

Cyclistic Case Study Mar21

Hezar K

2022-11-29

This is an analysis for Cyclistic Case Study for Google Data Analytics Course. This is an analysis for March 2021.

STEP ONE: INSTALL REQUIRED PACKAGES AND IMPORT DATA

Install the required packages. **Tidyverse** package to import and wrangling the data and **ggplot2** package for visualization of the data. **Lubridate** package for date parsing and **anytime** package for the datetime conversion.

- `install.packages("tidyverse")`
- `install.packages("ggplot2")`
- `install.packages("lubridate")`
- `install.packages("anytime")`

```
library(tidyverse)
```

```
## — Attaching packages — tidyverse 1.3.2 —
## ✓ ggplot2 3.4.0      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.4.1
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## — Conflicts — tidyverse_conflicts() —
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()     masks stats::lag()
```

```
library(lubridate)
```

```
## Loading required package: timechange
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(data.table)
```

```
##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(ggplot2)
library(anytime)
```

Import data from local drive.

```
Mar21 <- read_csv("C:/Users/theby/Documents/202103-divvy-tripdata.csv")
```

```
## Rows: 228496 Columns: 13
## — Column specification —
## Delimiter: ","
## chr (9): ride_id, rideable_type, started_at, ended_at, start_station_name, s...
## dbl (4): start_lat, start_lng, end_lat, end_lng
##
## 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.
```

STEP TWO: EXAMINE THE DATA

Examine the dataframe for an overview of the data. Review column names, `colnames()`, dimensions of the dataframe by row and column, `dim()`, the first, `head()`, and the last, `tail()`, six rows in the dataframe, the summary, `summary()`, statistics on the columns of the dataframe, and review the data type structure of columns, `str()`.

View(Mar21)

```
colnames(Mar21)
```

```
## [1] "ride_id"          "rideable_type"    "started_at"
## [4] "ended_at"         "start_station_name" "start_station_id"
## [7] "end_station_name" "end_station_id"   "start_lat"
## [10] "start_lng"        "end_lat"          "end_lng"
## [13] "member_casual"
```

```
nrow(Mar21)
```

```
## [1] 228496
```

```
dim(Mar21)
```

```
## [1] 228496      13
```

```
head(Mar21)
```

```
## # A tibble: 6 × 13
##   ride_id      ridea...1 start...2 ended...3 start...4 start...5 end_s...6 end_s...7 start...8
##   <chr>         <chr>    <chr>    <chr>    <chr>    <chr>    <chr>    <chr>    <dbl>
## 1 CFA86D4455AA1... classi... 3/16/2... 3/16/2... Humbol... 15651    Stave ... 13266      41.9
## 2 30D9DC61227D1... classi... 3/28/2... 3/28/2... Humbol... 15651    Centra... 18017      41.9
## 3 846D87A15682A... classi... 3/11/2... 3/11/2... Shield... 15443    Halste... TA1308... 41.8
## 4 994D05AA75A16... classi... 3/11/2... 3/11/2... Winthr... TA1308... Broadw... 13323      42.0
## 5 DF7464FBE92D8... classi... 3/21/2... 3/21/2... Glenwo... 525      Chicag... E008      42.0
## 6 CEBA8516FD17F... classi... 3/20/2... 3/20/2... Glenwo... 525      Chicag... E008      42.0
## # ... with 4 more variables: start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, and abbreviated variable names 1rideable_type,
## #   2started_at, 3ended_at, 4start_station_name, 5start_station_id,
## #   6end_station_name, 7end_station_id, 8start_lat
```

```
tail(Mar21)
```

```
## # A tibble: 6 × 13
##   ride_id      ridea...1 start...2 ended...3 start...4 start...5 end_s...6 end_s...7 start...8
##   <chr>         <chr>    <chr>    <chr>    <chr>    <chr>    <chr>    <chr>    <dbl>
## 1 081549DEA616C... electr... 3/14/2... 3/14/2... Larrab... TA1309... New St... TA1306... 41.9
## 2 9397BDD14798A... docked... 3/20/2... 3/20/2... Michig... 13042    New St... TA1306... 41.9
## 3 BBEB8D51AAD4... classi... 3/2/20... 3/2/20... Kingsb... KA1503... New St... TA1306... 41.9
## 4 637FF754DA0BD... classi... 3/9/20... 3/9/20... Michig... 13042    Clark ... KA1504... 41.9
## 5 F8F43A0B978A7... classi... 3/1/20... 3/1/20... Kingsb... KA1503... New St... TA1306... 41.9
## 6 3AE64EA5BF43C... electr... 3/26/2... 3/26/2... <NA>     <NA>     New St... TA1306... 41.9
## # ... with 4 more variables: start_lng <dbl>, end_lat <dbl>, end_lng <dbl>,
## #   member_casual <chr>, and abbreviated variable names 1rideable_type,
## #   2started_at, 3ended_at, 4start_station_name, 5start_station_id,
## #   6end_station_name, 7end_station_id, 8start_lat
```

```
summary(Mar21)
```

```
##      ride_id      rideable_type      started_at      ended_at
## Length:228496      Length:228496      Length:228496      Length:228496
## Class :character      Class :character      Class :character      Class :character
## Mode :character      Mode :character      Mode :character      Mode :character
##
##
##
## start_station_name start_station_id      end_station_name      end_station_id
## Length:228496      Length:228496      Length:228496      Length:228496
## Class :character      Class :character      Class :character      Class :character
## Mode :character      Mode :character      Mode :character      Mode :character
##
##
##
## start_lat      start_lng      end_lat      end_lng
## Min. :41.65      Min. : -87.78      Min. :41.64      Min. : -88.07
## 1st Qu.:41.88      1st Qu.: -87.66      1st Qu.:41.88      1st Qu.: -87.66
## Median :41.90      Median : -87.64      Median :41.90      Median : -87.64
## Mean :41.90      Mean : -87.64      Mean :41.90      Mean : -87.65
## 3rd Qu.:41.93      3rd Qu.: -87.63      3rd Qu.:41.93      3rd Qu.: -87.63
## Max. :42.07      Max. : -87.53      Max. :42.08      Max. : -87.53
##
##              NA's :167      NA's :167
## member_casual
## Length:228496
## Class :character
## Mode :character
##
##
##
##
```

```
str(Mar21)
```

```
## spc_tbl_ [228,496 × 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ride_id      : chr [1:228496] "CFA86D4455AA1030" "30D9DC61227D1AF3" "846D87A15682A284" "994D05AA75A168
F2" ...
## $ rideable_type : chr [1:228496] "classic_bike" "classic_bike" "classic_bike" "classic_bike" ...
## $ started_at   : chr [1:228496] "3/16/2021 8:32" "3/28/2021 1:26" "3/11/2021 21:17" "3/11/2021 13:26" ..
.
## $ ended_at     : chr [1:228496] "3/16/2021 8:36" "3/28/2021 1:36" "3/11/2021 21:33" "3/11/2021 13:55" ..
.
## $ start_station_name: chr [1:228496] "Humboldt Blvd & Armitage Ave" "Humboldt Blvd & Armitage Ave" "Shields A
ve & 28th Pl" "Winthrop Ave & Lawrence Ave" ...
## $ start_station_id : chr [1:228496] "15651" "15651" "15443" "TA1308000021" ...
## $ end_station_name : chr [1:228496] "Stave St & Armitage Ave" "Central Park Ave & Bloomingdale Ave" "Halsted
St & 35th St" "Broadway & Sheridan Rd" ...
## $ end_station_id   : chr [1:228496] "13266" "18017" "TA1308000043" "13323" ...
## $ start_lat        : num [1:228496] 41.9 41.9 41.8 42 42 ...
## $ start_lng        : num [1:228496] -87.7 -87.7 -87.6 -87.7 -87.7 ...
## $ end_lat          : num [1:228496] 41.9 41.9 41.8 42 42.1 ...
## $ end_lng          : num [1:228496] -87.7 -87.7 -87.6 -87.6 -87.7 ...
## $ member_casual    : chr [1:228496] "casual" "casual" "casual" "casual" ...
## - attr(*, "spec")=
## .. cols(
## ..   ride_id = col_character(),
## ..   rideable_type = col_character(),
## ..   started_at = col_character(),
## ..   ended_at = col_character(),
## ..   start_station_name = col_character(),
## ..   start_station_id = col_character(),
## ..   end_station_name = col_character(),
## ..   end_station_id = col_character(),
## ..   start_lat = col_double(),
## ..   start_lng = col_double(),
## ..   end_lat = col_double(),
## ..   end_lng = col_double(),
## ..   member_casual = col_character()
## .. )
## - attr(*, "problems")=<externalptr>
```

Columns `started_at` and `ended_at` need to be convert from character data type to date data type. **Str()** syntax confirms changes.

```
Mar21$started_at <- mdy_hm(Mar21$started_at)
Mar21$ended_at <- mdy_hm(Mar21$ended_at)
str(Mar21)
```

```
## spc_tbl_ [228,496 × 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ride_id      : chr [1:228496] "CFA86D4455AA1030" "30D9DC61227D1AF3" "846D87A15682A284" "994D05AA75A168
F2" ...
## $ rideable_type : chr [1:228496] "classic_bike" "classic_bike" "classic_bike" "classic_bike" ...
## $ started_at   : POSIXct[1:228496], format: "2021-03-16 08:32:00" "2021-03-28 01:26:00" ...
## $ ended_at     : POSIXct[1:228496], format: "2021-03-16 08:36:00" "2021-03-28 01:36:00" ...
## $ start_station_name: chr [1:228496] "Humboldt Blvd & Armitage Ave" "Humboldt Blvd & Armitage Ave" "Shields A
ve & 28th Pl" "Winthrop Ave & Lawrence Ave" ...
## $ start_station_id : chr [1:228496] "15651" "15651" "15443" "TA1308000021" ...
## $ end_station_name : chr [1:228496] "Stave St & Armitage Ave" "Central Park Ave & Bloomingdale Ave" "Halsted
St & 35th St" "Broadway & Sheridan Rd" ...
## $ end_station_id  : chr [1:228496] "13266" "18017" "TA1308000043" "13323" ...
## $ start_lat       : num [1:228496] 41.9 41.9 41.8 42 42 ...
## $ start_lng       : num [1:228496] -87.7 -87.7 -87.6 -87.7 -87.7 ...
## $ end_lat         : num [1:228496] 41.9 41.9 41.8 42 42.1 ...
## $ end_lng         : num [1:228496] -87.7 -87.7 -87.6 -87.6 -87.7 ...
## $ member_casual   : chr [1:228496] "casual" "casual" "casual" "casual" ...
## - attr(*, "spec")=
## .. cols(
## ..   ride_id = col_character(),
## ..   rideable_type = col_character(),
## ..   started_at = col_character(),
## ..   ended_at = col_character(),
## ..   start_station_name = col_character(),
## ..   start_station_id = col_character(),
## ..   end_station_name = col_character(),
## ..   end_station_id = col_character(),
## ..   start_lat = col_double(),
## ..   start_lng = col_double(),
## ..   end_lat = col_double(),
## ..   end_lng = col_double(),
## ..   member_casual = col_character()
## .. )
## - attr(*, "problems")=<externalptr>
```

Create new columns as for *date*, *month*, *day*, *year*, *day_of_week*, and *ride_length* in seconds.

```
Mar21$date <- as.Date(Mar21$started_at)
Mar21$month <- format(as.Date(Mar21$date), "%m")
Mar21$day <- format(as.Date(Mar21$date), "%d")
Mar21$year <- format(as.Date(Mar21$date), "%Y")
Mar21$day_of_week <- format(as.Date(Mar21$date), "%A")
Mar21$ride_length <- difftime(Mar21$ended_at, Mar21$started_at)
```

Convert *ride_length* column to numeric in order to run calculations on the data. First, check to see if the data type is numeric, and then convert if needed.

```
is.numeric(Mar21$ride_length)
```

```
## [1] FALSE
```

Recheck *ride_length* data type.

```
Mar21$ride_length <- as.numeric(as.character(Mar21$ride_length))
is.numeric(Mar21$ride_length)
```

```
## [1] TRUE
```

STEP THREE: CLEAN DATA

na.omit() will remove all NA from the dataframe.

```
Mar21 <- na.omit(Mar21)
```

Remove rows with the *ride_id* column character length is not 16. This will remove all the scientific ride ids that we noticed while examining the data.

```
Mar21 <- subset(Mar21, nchar(as.character(ride_id)) == 16)
```

Remove rows with the *ride_length* less than 1 minute.

```
Mar21 <- subset (Mar21, ride_length > "1")
```

STEP FOUR: ANALYZE DATA

Analyze the dataframe by find the **mean**, **median**, **max** (maximum), and **min** (minimum) of *ride_length*.

```
mean(Mar21$ride_length)
```

```
## [1] 1376.565
```

```
median(Mar21$ride_length)
```

```
## [1] 780
```

```
max(Mar21$ride_length)
```

```
## [1] 1900920
```

```
min(Mar21$ride_length)
```

```
## [1] 60
```

Run a statistical summary of the *ride_length*.

```
summary(Mar21$ride_length)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       60     420     780    1377    1380 1900920
```

Compare the members and casual users

```
aggregate(Mar21$ride_length ~ Mar21$member_casual, FUN = mean)
```

```
##      Mar21$member_casual Mar21$ride_length
## 1                casual      2318.7843
## 2                member      826.4151
```

```
aggregate(Mar21$ride_length ~ Mar21$member_casual, FUN = median)
```

```
##      Mar21$member_casual Mar21$ride_length
## 1                casual      1200
## 2                member       600
```

```
aggregate(Mar21$ride_length ~ Mar21$member_casual, FUN = max)
```

```
##      Mar21$member_casual Mar21$ride_length
## 1                casual    1900920
## 2                member     88020
```

```
aggregate(Mar21$ride_length ~ Mar21$member_casual, FUN = min)
```

```
##      Mar21$member_casual Mar21$ride_length
## 1                casual       60
## 2                member       60
```

Aggregate the average ride length by each day of the week for members and users.

```
aggregate(Mar21$ride_length ~ Mar21$member_casual + Mar21$day_of_week, FUN = mean)
```

```
##      Mar21$member_casual Mar21$day_of_week Mar21$ride_length
## 1          casual      Friday      1781.4072
## 2          member      Friday       752.8076
## 3          casual      Monday      2725.6271
## 4          member      Monday       831.0461
## 5          casual      Saturday     2538.5274
## 6          member      Saturday      938.6799
## 7          casual      Sunday      2475.2933
## 8          member      Sunday       961.3667
## 9          casual      Thursday     1814.9028
## 10         member      Thursday      713.5517
## 11         casual      Tuesday     2214.4371
## 12         member      Tuesday       801.9463
## 13         casual      Wednesday    1762.9929
## 14         member      Wednesday     760.9907
```

Sort the days of the week in order.

```
Mar21$day_of_week <- ordered(Mar21$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday",
"Friday", "Saturday"))
```

Assign the aggregate the average ride length by each day of the week for members and users to x.

```
x <- aggregate(Mar21$ride_length ~ Mar21$member_casual + Mar21$day_of_week, FUN = mean)

head(x)
```

```
##      Mar21$member_casual Mar21$day_of_week Mar21$ride_length
## 1          casual      Sunday      2475.2933
## 2          member      Sunday       961.3667
## 3          casual      Monday      2725.6271
## 4          member      Monday       831.0461
## 5          casual      Tuesday     2214.4371
## 6          member      Tuesday       801.9463
```

Find the average ride length of member riders and casual riders per day and assign it to y.

```
y <- Mar21 %>%
  mutate(weekday = wday(started_at)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n(),
            average_duration = mean(ride_length), .groups = 'drop') %>%
  arrange(member_casual, weekday)

head(y)
```

```
## # A tibble: 6 × 4
##   member_casual weekday number_of_rides average_duration
##   <chr>          <int>          <int>          <dbl>
## 1 casual            1          15854          2475.
## 2 casual            2          10716          2726.
## 3 casual            3           9265          2214.
## 4 casual            4           7638          1763.
## 5 casual            5           4783          1815.
## 6 casual            6           6822          1781.
```

Analyze the dataframe to find the frequency of member riders, casual riders, classic bikes, docked bikes, and electric bikes.

```
table(Mar21$member_casual)
```

```
##
## casual member
## 75315 128989
```

```
table(Mar21$rideable_type)
```

```
##
## classic_bike  docked_bike electric_bike
##      151080      15609      37615
```

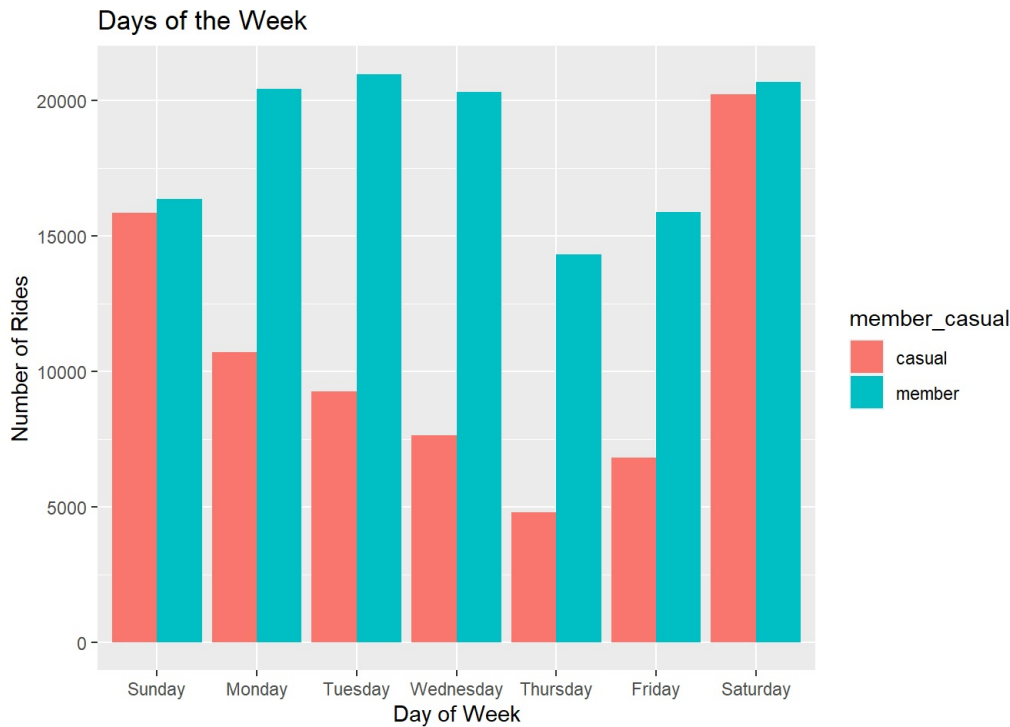
STEP FIVE: VISUALIZATION

Display full digits instead of scientific number.

```
options(scipen=999)
```

Plot the number of rides by user type during the week.

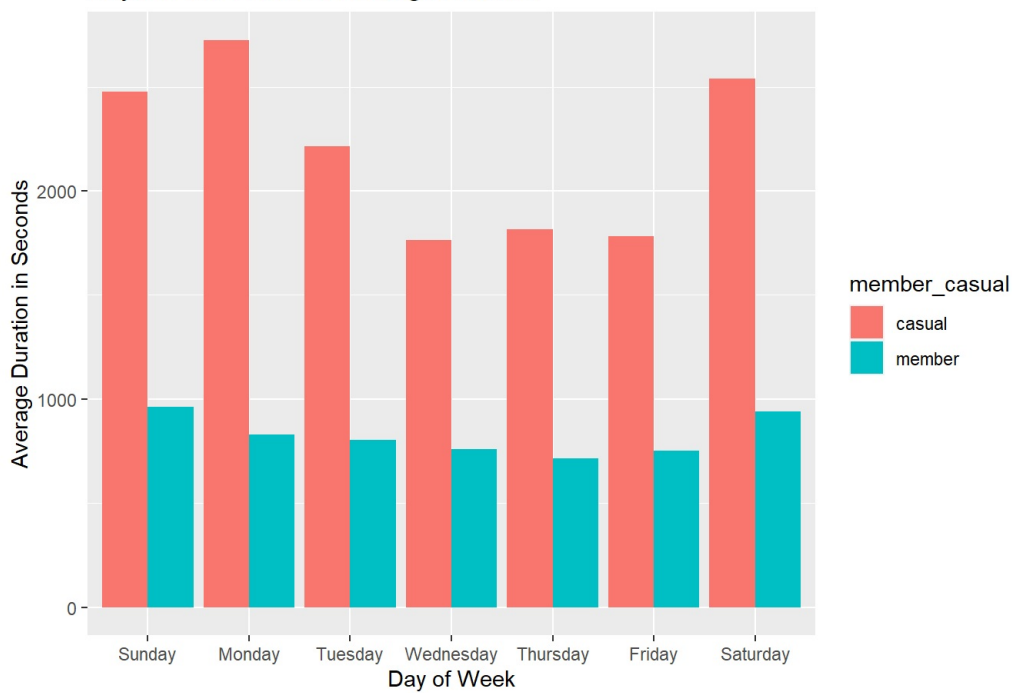
```
Mar21 %>%
  mutate(day_of_week) %>%
  group_by(member_casual, day_of_week) %>%
  summarise(number_of_rides = n(), average_duration = mean(ride_length), .groups = 'drop') %>%
  arrange(member_casual, day_of_week) %>%
  ggplot(aes(x = day_of_week, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge")+
  labs(x = "Day of Week",
       y = "Number of Rides",
       title= "Days of the Week")
```



Plot the duration of the ride by user type during the week.

```
Mar21 %>%
  mutate(day_of_week) %>%
  group_by(member_casual, day_of_week) %>%
  summarise(number_of_rides = n(), average_duration = mean(ride_length), .groups = 'drop') %>%
  arrange(member_casual, day_of_week) %>%
  ggplot(aes(x = day_of_week, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge") +
  labs(x = "Day of Week",
       y = "Average Duration in Seconds",
       title= "Days of the Week vs Average Duration")
```

Days of the Week vs Average Duration



Create new dataframe for plots for weekday trends vs weekend trends.

```
mc<- as.data.frame(table(Mar21$day_of_week,Mar21$member_casual))
```

Rename columns

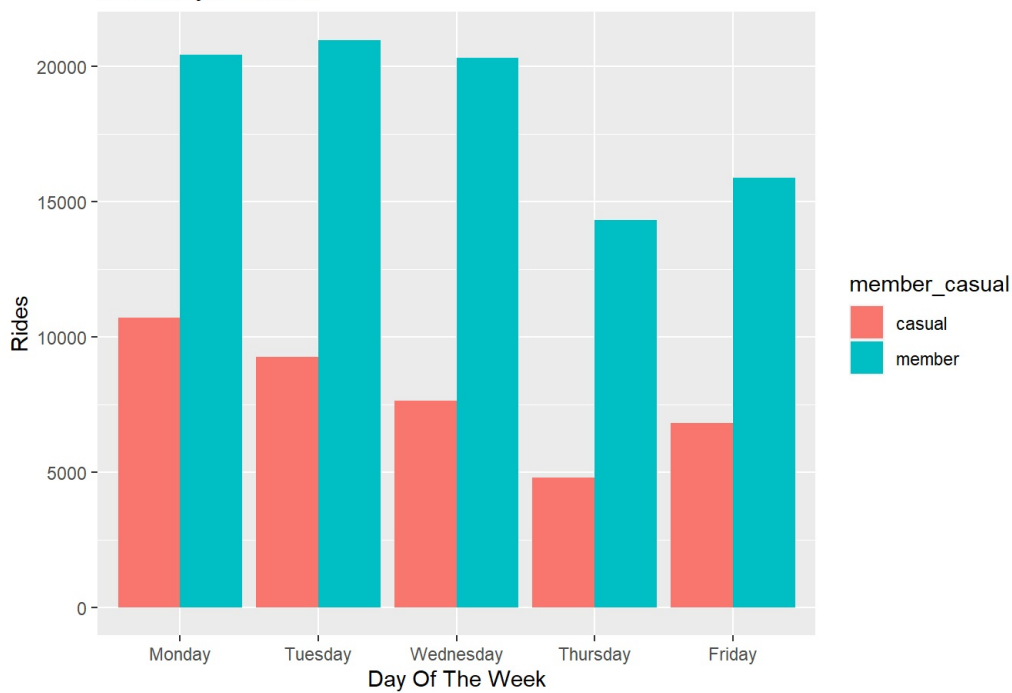
```
mc<-rename(mc, day_of_week = Var1, member_casual = Var2)
head(mc)
```

```
##  day_of_week member_casual  Freq
## 1    Sunday          casual 15854
## 2    Monday          casual 10716
## 3   Tuesday          casual  9265
## 4 Wednesday          casual  7638
## 5   Thursday          casual  4783
## 6    Friday          casual  6822
```

Weekday trends (Monday through Friday).

```
mc %>%
  filter(day_of_week == "Monday" |
         day_of_week == "Tuesday" |
         day_of_week == "Wednesday" |
         day_of_week == "Thursday" |
         day_of_week == "Friday") %>%
  ggplot(aes(x = day_of_week, y = Freq, fill = member_casual))+
  geom_bar(stat = "identity" , position = "dodge") +
  labs(title = "Weekdays Trends",
       x= "Day Of The Week",
       y = "Rides")
```

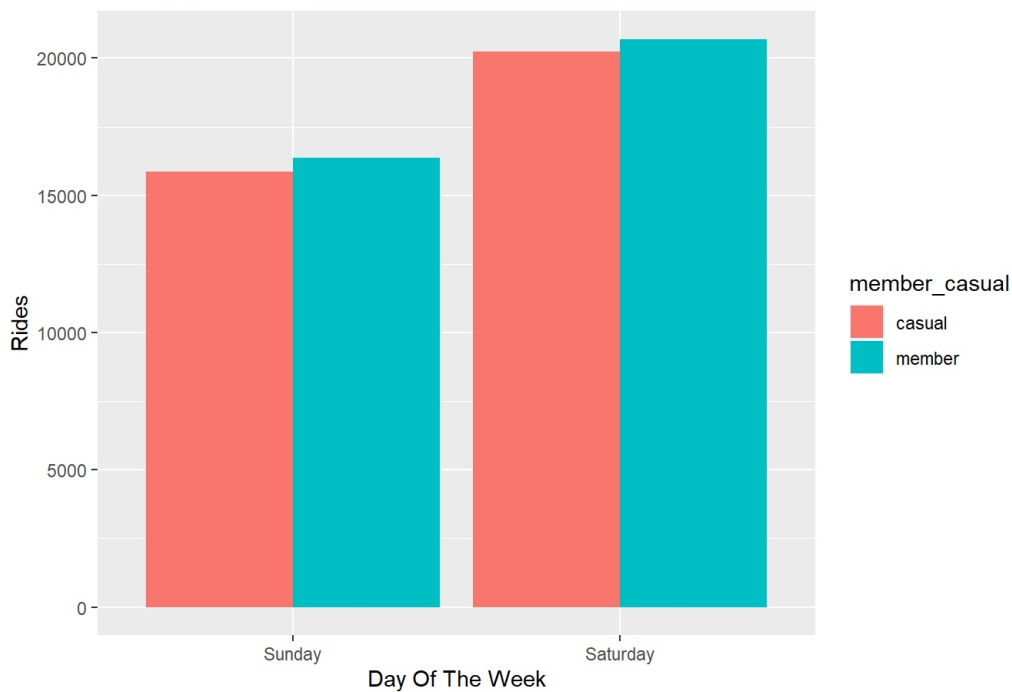

Weekdays Trends



Weekend trends (Sunday and Saturday).

```
mc %>%
  filter(day_of_week == "Sunday" |
         day_of_week == "Saturday") %>%
  ggplot(aes(x = day_of_week, y = Freq, fill = member_casual))+
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Weekends Trends",
       x = "Day Of The Week",
       y = "Rides")
```

Weekends Trends



Create dataframe for member and casual riders vs ride type

```
rt<- as.data.frame(table(Mar21$rideable_type,Mar21$member_casual))
```

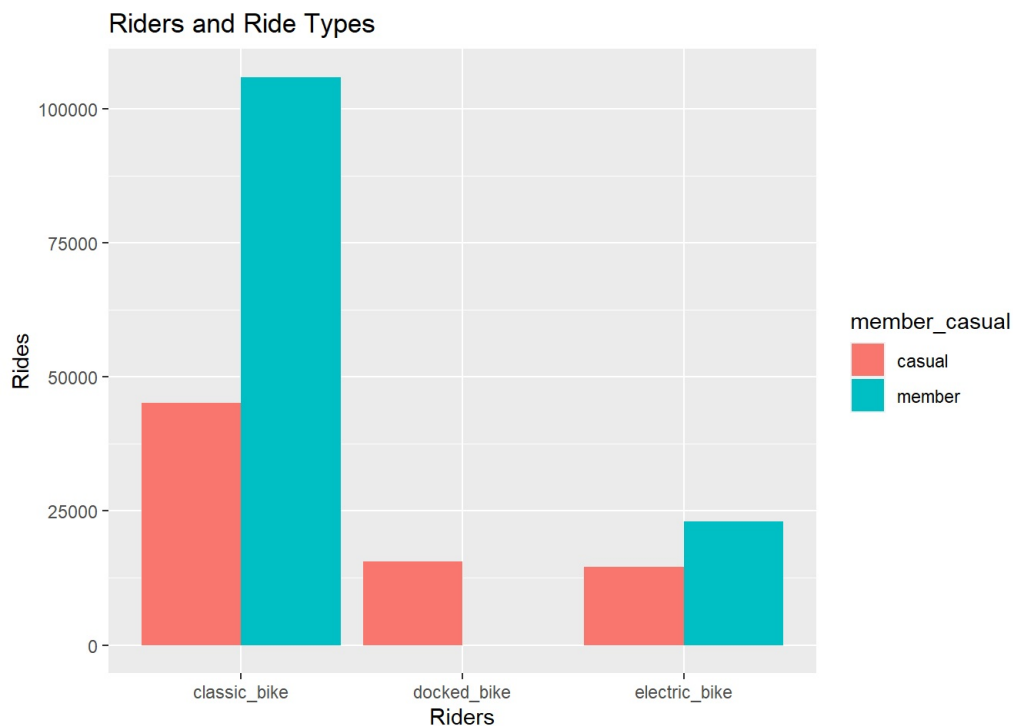
Rename columns.

```
rt<-rename(rt, rideable_type = Var1, member_casual = Var2)
head(rt)
```

```
##   rideable_type member_casual   Freq
## 1 classic_bike      casual  45159
## 2 docked_bike       casual  15609
## 3 electric_bike     casual  14547
## 4 classic_bike     member 105921
## 5 docked_bike      member    0
## 6 electric_bike    member  23068
```

Plot for bike user vs bike type.

```
rt %>%
  filter(member_casual == "member" |
         member_casual == "casual") %>%
  ggplot(aes(x = rideable_type, y = Freq, fill = member_casual))+
  geom_bar(stat = "identity", position = "dodge") +
  labs(title = "Riders and Ride Types",
       x = "Riders",
       y = "Rides")
```



STEP SIX: EXPORT ANALYZED DATA

Save the analyzed data as a new file. `fwrite(Mar21, "Mar21.csv")`