Cyclistic

ILYAS SACHA

2023-07-12

## About the company

**Cyclistic:** A bike-share program that features more than **5,800** bicycles and **600** docking stations. **Cyclistic** sets itself apart by also offering reclining bikes, hand tricycles, and cargo bikes, making bike-share more inclusive to people with disabilities and riders who can’t use a standard two-wheeled bike. The majority of riders opt for traditional bikes; about **8%** of riders use the assistive options. **Cyclistic** users are more likely to ride for leisure, but about **30%** use them to commute to work each day.

Customers who purchase single-ride or full-day passes are referred to as **casual riders**. Customers who purchase annual memberships are **Cyclistic members**. **Cyclistic’s** finance analysts have concluded that **annual members** are much more profitable than casual riders.

**Moreno** (Director of Marketing) has set a **clear goal**: Design marketing strategies aimed at converting **casual riders** into **annual members**. In order to do that, however, the marketing analyst team needs to better understand how **annual members** and **casual riders** differ, why **casual riders** would buy a **membership**, and how digital media could affect their marketing tactics. **Moreno** and her team are interested in analyzing the **Cyclistic** historical bike trip **data** to identify **trends**.

## About the task

### Business goal

aiming to increase the number of annual memberships and covert casual riders into annual members.

### Business task

How do annual members and casual riders use Cyclistic bikes differently?

### Stakeholders

**Primary stakeholders:** Lily Moreno  
**secondary stakeholders:** Cyclistic executive team

## about the data

* the data is located in a source [url](https://divvy-tripdata.s3.amazonaws.com/index.html), provided by Motivate International Inc.
* The files are listed in chronological order, starting from the earliest available data.
* The data files are organized per month for the year 2022. Each file contains data for one month. The file naming convention follows the pattern “YYYYMM-divvy-tripdata.csv”, where “YYYY” represents the four-digit year and “MM” represents the two-digit month.
* the data file “202201-divvy-tripdata.csv” contains data for January 2022, “202202-divvy-tripdata.csv” contains data for February 2022, and so on.
* The organization of the data allows for easy identification and retrieval of trip data based on the desired time period.
* the data is protected under this [license](https://ride.divvybikes.com/data-license-agreement)
* this data-privacy issues prohibit you from using riders’ personally identifiable information. This means that you won’t be able to connect pass purchases to credit card numbers to determine if casual riders live in the **Cyclistic** service area or if they have purchased multiple single passes.

## Data description

loading libraries

library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.2 ✔ readr 2.1.4  
## ✔ forcats 1.0.0 ✔ stringr 1.5.0  
## ✔ ggplot2 3.4.2 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.2 ✔ tidyr 1.3.0  
## ✔ purrr 1.0.1   
## ── 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(skimr)  
library(janitor)

##   
## Attaching package: 'janitor'  
##   
## The following objects are masked from 'package:stats':  
##   
## chisq.test, fisher.test

library(lubridate)

The data files contain 13 columns each. The column names are as follows:

trips\_data <- read\_csv("C:/Users/isach/OneDrive/Bureau/projects/dataset/202207-divvy-tripdata.csv")

## Rows: 823488 Columns: 13  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (9): ride\_id, rideable\_type, started\_at, ended\_at, start\_station\_name, s...  
## dbl (4): start\_lat, start\_lng, end\_lat, end\_lng  
##   
## 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.

* 9 of these variables are categorical.
* The other 4 variables are quantitative.
* Some of the files contain missing values in the columns for start\_station\_name, start\_station\_id, end\_station\_name, and end\_station\_id. However, latitude and longitude data are still present, which provides location information for the corresponding stations.

## cleaning the data

### tools used

**SQL:** To process and clean the data.  
**R :**  To analyse and visualize the data.

### **using R**

### Cleaning

**Merging the data frames** To simplify the process, I merged the separate data frames into one using R. This allows me to easily work with and analyze the combined dataset.

#Setting the Data File Folder Path  
folder\_path <- "C:/Users/isach/OneDrive/Bureau/projects/dataset"  
  
#create a list to store the data frames we will be working with.  
data\_frames <- list()  
  
# Get a list of all files in the folder  
file\_list <- list.files(path = folder\_path, full.names = TRUE)  
  
# Iterate through the files and import them into individual data frames  
for (file in file\_list) {  
 df <- read.csv(file)  
   
 # Rename columns to ensure consistency  
 colnames(df) <- c(  
 "ride\_id", "rideable\_type", "started\_at", "ended\_at", "start\_station\_name",   
 "start\_station\_id", "end\_station\_name", "end\_station\_id", "start\_lat",   
 "start\_lng", "end\_lat", "end\_lng", "member\_casual"  
 )  
 data\_frames[[file]] <- df  
}  
  
#Combine the individual data frames into one data frame  
combined\_cyclistic <- do.call(rbind, data\_frames)  
  
#saving our combined table  
write.csv(combined\_cyclistic, file = "combined\_cyclistic.csv", row.names = FALSE)

Now we have **13 columns** and **5,745,753 total rows**

#### removing bike\_id column

data\_ride\_id\_removed <- select(combined\_cyclistic, -ride\_id)  
colnames(data\_ride\_id\_removed)

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

### Cleaning the stations names

To handle missing station names, we will use the latitude and longitude coordinates to infer the names of the stations. By analyzing the geographical information, we can estimate the station names for the missing data points. However, since we don’t have access to the main database and the reverse geocoding method is not accurate, we will instead choose to remove the rows that have missing data. This way, we can ensure that the missing data doesn’t interfere with our analysis.

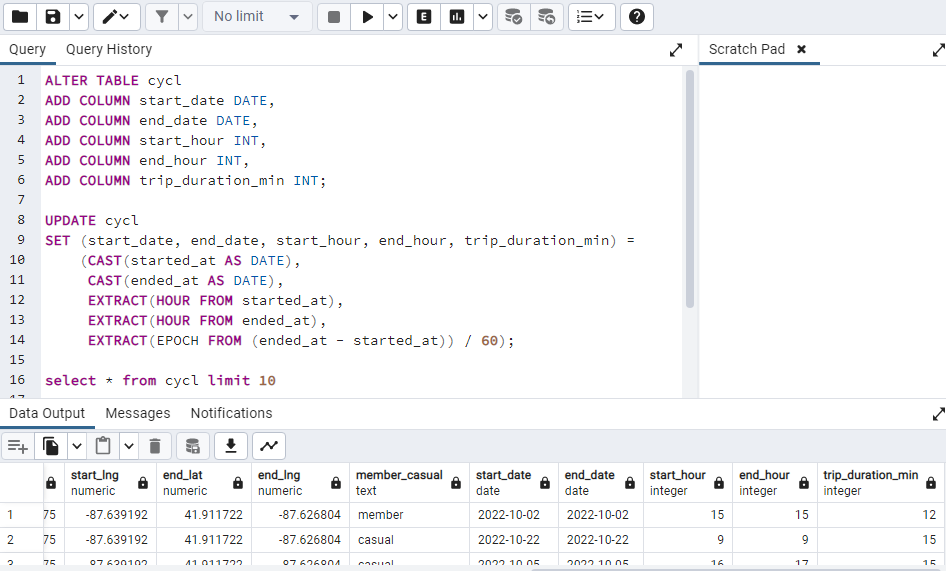
#### Cleaning station names in R

Removing rows were start\_station\_name, end\_station\_name have blank cells.

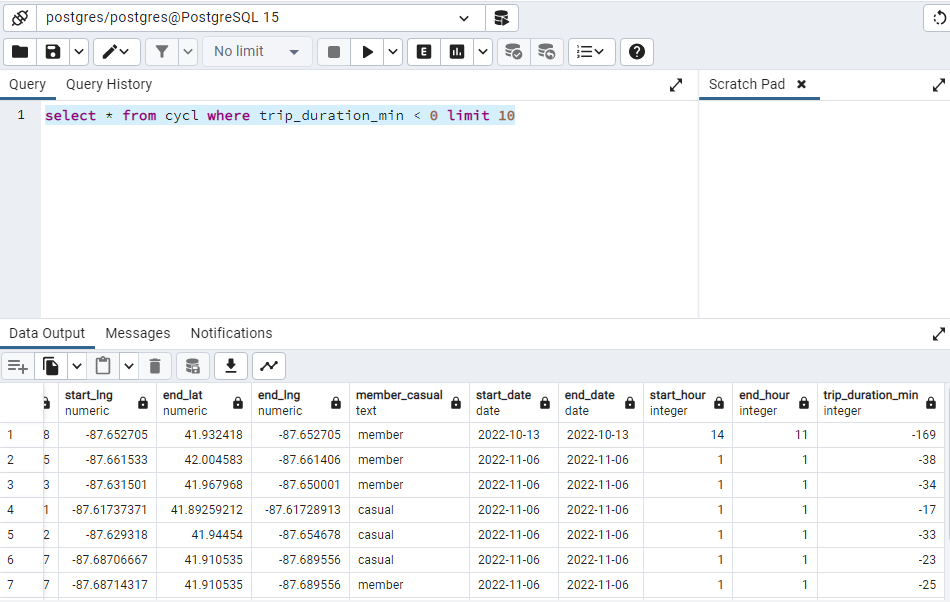
clean\_names\_start <- filter(data\_ride\_id\_removed,   
 data\_ride\_id\_removed$start\_station\_name != "")

repeating the same thing for end\_station\_name.

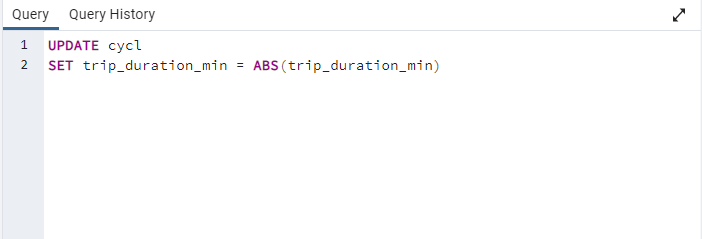
### **Using SQL**

I used SQL to clean the dates in our dataset and calculated the trip duration in minutes. .

**Notice that we have values that are less than 0 in the trip\_duration\_min column ?**



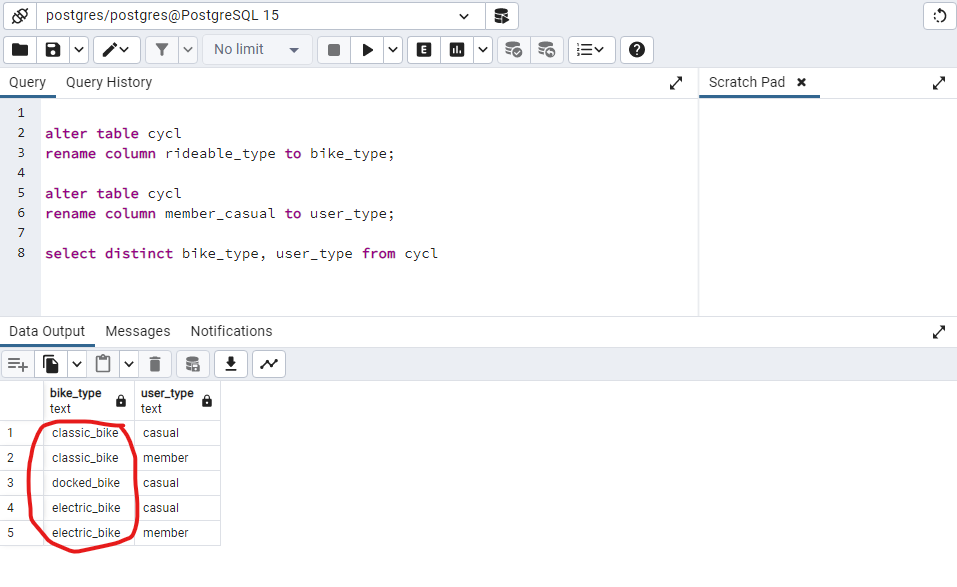
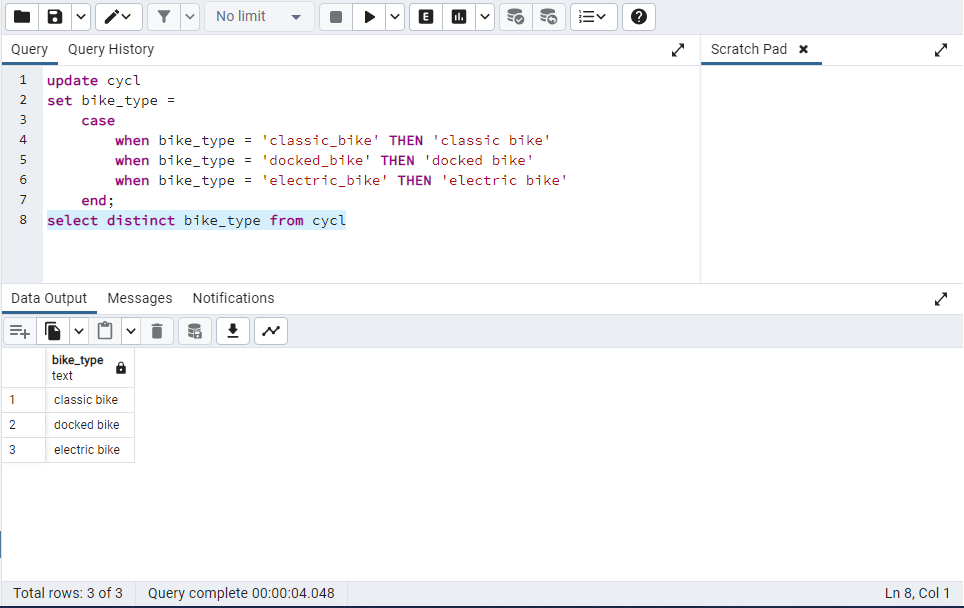
Let’s fix that !!

 Let’s now check.



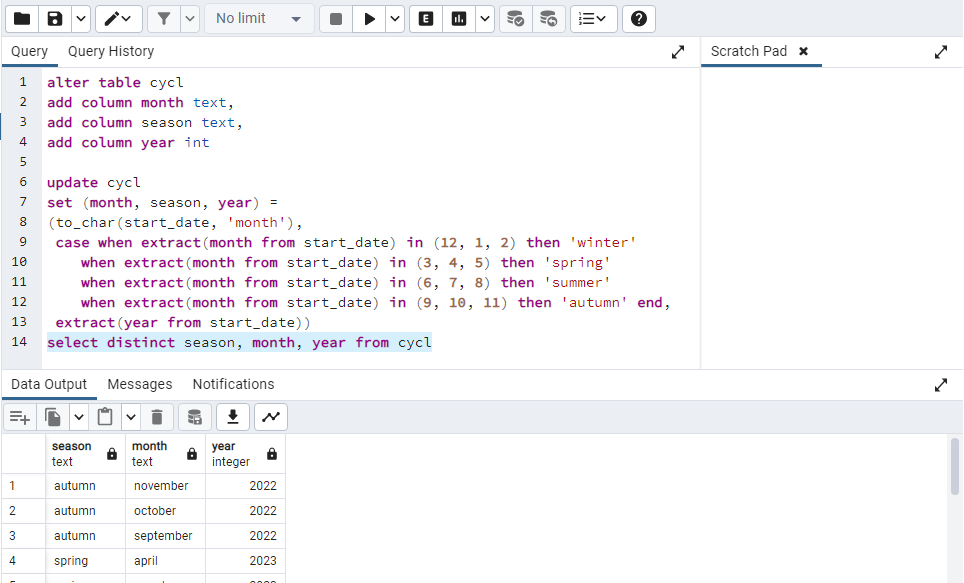
Now that our data is ready, let’s make one final modification before proceeding.

### Cleaning columns names

## Analysing and Visualizing the data

i added new columns to our data frame using **SQL**.



Importing our new data frame.

bikeshare <- read\_csv('C:/Users/isach/OneDrive/Bureau/projects/bikeshare.csv')

## Rows: 4437128 Columns: 20  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (8): bike\_type, start\_station\_name, start\_station\_id, end\_station\_name,...  
## dbl (8): start\_lat, start\_lng, end\_lat, end\_lng, start\_hour, end\_hour, trip...  
## dttm (2): started\_at, ended\_at  
## date (2): start\_date, end\_date  
##   
## 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.

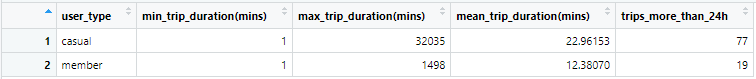
Taking a glimpse at our new data frame.

glimpse(bikeshare)

## Rows: 4,437,128  
## Columns: 20  
## $ bike\_type <chr> "classic bike", "electric bike", "electric bike", "…  
## $ started\_at <dttm> 2022-07-21 20:53:29, 2022-07-12 12:45:02, 2022-07-…  
## $ ended\_at <dttm> 2022-07-21 21:01:30, 2022-07-12 12:52:12, 2022-07-…  
## $ start\_station\_name <chr> "Clark St & Newport St", "Kingsbury St & Kinzie St"…  
## $ start\_station\_id <chr> "632", "KA1503000043", "KA1503000043", "KA150300004…  
## $ end\_station\_name <chr> "Clark St & Montrose Ave", "Halsted St & Willow St"…  
## $ end\_station\_id <chr> "KA1503000022", "TA1307000166", "21544", "21544", "…  
## $ start\_lat <dbl> 41.94454, 41.88911, 41.88906, 41.88918, 41.88918, 4…  
## $ start\_lng <dbl> -87.65468, -87.63827, -87.63855, -87.63851, -87.638…  
## $ end\_lat <dbl> 41.96159, 41.91387, 41.89662, 41.89662, 41.91387, 4…  
## $ end\_lng <dbl> -87.66604, -87.64875, -87.62858, -87.62858, -87.648…  
## $ user\_type <chr> "casual", "member", "casual", "casual", "member", "…  
## $ start\_date <date> 2022-07-21, 2022-07-12, 2022-07-31, 2022-07-17, 20…  
## $ end\_date <date> 2022-07-21, 2022-07-12, 2022-07-31, 2022-07-17, 20…  
## $ start\_hour <dbl> 21, 13, 17, 6, 7, 0, 10, 6, 19, 8, 20, 20, 19, 17, …  
## $ end\_hour <dbl> 22, 13, 17, 6, 7, 0, 10, 6, 19, 13, 20, 20, 19, 17,…  
## $ trip\_duration\_min <dbl> 8, 7, 9, 8, 12, 6, 27, 1, 1, 303, 1, 28, 15, 1, 5, …  
## $ month <chr> "july", "july", "july", "july", "july", "july", "ju…  
## $ season <chr> "summer", "summer", "summer", "summer", "summer", "…  
## $ year <dbl> 2022, 2022, 2022, 2022, 2022, 2022, 2022, 2022, 202…

### basic descriptive analysis

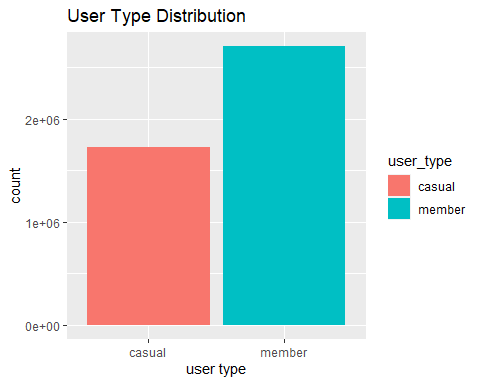
basic\_stats <- bikeshare %>%  
 select(user\_type, trip\_duration\_min) %>%  
 group\_by(user\_type) %>%  
 summarize(   
 'min\_trip\_duration\_mins' = min(trip\_duration\_min),  
 'max\_trip\_duration\_mins' = max(trip\_duration\_min),  
 'mean\_trip\_duration\_mins' = mean(trip\_duration\_min),  
 trips\_more\_than\_24h = sum(trip\_duration\_min >= 1440))



**N.B: casual users take longer trips than the annual members.**

#### Visualizing the user type distribution

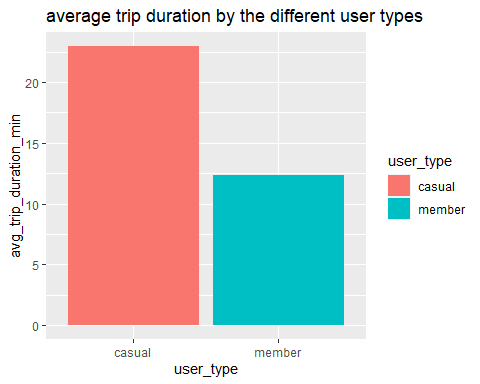
plot\_1 <- ggplot(data = bikeshare, mapping = aes(x=user\_type, fill = user\_type))+   
 geom\_bar()+  
 labs(title = 'User Type Distribution',  
 x = 'user type')  
plot\_1



**N.B: The number of annual members exceeds the number of casual users.**

#### Visualizing the average trip duration by user types

plot\_2 <- ggplot(data = basic\_stats)+  
 geom\_col(mapping = aes(x = user\_type, y= mean\_trip\_duration\_mins, fill = user\_type))+  
 labs(title = 'average trip duration by the different user types',  
 x = 'user\_type',  
 y = 'avg\_trip\_duration\_min')  
plot\_2



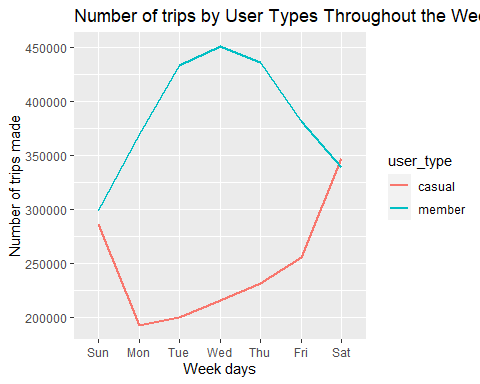
**N.B: casual users travel longer than the annual members**

#### Visualizing user type distribution across weekdays using a line chart

weekly\_count <- bikeshare %>%   
 group\_by(week\_days = wday(start\_date, label = TRUE), user\_type) %>%   
 summarize(number\_of\_trips = n(), .groups = "drop")  
  
plot\_3 <- ggplot(data = weekly\_count) +  
 geom\_line(mapping = aes(x = week\_days, y = number\_of\_trips, color = user\_type,  
 group = user\_type), size = 0.8) +  
 labs(title = 'Number of trips by User Types Throughout the Week',  
 x = 'Week days',  
 y = 'Number of trips made')

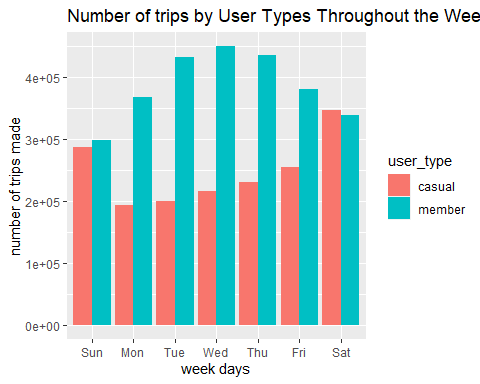
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

plot\_3



#### Visualizing user type distribution across weekdays

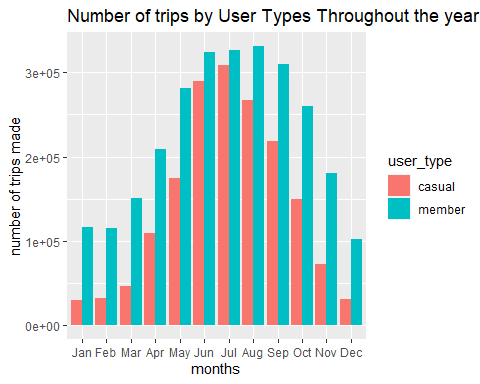
plot\_4 <- ggplot(data = weekly\_count)+  
 geom\_col(mapping = aes(x= week\_days, y= number\_of\_trips, fill = user\_type),   
 position = "dodge", size = 0.4)+  
 labs(title = 'Number of trips by User Types Throughout the Week',  
 x= 'week days',  
 y= 'number of trips made')  
plot\_4



**N.B:** - Weekdays: Annual members consistently have higher trip counts than casual users. - Friday: Casual user trips surpass annual members, indicating increased activity, while annual members decrease. - Saturday: Peak day for casual users with the highest trip counts, declining on Sunday. Annual members show the opposite trend.

#### Visualizing the monthly user type distribution

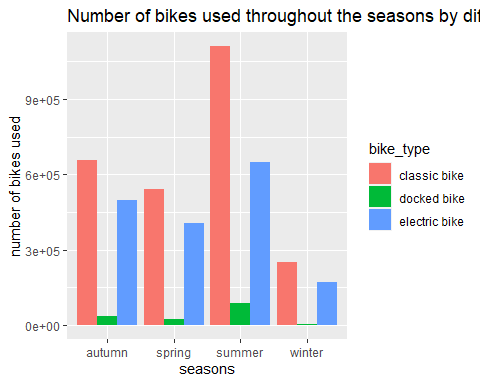
trips\_per\_month <- bikeshare %>%   
 group\_by(month = month(start\_date, label = TRUE), user\_type) %>%   
 summarize(number\_of\_trips = n(), .groups = 'drop')  
  
plot\_5 <- ggplot(data = trips\_per\_month)+  
 geom\_col(mapping = aes(x= month, y=number\_of\_trips, fill = user\_type),  
 position = 'dodge')+  
 labs(title = 'Number of trips by User Types Throughout the year',  
 x= 'months',  
 y= 'number of trips made')  
plot\_5



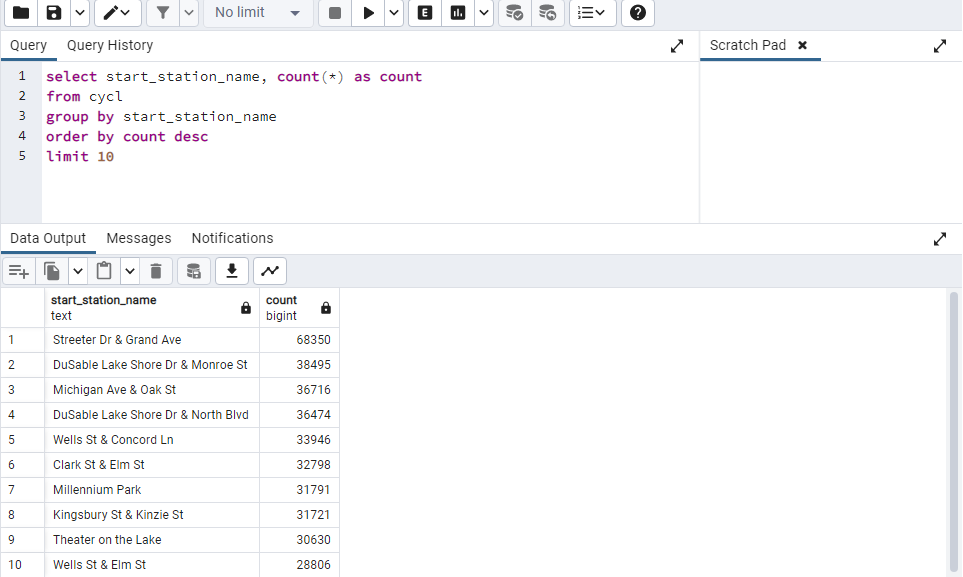
**N.B:**  Annual members uses Cyclistic services more frequently compared to casual users. However, an interesting pattern emerged during the spring months (March, April, May) where the number of casual members started to increase. This increase continued into the summer months, resulting in a similar level of bike rentals for both annual and casual members. However, as autumn arrived (September, October, November), the number of casual user rentals began to decrease once again. This indicates a seasonal fluctuation in the usage patterns, with casual users showing a notable increase during spring and summer, but declining in autumn and winter.

#### visualizing the diffrent types of bikes throughout the seasons

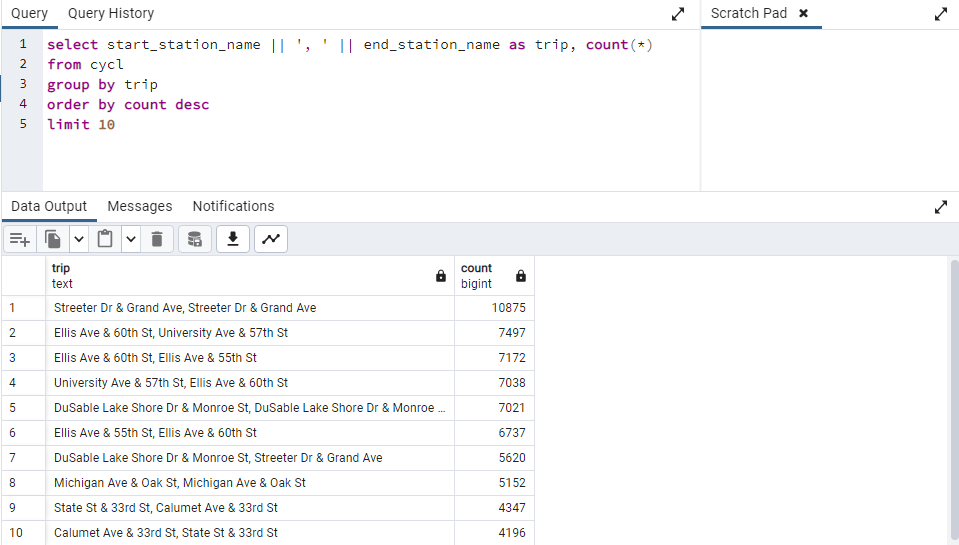
bikes <- bikeshare %>%  
 group\_by(season, bike\_type) %>%  
 summarise(number\_bikes\_used = n(), .groups = 'drop')  
  
plot\_6 <- ggplot(data = bikes)+  
 geom\_col(mapping = aes(x= season, y=number\_bikes\_used, fill = bike\_type),  
 position = 'dodge')+  
 labs(title = 'Number of bikes used throughout the seasons by diffrent bike types',  
 x= 'seasons',  
 y= 'number of bikes used')  
plot\_6



#### top 10 pick\_up locations

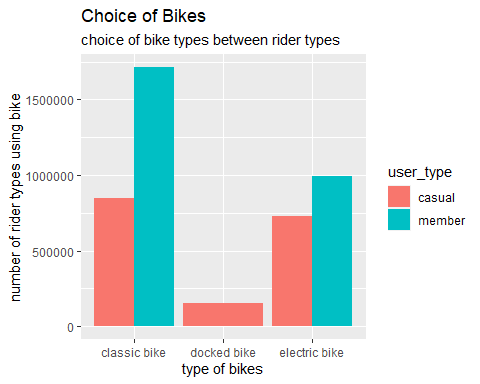


#### top 10 trips



#### Vsualizing the diffrent types of bikes by user types

bike\_types <- bikeshare %>%   
 select(user\_type, bike\_type) %>%   
 group\_by(bike\_type, user\_type) %>%   
 summarize(count\_of\_bikes = n(), .groups = 'drop')  
  
ggplot(data = bike\_types) +   
 geom\_col(mapping = aes(x = bike\_type, y = count\_of\_bikes,  
 fill = user\_type), position = "dodge") +   
 labs(title = "Choice of Bikes",   
 subtitle = "choice of bike types between rider types",   
 x = "type of bikes",   
 y = "number of rider types using bike")

 **N.B: casual riders uses all types of bikes. annual members prefer classic bikes. classic bikes are the most popular between the 3 types.**

## Conclusion

Our goal was to discover how do annual members and casual riders use Cyclistic bikes differently? By looking at the previous 12 months of Cyclistic trip data, we’re able to find out that :

* Annual members use Cyclistic bikes more often than casual users.
* casual users **travel longer** than the annual members.
* casual users are likely to use cyclistic services at **holidays** while annual members are likely to use cyclistic services **troughout the week**.
* During **spring** and **summer**, casual users start renting bikes in **larger numbers**, reaching a similar level as annual members. In **autumn**, casual user rentals **decrease** again.
* The usage of Cyclistic bikes varies with the seasons, with casual users renting **more bikes** during **warmer months** (spring, summer) but **fewer bikes** in **autumn** and **winter**.
* Classic bikes are **the most popular bikes** in all the seasons.
* Electric bikes are **most popular** in **summer**.

## Recommendations

Since our **gaol** is to **increase** the number of annual memberships and **covert** casual riders into annual members. We can:

* Collaborate with social media influencers to promote and use our bikes.
* Offer exclusive discounts, rewards, or incentives for annual members to encourage new sign-ups and retain existing members.
* Plan affordable bike trips to tourist spots, providing an exciting opportunity for annual members to explore and enjoy.
* Introduce a program where current annual members can get rewards or discounts by inviting casual riders to become annual members.
* Utilize targeted online advertising campaigns to reach potential annual members, focusing on the unique advantages and features of our bike-share service.