# Installation of packages and being able to use them

install.packages("tidyverse")

install.packages("tidyr")

install.packages("readr")

install.packages("dplyr")

install.packages("ggplot2")

install.packages("lubridate")

install.packages("ggpubr")

library(tidyverse)

library(tidyr)

library(readr)

library(dplyr)

library(ggplot2)

library(lubridate)

library(ggpubr)

# Uploading and importing of data sets

daily\_activity <- read.csv("Fitabase\_Data/dailyActivity\_merged.csv")

daily\_calories <- read.csv("Fitabase\_Data/dailyCalories\_merged.csv")

daily\_intensities <- read.csv("Fitabase\_Data/dailyIntensities\_merged.csv")

daily\_steps <- read.csv("Fitabase\_Data/dailySteps\_merged.csv")

hourly\_calories <- read.csv("Fitabase\_Data/hourlyCalories\_merged.csv")

hourly\_intensities <- read.csv("Fitabase\_Data/hourlyIntensities\_merged.csv")

hourly\_steps <- read.csv("Fitabase\_Data/hourlySteps\_merged.csv")

daily\_sleep <- read.csv("Fitabase\_Data/sleepDay\_merged.csv")

weight\_logs <- read.csv("Fitabase\_Data/weightLogInfo\_merged.csv")

# Determining structure of each data set

str(daily\_activity)

str(daily\_calories)

str(daily\_intensities)

str(daily\_sleep)

str(daily\_steps)

str(hourly\_calories)

str(hourly\_intensities)

str(hourly\_steps)

str(weight\_logs)

n\_distinct(daily\_activity$Id)

n\_distinct(daily\_calories$Id)

n\_distinct(daily\_intensities$Id)

n\_distinct(daily\_sleep$Id)

n\_distinct(daily\_steps$Id)

n\_distinct(hourly\_calories$Id)

n\_distinct(hourly\_intensities$Id)

n\_distinct(hourly\_steps$Id)

n\_distinct(weight\_logs$Id)

# Cleaning of each data set

# Reformatting dates

#daily\_sleep

daily\_sleep$SleepDay = as.POSIXct(daily\_sleep$SleepDay, format = "%m/%d/%Y %I:%M:%S %p",tz=Sys.timezone())

daily\_sleep$ActivityDay <- format(daily\_sleep$SleepDay, format = "%m/%d/%Y")

# hourly\_calories

hourly\_calories$ActivityHour = as.POSIXct(hourly\_calories$ActivityHour, format = "%m/%d/%Y %I:%M:%S %p",tz=Sys.timezone())

hourly\_calories$date <- format(hourly\_calories$ActivityHour, format = "%m/%d/%Y")

hourly\_calories$time <- format(hourly\_calories$ActivityHour, format = "%H/%M/%S")

# hourly\_intensities

hourly\_intensities$ActivityHour = as.POSIXct(hourly\_intensities$ActivityHour, format = "%m/%d/%Y %I:%M:%S %p",tz=Sys.timezone())

hourly\_intensities$date <- format(hourly\_intensities$ActivityHour, format = "%m/%d/%Y")

hourly\_intensities$time <- format(hourly\_intensities$ActivityHour, format = "%H/%M/%S")

# hourly\_steps

hourly\_steps$ActivityHour = as.POSIXct(hourly\_steps$ActivityHour, format = "%m/%d/%Y %I:%M:%S %p",tz=Sys.timezone())

hourly\_steps$date <- format(hourly\_steps$ActivityHour, format = "%m/%d/%Y")

hourly\_steps$time <- format(hourly\_steps$ActivityHour, format = "%H/%M/%S")

# weight\_logs

weight\_logs$Date = as.POSIXct(weight\_logs$Date, format = "%m/%d/%Y %I:%M:%S %p",tz=Sys.timezone())

weight\_logs$date <- format(weight\_logs$Date, format = "%m/%d/%Y")

weight\_logs$time <- format(weight\_logs$Date, format = "%H/%M/%S")

# Analysis of the Data

#Summary of daily\_activity & daily\_sleep data sets

summary(daily\_activity)

summary(daily\_sleep)

mean(daily\_sleep$TotalMinutesAsleep/60)

sum(daily\_activity$SedentaryMinutes)

sum(daily\_activity$LightlyActiveMinutes)

sum(daily\_activity$FairlyActiveMinutes)

sum(daily\_activity$VeryActiveMinutes)

#Determining times of the day at which users are typically are more active

ggplot(data=hourly\_steps, aes(time,StepTotal)) + geom\_col() + labs(x ="Time (hours)", y ="Total Steps Taken (all users)") + theme(axis.text.x=element\_text(angle=90,hjust=1,vjust=0.5)) + ggtitle("Time vs. Total Steps")

# Determining correlation of Calories to Very Active Minutes

ggplot(data=daily\_activity, aes(VeryActiveMinutes, Calories)) + geom\_point(fill="blue") + stat\_cor(method = "pearson", x.label = 500, y.label = 175) + labs(x="Very Active Minutes",y="Calories") + ggtitle("Correlation of Calories to Very Active Minutes") + geom\_smooth()

# Determining correlation of Calories to TotalDistance

ggplot(data=daily\_activity, aes(TotalDistance, Calories)) + geom\_point(fill="blue") + stat\_cor(method = "pearson", x.label = 500, y.label = 175) + labs(x="Total Distance",y="Calories") + ggtitle("Correlation of Calories to Total Distance") + geom\_smooth()