

# Cyclistic R Analysis

Faza Dhiyaulhaq

2023-01-22

## KAMPANYE TERBAIK ANNUAL MEMBERSHIP CYCLISTIC

### Pendahuluan

Dalam studi kasus ini, saya akan menghadapi banyak tugas yang biasa dilakukan seorang analis data junior di dunia nyata. Saya akan bekerja untuk sebuah perusahaan fiksi, Cyclistic, dan bertemu dengan karakter dan anggota tim yang berbeda. Untuk menjawab pertanyaan bisnis utama, saya akan mengikuti langkah-langkah proses analisis data: **bertanya, mempersiapkan, memproses, menganalisis, berbagi, dan bertindak.**

### Skenario

Saya adalah seorang analis data junior yang bekerja di tim analis pemasaran di Cyclistic, sebuah perusahaan berbagi sepeda di Chicago. Direktur pemasaran percaya bahwa kesuksesan perusahaan di masa depan tergantung pada upaya untuk memaksimalkan jumlah keanggotaan tahunan. Oleh karena itu, tim saya ingin memahami bagaimana pengendara kasual dan anggota tahunan menggunakan sepeda Cyclistic secara berbeda. Dari wawasan ini, tim saya akan merancang strategi pemasaran baru untuk mengubah pengendara biasa menjadi anggota tahunan. Tetapi pertama-tama, para eksekutif Cyclistic harus menyetujui rekomendasi saya, sehingga rekomendasi tersebut harus didukung dengan wawasan data yang menarik dan visualisasi data profesional.

### Karakteristik dan Tim

- Cyclistic: Program berbagi sepeda yang menampilkan lebih dari 5.800 sepeda dan 600 stasiun dok. Cyclistic membedakan diri dari perusahaan lain dengan juga menawarkan sepeda reclining, hand tricycles, dan sepeda kargo, menjadikan bike-share lebih inklusif bagi penyandang disabilitas dan pengendara yang tidak dapat menggunakan sepeda roda dua standar. Mayoritas pengendara memilih sepeda tradisional; sekitar 8% pengendara menggunakan opsi bantuan. Pengguna sepeda lebih cenderung bersepeda untuk bersantai, tetapi sekitar 30% menggunakannya untuk bepergian ke tempat kerja setiap hari.
- Lily Moreno: Direktur pemasaran dan manajer Anda. Moreno bertanggung jawab atas pengembangan kampanye dan inisiatif untuk mempromosikan program berbagi sepeda. Ini mungkin termasuk email, media sosial, dan saluran lainnya.
- Tim analitik pemasaran Cyclistic: Sebuah tim analis data yang bertanggung jawab untuk mengumpulkan, menganalisis, dan melaporkan data yang membantu memandu strategi pemasaran Cyclistic. Anda bergabung dengan tim ini enam bulan

lalu dan sibuk mempelajari misi dan tujuan bisnis Cyclistic — serta bagaimana Anda, sebagai analis data junior, dapat membantu Cyclistic untuk mencapainya.

- Tim eksekutif Cyclistic: Tim eksekutif yang sangat berorientasi pada detail akan memutuskan apakah mereka menyetujui program pemasaran yang direkomendasikan.

## Tentang Perusahaan

Pada 2016, Cyclistic meluncurkan penawaran berbagi sepeda yang sukses. Sejak itu, program tersebut telah berkembang menjadi armada 5.824 sepeda yang dilacak secara geografis dan dikunci ke dalam jaringan 692 stasiun di seluruh Chicago. Sepeda dapat diakses dari satu stasiun dan dikembalikan ke stasiun lainnya kapan pun di dalam sistem. Sampai saat ini, strategi pemasaran Cyclistic mengandalkan upaya untuk membangun kesadaran umum dan menarik segmen konsumen yang luas. Salah satu pendekatan yang membantu mewujudkan hal ini adalah fleksibilitas harga: tiket sekali jalan, tiket sehari penuh, dan keanggotaan tahunan. Pelanggan yang membeli tiket sekali jalan atau tiket sehari penuh disebut sebagai pengendara kasual. Pelanggan yang membeli keanggotaan tahunan adalah anggota Cyclistic.

Analisis keuangan Cyclistic telah menyimpulkan bahwa anggota tahunan jauh lebih menguntungkan dibandingkan pengendara biasa. Meskipun fleksibilitas harga membantu Cyclistic menarik lebih banyak pelanggan, Moreno percaya bahwa memaksimalkan jumlah anggota tahunan akan menjadi kunci pertumbuhan di masa depan. Daripada membuat kampanye pemasaran yang menargetkan pelanggan baru, Moreno percaya bahwa ada peluang untuk mengubah pengendara biasa menjadi anggota. Ia mencatat bahwa pengendara kasual sudah mengetahui program Cyclistic dan telah memilih Cyclistic untuk kebutuhan mobilitas mereka.

Moreno telah menetapkan tujuan yang jelas: Merancang strategi pemasaran yang bertujuan untuk mengubah pengendara biasa menjadi anggota tahunan. Namun, untuk melakukannya, tim analisis pemasaran perlu lebih memahami perbedaan antara anggota tahunan dan pengendara biasa, mengapa pengendara akan bersedia membeli keanggotaan, dan bagaimana media digital dapat memengaruhi taktik pemasaran mereka. Moreno dan timnya tertarik untuk menganalisis data perjalanan sepeda Cyclistic untuk mengidentifikasi tren.

### 1. Ask Phase

Dalam fase ini pemangku kepentingan mengajukan pertanyaan untuk tercapainya tujuan, pertanyaan yang diajukan yaitu: 1. Bagaimana anggota tahunan dan pengendara biasa menggunakan sepeda secara berbeda? 2. Mengapa pengendara biasa membeli keanggotaan tahunan Cyclistic? 3. Bagaimana Cyclistic menggunakan media digital untuk mempengaruhi pengendara biasa untuk menjadi anggota?

## 2. Prepare Phase

Dalam fase ini data perjalanan historis Cyclistic selama 12 bulan yaitu dari bulan Oktober 2021 sampai September 2022 digunakan untuk menganalisis dan mengidentifikasi tren.

**Di mana data Anda berada?** Data disimpan pada *cloud-based system* dan telah disediakan oleh Motivate International Inc.

**Bagaimana data diorganisasikan?** Data diorganisasikan, disatukan, dibersihkan, dan divisualisasikan menggunakan R untuk melihat data dari perspektif lain dan untuk membuat cerita yang lebih menarik dari data tersebut.

**Apakah ada masalah dengan bias atau kredibilitas dalam data ini? Apakah data Anda ROCCC?** Data yang digunakan dalam analisis ini diekstrak yang disediakan oleh Divvy (Motivate International Inc.) dengan [lisensi berikut](#).

Catatan: Cyclistic merupakan perusahaan fiktif dan data dari Divvy digunakan dengan tujuan studi kasus ini.

Data tersedia dalam bentuk repositori yang dikumpulkan dari waktu ke waktu. Tiap poin data merepresentasikan perjalanan sepeda dari satu dok stasiun ke dok stasiun berikutnya.

Untuk mengetahui apakah data tidak bias dan memenuhi kredibilitas, maka akan digunakan sistem "ROCCC" untuk mengetahuinya. Sistem ini yaitu singkatan dari reliable(diandalkan), Original(Asli), Comprehensive(Informatif), Current(Terkini), dan Cited(Dikutip)

**Reliability:** Data ini diambil dari sumber eksternal yang diberikan secara terbatas dan tidak dapat diandalkan sepenuhnya. Namun, bukan berarti kita tidak bisa menghasilkan *insight* dari data tersebut nantinya.

**Originality:** Data bersumber dari pihak pertama dan original

**Comprehensiveness:** Data cukup lengkap dan sebagian besar terhindar dari kesalahan manusia. Data ini berisi informasi yang dibutuhkan untuk menjawab pertanyaan.

**Current:** Data yang digunakan memberikan informasi yang terkini dan terus diperbarui.

**Cited:** Data dikutip dari pihak pertama secara langsung sehingga data memiliki kredibilitas yang cukup baik

### Menyatukan data yang dibutuhkan

#### #Q4 2021 Data Import

```
tripdata_202110 <- read.csv("202110-divvy-tripdata.csv")
tripdata_202111 <- read.csv("202111-divvy-tripdata.csv")
tripdata_202112 <- read.csv("202112-divvy-tripdata.csv")
```

#### #Q4 2021 Combine Data

##### #Check the Data

```
colnames(tripdata_202110)
colnames(tripdata_202111)
```

```

colnames(tripdata_202112)

str(tripdata_202110)
str(tripdata_202111)
str(tripdata_202112)

q4_2021 <- bind_rows(tripdata_202110,tripdata_202111,tripdata_202112)

#Q1 2022 Data Import
tripdata_202201 <- read.csv("202201-divvy-tripdata.csv")
tripdata_202202 <- read.csv("202202-divvy-tripdata.csv")
tripdata_202203 <- read.csv("202203-divvy-tripdata.csv")

#Q1 2022 Combine Data
q1_2022 <- bind_rows(tripdata_202201,tripdata_202202,tripdata_202203)

#Q2 2022 Data Import
tripdata_202204 <- read.csv("202204-divvy-tripdata.csv")
tripdata_202205 <- read.csv("202205-divvy-tripdata.csv")
tripdata_202206 <- read.csv("202206-divvy-tripdata.csv")

#Q2 2022 Combine Data
q2_2022 <- bind_rows(tripdata_202204,tripdata_202205,tripdata_202206)

#Q3 2022 Data Import
tripdata_202207 <- read.csv("202207-divvy-tripdata.csv")
tripdata_202208 <- read.csv("202208-divvy-tripdata.csv")
tripdata_202209 <- read.csv("202209-divvy-publictripdata.csv")

#Q3 2022 Combine Data
q3_2022 <- bind_rows(tripdata_202207,tripdata_202208,tripdata_202209)

#combine all trips
all_trips <- bind_rows(q4_2021,q1_2022,q2_2022,q3_2022)

```

## Menyiapkan Package yang dibutuhkan

### Install packages

```

install.packages("dplyr")

## Installing package into 'C:/Users/ASBIL/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## Error in contrib.url(repos, "source"): trying to use CRAN without setting
a mirror

```

```
install.packages("tidyverse")

## Installing package into 'C:/Users/ASBIL/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## Error in contrib.url(repos, "source"): trying to use CRAN without setting
a mirror

install.packages("ggplot2")

## Installing package into 'C:/Users/ASBIL/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## Error in contrib.url(repos, "source"): trying to use CRAN without setting
a mirror

install.packages("readr")

## Installing package into 'C:/Users/ASBIL/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## Error in contrib.url(repos, "source"): trying to use CRAN without setting
a mirror

install.packages("lubridate")

## Installing package into 'C:/Users/ASBIL/AppData/Local/R/win-library/4.2'
## (as 'lib' is unspecified)

## Error in contrib.url(repos, "source"): trying to use CRAN without setting
a mirror
```

### *Library*

```
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.2.2

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

library(tidyverse)

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

```
## — Attaching packages
## _____
## tidyverse 1.3.2 —

## ✓ ggplot2 3.4.0      ✓ purrr 0.3.5
## ✓ tibble 3.1.8      ✓ stringr 1.4.1
## ✓ tidyr 1.2.1       ✓ forcats 0.5.2
## ✓ readr 2.1.3

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

## Warning: package 'readr' was built under R version 4.2.2

## — Conflicts —
tidyverse_conflicts() —
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag() masks stats::lag()

library(ggplot2)
library(lubridate)

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

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

library(readr)
```

### 3. Process dan Analyze Phase

#### Membersihkan data

```
#Check data
colnames(all_trips)
nrow(all_trips)
dim(all_trips)
head(all_trips)
tail(all_trips)
str(all_trips)
summary(all_trips)

# Membuat kolom tanggal dari kolom started_at
all_trips$date <- as.Date(all_trips$started_at)

# Membuat kolom day, month, year dari kolom date
all_trips$day <- format(as.Date(all_trips$date), "%d")
```

```

all_trips$month <- format(as.Date(all_trips$date), "%m")
all_trips$year <- format(as.Date(all_trips$date), "%y")

# Membuat kolom ride_length untuk menghitung berapa detik lama perjalanan
all_trips$ride_length <- difftime(all_trips$ended_at, all_trips$started_at)

# Mengubah data "ride_length" dari Factor menjadi Numeric sehingga data dapat
dikalkulasi
is.factor(all_trips$ride_length)
all_trips$ride_length <- as.numeric(as.character(all_trips$ride_length))
is.numeric(all_trips$ride_length)

# Menghapus data yang rusak dan membuat versi 2 dari data
# Menghilangkan durasi berkendara yang < 0 detik
all_trips_v2 <- all_trips[!(all_trips$start_station_name == "HQ QR" |
all_trips$ride_length < 0),]
all_trips_v2$month <- as.numeric(as.character(all_trips_v2$month))
is.numeric(all_trips_v2$month)

# Membuat kolom nama hari dari kolom date
all_trips_v2$day_of_week <- wday(all_trips_v2$date, label=TRUE)

# Mengkonversi nama bulan menjadi 3 huruf
all_trips_v2$month <- month.abb[all_trips_v2$month]
all_trips_v2$month <- as.factor(as.character(all_trips_v2$month))
is.factor(all_trips_v2$month)
unique(all_trips_v2$month)

# Mengurutkan bulan dan hari sesuai urutan
all_trips_v2$day_of_week <- ordered(all_trips_v2$day_of_week,
levels=c("Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"))
all_trips_v2$month <- ordered(all_trips_v2$month, levels=c("Jan", "Feb",
"Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"))

# Mengubah membership value
all_trips_v2["member_casual"][all_trips_v2["member_casual"] == "casual"] <-
"Casual Member"
all_trips_v2["member_casual"][all_trips_v2["member_casual"] == "member"] <-
"Annual Member"
all_trips_v2["rideable_type"][all_trips_v2["rideable_type"] ==
"docked_bike"] <- "Docked Bike"
all_trips_v2["rideable_type"][all_trips_v2["rideable_type"] ==
"classic_bike"] <- "Classic Bike"
all_trips_v2["rideable_type"][all_trips_v2["rideable_type"] ==
"electric_bike"] <- "Electric Bike"

```

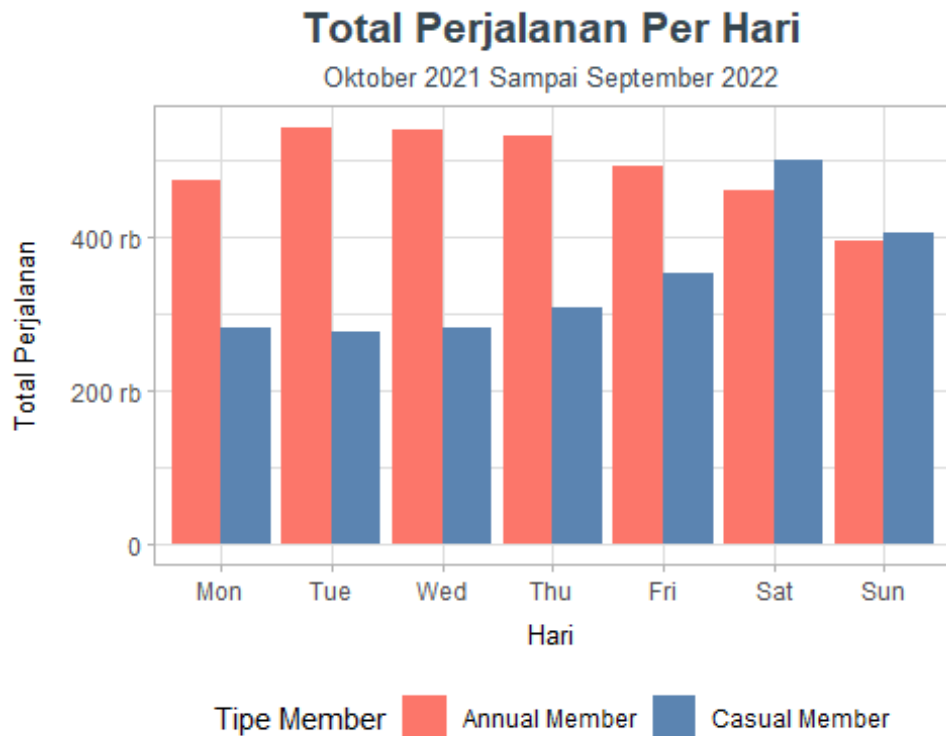
## 4. Share Phase (Data Visualisation)

### Total Perjalanan Per Hari

```
all_trips_v2 %>%
  group_by(member_casual, day_of_week) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, day_of_week) %>%
  ggplot(aes(x = day_of_week, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  scale_fill_manual("Tipe Member", values = c("Annual Member" =
"#FC766AFF", "Casual Member" = "#5B84B1FF")) +
  theme_light() +
  theme(plot.title =
element_text(face="bold",colour="#36454F",hjust=0.5,size = 16),
        plot.subtitle = element_text(colour="#36454F",hjust=0.5,size = 10),
        axis.title.x = element_text(size = 10, margin=margin(t=7)),
        axis.title.y = element_text(size = 10, margin=margin(r=12)),
        legend.position = "bottom") +
  labs(x='Hari', y='Total Perjalanan',
       title='Total Perjalanan Per Hari',
       subtitle="Oktober 2021 Sampai September 2022")+
  scale_y_continuous(breaks = c(0e+00, 2e+05, 4e+05), labels = c("0", "200
rb", "400 rb"))

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

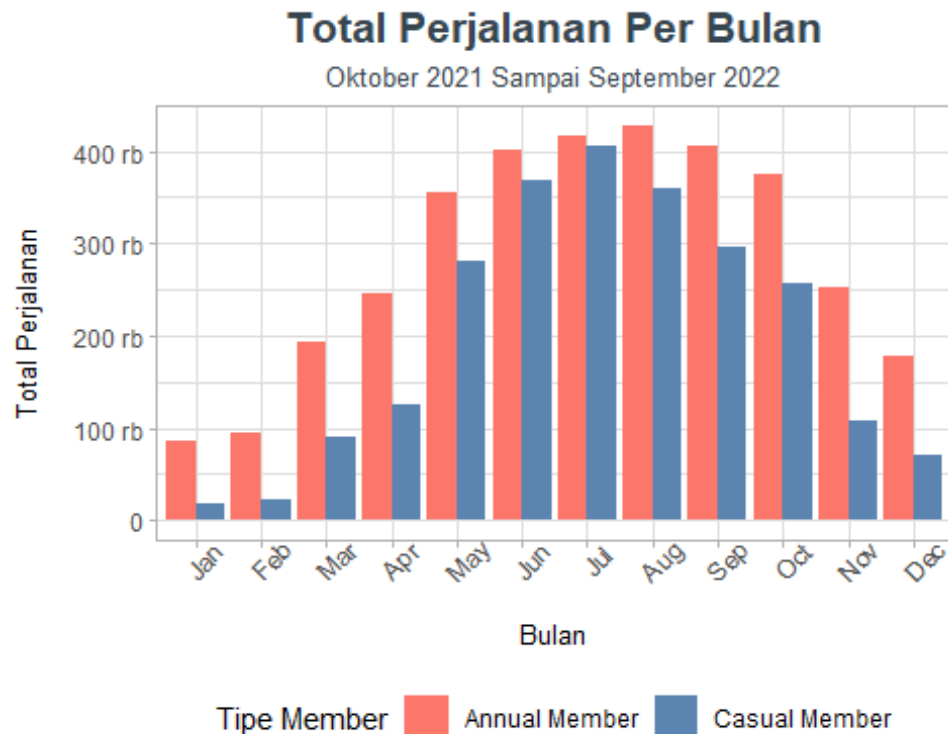




### Total Perjalanan Per Bulan

```
all_trips_v2 %>%
  group_by(member_casual, month) %>%
  summarise(number_of_rides = n()
             , average_duration = mean(ride_length)) %>%
  arrange(member_casual, month) %>%
  ggplot(aes(x = month, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge") +
  scale_fill_manual("Tipe Member", values = c("Annual Member" =
"#FC766AFF", "Casual Member" = "#5B84B1FF")) +
  theme_light() +
  theme(plot.title = element_text(face="bold", colour="#36454F", hjust=0.5,
size=16),
        plot.subtitle = element_text(colour="#36454F", hjust=0.5, size=10),
        axis.text.x = element_text(angle = 45),
        axis.title.x = element_text(size = 10, margin=margin(t=7)),
        axis.title.y = element_text(size = 10, margin=margin(r=12)),
        legend.position = "bottom") +
  labs(x='Bulan', y='Total Perjalanan',
       title='Total Perjalanan Per Bulan',
       subtitle="Oktober 2021 Sampai September 2022")+
  scale_y_continuous(breaks = c(0e+00, 1e+05, 2e+05, 3e+05, 4e+05), labels
= c("0", "100 rb", "200 rb", "300 rb", "400 rb"))

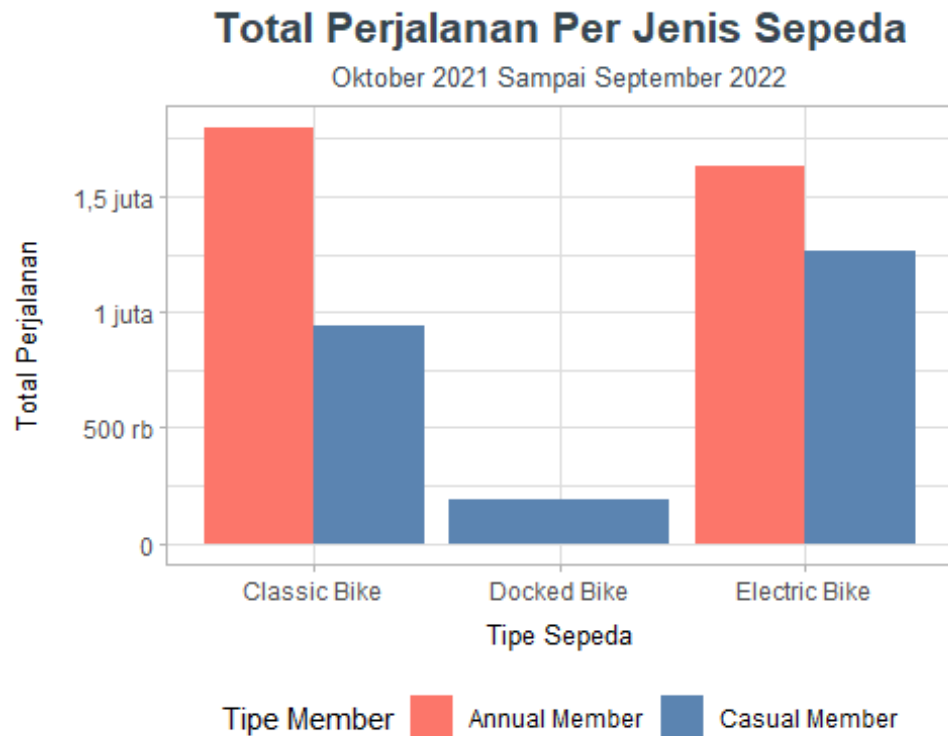
## `summarise()` has grouped output by 'member_casual'. You can override
using the
## `.groups` argument.
```



### Total Perjalanan Per Jenis Sepeda

```
all_trips_v2 %>%
  group_by(member_casual, rideable_type) %>%
  summarise(ride_type = n()) %>%
  ggplot(mapping=aes(x=rideable_type, y=ride_type, fill=member_casual,
group=member_casual)) +
  geom_col(position = "dodge") +
  scale_fill_manual("Tipe Member",
                    values = c("Annual Member" = "#FC766AFF", "Casual
Member" = "#5B84B1FF")) +
  theme_light() +
  theme(plot.title = element_text(face="bold", colour="#36454F", hjust=0.5,
size=16),
        plot.subtitle = element_text(colour="#36454F", hjust=0.5, size=10),
        axis.title.x = element_text(size = 10, margin=margin(t=7)),
        axis.title.y = element_text(size = 10, margin=margin(r=12)),
        legend.position = "bottom") +
  labs(x='Tipe Sepeda', y='Total Perjalanan',
        title='Total Perjalanan Per Jenis Sepeda',
        subtitle="Oktober 2021 Sampai September 2022")+
  scale_y_continuous(breaks = c(0, 500000, 1000000, 1500000),
                    labels = c("0", "500 rb", "1 juta", "1,5 juta"))

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



### Rata-rata Durasi Perjalanan Per Hari

```
all_trips_v2 %>%
  mutate(hours = (format(as.POSIXct(all_trips_v2$started_at), format =
"%H"))) %>%
  filter(day_of_week %in% c("Mon", "Tue", "Wed", "Thu", "Fri")) %>%
  group_by(member_casual, hours) %>%
  summarise(number_of_rides = n(),
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, hours) %>%
  ggplot(mapping=aes(x=hours, y=number_of_rides, color=member_casual,
group=member_casual)) +
  geom_line() +
  geom_point() +
  scale_color_manual(name="Tipe Member",
                    values = c("Annual Member" = "#FC766AFF", "Casual
Member" = "#5B84B1FF")) +
  theme_light() +
  theme(plot.title =
element_text(face="bold", colour="#36454F",hjust=0.5,size=16),
        plot.subtitle = element_text(colour="#36454F",hjust=0.5,size=10),
        axis.title.x = element_text(size = 10, margin=margin(t=7)),
        axis.title.y = element_text(size = 10, margin=margin(r=12)),
        legend.position = "bottom") +
  labs(x='Jam', y='Rata-rata Durasi Perjalanan',
       title='Rata-Rata Durasi Perjalanan Tiap Jam (Hari Kerja)',
       subtitle="Oktober 2021 Sampai September 2022",
```

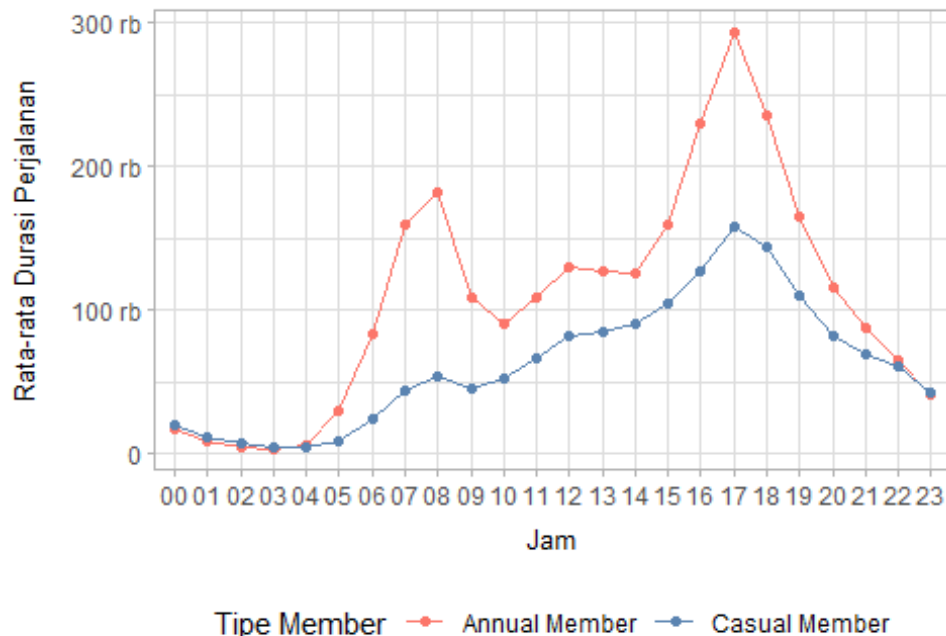
```
fill="Tipe Member") +
  scale_y_continuous(breaks = c(0e+00, 1e+05, 2e+05, 3e+05), labels = c("0",
"100 rb", "200 rb", "300 rb"))
```

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

## `.groups` argument.

## Rata-Rata Durasi Perjalanan Tiap Jam (Hari K

Oktober 2021 Sampai September 2022



## Rata-rata Durasi Perjalanan Tiap Jamnya (Weekday)

```
all_trips_v2 %>%
  mutate(hours = (format(as.POSIXct(all_trips_v2$started_at), format =
"%H")))) %>%
  filter(day_of_week %in% c("Mon", "Tue", "Wed", "Thu", "Fri")) %>%
  group_by(member_casual, hours) %>%
  summarise(number_of_rides = n()
, average_duration = mean(ride_length)) %>%
  arrange(member_casual, hours) %>%
  ggplot(mapping=aes(x=hours, y=number_of_rides, color=member_casual,
group=member_casual)) +
  geom_line() +
  geom_point() +
  scale_color_manual(name="Tipe Member",
values = c("Annual Member" = "#FC766AFF", "Casual
Member" = "#5B84B1FF")) +
  theme_light() +
  theme(plot.title =
element_text(face="bold", colour="#36454F", hjust=0.5, size=16),
```

```

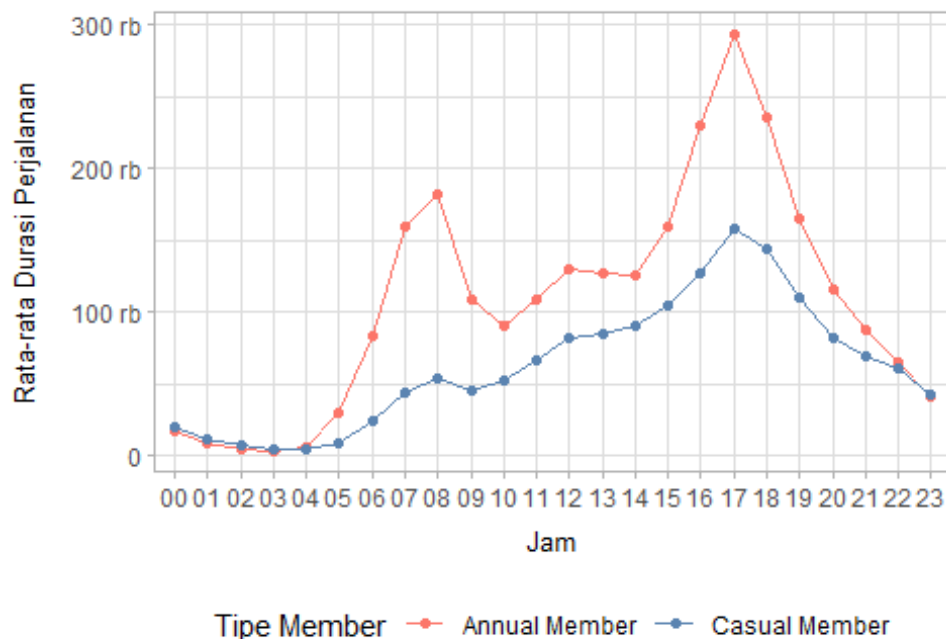
plot.subtitle = element_text(colour="#36454F",hjust=0.5,size=10),
axis.title.x = element_text(size = 10, margin=margin(t=7)),
axis.title.y = element_text(size = 10, margin=margin(r=12)),
legend.position = "bottom") +
labs(x='Jam', y='Rata-rata Durasi Perjalanan',
title='Rata-Rata Durasi Perjalanan Tiap Jam (Hari Kerja)',
subtitle="Oktober 2021 Sampai September 2022",
fill="Tipe Member") +
scale_y_continuous(breaks = c(0e+00, 1e+05, 2e+05, 3e+05), labels = c("0",
"100 rb", "200 rb", "300 rb"))

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

```

## Rata-Rata Durasi Perjalanan Tiap Jam (Hari K

Oktober 2021 Sampai September 2022



## Rata-rata Durasi Perjalanan Tiap Jamnya (Weekend)

```

all_trips_v2 %>%
  mutate(hours = (format(as.POSIXct(all_trips_v2$started_at), format =
"%H")))) %>%
  filter(day_of_week %in% c("Sat", "Sun")) %>%
  group_by(member_casual, hours) %>%
  summarise(number_of_rides = n()
,average_duration = mean(ride_length)) %>%
  arrange(member_casual, hours) %>%
  ggplot(mapping=aes(x=hours, y=number_of_rides, color=member_casual,
group=member_casual)) +
  geom_line() +

```

```

geom_point() +
scale_color_manual(name="Tipe Member",
                    values = c("Annual Member" = "#FC766AFF", "Casual
Member" = "#5B84B1FF")) +
theme_light() +
theme(plot.title =
element_text(face="bold", colour="#36454F", hjust=0.5, size=16),
        plot.subtitle = element_text(colour="#36454F", hjust=0.5, size=10),
        axis.title.x = element_text(size = 10, margin=margin(t=7)),
        axis.title.y = element_text(size = 10, margin=margin(r=12)),
        legend.position = "bottom"
) +
labs(x='Jam', y='Rata-rata Durasi Perjalanan',
      title='Rata-Rata Durasi Perjalanan Tiap Jamnya (Hari Libur)',
      subtitle="Oktober 2021 Sampai September 2022",
      fill="Tipe Member") +
scale_y_continuous(breaks = c(0, 20000, 40000, 60000, 80000), labels =
c("0", "20 rb", "40 rb", "60 rb", "80 rb"))

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

```

## Rata-Rata Durasi Perjalanan Tiap Jamnya (Hari

Oktober 2021 Sampai September 2022

