Cyclistic Citibike Analysis

Ashutosh Rajput

10/01/2022

Cyclistic Trip data Analysis 2019-2020

We will install the required and necessary packages for our research and Analysis.

Libraries

Loading the required libraries

```
library(tidyverse)  #helps wrangle data
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(lubridate)  #helps wrangle date attributes
```

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

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

```
library(ggplot2)  #helps visualize data
library(readr)
```

Working Directory

Setting the working directory to import the data from csv files into R

```
setwd("C:/Users/ashut/Desktop/My_projects/Google_data_analytics/Case_study-Citing Cyclistic Bike to Inc

getwd()
```

```
## [1] "C:/Users/ashut/Desktop/My_projects/Google_data_analytics/Case_study-Citing Cyclistic Bike to In
```

Importing data

Importing the datasets.

```
# Uploading Divvy datasets (csv files) into DF.
q2_2019 <- read_csv("Divvy_Trips_2019_Q2.csv")
```

```
## Rows: 1108163 Columns: 12

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): 03 - Rental Start Station Name, 02 - Rental End Station Name, User...
## dbl  (5): 01 - Rental Details Rental ID, 01 - Rental Details Bike ID, 03 - R...
## dttm (2): 01 - Rental Details Local Start Time, 01 - Rental Details Local En...

##
## 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.
```

```
q3_2019 <- read_csv("Divvy_Trips_2019_Q3.csv")
```

```
## Rows: 1640718 Columns: 12

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): from_station_name, to_station_name, usertype, gender
## dbl  (5): trip_id, bikeid, from_station_id, to_station_id, birthyear
## dttm (2): start_time, end_time

##
## 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.
```

```
q4_2019 <- read_csv("Divvy_Trips_2019_Q4.csv")
```

```
## Rows: 704054 Columns: 12

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): from_station_name, to_station_name, usertype, gender
## dbl  (5): trip_id, bikeid, from_station_id, to_station_id, birthyear
## dttm (2): start_time, end_time
```

```
##
## 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.
```

```
q1_2020 <- read_csv("Divvy_Trips_2020_Q1.csv")
```

```
## Rows: 426887 Columns: 13

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (5): ride_id, rideable_type, start_station_name, end_station_name, memb...
## dbl  (6): start_station_id, end_station_id, start_lat, start_lng, end_lat, e...
## dttm (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.
```

Wrangling data

Comparing the column names of the files.

In order to combine the files into 1 single file we need same column names with respective data. Hence observing the data.

```
colnames(q3_2019)
```

```
##  [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"          "gender"            "birthyear"
```

```
colnames(q4_2019)
```

```
##  [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"          "gender"            "birthyear"
```

```
colnames(q2_2019)
```

```
##  [1] "01 - Rental Details Rental ID"
##  [2] "01 - Rental Details Local Start Time"
##  [3] "01 - Rental Details Local End Time"
##  [4] "01 - Rental Details Bike ID"
##  [5] "01 - Rental Details Duration In Seconds Uncapped"
##  [6] "03 - Rental Start Station ID"
##  [7] "03 - Rental Start Station Name"
##  [8] "02 - Rental End Station ID"
##  [9] "02 - Rental End Station Name"
## [10] "User Type"
## [11] "Member Gender"
## [12] "05 - Member Details Member Birthday Year"
```

```
colnames(q1_2020)
```

```
##  [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"
```

Observation

We can see that the column names of q3, q2, q4-2019 does not match the most recent and fresh q1_2020.

Note - The column order does not matter. The column name must match what data you want to bind together into one.

- In order to match the column names and data in the file -
  - Rename the columns
  - Combine the data in all files into one.

```
# Renaming the column names as to match file q1_2020

(q4_2019 <- rename(q4_2019
                   ,ride_id = trip_id
                   ,rideable_type = bikeid
                   ,started_at = start_time
                   ,ended_at = end_time
                   ,start_station_name = from_station_name
                   ,start_station_id = from_station_id
                   ,end_station_name = to_station_name
                   ,end_station_id = to_station_id
                   ,member_casual = usertype))
```

```
## # A tibble: 704,054 x 12
##     ride_id started_at          ended_at            rideable_type tripduration
##       <dbl> <dttm>              <dttm>                      <dbl>        <dbl>
##  1 25223640 2019-10-01 00:01:39 2019-10-01 00:17:20          2215          940
##  2 25223641 2019-10-01 00:02:16 2019-10-01 00:06:34          6328          258
##  3 25223642 2019-10-01 00:04:32 2019-10-01 00:18:43          3003          850
##  4 25223643 2019-10-01 00:04:32 2019-10-01 00:43:43          3275         2350
##  5 25223644 2019-10-01 00:04:34 2019-10-01 00:35:42          5294         1867
##  6 25223645 2019-10-01 00:04:38 2019-10-01 00:10:51          1891          373
##  7 25223646 2019-10-01 00:04:52 2019-10-01 00:22:45          1061         1072
##  8 25223647 2019-10-01 00:04:57 2019-10-01 00:29:16          1274         1458
##  9 25223648 2019-10-01 00:05:20 2019-10-01 00:29:18          6011         1437
## 10 25223649 2019-10-01 00:05:20 2019-10-01 02:23:46          2957         8306
## # ... with 704,044 more rows, and 7 more variables: start_station_id <dbl>,
## #   start_station_name <chr>, end_station_id <dbl>, end_station_name <chr>,
## #   member_casual <chr>, gender <chr>, birthyear <dbl>
```

```r
(q3_2019 <- rename(q3_2019
                  ,ride_id = trip_id
                  ,rideable_type = bikeid
                  ,started_at = start_time
                  ,ended_at = end_time
                  ,start_station_name = from_station_name
                  ,start_station_id = from_station_id
                  ,end_station_name = to_station_name
                  ,end_station_id = to_station_id
                  ,member_casual = usertype))
```

```
## # A tibble: 1,640,718 x 12
##     ride_id started_at          ended_at            rideable_type tripduration
##       <dbl> <dttm>              <dttm>                      <dbl>        <dbl>
##  1 23479388 2019-07-01 00:00:27 2019-07-01 00:20:41          3591         1214
##  2 23479389 2019-07-01 00:01:16 2019-07-01 00:18:44          5353         1048
##  3 23479390 2019-07-01 00:01:48 2019-07-01 00:27:42          6180         1554
##  4 23479391 2019-07-01 00:02:07 2019-07-01 00:27:10          5540         1503
##  5 23479392 2019-07-01 00:02:13 2019-07-01 00:22:26          6014         1213
##  6 23479393 2019-07-01 00:02:21 2019-07-01 00:07:31          4941          310
##  7 23479394 2019-07-01 00:02:24 2019-07-01 00:23:12          3770         1248
##  8 23479395 2019-07-01 00:02:26 2019-07-01 00:28:16          5442         1550
##  9 23479396 2019-07-01 00:02:34 2019-07-01 00:28:57          2957         1583
## 10 23479397 2019-07-01 00:02:45 2019-07-01 00:29:14          6091         1589
## # ... with 1,640,708 more rows, and 7 more variables: start_station_id <dbl>,
## #   start_station_name <chr>, end_station_id <dbl>, end_station_name <chr>,
## #   member_casual <chr>, gender <chr>, birthyear <dbl>
```

```r
(q2_2019 <- rename(q2_2019
                  ,ride_id = "01 - Rental Details Rental ID"
                  ,rideable_type = "01 - Rental Details Bike ID"
                  ,started_at = "01 - Rental Details Local Start Time"
                  ,ended_at = "01 - Rental Details Local End Time"
                  ,start_station_name = "03 - Rental Start Station Name"
                  ,start_station_id = "03 - Rental Start Station ID"
                  ,end_station_name = "02 - Rental End Station Name"
                  ,end_station_id = "02 - Rental End Station ID"
                  ,member_casual = "User Type"))
```

```
## # A tibble: 1,108,163 x 12
##     ride_id started_at          ended_at            rideable_type
##       <dbl> <dttm>              <dttm>                      <dbl>
##  1 22178529 2019-04-01 00:02:22 2019-04-01 00:09:48          6251
##  2 22178530 2019-04-01 00:03:02 2019-04-01 00:20:30          6226
##  3 22178531 2019-04-01 00:11:07 2019-04-01 00:15:19          5649
##  4 22178532 2019-04-01 00:13:01 2019-04-01 00:18:58          4151
##  5 22178533 2019-04-01 00:19:26 2019-04-01 00:36:13          3270
##  6 22178534 2019-04-01 00:19:39 2019-04-01 00:23:56          3123
##  7 22178535 2019-04-01 00:26:33 2019-04-01 00:35:41          6418
##  8 22178536 2019-04-01 00:29:48 2019-04-01 00:36:11          4513
##  9 22178537 2019-04-01 00:32:07 2019-04-01 01:07:44          3280
## 10 22178538 2019-04-01 00:32:19 2019-04-01 01:07:39          5534
```

```
## # ... with 1,108,153 more rows, and 8 more variables:
## #   01 - Rental Details Duration In Seconds Uncapped <dbl>,
## #   start_station_id <dbl>, start_station_name <chr>, end_station_id <dbl>,
## #   end_station_name <chr>, member_casual <chr>, Member Gender <chr>,
## #   05 - Member Details Member Birthday Year <dbl>
```

Looking at the data sets with the **changed** column names.

```
# look for incongruencies, if any

glimpse(q1_2020)
```

```
## Rows: 426,887
## Columns: 13
## $ ride_id            <chr> "EACB19130B0CDA4A", "8FED874C809DC021", "789F3C21E4~
## $ rideable_type      <chr> "docked_bike", "docked_bike", "docked_bike", "docke~
## $ started_at         <dttm> 2020-01-21 20:06:59, 2020-01-30 14:22:39, 2020-01-~
## $ ended_at           <dttm> 2020-01-21 20:14:30, 2020-01-30 14:26:22, 2020-01-~
## $ start_station_name <chr> "Western Ave & Leland Ave", "Clark St & Montrose Av~
## $ start_station_id   <dbl> 239, 234, 296, 51, 66, 212, 96, 96, 212, 38, 117, 1~
## $ end_station_name   <chr> "Clark St & Leland Ave", "Southport Ave & Irving Pa~
## $ end_station_id     <dbl> 326, 318, 117, 24, 212, 96, 212, 212, 96, 100, 632,~
## $ start_lat          <dbl> 41.9665, 41.9616, 41.9401, 41.8846, 41.8856, 41.889~
## $ start_lng          <dbl> -87.6884, -87.6660, -87.6455, -87.6319, -87.6418, -~
## $ end_lat            <dbl> 41.9671, 41.9542, 41.9402, 41.8918, 41.8899, 41.884~
## $ end_lng            <dbl> -87.6674, -87.6644, -87.6530, -87.6206, -87.6343, -~
## $ member_casual      <chr> "member", "member", "member", "member", "member", "~
```

```
glimpse(q4_2019)
```

```
## Rows: 704,054
## Columns: 12
## $ ride_id            <dbl> 25223640, 25223641, 25223642, 25223643, 25223644, 2~
## $ started_at         <dttm> 2019-10-01 00:01:39, 2019-10-01 00:02:16, 2019-10-~
## $ ended_at           <dttm> 2019-10-01 00:17:20, 2019-10-01 00:06:34, 2019-10-~
## $ rideable_type      <dbl> 2215, 6328, 3003, 3275, 5294, 1891, 1061, 1274, 601~
## $ tripduration       <dbl> 940, 258, 850, 2350, 1867, 373, 1072, 1458, 1437, 8~
## $ start_station_id   <dbl> 20, 19, 84, 313, 210, 156, 84, 156, 156, 336, 77, 1~
## $ start_station_name <chr> "Sheffield Ave & Kingsbury St", "Throop (Loomis) St~
## $ end_station_id     <dbl> 309, 241, 199, 290, 382, 226, 142, 463, 463, 336, 5~
## $ end_station_name   <chr> "Leavitt St & Armitage Ave", "Morgan St & Polk St",~
## $ member_casual      <chr> "Subscriber", "Subscriber", "Subscriber", "Subscrib~
## $ gender             <chr> "Male", "Male", "Female", "Male", "Male", "Female",~
## $ birthyear          <dbl> 1987, 1998, 1991, 1990, 1987, 1994, 1991, 1995, 199~
```

```
glimpse(q3_2019)
```

```
## Rows: 1,640,718
## Columns: 12
## $ ride_id            <dbl> 23479388, 23479389, 23479390, 23479391, 23479392, 2~
## $ started_at         <dttm> 2019-07-01 00:00:27, 2019-07-01 00:01:16, 2019-07-~
## $ ended_at           <dttm> 2019-07-01 00:20:41, 2019-07-01 00:18:44, 2019-07-~
```

```
## $ rideable_type      <dbl> 3591, 5353, 6180, 5540, 6014, 4941, 3770, 5442, 295~
## $ tripduration       <dbl> 1214, 1048, 1554, 1503, 1213, 310, 1248, 1550, 1583~
## $ start_station_id   <dbl> 117, 381, 313, 313, 168, 300, 168, 313, 43, 43, 511~
## $ start_station_name <chr> "Wilton Ave & Belmont Ave", "Western Ave & Monroe S~
## $ end_station_id     <dbl> 497, 203, 144, 144, 62, 232, 62, 144, 195, 195, 84,~
## $ end_station_name   <chr> "Kimball Ave & Belmont Ave", "Western Ave & 21st St~
## $ member_casual      <chr> "Subscriber", "Customer", "Customer", "Customer", "~
## $ gender             <chr> "Male", NA, NA, NA, NA, "Male", NA, NA, NA, NA, NA,~
## $ birthyear          <dbl> 1992, NA, NA, NA, NA, 1990, NA, NA, NA, NA, NA, NA,~
```

```
glimpse(q2_2019)
```

```
## Rows: 1,108,163
## Columns: 12
## $ ride_id                                           <dbl> 22178529, 22178530,~
## $ started_at                                        <dttm> 2019-04-01 00:02:2~
## $ ended_at                                          <dttm> 2019-04-01 00:09:4~
## $ rideable_type                                     <dbl> 6251, 6226, 5649, 4~
## $ `01 - Rental Details Duration In Seconds Uncapped` <dbl> 446, 1048, 252, 357~
## $ start_station_id                                  <dbl> 81, 317, 283, 26, 2~
## $ start_station_name                                <chr> "Daley Center Plaza~
## $ end_station_id                                    <dbl> 56, 59, 174, 133, 1~
## $ end_station_name                                  <chr> "Desplaines St & Ki~
## $ member_casual                                     <chr> "Subscriber", "Subs~
## $ `Member Gender`                                   <chr> "Male", "Female", "~
## $ `05 - Member Details Member Birthday Year`        <dbl> 1975, 1984, 1990, 1~
```

Now, the column names matches.

But, the assignment of data type to ride_id and rideable_type is **dbl**, which doesnot match the data type of file q1_2020.

- Changing data type from dbl to character.
  - ride_id
  - rideable_type

```
q4_2019 <-  mutate(q4_2019, ride_id = as.character(ride_id)
                   ,rideable_type = as.character(rideable_type))
q3_2019 <-  mutate(q3_2019, ride_id = as.character(ride_id)
                   ,rideable_type = as.character(rideable_type))
q2_2019 <-  mutate(q2_2019, ride_id = as.character(ride_id)
                   ,rideable_type = as.character(rideable_type))
```

Combing the data in Rows into 1 single file

```
# Adding up quarter's data frames into one big data frame
all_trips <- bind_rows(q2_2019, q3_2019, q4_2019, q1_2020)

# Remove lat, long, birthyear, and gender fields as this data is not in 2020 datasets.
all_trips <- all_trips %>%
  select(-c(start_lat, start_lng, end_lat, end_lng, birthyear, gender, "01 - Rental Details Duration In
```

Preparing Data for Analysis.

Observing our file.

```
#List of column names
colnames(all_trips)
```

```
## [1] "ride_id"            "started_at"         "ended_at"
## [4] "rideable_type"      "start_station_id"   "start_station_name"
## [7] "end_station_id"     "end_station_name"   "member_casual"
```

```
#Rows in our data frame.
nrow(all_trips)
```

```
## [1] 3879822
```

```
head(all_trips)
```

```
## # A tibble: 6 x 9
##   ride_id started_at          ended_at            rideable_type start_station_id
##   <chr>   <dttm>              <dttm>              <chr>                     <dbl>
## 1 221785~ 2019-04-01 00:02:22 2019-04-01 00:09:48 6251                         81
## 2 221785~ 2019-04-01 00:03:02 2019-04-01 00:20:30 6226                        317
## 3 221785~ 2019-04-01 00:11:07 2019-04-01 00:15:19 5649                        283
## 4 221785~ 2019-04-01 00:13:01 2019-04-01 00:18:58 4151                         26
## 5 221785~ 2019-04-01 00:19:26 2019-04-01 00:36:13 3270                        202
## 6 221785~ 2019-04-01 00:19:39 2019-04-01 00:23:56 3123                        420
## # ... with 4 more variables: start_station_name <chr>, end_station_id <dbl>,
## #   end_station_name <chr>, member_casual <chr>
```

```
tail(all_trips)
```

```
## # A tibble: 6 x 9
##   ride_id started_at          ended_at            rideable_type start_station_id
##   <chr>   <dttm>              <dttm>              <chr>                     <dbl>
## 1 6F4D22~ 2020-03-10 10:40:27 2020-03-10 10:40:29 docked_bike                675
## 2 ADDAA3~ 2020-03-10 10:40:06 2020-03-10 10:40:07 docked_bike                675
## 3 82B10F~ 2020-03-07 15:25:55 2020-03-07 16:14:03 docked_bike                161
## 4 AA0D5A~ 2020-03-01 13:12:38 2020-03-01 13:38:29 docked_bike                141
## 5 329636~ 2020-03-07 18:02:45 2020-03-07 18:13:18 docked_bike                672
## 6 064EC7~ 2020-03-08 13:03:57 2020-03-08 13:32:27 docked_bike                110
## # ... with 4 more variables: start_station_name <chr>, end_station_id <dbl>,
## #   end_station_name <chr>, member_casual <chr>
```

```
# Basic layout of columns there types and data.
str(all_trips)
```

```
## tibble [3,879,822 x 9] (S3: tbl_df/tbl/data.frame)
##  $ ride_id           : chr [1:3879822] "22178529" "22178530" "22178531" "22178532" ...
```

```
##  $ started_at        : POSIXct[1:3879822], format: "2019-04-01 00:02:22" "2019-04-01 00:03:02" ...
##  $ ended_at          : POSIXct[1:3879822], format: "2019-04-01 00:09:48" "2019-04-01 00:20:30" ...
##  $ rideable_type     : chr [1:3879822] "6251" "6226" "5649" "4151" ...
##  $ start_station_id  : num [1:3879822] 81 317 283 26 202 420 503 260 211 211 ...
##  $ start_station_name: chr [1:3879822] "Daley Center Plaza" "Wood St & Taylor St" "LaSalle St & Jacks
##  $ end_station_id    : num [1:3879822] 56 59 174 133 129 426 500 499 211 211 ...
##  $ end_station_name  : chr [1:3879822] "Desplaines St & Kinzie St" "Wabash Ave & Roosevelt Rd" "Canal
##  $ member_casual     : chr [1:3879822] "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
```

```
# With this we will get majority of the basic statistical data we need to know about our columns in our
summary(all_trips)
```

```
##    ride_id            started_at                     ended_at
##  Length:3879822     Min.   :2019-04-01 00:02:22   Min.   :2019-04-01 00:09:48
##  Class :character   1st Qu.:2019-06-23 07:49:09   1st Qu.:2019-06-23 08:20:27
##  Mode  :character   Median :2019-08-14 17:43:38   Median :2019-08-14 18:02:04
##                     Mean   :2019-08-26 00:49:59   Mean   :2019-08-26 01:14:37
##                     3rd Qu.:2019-10-12 12:10:21   3rd Qu.:2019-10-12 12:36:16
##                     Max.   :2020-03-31 23:51:34   Max.   :2020-05-19 20:10:34
##
##  rideable_type      start_station_id start_station_name end_station_id
##  Length:3879822     Min.   :  1.0    Length:3879822     Min.   :  1.0
##  Class :character   1st Qu.: 77.0    Class :character   1st Qu.: 77.0
##  Mode  :character   Median :174.0    Mode  :character   Median :174.0
##                     Mean   :202.9                       Mean   :203.8
##                     3rd Qu.:291.0                       3rd Qu.:291.0
##                     Max.   :675.0                       Max.   :675.0
##                                                         NA's   :1
##  end_station_name   member_casual
##  Length:3879822     Length:3879822
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
##
```

Correcting Errors.

- Data gaps Found -
  - The terms in member_casual used before 2020 data is different.
  - ride length and days on which ride took place.

```
table(all_trips$member_casual)
```

```
##
##     casual   Customer     member Subscriber
##      48480     857474     378407    2595461
```

Here we can observe the column should had only 2 values but there are 4 terms.

Hence, combining Subscriber into member and Customer into casual.

```
all_trips <- all_trips %>%
  mutate(member_casual = recode(member_casual
                          ,"Subscriber" = "member"
                          ,"Customer" = "casual"))

table(all_trips$member_casual)
```

```
##
##  casual  member
##  905954 2973868
```

Adding Information

Date Column

Adding columns that will list the date, month, day, and year of each ride

This will allow us to aggregate ride data for each month, day, or year . . . before completing these operations we could only aggregate at the ride level

```
#Getting into default format i.e. yyyy-mm-dd
all_trips$date <- as.Date(all_trips$started_at)


# Getting month out of date
all_trips$month <- format(as.Date(all_trips$date), "%m")


# Getting day out of the date
all_trips$day <- format(as.Date(all_trips$date), "%d")


# Getting Year out of date
all_trips$year <- format(as.Date(all_trips$date), "%Y")


#Getting which day was it on that date
all_trips$day_of_week <- format(as.Date(all_trips$date), "%A")
```

Ride Length

Adding a "ride_length" column with calculation to all_trips (in seconds)

```
all_trips$ride_length <- difftime(all_trips$ended_at,all_trips$started_at)

glimpse(all_trips)
```

```
## Rows: 3,879,822
## Columns: 15
```

```
## $ ride_id            <chr> "22178529", "22178530", "22178531", "22178532", "22~
## $ started_at         <dttm> 2019-04-01 00:02:22, 2019-04-01 00:03:02, 2019-04-~
## $ ended_at           <dttm> 2019-04-01 00:09:48, 2019-04-01 00:20:30, 2019-04-~
## $ rideable_type      <chr> "6251", "6226", "5649", "4151", "3270", "3123", "64~
## $ start_station_id   <dbl> 81, 317, 283, 26, 202, 420, 503, 260, 211, 211, 304~
## $ start_station_name <chr> "Daley Center Plaza", "Wood St & Taylor St", "LaSal~
## $ end_station_id     <dbl> 56, 59, 174, 133, 129, 426, 500, 499, 211, 211, 232~
## $ end_station_name   <chr> "Desplaines St & Kinzie St", "Wabash Ave & Roosevel~
## $ member_casual      <chr> "member", "member", "member", "member", "member", "~
## $ date               <date> 2019-04-01, 2019-04-01, 2019-04-01, 2019-04-01, 20~
## $ month              <chr> "04", "04", "04", "04", "04", "04", "04", "04", "04~
## $ day                <chr> "01", "01", "01", "01", "01", "01", "01", "01", "01~
## $ year               <chr> "2019", "2019", "2019", "2019", "2019", "2019", "20~
## $ day_of_week        <chr> "Monday", "Monday", "Monday", "Monday", "Monday", "~
## $ ride_length        <drtn> 446 secs, 1048 secs, 252 secs, 357 secs, 1007 secs~
```

```
# str(all_trips)
```

As we can observe the data is in the time - double format and seconds added. It is good to visualize and understand but not useful for calculations.

Hence, - Converting "ride_length" from double to numeric so that we can run calculations on the data.

```
is.double(all_trips$ride_length)
```

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

```
all_trips$ride_length <- as.numeric(as.character(all_trips$ride_length))

is.numeric(all_trips$ride_length)
```

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

New Clear And Clean Dataset

As the dataset contained a few hundred entries when bikes were taken out of docks and checked for quality by Divvy or ride_length was negative.

A new version of updated data set will be needed for our Analysis.

```
all_trips_v2 <- all_trips[!(all_trips$start_station_name == "HQ QR" | all_trips$ride_length<0),]

glimpse(all_trips_v2)
```

```
## Rows: 3,876,042
## Columns: 15
## $ ride_id            <chr> "22178529", "22178530", "22178531", "22178532", "22~
## $ started_at         <dttm> 2019-04-01 00:02:22, 2019-04-01 00:03:02, 2019-04-~
## $ ended_at           <dttm> 2019-04-01 00:09:48, 2019-04-01 00:20:30, 2019-04-~
## $ rideable_type      <chr> "6251", "6226", "5649", "4151", "3270", "3123", "64~
## $ start_station_id   <dbl> 81, 317, 283, 26, 202, 420, 503, 260, 211, 211, 304~
```

```
## $ start_station_name <chr> "Daley Center Plaza", "Wood St & Taylor St", "LaSal~
## $ end_station_id     <dbl> 56, 59, 174, 133, 129, 426, 500, 499, 211, 211, 232~
## $ end_station_name   <chr> "Desplaines St & Kinzie St", "Wabash Ave & Roosevel~
## $ member_casual      <chr> "member", "member", "member", "member", "member", "~
## $ date               <date> 2019-04-01, 2019-04-01, 2019-04-01, 2019-04-01, 20~
## $ month              <chr> "04", "04", "04", "04", "04", "04", "04", "04", "04~
## $ day                <chr> "01", "01", "01", "01", "01", "01", "01", "01", "01~
## $ year               <chr> "2019", "2019", "2019", "2019", "2019", "2019", "20~
## $ day_of_week        <chr> "Monday", "Monday", "Monday", "Monday", "Monday", "~
## $ ride_length        <dbl> 446, 1048, 252, 357, 1007, 257, 548, 383, 2137, 212~
```

Statistical Analysis

Mean

This is the Average of total ride_length / rides

```
mean(all_trips_v2$ride_length) #average (total ride length / rides)
```

```
## [1] 1479.139
```

Midpoint

This is showing the midpoint of the ride_length

```
median(all_trips_v2$ride_length) #midpoint number in the ascending array of ride lengths
```

```
## [1] 712
```

Min

This is showing the shortest ride in Seconds.

```
min(all_trips_v2$ride_length) #min of ride_length
```

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

Max

This is showing the longest ride in Seconds.

```
max(all_trips_v2$ride_length) #max of ride_length
```

```
## [1] 9387024
```

Summary

All the mean, median, min and max in one place.

```
summary(all_trips_v2$ride_length)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1     412     712    1479    1289 9387024
```

Annual v/s Casual Members

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = mean)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                3552.7502
## 2                     member                 850.0662
```

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = median)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                      1546
## 2                     member                       589
```

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = max)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                   9387024
## 2                     member                   9056634
```

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual, FUN = min)
```

```
##   all_trips_v2$member_casual all_trips_v2$ride_length
## 1                     casual                         2
## 2                     member                         1
```

Based on days

```
aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual + all_trips_v2$day_of_week, FUN = mean)
```

Ride Length of Casual and Annual members

```
##    all_trips_v2$member_casual all_trips_v2$day_of_week all_trips_v2$ride_length
## 1                      casual                   Friday                3773.8351
## 2                      member                   Friday                 824.5305
## 3                      casual                   Monday                3372.2869
## 4                      member                   Monday                 842.5726
## 5                      casual                 Saturday                3331.9138
## 6                      member                 Saturday                 968.9337
## 7                      casual                   Sunday                3581.4054
## 8                      member                   Sunday                 919.9746
## 9                      casual                 Thursday                3682.9847
## 10                     member                 Thursday                 823.9278
## 11                     casual                  Tuesday                3596.3599
## 12                     member                  Tuesday                 826.1427
## 13                     casual                Wednesday                3718.6619
## 14                     member                Wednesday                 823.9996
```

Here the days are in random order. So we will **order them in a generalized format** for easy viewing.

```
all_trips_v2$day_of_week <- ordered(all_trips_v2$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "
```

Now visualizing clearly. The average ride time by each day for members vs casual users.

```
#the average ride time by each day for members vs casual users

aggregate(all_trips_v2$ride_length ~ all_trips_v2$member_casual + all_trips_v2$day_of_week, FUN = mean)
```

```
##    all_trips_v2$member_casual all_trips_v2$day_of_week all_trips_v2$ride_length
## 1                      casual                   Sunday                3581.4054
## 2                      member                   Sunday                 919.9746
## 3                      casual                   Monday                3372.2869
## 4                      member                   Monday                 842.5726
## 5                      casual                  Tuesday                3596.3599
## 6                      member                  Tuesday                 826.1427
## 7                      casual                Wednesday                3718.6619
## 8                      member                Wednesday                 823.9996
## 9                      casual                 Thursday                3682.9847
## 10                     member                 Thursday                 823.9278
## 11                     casual                   Friday                3773.8351
## 12                     member                   Friday                 824.5305
## 13                     casual                 Saturday                3331.9138
## 14                     member                 Saturday                 968.9337
```

New version of data -v3

Now, we want to plot the data and get insights from the data.

```
all_trips_v3 <- all_trips_v2 %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%  #creates weekday field using wday()
  group_by(member_casual, weekday) %>%                  #groups by user type and weekday
  summarise(number_of_rides = n()                           #calculates the number of rides and average
  ,average_duration = mean(ride_length)) %>%                # calculates the average duration
  arrange(member_casual, weekday)
```

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

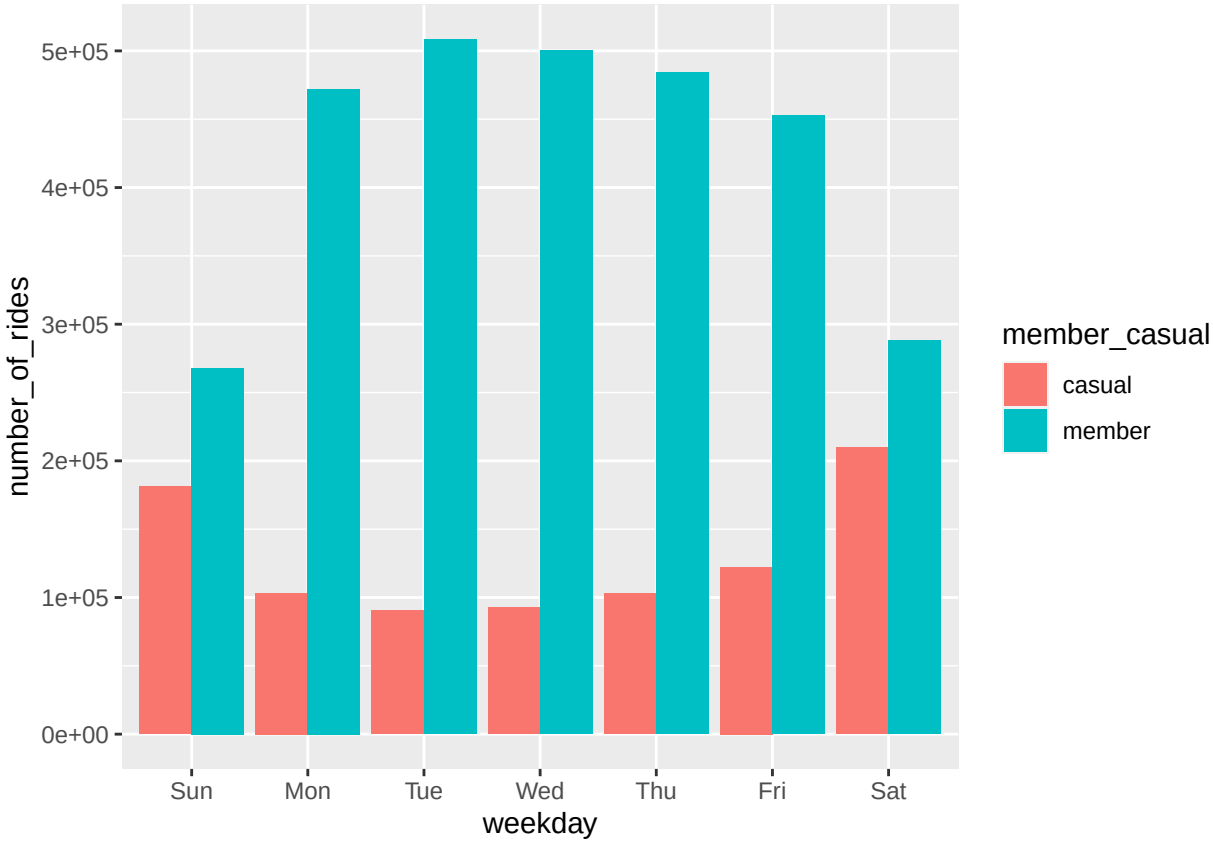
```
(all_trips_v3)
```

```
## # A tibble: 14 x 4
## # Groups:   member_casual [2]
##    member_casual weekday number_of_rides average_duration
##    <chr>         <ord>             <int>            <dbl>
##  1 casual        Sun              181293            3581.
##  2 casual        Mon              103296            3372.
##  3 casual        Tue               90510            3596.
##  4 casual        Wed               92457            3719.
##  5 casual        Thu              102679            3683.
##  6 casual        Fri              122404            3774.
##  7 casual        Sat              209543            3332.
##  8 member        Sun              267965             920.
##  9 member        Mon              472196             843.
## 10 member        Tue              508445             826.
## 11 member        Wed              500329             824.
## 12 member        Thu              484177             824.
## 13 member        Fri              452790             825.
## 14 member        Sat              287958             969.
```

Visualization

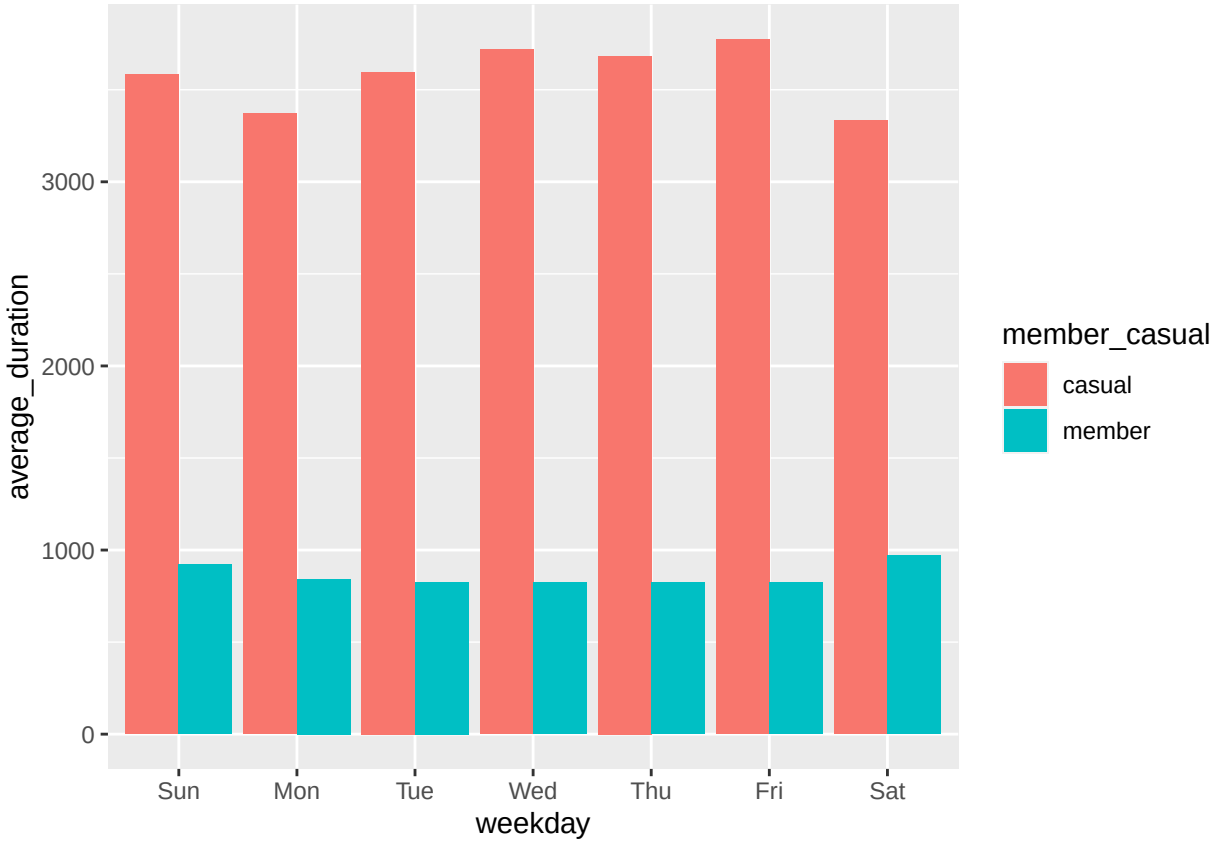
Number of rides by rider type

```
# Plot of No. of rides v/s week days - based on members.
all_trips_v3 %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge")
```



Average duration

```
all_trips_v3 %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  geom_col(position = "dodge")
```



Results

- Based on the data and Visualizations -
  - Based on Number of rides, We observe that **Member** uses bikes more **from Monday to Friday**, while **casual members** use bikes more on **saturdays and sundays**.
  - Based on duration **Casual members** uses bikes for approximately **58 Minutes and 15 seconds**. And **Annual members** uses bikes for approximately **16 Minutes and 35 seconds**.

Further Analysis

If required - for download

```
# Just remove the comments and input your download location for your analysis..

# write.csv(all_trips_v3, file = 'avg_ride_length.csv')
```

Thankyou

Credits and Inspired by - Google, Kevin Hartman.