Task-01

Name : Imanuel Adipranata

ID : 22056482

Github : https://github.com/22056482/R_Project

Importing necessary libraries

```
library(dplyr)
```

```
##
## 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(knitr)
```

Loading Data

```
user_data <- read.csv("../data/users.csv")
head(user_data)
```

```
##   user_id     name review_count average_stars member_since
## 1     u_0     Alan           32          2.08   2019-04-05
## 2     u_1     Joel           90          1.97   2015-11-15
## 3     u_2   Claire           93          1.10   2021-10-05
## 4     u_3 Samantha           59          3.01   2017-05-15
## 5     u_4  Monique           42          4.44   2021-04-05
## 6     u_5    Lucas           62          1.63   2023-09-16
```

As you can see there are several missing value on the data. We need to assign NA value to the missing data because R will read that as empty string (“ “)

```
user_data[user_data == ""] <- NA
```

Since each entity can already be identified using *user_id*, the *name* column can be ignored.

```
user_data <- subset(user_data, select = -name)
```

NA Handling

```
colSums(is.na(user_data)) # print count of NA value for each columns
```

```
##       user_id  review_count average_stars  member_since
##             1             0             0          1160
```

There is one missing value in the *user_id* column and 1160 missing values in the *member_since* column. We can remove the row that does not have a *user_id*.

```r
(1160 / nrow(user_data)) * 100 # calculate proportion of missing value on member_since column
```

```
## [1] 2.989614
```

Since the proportion of missing values in the member_since column is only around 3%, we can also remove the rows with missing values in that column.

```r
user_data <- na.omit(user_data) # remove row with NA value
colSums(is.na(user_data)) # check again NA value on user_data
```

```
##       user_id  review_count average_stars  member_since
##             0             0             0             0
```

Grouping

The users will be grouped into 3 group: Veteran, Intermediate and New (based on their member since date) before 2017, between 2017-2022, and after 2022 respectively.

```r
user_data <- user_data %>%
  mutate(
    # Extract the year from the date string and convert it to a numeric format
    year_joined = as.integer(substr(member_since, 1, 4)),

    # Create a column called member_category based on the year_joined
    member_category = case_when(
      year_joined < 2017 ~ "Veteran",
      year_joined >= 2017 & year_joined <= 2022 ~ "Intermediate",
      year_joined > 2022 ~ "New"
    )
  )
user_data <- subset(user_data, select = -year_joined) # remove year_joined column

head(user_data)
```

```
##   user_id review_count average_stars member_since member_category
## 1     u_0           32          2.08   2019-04-05    Intermediate
## 2     u_1           90          1.97   2015-11-15         Veteran
## 3     u_2           93          1.10   2021-10-05    Intermediate
## 4     u_3           59          3.01   2017-05-15    Intermediate
## 5     u_4           42          4.44   2021-04-05    Intermediate
## 6     u_5           62          1.63   2023-09-16             New
```

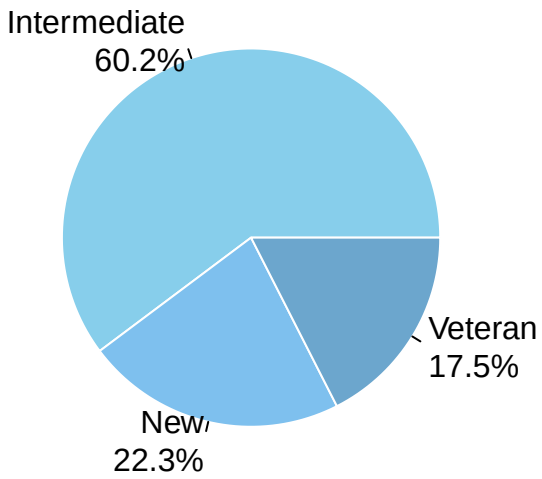
Data Exploratory

```r
summary(user_data)
```

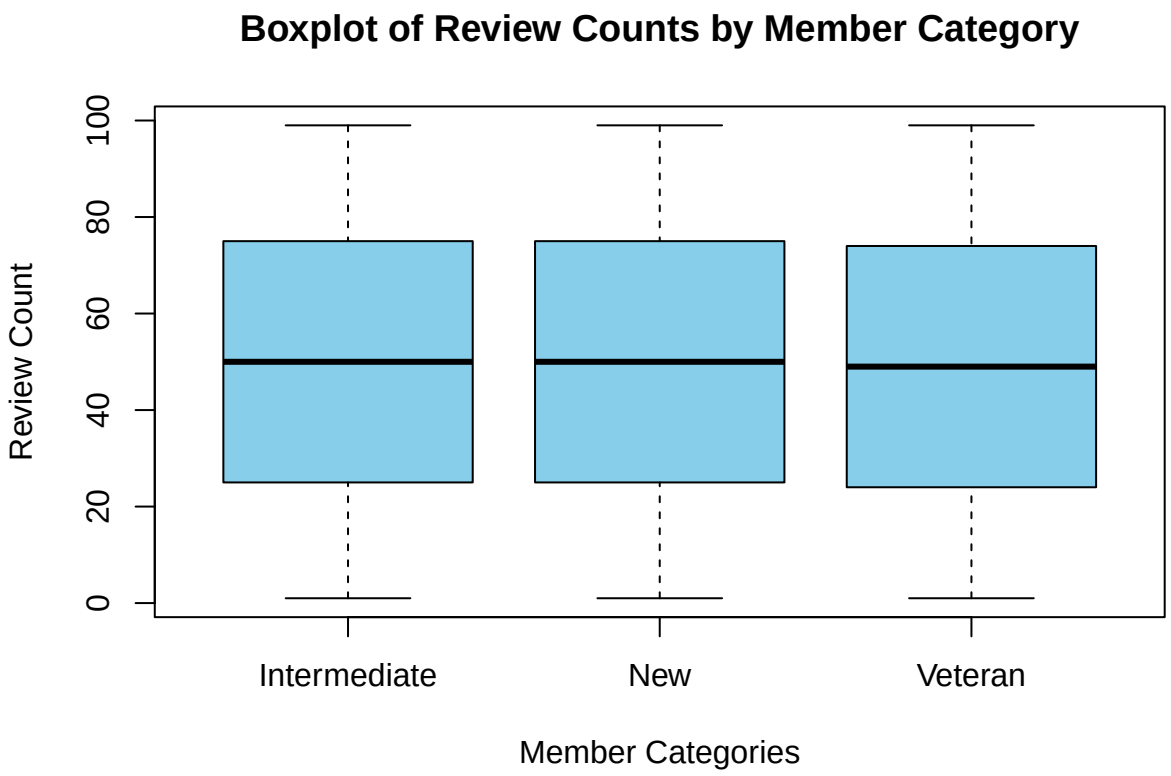
```
##    user_id            review_count   average_stars   member_since
##  Length:37640       Min.   : 1.00   Min.   :1.000   Length:37640
##  Class :character   1st Qu.:25.00   1st Qu.:2.000   Class :character
##  Mode  :character   Median :50.00   Median :3.000   Mode  :character
##                     Mean   :49.96   Mean   :2.997
##                     3rd Qu.:75.00   3rd Qu.:3.990
##                     Max.   :99.00   Max.   :5.000
##  member_category
##  Length:37640
##  Class :character
##  Mode  :character
##
```

```
##
##
```

```r
# Calculate the frequency of each member category
category_counts <- table(user_data$member_category)

# Calculate the percentage for each category
percentages <- round(100 * category_counts / sum(category_counts), 1)

# Create a new label that includes both the category name and its percentage
labels_with_percentages <- paste(names(category_counts), "\n", percentages, "%", sep = "")

colors <- c("skyblue", "skyblue2","skyblue3")

pie(category_counts,
    labels = labels_with_percentages,
    main = "Distribution of Member Categories",
    col = colors,
    border = "white",
    cex = 1) # font size
```

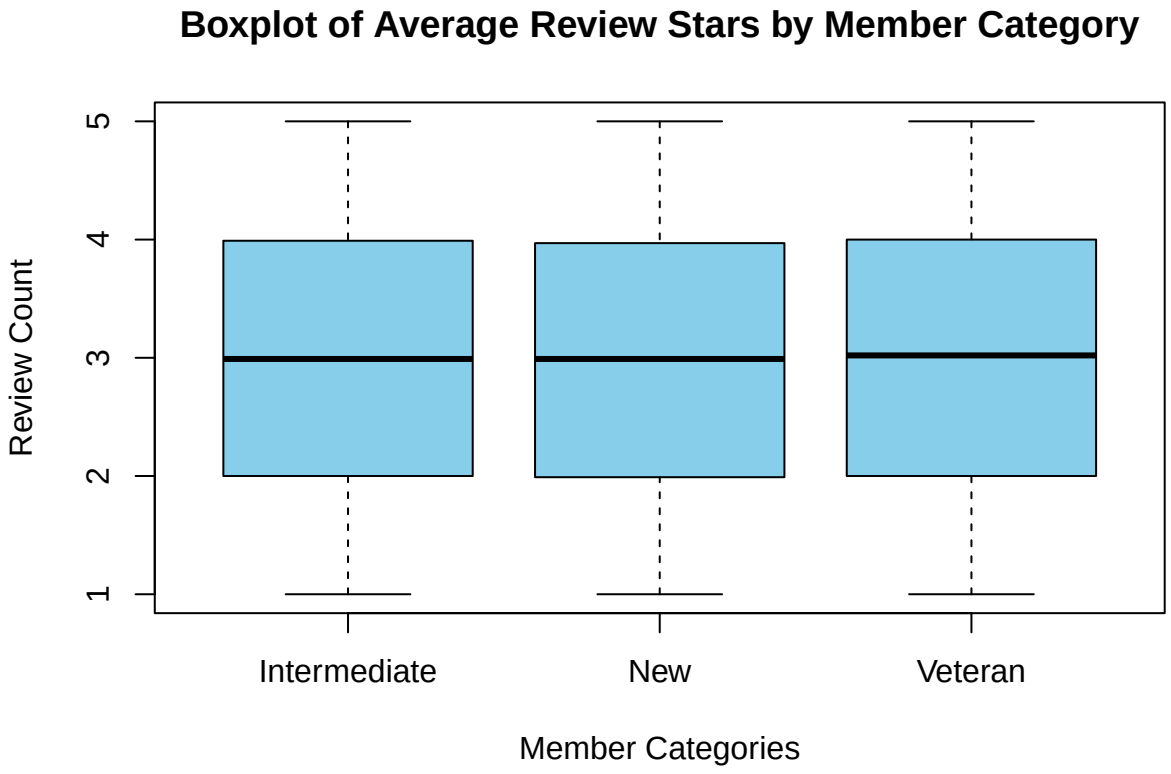



```r
boxplot(review_count ~ member_category, # formula = y - group
        data = user_data,
        xlab = "Member Categories",
        ylab = "Review Count",
        main = "Boxplot of Review Counts by Member Category",
        col = "skyblue",
        border = "black"
)
```



This plot indicates that all three categories have a similar range and median of review counts, suggesting no major difference in review activity based on member category.

```
boxplot(average_stars ~ member_category, # formula = y - group
        data = user_data,
        xlab = "Member Categories",
        ylab = "Review Count",
        main = "Boxplot of Average Review Stars by Member Category",
        col = "skyblue",
        border = "black"
)
```




```r
summary_table <- user_data %>%
  group_by(member_category) %>%
  summarise( # calcullation for each categories
    user_count = n(), # calculate user count
    avg_review_count = mean(review_count), # averaging rata-rata review_count
    avg_average_stars = mean(average_stars) # averaging average_stars
  ) %>%
  # rename columns
  rename(
    "Member Categories" = member_category,
    "Users" = user_count,
    "Average Review Count" = avg_review_count,
    "Average Review Stars" = avg_average_stars
  )

kable(summary_table,
      caption = "User Data Summary by Member Category",
      format = "pipe",
      align = "c")
```

Table 1: User Data Summary by Member Category

Member Categories	Users	Average Review Count	Average Review Stars
Intermediate	22671	50.17582	2.997428
New	8382	49.99463	2.987888
Veteran	6587	49.15576	3.004920

The conclusion is that there are no significant differences between each user group. However, there is a considerable difference in the number of users in each group.