Sample Data Analysis Project (Cycling Ridership Analysis)

In this project, we will import data collected by a bikeshare company in NYC. The business task at hand is to examine the relationship/difference between casual riders and members

Data Wrangling

Loading the required packages and importing data into Rstudio

```
library(tidyverse)
library(lubridate)
library(readr)
```

```
january_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202101-divvy-tripdata
february_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202102-divvy-tripdat
march_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202103-divvy-tripdata.cs
april_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202104-divvy-tripdata.cs
may_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202105-divvy-tripdata.csv
june_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202106-divvy-tripdata.csv
july_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202107-divvy-tripdata.csv
august_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202108-divvy-tripdata.
september_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202109-divvy-tripda
october_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202110-divvy-tripdata
november_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202111-divvy-tripdat
december_data <- read_csv("C:/Users/Stai Ndirangu/Desktop/Divvy Data/2021 CSV Data/202112-divvy-tripdat
```

Next we combine the imported dataframes into one big dataframe

```
whole_year_data <- bind_rows(january_data ,february_data ,march_data ,april_data ,may_data ,june_data ,
```

Now we separate the 'started_at' column into year, month, day of week

```
whole_year_data$date <- as.Date(whole_year_data$started_at)
whole_year_data$month <- format(as.Date(whole_year_data$date), "%m")
whole_year_data$day <- format(as.Date(whole_year_data$date), "%d")
whole_year_data$year <- format(as.Date(whole_year_data$date), "%Y")
whole_year_data$day_of_week <- format(as.Date(whole_year_data$date), "%A")
```

Then we add a column to include ride length

```
whole_year_data$ride_length <-difftime(whole_year_data$ended_at,whole_year_data$started_at)
```

Next step is to check data type of ride length column and converting it from factor to numeric so that we can run calculations on the data

```
is.factor(whole_year_data$ride_length)
whole_year_data$ride_length <- as.numeric(as.character(whole_year_data$ride_length))
is.numeric(whole_year_data$ride_length)
```

Final step in cleaning the data is checking for and removing bad data i.e negative trips and testing trips

```
negative_ride_length <- whole_year_data %>% count(ride_length<0)
whole_year_data_v2 <- whole_year_data[!(whole_year_data$ride_length<0),]
```

At this stage we can investigate our data and start gaining some insights. Lets do some analysis on the clean data frame on ride length variable and compare the descriptive stats of casuals and members

```
summary(whole_year_data_v2$ride_length)
aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual, FUN = mean)
aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual, FUN = median)
aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual, FUN = max)
aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual, FUN = min)
```

We can go further by computing average ride time by day for members vs casual users

```
aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual + whole_year_data_v2$day_of_
```

The order of the days of the week in the values generated above is wrong so lets correct that and run the fuction again

```
whole_year_data_v2$day_of_week <- ordered(whole_year_data_v2$day_of_week, levels=c("Sunday", "Monday", "
```

```
counts <- aggregate(whole_year_data_v2$ride_length ~ whole_year_data_v2$member_casual + whole_year_data_
```

Since our data is now well cleaned and ready for analysis, lets analyze ridership data by type and weekday

```
analysis_results <- whole_year_data_v2 %>%
  mutate(weekday=wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides=n(), average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)
```

Visualize ridership data by member type

With our data now properly sorted and a final dataframe generated for final analysis, lets first load ggplot2 and then create a visual for ridership data by member type

```
library(ggplot2)

ggplot(data = analysis_results, aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  geom_col(position = "dodge")
```

We can also visualize ridership data by average duration

```
ggplot(data = analysis_results, aes(x = weekday, y = average_duration, fill = member_casual))+
  geom_col(position = "dodge")
```

Exporting the csv files for further analysis

More visualization and creation of reports will be done later and on other tools in this case Power Bi or Tableau. To do that, we copy the dataframes developed here into csv files and save them in our local computer

```
write.csv(analysis_results, file = "C:\\Users\\Stai Ndirangu\\Desktop\\Divvy Data\\Analysis results\\ana
write.csv(counts, file = "C:\\Users\\Stai Ndirangu\\Desktop\\Divvy Data\\Analysis results\\counts.csv")
```