

```
R version 4.2.0 (2022-04-22 ucrt) -- "Vigorous Calisthenics"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
```

```
Natural language support but running in an English locale
```

```
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
```

```
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

```
[Previously saved workspace restored]
```

```
> install.packages("tidyverse")
--- Please select a CRAN mirror for use in this session ---
Warning: unable to access index for repository https://cran.microsoft.com/src/contrib:
cannot open URL 'https://cran.microsoft.com/src/contrib/PACKAGES'
Warning: unable to access index for repository https://cran.microsoft.com/bin/windows/contrib/4.2
:
cannot open URL 'https://cran.microsoft.com/bin/windows/contrib/4.2/PACKAGES'
Warning message:
package 'tidyverse' is not available for this version of R
```

```
A version of this package for your version of R might be available elsewhere,
see the ideas at
```

```
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
> getwd()
[1] "C:/Users/richy/Documents"
> install.packages("tidyverse")
Warning: unable to access index for repository https://cran.microsoft.com/src/contrib:
cannot open URL 'https://cran.microsoft.com/src/contrib/PACKAGES'
Warning: unable to access index for repository https://cran.microsoft.com/bin/windows/contrib/4.2
:
cannot open URL 'https://cran.microsoft.com/bin/windows/contrib/4.2/PACKAGES'
Warning message:
package 'tidyverse' is not available for this version of R
```

```
A version of this package for your version of R might be available elsewhere,
see the ideas at
```

```
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
> install.packages("ggplot2")
Warning: unable to access index for repository https://cran.microsoft.com/src/contrib:
cannot open URL 'https://cran.microsoft.com/src/contrib/PACKAGES'
Warning: unable to access index for repository https://cran.microsoft.com/bin/windows/contrib/4.2
:
cannot open URL 'https://cran.microsoft.com/bin/windows/contrib/4.2/PACKAGES'
Warning message:
package 'ggplot2' is not available for this version of R
```

```
A version of this package for your version of R might be available elsewhere,
see the ideas at
```

```
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
```

```
> install.packages("dplyr")
Warning: unable to access index for repository https://cran.microsoft.com/src/contrib:
cannot open URL 'https://cran.microsoft.com/src/contrib/PACKAGES'
Warning: unable to access index for repository https://cran.microsoft.com/bin/windows/contrib/4.2
:
cannot open URL 'https://cran.microsoft.com/bin/windows/contrib/4.2/PACKAGES'
Warning message:
package 'dplyr' is not available for this version of R
```

```
A version of this package for your version of R might be available elsewhere,
```

```

see the ideas at
https://cran.r-project.org/doc/manuals/r-patched/R-admin.html#Installing-packages
>
> library(tidyverse)
— Attaching packages — tidyverse 1.3.1 —
✓ ggplot2 3.3.6      ✓ purrr 0.3.4
✓ tibble 3.1.7       ✓ dplyr 1.0.9
✓ tidyr 1.2.0        ✓ stringr 1.4.0
✓ readr 2.1.2        ✓ forcats 0.5.1
— Conflicts — tidyverse_conflicts() —
✗ dplyr::filter() masks stats::filter()
✗ dplyr::lag() masks stats::lag()
> library(ggplot2)
> library(dplyr)
> setwd("C:/Users/richy/Desktop/Google Capstone Project")
> getwd()
[1] "C:/Users/richy/Desktop/Google Capstone Project"
> Dec_2020 <- read_csv("202011-divvy-tripdata.csv")
[[1minindexing]] [[34m202011-divvy-tripdata.csv]] [[=====]] [[32m2.15GB/s]] [[0m, e
ta: [[36m 0s]] [[1minindexing]] [[34m202011-divvy-tripdata.csv]] [[=====]] [[32
m2.15GB/s]] [[0m, eta: [[36m 0s]] [[0m
Rows: 259716 Columns: 13
— Column specification —
Delimiter: ",",
chr (5): ride_id, rideable_type, start_station_name, end_station_name, m...
dbl (6): start_station_id, end_station_id, start_lat, start_lng, end_lat...
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.
>
> Jan_2021 <- read_csv("202012-divvy-tripdata.csv")
[[1minindexing]] [[34m202012-divvy-tripdata.csv]] [[=====]] [[32m2.15GB/s]] [[0m, e
ta: [[36m 0s]] [[0m
Rows: 131573 Columns: 13
— Column specification —
Delimiter: ",",
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
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.
> Feb_2021 <- read_csv("202101-divvy-tripdata.csv")
[[1minindexing]] [[34m202101-divvy-tripdata.csv]] [[=====]] [[32m2.15GB/s]] [[0m, e
ta: [[36m 0s]] [[0m
Rows: 96834 Columns: 13
— Column specification —
Delimiter: ",",
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
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.
> Mar_2021 <- read_csv("202102-divvy-tripdata.csv")
[[1minindexing]] [[34m202102-divvy-tripdata.csv]] [[=====]] [[32m2.15GB/s]] [[0m, e
ta: [[36m 0s]] [[0m
Rows: 49622 Columns: 13
— Column specification —
Delimiter: ",",
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
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.
> Apr_2021 <- read_csv("202103-divvy-tripdata.csv")
[[1minindexing]] [[34m202103-divvy-tripdata.csv]] [[=====]] [[32m2.15GB/s]] [[0m, e
ta: [[36m 0s]] [[1minindexing]] [[34m202103-divvy-tripdata.csv]] [[=====]] [[32

```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 228496 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
```

```
> May_2021 <- read_csv("202104-divvy-tripdata.csv")
```

```
[1mindexing [0m [34m202104-divvy-tripdata.csv [0m [=====] [32m2.15GB/s [0m, e
```

```
ta: [36m 0s [0m [1mindexing [0m [34m202104-divvy-tripdata.csv [0m [=====] [32
```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 337230 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
```

```
> Jun_2021 <- read_csv("202105-divvy-tripdata.csv")
```

```
[1mindexing [0m [34m202105-divvy-tripdata.csv [0m [=====] [32m2.15GB/s [0m, e
```

```
ta: [36m 0s [0m [1mindexing [0m [34m202105-divvy-tripdata.csv [0m [=====] [32
```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 531633 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
```

```
> Jul_2021 <- read_csv("202106-divvy-tripdata.csv")
```

```
[1mindexing [0m [34m202106-divvy-tripdata.csv [0m [=====] [32m2.15GB/s [0m, e
```

```
ta: [36m 0s [0m [1mindexing [0m [34m202106-divvy-tripdata.csv [0m [=====] [32
```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 729595 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
```

```
> Aug_2021 <- read_csv("202107-divvy-tripdata.csv")
```

```
[1mindexing [0m [34m202107-divvy-tripdata.csv [0m [=====] [32m2.15GB/s [0m, e
```

```
ta: [36m 0s [0m [1mindexing [0m [34m202107-divvy-tripdata.csv [0m [=====] [32
```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 822410 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
```

```
> Sep_2021 <- read_csv("202108-divvy-tripdata.csv")
```

```
[1mindexing [0m [34m202108-divvy-tripdata.csv [0m [=====] [32m2.15GB/s [0m, e
```

```
ta: [36m 0s [0m [1mindexing [0m [34m202108-divvy-tripdata.csv [0m [=====] [32
```

```
m2.15GB/s [0m, eta: [36m 0s [0m
```

```
Rows: 804352 Columns: 13
```

```
— Column specification —
```

```
Delimiter: ","
```

```
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
```

```
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.
> Oct_2021 <- read_csv("202109-divvy-tripdata.csv")
[[1minindexing[[0m [[34m202109-divvy-tripdata.csv[[0m [=====] [[32m2.15GB/s[[0m, e
ta: [[36m 0s[[0m[[1minindexing[[0m [[34m202109-divvy-tripdata.csv[[0m [=====] [[32
m2.15GB/s[[0m, eta: [[36m 0s[[0m

Rows: 756147 Columns: 13
```

```
— Column specification —————
Delimiter: ","
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
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.
> Nov_2021 <- read_csv("202110-divvy-tripdata.csv")
[[1minindexing[[0m [[34m202110-divvy-tripdata.csv[[0m [=====] [[32m2.15GB/s[[0m, e
ta: [[36m 0s[[0m[[1minindexing[[0m [[34m202110-divvy-tripdata.csv[[0m [=====] [[32
m2.15GB/s[[0m, eta: [[36m 0s[[0m

Rows: 631226 Columns: 13
```

```
— Column specification —————
Delimiter: ","
chr (7): ride_id, rideable_type, start_station_name, start_station_id, e...
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.
> # check to see all col names are named equally
```

```
> colnames(Dec_2020)
 [1] "ride_id"           "rideable_type"      "started_at"
 [4] "ended_at"          "start_station_name" "start_station_id"
 [7] "end_station_name"  "end_station_id"     "start_lat"
[10] "start_lng"         "end_lat"            "end_lng"
[13] "member_casual"
> colnames(Mar_2021)
 [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(Jun_2021)
 [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(Sep_2021)
 [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"
```

```
> Dec_2020 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> Jan_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> Feb_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> Mar_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> Apr_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> May_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
> Jun_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                             end_station_id = as.character(end_station_id))
```

```

> Jul_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                      end_station_id = as.character(end_station_id))
> Aug_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                      end_station_id = as.character(end_station_id))
> Sep_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                      end_station_id = as.character(end_station_id))
> Oct_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                      end_station_id = as.character(end_station_id))
> Nov_2021 <- mutate(Dec_2020, start_station_id = as.character(start_station_id),
+                      end_station_id = as.character(end_station_id))
> #Combining the data sets into Seasons
> Winter <- bind_rows(Dec_2020, Jan_2021, Feb_2021) #as q1
>
> Spring <- bind_rows(Mar_2021, Apr_2021, May_2021) #as q2
>
> Summer <- bind_rows(Jun_2021, Jul_2021, Aug_2021) #as q3
>
> Autumn <- bind_rows(Sep_2021, Oct_2021, Nov_2021) #as q4
> #Inspect the data
> str(Winter)
tibble [779,148 × 13] (S3: tbl_df/tbl/data.frame)
 $ ride_id      : chr [1:779148] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E
533E89C32080B9E" ...
 $ rideable_type : chr [1:779148] "electric_bike" "electric_bike" "electric_bike" "electric_b
ike" ...
 $ started_at    : POSIXct[1:779148], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" "20
20-11-01 00:34:05" "2020-11-01 00:45:16" ...
 $ ended_at      : POSIXct[1:779148], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" "20
20-11-01 01:03:06" "2020-11-01 00:54:31" ...
 $ start_station_name: chr [1:779148] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake S
hore Dr & Monroe St" "Leavitt St & Chicago Ave" ...
 $ start_station_id : chr [1:779148] "110" "672" "76" "659" ...
 $ end_station_name : chr [1:779148] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federa
l St & Polk St" "Stave St & Armitage Ave" ...
 $ end_station_id    : chr [1:779148] "211" "29" "41" "185" ...
 $ start_lat         : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
 $ start_lng         : num [1:779148] -87.6 -87.6 -87.6 -87.7 -87.6 ...
 $ end_lat           : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
 $ end_lng           : num [1:779148] -87.6 -87.7 -87.6 -87.7 -87.6 ...
 $ member_casual     : chr [1:779148] "casual" "casual" "casual" "casual" ...
> str(Spring)
tibble [779,148 × 13] (S3: tbl_df/tbl/data.frame)
 $ ride_id      : chr [1:779148] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E
533E89C32080B9E" ...
 $ rideable_type : chr [1:779148] "electric_bike" "electric_bike" "electric_bike" "electric_b
ike" ...
 $ started_at    : POSIXct[1:779148], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" "20
20-11-01 00:34:05" "2020-11-01 00:45:16" ...
 $ ended_at      : POSIXct[1:779148], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" "20
20-11-01 01:03:06" "2020-11-01 00:54:31" ...
 $ start_station_name: chr [1:779148] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake S
hore Dr & Monroe St" "Leavitt St & Chicago Ave" ...
 $ start_station_id : chr [1:779148] "110" "672" "76" "659" ...
 $ end_station_name : chr [1:779148] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federa
l St & Polk St" "Stave St & Armitage Ave" ...
 $ end_station_id    : chr [1:779148] "211" "29" "41" "185" ...
 $ start_lat         : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
 $ start_lng         : num [1:779148] -87.6 -87.6 -87.6 -87.7 -87.6 ...
 $ end_lat           : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
 $ end_lng           : num [1:779148] -87.6 -87.7 -87.6 -87.7 -87.6 ...
 $ member_casual     : chr [1:779148] "casual" "casual" "casual" "casual" ...
> str(Summer)
tibble [779,148 × 13] (S3: tbl_df/tbl/data.frame)
 $ ride_id      : chr [1:779148] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E
533E89C32080B9E" ...
 $ rideable_type : chr [1:779148] "electric_bike" "electric_bike" "electric_bike" "electric_b
ike" ...
 $ started_at    : POSIXct[1:779148], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" "20
20-11-01 00:34:05" "2020-11-01 00:45:16" ...
 $ ended_at      : POSIXct[1:779148], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" "20

```

```

20-11-01 01:03:06" "2020-11-01 00:54:31" ...
$ start_station_name: chr [1:779148] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake S
hore Dr & Monroe St" "Leavitt St & Chicago Ave" ...
$ start_station_id : chr [1:779148] "110" "672" "76" "659" ...
$ end_station_name : chr [1:779148] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federa
l St & Polk St" "Stave St & Armitage Ave" ...
$ end_station_id : chr [1:779148] "211" "29" "41" "185" ...
$ start_lat : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
$ start_lng : num [1:779148] -87.6 -87.6 -87.6 -87.7 -87.6 ...
$ end_lat : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
$ end_lng : num [1:779148] -87.6 -87.7 -87.6 -87.7 -87.6 ...
$ member_casual : chr [1:779148] "casual" "casual" "casual" "casual" ...
> str(Autumn)
tibble [779,148 × 13] (S3: tbl_df/tbl/data.frame)
$ ride_id : chr [1:779148] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E
533E89C32080B9E" ...
$ rideable_type : chr [1:779148] "electric_bike" "electric_bike" "electric_bike" "electric_b
ike" ...
$ started_at : POSIXct[1:779148], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" "20
20-11-01 00:34:05" "2020-11-01 00:45:16" ...
$ ended_at : POSIXct[1:779148], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" "20
20-11-01 01:03:06" "2020-11-01 00:54:31" ...
$ start_station_name: chr [1:779148] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake S
hore Dr & Monroe St" "Leavitt St & Chicago Ave" ...
$ start_station_id : chr [1:779148] "110" "672" "76" "659" ...
$ end_station_name : chr [1:779148] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federa
l St & Polk St" "Stave St & Armitage Ave" ...
$ end_station_id : chr [1:779148] "211" "29" "41" "185" ...
$ start_lat : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
$ start_lng : num [1:779148] -87.6 -87.6 -87.6 -87.7 -87.6 ...
$ end_lat : num [1:779148] 41.9 41.9 41.9 41.9 41.9 ...
$ end_lng : num [1:779148] -87.6 -87.7 -87.6 -87.7 -87.6 ...
$ member_casual : chr [1:779148] "casual" "casual" "casual" "casual" ...
> #Combine all data into one trip
> all_trips <- bind_rows(Winter, Spring, Summer, Autumn)
> # Remove unwanted columns
> all_trips <- subset(all_trips, select = -c(start_lat, start_lng, end_lat, end_lng))
> colnames(all_trips) #show all colnames
[1] "ride_id" "rideable_type" "started_at" "ended_at" "start_st
ation_name" "start_station_id" "end_station_name"
[8] "end_station_id" "member_casual"
>
> dim(all_trips) #Dimensions of the whole Data
[1] 3116592 9
> nrow(all_trips) #no of rows
[1] 3116592
> head(all_trips) #load the first few rows of the data
# A tibble: 6 × 9
  ride_id rideable_type started_at ended_at start_station_name
  <chr> <chr> <dtm> <dtm> <chr>
  <chr> <chr> <chr> <chr> <chr>
1 BD0A6FF6FFF9B921 electric_bike 2020-11-01 13:36:00 2020-11-01 13:45:40 Dearborn St & Erie St
  110 St. Clair St & Er... 211 casual
2 96A7A7A4BDE4F82D electric_bike 2020-11-01 10:03:26 2020-11-01 10:14:45 Franklin St & Illinois S
  672 Noble St & Milwau... 29 casual
3 C61526D06582BDC5 electric_bike 2020-11-01 00:34:05 2020-11-01 01:03:06 Lake Shore Dr & Monroe S
  76 Federal St & Polk... 41 casual
4 E533E89C32080B9E electric_bike 2020-11-01 00:45:16 2020-11-01 00:54:31 Leavitt St & Chicago Ave
  659 Stave St & Armita... 185 casual
5 1C9F4EF18C168C60 electric_bike 2020-11-01 15:43:25 2020-11-01 16:16:52 Buckingham Fountain
  2 Buckingham Founta... 2 casual
6 7259585D8276D338 electric_bike 2020-11-14 15:55:17 2020-11-14 16:44:38 Wabash Ave & 16th St
  72 Lake Shore Dr & M... 76 casual
> summary(all_trips) #Shows the calculated summary
  ride_id rideable_type started_at ended_at
  start_station_name start_station_id end_station_name
Length:3116592 Length:3116592 Min. :2020-11-01 00:00:08.00 Min. :2020-11-01 00:02:
20.00 Length:3116592 Length:3116592 Length:3116592
Class :character Class :character 1st Qu.:2020-11-06 20:13:53.75 1st Qu.:2020-11-06 20:36:

```

```

08.25   Class :character   Class :character   Class :character
Mode    :character   Mode    :character   Median :2020-11-10 16:17:15.50   Median :2020-11-10 16:32:
15.00   Mode    :character   Mode    :character   Mode    :character
                                Mean    :2020-11-13 05:20:02.92   Mean    :2020-11-13 05:39:
45.40
                                3rd Qu.:2020-11-19 15:59:47.50   3rd Qu.:2020-11-19 16:18:
54.50
                                Max.    :2020-11-30 23:56:22.00   Max.    :2020-12-01 17:28:
22.00
end_station_id   member_casual
Length:3116592   Length:3116592
Class :character   Class :character
Mode    :character   Mode    :character

```

```

> #drop values with NA
> all_trips <- na.omit(all_trips)
> # Filter out started_at data that are less than ended_at data, avoiding negative value
> all_trips <- all_trips %>% filter(all_trips$started_at < all_trips$ended_at)
> #Create a new column and name it ride_length
>

> all_trips$ride_length <- all_trips$ended_at - all_trips$started_at
> #Format the column to hour, minutes, and seconds
> all_trips$ride_length <- hms::hms(seconds_to_period(all_trips$ride_length))
Error in seconds_to_period(all_trips$ride_length) :
  could not find function "seconds_to_period"
> library(lubridate)

```

Attaching package: 'lubridate'

The following objects are masked from 'package:base':

date, intersect, setdiff, union

```

> #Format the column to hour, minutes, and seconds
> all_trips$ride_length <- hms::hms(seconds_to_period(all_trips$ride_length))
> #Convert ride_length to number for easy calculation.
> all_trips <- mutate(all_trips, ride_length = as.numeric(ride_length))
> #Create a new column called day_of_week
> all_trips$day_of_week <- wday(all_trips$started_at, label = TRUE)
>
> #Save file as a CSV
> #all_trips %>% write.csv("clean_ride.csv")
> #all_trips %>% write.csv("dope_ride.csv")
> all_trips %>% write.csv("dope_ride.csv")
> # Descriptive analysis on ride_length
> mean(all_trips$ride_length) #average ride
[1] 1203.182
> quantile(all_trips$ride_length, .25) #Q1
25%
388
> quantile(all_trips$ride_length, .50) #Median
50%
692
> #median(all_trips$ride_length) #midpoint
> quantile(all_trips$ride_length, .75) #Q3
75%
1285
> max(all_trips$ride_length) #longest ride
[1] 2156040
> min(all_trips$ride_length) #shortest ride
[1] 1
> sd(all_trips$ride_length) #Standard Deviation
[1] 8762.412
> summary(all_trips$ride_length)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
     1      388     692    1203    1285 2156040
> #Compare ride_length of member_casual

```

```

> aggregate(all_trips$ride_length ~ all_trips$member_casual, FUN = mean)
  all_trips$member_casual all_trips$ride_length
1                casual      2008.8105
2                member      809.4879
> aggregate(all_trips$ride_length ~ all_trips$member_casual, FUN = median)
  all_trips$member_casual all_trips$ride_length
1                casual      1028
2                member       584
> aggregate(all_trips$ride_length ~ all_trips$member_casual, FUN = max)
  all_trips$member_casual all_trips$ride_length
1                casual     2156040
2                member     89996
> aggregate(all_trips$ride_length ~ all_trips$member_casual, FUN = min)
  all_trips$member_casual all_trips$ride_length
1                casual         1
2                member         1
> #Average ride_length by each day for member_casual column.
> aggregate(all_trips$ride_length ~ all_trips$member_casual + all_trips$day_of_week, FUN = mean)
  all_trips$member_casual all_trips$day_of_week all_trips$ride_length
1                casual      Sun      2459.6935
2                member      Sun      882.9196
3                casual     Mon     1725.7497
4                member     Mon      745.4346
5                casual     Tue     1728.4028
6                member     Tue      761.2670
7                casual     Wed     1535.9677
8                member     Wed      754.6756
9                casual     Thu     1827.8159
10               member     Thu      779.3516
11               casual     Fri     1919.1907
12               member     Fri      816.4063
13               casual     Sat     2236.8528
14               member     Sat      934.1594
> save.image("C:\\Users\\richy\\Desktop\\Google Capstone Project\\Capstone Project 1")
> #Number of rides by ride type
> all_trips %>%
+   mutate(weekday = wday(started_at, label = TRUE)) %>% #weekday field
+   group_by(member_casual, weekday) %>% #group by usertype and weekday
+   summarise(number_of_rides = n()
+             ,average_duration = mean(ride_length)) %>% #average duration
+   arrange(member_casual, weekday) %>% #Sort
+   #visualization
+   ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
+   geom_col(position = "dodge")
`summarise()` has grouped output by 'member_casual'. You can override using the `groups` argument.
> ## `summarise()` has grouped output by 'member_casual'. You can override using the `groups` argument.
> #Average duration
> all_trips %>%
+   mutate(weekday = wday(started_at, label = TRUE)) %>%
+   group_by(member_casual, weekday) %>%
+   summarise(number_of_rides = n()
+             ,average_duration = mean(ride_length)) %>%
+   arrange(member_casual, weekday) %>%
+   ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
+   geom_col(position = "dodge")
`summarise()` has grouped output by 'member_casual'. You can override using the `groups` argument.
> #Distributed within the weekday
> all_trips %>%
+   group_by(day_of_week) %>%
+   summarise(count = length(ride_id),
+             '%' = (length(ride_id) / nrow(all_trips)) * 100,
+             'member' = (sum(member_casual == "member") / length(ride_id)) * 100,
+             'casual' = (sum(member_casual == "casual") / length(ride_id)) * 100,
+             'Member x Casual Perc Diferrent' = member - casual)
# A tibble: 7 × 6
  day_of_week count    `%` member casual `Member x Casual Perc Diferrent`
  <ord>      <int> <dbl> <dbl> <dbl> <dbl>

```



```

1 Sun      391584  14.7  59.1  40.9      18.2
2 Mon      408768  15.3  74.2  25.8      48.4
3 Tue      312036  11.7  74.5  25.5      49.0
4 Wed      322392  12.1  74.7  25.3      49.4
5 Thu      350700  13.2  71.1  28.9      42.3
6 Fri      403296  15.1  67.2  32.8      34.3
7 Sat      474216  17.8  54.9  45.1      9.89
> ggplot(all_trips, aes(day_of_week, fill=member_casual)) +
+   geom_bar() +
+   labs(x="Weekday", title = "Distribution by weekday")
> #coord_flip()
> #Distributed by ride type
> all_trips %>%
+   group_by(rideable_type) %>%
+   summarise(count = length(ride_id),
+             '%' = (length(ride_id) / nrow(all_trips)) * 100,
+             'member' = (sum(member_casual == "member") / length(ride_id)) * 100,
+             'casual' = (sum(member_casual == "casual") / length(ride_id)) * 100,
+             'Member x Casual Perc Diferrent' = member - casual)
# A tibble: 2 × 6
  rideable_type    count    `%` member casual `Member x Casual Perc Diferrent`
  <chr>          <int> <dbl> <dbl> <dbl> <dbl>
1 docked_bike   1805208  67.8  69.6  30.4    39.2
2 electric_bike 857784   32.2  62.1  37.9    24.2
> ggplot(all_trips, aes(rideable_type, fill=member_casual)) +
+   geom_bar() +
+   labs(x="Riderable type", title = "Distribution by type of bikes")
> save.image("C:\\Users\\richy\\Desktop\\Google Capstone Project\\Capstone Project 1")
> local({fn<-choose.files(filters=Filters[c('R','txt','All'),],index=4)
+ file.show(fn,header=fn,title='')})
> local({fn<-choose.files(filters=Filters[c('R','txt','All'),],index=4)
+ file.show(fn,header=fn,title='')})
>

```