while customers are leisure riders, requiring different business strategies for each segment.

Summary

In summary, the data clearly distinguishes two distinct user segments: Subscribers are regular commuters with predictable, short-duration trips during weekdays, while Customers are recreational users with longer, more variable rides concentrated on weekends.

Recommendations

1. Ensure the subscribers have adequate bikes to use during commute.
2. Study weekend customer behavior patterns to understand their transportation needs and preferences.
3. Collect more data for different seasons and the user demographic.