

bike_share_company

Anord Nkuba

2023-04-15

```
#load packages
```

```
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.4.0      v purrr   0.3.5
## v tibble  3.1.8      v dplyr  1.0.10
## v tidyr   1.2.1      v stringr 1.4.1
## v readr   2.1.3      v forcats 0.5.2
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.2.2
```

```
## Loading required package: timechange
```

```
## Warning: package 'timechange' was built under R version 4.2.2
```

```
##
## Attaching package: 'lubridate'
##
## The following objects are masked from 'package:base':
##
##   date, intersect, setdiff, union
```

```
library(janitor)
```

```
## Warning: package 'janitor' was built under R version 4.2.3
```

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

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.2.2
```

```
##
## Attaching package: 'data.table'
##
## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
##
## The following objects are masked from 'package:dplyr':
##
##     between, first, last
##
## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(readr)
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.2.3
```

```
##
## Attaching package: 'psych'
##
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(hrbrthemes)
```

```
## Warning: package 'hrbrthemes' was built under R version 4.2.3
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.
##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and
##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
library(ggplot2)
```

```
importing data
```

```
clean_trip_final9 <- read_csv("clean_trip_final_tableau.csv")
```

```
## Rows: 3569098 Columns: 8
## -- Column specification -----
## Delimiter: ","
```

```
## chr (4): bike_type, costumer_type, month, week_day
## dbl (2): year, ride_length
## dtm (1): started_at
## time (1): time
##
## 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.
```

Total no. of customers

```
print(table(clean_trip_final9$costumer_type))
```

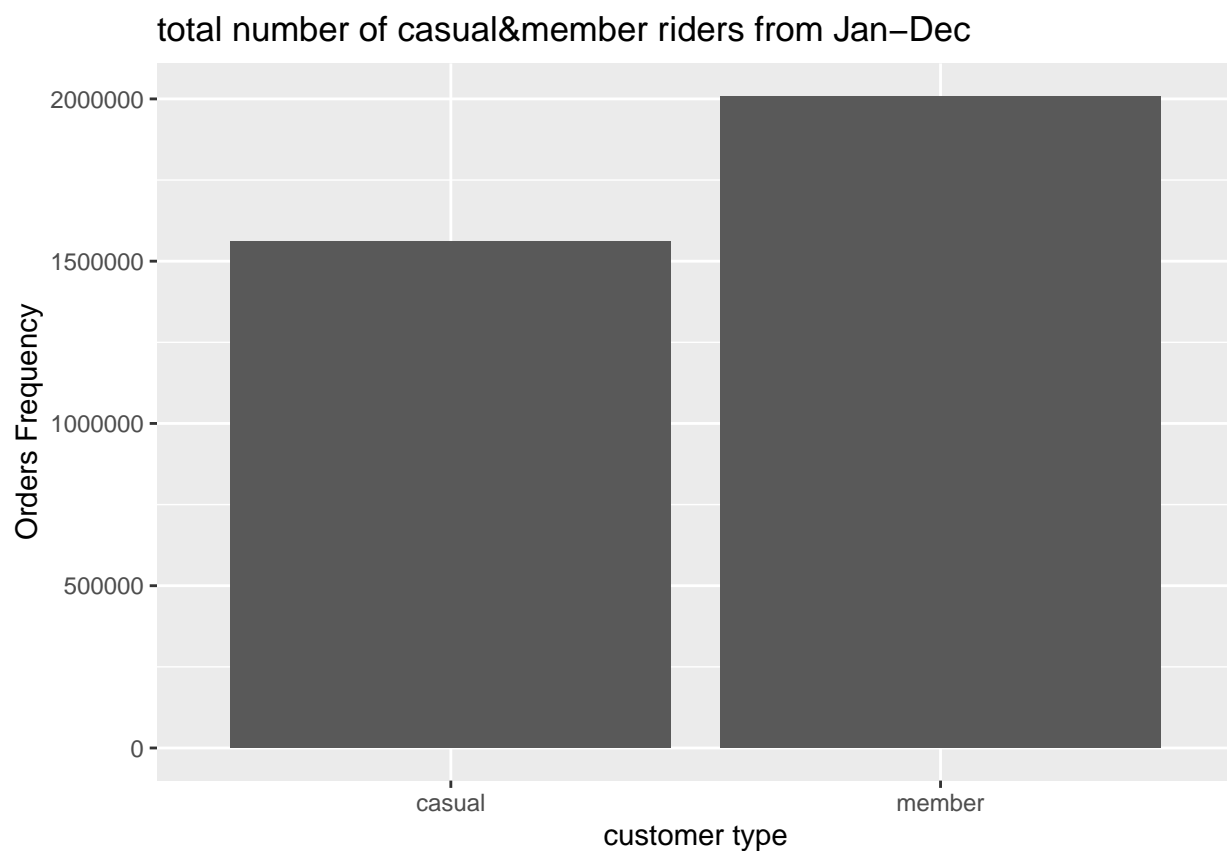
```
##
## casual member
## 1560216 2008882
```

#total number of casual and member customer from January to December 2022

```
total_customer <- (table(clean_trip_final9$costumer_type))
```

Data visualization total number of casual and member customer from January to December 2022

```
ggplot(data = data.frame(total_customer), aes(x = Var1, y = Freq)) + geom_bar(stat = "identity") +
  labs(title="total number of casual&member riders from Jan-Dec", x="customer type", y="Orders Frequency")
```



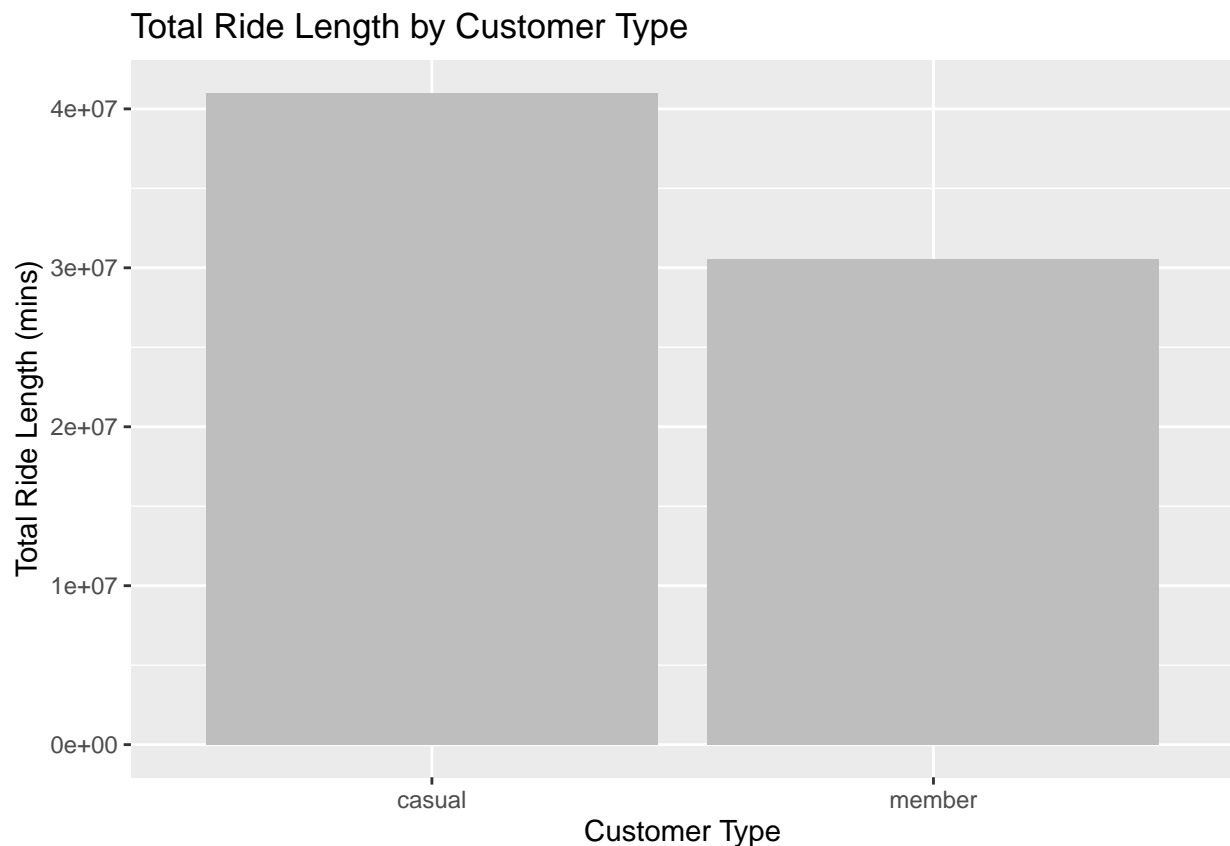
#Total rides for each customer type in minutes

```
total_ride <- setNames(aggregate(ride_length ~ costumer_type, clean_trip_final9, sum), c("customer_type"))
print(total_ride)
```

```
##   customer_type total_ride_min
## 1      casual      41018445
## 2      member      30522333
```

```
#data visualization of total rides by each customer type
```

```
ggplot(total_ride, aes(x=customer_type, y=total_ride_min)) +
  geom_bar(stat="identity", fill="grey") +
  labs(title="Total Ride Length by Customer Type", x="Customer Type", y="Total Ride Length (mins)")
```



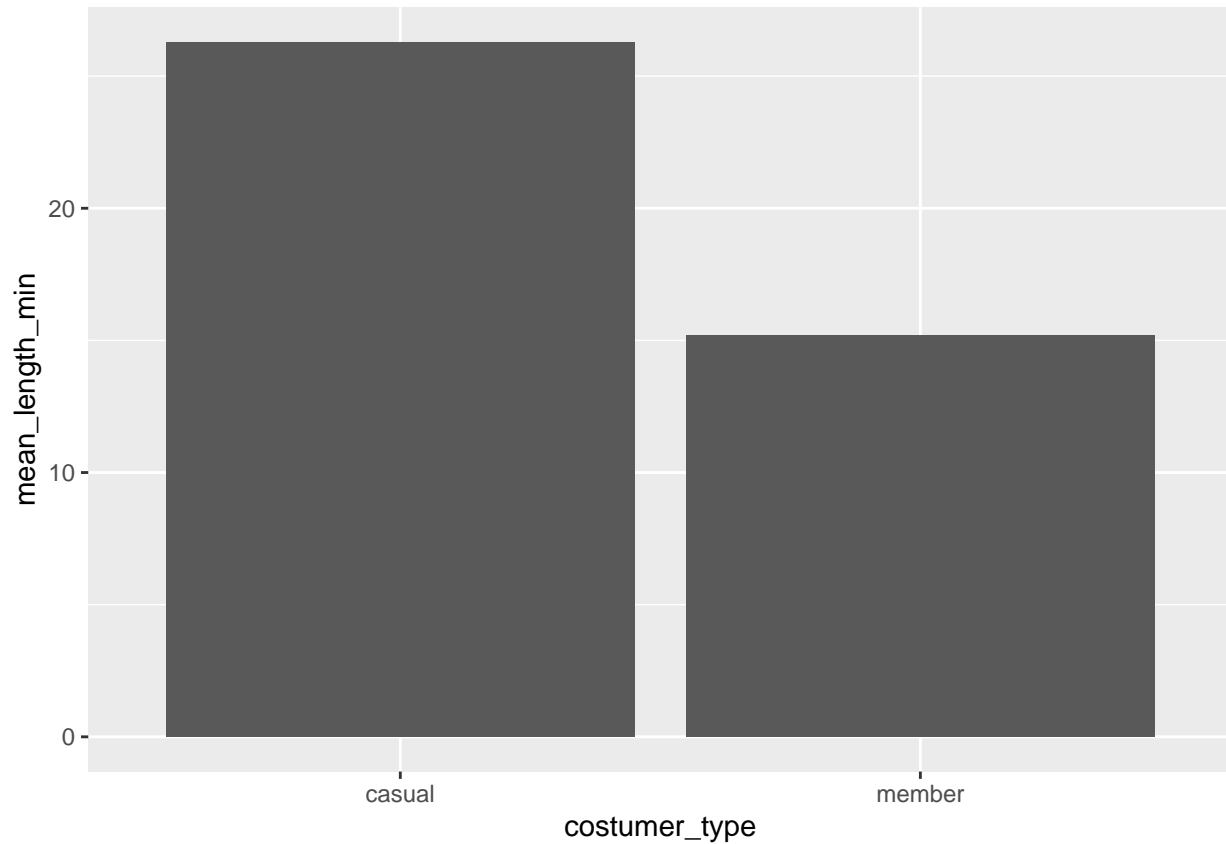
```
#Differences between members and casual riders in terms of length of ride
```

```
print(clean_trip_final9 %>%
  group_by(costumer_type) %>%
  summarise(min_length_mins = min(ride_length), max_length_min = max(ride_length),
    median_length_mins = median(ride_length), mean_length_min = mean(ride_length)))
```

```
## # A tibble: 2 x 5
##   costumer_type min_length_mins max_length_min median_length_mins mean_length_min
##   <chr>          <dbl>          <dbl>          <dbl>          <dbl>
## 1 casual              5          1397.           15.7           26.3
## 2 member              5          1394.           11.3           15.2
## # ... with abbreviated variable name 1: mean_length_min
```

```
#data visualization average ride length between casual and member riders from Jan to Dec
```

```
mean_ride_length <- clean_trip_final9 %>% group_by(costumer_type) %>% summarise(mean_length_min = mean(ride_length))
ggplot(data = mean_ride_length, aes(x = costumer_type, y = mean_length_min)) + geom_bar(stat = "identity")
```



```
#Average ride_length for users by day_of_week and Number of total rides by day_of_week
```

```
print(clean_trip_final9 %>%
  group_by(week_day) %>%
  summarise(Avg_length = mean(ride_length),
            number_of_ride = n()))
```

```
## # A tibble: 7 x 3
##   week_day Avg_length number_of_ride
##   <chr>      <dbl>         <int>
## 1 Friday      19.3         495797
## 2 Monday      19.7         468998
## 3 Saturday    23.4         603329
## 4 Sunday      23.6         507453
## 5 Thursday    18.2         518832
## 6 Tuesday     17.8         482956
## 7 Wednesday   17.6         491733
```

```
#Average ride_length by month
```

```
print(clean_trip_final9 %>%
      group_by(month) %>%
      summarise(Avg_length = mean(ride_length),
                 number_of_ride = n()))
```

```
## # A tibble: 12 x 3
##   month Avg_length number_of_ride
##   <chr>   <dbl>         <int>
## 1 Apr_22    19.7         216605
## 2 Aug_22    20.4         507431
## 3 Dec_22    14.4          97420
## 4 Feb_22    15.6          66127
## 5 Jan_22    14.4          59077
## 6 Jul_22    21.8         545700
## 7 Jun_22    21.7         529778
## 8 Mar_22    19.7         171177
## 9 May_22    22.6         423258
## 10 Nov_22   15.8         190696
## 11 Oct_22   18.1         323468
## 12 Sep_22   19.3         438361
```

#Average ride length comparison by each week day according to each customer type

```
print(aggregate(clean_trip_final9$ride_length ~ clean_trip_final9$costumer_type +
                 clean_trip_final9$week_day, FUN = mean))
```

```
##   clean_trip_final9$costumer_type clean_trip_final9$week_day
## 1                casual      Friday
## 2                member      Friday
## 3                casual      Monday
## 4                member      Monday
## 5                casual      Saturday
## 6                member      Saturday
## 7                casual      Sunday
## 8                member      Sunday
## 9                casual      Thursday
## 10               member      Thursday
## 11               casual      Tuesday
## 12               member      Tuesday
## 13               casual      Wednesday
## 14               member      Wednesday
##   clean_trip_final9$ride_length
## 1                24.68277
## 2                14.95213
## 3                27.18917
## 4                14.84388
## 5                28.76635
## 6                16.72224
## 7                29.44556
## 8                16.70597
## 9                23.65091
## 10               14.70303
```

```
## 11          23.97730
## 12          14.45575
## 13          23.03738
## 14          14.51123
```

#Average ride length comparison by each month according to each customer type

```
print(aggregate(clean_trip_final9$ride_length ~ clean_trip_final9$costumer_type +
                clean_trip_final9$month, FUN = mean))
```

```
##      clean_trip_final9$costumer_type clean_trip_final9$month
## 1          casual          Apr_22
## 2          member          Apr_22
## 3          casual          Aug_22
## 4          member          Aug_22
## 5          casual          Dec_22
## 6          member          Dec_22
## 7          casual          Feb_22
## 8          member          Feb_22
## 9          casual          Jan_22
## 10         member          Jan_22
## 11         casual          Jul_22
## 12         member          Jul_22
## 13         casual          Jun_22
## 14         member          Jun_22
## 15         casual          Mar_22
## 16         member          Mar_22
## 17         casual          May_22
## 18         member          May_22
## 19         casual          Nov_22
## 20         member          Nov_22
## 21         casual          Oct_22
## 22         member          Oct_22
## 23         casual          Sep_22
## 24         member          Sep_22
##      clean_trip_final9$ride_length
## 1          28.05020
## 2          14.45242
## 3          25.70483
## 4          15.61847
## 5          18.09232
## 6          13.12813
## 7          24.13444
## 8          13.53285
## 9          20.80305
## 10         12.99219
## 11         27.25770
## 12         16.03802
## 13         27.23204
## 14         16.16625
## 15         28.81221
## 16         14.69221
## 17         29.90076
```

```
## 18          16.00108
## 19          20.30176
## 20          13.74733
## 21          23.30140
## 22          14.63487
## 23          24.37023
## 24          15.28550
```

#Analyze rider length data by customer type and weekday

```
print(clean_trip_final9 %>%
  group_by(costumer_type, week_day) %>%
  summarise(number_of_ride = n(),
            avgerage_duration = mean(ride_length),
            median_duration = median(ride_length),
            max_duration = max(ride_length),
            min_duration = min(ride_length)))
```

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

```
## # A tibble: 14 x 7
## # Groups:   costumer_type [2]
##   costumer_type week_day  number_of_ride avgerage_dur~1 media~2 max_d~3 min_d~4
##   <chr>         <chr>          <int>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 casual      Friday           219763         24.7     15.0    1391.     5
## 2 casual      Monday           185652         27.2     15.9    1386.     5
## 3 casual      Saturday         333598         28.8     17.6    1397.     5
## 4 casual      Sunday           272798         29.4     17.9    1393.     5
## 5 casual      Thursday         200916         23.7     14.1    1394.     5
## 6 casual      Tuesday          170434         24.0     14.0    1396.     5
## 7 casual      Wednesday        177055         23.0     13.8    1390.     5
## 8 member      Friday           276034         15.0     11.1    1373.     5
## 9 member      Monday           283346         14.8     11      1394.     5
## 10 member     Saturday         269731         16.7     12.4    1366.     5
## 11 member     Sunday           234655         16.7     12.3    1304.     5
## 12 member     Thursday         317916         14.7     11.1    1390.     5
## 13 member     Tuesday          312522         14.5     10.9    1358.     5
## 14 member     Wednesday        314678         14.5     11      1330.     5
## # ... with abbreviated variable names 1: avgerage_duration, 2: median_duration,
## #   3: max_duration, 4: min_duration
```

#Analyze rider length data by customer type and month

```
print(clean_trip_final9 %>%
  group_by(costumer_type, month) %>%
  summarise(nummber_of_ride = n(),
            average_duration = mean(ride_length),
            median_duration = median(ride_length),
            max_duration = max(ride_length),
            min_duration = min(ride_length)))
```



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

```
## # A tibble: 24 x 7
## # Groups:   costumer_type [2]
##   costumer_type month   nummber_of_ride average_duration media~1 max_d~2 min_d~3
##   <chr>         <chr>         <int>          <dbl>    <dbl>    <dbl>    <dbl>
## 1 casual      Apr_22           82986          28.1     16.7    1385.     5
## 2 casual      Aug_22          240418          25.7     15.5    1386.     5
## 3 casual      Dec_22          24453           18.1     10.8    1386.     5
## 4 casual      Feb_22          13096           24.1     12.9    1388.     5
## 5 casual      Jan_22          10607           20.8     11.8    1380.     5
## 6 casual      Jul_22          280184          27.3     16.6    1394.     5
## 7 casual      Jun_22          264238          27.2     16.6    1396.     5
## 8 casual      Mar_22           60617           28.8     17.2    1381.     5
## 9 casual      May_22          200820           29.9     18.0    1390.     5
## 10 casual     Nov_22           59782           20.3     12.1    1367.     5
## # ... with 14 more rows, and abbreviated variable names 1: median_duration,
## # 2: max_duration, 3: min_duration
```