Project_CS1_Capstone_v1

Gustavo M Silva

2022-08-01

Installing Packages

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

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

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

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("ggplot2")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("dplyr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
install.packages("magrittr")
```

```
## Installing package into '/cloud/lib/x86_64-pc-linux-gnu-library/4.2'
## (as 'lib' is unspecified)
```

```
library("tidyverse")
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
```

```
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library("lubridate")
```

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

```r
library("ggplot2")
library("dplyr")
library("magrittr")
```

```
##
## Attaching package: 'magrittr'
##
## The following object is masked from 'package:purrr':
##
##     set_names
##
## The following object is masked from 'package:tidyr':
##
##     extract
```

Reading and defining file names

```r
getwd()
```

```
## [1] "/cloud/project/Capstone_Case_Study_1"
```

```r
setwd("/cloud/project/Capstone_Case_Study_1")

January_2022 <- read_csv("2022_January_Trip_Data.csv")
```

```
## Rows: 103770 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
## 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.
```

```r
Febraury_2022 <- read_csv("2022_February_Trip_Data.csv")
```

```
## Rows: 115609 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
## 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.
```

```r
March_2022 <- read_csv("2022_March_Trip_Data.csv")
```

```
## Rows: 284042 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
## 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.
```

Reading Column names for comparison

```
colnames(January_2022)
```

```
##  [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"
```

```
colnames(Febraury_2022)
```

```
##  [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"
```

```
colnames(March_2022)
```

```
##  [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"
```

Renaming Columns for Data consistency

```
(January_2022 <- rename(January_2022
                  ,trip_id = ride_id
                  ,bikeid  = rideable_type
                  ,start_time = started_at
                  ,end_time = ended_at
                  ,from_station_name = start_station_name
                  ,from_station_id = start_station_id
                  ,to_station_name = end_station_name
                  ,to_station_id = end_station_id
                  ,usertype = member_casual))
```

```
## # A tibble: 103,770 x 13
##    trip_id       bikeid start_time          end_time            from_station_na~
##    <chr>         <chr>  <dttm>              <dttm>              <chr>
##  1 C2F7DD78E82E~ elect~ 2022-01-13 11:59:47 2022-01-13 12:02:44 Glenwood Ave & ~
##  2 A6CF8980A652~ elect~ 2022-01-10 08:41:56 2022-01-10 08:46:17 Glenwood Ave & ~
##  3 BD0F91DFF741~ class~ 2022-01-25 04:53:40 2022-01-25 04:58:01 Sheffield Ave &~
##  4 CBB80ED41910~ class~ 2022-01-04 00:18:04 2022-01-04 00:33:00 Clark St & Bryn~
##  5 DDC963BFDDA5~ class~ 2022-01-20 01:31:10 2022-01-20 01:37:12 Michigan Ave & ~
##  6 A39C6F6CC058~ class~ 2022-01-11 18:48:09 2022-01-11 18:51:31 Wood St & Chica~
##  7 BDC4AB637EDF~ class~ 2022-01-30 18:32:52 2022-01-30 18:49:26 Oakley Ave & Ir~
##  8 81751A3186E5~ class~ 2022-01-22 12:20:02 2022-01-22 12:32:06 Sheffield Ave &~
##  9 154222B86A33~ elect~ 2022-01-17 07:34:41 2022-01-17 08:00:08 Racine Ave & 15~
## 10 72DC25B2DD46~ class~ 2022-01-28 15:27:53 2022-01-28 15:35:16 LaSalle St & Ja~
## # ... with 103,760 more rows, and 8 more variables: from_station_id <chr>,
```

```
## #   to_station_name <chr>, to_station_id <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, usertype <chr>
```

```
(Febraury_2022 <- rename(Febraury_2022
                   ,trip_id = ride_id
                  ,bikeid  = rideable_type
                  ,start_time = started_at
                  ,end_time = ended_at
                  ,from_station_name = start_station_name
                  ,from_station_id = start_station_id
                  ,to_station_name = end_station_name
                  ,to_station_id = end_station_id
                  ,usertype = member_casual))
```

```
## # A tibble: 115,609 x 13
##    trip_id       bikeid start_time          end_time            from_station_na~
##    <chr>         <chr>  <dttm>              <dttm>              <chr>
##  1 E1E065E7ED28~ class~ 2022-02-19 18:08:41 2022-02-19 18:23:56 State St & Rand~
##  2 1602DCDC5B30~ class~ 2022-02-20 17:41:30 2022-02-20 17:45:56 Halsted St & Wr~
##  3 BE7DD2AF4B55~ class~ 2022-02-25 18:55:56 2022-02-25 19:09:34 State St & Rand~
##  4 A1789BDF8444~ class~ 2022-02-14 11:57:03 2022-02-14 12:04:00 Southport Ave &~
##  5 07DE78092C62~ class~ 2022-02-16 05:36:06 2022-02-16 05:39:00 State St & Rand~
##  6 9A2F204F04AB~ class~ 2022-02-07 09:51:57 2022-02-07 10:07:53 St. Clair St & ~
##  7 D1E6BB679BDE~ class~ 2022-02-14 10:38:54 2022-02-14 10:42:43 Wells St & Elm ~
##  8 DE23C1DC29B4~ class~ 2022-02-08 20:12:33 2022-02-08 20:21:16 State St & Rand~
##  9 3E314B0F4666~ elect~ 2022-02-25 13:49:05 2022-02-25 13:54:43 Larrabee St & A~
## 10 04ED4D3E37D2~ class~ 2022-02-06 07:36:15 2022-02-06 07:42:05 Morgan St & 18t~
## # ... with 115,599 more rows, and 8 more variables: from_station_id <chr>,
## #   to_station_name <chr>, to_station_id <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, usertype <chr>
```

```
(March_2022 <- rename(March_2022
                  ,trip_id = ride_id
                  ,bikeid  = rideable_type
                  ,start_time = started_at
                  ,end_time = ended_at
                  ,from_station_name = start_station_name
                  ,from_station_id = start_station_id
                  ,to_station_name = end_station_name
                  ,to_station_id = end_station_id
                  ,usertype = member_casual))
```

```
## # A tibble: 284,042 x 13
##    trip_id       bikeid start_time          end_time            from_station_na~
##    <chr>         <chr>  <dttm>              <dttm>              <chr>
##  1 47EC0A7F82E6~ class~ 2022-03-21 13:45:01 2022-03-21 13:51:18 Wabash Ave & Wa~
##  2 8494861979B0~ elect~ 2022-03-16 09:37:16 2022-03-16 09:43:34 Michigan Ave & ~
##  3 EFE527AF80B6~ class~ 2022-03-23 19:52:02 2022-03-23 19:54:48 Broadway & Berw~
##  4 9F446FD9DEE3~ class~ 2022-03-01 19:12:26 2022-03-01 19:22:14 Wabash Ave & Wa~
##  5 431128AD9AFF~ class~ 2022-03-21 18:37:01 2022-03-21 19:19:11 DuSable Lake Sh~
##  6 9AA8A13AF7A8~ class~ 2022-03-07 17:10:22 2022-03-07 17:15:04 Bissell St & Ar~
##  7 28E3387BFE2A~ elect~ 2022-03-10 17:21:22 2022-03-10 17:24:39 Bissell St & Ar~
##  8 74831EB3EA9C~ class~ 2022-03-05 12:31:37 2022-03-05 12:42:54 DuSable Lake Sh~
##  9 BD70E7114BC4~ elect~ 2022-03-17 17:32:44 2022-03-17 17:43:27 Western Ave & W~
## 10 482458CD09B6~ class~ 2022-03-04 19:06:32 2022-03-04 19:19:46 Sheffield Ave &~
```

```
## # ... with 284,032 more rows, and 8 more variables: from_station_id <chr>,
## #   to_station_name <chr>, to_station_id <chr>, start_lat <dbl>,
## #   start_lng <dbl>, end_lat <dbl>, end_lng <dbl>, usertype <chr>
```

Inspect for Data differences

```
str(January_2022)
```

```
## spec_tbl_df [103,770 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ trip_id          : chr [1:103770] "C2F7DD78E82EC875" "A6CF8980A652D272" "BD0F91DFF741C66D" "CBB80
##  $ bikeid           : chr [1:103770] "electric_bike" "electric_bike" "classic_bike" "classic_bike" .
##  $ start_time       : POSIXct[1:103770], format: "2022-01-13 11:59:47" "2022-01-10 08:41:56" ...
##  $ end_time         : POSIXct[1:103770], format: "2022-01-13 12:02:44" "2022-01-10 08:46:17" ...
##  $ from_station_name: chr [1:103770] "Glenwood Ave & Touhy Ave" "Glenwood Ave & Touhy Ave" "Sheffield
##  $ from_station_id  : chr [1:103770] "525" "525" "TA1306000016" "KA1504000151" ...
##  $ to_station_name  : chr [1:103770] "Clark St & Touhy Ave" "Clark St & Touhy Ave" "Greenview Ave & M
##  $ to_station_id    : chr [1:103770] "RP-007" "RP-007" "TA1307000001" "TA1309000021" ...
##  $ start_lat        : num [1:103770] 42 42 41.9 42 41.9 ...
##  $ start_lng        : num [1:103770] -87.7 -87.7 -87.7 -87.7 -87.6 ...
##  $ end_lat          : num [1:103770] 42 42 41.9 42 41.9 ...
##  $ end_lng          : num [1:103770] -87.7 -87.7 -87.7 -87.7 -87.6 ...
##  $ usertype         : chr [1:103770] "casual" "casual" "member" "casual" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   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>
```

```
str(Febraury_2022)
```

```
## spec_tbl_df [115,609 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ trip_id          : chr [1:115609] "E1E065E7ED285C02" "1602DCDC5B30FFE3" "BE7DD2AF4B55C4AF" "A1789
##  $ bikeid           : chr [1:115609] "classic_bike" "classic_bike" "classic_bike" "classic_bike" ...
##  $ start_time       : POSIXct[1:115609], format: "2022-02-19 18:08:41" "2022-02-20 17:41:30" ...
##  $ end_time         : POSIXct[1:115609], format: "2022-02-19 18:23:56" "2022-02-20 17:45:56" ...
##  $ from_station_name: chr [1:115609] "State St & Randolph St" "Halsted St & Wrightwood Ave" "State St
##  $ from_station_id  : chr [1:115609] "TA1305000029" "TA1309000061" "TA1305000029" "13235" ...
##  $ to_station_name  : chr [1:115609] "Clark St & Lincoln Ave" "Southport Ave & Wrightwood Ave" "Canal
##  $ to_station_id    : chr [1:115609] "13179" "TA1307000113" "13011" "13323" ...
##  $ start_lat        : num [1:115609] 41.9 41.9 41.9 41.9 41.9 ...
##  $ start_lng        : num [1:115609] -87.6 -87.6 -87.6 -87.7 -87.6 ...
##  $ end_lat          : num [1:115609] 41.9 41.9 41.9 42 41.9 ...
##  $ end_lng          : num [1:115609] -87.6 -87.7 -87.6 -87.6 -87.6 ...
```

```
##  $ usertype         : chr [1:115609] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   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>
```

```
str(March_2022)
```

```
## spec_tbl_df [284,042 x 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ trip_id          : chr [1:284042] "47EC0A7F82E65D52" "8494861979B0F477" "EFE527AF80B66109" "9F446
##  $ bikeid           : chr [1:284042] "classic_bike" "electric_bike" "classic_bike" "classic_bike" ..
##  $ start_time       : POSIXct[1:284042], format: "2022-03-21 13:45:01" "2022-03-16 09:37:16" ...
##  $ end_time         : POSIXct[1:284042], format: "2022-03-21 13:51:18" "2022-03-16 09:43:34" ...
##  $ from_station_name: chr [1:284042] "Wabash Ave & Wacker Pl" "Michigan Ave & Oak St" "Broadway & Be
##  $ from_station_id  : chr [1:284042] "TA1307000131" "13042" "13109" "TA1307000131" ...
##  $ to_station_name  : chr [1:284042] "Kingsbury St & Kinzie St" "Orleans St & Chestnut St (NEXT Apts
##  $ to_station_id    : chr [1:284042] "KA1503000043" "620" "15578" "TA1305000025" ...
##  $ start_lat        : num [1:284042] 41.9 41.9 42 41.9 41.9 ...
##  $ start_lng        : num [1:284042] -87.6 -87.6 -87.7 -87.6 -87.6 ...
##  $ end_lat          : num [1:284042] 41.9 41.9 42 41.9 41.9 ...
##  $ end_lng          : num [1:284042] -87.6 -87.6 -87.7 -87.6 -87.7 ...
##  $ usertype         : chr [1:284042] "member" "member" "member" "member" ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   ride_id = col_character(),
##   ..   rideable_type = col_character(),
##   ..   started_at = col_datetime(format = ""),
##   ..   ended_at = col_datetime(format = ""),
##   ..   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>
```

Convert ride_id and rideable_type to character so that they can stack correctly

```
January_2022 %<>% mutate(January_2022, trip_id = as.character(trip_id)
                  ,bikeid = as.character(bikeid))
Febraury_2022 %<>% mutate(Febraury_2022, trip_id = as.character(trip_id)
                  ,bikeid = as.character(bikeid))
March_2022 %<>% mutate(March_2022, trip_id = as.character(trip_id)
                  ,bikeid = as.character(bikeid))
```

Stack individual quarter's data frames into one big data frame

```
all_trips <- bind_rows(January_2022, Febraury_2022, March_2022)
```

Remove unecessary columns

```
all_trips_v2 = subset(all_trips, select = -c(start_lat, start_lng, end_lat, end_lng))
```

CleanUp Data

```
colnames(all_trips_v2)  #List of column names
```

```
## [1] "trip_id"           "bikeid"            "start_time"
## [4] "end_time"          "from_station_name" "from_station_id"
## [7] "to_station_name"   "to_station_id"     "usertype"
```

```
nrow(all_trips_v2)  #How many rows are in data frame?
```

```
## [1] 503421
```

```
dim(all_trips_v2)  #Dimensions of the data frame?
```

```
## [1] 503421      9
```

```
head(all_trips_v2)  #See the first 6 rows of data frame.  Also tail(all_trips)
```

```
## # A tibble: 6 x 9
##   trip_id        bikeid start_time          end_time            from_station_na~
##   <chr>          <chr>  <dttm>              <dttm>              <chr>
## 1 C2F7DD78E82EC~ elect~ 2022-01-13 11:59:47 2022-01-13 12:02:44 Glenwood Ave & ~
## 2 A6CF8980A652D~ elect~ 2022-01-10 08:41:56 2022-01-10 08:46:17 Glenwood Ave & ~
## 3 BD0F91DFF741C~ class~ 2022-01-25 04:53:40 2022-01-25 04:58:01 Sheffield Ave &~
## 4 CBB80ED419105~ class~ 2022-01-04 00:18:04 2022-01-04 00:33:00 Clark St & Bryn~
## 5 DDC963BFDDA51~ class~ 2022-01-20 01:31:10 2022-01-20 01:37:12 Michigan Ave & ~
## 6 A39C6F6CC0586~ class~ 2022-01-11 18:48:09 2022-01-11 18:51:31 Wood St & Chica~
## # ... with 4 more variables: from_station_id <chr>, to_station_name <chr>,
## #   to_station_id <chr>, usertype <chr>
```

```
str(all_trips_v2)  #See list of columns and data types (numeric, character, etc)
```

```
## tibble [503,421 x 9] (S3: tbl_df/tbl/data.frame)
##  $ trip_id          : chr [1:503421] "C2F7DD78E82EC875" "A6CF8980A652D272" "BD0F91DFF741C66D" "CBB80
##  $ bikeid           : chr [1:503421] "electric_bike" "electric_bike" "classic_bike" "classic_bike" .
##  $ start_time       : POSIXct[1:503421], format: "2022-01-13 11:59:47" "2022-01-10 08:41:56" ...
##  $ end_time         : POSIXct[1:503421], format: "2022-01-13 12:02:44" "2022-01-10 08:46:17" ...
##  $ from_station_name: chr [1:503421] "Glenwood Ave & Touhy Ave" "Glenwood Ave & Touhy Ave" "Sheffiel
##  $ from_station_id  : chr [1:503421] "525" "525" "TA1306000016" "KA1504000151" ...
```

```
##  $ to_station_name  : chr [1:503421] "Clark St & Touhy Ave" "Clark St & Touhy Ave" "Greenview Ave & I
##  $ to_station_id    : chr [1:503421] "RP-007" "RP-007" "TA1307000001" "TA1309000021" ...
##  $ usertype         : chr [1:503421] "casual" "casual" "member" "casual" ...
```

```
summary(all_trips_v2)  #Statistical summary of data. Mainly for numerics
```

```
##    trip_id             bikeid            start_time
##  Length:503421      Length:503421      Min.   :2022-01-01 00:00:05.00
##  Class :character   Class :character   1st Qu.:2022-02-08 18:01:49.00
##  Mode  :character   Mode  :character   Median :2022-03-04 20:13:12.00
##                                        Mean   :2022-02-25 19:04:47.05
##                                        3rd Qu.:2022-03-17 16:21:33.00
##                                        Max.   :2022-03-31 23:59:47.00
##     end_time                       from_station_name  from_station_id
##  Min.   :2022-01-01 00:01:48.00   Length:503421      Length:503421
##  1st Qu.:2022-02-08 18:18:23.00   Class :character   Class :character
##  Median :2022-03-04 20:32:03.00   Mode  :character   Mode  :character
##  Mean   :2022-02-25 19:21:38.25
##  3rd Qu.:2022-03-17 16:38:40.00
##  Max.   :2022-04-01 22:10:12.00
##  to_station_name    to_station_id        usertype
##  Length:503421      Length:503421      Length:503421
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
```

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

```
all_trips_v2$date <- as.Date(all_trips_v2$start_time) #The default format is yyyy-mm-dd
all_trips_v2$month <- format(as.Date(all_trips_v2$date), "%m")
all_trips_v2$day <- format(as.Date(all_trips_v2$date), "%d")
all_trips_v2$year <- format(as.Date(all_trips_v2$date), "%Y")
all_trips_v2$day_of_week <- format(as.Date(all_trips_v2$date), "%A")
```

Calculated field to obtain the Ride Length

```
all_trips_v2$ride_length <- difftime(all_trips_v2$end_time,all_trips_v2$start_time)
```

Inspect the structure of the columns

```
str(all_trips_v2)
```

```
## tibble [503,421 x 15] (S3: tbl_df/tbl/data.frame)
##  $ trip_id          : chr [1:503421] "C2F7DD78E82EC875" "A6CF8980A652D272" "BD0F91DFF741C66D" "CBB80
##  $ bikeid           : chr [1:503421] "electric_bike" "electric_bike" "classic_bike" "classic_bike" .
##  $ start_time       : POSIXct[1:503421], format: "2022-01-13 11:59:47" "2022-01-10 08:41:56" ...
##  $ end_time         : POSIXct[1:503421], format: "2022-01-13 12:02:44" "2022-01-10 08:46:17" ...
##  $ from_station_name: chr [1:503421] "Glenwood Ave & Touhy Ave" "Glenwood Ave & Touhy Ave" "Sheffiel
##  $ from_station_id  : chr [1:503421] "525" "525" "TA1306000016" "KA1504000151" ...
##  $ to_station_name  : chr [1:503421] "Clark St & Touhy Ave" "Clark St & Touhy Ave" "Greenview Ave & I
##  $ to_station_id    : chr [1:503421] "RP-007" "RP-007" "TA1307000001" "TA1309000021" ...
##  $ usertype         : chr [1:503421] "casual" "casual" "member" "casual" ...
```

```
##  $ date             : Date[1:503421], format: "2022-01-13" "2022-01-10" ...
##  $ month            : chr [1:503421] "01" "01" "01" "01" ...
##  $ day              : chr [1:503421] "13" "10" "25" "04" ...
##  $ year             : chr [1:503421] "2022" "2022" "2022" "2022" ...
##  $ day_of_week      : chr [1:503421] "Thursday" "Monday" "Tuesday" "Tuesday" ...
##  $ ride_length      : 'difftime' num [1:503421] 177 261 261 896 ...
##   ..- attr(*, "units")= chr "secs"
```

Convert "ride_length" from Factor to numeric so we can run calculations on the data

```
is.factor(all_trips_v2$ride_length)
```

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

```
all_trips_v2$ride_length <- as.numeric(as.character(all_trips_v2$ride_length))
is.numeric(all_trips_v2$ride_length)
```

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

Summary Analysis

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

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    -356     306     525    1011     946 2061244
```

Compare members and casual users

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

```
##   all_trips_v2$usertype all_trips_v2$ride_length
## 1                casual                1879.5904
## 2                member                 709.4548
```

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

```
##   all_trips_v2$usertype all_trips_v2$ride_length
## 1                casual                      774
## 2                member                      466
```

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

```
##   all_trips_v2$usertype all_trips_v2$ride_length
## 1                casual                 2061244
## 2                member                   93594
```

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

```
##   all_trips_v2$usertype all_trips_v2$ride_length
## 1                casual                    -356
## 2                member                       0
```

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

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

```
##    all_trips_v2$usertype all_trips_v2$day_of_week all_trips_v2$ride_length
```

```
## 1                casual                  Friday               1479.3246
## 2                member                  Friday                687.7295
## 3                casual                  Monday               1943.8432
## 4                member                  Monday                719.4456
## 5                casual                Saturday               2122.5861
## 6                member                Saturday                770.4544
## 7                casual                  Sunday               2185.6870
## 8                member                  Sunday                784.5626
## 9                casual                Thursday               1828.2596
## 10               member                Thursday                664.4349
## 11               casual                 Tuesday               1469.4102
## 12               member                 Tuesday                678.1599
## 13               casual               Wednesday               1793.5348
## 14               member               Wednesday                698.2336
```

Sort by Week Day

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

Analyze ridership data by type and weekday

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

```
##    all_trips_v2$usertype all_trips_v2$day_of_week all_trips_v2$ride_length
## 1                 casual                  Sunday               2185.6870
## 2                 member                  Sunday                784.5626
## 3                 casual                  Monday               1943.8432
## 4                 member                  Monday                719.4456
## 5                 casual                 Tuesday               1469.4102
## 6                 member                 Tuesday                678.1599
## 7                 casual               Wednesday               1793.5348
## 8                 member               Wednesday                698.2336
## 9                 casual                Thursday               1828.2596
## 10                member                Thursday                664.4349
## 11                casual                  Friday               1479.3246
## 12                member                  Friday                687.7295
## 13                casual                Saturday               2122.5861
## 14                member                Saturday                770.4544
```

Visualize data by type and weekday

```
all_trips_v2 %>%
  mutate (day_of_week = wday(start_time, label = TRUE)) %>%  #creates weekday field using wday()
group_by(usertype, day_of_week) %>%  #groups by usertype and weekday
summarise(number_of_rides = n()                          #calculates the number of rides and average duratio
  ,average_duration = mean(ride_length))    %>% # calculates the average duration
arrange(usertype, day_of_week)                           # sorts
```

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

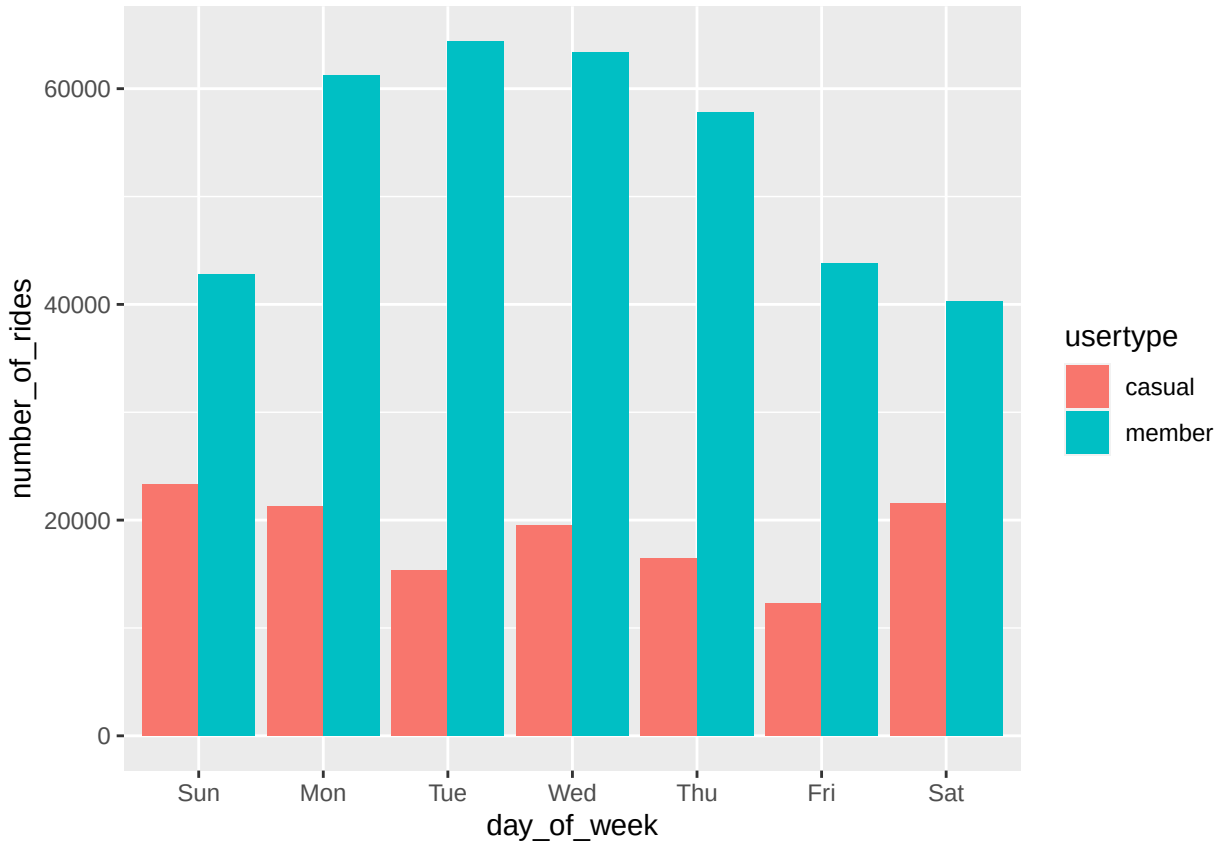
```
## # A tibble: 14 x 4
## # Groups:   usertype [2]
##    usertype day_of_week number_of_rides average_duration
```

```
##    <chr>    <ord>                <int>            <dbl>
##  1 casual   Sun                  23296            2186.
##  2 casual   Mon                  21283            1944.
##  3 casual   Tue                  15335            1469.
##  4 casual   Wed                  19552            1794.
##  5 casual   Thu                  16446            1828.
##  6 casual   Fri                  12313            1479.
##  7 casual   Sat                  21593            2123.
##  8 member   Sun                  42748             785.
##  9 member   Mon                  61198             719.
## 10 member   Tue                  64420             678.
## 11 member   Wed                  63351             698.
## 12 member   Thu                  57787             664.
## 13 member   Fri                  43804             688.
## 14 member   Sat                  40295             770.
```

Visualize the number of rides by rider type

```
all_trips_v2 %>%
  mutate(day_of_week = wday(start_time, label = TRUE)) %>%
  group_by(usertype, day_of_week) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(usertype, day_of_week)  %>%
  ggplot(aes(x = day_of_week, y = number_of_rides, fill = usertype)) +
  geom_col(position = "dodge")
```

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

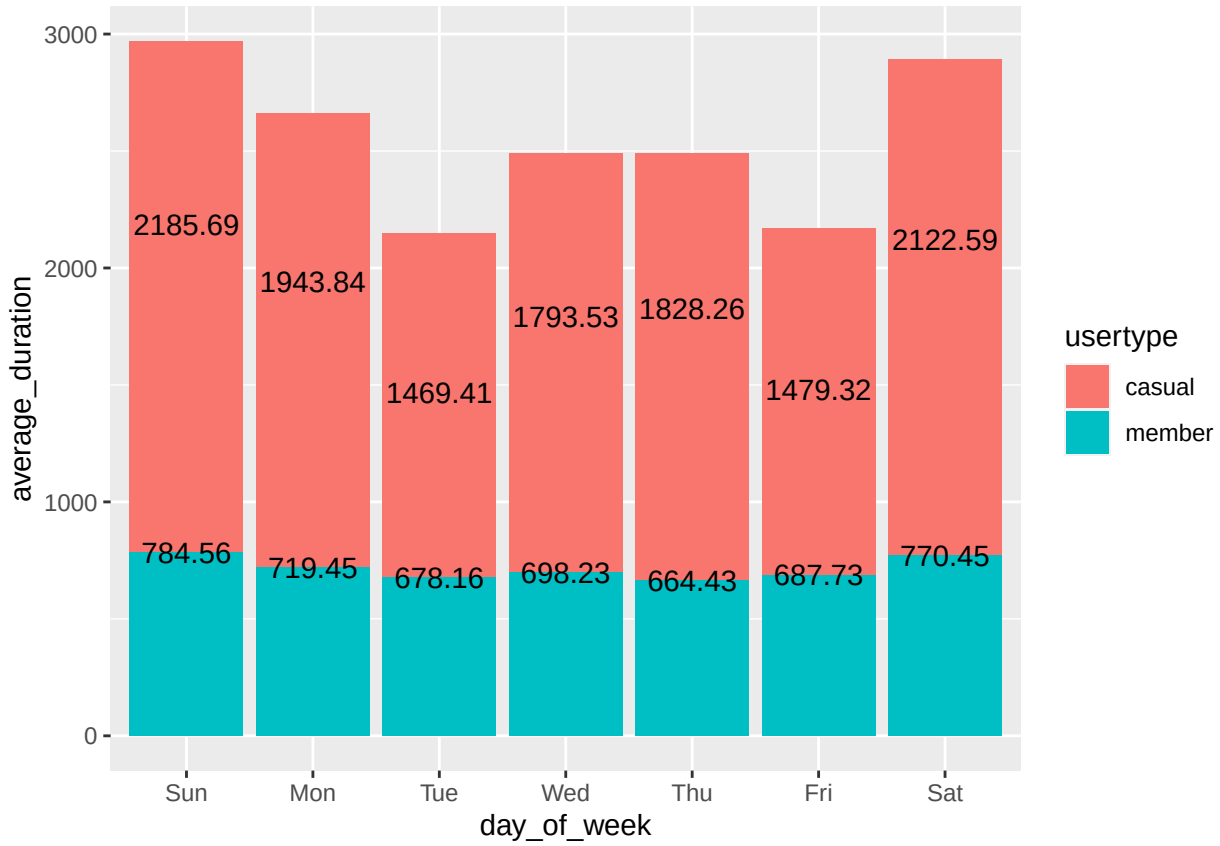


Visualize for average duration

```
all_trips_v2 %>%
  mutate(day_of_week = wday(start_time, label = TRUE)) %>%
  group_by(usertype, day_of_week) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(usertype, day_of_week)  %>%
  ggplot(aes(x = day_of_week, y = average_duration, fill = usertype)) +
  geom_col(position = "stack") +
  geom_text(
    aes(label = sprintf("%0.2f", average_duration),
    hjust = 0.5, nudge_x = 1.5))
```

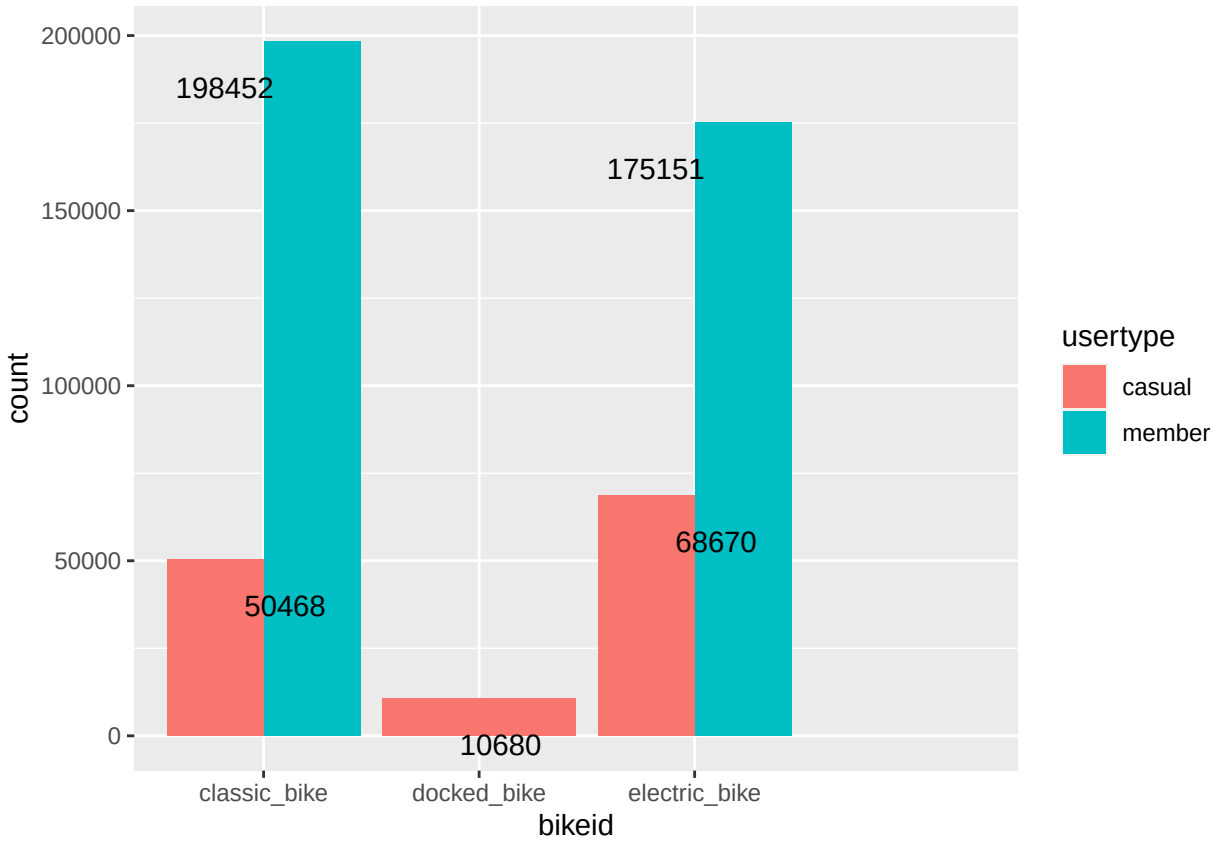
```
## `summarise()` has grouped output by 'usertype'. You can override using the
## `.groups` argument.
```

```
## Warning: Ignoring unknown aesthetics: nudge_x
```




```
#geom_text(aes(label=sprintf("%0.2f", Proportion))
###  geom_text(aes(label = average_duration, vjust = 0.5, colour = "White"))
```

```
all_trips_v2 %>%
 ggplot(aes(x = bikeid, fill = usertype)) +
  geom_bar(position = 'dodge') +
  geom_text(aes(label = ..count..), stat = "count", nudge_x = 1.5, hjust = 4.2, vjust = 2.7, colour = "
```



Findings

- Users with memberships tend to use the bikes more often than casual users.
- Casual users spend more time on average than user swith memberships plans, specially during Saturday and Sunday.
- The average ride duration of casual users is almost 3x more than the average ride duration of members users, that is probably because users with memberships use the bikes to commute to work and not for pleasure time.
- Members are not showing much interested in the docked bikes.
- Members have a higher interest in the Classic bike
- Would be interesting to have a picture of the demographic

Recommendation

- Get the casual users converted to membership plans. Develop a new pricing system based on the ride duration for casual users, where the price to become a member would be the same for higher ride durations.
- Offer Weekends or Seasonal pricing packages.
- Reconsider the need to having Docked bikes, and confirm if the existence of them justify the cost.