The Cyclist Case Study

This Cyclist case study is one of the three case studies given at the end of my Google Analytics Certification as Capstone Project.

This Google Certification included 7 modules before this project, which encompasses all the steps needed for an effective data analysis. The steps involved are:

- Ask
- Prepare
- Process
- Analyse
- Share
- Act

Tools Used:

- Excel
- SQL
- R programming
- Tableau

Background

Cyclistic, a bike-share program launched in 2016, has grown to 5,824 geotracked bicycles across 692 stations in Chicago. The program offers flexible pricing plans: single-ride passes, full-day passes, and annual memberships. Casual riders use single-ride or full-day passes, while annual members are more profitable for Cyclistic.

To boost growth, **Cyclistic aims to convert casual riders into annual members**. The marketing team, need to understand the differences between casual riders and annual members, reasons casual riders might buy memberships, and the impact of digital media on their tactics. Analysing historical bike trip data will help identify trends and inform their strategies.

Company's future success depends on maximizing the number of annual members.

Questions

Three questions will guide the future marketing program:

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?

After reading the objectives carefully and approaching the task with some basic questions in my mind, I began to understand that I just need to connect the little pieces in order to create a method to answer the questions.

The Data

The Data I'm working on is of First Party Type.

The data has been made available by Motivate International Inc. under this Licence (https://ride.divvybikes.com/data-license-agreement)

The data integrity was checked and deemed unbiased. Here is the source of data

https://divvy-tripdata.s3.amazonaws.com/index.html

This includes data from 2022 but in order to completely make my data ROCCC i.e. Reliable, original, comprehensive, current and cited, I will use data of 2022 from January to December, so I will work in a total of 12 csv files.

Data cleaning and manipulation

After downloading and unzipping all 12 zip files, I placed them in a temporary folder on my desktop. As a precaution, I made copies of each file. Then, I opened Google Sheets and began importing the files one by one, performing the following steps:

- I started by cleaning and manipulating the data using Google Sheets, ensuring it was stored correctly and free of inconsistencies.
- I removed all the duplicate entries and made sure I don't have any null values which can bias my analysis.
- I fixed the column width and aligned all the data so that it is easy to read and understand the data.
- Hide the unnecessary columns (start_station_name, end_station_name) to make my data look cleaner. I didn't delete those columns as I might need it for further analysis.
- Changed the format of started_at and ended_at columns to DATETIME.
- Created a new column ride_length for the duration of ride time by subtracting column ended_at and started_at (=D2-C2).
- Formatted the ride_length as time.

- Created a column named `day_of_week` to indicate the day each ride started, using the `WEEKDAY` function (`=WEEKDAY(C2,1)`).
- Now I was ready to perform some early level data analysis:
  - First I calculated the mean time of the ride_time by selecting whole ride_length column by using average function (`AVERAGE(N:N)`).

Note: 1 represents Sunday. We can change it as per our preference.

Since the datasets were too large that I had to do each step every time for each of the 12 files. Also, I wasn't able to conduct further analysis on the whole data set as one.

NOTE: As working with sheets were too time consuming, I had to switch to SQL to handle the datasets more conveniently.

Big Query: further data cleaning and manipulation via SQL

Since the datasets were too large to handle by sheets as some of them had 5 lakh rows. The best way out was SQL. So I shifted to **Google Big Query** where I created a new project and started working on.

- I created a new project for my analysis and created a "Cyclist Dataset".
- I uploaded all 12 files within the Cyclist Dataset and joined the tables Quarter wise so to perform analysis on different quarters.

Here's the code I used to combine 3 months in a quarter:

```
CREATE TABLE Google_analytics.quarter1 AS
SELECT * FROM `principal-rope-418521.Google_analytics.jan22`
UNION ALL
SELECT * FROM `principal-rope-418521.Google_analytics.feb22`
UNION ALL
SELECT * FROM `principal-rope-418521.Google_analytics.mar22`
```

NOTE: Ran into an error as **ride_length** column of feb22 was of **STRING** datatype and the rest were in **TIME** format. Corrected my mistake and I was good to go.

Ran some basic queries to check if all my data are in order in the new table "quarter1". Also tried to look if the data shows any pattern or inconsistencies.

- The query returned total 389956 rows which matches with the total number of rows from all 3 datasets. Also, I found inconsistency in 2 rows so I deleted those two to avoid any error in my analysis.

```
SELECT * FROM `Google_analytics.quarter1`
ORDER BY ride_length;
```

Press Alt+F1 for Accessibility

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

ow	start_lat	start_lng	end_lat	end_lng	member_casual	ride_length	day_of_week
1	41.93668845	-87.63682902	41.93668845	-87.63682902	casual	##########	
2	41.936313	-87.652522	41.93625348	-87.6526621	casual	##########	
3	41.91	-87.68	41.91	-87.68	casual	00:00	
4	41.91	-87.68	41.91	-87.68	member	00:00	

Results per page: 50 | 1 – 50 of 389956

- Also found that there are lots of inconsistencies in calculating ride_time which I might have not noticed in spreadsheets. Hence, I dropped the ride_length column first and re-calculated it using **time_diff()** function in SQL.

```
ALTER TABLE Google_analytics.quarter1
DROP COLUMN ride_length
```

```
ALTER TABLE principal-rope-418521.Google_analytics.quarter1
ADD COLUMN ride_length INT;
```

In order to use DML commands I had to upgrade my Big Query to billing account. So, I had to done it using spreadsheets only. I crossed checked all the entries and now I was good to go.

Below is the command which I had to use to update the column.

```
update Google_analytics.quarter1
set ride_length = time_diff(ended_at, started_at, minute)
```

- I again checked if I had any null values in my data?
  I found out that there are few null values present in my data so I used DELETE command as follows to remove all null values.

  ```
  DELETE FROM Google_analytics.quarter1
  WHERE column_names is NULL
  (column_names = ride_id || rideable_type || started_at, .....)
  ```

2022 Quarter 1 Exploratory Analysis

Now my data is clean and I am good to go with my analysis.

1. Checking for the distribution of members and casual members percent wise.

```
SELECT total_trips,
  total_member_trips,
  total_casual_trips,
  ROUND(total_member_trips/total_trips,4)*100 AS member_pert,
  ROUND(total_casual_trips/total_trips,4)*100 AS casual_pert
FROM(
  SELECT
    count(ride_id) as total_trips,
    countif(member_casual = 'member') AS total_member_trips,
    countif(member_casual = 'casual') AS total_casual_trips
  FROM principal-rope-418521.Google_analytics.quarter1

)
```

Processing location: US

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	total_trips	total_member_trips	total_casual_trips	member_pert	casual_pert
1	385289	290384	94905	75.37	24.63

- Total Membership members : 290384
- Total Casual Members : 94905
- Membership Percentage : 75.37 %
- Casual Member Percentage : 24.63 %

2. Checking for the max, min and average ride length membership wise.

```
SELECT avg(time_diff(ride_length,TIME'00:00:00',second)) AS total_avg_ride_length,
  (SELECT avg(time_diff(ride_length,TIME'00:00:00',second))
  FROM principal-rope-418521.Google_analytics.quarter1
  WHERE member_casual = 'member') AS avg_member_ride_length,
  (SELECT avg(time_diff(ride_length,TIME'00:00:00',second))
  FROM principal-rope-418521.Google_analytics.quarter1
  WHERE member_casual = 'casual') AS avg_casual_ride_length
FROM principal-rope-418521.Google_analytics.quarter1
```

Processing location: US

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

ow	total_avg_ride_length	avg_member_ride_le	avg_casual_ride_leng
1	871.1515252187...	667.7022838723...	1493.651967757...

Here, we can clearly see the average ride length of casual riders is around 826 seconds (i.e. 13.7 minutes) more than the member riders.

An interesting question arises why casual riders are riding more than member riders?

- Are there any outliers affecting the results?
- Are we missing something in our analysis?

Checking for any outliers…

```
SELECT
member_casual,
MAX(ride_length) AS ride_length_MAX
FROM principal-rope-418521.Google_analytics.quarter1
GROUP BY
member_casual
ORDER BY
ride_length_MAX DESC
LIMIT 10
```

Processing location: US

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	member_casual	ride_length_MAX
1	casual	23:59:00
2	member	23:56:00

Similarly, we can calculate minimum using following functions: **Min(ride_length)**

Findling: we found that maximum ride length is from casual membership. But not much that it can influence the results. There must be other factors.

Median ride length per day:

Median ride length for annual members:

median_ride_length	member_casual	day_of_week
00:08:29	member	1
00:08:13	member	7
00:07:54	member	2
00:07:52	member	4
00:07:36	member	6
00:07:35	member	3
00:07:32	member	5

Median ride_length for casual riders

Median_ride_length	member_casual	day_of_week
00:15:37	casual	1
00:14:22	casual	7
00:14:12	casual	2
00:13:08	casual	6
00:12:34	casual	4
00:11:02	casual	5
00:10:35	casual	3

***SUNDAY =1 and so on…**

Very interesting! The median ride length for casual riders on the top five days (SUN, SAT, MON, TUE, WED) is nearly double the amount for annual members on their top five days (SAT, SUN, MON, TUE, WED).

Total rides per day:

Let's find out total rides per day, which might tell us something.

```
SELECT day_of_week, count(distinct ride_id) AS total_rides,
SUM(CASE WHEN member_casual = 'member' THEN 1 ELSE 0 END) AS member_trips,
SUM(CASE WHEN member_casual = 'casual' THEN 1 ELSE 0 END) AS casual_trips
FROM principal-rope-418521.Google_analytics.quarter1
GROUP BY 1
ORDER BY day_of_week
```

Processing location: US

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	total_rides	member_trips	casual_trips
1	1	50161	32303	17858
2	2	63597	47954	15643
3	3	61183	50564	10621
4	4	64428	50207	14222
5	5	56733	45184	11550
6	6	42059	33580	8479
7	7	47124	30592	16532

Most Popular Stations:

```
SELECT start_station_name, count(distinct ride_id) AS total,
SUM(CASE WHEN member_casual = 'member' THEN 1 ELSE 0 END) AS member_trips,
SUM(CASE WHEN member_casual = 'casual' THEN 1 ELSE 0 END) AS casual_trips
FROM principal-rope-418521.Google_analytics.quarter1
GROUP BY 1
ORDER BY total desc
```

Processing location: US

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	start_station_name	total	member_trips	casual_trips
1	Kingsbury St & Kinzie St	4130	3624	506
2	Streeter Dr & Grand Ave	3630	1009	2621
3	Clark St & Elm St	3299	2635	664
4	Clinton St & Madison St	3169	2710	459
5	University Ave & 57th St	3153	2733	420
6	Ellis Ave & 60th St	3131	2767	364
7	Clinton St & Washington Blvd	3079	2752	327
8	Wells St & Concord Ln	2846	2157	689
9	Wells St & Elm St	2791	2073	718

We can see that the most popular station among riders is Kingsbury where member riders is far more than casual riders. To look more into the data, we can sort this as per member_trips or casual_trips using ORDER BY function.

CASUAL DESC and **MEMBER DESC** in two separate queries, we can compare the top ten start stations for both:

```
SELECT start_station_name,
  SUM(CASE WHEN member_casual = 'member' THEN 1 ELSE 0 END) AS member_trips,
FROM principal-rope-418521.Google_analytics.quarter1
GROUP BY 1
ORDER BY member_trips desc
LIMIT 10
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

Row	start_station_name	member_trips
1	Kingsbury St & Kinzie St	3624
2	Ellis Ave & 60th St	2767
3	Clinton St & Washington Blvd	2752
4	University Ave & 57th St	2733
5	Clinton St & Madison St	2710
6	Clark St & Elm St	2635
7	Wells St & Concord Ln	2157
8	Dearborn St & Erie St	2140
9	Clinton St & Jackson Blvd	2121
10	Wells St & Elm St	2073

```
SELECT start_station_name,
  SUM(CASE WHEN member_casual = 'casual' THEN 1 ELSE 0 END) AS casual_trips,
FROM principal-rope-418521.Google_analytics.quarter1
GROUP BY 1
ORDER BY casual_trips desc
LIMIT 10
```

Query results

JOB INFORMATION | RESULTS | CHART | JSON | EXECUTION DETAILS

Row	start_station_name	casual_trips
1	Streeter Dr & Grand Ave	2621
2	DuSable Lake Shore Dr & Monr...	1639
3	Millennium Park	1335
4	Shedd Aquarium	1129
5	Michigan Ave & Oak St	824
6	Wells St & Elm St	718
7	Wells St & Concord Ln	689
8	DuSable Lake Shore Dr & North...	682
9	Clark St & Elm St	664
10	Indiana Ave & Roosevelt Rd	656

So, there are some stations which belong to top 10 of both the list, these are the stations preferred by both casual and annual members. **A full-fledged marketing campaign can be done near the stations crowded by casual members.** Same analysis can be performed on QUARTER2, QUARTER3 AND QUARTER4 just by slightly modifying the code (only have to replace the file names and we're done with the basic analysis. Numbers and text can only tell you this much, to know your data more, you have to use visualisation tools. I have two Visualisation tools: Tableau and R (it's more than just a visualisation tool). Tableau is simple to use but connecting data from Big Query to Tableau requires Tableau Desktop which is not free. That's why I'll go with R here.

Visualisation with R

```
library(tidyverse)
library(readr)
library(ggplot2)
```

```
trip_quarter1 <- read_csv("C:\Users\pushp\Downloads\quarter1.csv")
```

Error: '\U' used without hex digits in character string (<input>:1:31)

- Not sure why I was getting this error, but R community came to the rescue and I was good to go.
SPOT THE DIFFERENCE

```
trip_quarter1 <- read_csv("C:/Users/pushp/Downloads/quarter1.csv")
view(trip_quarter1)
```

This returns the dataset "quarter_1".
Similarly, we can do this for all other quarters and then bind them together.

Binding Datasets in R:

```
trip_2022 <- rbind(trip_quarter1, trip_quarter2, trip_quarter3, trip_quarter4)
view(trip_2022)
```

Now we have all the 4 quarters binded together.
We can see all the rows and columns are in order. Now, we can move forward with our visualization.

Total Rides taken by Members & Casual Riders

Code:

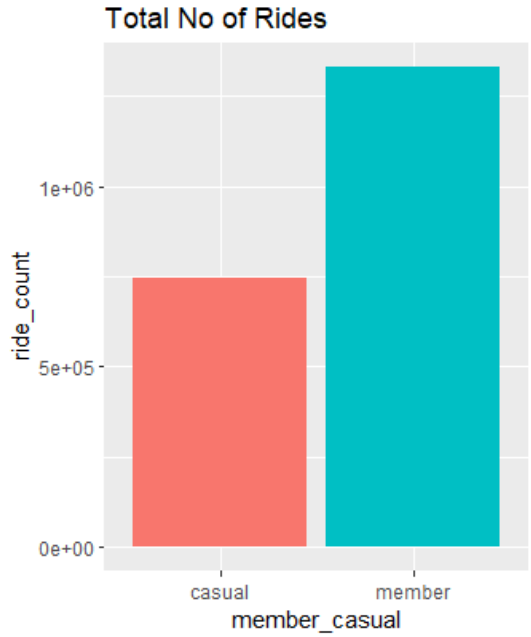
```
trip_2022 %>%
+ group_by(member_casual) %>%
+ summarise(ride_count=length(ride_id)) %>%
+ ggplot()+geom_col(mapping = aes(x=member_casual,y=ride_count,
fill=member_casual,),show.legend = "false") +
+ labs(title = "Total No of Rides")
```

Comparison of Total rides with the Type of Ride

code:

```
trip_2022 %>%
+ group_by(member_casual, rideable_type) %>%
+ summarise(number_of_rides = n(), .groups = "drop") %>%
+ ggplot() + geom_col(mapping = aes(x = rideable_type, y = number_of_rides,
fill = member_casual), show.legend = TRUE) +
+ labs(title = "Total no. of Rides vs. Ride Type")
```

output:



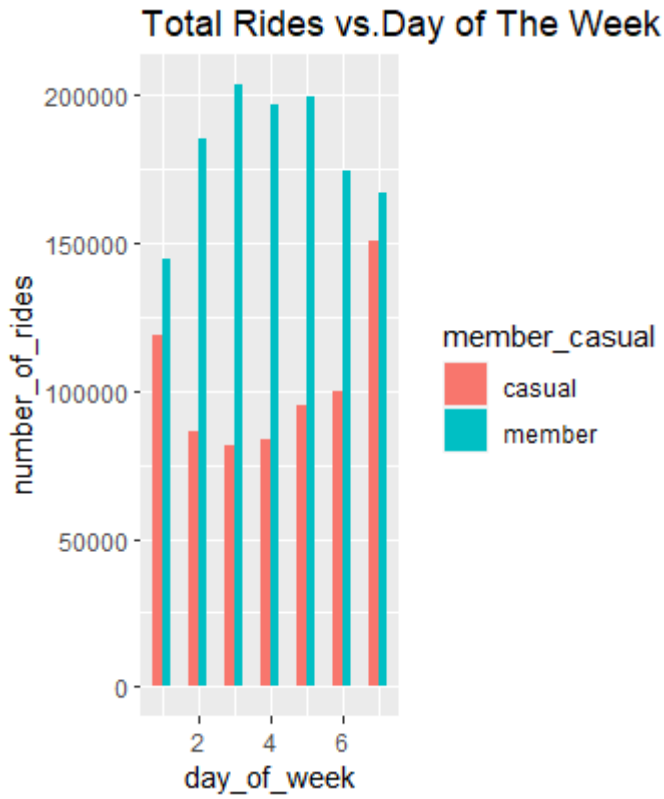
We see annual members travelled almost twice as that of annual members, This confirms that annual members contribute more to the revenue generation as compared to casual members.

Days of the Week with No. of Rides taken by Riders

Code:

```
trip_2022 %>%
+ group_by(member_casual, day_of_week) %>%
+ summarise(number_of_rides=n(), .groups = "drop") %>%
+ ggplot(aes(x = day_of_week, y = number_of_rides, fill = member_casual)) +
+ labs(title = "Total Rides vs.Day of The Week") +
+ geom_col(width = 0.5, position = position_dodge(width = 0.5)) +
+ scale_y_continuous(labels = function(x) format(x,scientific = FALSE))
```

Output:



An intriguing observation emerges from the analysis: casual members demonstrate greater travel activity on weekends, whereas annual members exhibit higher travel volumes on weekdays, potentially indicating a weekday commuting pattern, especially for work-related purposes.

Average Ride Length by Day of the Week

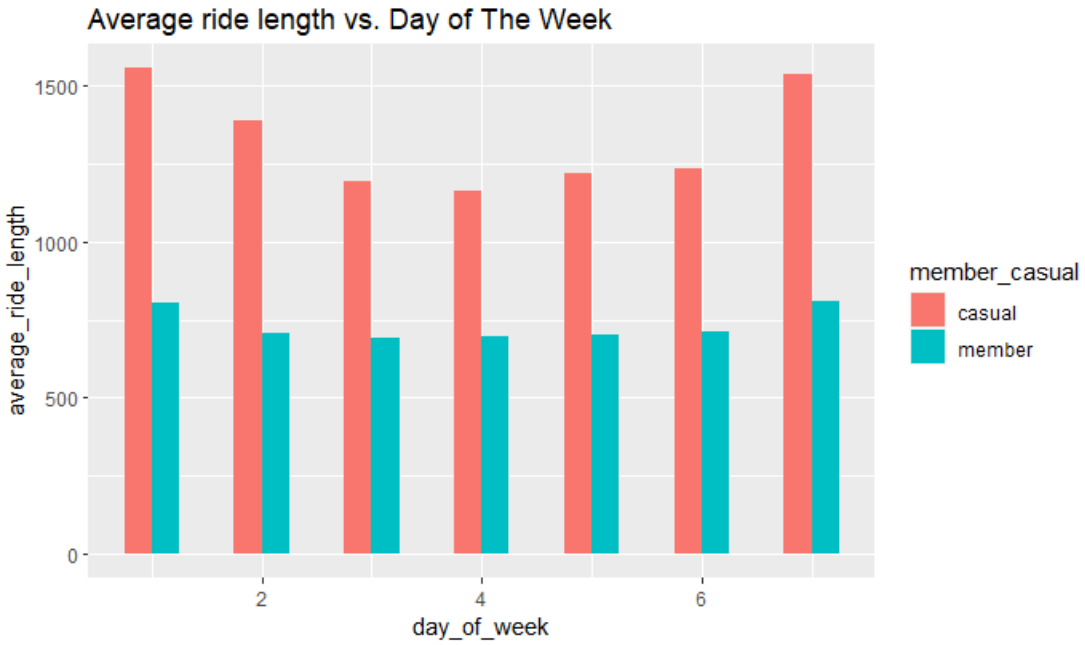
To plot average ride by day of the week, we first need to find average trip duration,

```
trip_2022 %>%
+ group_by(member_casual) %>%
summarise(average_ride_length=mean(ride_length),median_ride_length=median(rid
e_length),max_ride_length=max(ride_length),min_ride_length=min(ride_length))
```

Code:

```
trip_2022 %>%
+ group_by(member_casual,day_of_week) %>%
+ summarise(average_ride_length=mean(ride_length), .groups = "drop") %>%
+ ggplot(aes(x = day_of_week, y = average_ride_length, fill =
member_casual)) +
+ geom_col(width = 0.5, position = position_dodge(width = 0.5)) +
+ labs(title = "Average ride length vs. Day of The Week")
```

Output:



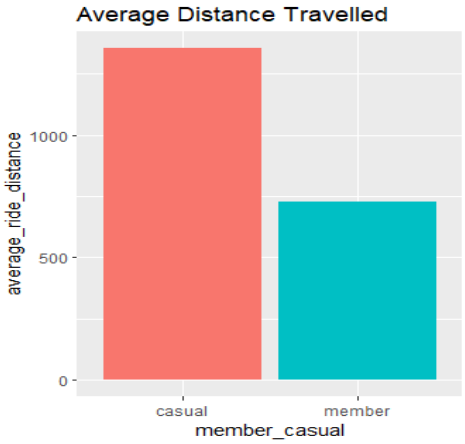
The graph illustrates a consistent trend in the data, revealing that Saturdays and Sundays exhibit the highest levels of travel activity.

Comparing Casual and Member Rides by Distance

Code:

```
trip_2022 %>%
+ group_by(member_casual) %>%
+ summarise(average_ride_distance = mean(ride_length)) %>%
+ ggplot() + geom_col(mapping = aes(x = member_casual, y =
average_ride_distance, fill = member_casual), show.legend = FALSE) +
+ labs(title = "Mean Distance Travelled")
```

Output:



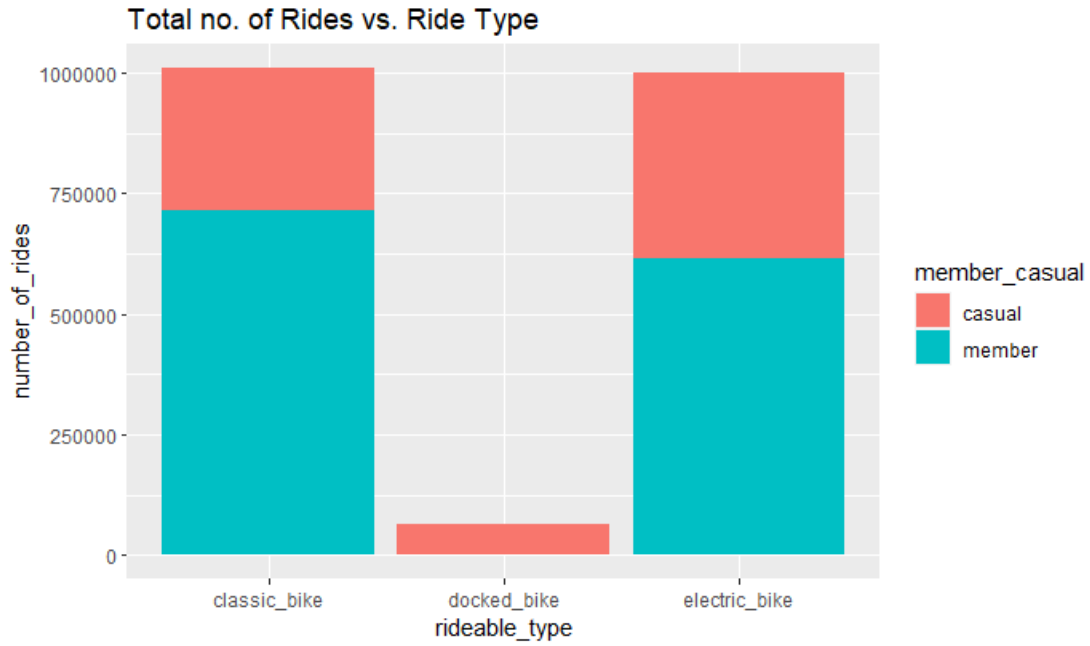
Undoubtedly, annual members have a higher ride count, but it is noteworthy that casual members travel twice the distance compared to annual members. We can develop targeted marketing campaigns to attract and encourage more casual members, emphasizing the benefit of longer rides and highlighting the potential savings in terms of distance covered. Promote features such as scenic routes, group ride opportunities, or challenges that appeal to casual riders' preference for longer distances. The company can also consider introducing pricing plans or discounts that cater specifically to riders who cover longer distances, incentivizing them to become regular users. This can help increase engagement and loyalty among casual riders.

Comparison of Total rides with the Type of Ride

Code:

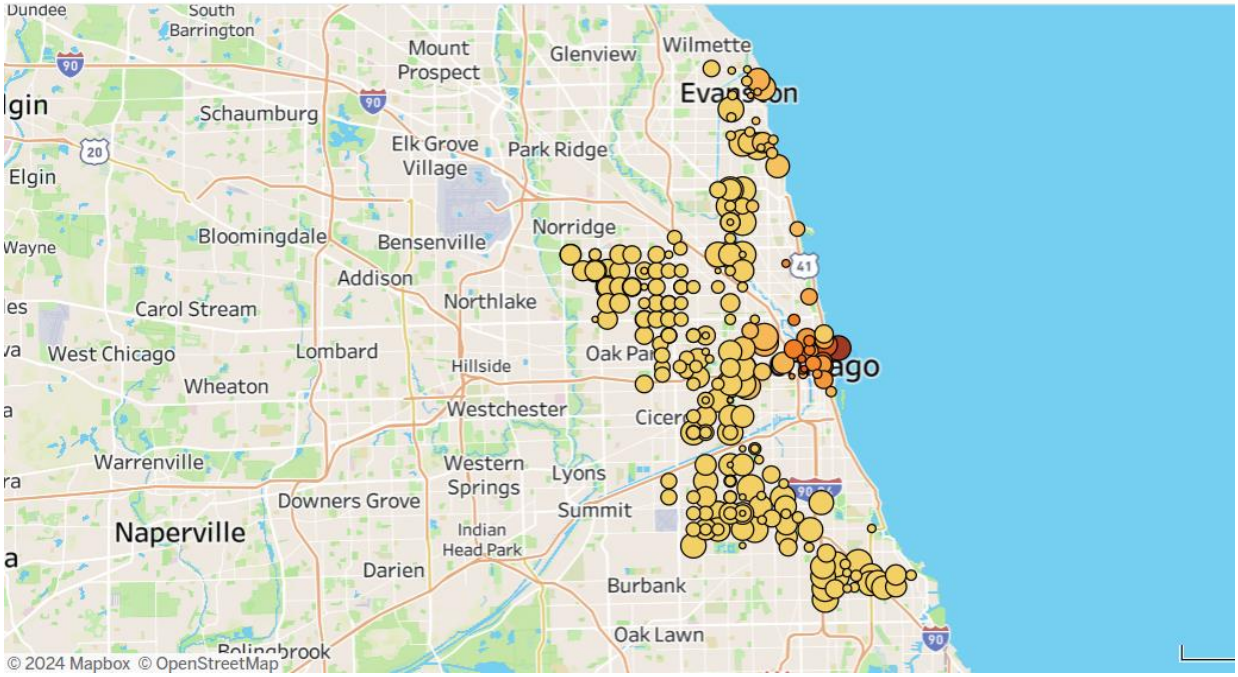
```
trip_2022 %>%
+ group_by(member_casual, rideable_type) %>%
+ summarise(number_of_rides = n(), .groups = "drop") %>%
+ ggplot() + geom_col(mapping = aes(x = rideable_type, y = number_of_rides,
fill = member_casual), show.legend = TRUE) +
+ labs(title = "Total no. of Rides vs. Ride Type")
```

Output:

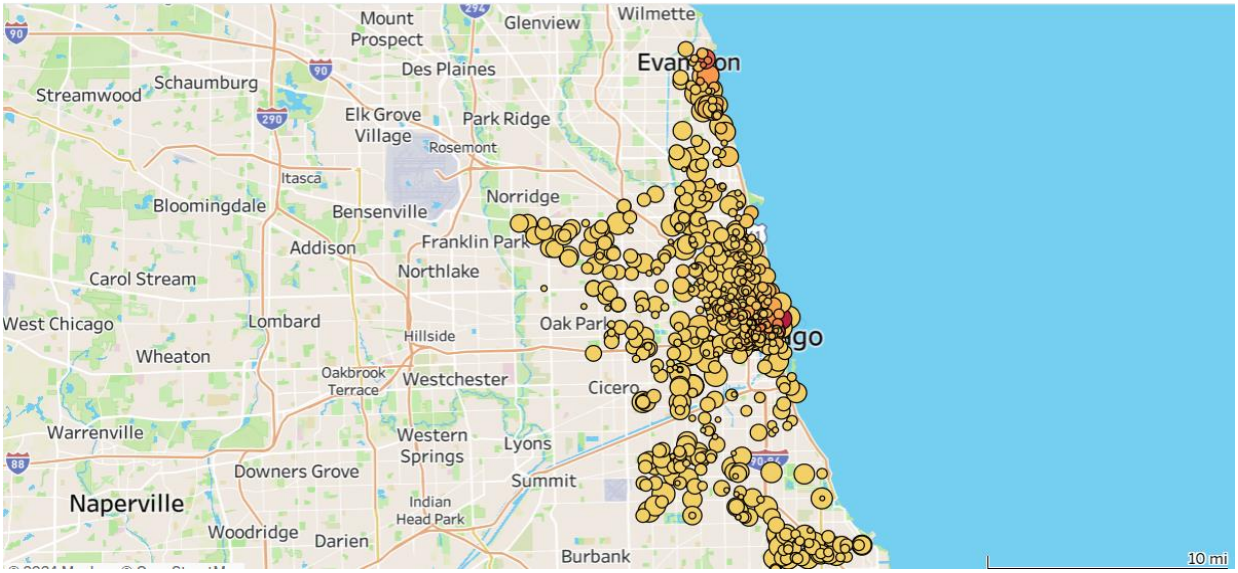


We see the share of electric bikes is almost the same as that of classic bike. This shows there is a growing preference for electric bikes among riders, the company can consider expanding its electric bike fleet. This can help attract and retain customers, particularly those who prefer the convenience and benefits of electric bikes. Also, The data on bike usage can inform targeted marketing campaigns and promotions. For example, if there is a need to promote electric bikes more, the company can highlight their features and benefits in marketing materials or offer incentives to encourage more riders to try them.

Most Popular Starting Station



Most Popular Ending Station



This shows the most popular start and end stations. The popularity of specific stations can inform targeted marketing and promotional efforts. The company can focus its marketing campaigns on these stations to attract more riders and raise awareness about the services offered. Promotions such as discounts, loyalty programs, or exclusive offers can be tailored to these popular stations to encourage ridership and create a positive association with the brand.

By leveraging the knowledge of the most popular stations, Cyclitic can optimise bike distribution, improve infrastructure, target marketing efforts, form strategic partnerships, and evaluate performance. These actions contribute to enhancing the overall user experience, attracting more riders, and ultimately increasing the company's success in the bike-sharing market.

Top Recommendation

- **Offer discounted annual membership plans or incentives specifically designed to encourage casual riders to become annual members.**
- **Highlight the long-term cost savings and benefits of regular usage.**
- **Develop targeted marketing campaigns at the most popular stations showcasing the advantages of annual membership, such as access to exclusive features, priority bike availability, or special events.**
- **Offer trial periods or short-term discounted memberships to allow casual riders to experience the benefits of being an annual member before committing to a full-term membership.**
- **Provide personalised recommendations or tailored rewards based on the riding patterns of casual riders, demonstrating the value they can gain as annual members.**