Google_Data_Analytics_Capstone_Project_Cyclistic-bike-share-analysis

Ishan Perera

2024-07-08

Background

Scenario

I am 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, our team wants to understand how casual riders and annual members use Cyclistic bikes differently. From these insights, our team will design a new marketing strategy to convert casual riders into annual members.

About the company

In 2016, Cyclistic launched a successful bike-share offering. Since then, the program has grown to a fleet of 5,824 bicycles that are geotracked and locked into a network of 692 stations across Chicago. The bikes can be unlocked from one station and returned to any other station in the system anytime.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes, and annual memberships. 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. Although the pricing flexibility helps Cyclistic attract more customers, The director 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, the director believes there is a solid opportunity to convert casual riders into members. She notes that casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.

Goal:

Design marketing strategies aimed at converting casual riders into annual members

Business Question:

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

Data Preparation and Process

I used R to analyze the data because it could handle all of the information quicker than Excel. First I Downloaded data and store it appropriately in my desktop and follow the below steps.

Data source: https://divvy-tripdata.s3.amazonaws.com/index.html/ (https://divvy-tripdata.s3.amazonaws.com/index.html/)

1. Load the libraries as required

```
library(tidyverse) #calculations
library(lubridate) #dates
library(ggplot2) # data visualization
```

2. Uploaded all of the original data into R studio. (4 CSV Files. From 2024 Jan to 2024 April)

```
jan24_df <- read.csv("202401-divvy-tripdata.csv")
feb24_df <- read.csv("202402-divvy-tripdata.csv")
mar24_df <- read.csv("202403-divvy-tripdata.csv")
apr24_df <- read.csv("202404-divvy-tripdata.csv")
```

3. Merged the 4 dataframes into single data frame

```
Cyclistic_df <- rbind(jan24_df, feb24_df, mar24_df, apr24_df)
```

Get a really quick idea of what's in this data set

```
glimpse(Cyclistic_df) # before clean the data
```

```
## Rows: 1,084,749
## Columns: 13
## $ ride_id            <chr> "C1D650626C8C899A", "EECD38BDB25BFCB0", "F4A9CE7806…
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", …
## $ started_at         <chr> "2024-01-12 15:30:27", "2024-01-08 15:45:46", "2024…
## $ ended_at           <chr> "2024-01-12 15:37:59", "2024-01-08 15:52:59", "2024…
## $ start_station_name <chr> "Wells St & Elm St", "Wells St & Elm St", "Wells St…
## $ start_station_id   <chr> "KA1504000135", "KA1504000135", "KA1504000135", "TA…
## $ end_station_name   <chr> "Kingsbury St & Kinzie St", "Kingsbury St & Kinzie …
## $ end_station_id     <chr> "KA1503000043", "KA1503000043", "KA1503000043", "13…
## $ start_lat          <dbl> 41.90327, 41.90294, 41.90295, 41.88430, 41.94880, 4…
## $ start_lng          <dbl> -87.63474, -87.63444, -87.63447, -87.63396, -87.675…
## $ end_lat            <dbl> 41.88918, 41.88918, 41.88918, 41.92182, 41.88918, 4…
## $ end_lng            <dbl> -87.63851, -87.63851, -87.63851, -87.64414, -87.638…
## $ member_casual      <chr> "member", "member", "member", "member", "member", "…
```

We can see the in this data set 1,084,749 rows and 13 columns included before clean up the data.

Create new data frame to contain new columns

```
Cyclistic_data <- Cyclistic_df
```

Preview of the column names and the first few rows of this data set

```
head(Cyclistic_data)
```

```
##            ride_id rideable_type          started_at            ended_at
## 1 C1D650626C8C899A electric_bike 2024-01-12 15:30:27 2024-01-12 15:37:59
## 2 EECD38BDB25BFCB0 electric_bike 2024-01-08 15:45:46 2024-01-08 15:52:59
## 3 F4A9CE78061F17F7 electric_bike 2024-01-27 12:27:19 2024-01-27 12:35:19
## 4 0A0D9E15EE50B171  classic_bike 2024-01-29 16:26:17 2024-01-29 16:56:06
## 5 33FFC9805E3EFF9A  classic_bike 2024-01-31 05:43:23 2024-01-31 06:09:35
## 6 C96080812CD285C5  classic_bike 2024-01-07 11:21:24 2024-01-07 11:30:03
##           start_station_name start_station_id          end_station_name
## 1          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 2          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 3          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 4     Wells St & Randolph St     TA1305000030 Larrabee St & Webster Ave
## 5 Lincoln Ave & Waveland Ave            13253  Kingsbury St & Kinzie St
## 6          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
##   end_station_id start_lat start_lng  end_lat   end_lng member_casual
## 1   KA1503000043  41.90327 -87.63474 41.88918 -87.63851        member
## 2   KA1503000043  41.90294 -87.63444 41.88918 -87.63851        member
## 3   KA1503000043  41.90295 -87.63447 41.88918 -87.63851        member
## 4          13193  41.88430 -87.63396 41.92182 -87.64414        member
## 5   KA1503000043  41.94880 -87.67528 41.88918 -87.63851        member
## 6   KA1503000043  41.90322 -87.63432 41.88918 -87.63851        member
```

4. Add two columns data set (ride_length and day_of_Week)

- Calculate ride length by subtracting ended_at time from started_at time and converted it to minutes

```
Cyclistic_data$ride_length <- difftime(Cyclistic_df$ended_at, Cyclistic_df$started_at, units = "mins")
# subtracting ended_at time from started_at time and converted it to minutes
```

- Calculate the day of the week

```
Cyclistic_data$date <- as.Date(Cyclistic_data$started_at) # convert date
Cyclistic_data$day_of_week <- wday(Cyclistic_df$started_at) # return day of the week as a decimal number
Cyclistic_data$day_of_week <- format(as.Date(Cyclistic_data$date), "%A")
```

Preview of the column names and the first few rows of this data set

```
head(Cyclistic_data)
```

```
##            ride_id rideable_type          started_at            ended_at
## 1 C1D650626C8C899A electric_bike 2024-01-12 15:30:27 2024-01-12 15:37:59
## 2 EECD38BDB25BFCB0 electric_bike 2024-01-08 15:45:46 2024-01-08 15:52:59
## 3 F4A9CE78061F17F7 electric_bike 2024-01-27 12:27:19 2024-01-27 12:35:19
## 4 0A0D9E15EE50B171  classic_bike 2024-01-29 16:26:17 2024-01-29 16:56:06
## 5 33FFC9805E3EFF9A  classic_bike 2024-01-31 05:43:23 2024-01-31 06:09:35
## 6 C96080812CD285C5  classic_bike 2024-01-07 11:21:24 2024-01-07 11:30:03
##           start_station_name start_station_id          end_station_name
## 1          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 2          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 3          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
## 4     Wells St & Randolph St     TA1305000030 Larrabee St & Webster Ave
## 5 Lincoln Ave & Waveland Ave            13253  Kingsbury St & Kinzie St
## 6          Wells St & Elm St     KA1504000135  Kingsbury St & Kinzie St
##   end_station_id start_lat start_lng  end_lat   end_lng member_casual
## 1   KA1503000043  41.90327 -87.63474 41.88918 -87.63851        member
## 2   KA1503000043  41.90294 -87.63444 41.88918 -87.63851        member
## 3   KA1503000043  41.90295 -87.63447 41.88918 -87.63851        member
## 4          13193  41.88430 -87.63396 41.92182 -87.64414        member
## 5   KA1503000043  41.94880 -87.67528 41.88918 -87.63851        member
## 6   KA1503000043  41.90322 -87.63432 41.88918 -87.63851        member
##      ride_length       date day_of_week
## 1  7.533333 mins 2024-01-12      Friday
## 2  7.216667 mins 2024-01-08      Monday
## 3  8.000000 mins 2024-01-27    Saturday
## 4 29.816667 mins 2024-01-29      Monday
## 5 26.200000 mins 2024-01-31   Wednesday
## 6  8.650000 mins 2024-01-07      Sunday
```

5. Clean the Data

```
Cyclistic_data <- na.omit(Cyclistic_data) #remove rows with NA values
Cyclistic_data <- distinct(Cyclistic_data) #remove duplicate rows
Cyclistic_data <- Cyclistic_data[!(Cyclistic_data$ride_length <=0),] #remove where ride_length is 0 or negative
```

Get a really quick idea of what’s in this data set after the clean up

```
glimpse(Cyclistic_data) # After clean the data
```

```
## Rows: 1,082,849
## Columns: 16
## $ ride_id            <chr> "C1D650626C8C899A", "EECD38BDB25BFCB0", "F4A9CE7806…
## $ rideable_type      <chr> "electric_bike", "electric_bike", "electric_bike", …
## $ started_at         <chr> "2024-01-12 15:30:27", "2024-01-08 15:45:46", "2024…
## $ ended_at           <chr> "2024-01-12 15:37:59", "2024-01-08 15:52:59", "2024…
## $ start_station_name <chr> "Wells St & Elm St", "Wells St & Elm St", "Wells St…
## $ start_station_id   <chr> "KA1504000135", "KA1504000135", "KA1504000135", "TA…
## $ end_station_name   <chr> "Kingsbury St & Kinzie St", "Kingsbury St & Kinzie …
## $ end_station_id     <chr> "KA1503000043", "KA1503000043", "KA1503000043", "13…
## $ start_lat          <dbl> 41.90327, 41.90294, 41.90295, 41.88430, 41.94880, 4…
## $ start_lng          <dbl> -87.63474, -87.63444, -87.63447, -87.63396, -87.675…
## $ end_lat            <dbl> 41.88918, 41.88918, 41.88918, 41.92182, 41.88918, 4…
## $ end_lng            <dbl> -87.63851, -87.63851, -87.63851, -87.64414, -87.638…
## $ member_casual      <chr> "member", "member", "member", "member", "member", "…
## $ ride_length        <drtn> 7.533333 mins, 7.216667 mins, 8.000000 mins, 29.81…
## $ date               <date> 2024-01-12, 2024-01-08, 2024-01-27, 2024-01-29, 20…
## $ day_of_week        <chr> "Friday", "Monday", "Saturday", "Monday", "Wednesda…
```

After the clean up We can see 1,082,849 rows (before clean up 1,084,749 rows) and 16 columns included

Data Analyze and Visualization

Total Rides (From 2024 Jan to 2024 Apr)

1. Total number of rides

```
nrow(Cyclistic_data)
```

```
## [1] 1082849
```

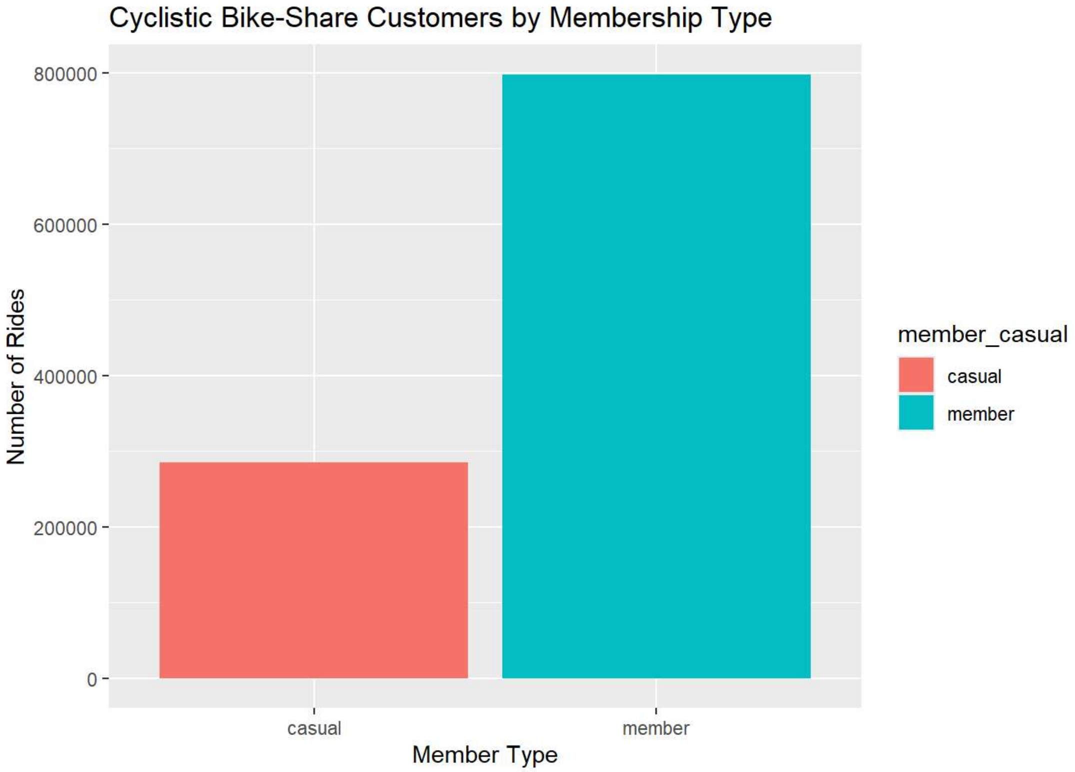
2. Total number of rides by member type

```
Cyclistic_data %>%
  group_by(member_casual) %>%
  count(member_casual)
```

```
## # A tibble: 2 × 2
## # Groups:   member_casual [2]
##   member_casual      n
##   <chr>          <int>
## 1 casual        284880
## 2 member        797969
```

```
options(scipen = 999)
ggplot(data = Cyclistic_data) +
  geom_bar(mapping = aes(x = member_casual, fill = member_casual, stat = "identity")) +
  labs(x = "Member Type", y = "Number of Rides", title = "Cyclistic Bike-Share Customers by Membership Type")
```

```
## Warning in geom_bar(mapping = aes(x = member_casual, fill = member_casual, :
## Ignoring unknown aesthetics: stat
```



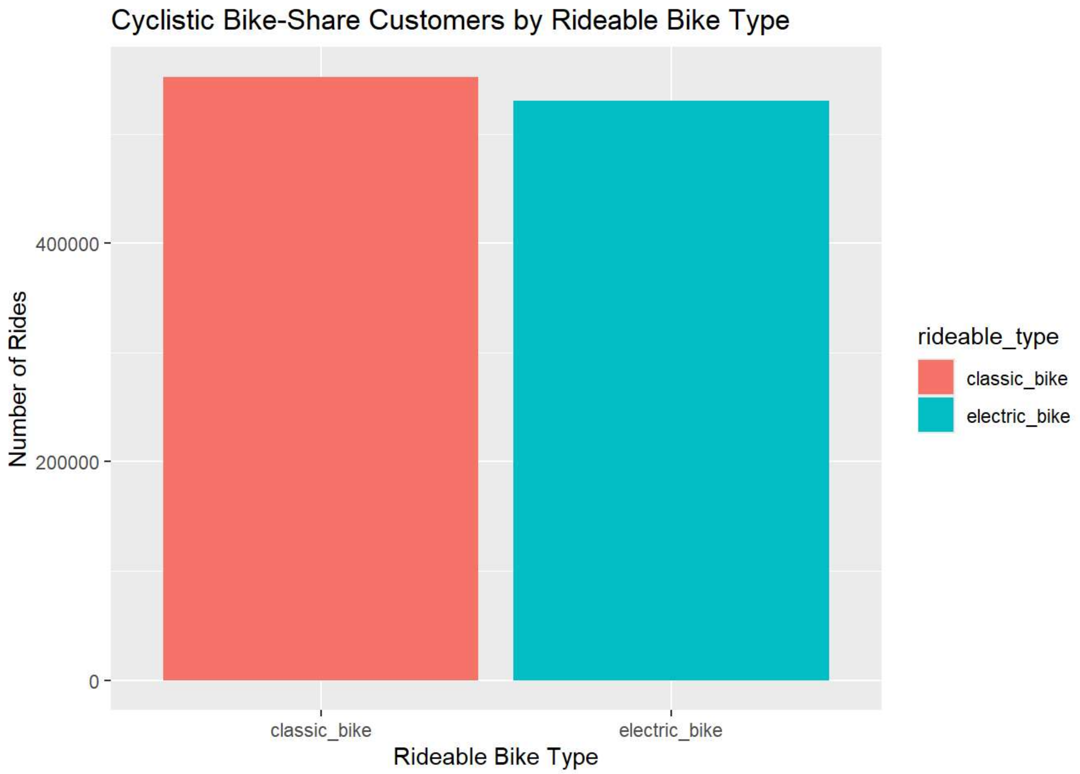
3. Total number of rides by bike type

```
Cyclistic_data %>%
  group_by(rideable_type) %>%
  count(rideable_type)
```

```
## # A tibble: 2 × 2
## # Groups:   rideable_type [2]
##   rideable_type      n
##   <chr>          <int>
## 1 classic_bike  552304
## 2 electric_bike 530545
```

```
options(scipen = 999)
ggplot(data = Cyclistic_data) +
  geom_bar(mapping = aes(x = rideable_type, fill = rideable_type, stat = "identity")) +
  labs(x = "Rideable Bike Type", y = "Number of Rides", title = "Cyclistic Bike-Share Customers by Rideable Bike Type")
```

```
## Warning in geom_bar(mapping = aes(x = rideable_type, fill = rideable_type, :
## Ignoring unknown aesthetics: stat
```



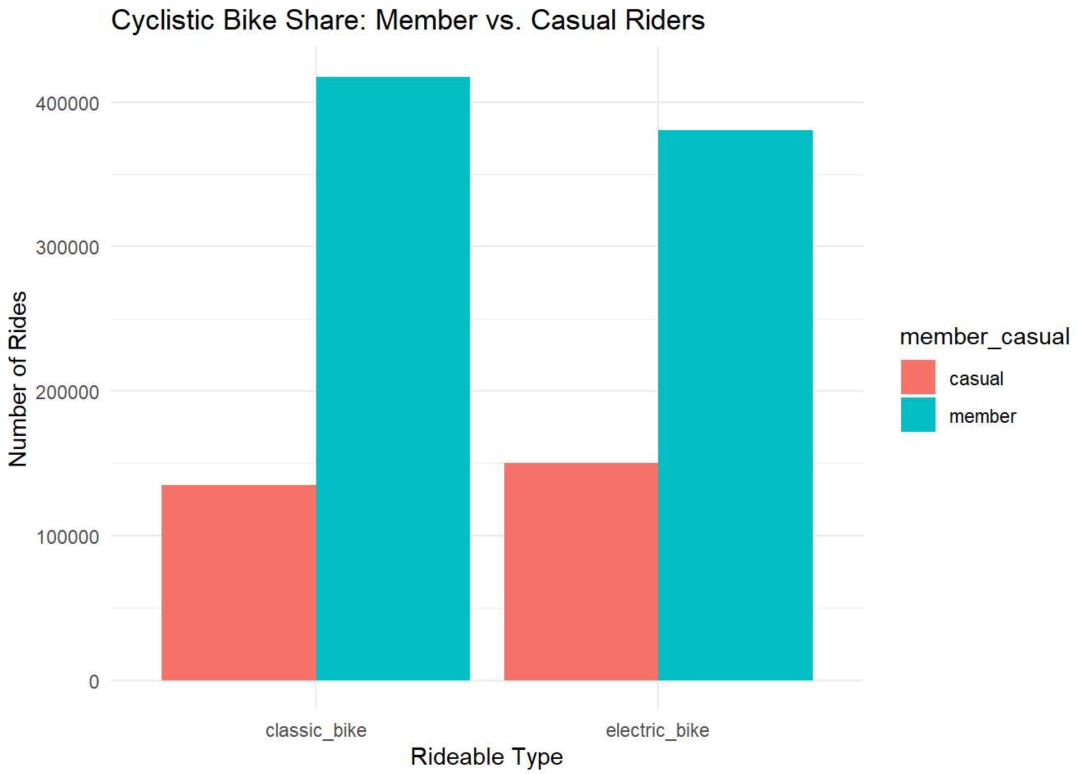
4. Total number of rides by member type and bike type

```
Cyclistic_data %>%
  group_by(member_casual, rideable_type) %>%
  count(rideable_type)
```

```
## # A tibble: 4 × 3
## # Groups:   member_casual, rideable_type [4]
##   member_casual rideable_type      n
##   <chr>         <chr>          <int>
## 1 casual        classic_bike  134713
## 2 casual        electric_bike 150167
## 3 member        classic_bike  417591
## 4 member        electric_bike 380378
```

```
Cyclistic_data %>%
  group_by(member_casual, rideable_type) %>%
  summarise(count = n()) %>%
  ggplot(aes(x = rideable_type, y = count, fill = member_casual)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Rideable Type", y = "Number of Rides", title = "Cyclistic Bike Share: Member vs. Casual Riders") +
  theme_minimal()
```

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



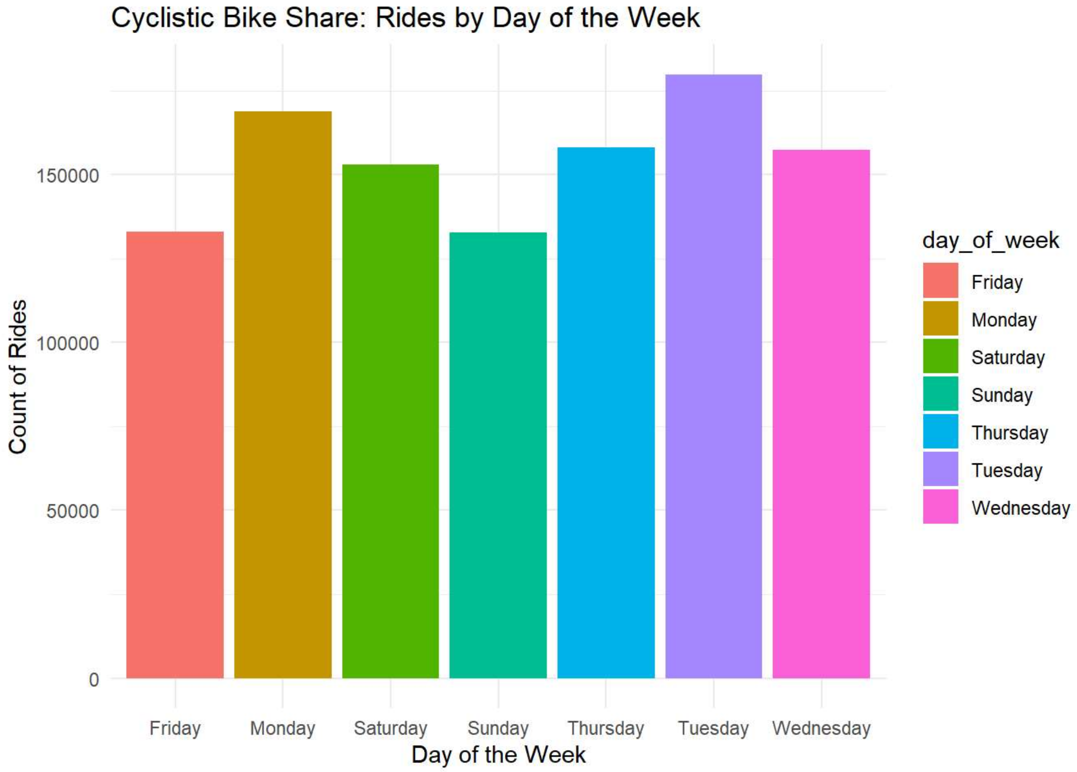
5. Total number of rides by day of the week

```
Cyclistic_data %>%
  count(day_of_week)
```

```
##   day_of_week      n
## 1      Friday 133002
## 2      Monday 168753
## 3    Saturday 152892
## 4      Sunday 132834
## 5    Thursday 158135
## 6     Tuesday 179923
## 7   Wednesday 157310
```

```
options(scipen = 999)
ggplot(data = Cyclistic_data) +
  geom_bar(mapping = aes(x = day_of_week, fill = day_of_week, stat = "identity")) +
  labs(x = "Day of the Week", y = "Count of Rides", title = "Cyclistic Bike Share: Rides by Day of the Week") +
  theme_minimal()
```

```
## Warning in geom_bar(mapping = aes(x = day_of_week, fill = day_of_week, stat =
## "identity")): Ignoring unknown aesthetics: stat
```



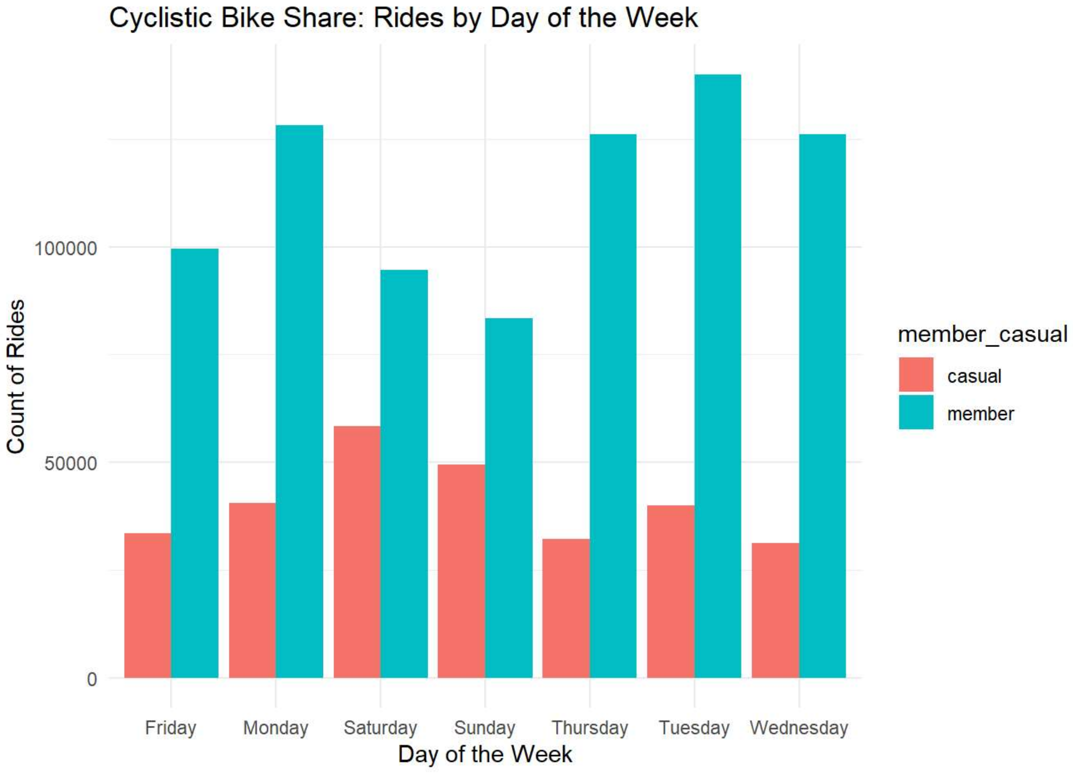
6. Total number of rides by member type and day of the week

```
Cyclistic_data %>%
  group_by(member_casual) %>%
  count(day_of_week)
```

```
## # A tibble: 14 × 3
## # Groups:   member_casual [2]
##    member_casual day_of_week      n
##    <chr>         <chr>        <int>
##  1 casual        Friday       33404
##  2 casual        Monday       40559
##  3 casual        Saturday     58316
##  4 casual        Sunday       49433
##  5 casual        Thursday     32087
##  6 casual        Tuesday      39902
##  7 casual        Wednesday    31179
##  8 member        Friday       99598
##  9 member        Monday      128194
## 10 member        Saturday     94576
## 11 member        Sunday       83401
## 12 member        Thursday    126048
## 13 member        Tuesday     140021
## 14 member        Wednesday   126131
```

```
Cyclistic_data %>%
  group_by(member_casual, day_of_week) %>%
  summarise(count = n()) %>%
  ggplot(aes(x = day_of_week, y = count, fill = member_casual)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Day of the Week", y = "Count of Rides", title = "Cyclistic Bike Share: Rides by Day of the Week") +
  theme_minimal()
```

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



Average Ride Length (From 2024 Jan to 2024 Apr)

1. Average ride length

```
cyclistic_avgRide <- mean(Cyclistic_data$ride_length)
print(cyclistic_avgRide)
```

```
## Time difference of 13.88265 mins
```

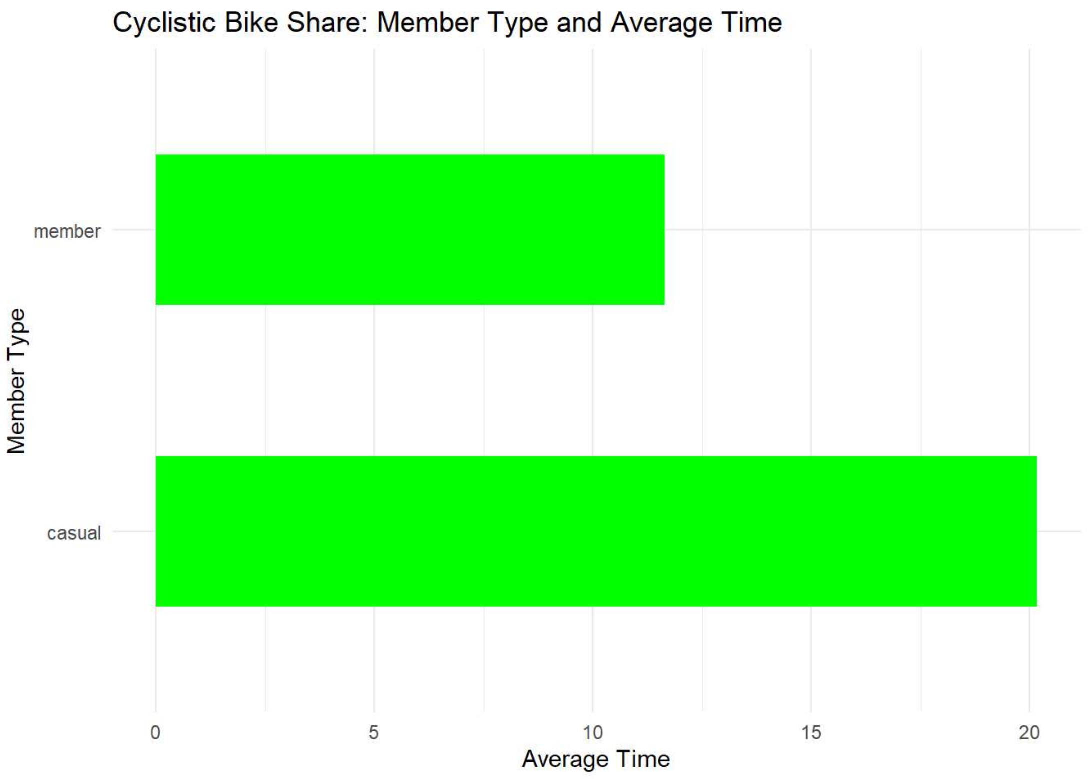
2. Average ride length by Member type

```
cyclistic_avgMeber <- Cyclistic_data %>% group_by(member_casual) %>%
                        summarise_at(vars(ride_length),list(time = mean))
cyclistic_avgMeber
```

```
## # A tibble: 2 × 2
##   member_casual time
##   <chr>         <drtn>
## 1 casual        20.16469 mins
## 2 member        11.63992 mins
```

```
ggplot(cyclistic_avgMeber) +
  geom_col(aes(x = time, y = member_casual), fill = 'green', width = 0.5)+
  labs(x = "Average Time", y = "Member Type", title = "Cyclistic Bike Share: Member Type and Average Time") +
  theme_minimal()
```

```
## Don't know how to automatically pick scale for object of type <difftime>.
## Defaulting to continuous.
```



3. Average ride length by Bike type

```
Cyclistic_data %>% group_by(rideable_type) %>%
  summarise_at(vars(ride_length),
               list(time = mean))
```

```
## # A tibble: 2 × 2
##   rideable_type time
##   <chr>         <drtn>
## 1 classic_bike  16.57842 mins
## 2 electric_bike 11.07632 mins
```

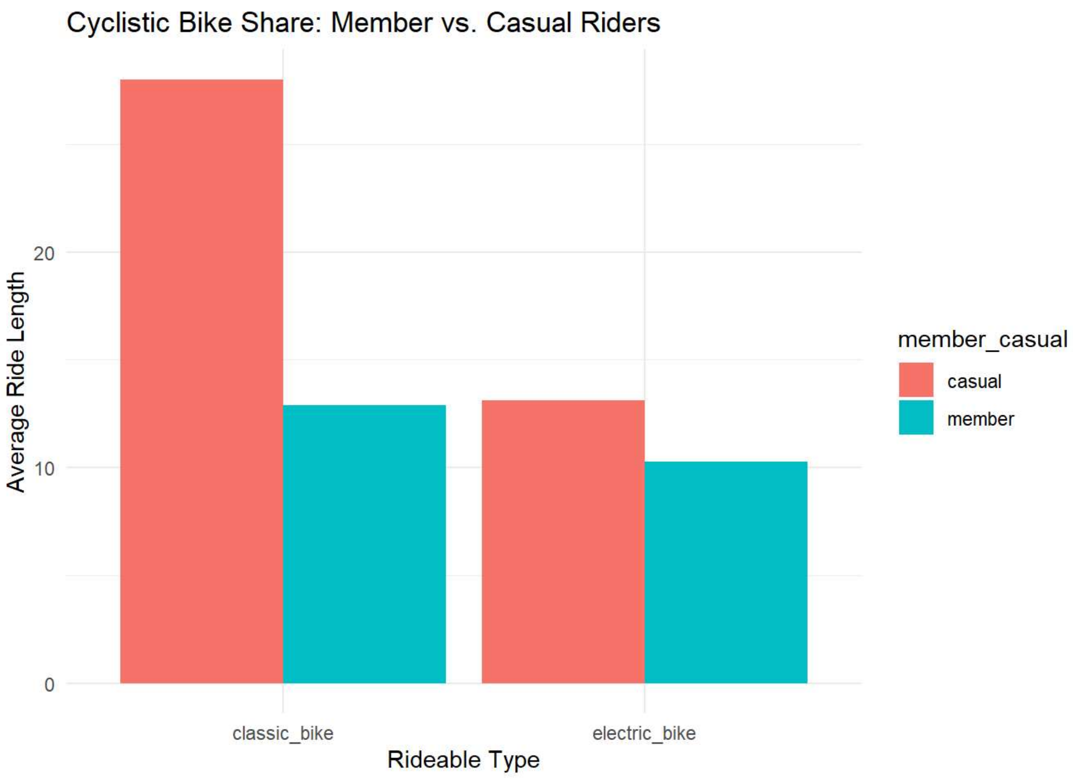
4. Average ride length by Bike type and member type

```
Cyclistic_data %>% group_by(member_casual, rideable_type) %>%
  summarise_at(vars(ride_length),list(time = mean))
```

```
## # A tibble: 4 × 3
## # Groups:   member_casual [2]
##   member_casual rideable_type time
##   <chr>         <chr>         <drtn>
## 1 casual        classic_bike  28.00935 mins
## 2 casual        electric_bike 13.12734 mins
## 3 member        classic_bike  12.89085 mins
## 4 member        electric_bike 10.26661 mins
```

```
Cyclistic_data %>%
  group_by(member_casual, rideable_type) %>%
  summarise_at(vars(ride_length), list(time = mean)) %>%
  ggplot(aes(x = rideable_type, y = time, fill = member_casual)) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(x = "Rideable Type", y = "Average Ride Length", title = "Cyclistic Bike Share: Member vs. Casual Riders") +
  theme_minimal()
```

```
## Don't know how to automatically pick scale for object of type <difftime>.
## Defaulting to continuous.
```



5. Average ride length by Day of the week

```
Cyclistic_data %>% group_by(day_of_week) %>%
  summarise_at(vars(ride_length),
               list(time = mean))
```

```
## # A tibble: 7 × 2
##   day_of_week time
##   <chr>       <drtn>
## 1 Friday      12.58891 mins
## 2 Monday      13.53511 mins
## 3 Saturday    16.91009 mins
## 4 Sunday      17.43339 mins
## 5 Thursday    11.79311 mins
## 6 Tuesday     13.20958 mins
## 7 Wednesday   12.27893 mins
```

6. Average ride length by Day of the week and member type

```
Cyclistic_data %>% group_by(member_casual, day_of_week) %>%
  summarise_at(vars(ride_length),
               list(time = mean))
```

```
## # A tibble: 14 × 3
## # Groups:   member_casual [2]
##    member_casual day_of_week time
##    <chr>         <chr>       <drtn>
##  1 casual        Friday      17.76393 mins
##  2 casual        Monday      20.02672 mins
##  3 casual        Saturday    23.44299 mins
##  4 casual        Sunday      25.11897 mins
##  5 casual        Thursday    15.21950 mins
##  6 casual        Tuesday     18.56450 mins
##  7 casual        Wednesday   16.06695 mins
##  8 member        Friday      10.85327 mins
##  9 member        Monday      11.48124 mins
## 10 member        Saturday    12.88187 mins
## 11 member        Sunday      12.87804 mins
## 12 member        Thursday    10.92089 mins
## 13 member        Tuesday     11.68357 mins
## 14 member        Wednesday   11.34255 mins
```

Summary and Conclusion

1. Total number of rides from 2024 Jan to 2024 Apr was 1,082,849. Members had more rides with 797,969 total rides or 74% and casual riders had 284,880 total rides or 26%.
2. Casual riders more prefer to use electric bike rather than classic bike, But members prefer to use electric bike.
3. More rides reported for casual members on the Monday and Tuesday but casual riders more prefer to use bikes on Saturday and Sunday.
4. Average ride length for both users was about 14 minutes. But casual riders have more average ride length (about 20 minutes) than members (about 12 minutes).
5. Weekends have the more average ride length comapre to the weekdays.

Until now, Cyclistic's marketing strategy relied on building general awareness and appealing to broad consumer segments. One approach that helped make these things possible was the flexibility of its pricing plans: single-ride passes, full-day passes, and annual memberships. Customers who purchase single-ride or full-day passes are referred to as casual riders. Customers who purchase annual memberships are Cyclistic members.

We can concluded that annual members are much more profitable than casual riders. Although the pricing flexibility will help Cyclistic attract more customers.

Maximizing the number of annual members will be key to future growth rather than creating a marketing campaign that targets all-new customers.

There is a solid opportunity to convert casual riders into members. Casual riders are already aware of the Cyclistic program and have chosen Cyclistic for their mobility needs.