

Bikeshare-Project-R-scripts

Antik Salimullah

2023

#Load the libraries that will help you clean and analyze data

```
library(lubridate)

## Loading required package: timechange

##
## Attaching package: 'lubridate'

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

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_conflict
s() —
## ✗ lubridate::as.difftime() masks base::as.difftime()
## ✗ lubridate::date()        masks base::date()
## ✗ dplyr::filter()          masks stats::filter()
## ✗ lubridate::intersect()   masks base::intersect()
## ✗ dplyr::lag()              masks stats::lag()
## ✗ lubridate::setdiff()     masks base::setdiff()
## ✗ lubridate::union()       masks base::union()

library(ggplot2)
library(dplyr)
library(hms)

##
## Attaching package: 'hms'
##
## The following object is masked from 'package:lubridate':
```

```
##
##      hms

# This is how the data was imported into the file

X202111_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2021
11-divvy-tripdata.csv")

## Rows: 359978 Columns: 13
## — Column specification —————
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dtm  (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.

X202112_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2021
12-divvy-tripdata.csv")

## Rows: 247540 Columns: 13
## — Column specification —————
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dtm  (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.

X202201_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2022
01-divvy-tripdata.csv")

## Rows: 103770 Columns: 13
## — Column specification —————
## Delimiter: ","
## chr  (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
## dbl  (4): start_lat, start_lng, end_lat, end_lng
## dtm  (2): started_at, ended_at
##
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
X202202_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/202202-divvy-tripdata.csv")
```

```
## Rows: 115609 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end_station_name, end_station_id, end_station_latitude, end_station_longitude
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
X202203_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/202203-divvy-tripdata.csv")
```

```
## Rows: 284042 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end_station_name, end_station_id, end_station_latitude, end_station_longitude
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
X202204_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/202204-divvy-tripdata.csv")
```

```
## Rows: 371249 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, end_station_name, end_station_id, end_station_latitude, end_station_longitude
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
X202205_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2022
05-divvy-tripdata.csv")
```

```
## Rows: 634858 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.
```

```
X202206_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2022
06-divvy-tripdata.csv")
```

```
## Rows: 769204 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.
```

```
X202207_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2022
07-divvy-tripdata.csv")
```

```
## Rows: 823488 Columns: 13
```

```
## — Column specification —————
```

```
## Delimiter: ","
```

```
## chr (7): ride_id, rideable_type, start_station_name, start_station_id, en
d_...
```

```
## dbl (4): start_lat, start_lng, end_lat, end_lng
```

```
## dtm (2): started_at, ended_at
```

```
##
```

```
## i Use `spec()` to retrieve the full column specification for this data.
```

```
## i Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.
```

```
X202208_divvy_tripdata <- read_csv("Desktop/Bikeshare project files/CSV1/2022
08-divvy-tripdata.csv")
```



```

, 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: 359,978 x 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr> <dtm>      <dtm>      <chr> <
chr>
## 1 7C00A93E10556... elect... 2021-11-27 13:27:38 2021-11-27 13:46:38 <NA> <
NA>
## 2 90854840DFD50... elect... 2021-11-27 13:38:25 2021-11-27 13:56:10 <NA> <
NA>
## 3 0A7D10CDD1440... elect... 2021-11-26 22:03:34 2021-11-26 22:05:56 <NA> <
NA>
## 4 2F3BE33085BCF... elect... 2021-11-27 09:56:49 2021-11-27 10:01:50 <NA> <
NA>
## 5 D67B4781A1992... elect... 2021-11-26 19:09:28 2021-11-26 19:30:41 <NA> <
NA>
## 6 02F85C2C3C5F7... elect... 2021-11-26 18:34:07 2021-11-26 18:52:49 Michig... 1
3042
## 7 EF780B807EF78... elect... 2021-11-27 13:31:12 2021-11-27 13:37:12 <NA> <
NA>
## 8 17069CC749126... elect... 2021-11-27 14:33:56 2021-11-27 14:34:38 <NA> <
NA>
## 9 93FC4662BDC5C... elect... 2021-11-27 09:14:33 2021-11-27 09:19:36 <NA> <
NA>
## 10 B06B064398A36... elect... 2021-11-27 16:13:31 2021-11-27 16:22:50 <NA> <
NA>
## # ... with 359,968 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   1from_station_name, 2from_station_id

(X202112_divvy_tripdata <- rename(X202112_divvy_tripdata
, 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: 247,540 x 13
##   trip_id      bikeid start_time      end_time      from_...1 f

```

```

rom_...2
##   <chr>           <chr> <dtm>           <dtm>           <chr>   <
chr>
##  1 46F8167220E44... elect... 2021-12-07 15:06:07 2021-12-07 15:13:42 Laflin... 1
3307
##  2 73A77762838B3... elect... 2021-12-11 03:43:29 2021-12-11 04:10:23 LaSall... K
P1705...
##  3 4CF42452054F5... elect... 2021-12-15 23:10:28 2021-12-15 23:23:14 Halste... K
A1504...
##  4 3278BA87BF698... class... 2021-12-26 16:16:10 2021-12-26 16:30:53 Halste... K
A1504...
##  5 6FF54232576A3... elect... 2021-12-30 11:31:05 2021-12-30 11:51:21 Leavit... 1
8058
##  6 93E8D79490E3A... class... 2021-12-01 18:28:36 2021-12-01 18:38:03 Wabash... S
L-012
##  7 135518383C9CA... elect... 2021-12-09 15:49:51 2021-12-09 15:55:49 Laflin... 1
3307
##  8 CDF93C212EA9B... class... 2021-12-31 13:02:08 2021-12-31 13:20:59 Clark ... K
A1503...
##  9 0FCA94C5F2653... elect... 2021-12-20 14:25:53 2021-12-20 14:35:48 Clark ... K
A1503...
## 10 8E913048B7BCA... class... 2021-12-19 16:58:32 2021-12-19 17:15:01 Fairba... 1
8003
## # ... with 247,530 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   1from_station_name, 2from_station_id

(X202201_divvy_tripdata <- rename(X202201_divvy_tripdata
  ,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 × 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>           <chr> <dtm>           <dtm>           <chr>   <
chr>
##  1 C2F7DD78E82EC... elect... 2022-01-13 11:59:47 2022-01-13 12:02:44 Glenwo... 5
25
##  2 A6CF8980A652D... elect... 2022-01-10 08:41:56 2022-01-10 08:46:17 Glenwo... 5
25
##  3 BD0F91DFF741C... class... 2022-01-25 04:53:40 2022-01-25 04:58:01 Sheffi... T
A1306...

```

```

## 4 CBB80ED419105... class... 2022-01-04 00:18:04 2022-01-04 00:33:00 Clark ... K
A1504...
## 5 DDC963BFDDA51... class... 2022-01-20 01:31:10 2022-01-20 01:37:12 Michig... T
A1309...
## 6 A39C6F6CC0586... class... 2022-01-11 18:48:09 2022-01-11 18:51:31 Wood S... 6
37
## 7 BDC4AB637EDF9... class... 2022-01-30 18:32:52 2022-01-30 18:49:26 Oakley... K
A1504...
## 8 81751A3186E59... class... 2022-01-22 12:20:02 2022-01-22 12:32:06 Sheffi... T
A1306...
## 9 154222B86A338... elect... 2022-01-17 07:34:41 2022-01-17 08:00:08 Racine... 1
3304
## 10 72DC25B2DD467... class... 2022-01-28 15:27:53 2022-01-28 15:35:16 LaSall... T
A1309...
## # ... with 103,760 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   ^from_station_name, ^2from_station_id

(X202202_divvy_tripdata <- rename(X202202_divvy_tripdata
  ,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 × 13
##   trip_id      bikeid start_time      end_time      from_...^1 f
rom_...^2
##   <chr>      <chr>  <dtm>      <dtm>      <chr>    <
chr>
## 1 E1E065E7ED285... class... 2022-02-19 18:08:41 2022-02-19 18:23:56 State ... T
A1305...
## 2 1602DCDC5B30F... class... 2022-02-20 17:41:30 2022-02-20 17:45:56 Halste... T
A1309...
## 3 BE7DD2AF4B55C... class... 2022-02-25 18:55:56 2022-02-25 19:09:34 State ... T
A1305...
## 4 A1789BDF84441... class... 2022-02-14 11:57:03 2022-02-14 12:04:00 Southp... 1
3235
## 5 07DE78092C62F... class... 2022-02-16 05:36:06 2022-02-16 05:39:00 State ... T
A1305...
## 6 9A2F204F04AB7... class... 2022-02-07 09:51:57 2022-02-07 10:07:53 St. Cl... 1
3016
## 7 D1E6BB679BDED... class... 2022-02-14 10:38:54 2022-02-14 10:42:43 Wells ... K
A1504...
## 8 DE23C1DC29B40... class... 2022-02-08 20:12:33 2022-02-08 20:21:16 State ... T

```



```

A1305...
## 9 3E314B0F46667... elect... 2022-02-25 13:49:05 2022-02-25 13:54:43 Larrab... T
A1309...
## 10 04ED4D3E37D23... class... 2022-02-06 07:36:15 2022-02-06 07:42:05 Morgan... 1
3163
## # ... with 115,599 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   ^from_station_name, ^from_station_id

(X202203_divvy_tripdata <- rename(X202203_divvy_tripdata
  ,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 × 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr>   <dtm>      <dtm>      <chr>   <
chr>
## 1 47EC0A7F82E65... class... 2022-03-21 13:45:01 2022-03-21 13:51:18 Wabash... T
A1307...
## 2 8494861979B0F... elect... 2022-03-16 09:37:16 2022-03-16 09:43:34 Michig... 1
3042
## 3 EFE527AF80B66... class... 2022-03-23 19:52:02 2022-03-23 19:54:48 Broadw... 1
3109
## 4 9F446FD9DEE3F... class... 2022-03-01 19:12:26 2022-03-01 19:22:14 Wabash... T
A1307...
## 5 431128AD9AFFE... class... 2022-03-21 18:37:01 2022-03-21 19:19:11 DuSabl... L
F-005
## 6 9AA8A13AF7A85... class... 2022-03-07 17:10:22 2022-03-07 17:15:04 Bissel... 1
3059
## 7 28E3387BFE2A5... elect... 2022-03-10 17:21:22 2022-03-10 17:24:39 Bissel... 1
3059
## 8 74831EB3EA9CF... class... 2022-03-05 12:31:37 2022-03-05 12:42:54 DuSabl... L
F-005
## 9 BD70E7114BC48... elect... 2022-03-17 17:32:44 2022-03-17 17:43:27 Wester... 1
3068
## 10 482458CD09B6F... class... 2022-03-04 19:06:32 2022-03-04 19:19:46 Sheffi... T
A1309...
## # ... with 284,032 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   ^from_station_name, ^from_station_id

```

```
(X202204_divvy_tripdata <- rename(X202204_divvy_tripdata
  ,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: 371,249 × 13

##	trip_id	bikeid	start_time	end_time	from_... ¹	f
##	<chr>	<chr>	<dtm>	<dtm>	<chr>	<
##	1 3564070EEFD12...	elect...	2022-04-06 17:42:48	2022-04-06 17:54:36	Paulin...	5
##	2 0B820C7FCF22F...	class...	2022-04-24 19:23:07	2022-04-24 19:43:17	Wentwo...	1
##	3 89EEEE32293F0...	class...	2022-04-20 19:29:08	2022-04-20 19:35:16	Halste...	T
##	4 84D4751AEB318...	class...	2022-04-22 21:14:06	2022-04-22 21:23:29	Wentwo...	1
##	5 5664BCF0D1DE7...	elect...	2022-04-16 15:56:30	2022-04-16 16:02:11	Halste...	T
##	6 AA9EB7BD2E1FC...	class...	2022-04-21 16:52:33	2022-04-21 16:56:51	Despla...	1
##	7 9E10667D54A73...	class...	2022-04-04 17:10:52	2022-04-04 17:15:30	Despla...	1
##	8 22291F983B344...	class...	2022-04-05 08:52:21	2022-04-05 09:04:45	Frankl...	T
##	9 7E14444DA4A4B...	elect...	2022-04-29 11:38:20	2022-04-29 11:38:52	Halste...	3
##	10 D55A28D2B63A7...	elect...	2022-04-29 23:38:33	2022-04-29 23:40:02	Halste...	3

... with 371,239 more rows, 7 more variables: to_station_name <chr>,
 ## # to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
 ## # end_lng <dbl>, usertype <chr>, and abbreviated variable names
 ## # ¹from_station_name, ²from_station_id

```
(X202205_divvy_tripdata <- rename(X202205_divvy_tripdata
  ,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: 634,858 × 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr> <dtm>      <dtm>      <chr> <
chr>
## 1 EC2DE40644C6B... class... 2022-05-23 23:06:58 2022-05-23 23:40:19 Wabash... T
A1307...
## 2 1C31AD03897EE... class... 2022-05-11 08:53:28 2022-05-11 09:31:22 DuSabl... 1
3300
## 3 1542FBEC83041... class... 2022-05-26 18:36:28 2022-05-26 18:58:18 Clinto... T
A1305...
## 4 6FF5985292452... class... 2022-05-10 07:30:07 2022-05-10 07:38:49 Clinto... T
A1305...
## 5 483C52CAAE12E... class... 2022-05-10 17:31:56 2022-05-10 17:36:57 Clinto... T
A1305...
## 6 C0A3AA5A614DC... class... 2022-05-04 14:48:55 2022-05-04 14:56:04 Carpen... 1
3196
## 7 F2AF43A242DF4... class... 2022-05-27 12:41:48 2022-05-27 12:50:41 Noble ... 1
3290
## 8 377BE1F5F0E39... docke... 2022-05-29 19:19:24 2022-05-29 19:31:34 Halste... T
A1309...
## 9 B136E0C969773... class... 2022-05-16 17:48:44 2022-05-16 18:05:26 Clinto... T
A1305...
## 10 75F6A50A05E0A... elect... 2022-05-11 07:29:29 2022-05-11 07:30:57 Southp... 1
3235
## # ... with 634,848 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   1from_station_name, 2from_station_id

(X202206_divvy_tripdata <- rename(X202206_divvy_tripdata
, 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: 769,204 × 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr> <dtm>      <dtm>      <chr> <
chr>
## 1 600CFD130D0FD... elect... 2022-06-30 17:27:53 2022-06-30 17:35:15 <NA> <

```

```

NA>
## 2 F5E6B5C1682C6... elect... 2022-06-30 18:39:52 2022-06-30 18:47:28 <NA> <
NA>
## 3 B6EB6D27BAD77... elect... 2022-06-30 11:49:25 2022-06-30 12:02:54 <NA> <
NA>
## 4 C9C320375DE1D... elect... 2022-06-30 11:15:25 2022-06-30 11:19:43 <NA> <
NA>
## 5 56C055851023B... elect... 2022-06-29 23:36:50 2022-06-29 23:45:17 <NA> <
NA>
## 6 B664188E8163D... elect... 2022-06-30 16:42:10 2022-06-30 16:58:22 <NA> <
NA>
## 7 338C05A3E90D6... elect... 2022-06-30 18:39:07 2022-06-30 19:05:02 <NA> <
NA>
## 8 C037F5F410778... elect... 2022-06-30 12:46:14 2022-06-30 14:12:48 <NA> <
NA>
## 9 C19B08D794D1C... elect... 2022-06-30 11:09:38 2022-06-30 11:10:25 <NA> <
NA>
## 10 6E9E3A041C14E... elect... 2022-06-30 11:05:46 2022-06-30 11:09:11 <NA> <
NA>
## # ... with 769,194 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   ^1from_station_name, ^2from_station_id

(X202207_divvy_tripdata <- rename(X202207_divvy_tripdata
  ,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: 823,488 × 13
##   trip_id      bikeid start_time      end_time      from_...^1 f
rom_...^2
##   <chr>      <chr> <dtm>      <dtm>      <chr> <
chr>
## 1 954144C2F67B1... class... 2022-07-05 08:12:47 2022-07-05 08:24:32 Ashlan... 1
3224
## 2 292E027607D21... class... 2022-07-26 12:53:38 2022-07-26 12:55:31 Buckin... 1
5541
## 3 57765852588AD... class... 2022-07-03 13:58:49 2022-07-03 14:06:32 Buckin... 1
5541
## 4 B5B6BE4431459... class... 2022-07-31 17:44:21 2022-07-31 18:42:50 Buckin... 1
5541
## 5 A4C331F2A00E7... class... 2022-07-13 19:49:06 2022-07-13 20:15:24 Wabash... T
A1307...

```

```

## 6 579D73BE2ED88... elect... 2022-07-01 17:04:35 2022-07-01 17:13:18 Despla... 1
5535
## 7 EFE518CCEE333... class... 2022-07-18 18:11:01 2022-07-18 18:22:30 Marque... 2
0239
## 8 315FE7B7B3F6D... class... 2022-07-28 20:38:18 2022-07-28 21:09:11 Wabash... T
A1307...
## 9 EE3C4A1E66766... class... 2022-07-10 22:55:59 2022-07-10 23:01:32 Wabash... T
A1307...
## 10 1EE6C93A547A1... elect... 2022-07-10 09:35:58 2022-07-10 09:47:25 Ashlan... 1
3224
## # ... with 823,478 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   ^from_station_name, ^from_station_id

(X202208_divvy_tripdata <- rename(X202208_divvy_tripdata
  ,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: 785,932 x 13
##   trip_id      bikeid start_time      end_time      from_...^1 f
rom_...^2
##   <chr>      <chr>   <dtm>      <dtm>      <chr>   <
chr>
## 1 550CF7EFEAE0C... elect... 2022-08-07 21:34:15 2022-08-07 21:41:46 <NA>   <
NA>
## 2 DAD198F405F9C... elect... 2022-08-08 14:39:21 2022-08-08 14:53:23 <NA>   <
NA>
## 3 E6F2BC47B65CB... elect... 2022-08-08 15:29:50 2022-08-08 15:40:34 <NA>   <
NA>
## 4 F597830181C2E... elect... 2022-08-08 02:43:50 2022-08-08 02:58:53 <NA>   <
NA>
## 5 0CE689BB4E313... elect... 2022-08-07 20:24:06 2022-08-07 20:29:58 <NA>   <
NA>
## 6 BFA7E7CC69860... elect... 2022-08-08 13:06:08 2022-08-08 13:19:09 <NA>   <
NA>
## 7 68C474A4E92F2... elect... 2022-08-08 14:02:40 2022-08-08 14:11:36 <NA>   <
NA>
## 8 14A985A3838AA... elect... 2022-08-07 20:56:17 2022-08-07 21:14:14 <NA>   <
NA>
## 9 E724B94BCE2E7... elect... 2022-08-07 21:30:05 2022-08-07 21:41:28 <NA>   <
NA>
## 10 1AA3756A6F818... elect... 2022-08-07 23:53:05 2022-08-08 00:04:14 <NA>   <

```

```

NA>
## # ... with 785,922 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #     ^1from_station_name, ^2from_station_id

(X202209_divvy_tripdata <- rename(X202209_divvy_tripdata
  ,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: 701,339 x 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr> <dtm>      <dtm>      <chr> <
chr>
## 1 5156990AC19CA... elect... 2022-09-01 08:36:22 2022-09-01 08:39:05 <NA> <
NA>
## 2 E12D4A16BF51C... elect... 2022-09-01 17:11:29 2022-09-01 17:14:45 <NA> <
NA>
## 3 A02B53CD7DB72... elect... 2022-09-01 17:15:50 2022-09-01 17:16:12 <NA> <
NA>
## 4 C82E05FEE872D... elect... 2022-09-01 09:00:28 2022-09-01 09:10:32 <NA> <
NA>
## 5 4DEEB4550A266... elect... 2022-09-01 07:30:11 2022-09-01 07:32:36 <NA> <
NA>
## 6 B1721F8C7C3AC... elect... 2022-09-01 12:04:25 2022-09-01 12:21:08 <NA> <
NA>
## 7 626FFDF9B539B... elect... 2022-09-01 12:05:51 2022-09-01 12:13:09 <NA> <
NA>
## 8 87DD84E603661... elect... 2022-09-01 06:24:57 2022-09-01 06:31:14 <NA> <
NA>
## 9 93EF429B574F2... elect... 2022-09-01 16:29:28 2022-09-01 16:40:51 <NA> <
NA>
## 10 3D318B7D137DD... elect... 2022-09-01 17:37:35 2022-09-01 17:52:46 <NA> <
NA>
## # ... with 701,329 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #     ^1from_station_name, ^2from_station_id

(X202210_divvy_tripdata <- rename(X202210_divvy_tripdata
  ,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: 558,685 × 13
##   trip_id      bikeid start_time      end_time      from_...1 f
rom_...2
##   <chr>      <chr> <dtm>      <dtm>      <chr>    <
chr>
##   1 A50255C1E1794... class... 2022-10-14 17:13:30 2022-10-14 17:19:39 Noble ... 1
3290
##   2 DB692A70BD2DD... elect... 2022-10-01 16:29:26 2022-10-01 16:49:06 Damen ... 1
3288
##   3 3C02727AAF60F... elect... 2022-10-19 18:55:40 2022-10-19 19:03:30 Hoyne ... 6
55
##   4 47E653FDC2D99... elect... 2022-10-31 07:52:36 2022-10-31 07:58:49 Rush S... K
A1504...
##   5 8B5407BE53515... class... 2022-10-13 18:41:03 2022-10-13 19:26:18 900 W ... 1
3028
##   6 A177C92E9F021... elect... 2022-10-13 15:53:27 2022-10-13 15:59:17 900 W ... 1
3028
##   7 DF5EC7678DE3C... elect... 2022-10-06 15:51:21 2022-10-06 15:55:06 900 W ... 1
3028
##   8 407DE6D80130A... class... 2022-10-26 17:30:10 2022-10-26 17:37:57 Rush S... K
A1504...
##   9 45EEAF68A1A05... class... 2022-10-22 09:47:56 2022-10-22 09:57:42 Noble ... 1
3290
##  10 66CD8E4D0C38C... elect... 2022-10-24 12:39:47 2022-10-24 12:48:36 Noble ... 1
3290
## # ... with 558,675 more rows, 7 more variables: to_station_name <chr>,
## #   to_station_id <chr>, start_lat <dbl>, start_lng <dbl>, end_lat <dbl>,
## #   end_lng <dbl>, usertype <chr>, and abbreviated variable names
## #   1from_station_name, 2from_station_id

# Combine all files into one table (dataframe)
all_trips <- bind_rows(X202111_divvy_tripdata
, X202112_divvy_tripdata
, X202201_divvy_tripdata
, X202202_divvy_tripdata
, X202203_divvy_tripdata
, X202204_divvy_tripdata
, X202205_divvy_tripdata
, X202206_divvy_tripdata
, X202207_divvy_tripdata
, X202208_divvy_tripdata
, X202209_divvy_tripdata

```

```

,X202210_divvy_tripdata)

# Check the column names on the combined spreadsheet
colnames(all_trips)

## [1] "trip_id"          "bikeid"           "start_time"
## [4] "end_time"         "from_station_name" "from_station_id"
## [7] "to_station_name"  "to_station_id"    "start_lat"
## [10] "start_lng"        "end_lat"          "end_lng"
## [13] "usertype"

# Get rid of "start_lat", "start_lng", "end_lat", "end_lng" as columns
all_trips = subset(all_trips, select = -c(start_lat,start_lng,end_lat,end_lng))

# In the "member_casual" column, replace "Subscriber" with "member" and "Customer" with "casual"
all_trips <- all_trips %>%
  mutate(member_casual = recode(usertype
                                , "Subscriber" = "member"
                                , "Customer" = "casual"))

# Begin by seeing how many observations fall under each usertype
table(all_trips$member_casual)

##
## casual member
## 2353033 3402661

# Add columns that 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
# The default format is yyyy-mm-dd
all_trips$date <- as.Date(all_trips$start_time, format = "%Y-%m/%-d %H:%M:%S"
, tz = Sys.timezone())
all_trips$month <- format(as.Date(all_trips$date), format = "%m")
all_trips$day <- format(as.Date(all_trips$date), format = "%d")
all_trips$year <- format(as.Date(all_trips$date), format = "%Y")
all_trips$day_of_week <- format(as.Date(all_trips$date), format = "%A")
all_trips$time <- format(as.POSIXct(all_trips$start_time), format = "%H:%M:%S")

#character types
str(all_trips)

## tibble [5,755,694 × 16] (S3: tbl_df/tbl/data.frame)
## $ trip_id      : chr [1:5755694] "7C00A93E10556E47" "90854840DFD508BA"
##               "0A7D10CDD144061C" "2F3BE33085BCFF02" ...
## $ bikeid       : chr [1:5755694] "electric_bike" "electric_bike" "electric_bike"
##               "electric_bike" ...
## $ start_time   : POSIXct[1:5755694], format: "2021-11-27 13:27:38" "2

```



```

021-11-27 13:38:25" ...
## $ end_time      : POSIXct[1:5755694], format: "2021-11-27 13:46:38" "2
021-11-27 13:56:10" ...
## $ from_station_name: chr [1:5755694] NA NA NA NA ...
## $ from_station_id  : chr [1:5755694] NA NA NA NA ...
## $ to_station_name  : chr [1:5755694] NA NA NA NA ...
## $ to_station_id    : chr [1:5755694] NA NA NA NA ...
## $ usertype         : chr [1:5755694] "casual" "casual" "casual" "casual"
...
## $ member_casual    : chr [1:5755694] "casual" "casual" "casual" "casual"
...
## $ date             : Date[1:5755694], format: "2021-11-27" "2021-11-27" .
..
## $ month            : chr [1:5755694] "11" "11" "11" "11" ...
## $ day              : chr [1:5755694] "27" "27" "26" "27" ...
## $ year             : chr [1:5755694] "2021" "2021" "2021" "2021" ...
## $ day_of_week      : chr [1:5755694] "Saturday" "Saturday" "Friday" "Satu
rday" ...
## $ time             : chr [1:5755694] "13:27:38" "13:38:25" "22:03:34" "09
:56:49" ...

# Add a "ride_length" calculation to all_trips (in seconds)
all_trips$ride_length_seconds <- difftime(all_trips$end_time,all_trips$start_
time)
all_trips$ride_length_minutes <- difftime(all_trips$end_time,all_trips$start_
time, units = c("mins"))

#Check format of data
is.factor(all_trips$ride_length_seconds)

## [1] FALSE

is.factor(all_trips$ride_length_minutes)

## [1] FALSE

is.factor(all_trips$time)

## [1] FALSE

# Convert "ride_length" from Factor to numeric so we can run calculations on
the data
all_trips$ride_length_minutes <- as.numeric(as.character(all_trips$ride_lengt
h_minutes))
all_trips$ride_length_seconds <- as.numeric(as.character(all_trips$ride_lengt
h_seconds))
all_trips$time <- as_hms(as.character(all_trips$time))
is.numeric(all_trips$ride_length_minutes)

## [1] TRUE

is.numeric(all_trips$ride_length_seconds)

```

```
## [1] TRUE

is.numeric(all_trips$time)

## [1] FALSE

# check if the columns have converted properly
str(all_trips)

## tibble [5,755,694 × 18] (S3: tbl_df/tbl/data.frame)
## $ trip_id      : chr [1:5755694] "7C00A93E10556E47" "90854840DFD508
##                BA" "0A7D10CDD144061C" "2F3BE33085BCFF02" ...
## $ bikeid       : chr [1:5755694] "electric_bike" "electric_bike" "e
##                lectric_bike" "electric_bike" ...
## $ start_time    : POSIXct[1:5755694], format: "2021-11-27 13:27:38"
##                "2021-11-27 13:38:25" ...
## $ end_time      : POSIXct[1:5755694], format: "2021-11-27 13:46:38"
##                "2021-11-27 13:56:10" ...
## $ from_station_name : chr [1:5755694] NA NA NA NA ...
## $ from_station_id  : chr [1:5755694] NA NA NA NA ...
## $ to_station_name  : chr [1:5755694] NA NA NA NA ...
## $ to_station_id    : chr [1:5755694] NA NA NA NA ...
## $ usertype        : chr [1:5755694] "casual" "casual" "casual" "casual
##                " ...
## $ member_casual    : chr [1:5755694] "casual" "casual" "casual" "casual
##                " ...
## $ date            : Date[1:5755694], format: "2021-11-27" "2021-11-27"
##                ...
## $ month           : chr [1:5755694] "11" "11" "11" "11" ...
## $ day             : chr [1:5755694] "27" "27" "26" "27" ...
## $ year            : chr [1:5755694] "2021" "2021" "2021" "2021" ...
## $ day_of_week      : chr [1:5755694] "Saturday" "Saturday" "Friday" "Sa
##                turday" ...
## $ time            : 'hms' num [1:5755694] 13:27:38 13:38:25 22:03:34 0
##                9:56:49 ...
## ..- attr(*, "units")= chr "secs"
## $ ride_length_seconds: num [1:5755694] 1140 1065 142 301 1273 ...
## $ ride_length_minutes: num [1:5755694] 19 17.75 2.37 5.02 21.22 ...

#Round the minutes
round(all_trips$ride_length_minutes, digits = 2)

## [13225]    37.88    10.63    11.53    11.60     8.92    10.08    131.80
##                1.97
## [13233]     2.38     1.25     1.67     2.17     1.77     1.85     9.60
##                5.57
## [13241]     8.25     2.43     8.55     9.55     9.87     3.93     9.47
##                10.00
## [ reached getOption("max.print") -- omitted 5655695 entries ]
```

We will create a new version of the dataframe (v2) since data (trips<0) is being removed

```
all_trips_v2 <- all_trips[!all_trips$ride_length_seconds<0,]
```

check to see if any rows of zeros came through

```
all_trips_v2[all_trips_v2$ride_length_seconds<0,]
```

```
## # A tibble: 0 × 18
```

```
## # ... with 18 variables: trip_id <chr>, bikeid <chr>, start_time <dtm>,  
## #   end_time <dtm>, from_station_name <chr>, from_station_id <chr>,  
## #   to_station_name <chr>, to_station_id <chr>, usertype <chr>,  
## #   member_casual <chr>, date <date>, month <chr>, day <chr>, year <chr>,  
## #   day_of_week <chr>, time <time>, ride_length_seconds <dbl>,  
## #   ride_length_minutes <dbl>
```

#check for any rides that were 0 seconds long and delete them from the dataframe

```
all_trips_v2[all_trips_v2$ride_length_seconds==0,]
```

```
## # A tibble: 454 × 18
```

```
##   trip_id      bikeid start_time      end_time      from_...1 f  
rom_...2  
##   <chr>      <chr>  <dtm>      <dtm>      <chr>  <  
chr>
```

```
## 1 9613FC33CDDD1... elect... 2021-11-15 20:37:25 2021-11-15 20:37:25 Univer... 6  
05
```

```
## 2 9C86955118707... elect... 2021-11-03 21:10:06 2021-11-03 21:10:06 Kimbal... K  
A1504...
```

```
## 3 8A59BB1E1F124... elect... 2021-11-07 14:49:48 2021-11-07 14:49:48 Cornel... K  
A1503...
```

```
## 4 06BA797C3F934... elect... 2021-11-03 17:14:29 2021-11-03 17:14:29 Ellis ... T  
A1309...
```

```
## 5 739013621E83C... elect... 2021-11-09 14:20:18 2021-11-09 14:20:18 Shedd ... 1  
5544
```

```
## 6 C12A812CA5389... elect... 2021-11-10 18:02:37 2021-11-10 18:02:37 Univer... K  
A1503...
```

```
## 7 C52B460E50078... elect... 2021-11-24 23:26:40 2021-11-24 23:26:40 Halste... T  
A1309...
```

```
## 8 64C6EC51B68F3... elect... 2021-11-13 07:22:51 2021-11-13 07:22:51 Delano... K  
A1706...
```

```
## 9 1524939F06EFF... elect... 2021-11-03 12:41:45 2021-11-03 12:41:45 Univer... K  
A1503...
```

```
## 10 EAD833279D227... elect... 2021-11-25 16:39:26 2021-11-25 16:39:26 Univer... K  
A1503...
```

```
## # ... with 444 more rows, 12 more variables: to_station_name <chr>,  
## #   to_station_id <chr>, usertype <chr>, member_casual <chr>, date <date>,  
## #   month <chr>, day <chr>, year <chr>, day_of_week <chr>, time <time>,  
## #   ride_length_seconds <dbl>, ride_length_minutes <dbl>, and abbreviated  
## #   variable names 1from_station_name, 2from_station_id
```

```

all_trips_v2 <- all_trips_v2[!all_trips_v2$ride_length_seconds==0,]

#check your new dataframe
head(all_trips_v2)

## # A tibble: 6 × 18
##   trip_id bikeid start_time      end_time      from_...1 from_...2 t
##   <chr>   <chr>   <dtm>          <dtm>          <chr>   <chr>   <
##   <chr>
## 1 7C00A9... elect... 2021-11-27 13:27:38 2021-11-27 13:46:38 <NA>   <NA>   <
##   NA>
## 2 908548... elect... 2021-11-27 13:38:25 2021-11-27 13:56:10 <NA>   <NA>   <
##   NA>
## 3 0A7D10... elect... 2021-11-26 22:03:34 2021-11-26 22:05:56 <NA>   <NA>   <
##   NA>
## 4 2F3BE3... elect... 2021-11-27 09:56:49 2021-11-27 10:01:50 <NA>   <NA>   <
##   NA>
## 5 D67B47... elect... 2021-11-26 19:09:28 2021-11-26 19:30:41 <NA>   <NA>   <
##   NA>
## 6 02F85C... elect... 2021-11-26 18:34:07 2021-11-26 18:52:49 Michig... 13042   <
##   NA>
## # ... with 11 more variables: to_station_id <chr>, usertype <chr>,
## #   member_casual <chr>, date <date>, month <chr>, day <chr>, year <chr>,
## #   day_of_week <chr>, time <time>, ride_length_seconds <dbl>,
## #   ride_length_minutes <dbl>, and abbreviated variable names
## #   1from_station_name, 2from_station_id, 3to_station_name

str(all_trips_v2)

## tibble [5,755,128 × 18] (S3: tbl_df/tbl/data.frame)
##  $ trip_id      : chr [1:5755128] "7C00A93E10556E47" "90854840DFD508
##   BA" "0A7D10CDD144061C" "2F3BE33085BCFF02" ...
##  $ bikeid       : chr [1:5755128] "electric_bike" "electric_bike" "e
##   lectric_bike" "electric_bike" ...
##  $ start_time   : POSIXct[1:5755128], format: "2021-11-27 13:27:38"
##   "2021-11-27 13:38:25" ...
##  $ end_time     : POSIXct[1:5755128], format: "2021-11-27 13:46:38"
##   "2021-11-27 13:56:10" ...
##  $ from_station_name : chr [1:5755128] NA NA NA NA ...
##  $ from_station_id  : chr [1:5755128] NA NA NA NA ...
##  $ to_station_name  : chr [1:5755128] NA NA NA NA ...
##  $ to_station_id    : chr [1:5755128] NA NA NA NA ...
##  $ usertype       : chr [1:5755128] "casual" "casual" "casual" "casual
##   " ...
##  $ member_casual   : chr [1:5755128] "casual" "casual" "casual" "casual
##   " ...
##  $ date          : Date[1:5755128], format: "2021-11-27" "2021-11-27"
##   ...
##  $ month         : chr [1:5755128] "11" "11" "11" "11" ...

```

```
## $ day          : chr [1:5755128] "27" "27" "26" "27" ...
## $ year         : chr [1:5755128] "2021" "2021" "2021" "2021" ...
## $ day_of_week  : chr [1:5755128] "Saturday" "Saturday" "Friday" "Sa
turday" ...
## $ time         : 'hms' num [1:5755128] 13:27:38 13:38:25 22:03:34 0
9:56:49 ...
## ..- attr(*, "units")= chr "secs"
## $ ride_length_seconds: num [1:5755128] 1140 1065 142 301 1273 ...
## $ ride_length_minutes: num [1:5755128] 19 17.75 2.37 5.02 21.22 ...
```

```
summary(all_trips_v2)
```

```
##   trip_id          bikeid          start_time
## Length:5755128      Length:5755128      Min.   :2021-11-01 00:00:14.00
## Class :character    Class :character    1st Qu.:2022-04-27 16:41:34.50
## Mode  :character    Mode  :character    Median :2022-06-30 18:31:01.00
##                                     Mean   :2022-06-13 23:06:03.89
##                                     3rd Qu.:2022-08-24 19:51:25.75
##                                     Max.   :2022-10-31 23:59:33.00
##   end_time          from_station_name from_station_id
## Min.   :2021-11-01 00:04:06.00      Length:5755128      Length:5755128
## 1st Qu.:2022-04-27 16:52:32.50      Class :character    Class :character
## Median :2022-06-30 18:49:25.00      Mode  :character    Mode  :character
## Mean   :2022-06-13 23:25:30.53
## 3rd Qu.:2022-08-24 20:09:21.75
## Max.   :2022-11-07 04:53:58.00
## to_station_name    to_station_id      usertype      member_casual
## Length:5755128      Length:5755128      Length:5755128      Length:5755128
## Class :character    Class :character    Class :character    Class :character
## Mode  :character    Mode  :character    Mode  :character    Mode  :character
##
##
##   date          month          day          year
## Min.   :2021-10-31      Length:5755128      Length:5755128      Length:5755128
## 1st Qu.:2022-04-27      Class :character    Class :character    Class :character
## Median :2022-06-30      Mode  :character    Mode  :character    Mode  :character
## Mean   :2022-06-13
## 3rd Qu.:2022-08-24
## Max.   :2022-10-31
## day_of_week      time          ride_length_seconds ride_length_minu
tes
## Length:5755128      Length:5755128      Min.   :      1      Min.   :      0.02
## Class :character    Class1:hms          1st Qu.:      351      1st Qu.:      5.85
## Mode  :character    Class2:diffftime    Median :      621      Median :     10.35
##                                     Mode  :numeric      Mean   :     1167      Mean   :     19.44
##                                     3rd Qu.:     1115      3rd Qu.:     18.58
##                                     Max.   :    2483235      Max.   :    41387.25
```

```

# Descriptive analysis on ride_length (all figures in seconds)
mean(all_trips_v2$ride_length_minutes) #straight average (total ride length /
rides)

## [1] 19.44413

median(all_trips_v2$ride_length_minutes) #midpoint number in the ascending ar
ray of ride lengths

## [1] 10.35

max(all_trips_v2$ride_length_minutes) #Longest ride

## [1] 41387.25

min(all_trips_v2$ride_length_minutes) #shortest ride

## [1] 0.01666667

# See the average ride time by each day for members vs casual users
aggregate(all_trips_v2$ride_length_minutes ~ all_trips_v2$usertype + all_trip
s_v2$day_of_week, FUN = mean)

##      all_trips_v2$usertype all_trips_v2$day_of_week
## 1          casual      Friday
## 2          member      Friday
## 3          casual     Monday
## 4          member     Monday
## 5          casual    Saturday
## 6          member    Saturday
## 7          casual     Sunday
## 8          member     Sunday
## 9          casual   Thursday
## 10         member   Thursday
## 11         casual    Tuesday
## 12         member    Tuesday
## 13         casual   Wednesday
## 14         member   Wednesday
##      all_trips_v2$ride_length_minutes
## 1          28.21357
## 2          12.51635
## 3          29.07824
## 4          12.28279
## 5          32.69342
## 6          14.21156
## 7          33.85162
## 8          14.09128
## 9          25.78009
## 10         12.23618
## 11         26.18603
## 12         12.13604

```

```
## 13                24.66755
## 14                12.08513
```

```
aggregate(all_trips_v2$ride_length_minutes ~ all_trips_v2$usertype + all_trips_v2$month, FUN = mean)
```

```
##      all_trips_v2$usertype all_trips_v2$month all_trips_v2$ride_length_minut
es
## 1          casual         01                27.664
50
## 2          member         01                11.965
22
## 3          casual         02                26.690
83
## 4          member         02                11.417
74
## 5          casual         03                32.632
63
## 6          member         03                11.950
48
## 7          casual         04                29.503
74
## 8          member         04                11.502
40
## 9          casual         05                30.892
89
## 10         member         05                13.365
49
## 11         casual         06                32.119
48
## 12         member         06                13.997
67
## 13         casual         07                29.298
65
## 14         member         07                13.723
71
## 15         casual         08                29.256
96
## 16         member         08                13.382
43
## 17         casual         09                27.974
88
## 18         member         09                12.972
17
## 19         casual         10                26.412
58
## 20         member         10                11.962
00
## 21         casual         11                23.147
65
```

```
## 22          member          11          11.311
01
## 23          casual          12          24.253
44
## 24          member          12          11.017
44
```

Notice that the days of the week are out of order. Let's fix that.

```
all_trips_v2$day_of_week <- ordered(all_trips_v2$day_of_week, levels=c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"))
```

Now Lets run it again to check the order

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

```
##    all_trips_v2$usertype all_trips_v2$day_of_week
## 1          casual      Sunday
## 2          member      Sunday
## 3          casual     Monday
## 4          member     Monday
## 5          casual     Tuesday
## 6          member     Tuesday
## 7          casual   Wednesday
## 8          member   Wednesday
## 9          casual   Thursday
## 10         member   Thursday
## 11         casual    Friday
## 12         member    Friday
## 13         casual   Saturday
## 14         member   Saturday
```

```
##    all_trips_v2$ride_length_minutes
## 1          33.85162
## 2          14.09128
## 3          29.07824
## 4          12.28279
## 5          26.18603
## 6          12.13604
## 7          24.66755
## 8          12.08513
## 9          25.78009
## 10         12.23618
## 11         28.21357
## 12         12.51635
## 13         32.69342
## 14         14.21156
```

```
aggregate(all_trips_v2$ride_length_minutes ~ all_trips_v2$usertype + all_trips_v2$month, FUN = mean)
```

```
##    all_trips_v2$usertype all_trips_v2$month all_trips_v2$ride_length_minutes
```


## 1 50	casual	01	27.664
## 2 22	member	01	11.965
## 3 83	casual	02	26.690
## 4 74	member	02	11.417
## 5 63	casual	03	32.632
## 6 48	member	03	11.950
## 7 74	casual	04	29.503
## 8 40	member	04	11.502
## 9 89	casual	05	30.892
## 10 49	member	05	13.365
## 11 48	casual	06	32.119
## 12 67	member	06	13.997
## 13 65	casual	07	29.298
## 14 71	member	07	13.723
## 15 96	casual	08	29.256
## 16 43	member	08	13.382
## 17 88	casual	09	27.974
## 18 17	member	09	12.972
## 19 58	casual	10	26.412
## 20 00	member	10	11.962
## 21 65	casual	11	23.147
## 22 01	member	11	11.311
## 23 44	casual	12	24.253
## 24 44	member	12	11.017

Analyze ridership data by type and weekday and month separately

```
all_trips_v2 %>%
  mutate(weekday = wday(start_time, label = TRUE)) %>%
  group_by(usertype, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length_minutes), time = format(mean
  (strptime(time, "%H:%M:%S")), "%H:%M:%S")) %>%
  arrange(usertype, weekday)
```

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

```
## # A tibble: 14 × 5
## # Groups:   usertype [2]
##   usertype weekday number_of_rides average_duration time
##   <chr>      <ord>          <int>          <dbl> <chr>
## 1 casual    Sun             397020          33.9 13:55:14
## 2 casual    Mon             284944          29.3 15:07:00
## 3 casual    Tue             264360          26.0 15:19:51
## 4 casual    Wed             275374          24.9 15:34:38
## 5 casual    Thu             306916          25.5 15:37:26
## 6 casual    Fri             338918          27.9 15:26:10
## 7 casual    Sat             485217          32.6 14:49:29
## 8 member    Sun             395795          14.1 14:07:24
## 9 member    Mon             489653          12.3 14:34:42
## 10 member   Tue             523669          12.1 14:27:20
## 11 member   Wed             529625          12.1 14:35:00
## 12 member   Thu             532623          12.2 14:37:47
## 13 member   Fri             476662          12.5 14:35:29
## 14 member   Sat             454352          14.2 14:25:16
```

```
all_trips_v2 %>%
  mutate(month = month(start_time, label = TRUE)) %>%
  group_by(usertype, month) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length_minutes), time = format(mean
  (strptime(time, "%H:%M:%S")), "%H:%M:%S")) %>%
  arrange(usertype, month) %>%
  print(n = 24)
```

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

```
## # A tibble: 24 × 5
## # Groups:   usertype [2]
##   usertype month number_of_rides average_duration time
##   <chr>      <ord>          <int>          <dbl> <chr>
## 1 casual    Jan             18517          30.4 14:10:18
## 2 casual    Feb             21414          26.7 14:39:09
## 3 casual    Mar             89874          32.6 14:59:31
## 4 casual    Apr             126398          29.5 14:58:21
```

```
## 5 casual May 280387 30.9 15:11:13
## 6 casual Jun 369022 32.1 15:22:49
## 7 casual Jul 406013 29.3 15:14:55
## 8 casual Aug 358886 29.3 15:07:36
## 9 casual Sep 296664 28.0 14:56:03
## 10 casual Oct 208961 26.4 14:30:03
## 11 casual Nov 106884 23.1 14:33:41
## 12 casual Dec 69729 23.5 14:29:01
## 13 member Jan 85248 12.0 14:05:38
## 14 member Feb 94190 11.4 14:04:39
## 15 member Mar 194150 12.0 14:16:04
## 16 member Apr 244820 11.5 14:20:05
## 17 member May 354423 13.4 14:36:57
## 18 member Jun 400116 14.0 14:45:30
## 19 member Jul 417403 13.7 14:45:27
## 20 member Aug 426969 13.4 14:40:34
## 21 member Sep 404603 13.0 14:32:52
## 22 member Oct 349659 12.0 14:22:37
## 23 member Nov 253008 11.3 14:12:00
## 24 member Dec 177790 11.0 14:02:47
```

#Figure out Top most visited starting stations

```
Top_20_Station <- all_trips_v2 %>%
  group_by(usertype, from_station_name) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length_minutes)) %
>%
  arrange(usertype, from_station_name)
```

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

#Remove missing values from your data

```
Top_20_Station <- Top_20_Station[complete.cases(Top_20_Station$from_station_name),]
```

#Put the table in order of most number of rides

```
Top_20_Stations <- Top_20_Station[order(Top_20_Station$usertype, -Top_20_Station$number_of_rides),]
```

Analyze ridership data by type and weekday and month combined

```
all_trips_v2 %>%
  mutate(month = month(start_time, label = TRUE)) %>%
  mutate(weekday = wday(start_time, label = TRUE)) %>%
  group_by(usertype, month, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration_minutes = mean(ride_length_minutes), time_of_day = format(mean(strptime(time, "%H:%M:%S")), "%H:%M:%S")) %>%
```

```

arrange(usertype, month, weekday) %>%
print(n = 140)

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

## # A tibble: 168 × 6
## # Groups:   usertype, month [24]
##   usertype month weekday number_of_rides average_duration_minutes time_o
f_day
##   <chr>      <ord> <ord>          <int>          <dbl> <chr>
## 1 casual    Jan   Sun            2515          26.6 13:41:
46
## 2 casual    Jan   Mon            2429          28.1 14:45:
33
## 3 casual    Jan   Tue            2393          19.4 15:12:
26
## 4 casual    Jan   Wed            2389          36.2 14:34:
03
## 5 casual    Jan   Thu            2543          35.4 14:56:
51
## 6 casual    Jan   Fri            2457          24.6 14:42:
34
## 7 casual    Jan   Sat            3791          38.0 12:20:
18
## 8 casual    Feb   Sun            4206          33.0 14:36:
04
## 9 casual    Feb   Mon            4405          24.9 15:14:
19
## 10 casual   Feb   Tue            2787          26.9 14:36:
21
## 11 casual   Feb   Wed            2623          23.1 14:22:
06
## 12 casual   Feb   Thu            1878          27.5 13:49:
24
## 13 casual   Feb   Fri            2698          22.1 15:12:
48
## 14 casual   Feb   Sat            2817          27.2 14:08:
23
## 15 casual   Mar   Sun           16575          38.8 14:34:
40
## 16 casual   Mar   Mon          14448          35.4 15:43:
43
## 17 casual   Mar   Tue          10152          25.0 14:57:
03
## 18 casual   Mar   Wed          14540          30.1 15:31:
14
## 19 casual   Mar   Thu          12023          29.9 14:44:
50

```

## 01	20	casual	Mar	Fri	7156	25.7 14:22:
## 56	21	casual	Mar	Sat	14980	36.3 14:44:
## 11	22	casual	Apr	Sun	19384	33.0 13:54:
## 12	23	casual	Apr	Mon	12059	29.1 15:03:
## 18	24	casual	Apr	Tue	14547	26.3 15:22:
## 14	25	casual	Apr	Wed	10457	21.3 14:33:
## 32	26	casual	Apr	Thu	16777	25.9 15:46:
## 43	27	casual	Apr	Fri	16848	26.1 15:00:
## 16	28	casual	Apr	Sat	36326	34.8 15:05:
## 53	29	casual	May	Sun	55317	33.7 14:04:
## 33	30	casual	May	Mon	47465	32.4 15:10:
## 12	31	casual	May	Tue	35065	26.7 15:26:
## 46	32	casual	May	Wed	24060	25.8 15:50:
## 32	33	casual	May	Thu	33396	29.3 15:58:
## 59	34	casual	May	Fri	32237	29.5 15:40:
## 14	35	casual	May	Sat	52847	33.4 15:05:
## 51	36	casual	Jun	Sun	65847	36.1 14:20:
## 11	37	casual	Jun	Mon	37001	31.5 15:07:
## 08	38	casual	Jun	Tue	38818	31.5 15:46:
## 37	39	casual	Jun	Wed	48385	29.1 16:02:
## 51	40	casual	Jun	Thu	57972	30.2 15:59:
## 54	41	casual	Jun	Fri	55867	32.6 15:48:
## 33	42	casual	Jun	Sat	65132	32.2 14:55:
## 46	43	casual	Jul	Sun	78246	33.6 14:23:
## 39	44	casual	Jul	Mon	43969	31.1 15:09:

## 35	45	casual	Jul	Tue	41442	26.4 15:32:
## 35	46	casual	Jul	Wed	42850	24.0 15:52:
## 00	47	casual	Jul	Thu	47787	24.3 15:49:
## 23	48	casual	Jul	Fri	56497	26.1 15:31:
## 50	49	casual	Jul	Sat	95222	32.9 15:07:
## 40	50	casual	Aug	Sun	48151	33.7 13:29:
## 58	51	casual	Aug	Mon	42358	28.9 15:18:
## 23	52	casual	Aug	Tue	51503	28.2 15:35:
## 01	53	casual	Aug	Wed	51486	25.6 15:36:
## 31	54	casual	Aug	Thu	42348	25.2 15:42:
## 18	55	casual	Aug	Fri	56859	30.2 15:31:
## 08	56	casual	Aug	Sat	66181	31.9 14:45:
## 16	57	casual	Sep	Sun	36253	34.7 13:29:
## 52	58	casual	Sep	Mon	31047	28.6 14:49:
## 46	59	casual	Sep	Tue	29584	21.2 15:08:
## 35	60	casual	Sep	Wed	33502	22.8 15:24:
## 58	61	casual	Sep	Thu	45833	21.2 15:31:
## 06	62	casual	Sep	Fri	56380	27.1 15:24:
## 58	63	casual	Sep	Sat	64065	35.4 14:36:
## 59	64	casual	Oct	Sun	44676	32.4 13:09:
## 13	65	casual	Oct	Mon	27233	22.2 14:54:
## 06	66	casual	Oct	Tue	15692	23.3 14:06:
## 50	67	casual	Oct	Wed	20590	20.1 15:21:
## 02	68	casual	Oct	Thu	22583	19.6 15:06:
## 57	69	casual	Oct	Fri	25974	25.3 15:23:

## 20	70 casual	Oct	Sat	52213	30.3 14:30:
## 47	71 casual	Nov	Sun	17414	28.8 13:19:
## 11	72 casual	Nov	Mon	14798	22.4 14:49:
## 30	73 casual	Nov	Tue	16138	17.6 14:52:
## 36	74 casual	Nov	Wed	13789	18.1 14:53:
## 50	75 casual	Nov	Thu	11191	20.5 14:39:
## 36	76 casual	Nov	Fri	12994	21.3 14:56:
## 08	77 casual	Nov	Sat	20560	29.2 14:39:
## 20	78 casual	Dec	Sun	8436	29.6 13:06:
## 20	79 casual	Dec	Mon	7732	23.9 14:51:
## 21	80 casual	Dec	Tue	6239	22.4 14:30:
## 18	81 casual	Dec	Wed	10703	23.2 14:58:
## 08	82 casual	Dec	Thu	12585	22.4 15:07:
## 35	83 casual	Dec	Fri	12951	21.9 14:28:
## 32	84 casual	Dec	Sat	11083	22.7 14:04:
## 43	85 member	Jan	Sun	8993	13.1 14:09:
## 52	86 member	Jan	Mon	13374	11.5 14:11:
## 58	87 member	Jan	Tue	13754	12.1 14:16:
## 32	88 member	Jan	Wed	12785	11.6 14:01:
## 43	89 member	Jan	Thu	14011	11.6 14:15:
## 57	90 member	Jan	Fri	11352	11.9 14:06:
## 05	91 member	Jan	Sat	10979	12.4 13:31:
## 45	92 member	Feb	Sun	11686	12.3 14:26:
## 03	93 member	Feb	Mon	18375	11.4 14:20:
## 18	94 member	Feb	Tue	16259	11.3 14:02:

## 95 member	Feb	Wed	14608	10.8 13:45:
52				
## 96 member	Feb	Thu	11633	11.1 13:20:
04				
## 97 member	Feb	Fri	11960	11.7 14:23:
53				
## 98 member	Feb	Sat	9669	11.6 14:10:
53				
## 99 member	Mar	Sun	22064	13.5 14:29:
34				
## 100 member	Mar	Mon	29447	12.6 14:46:
18				
## 101 member	Mar	Tue	34405	11.0 14:13:
47				
## 102 member	Mar	Wed	35956	12.0 14:09:
01				
## 103 member	Mar	Thu	32139	10.8 14:02:
25				
## 104 member	Mar	Fri	20492	11.1 13:54:
00				
## 105 member	Mar	Sat	19647	13.7 14:17:
54				
## 106 member	Apr	Sun	25456	12.4 14:14:
25				
## 107 member	Apr	Mon	33928	10.8 14:27:
46				
## 108 member	Apr	Tue	40431	10.8 14:24:
13				
## 109 member	Apr	Wed	32385	10.5 14:02:
32				
## 110 member	Apr	Thu	38594	11.4 14:34:
22				
## 111 member	Apr	Fri	35961	11.0 14:09:
04				
## 112 member	Apr	Sat	38065	13.6 14:23:
27				
## 113 member	May	Sun	48770	14.5 14:03:
15				
## 114 member	May	Mon	62059	13.4 14:40:
33				
## 115 member	May	Tue	59540	12.8 14:38:
40				
## 116 member	May	Wed	45095	12.3 14:47:
49				
## 117 member	May	Thu	51659	13.2 14:43:
39				
## 118 member	May	Fri	42307	12.8 14:49:
21				
## 119 member	May	Sat	44993	14.7 14:36:
03				


```

## 120 member Jun Sun 49057 15.7 14:19:
40
## 121 member Jun Mon 46785 13.4 14:36:
19
## 122 member Jun Tue 54980 13.6 14:47:
31
## 123 member Jun Wed 68791 13.1 14:56:
34
## 124 member Jun Thu 73495 13.7 15:00:
09
## 125 member Jun Fri 57319 13.8 14:57:
49
## 126 member Jun Sat 49689 15.2 14:26:
13
## 127 member Jul Sun 58778 15.2 14:24:
07
## 128 member Jul Mon 49847 13.4 14:44:
45
## 129 member Jul Tue 57516 12.9 14:46:
06
## 130 member Jul Wed 59608 12.9 14:56:
35
## 131 member Jul Thu 61151 13.0 14:59:
03
## 132 member Jul Fri 61641 13.0 14:46:
37
## 133 member Jul Sat 68862 15.3 14:40:
52
## 134 member Aug Sun 42968 14.6 13:59:
13
## 135 member Aug Mon 62598 12.5 14:50:
42
## 136 member Aug Tue 76711 13.1 14:44:
26
## 137 member Aug Wed 76613 13.0 14:45:
31
## 138 member Aug Thu 57504 13.0 14:52:
27
## 139 member Aug Fri 58697 13.6 14:49:
29
## 140 member Aug Sat 51878 14.7 14:26:
21
## # ... with 28 more rows

```

```

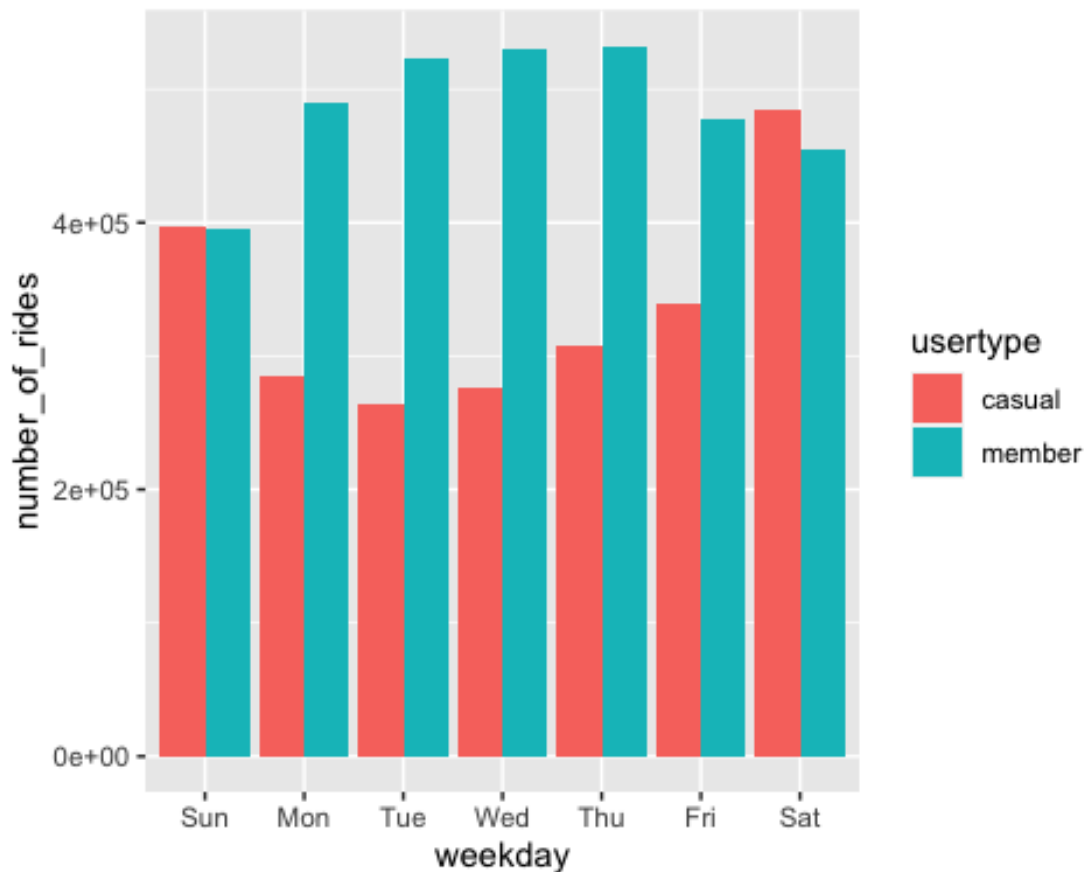
#creates weekday field using wday()
#groups by usertype and weekday
#calculates the number of rides and average duration
# calculates the average duration
# sorts

```

Let's visualize the number of rides by weekday and month

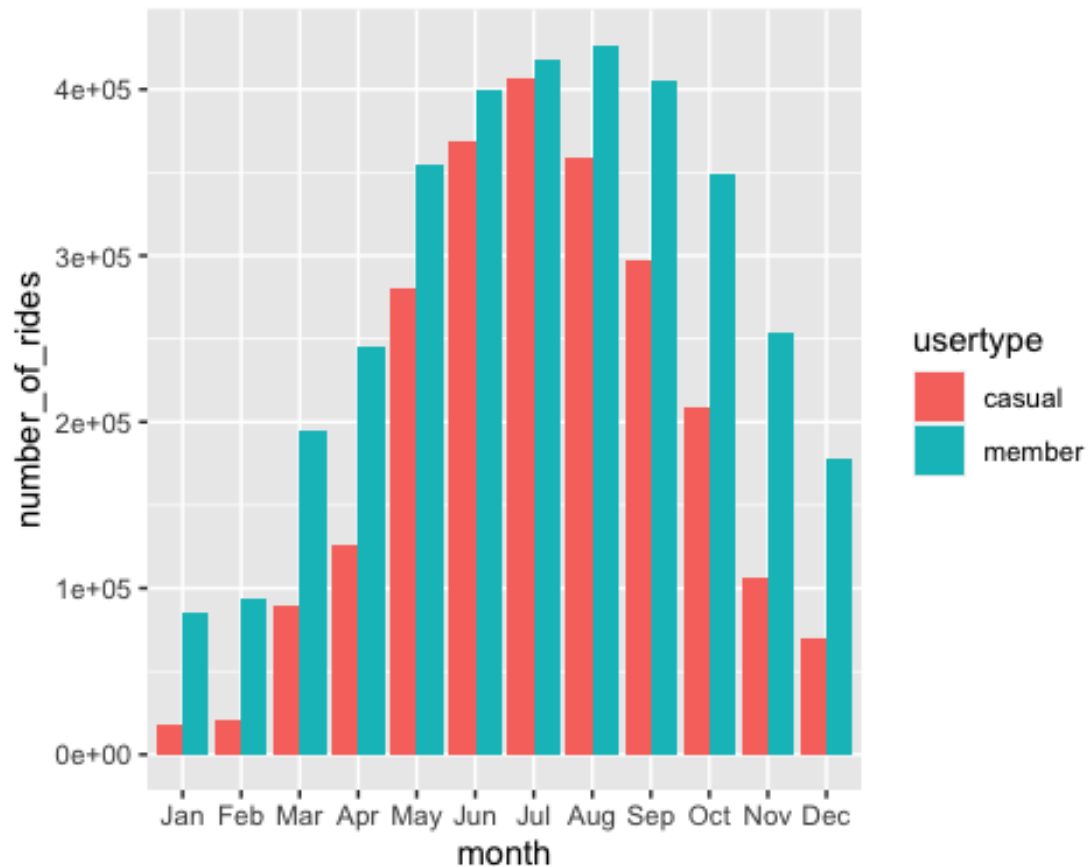
```
all_trips_v2 %>%  
  mutate(weekday = wday(start_time, label = TRUE)) %>%  
  group_by(usertype, weekday) %>%  
  summarise(number_of_rides = n()  
            , average_duration_minutes = mean(ride_length_minutes)) %>%  
  arrange(usertype, weekday) %>%  
  ggplot(aes(x = weekday, y = number_of_rides,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

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



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

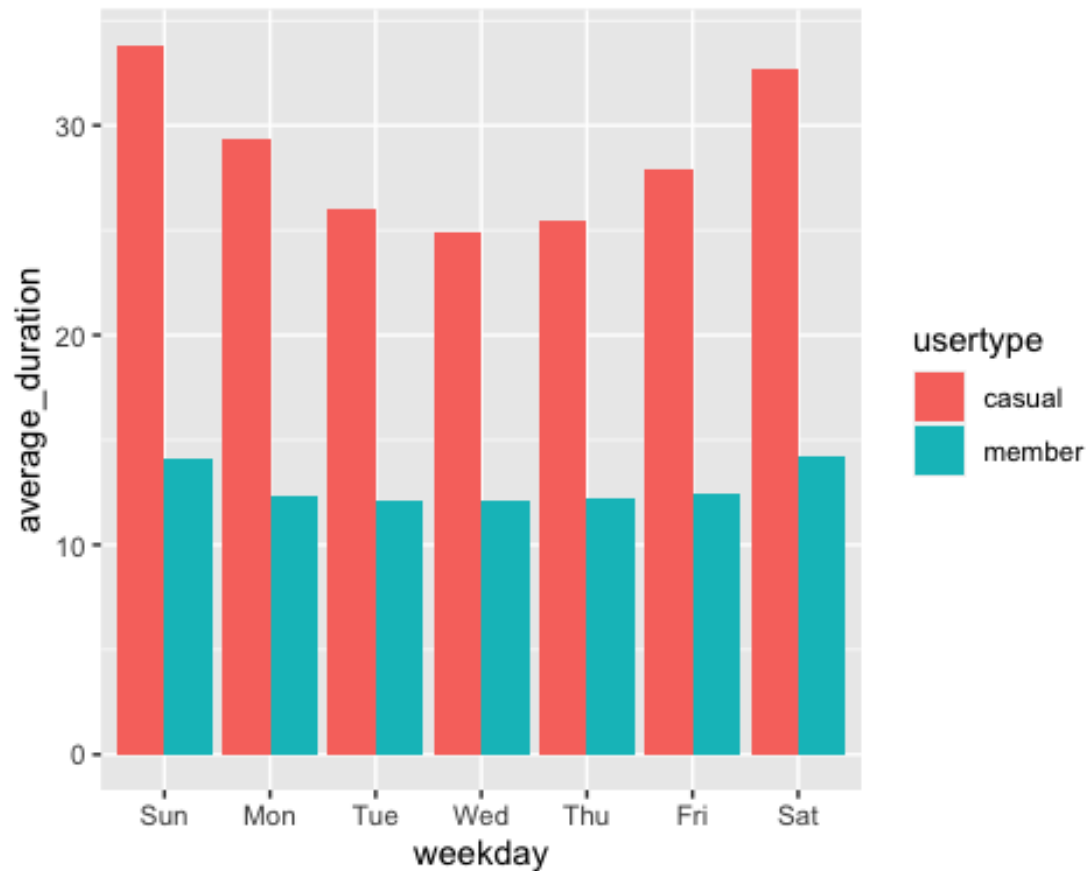
`summarise()` has grouped output by 'usertype'. You can override using the
`.groups` argument.



Let's create a visualization for average duration by week and by month

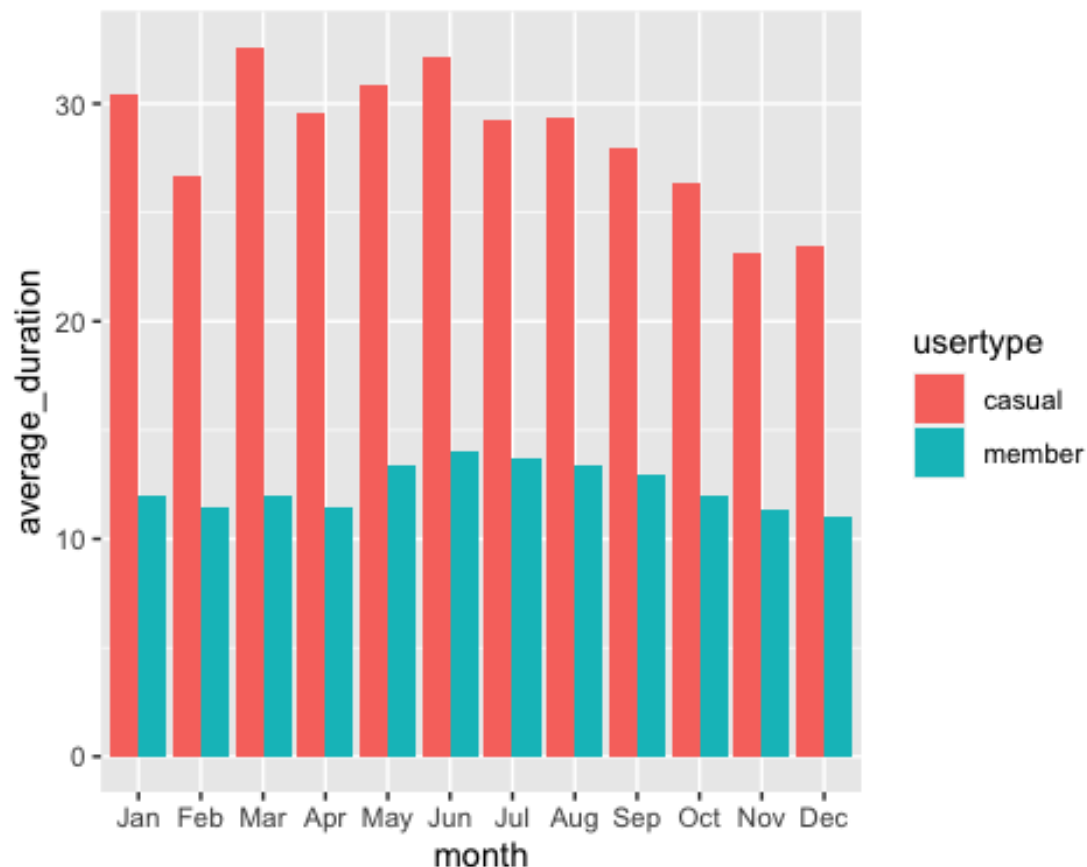
```
all_trips_v2 %>%
  mutate(weekday = wday(start_time, label = TRUE)) %>%
  group_by(usertype, weekday) %>%
  summarise(number_of_rides = n()
            , average_duration = mean(ride_length_minutes)) %>%
  arrange(usertype, weekday) %>%
  ggplot(aes(x = weekday, y = average_duration, fill = usertype))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

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



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

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



Create a csv file that we will visualize in Tableau

```
count_weekly <- aggregate(all_trips_v2$ride_length_minutes ~ all_trips_v2$user
type + all_trips_v2$day_of_week, FUN = mean)
```

```
count_monthly <- aggregate(all_trips_v2$ride_length_minutes ~ all_trips_v2$user
type + all_trips_v2$month, FUN = mean)
```

```
Number_of_rides_weekly <- all_trips_v2 %>%
  mutate(weekday = wday(start_time, label = TRUE)) %>%
  group_by(usertype, weekday) %>%
  summarise(number_of_rides = n())
```

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

```
Number_of_rides_monthly <- all_trips_v2 %>%
  mutate(month = month(start_time, label = TRUE)) %>%
  group_by(usertype, month) %>%
  summarise(number_of_rides = n())
```

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

```

all_trips_v3 <- all_trips_v2 %>%
  mutate(month = month(start_time, label = TRUE)) %>%
  mutate(weekday = wday(start_time, label = TRUE)) %>%
  group_by(usertype, month, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration_minutes = mean(ride_length_minutes), average_time_of_day = format(mean(strptime(time, "%H:%M:%S")), "%H:%M:%S")) %>%
  arrange(usertype, month, weekday)

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

#Export the CSV file
write.csv(count_weekly, file = '~/Desktop/Bikeshare project files/avg_ride_length_weekly.csv')
write.csv(count_monthly, file = '~/Desktop/Bikeshare project files/avg_ride_length_monthly.csv')
write.csv(Number_of_rides_weekly, file = '~/Desktop/Bikeshare project files/Number_of_rides_weekly.csv')
write.csv(Number_of_rides_monthly, file = '~/Desktop/Bikeshare project files/Number_of_rides_monthly.csv')
write.csv(all_trips_v3, file = '~/Desktop/Bikeshare project files/all_trips_v3.csv')
write.csv(Top_20_Stations, file = '~/Desktop/Bikeshare project files/Top_20_stations.csv')

```