# Google Data Analytics Capstone: Cyclistic Bike-Share Analysis Case Study

#### Trang Vu

#### 09/11/2021

This is a case study for the Google Data Analytics Professional Certificate. The project provides the Cyclistic Datasets for the learners to follow the steps of data analysis process: ask, prepare, process, analyze, share and act in order to answer the key business problems.

# Phrase 1: Ask

In this phase, I need to do two things. I define the problem to be solved and I make sure that I fully understand stakeholder expectations.

### About the company

The direct of the marking team Lily Moreno believes that maximizing the number of annual members will be key to future growth. Rather than creating a marketing campaign that targets all-new customers, Moreno believes there is a very good chance to convert casual riders into members. Moreno has set a clear goal: Design marketing strategies aimed at converting casual riders into annual members.

###Business Task Analyze the most recent 12 month Cyclistic Customer Data (from 10/2020 to 09/2021) in order to ansIr the key questions:

1. How do annual members and casual riders use Cyclistic bikes differently?
2. Why would casual riders buy Cyclistic annual memberships?
3. How can Cyclistic use digital media to influence casual riders to become members?

### Key Stakeholders:

Cyclistic executive team, Lily Moreno: The director of marketing and my manager.

# Phrase 2: Prepare

This is where the data analysts collect and store data so later I will use for the upcoming analysis process. In this phrase, I will learn more about the different types of data and how to identify which kinds of data are most useful for solving a particular problem.

### Import libraries

#helps wrangle data

library(tidyverse)

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

## v ggplot2 3.3.5 v purrr 0.3.4

## v tibble 3.1.4 v dplyr 1.0.7

## v tidyr 1.1.3 v stringr 1.4.0

## v readr 2.0.1 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --

## x dplyr::filter() masks stats::filter()

## x dplyr::lag() masks stats::lag()

#helps wrangle date attributes

library(lubridate)

##

## Attaching package: 'lubridate'

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

##

## date, intersect, setdiff, union

#helps visualize data

library(ggplot2)

## Step 1: Load datasets

Upload Divvy datasets (csv files) here.

d10\_2020 <- read\_csv("202010-divvy-tripdata.csv")

## Rows: 388653 Columns: 13

## -- Column specification --------------------------------------------------------

## Delimiter: ","

## chr (5): ride\_id, rideable\_type, start\_station\_name, end\_station\_name, memb...

## dbl (6): start\_station\_id, end\_station\_id, start\_lat, start\_lng, end\_lat, e...

## dttm (2): started\_at, ended\_at

##

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

## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

d11\_2020 <- read\_csv("202011-divvy-tripdata.csv")

## Rows: 259716 Columns: 13

## -- Column specification --------------------------------------------------------

## Delimiter: ","

## chr (5): ride\_id, rideable\_type, start\_station\_name, end\_station\_name, memb...

## dbl (6): start\_station\_id, end\_station\_id, start\_lat, start\_lng, end\_lat, e...

## dttm (2): started\_at, ended\_at

##

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

## i Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

d12\_2020 <- read\_csv("202012-divvy-tripdata.csv")

## Rows: 131573 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.

d01\_2021 <- read\_csv("202101-divvy-tripdata.csv")

## Rows: 96834 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.

d02\_2021 <- read\_csv("202102-divvy-tripdata.csv")

## Rows: 49622 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.

d03\_2021 <- read\_csv("202103-divvy-tripdata.csv")

## Rows: 228496 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.

d04\_2021 <- read\_csv("202104-divvy-tripdata.csv")

## Rows: 337230 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.

d05\_2021 <- read\_csv("202105-divvy-tripdata.csv")

## Rows: 531633 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.

d06\_2021 <- read\_csv("202106-divvy-tripdata.csv")

## Rows: 729595 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.

d07\_2021 <- read\_csv("202107-divvy-tripdata.csv")

## Rows: 822410 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.

d08\_2021 <- read\_csv("202108-divvy-tripdata.csv")

## Rows: 804352 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.

d09\_2021 <- read\_csv("202109-divvy-tripdata.csv")

## Rows: 756147 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.

## Step 2: Wrangle data and combine into a single file

### Compare column names each of the files

As all names are already consistent - they do not need to be renamed.

colnames(d10\_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(d11\_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(d12\_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(d01\_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(d02\_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(d03\_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(d04\_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(d05\_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(d06\_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(d07\_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(d08\_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(d09\_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"

### Inspect the dataframes and look for incongruencies

str(d10\_2020)

## spec\_tbl\_df [388,653 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:388653] "ACB6B40CF5B9044C" "DF450C72FD109C01" "B6396B54A15AC0DF" "44A4AEE261B9E854" ...

## $ rideable\_type : chr [1:388653] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:388653], format: "2020-10-31 19:39:43" "2020-10-31 23:50:08" ...

## $ ended\_at : POSIXct[1:388653], format: "2020-10-31 19:57:12" "2020-11-01 00:04:16" ...

## $ start\_station\_name: chr [1:388653] "Lakeview Ave & Fullerton Pkwy" "Southport Ave & Waveland Ave" "Stony Island Ave & 67th St" "Clark St & Grace St" ...

## $ start\_station\_id : num [1:388653] 313 227 102 165 190 359 313 125 NA 174 ...

## $ end\_station\_name : chr [1:388653] "Rush St & Hubbard St" "Kedzie Ave & Milwaukee Ave" "University Ave & 57th St" "Broadway & Sheridan Rd" ...

## $ end\_station\_id : num [1:388653] 125 260 423 256 185 53 125 313 199 635 ...

## $ start\_lat : num [1:388653] 41.9 41.9 41.8 42 41.9 ...

## $ start\_lng : num [1:388653] -87.6 -87.7 -87.6 -87.7 -87.7 ...

## $ end\_lat : num [1:388653] 41.9 41.9 41.8 42 41.9 ...

## $ end\_lng : num [1:388653] -87.6 -87.7 -87.6 -87.7 -87.7 ...

## $ member\_casual : chr [1:388653] "casual" "casual" "casual" "casual" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_double(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_double(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d11\_2020)

## spec\_tbl\_df [259,716 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:259716] "BD0A6FF6FFF9B921" "96A7A7A4BDE4F82D" "C61526D06582BDC5" "E533E89C32080B9E" ...

## $ rideable\_type : chr [1:259716] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:259716], format: "2020-11-01 13:36:00" "2020-11-01 10:03:26" ...

## $ ended\_at : POSIXct[1:259716], format: "2020-11-01 13:45:40" "2020-11-01 10:14:45" ...

## $ start\_station\_name: chr [1:259716] "Dearborn St & Erie St" "Franklin St & Illinois St" "Lake Shore Dr & Monroe St" "Leavitt St & Chicago Ave" ...

## $ start\_station\_id : num [1:259716] 110 672 76 659 2 72 76 NA 58 394 ...

## $ end\_station\_name : chr [1:259716] "St. Clair St & Erie St" "Noble St & Milwaukee Ave" "Federal St & Polk St" "Stave St & Armitage Ave" ...

## $ end\_station\_id : num [1:259716] 211 29 41 185 2 76 72 NA 288 273 ...

## $ start\_lat : num [1:259716] 41.9 41.9 41.9 41.9 41.9 ...

## $ start\_lng : num [1:259716] -87.6 -87.6 -87.6 -87.7 -87.6 ...

## $ end\_lat : num [1:259716] 41.9 41.9 41.9 41.9 41.9 ...

## $ end\_lng : num [1:259716] -87.6 -87.7 -87.6 -87.7 -87.6 ...

## $ member\_casual : chr [1:259716] "casual" "casual" "casual" "casual" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_double(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_double(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d12\_2020)

## spec\_tbl\_df [131,573 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:131573] "70B6A9A437D4C30D" "158A465D4E74C54A" "5262016E0F1F2F9A" "BE119628E44F871E" ...

## $ rideable\_type : chr [1:131573] "classic\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:131573], format: "2020-12-27 12:44:29" "2020-12-18 17:37:15" ...

## $ ended\_at : POSIXct[1:131573], format: "2020-12-27 12:55:06" "2020-12-18 17:44:19" ...

## $ start\_station\_name: chr [1:131573] "Aberdeen St & Jackson Blvd" NA NA NA ...

## $ start\_station\_id : chr [1:131573] "13157" NA NA NA ...

## $ end\_station\_name : chr [1:131573] "Desplaines St & Kinzie St" NA NA NA ...

## $ end\_station\_id : chr [1:131573] "TA1306000003" NA NA NA ...

## $ start\_lat : num [1:131573] 41.9 41.9 41.9 41.9 41.8 ...

## $ start\_lng : num [1:131573] -87.7 -87.7 -87.7 -87.7 -87.6 ...

## $ end\_lat : num [1:131573] 41.9 41.9 41.9 41.9 41.8 ...

## $ end\_lng : num [1:131573] -87.6 -87.7 -87.7 -87.7 -87.6 ...

## $ member\_casual : chr [1:131573] "member" "member" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d01\_2021)

## spec\_tbl\_df [96,834 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:96834] "E19E6F1B8D4C42ED" "DC88F20C2C55F27F" "EC45C94683FE3F27" "4FA453A75AE377DB" ...

## $ rideable\_type : chr [1:96834] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:96834], format: "2021-01-23 16:14:19" "2021-01-27 18:43:08" ...

## $ ended\_at : POSIXct[1:96834], format: "2021-01-23 16:24:44" "2021-01-27 18:47:12" ...

## $ start\_station\_name: chr [1:96834] "California Ave & Cortez St" "California Ave & Cortez St" "California Ave & Cortez St" "California Ave & Cortez St" ...

## $ start\_station\_id : chr [1:96834] "17660" "17660" "17660" "17660" ...

## $ end\_station\_name : chr [1:96834] NA NA NA NA ...

## $ end\_station\_id : chr [1:96834] NA NA NA NA ...

## $ start\_lat : num [1:96834] 41.9 41.9 41.9 41.9 41.9 ...

## $ start\_lng : num [1:96834] -87.7 -87.7 -87.7 -87.7 -87.7 ...

## $ end\_lat : num [1:96834] 41.9 41.9 41.9 41.9 41.9 ...

## $ end\_lng : num [1:96834] -87.7 -87.7 -87.7 -87.7 -87.7 ...

## $ member\_casual : chr [1:96834] "member" "member" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d02\_2021)

## spec\_tbl\_df [49,622 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:49622] "89E7AA6C29227EFF" "0FEFDE2603568365" "E6159D746B2DBB91" "B32D3199F1C2E75B" ...

## $ rideable\_type : chr [1:49622] "classic\_bike" "classic\_bike" "electric\_bike" "classic\_bike" ...

## $ started\_at : POSIXct[1:49622], format: "2021-02-12 16:14:56" "2021-02-14 17:52:38" ...

## $ ended\_at : POSIXct[1:49622], format: "2021-02-12 16:21:43" "2021-02-14 18:12:09" ...

## $ start\_station\_name: chr [1:49622] "Glenwood Ave & Touhy Ave" "Glenwood Ave & Touhy Ave" "Clark St & Lake St" "Wood St & Chicago Ave" ...

## $ start\_station\_id : chr [1:49622] "525" "525" "KA1503000012" "637" ...

## $ end\_station\_name : chr [1:49622] "Sheridan Rd & Columbia Ave" "Bosworth Ave & Howard St" "State St & Randolph St" "Honore St & Division St" ...

## $ end\_station\_id : chr [1:49622] "660" "16806" "TA1305000029" "TA1305000034" ...

## $ start\_lat : num [1:49622] 42 42 41.9 41.9 41.8 ...

## $ start\_lng : num [1:49622] -87.7 -87.7 -87.6 -87.7 -87.6 ...

## $ end\_lat : num [1:49622] 42 42 41.9 41.9 41.8 ...

## $ end\_lng : num [1:49622] -87.7 -87.7 -87.6 -87.7 -87.6 ...

## $ member\_casual : chr [1:49622] "member" "casual" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d03\_2021)

## spec\_tbl\_df [228,496 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:228496] "CFA86D4455AA1030" "30D9DC61227D1AF3" "846D87A15682A284" "994D05AA75A168F2" ...

## $ rideable\_type : chr [1:228496] "classic\_bike" "classic\_bike" "classic\_bike" "classic\_bike" ...

## $ started\_at : POSIXct[1:228496], format: "2021-03-16 08:32:30" "2021-03-28 01:26:28" ...

## $ ended\_at : POSIXct[1:228496], format: "2021-03-16 08:36:34" "2021-03-28 01:36:55" ...

## $ start\_station\_name: chr [1:228496] "Humboldt Blvd & Armitage Ave" "Humboldt Blvd & Armitage Ave" "Shields Ave & 28th Pl" "Winthrop Ave & Lawrence Ave" ...

## $ start\_station\_id : chr [1:228496] "15651" "15651" "15443" "TA1308000021" ...

## $ end\_station\_name : chr [1:228496] "Stave St & Armitage Ave" "Central Park Ave & Bloomingdale Ave" "Halsted St & 35th St" "Broadway & Sheridan Rd" ...

## $ end\_station\_id : chr [1:228496] "13266" "18017" "TA1308000043" "13323" ...

## $ start\_lat : num [1:228496] 41.9 41.9 41.8 42 42 ...

## $ start\_lng : num [1:228496] -87.7 -87.7 -87.6 -87.7 -87.7 ...

## $ end\_lat : num [1:228496] 41.9 41.9 41.8 42 42.1 ...

## $ end\_lng : num [1:228496] -87.7 -87.7 -87.6 -87.6 -87.7 ...

## $ member\_casual : chr [1:228496] "casual" "casual" "casual" "casual" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d04\_2021)

## spec\_tbl\_df [337,230 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:337230] "6C992BD37A98A63F" "1E0145613A209000" "E498E15508A80BAD" "1887262AD101C604" ...

## $ rideable\_type : chr [1:337230] "classic\_bike" "docked\_bike" "docked\_bike" "classic\_bike" ...

## $ started\_at : POSIXct[1:337230], format: "2021-04-12 18:25:36" "2021-04-27 17:27:11" ...

## $ ended\_at : POSIXct[1:337230], format: "2021-04-12 18:56:55" "2021-04-27 18:31:29" ...

## $ start\_station\_name: chr [1:337230] "State St & Pearson St" "Dorchester Ave & 49th St" "Loomis Blvd & 84th St" "Honore St & Division St" ...

## $ start\_station\_id : chr [1:337230] "TA1307000061" "KA1503000069" "20121" "TA1305000034" ...

## $ end\_station\_name : chr [1:337230] "Southport Ave & Waveland Ave" "Dorchester Ave & 49th St" "Loomis Blvd & 84th St" "Southport Ave & Waveland Ave" ...

## $ end\_station\_id : chr [1:337230] "13235" "KA1503000069" "20121" "13235" ...

## $ start\_lat : num [1:337230] 41.9 41.8 41.7 41.9 41.7 ...

## $ start\_lng : num [1:337230] -87.6 -87.6 -87.7 -87.7 -87.7 ...

## $ end\_lat : num [1:337230] 41.9 41.8 41.7 41.9 41.7 ...

## $ end\_lng : num [1:337230] -87.7 -87.6 -87.7 -87.7 -87.7 ...

## $ member\_casual : chr [1:337230] "member" "casual" "casual" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d05\_2021)

## spec\_tbl\_df [531,633 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:531633] "C809ED75D6160B2A" "DD59FDCE0ACACAF3" "0AB83CB88C43EFC2" "7881AC6D39110C60" ...

## $ rideable\_type : chr [1:531633] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:531633], format: "2021-05-30 11:58:15" "2021-05-30 11:29:14" ...

## $ ended\_at : POSIXct[1:531633], format: "2021-05-30 12:10:39" "2021-05-30 12:14:09" ...

## $ start\_station\_name: chr [1:531633] NA NA NA NA ...

## $ start\_station\_id : chr [1:531633] NA NA NA NA ...

## $ end\_station\_name : chr [1:531633] NA NA NA NA ...

## $ end\_station\_id : chr [1:531633] NA NA NA NA ...

## $ start\_lat : num [1:531633] 41.9 41.9 41.9 41.9 41.9 ...

## $ start\_lng : num [1:531633] -87.6 -87.6 -87.7 -87.7 -87.7 ...

## $ end\_lat : num [1:531633] 41.9 41.8 41.9 41.9 41.9 ...

## $ end\_lng : num [1:531633] -87.6 -87.6 -87.7 -87.7 -87.7 ...

## $ member\_casual : chr [1:531633] "casual" "casual" "casual" "casual" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d06\_2021)

## spec\_tbl\_df [729,595 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:729595] "99FEC93BA843FB20" "06048DCFC8520CAF" "9598066F68045DF2" "B03C0FE48C412214" ...

## $ rideable\_type : chr [1:729595] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:729595], format: "2021-06-13 14:31:28" "2021-06-04 11:18:02" ...

## $ ended\_at : POSIXct[1:729595], format: "2021-06-13 14:34:11" "2021-06-04 11:24:19" ...

## $ start\_station\_name: chr [1:729595] NA NA NA NA ...

## $ start\_station\_id : chr [1:729595] NA NA NA NA ...

## $ end\_station\_name : chr [1:729595] NA NA NA NA ...

## $ end\_station\_id : chr [1:729595] NA NA NA NA ...

## $ start\_lat : num [1:729595] 41.8 41.8 41.8 41.8 41.8 ...

## $ start\_lng : num [1:729595] -87.6 -87.6 -87.6 -87.6 -87.6 ...

## $ end\_lat : num [1:729595] 41.8 41.8 41.8 41.8 41.8 ...

## $ end\_lng : num [1:729595] -87.6 -87.6 -87.6 -87.6 -87.6 ...

## $ member\_casual : chr [1:729595] "member" "member" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d07\_2021)

## spec\_tbl\_df [822,410 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:822410] "0A1B623926EF4E16" "B2D5583A5A5E76EE" "6F264597DDBF427A" "379B58EAB20E8AA5" ...

## $ rideable\_type : chr [1:822410] "docked\_bike" "classic\_bike" "classic\_bike" "classic\_bike" ...

## $ started\_at : POSIXct[1:822410], format: "2021-07-02 14:44:36" "2021-07-07 16:57:42" ...

## $ ended\_at : POSIXct[1:822410], format: "2021-07-02 15:19:58" "2021-07-07 17:16:09" ...

## $ start\_station\_name: chr [1:822410] "Michigan Ave & Washington St" "California Ave & Cortez St" "Wabash Ave & 16th St" "California Ave & Cortez St" ...

## $ start\_station\_id : chr [1:822410] "13001" "17660" "SL-012" "17660" ...

## $ end\_station\_name : chr [1:822410] "Halsted St & North Branch St" "Wood St & Hubbard St" "Rush St & Hubbard St" "Carpenter St & Huron St" ...

## $ end\_station\_id : chr [1:822410] "KA1504000117" "13432" "KA1503000044" "13196" ...

## $ start\_lat : num [1:822410] 41.9 41.9 41.9 41.9 41.9 ...

## $ start\_lng : num [1:822410] -87.6 -87.7 -87.6 -87.7 -87.7 ...

## $ end\_lat : num [1:822410] 41.9 41.9 41.9 41.9 41.9 ...

## $ end\_lng : num [1:822410] -87.6 -87.7 -87.6 -87.7 -87.7 ...

## $ member\_casual : chr [1:822410] "casual" "casual" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d08\_2021)

## spec\_tbl\_df [804,352 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:804352] "99103BB87CC6C1BB" "EAFCCCFB0A3FC5A1" "9EF4F46C57AD234D" "5834D3208BFAF1DA" ...

## $ rideable\_type : chr [1:804352] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:804352], format: "2021-08-10 17:15:49" "2021-08-10 17:23:14" ...

## $ ended\_at : POSIXct[1:804352], format: "2021-08-10 17:22:44" "2021-08-10 17:39:24" ...

## $ start\_station\_name: chr [1:804352] NA NA NA NA ...

## $ start\_station\_id : chr [1:804352] NA NA NA NA ...

## $ end\_station\_name : chr [1:804352] NA NA NA NA ...

## $ end\_station\_id : chr [1:804352] NA NA NA NA ...

## $ start\_lat : num [1:804352] 41.8 41.8 42 42 41.8 ...

## $ start\_lng : num [1:804352] -87.7 -87.7 -87.7 -87.7 -87.6 ...

## $ end\_lat : num [1:804352] 41.8 41.8 42 42 41.8 ...

## $ end\_lng : num [1:804352] -87.7 -87.6 -87.7 -87.7 -87.6 ...

## $ member\_casual : chr [1:804352] "member" "member" "member" "member" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

str(d09\_2021)

## spec\_tbl\_df [756,147 x 13] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:756147] "9DC7B962304CBFD8" "F930E2C6872D6B32" "6EF72137900BB910" "78D1DE133B3DBF55" ...

## $ rideable\_type : chr [1:756147] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:756147], format: "2021-09-28 16:07:10" "2021-09-28 14:24:51" ...

## $ ended\_at : POSIXct[1:756147], format: "2021-09-28 16:09:54" "2021-09-28 14:40:05" ...

## $ start\_station\_name: chr [1:756147] NA NA NA NA ...

## $ start\_station\_id : chr [1:756147] NA NA NA NA ...

## $ end\_station\_name : chr [1:756147] NA NA NA NA ...

## $ end\_station\_id : chr [1:756147] NA NA NA NA ...

## $ start\_lat : num [1:756147] 41.9 41.9 41.8 41.8 41.9 ...

## $ start\_lng : num [1:756147] -87.7 -87.6 -87.7 -87.7 -87.7 ...

## $ end\_lat : num [1:756147] 41.9 42 41.8 41.8 41.9 ...

## $ end\_lng : num [1:756147] -87.7 -87.7 -87.7 -87.7 -87.7 ...

## $ member\_casual : chr [1:756147] "casual" "casual" "casual" "casual" ...

## - attr(\*, "spec")=

## .. cols(

## .. ride\_id = col\_character(),

## .. rideable\_type = col\_character(),

## .. started\_at = col\_datetime(format = ""),

## .. ended\_at = col\_datetime(format = ""),

## .. start\_station\_name = col\_character(),

## .. start\_station\_id = col\_character(),

## .. end\_station\_name = col\_character(),

## .. end\_station\_id = col\_character(),

## .. start\_lat = col\_double(),

## .. start\_lng = col\_double(),

## .. end\_lat = col\_double(),

## .. end\_lng = col\_double(),

## .. member\_casual = col\_character()

## .. )

## - attr(\*, "problems")=<externalptr>

### Inspect the dataframes and look for incongruencies

After the above comparison, I need to convert start\_station\_id to character so I can perform calculations correctly later on.

d10\_2020 <- mutate(d10\_2020, start\_station\_id = as.character(start\_station\_id)

,end\_station\_id = as.character(end\_station\_id))

d11\_2020 <- mutate(d11\_2020, start\_station\_id = as.character(start\_station\_id)

,end\_station\_id = as.character(end\_station\_id))

### Stack individual quarter’s data frames into one big data frame

all\_trips <- bind\_rows(d10\_2020, d11\_2020, d12\_2020, d01\_2021, d02\_2021, d03\_2021, d04\_2021, d05\_2021, d06\_2021, d07\_2021, d08\_2021, d09\_2021)

### Remove start\_lat, start\_lng, end\_lat, end\_lng fields as this data was dropped beginning in 2020

all\_trips <- all\_trips %>%

select(-c(start\_lat,start\_lng,end\_lat,end\_lng))

colnames(all\_trips)

## [1] "ride\_id" "rideable\_type" "started\_at"

## [4] "ended\_at" "start\_station\_name" "start\_station\_id"

## [7] "end\_station\_name" "end\_station\_id" "member\_casual"

# Phrase 3: Process

A process known as data cleaning is the fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. What I aim to achieve is clean data.

## Step 3: Clean up and add data to prepare for analysis

### Inspect the new table that has been created

#List of column names

colnames(all\_trips)

## [1] "ride\_id" "rideable\_type" "started\_at"

## [4] "ended\_at" "start\_station\_name" "start\_station\_id"

## [7] "end\_station\_name" "end\_station\_id" "member\_casual"

#How many rows are in data frame?

nrow(all\_trips)

## [1] 5136261

#Dimensions of the data frame?

dim(all\_trips)

## [1] 5136261 9

#See the first 6 rows of data frame. Also tail(all\_trips)

head(all\_trips)

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

str(all\_trips)

## tibble [5,136,261 x 9] (S3: tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:5136261] "ACB6B40CF5B9044C" "DF450C72FD109C01" "B6396B54A15AC0DF" "44A4AEE261B9E854" ...

## $ rideable\_type : chr [1:5136261] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:5136261], format: "2020-10-31 19:39:43" "2020-10-31 23:50:08" ...

## $ ended\_at : POSIXct[1:5136261], format: "2020-10-31 19:57:12" "2020-11-01 00:04:16" ...

## $ start\_station\_name: chr [1:5136261] "Lakeview Ave & Fullerton Pkwy" "Southport Ave & Waveland Ave" "Stony Island Ave & 67th St" "Clark St & Grace St" ...

## $ start\_station\_id : chr [1:5136261] "313" "227" "102" "165" ...

## $ end\_station\_name : chr [1:5136261] "Rush St & Hubbard St" "Kedzie Ave & Milwaukee Ave" "University Ave & 57th St" "Broadway & Sheridan Rd" ...

## $ end\_station\_id : chr [1:5136261] "125" "260" "423" "256" ...

## $ member\_casual : chr [1:5136261] "casual" "casual" "casual" "casual" ...

#Statistical summary of data. Mainly for numberics

summary(all\_trips)

## ride\_id rideable\_type started\_at

## Length:5136261 Length:5136261 Min. :2020-10-01 00:00:06

## Class :character Class :character 1st Qu.:2021-04-11 18:50:57

## Mode :character Mode :character Median :2021-06-21 18:01:31

## Mean :2021-05-25 22:30:57

## 3rd Qu.:2021-08-11 21:13:51

## Max. :2021-09-30 23:59:48

## ended\_at start\_station\_name start\_station\_id

## Min. :2020-10-01 00:05:09 Length:5136261 Length:5136261

## 1st Qu.:2021-04-11 19:15:05 Class :character Class :character

## Median :2021-06-21 18:20:59 Mode :character Mode :character

## Mean :2021-05-25 22:51:34

## 3rd Qu.:2021-08-11 21:33:57

## Max. :2021-10-01 22:55:35

## end\_station\_name end\_station\_id member\_casual

## Length:5136261 Length:5136261 Length:5136261

## Class :character Class :character Class :character

## Mode :character Mode :character Mode :character

##

##

##

### Remove inconsistency

There are four unique values in member\_casual subscriber, member, customer, casual but 2020 on wards these member has been changed into two unique values that are member, casual.

table(all\_trips$member\_casual)

##

## casual member

## 2358287 2777974

all\_trips <- all\_trips %>%

mutate(member\_casual = recode(member\_casual

,"Subscriber" = "member"

,"Customer" = "Casual"))

table(all\_trips$member\_casual)

##

## casual member

## 2358287 2777974

### Day (Add new columns)

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 I could only aggregate at the ride level.

#The default format is yyyy-mm-dd

all\_trips$date <- as.Date(all\_trips$started\_at)

all\_trips$month <- format(as.Date(all\_trips$date), "%m")

all\_trips$day <- format(as.Date(all\_trips$date), "%d")

all\_trips$year <- format(as.Date(all\_trips$date), "%Y")

all\_trips$day\_of\_Iek <- format(as.Date(all\_trips$date), "%A")

### Ride length (Add new column)

ride\_length is the distance betIen started time and ended time.

# Add a "ride\_length" calculation to all\_trips (in minutes)

all\_trips$ride\_length <- difftime(all\_trips$ended\_at,all\_trips$started\_at,units = "mins")

head(all\_trips$ride\_length)

## Time differences in mins

## [1] 17.483333 14.133333 8.350000 2.866667 16.216667 7.650000

# Inspect the structure of the columns

str(all\_trips)

## tibble [5,136,261 x 15] (S3: tbl\_df/tbl/data.frame)

## $ ride\_id : chr [1:5136261] "ACB6B40CF5B9044C" "DF450C72FD109C01" "B6396B54A15AC0DF" "44A4AEE261B9E854" ...

## $ rideable\_type : chr [1:5136261] "electric\_bike" "electric\_bike" "electric\_bike" "electric\_bike" ...

## $ started\_at : POSIXct[1:5136261], format: "2020-10-31 19:39:43" "2020-10-31 23:50:08" ...

## $ ended\_at : POSIXct[1:5136261], format: "2020-10-31 19:57:12" "2020-11-01 00:04:16" ...

## $ start\_station\_name: chr [1:5136261] "Lakeview Ave & Fullerton Pkwy" "Southport Ave & Waveland Ave" "Stony Island Ave & 67th St" "Clark St & Grace St" ...

## $ start\_station\_id : chr [1:5136261] "313" "227" "102" "165" ...

## $ end\_station\_name : chr [1:5136261] "Rush St & Hubbard St" "Kedzie Ave & Milwaukee Ave" "University Ave & 57th St" "Broadway & Sheridan Rd" ...

## $ end\_station\_id : chr [1:5136261] "125" "260" "423" "256" ...

## $ member\_casual : chr [1:5136261] "casual" "casual" "casual" "casual" ...

## $ date : Date[1:5136261], format: "2020-10-31" "2020-10-31" ...

## $ month : chr [1:5136261] "10" "10" "10" "10" ...

## $ day : chr [1:5136261] "31" "31" "31" "31" ...

## $ year : chr [1:5136261] "2020" "2020" "2020" "2020" ...

## $ day\_of\_Iek : chr [1:5136261] "Saturday" "Saturday" "Saturday" "Saturday" ...

## $ ride\_length : 'difftime' num [1:5136261] 17.4833333333333 14.1333333333333 8.35 2.86666666666667 ...

## ..- attr(\*, "units")= chr "mins"

# Convert "ride\_length" from Factor to numeric so I can run calculations on the data

is.factor(all\_trips$ride\_length)

## [1] FALSE

all\_trips$ride\_length <- as.numeric(as.character(all\_trips$ride\_length))

is.numeric(all\_trips$ride\_length)

## [1] TRUE

# Remove "bad" data

# The dataframe includes a few hundred entries when bikes Ire taken out of docks and checked for quality by Divvy or ride\_length was negative

# I will create a new version of the dataframe (v2) since data is being removed

all\_trips\_v2 <- all\_trips[!(all\_trips$start\_station\_name == "HQ QR" | all\_trips$ride\_length<0),]

### Remove NA

Remove the missing values in the dataset.

#Check the missing values in the dataset.

colSums(is.na(all\_trips\_v2))

## ride\_id rideable\_type started\_at ended\_at

## 523409 523409 523409 523409

## start\_station\_name start\_station\_id end\_station\_name end\_station\_id

## 523409 523726 781675 781869

## member\_casual date month day

## 523409 523409 523409 523409

## year day\_of\_Iek ride\_length

## 523409 523409 523409

#Remove NA

all\_trips\_v3 <- all\_trips\_v2[!(is.na(all\_trips\_v2$start\_station\_id) | is.na(all\_trips\_v2$end\_station\_id) | is.na(all\_trips\_v2$member\_casual) | is.na(all\_trips\_v2$end\_station\_name)),]

table(all\_trips\_v3$member\_casual)

##

## casual member

## 1963854 2386998

#Check again for the missing values in the dataset.

colSums(is.na(all\_trips\_v3))

## ride\_id rideable\_type started\_at ended\_at

## 0 0 0 0

## start\_station\_name start\_station\_id end\_station\_name end\_station\_id

## 0 0 0 0

## member\_casual date month day

## 0 0 0 0

## year day\_of\_Iek ride\_length

## 0 0 0

# Phrase 4: Analyze

Analyzing the data I’ve collected involves using tools to transform and organize that information so that I can draw useful conclusions, make predictions, and drive informed decision-making.

### Conduct Descriptive analysis

Firstly, I need to look at the basic descriptive statistics of the data.

# Statistic summary of ride length in minutes

summary(all\_trips\_v3$ride\_length)

## Min. 1st Qu. Median Mean 3rd Qu. Max.

## 0.00 7.22 12.70 22.65 22.98 55944.15

# Compare members and casual users

aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual, FUN = mean)

aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual, FUN = median)

aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual, FUN = max)

aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual, FUN = min)

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

all\_trips\_v3$day\_of\_Iek <- ordered(all\_trips\_v3$day\_of\_Iek, levels=c("Sunday", "Monday", "Tuesday", "Idnesday", "Thursday", "Friday", "Saturday"))

Now, let’s run the average ride time by each day for members vs casual users.

aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual + all\_trips\_v3$day\_of\_Iek, FUN = mean)

The I will look at the total number of rides and the average ride duration (in seconds) by weekday for casual customers and members.

# analyze ridership data by type and Iekday

all\_trips\_v3 %>%

#creates Iekday field using wday()

mutate(Iekday = wday(started\_at, label = TRUE)) %>%

#groups by usertype and Iekday

group\_by(member\_casual, Iekday) %>%

#calculates the number of rides and average duration

summarise(number\_of\_rides = n()

#calculates the average duration

,average\_duration = mean(ride\_length)) %>%

#sorts

arrange(member\_casual, Iekday)

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

# Phrase 5: Share

Here I learn how data analysts interpret results and share them with others to help stakeholders make effective data-driven decisions. In the share phase, visualization is a data analyst’s best friend.

### Visualization 1: Total number of rides by rider type

Let’s visualize the number of rides by rider type.

all\_trips\_v3 %>%

mutate(Iekday = wday(started\_at, label = TRUE)) %>%

group\_by(member\_casual, Iekday) %>%

summarise(number\_of\_rides = n(),average\_duration = mean(ride\_length)) %>%

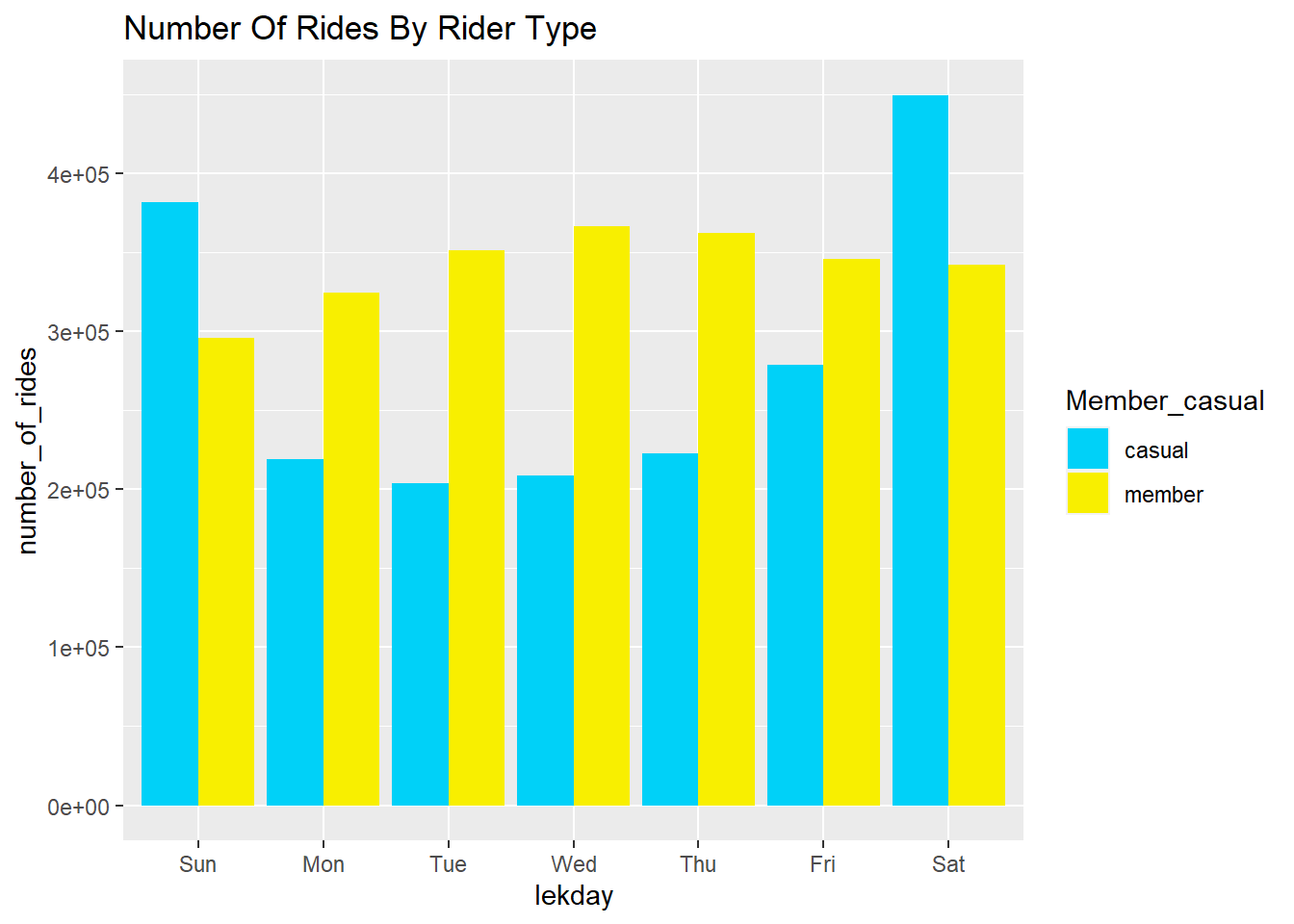
arrange(member\_casual, Iekday) %>%

ggplot(aes(x = Iekday, y = number\_of\_rides, fill = member\_casual)) + geom\_col(position = "dodge") +

scale\_fill\_manual("Member\_casual",values = c('#00D1F8', '#F8EF00'))+

ggtitle( "Number Of Rides By Rider Type")

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



### Visualization 2: Average Cyclistic Bike-Share Rides

Let’s create a visualization for average duration.

all\_trips\_v3 %>%

mutate(Iekday = wday(started\_at, label = TRUE)) %>%

group\_by(member\_casual, Iekday) %>%

summarise(number\_of\_rides = n()

,average\_duration = mean(ride\_length)) %>%

arrange(member\_casual, Iekday) %>%

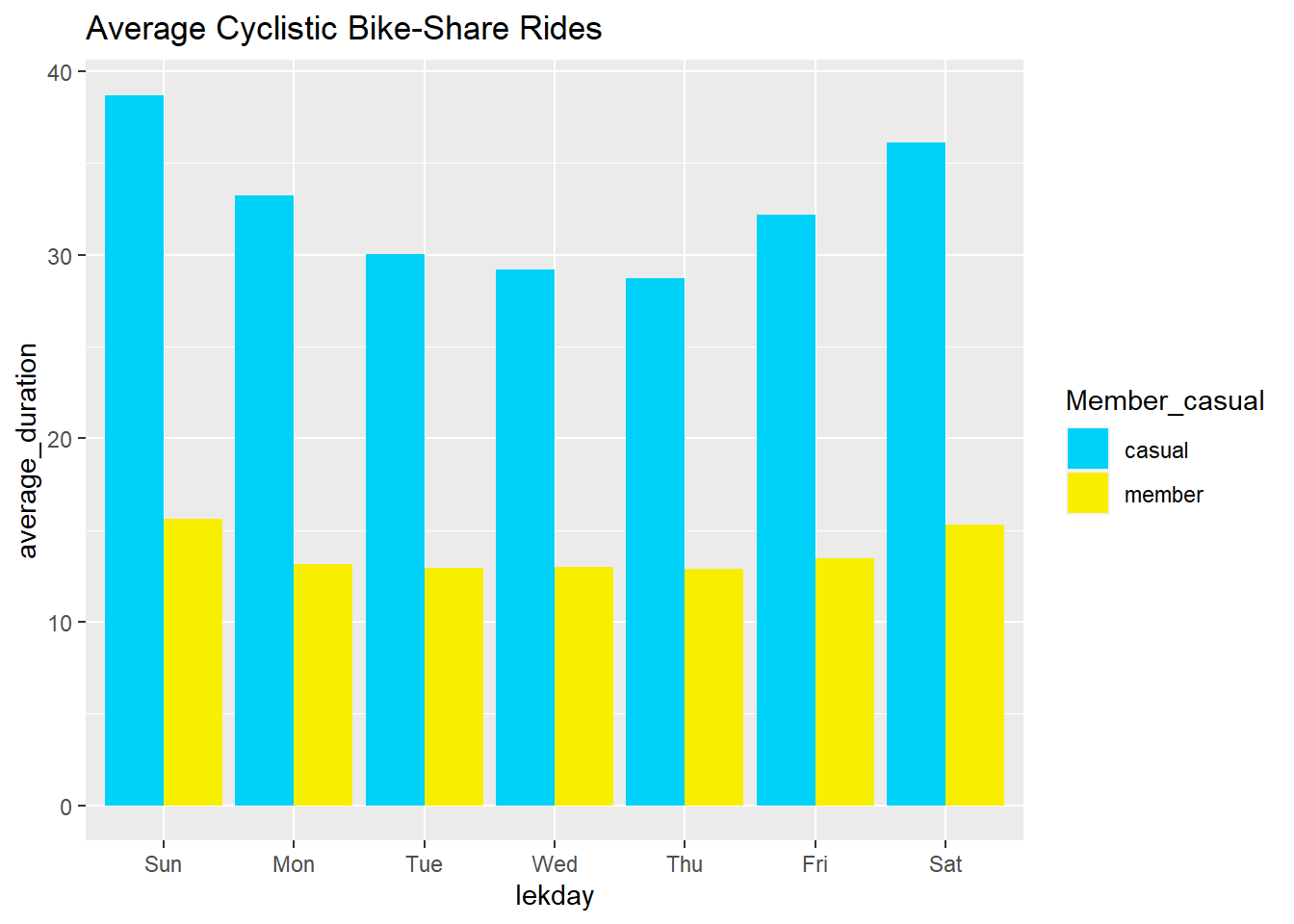
ggplot(aes(x = Iekday, y = average\_duration, fill = member\_casual)) +

geom\_col(position = "dodge")+

scale\_fill\_manual("Member\_casual",values = c('#00D1F8', '#F8EF00'))+

ggtitle( "Average Cyclistic Bike-Share Rides")

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



## Export summary file for further analysis

Exported the data as a csv file.

# Create a csv file that I will visualize in Excel, Tableau, or my presentation software

counts <- aggregate(all\_trips\_v3$ride\_length ~ all\_trips\_v3$member\_casual + all\_trips\_v3$day\_of\_Iek, FUN = mean)

write.csv(counts, file = 'avg\_ride\_length.csv')

# Phrase 6: ACT

Now, I know the problem, Let’s solve it! This is the phase where I need carefully go through our data problem and the analysis I made to make a data-driven decision.

### Key Findings

Based on the “Number of Rides By Rider Type” graph, we can see that members usually use bike on weekdays while the casual members mostly use bike during their weekend. It can be explained that the members use bike to commute to work on the daily basic while the casual members just just bike for their leisure on the weekend.

According to the “Average Cyclistic Bike-Share Rides” graph, we also see that casual members usually use bike for a longer period of time while members consistently use bike for a shorter time.

### Recommendations

1. Charge higher price for non-members during the weekends in order to encourage the casual members to sign up for membership.
2. Change the pricing system as follows:

* Limiting the hours for the non-members during weekends.
* Allow annual members to use bike for the higher duration compared to the non-members.

By following these recommendations, the Cyclistic can convert more casual members into the annual members.