Cyclistic Case Study with R

Sadhia Rahman Meem

2024-05-07

## Cyclistic Bike Sharing Company Case Study with R

**Scenario:**

You are a junior data analyst working on the marketing analyst team at Cyclistic, a bike-share company in Chicago. The director of marketing believes the company’s future success depends on maximizing the number of annual memberships. Therefore, your team wants to understand how casual riders and annual members use Cyclistic bikes differently.From these insights, your team will design a new marketing strategy to convert casual riders into annual members. But first, Cyclistic executives must approve your recommendations, so they must be backed up with compelling data insights and professional data visualizations

What are the two Customer types? Casual Riders and Annual Members: Customers who purchase single-ride or full-day passes are referred to as casual riders.Customers who purchase annual memberships are Cyclistic members.

**Business Task:**

Analyze bike trips over the 12 months to identify trends and determine how Casual Riders differ from Annual members, and recommend marketing strategies for converting casual riders into annual members.

#### Let’s begin by loading tidyverse and other essential packages for data cleaning, manipulation and visualization

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(dplyr)  
library(lubridate)  
library(readr)  
library(tidyr)  
library(ggplot2)

*Now let’s import 12 months dataset for 2023 bike trips*

X202301\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202301-divvy-tripdata.csv")

## Rows: 190301 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.

X202302\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202302-divvy-tripdata.csv")

## Rows: 190445 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.

X202303\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202303-divvy-tripdata.csv")

## Rows: 258678 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.

X202304\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202304-divvy-tripdata.csv")

## Rows: 426590 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.

X202305\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202305-divvy-tripdata.csv")

## Rows: 604827 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.

X202306\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202306-divvy-tripdata.csv")

## Rows: 719618 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.

X202307\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202307-divvy-tripdata.csv")

## Rows: 767650 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.

X202308\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202308-divvy-tripdata.csv")

## Rows: 771693 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.

X202309\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202309-divvy-tripdata.csv")

## Rows: 666371 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.

X202310\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202310-divvy-tripdata.csv")

## Rows: 537113 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.

X202311\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202311-divvy-tripdata.csv")

## Rows: 362518 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.

X202312\_divvy\_tripdata <- read\_csv("C:/Users/Sadhi/Downloads/bike\_trip\_data/202312-divvy-tripdata.csv")

## Rows: 224073 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.

*Inspecting the columns for each dataset to make sure columns have same data type*

colnames (X202301\_divvy\_tripdata)

## [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 (X202302\_divvy\_tripdata)

## [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 (X202302\_divvy\_tripdata)

## [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 (X202304\_divvy\_tripdata)

## [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 (X202305\_divvy\_tripdata)

## [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 (X202306\_divvy\_tripdata)

## [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 (X202307\_divvy\_tripdata)

## [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 (X202308\_divvy\_tripdata)

## [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 (X202309\_divvy\_tripdata)

## [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 (X202310\_divvy\_tripdata)

## [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 (X202311\_divvy\_tripdata)

## [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 (X202312\_divvy\_tripdata)

## [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"

*Quick summary of the datasets*

X202301\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:190301 Length:190301 Min. :2023-01-01 00:01:58.00   
## Class :character Class :character 1st Qu.:2023-01-09 07:34:39.00   
## Mode :character Mode :character Median :2023-01-14 16:26:15.00   
## Mean :2023-01-15 09:12:02.25   
## 3rd Qu.:2023-01-21 14:24:46.00   
## Max. :2023-01-31 23:56:09.00   
## ended\_at member\_casual   
## Min. :2023-01-01 00:02:41.0 Length:190301   
## 1st Qu.:2023-01-09 07:45:42.0 Class :character   
## Median :2023-01-14 16:44:41.0 Mode :character   
## Mean :2023-01-15 09:25:02.2   
## 3rd Qu.:2023-01-21 14:37:42.0   
## Max. :2023-02-04 04:27:03.0

X202302\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:190445 Length:190445 Min. :2023-02-01 00:01:34.00   
## Class :character Class :character 1st Qu.:2023-02-08 18:25:33.00   
## Mode :character Mode :character Median :2023-02-14 21:36:30.00   
## Mean :2023-02-15 14:49:57.50   
## 3rd Qu.:2023-02-21 21:19:11.00   
## Max. :2023-02-28 23:59:31.00   
## ended\_at member\_casual   
## Min. :2023-02-01 00:08:42.00 Length:190445   
## 1st Qu.:2023-02-08 18:36:24.00 Class :character   
## Median :2023-02-14 21:51:19.00 Mode :character   
## Mean :2023-02-15 15:03:29.47   
## 3rd Qu.:2023-02-21 21:34:27.00   
## Max. :2023-03-06 15:09:53.00

X202303\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:258678 Length:258678 Min. :2023-03-01 00:00:50.00   
## Class :character Class :character 1st Qu.:2023-03-08 11:42:43.25   
## Mode :character Mode :character Median :2023-03-16 19:54:29.50   
## Mean :2023-03-16 22:25:38.67   
## 3rd Qu.:2023-03-24 19:22:40.25   
## Max. :2023-03-31 23:59:28.00   
## ended\_at member\_casual   
## Min. :2023-03-01 00:04:17.00 Length:258678   
## 1st Qu.:2023-03-08 11:55:00.00 Class :character   
## Median :2023-03-16 20:07:52.00 Mode :character   
## Mean :2023-03-16 22:38:43.47   
## 3rd Qu.:2023-03-24 19:35:15.25   
## Max. :2023-04-03 11:41:11.00

X202304\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:426590 Length:426590 Min. :2023-04-01 00:00:02.00   
## Class :character Class :character 1st Qu.:2023-04-10 13:22:13.00   
## Mode :character Mode :character Median :2023-04-15 09:52:13.00   
## Mean :2023-04-16 05:15:35.58   
## 3rd Qu.:2023-04-22 10:20:15.50   
## Max. :2023-04-30 23:59:05.00   
## ended\_at member\_casual   
## Min. :2023-04-01 00:03:10.00 Length:426590   
## 1st Qu.:2023-04-10 13:39:34.50 Class :character   
## Median :2023-04-15 10:11:11.00 Mode :character   
## Mean :2023-04-16 05:32:48.23   
## 3rd Qu.:2023-04-22 10:35:57.50   
## Max. :2023-05-03 10:37:12.00

X202305\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:604827 Length:604827 Min. :2023-05-01 00:00:33.00   
## Class :character Class :character 1st Qu.:2023-05-09 21:45:28.50   
## Mode :character Mode :character Median :2023-05-18 15:55:07.00   
## Mean :2023-05-17 18:36:34.06   
## 3rd Qu.:2023-05-25 12:39:23.00   
## Max. :2023-05-31 23:59:58.00   
## ended\_at member\_casual   
## Min. :2023-05-01 00:04:28.00 Length:604827   
## 1st Qu.:2023-05-09 21:59:55.00 Class :character   
## Median :2023-05-18 16:12:23.00 Mode :character   
## Mean :2023-05-17 18:55:36.07   
## 3rd Qu.:2023-05-25 12:53:21.00   
## Max. :2023-06-07 23:04:26.00

X202306\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:719618 Length:719618 Min. :2023-06-01 00:00:44.00   
## Class :character Class :character 1st Qu.:2023-06-07 23:00:11.00   
## Mode :character Mode :character Median :2023-06-16 13:17:35.00   
## Mean :2023-06-15 20:34:24.73   
## 3rd Qu.:2023-06-23 13:31:18.25   
## Max. :2023-06-30 23:59:56.00   
## ended\_at member\_casual   
## Min. :2023-06-01 00:02:56.00 Length:719618   
## 1st Qu.:2023-06-07 23:26:36.25 Class :character   
## Median :2023-06-16 13:35:12.00 Mode :character   
## Mean :2023-06-15 20:54:23.76   
## 3rd Qu.:2023-06-23 13:52:07.50   
## Max. :2023-07-10 20:26:44.00

X202307\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:767650 Length:767650 Min. :2023-07-01 00:00:00.00   
## Class :character Class :character 1st Qu.:2023-07-09 14:55:06.75   
## Mode :character Mode :character Median :2023-07-17 15:06:52.00   
## Mean :2023-07-17 05:56:09.47   
## 3rd Qu.:2023-07-24 16:15:23.50   
## Max. :2023-07-31 23:59:56.00   
## ended\_at member\_casual   
## Min. :2023-07-01 00:01:26.00 Length:767650   
## 1st Qu.:2023-07-09 15:19:39.00 Class :character   
## Median :2023-07-17 15:27:50.00 Mode :character   
## Mean :2023-07-17 06:17:53.75   
## 3rd Qu.:2023-07-24 16:32:51.75   
## Max. :2023-08-12 04:53:41.00

X202308\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:771693 Length:771693 Min. :2023-08-01 00:00:06.00   
## Class :character Class :character 1st Qu.:2023-08-08 17:16:56.00   
## Mode :character Mode :character Median :2023-08-16 17:08:21.00   
## Mean :2023-08-16 10:27:03.76   
## 3rd Qu.:2023-08-24 07:55:40.00   
## Max. :2023-08-31 23:59:44.00   
## ended\_at member\_casual   
## Min. :2023-08-01 00:01:03.00 Length:771693   
## 1st Qu.:2023-08-08 17:33:35.00 Class :character   
## Median :2023-08-16 17:24:27.00 Mode :character   
## Mean :2023-08-16 10:49:29.48   
## 3rd Qu.:2023-08-24 08:10:24.00   
## Max. :2023-10-10 04:56:16.00

X202309\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:666371 Length:666371 Min. :2023-09-01 00:00:44.00   
## Class :character Class :character 1st Qu.:2023-09-08 07:26:38.00   
## Mode :character Mode :character Median :2023-09-15 19:28:05.00   
## Mean :2023-09-16 00:11:48.34   
## 3rd Qu.:2023-09-23 15:03:51.00   
## Max. :2023-09-30 23:59:57.00   
## ended\_at member\_casual   
## Min. :2023-09-01 00:03:06.00 Length:666371   
## 1st Qu.:2023-09-08 07:39:33.50 Class :character   
## Median :2023-09-15 19:45:44.00 Mode :character   
## Mean :2023-09-16 00:29:40.64   
## 3rd Qu.:2023-09-23 15:26:19.00   
## Max. :2023-10-02 00:59:24.00

X202310\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:537113 Length:537113 Min. :2023-10-01 00:00:05.00   
## Class :character Class :character 1st Qu.:2023-10-06 19:18:31.00   
## Mode :character Mode :character Median :2023-10-14 19:56:05.00   
## Mean :2023-10-15 01:22:49.41   
## 3rd Qu.:2023-10-22 17:37:28.00   
## Max. :2023-10-31 23:59:57.00   
## ended\_at member\_casual   
## Min. :2023-10-01 00:02:02.00 Length:537113   
## 1st Qu.:2023-10-06 19:31:45.00 Class :character   
## Median :2023-10-14 20:09:10.00 Mode :character   
## Mean :2023-10-15 01:38:30.47   
## 3rd Qu.:2023-10-22 17:55:49.00   
## Max. :2023-11-01 21:23:59.00

X202311\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:362518 Length:362518 Min. :2023-11-01 00:01:46.00   
## Class :character Class :character 1st Qu.:2023-11-07 08:30:44.50   
## Mode :character Mode :character Median :2023-11-13 14:10:11.00   
## Mean :2023-11-14 01:44:47.81   
## 3rd Qu.:2023-11-19 10:35:17.25   
## Max. :2023-11-30 23:59:14.00   
## ended\_at member\_casual   
## Min. :2023-10-25 07:31:46.00 Length:362518   
## 1st Qu.:2023-11-07 08:41:43.25 Class :character   
## Median :2023-11-13 14:27:09.50 Mode :character   
## Mean :2023-11-14 01:58:22.95   
## 3rd Qu.:2023-11-19 10:52:09.75   
## Max. :2023-12-01 20:42:31.00

X202312\_divvy\_tripdata %>%   
 select(ride\_id,rideable\_type,started\_at,ended\_at,member\_casual)%>%  
 summary()

## ride\_id rideable\_type started\_at   
## Length:224073 Length:224073 Min. :2023-12-01 00:00:03.00   
## Class :character Class :character 1st Qu.:2023-12-07 16:18:35.00   
## Mode :character Mode :character Median :2023-12-13 12:05:44.00   
## Mean :2023-12-14 08:30:56.74   
## 3rd Qu.:2023-12-20 14:14:23.00   
## Max. :2023-12-31 23:59:38.00   
## ended\_at member\_casual   
## Min. :2023-12-01 00:04:12.00 Length:224073   
## 1st Qu.:2023-12-07 16:30:49.00 Class :character   
## Median :2023-12-13 12:16:31.00 Mode :character   
## Mean :2023-12-14 08:44:20.97   
## 3rd Qu.:2023-12-20 14:28:48.00   
## Max. :2024-01-01 23:50:51.00

*Let’s combine the individual datasets into one larger dataset for full analysis*

bike\_trips\_2023 <- bind\_rows(X202301\_divvy\_tripdata, X202302\_divvy\_tripdata, X202303\_divvy\_tripdata, X202304\_divvy\_tripdata,X202305\_divvy\_tripdata, X202306\_divvy\_tripdata,X202307\_divvy\_tripdata, X202308\_divvy\_tripdata, X202309\_divvy\_tripdata,X202310\_divvy\_tripdata, X202311\_divvy\_tripdata, X202312\_divvy\_tripdata)

*Quick overview of the combined dataset*

print(bike\_trips\_2023)

## # A tibble: 5,719,877 × 13  
## ride\_id rideable\_type started\_at ended\_at   
## <chr> <chr> <dttm> <dttm>   
## 1 F96D5A74A3E41399 electric\_bike 2023-01-21 20:05:42 2023-01-21 20:16:33  
## 2 13CB7EB698CEDB88 classic\_bike 2023-01-10 15:37:36 2023-01-10 15:46:05  
## 3 BD88A2E670661CE5 electric\_bike 2023-01-02 07:51:57 2023-01-02 08:05:11  
## 4 C90792D034FED968 classic\_bike 2023-01-22 10:52:58 2023-01-22 11:01:44  
## 5 3397017529188E8A classic\_bike 2023-01-12 13:58:01 2023-01-12 14:13:20  
## 6 58E68156DAE3E311 electric\_bike 2023-01-31 07:18:03 2023-01-31 07:21:16  
## 7 2F7194B6012A98D4 electric\_bike 2023-01-15 21:18:36 2023-01-15 21:32:36  
## 8 DB1CF84154D6A049 classic\_bike 2023-01-25 10:49:01 2023-01-25 10:58:22  
## 9 34EAB943F88C4C5D electric\_bike 2023-01-25 20:49:47 2023-01-25 21:02:14  
## 10 BC8AB1AA51DA9115 classic\_bike 2023-01-06 16:37:19 2023-01-06 16:49:52  
## # ℹ 5,719,867 more rows  
## # ℹ 9 more variables: start\_station\_name <chr>, start\_station\_id <chr>,  
## # end\_station\_name <chr>, end\_station\_id <chr>, start\_lat <dbl>,  
## # start\_lng <dbl>, end\_lat <dbl>, end\_lng <dbl>, member\_casual <chr>

glimpse(bike\_trips\_2023)

## Rows: 5,719,877  
## Columns: 13  
## $ ride\_id <chr> "F96D5A74A3E41399", "13CB7EB698CEDB88", "BD88A2E670…  
## $ rideable\_type <chr> "electric\_bike", "classic\_bike", "electric\_bike", "…  
## $ started\_at <dttm> 2023-01-21 20:05:42, 2023-01-10 15:37:36, 2023-01-…  
## $ ended\_at <dttm> 2023-01-21 20:16:33, 2023-01-10 15:46:05, 2023-01-…  
## $ start\_station\_name <chr> "Lincoln Ave & Fullerton Ave", "Kimbark Ave & 53rd …  
## $ start\_station\_id <chr> "TA1309000058", "TA1309000037", "RP-005", "TA130900…  
## $ end\_station\_name <chr> "Hampden Ct & Diversey Ave", "Greenwood Ave & 47th …  
## $ end\_station\_id <chr> "202480.0", "TA1308000002", "599", "TA1308000002", …  
## $ start\_lat <dbl> 41.92407, 41.79957, 42.00857, 41.79957, 41.79957, 4…  
## $ start\_lng <dbl> -87.64628, -87.59475, -87.69048, -87.59475, -87.594…  
## $ end\_lat <dbl> 41.93000, 41.80983, 42.03974, 41.80983, 41.80983, 4…  
## $ end\_lng <dbl> -87.64000, -87.59938, -87.69941, -87.59938, -87.599…  
## $ member\_casual <chr> "member", "member", "casual", "member", "member", "…

colnames(bike\_trips\_2023)

## [1] "ride\_id" "rideable\_type" "started\_at"   
## [4] "ended\_at" "start\_station\_name" "start\_station\_id"   
## [7] "end\_station\_name" "end\_station\_id" "start\_lat"   
## [10] "start\_lng" "end\_lat" "end\_lng"   
## [13] "member\_casual"

nrow(bike\_trips\_2023)

## [1] 5719877

summary(bike\_trips\_2023)

## ride\_id rideable\_type started\_at   
## Length:5719877 Length:5719877 Min. :2023-01-01 00:01:58.00   
## Class :character Class :character 1st Qu.:2023-05-21 12:50:44.00   
## Mode :character Mode :character Median :2023-07-20 18:02:50.00   
## Mean :2023-07-16 10:27:50.01   
## 3rd Qu.:2023-09-16 20:08:49.00   
## Max. :2023-12-31 23:59:38.00   
##   
## ended\_at start\_station\_name start\_station\_id   
## Min. :2023-01-01 00:02:41.00 Length:5719877 Length:5719877   
## 1st Qu.:2023-05-21 13:14:09.00 Class :character Class :character   
## Median :2023-07-20 18:19:47.00 Mode :character Mode :character   
## Mean :2023-07-16 10:46:00.18   
## 3rd Qu.:2023-09-16 20:28:10.00   
## Max. :2024-01-01 23:50:51.00   
##   
## end\_station\_name end\_station\_id start\_lat start\_lng   
## Length:5719877 Length:5719877 Min. :41.63 Min. :-87.94   
## Class :character Class :character 1st Qu.:41.88 1st Qu.:-87.66   
## Mode :character Mode :character Median :41.90 Median :-87.64   
## Mean :41.90 Mean :-87.65   
## 3rd Qu.:41.93 3rd Qu.:-87.63   
## Max. :42.07 Max. :-87.46   
##   
## end\_lat end\_lng member\_casual   
## Min. : 0.00 Min. :-88.16 Length:5719877   
## 1st Qu.:41.88 1st Qu.:-87.66 Class :character   
## Median :41.90 Median :-87.64 Mode :character   
## Mean :41.90 Mean :-87.65   
## 3rd Qu.:41.93 3rd Qu.:-87.63   
## Max. :42.18 Max. : 0.00   
## NA's :6990 NA's :6990

*Lets make a new column called ‘day\_of\_week’*

bike\_trips\_2023 <- bike\_trips\_2023 %>%  
 mutate(day\_of\_week= wday(bike\_trips\_2023$started\_at, label=TRUE, abbr= TRUE))  
  
bike\_trips\_2023 # Print the dataframe

## # A tibble: 5,719,877 × 14  
## ride\_id rideable\_type started\_at ended\_at   
## <chr> <chr> <dttm> <dttm>   
## 1 F96D5A74A3E41399 electric\_bike 2023-01-21 20:05:42 2023-01-21 20:16:33  
## 2 13CB7EB698CEDB88 classic\_bike 2023-01-10 15:37:36 2023-01-10 15:46:05  
## 3 BD88A2E670661CE5 electric\_bike 2023-01-02 07:51:57 2023-01-02 08:05:11  
## 4 C90792D034FED968 classic\_bike 2023-01-22 10:52:58 2023-01-22 11:01:44  
## 5 3397017529188E8A classic\_bike 2023-01-12 13:58:01 2023-01-12 14:13:20  
## 6 58E68156DAE3E311 electric\_bike 2023-01-31 07:18:03 2023-01-31 07:21:16  
## 7 2F7194B6012A98D4 electric\_bike 2023-01-15 21:18:36 2023-01-15 21:32:36  
## 8 DB1CF84154D6A049 classic\_bike 2023-01-25 10:49:01 2023-01-25 10:58:22  
## 9 34EAB943F88C4C5D electric\_bike 2023-01-25 20:49:47 2023-01-25 21:02:14  
## 10 BC8AB1AA51DA9115 classic\_bike 2023-01-06 16:37:19 2023-01-06 16:49:52  
## # ℹ 5,719,867 more rows  
## # ℹ 10 more variables: start\_station\_name <chr>, start\_station\_id <chr>,  
## # end\_station\_name <chr>, end\_station\_id <chr>, start\_lat <dbl>,  
## # start\_lng <dbl>, end\_lat <dbl>, end\_lng <dbl>, member\_casual <chr>,  
## # day\_of\_week <ord>

*Now let’s calculate the ride length in seconds by subtracting ended\_at from started\_at*

bike\_trips\_2023 <- bike\_trips\_2023 %>%  
 mutate(ride\_length = ended\_at - started\_at)  
  
bike\_trips\_2023 #print the dataset

## # A tibble: 5,719,877 × 15  
## ride\_id rideable\_type started\_at ended\_at   
## <chr> <chr> <dttm> <dttm>   
## 1 F96D5A74A3E41399 electric\_bike 2023-01-21 20:05:42 2023-01-21 20:16:33  
## 2 13CB7EB698CEDB88 classic\_bike 2023-01-10 15:37:36 2023-01-10 15:46:05  
## 3 BD88A2E670661CE5 electric\_bike 2023-01-02 07:51:57 2023-01-02 08:05:11  
## 4 C90792D034FED968 classic\_bike 2023-01-22 10:52:58 2023-01-22 11:01:44  
## 5 3397017529188E8A classic\_bike 2023-01-12 13:58:01 2023-01-12 14:13:20  
## 6 58E68156DAE3E311 electric\_bike 2023-01-31 07:18:03 2023-01-31 07:21:16  
## 7 2F7194B6012A98D4 electric\_bike 2023-01-15 21:18:36 2023-01-15 21:32:36  
## 8 DB1CF84154D6A049 classic\_bike 2023-01-25 10:49:01 2023-01-25 10:58:22  
## 9 34EAB943F88C4C5D electric\_bike 2023-01-25 20:49:47 2023-01-25 21:02:14  
## 10 BC8AB1AA51DA9115 classic\_bike 2023-01-06 16:37:19 2023-01-06 16:49:52  
## # ℹ 5,719,867 more rows  
## # ℹ 11 more variables: start\_station\_name <chr>, start\_station\_id <chr>,  
## # end\_station\_name <chr>, end\_station\_id <chr>, start\_lat <dbl>,  
## # start\_lng <dbl>, end\_lat <dbl>, end\_lng <dbl>, member\_casual <chr>,  
## # day\_of\_week <ord>, ride\_length <drtn>

*Convert the ride\_length into numeric data*

bike\_trips\_2023 <- bike\_trips\_2023 %>%  
 mutate(ride\_length= as.numeric(ride\_length))  
  
is.numeric(bike\_trips\_2023$ride\_length)

## [1] TRUE

*Let’s remove any negative and 0 values from ride\_length*

cleaned\_bike\_trips\_2023 <- bike\_trips\_2023[!(bike\_trips\_2023$ride\_length <0),]  
  
cleaned\_bike\_trips\_2023 #prints the dataset

## # A tibble: 5,719,605 × 15  
## ride\_id rideable\_type started\_at ended\_at   
## <chr> <chr> <dttm> <dttm>   
## 1 F96D5A74A3E41399 electric\_bike 2023-01-21 20:05:42 2023-01-21 20:16:33  
## 2 13CB7EB698CEDB88 classic\_bike 2023-01-10 15:37:36 2023-01-10 15:46:05  
## 3 BD88A2E670661CE5 electric\_bike 2023-01-02 07:51:57 2023-01-02 08:05:11  
## 4 C90792D034FED968 classic\_bike 2023-01-22 10:52:58 2023-01-22 11:01:44  
## 5 3397017529188E8A classic\_bike 2023-01-12 13:58:01 2023-01-12 14:13:20  
## 6 58E68156DAE3E311 electric\_bike 2023-01-31 07:18:03 2023-01-31 07:21:16  
## 7 2F7194B6012A98D4 electric\_bike 2023-01-15 21:18:36 2023-01-15 21:32:36  
## 8 DB1CF84154D6A049 classic\_bike 2023-01-25 10:49:01 2023-01-25 10:58:22  
## 9 34EAB943F88C4C5D electric\_bike 2023-01-25 20:49:47 2023-01-25 21:02:14  
## 10 BC8AB1AA51DA9115 classic\_bike 2023-01-06 16:37:19 2023-01-06 16:49:52  
## # ℹ 5,719,595 more rows  
## # ℹ 11 more variables: start\_station\_name <chr>, start\_station\_id <chr>,  
## # end\_station\_name <chr>, end\_station\_id <chr>, start\_lat <dbl>,  
## # start\_lng <dbl>, end\_lat <dbl>, end\_lng <dbl>, member\_casual <chr>,  
## # day\_of\_week <ord>, ride\_length <dbl>

*Find mean, median, max, min ride\_length*

mean(cleaned\_bike\_trips\_2023$ride\_length) #average ride\_length

## [1] 1091.15

median(cleaned\_bike\_trips\_2023$ride\_length) #median ride\_length

## [1] 572

max(cleaned\_bike\_trips\_2023$ride\_length) #longest ride

## [1] 5909344

min(cleaned\_bike\_trips\_2023$ride\_length) #shortest ride

## [1] 0

*Lets calculate the average, median,max, min ride\_length by user type: member vs casual*

cleaned\_bike\_trips\_2023 %>%  
 group\_by(member\_casual)%>%  
 summarise(average\_ride\_length=mean(ride\_length), median\_ride\_length=median(ride\_length),   
 max\_ride\_length=max(ride\_length),min\_ride\_length=min(ride\_length))

## # A tibble: 2 × 5  
## member\_casual average\_ride\_length median\_ride\_length max\_ride\_length  
## <chr> <dbl> <dbl> <dbl>  
## 1 casual 1695. 711 5909344  
## 2 member 752. 511 93580  
## # ℹ 1 more variable: min\_ride\_length <dbl>

**Note**: *the longest ride (max ride\_length) for casual riders is 5909344 sec or approximately 68 days, which seems a bit out of place but since we assume that the riders are able to dock bike at any station I guess this would kinda makes sense*

*Creating a new data frame and calculating average ride length by each user type*

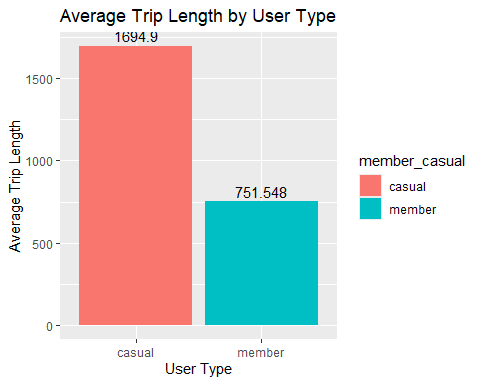
df\_bike\_avg<- cleaned\_bike\_trips\_2023 %>%  
 select(member\_casual, ride\_length) %>%  
 group\_by(member\_casual) %>%  
 summarise(average\_trip\_length= mean(ride\_length)) %>%  
 arrange(member\_casual)  
  
df\_bike\_avg # Print the dataframe

## # A tibble: 2 × 2  
## member\_casual average\_trip\_length  
## <chr> <dbl>  
## 1 casual 1695.  
## 2 member 752.

### Here comes the fun part: Data Visualization :)

*We will create a bar chart depicting Average Trip Length by each User Type*

ggplot(data = df\_bike\_avg) +  
 geom\_bar(aes(x = member\_casual, y = average\_trip\_length, fill = member\_casual), stat = "identity") +  
 geom\_text(aes(label = signif(average\_trip\_length),x=member\_casual, y = average\_trip\_length), vjust = -0.4) +  
 labs(x = "User Type", y = "Average Trip Length", title = "Average Trip Length by User Type")



In this plot, we can see that Casual rider have a higher average trip duration than Members

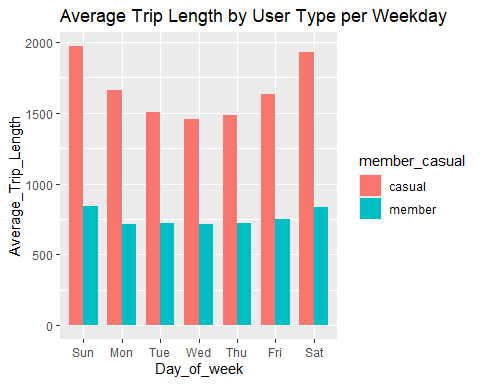
*Next we will calculate Average\_trip\_length by each User Type per day in a dataframe called “df\_avg\_per\_weekday”*

df\_avg\_per\_weekday <- cleaned\_bike\_trips\_2023 %>%  
 select(member\_casual, ride\_length, day\_of\_week) %>%  
 group\_by(member\_casual, day\_of\_week) %>%  
 summarise(average\_trip\_length=mean(ride\_length),.groups = "drop") %>%  
 arrange(member\_casual, day\_of\_week)   
  
df\_avg\_per\_weekday #print the dataframe

## # A tibble: 14 × 3  
## member\_casual day\_of\_week average\_trip\_length  
## <chr> <ord> <dbl>  
## 1 casual Sun 1972.  
## 2 casual Mon 1663.  
## 3 casual Tue 1505.  
## 4 casual Wed 1458.  
## 5 casual Thu 1484.  
## 6 casual Fri 1636.  
## 7 casual Sat 1928.  
## 8 member Sun 839.  
## 9 member Mon 714.  
## 10 member Tue 721.  
## 11 member Wed 717.  
## 12 member Thu 721.  
## 13 member Fri 749.  
## 14 member Sat 836.

#### Data Visualization: A column chart depicting the Average\_trip\_length by User Type per Weekday

ggplot(data = df\_avg\_per\_weekday, (aes(x = day\_of\_week, y = average\_trip\_length, fill = member\_casual)))+  
 geom\_col(width = 0.75, position = position\_dodge(width = 0.75)) +  
 labs(x = "Day\_of\_week", y = "Average\_Trip\_Length" , title = "Average Trip Length by User Type per Weekday")



Here we can see again that Casual riders have a higher average trip length per day compared to Members

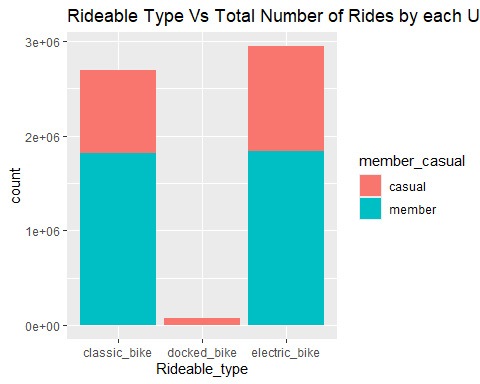
*Lastly we will look at the types of bikes vs user type*

cleaned\_bike\_trips\_2023 %>%  
 group\_by(rideable\_type) %>%  
 summarise(count= length(ride\_id))

## # A tibble: 3 × 2  
## rideable\_type count  
## <chr> <int>  
## 1 classic\_bike 2695968  
## 2 docked\_bike 78287  
## 3 electric\_bike 2945350

*Now let’s visualize bike types and user types*

ggplot(data = cleaned\_bike\_trips\_2023, aes(x= rideable\_type, fill=member\_casual))+  
 geom\_bar()+  
 labs(x="Rideable\_type", title= "Rideable Type Vs Total Number of Rides by each User type")



This plot shows that members prefer to use classic and electric bikes more than casual riders, and docked bikes are only used by casual riders

#### Key Findings:

* Casual riders have a higher average trip duration than Annual members
* Casual riders also have a higher average trip length per weekday compared to Annual members, however the trip average for Annual members are pretty consistent among all seven days of the week while, Casual riders have a spike on the Weekends
* We can also assume that most annual members have day job and perhaps use bikes for commuting
* Annual members prefer to use classic and electric bikes more than casual riders
* Docked bikes are only used by Casual riders

#### Recommendations for Cyclistic

* Member benefits: Discounts on new cyclistic memberships to target casual riders
* Investing in technology and product improvement to enhance customer experience for Annual members and making membership signup process
* Fitness goals: Cyclistic could collaborate with a well known fitness app or apply a new feature on the cyclistic app that tracks customer’s overall fitness improvement from bike rides and long term effects on their health
* Cost effectiveness: in the app we can show side by side comparison of single day or full day ride passes vs Annual membership to show how much money Customers can save by converting to annual memberships

#### Conclusion:

After analyzing and visualizing Cyclistic bike trip data, it’s evident that Casual riders take longer bike trips than Annual members. In order to convert Casual riders into Annual members,we can implement marketing strategies aimed at improving user experience, enhancing the cost-effectiveness of memberships, and addressing fitness goals for riders