```
######################################################################

# Project assignment - Haris Muhammad (s1037924)
# 27-10-2021

# Set my working directory
setwd("C:/Users/haris/OneDrive/Desktop/R_project")

######################################################################

# Exercise 1

# Read in the data
data <- read.csv("Raw_data/2021_COVID-19_casus.csv")

# Remove errors in the data set
#1 Checking for Duplicates in the Data Set
duplicated(data)

#2 Checking if Ja is in the Column Deceased
subset(data, Deceased == "Ja",)
#It seems to be that there are no rows with Ja in the column after using
the subset function.

data_without_errors <- unique(data)


######################################################################

# Exercise 2

#It is important to find out the structure of the data as well as summary
statistics
str(data_without_errors)
summary(data_without_errors)

data_without_errors$Agegroup <- as.factor(data_without_errors$Agegroup)

#Checking if Unknown Agegroup exists in the data set
subset(data_without_errors, Agegroup == "Unknown",)
#No "Unknown" age group

#As H_ostpital_admission and Deceased are characters, they need to be
converted into integers for future calculations.
data_without_errors$Deceased <- as.factor(data_without_errors$Deceased)
data_without_errors$Deceased <- as.integer(data_without_errors$Deceased)

#Deceased 1 = No, 2 = Unknow, 3 = Yes

data_without_errors$Hospital_admission <-
as.factor(data_without_errors$Hospital_admission)
data_without_errors$Hospital_admission <-
as.integer(data_without_errors$Hospital_admission)

#Hospital_admission 1 = No, 2 = Unknow, 3 = Yes

#Table for deceased
```

```
deceased_table <- table(data_without_errors$Deceased,
data_without_errors$Agegroup)

#barplot for deceased per Agegroup
barplot(deceased_table[3, ], main = "Deceased Per Age-group", xlab =
"Age-group", ylab = "Deceased", cex.axis = 1.5, cex.names = 1.5, cex.main
= 1.5, cex.lab = 1.5, legend = c(colnames(deceased_table)), col =
c("Green", "Blue", "Purple", "Red", "Yellow", "Orange"), ylim = c(0,
8000))

#Table for Hospitalized
admissioned_table <- table(data_without_errors$Hospital_admission,
data_without_errors$Agegroup)

#barplot for hospitalized per Agegroup
barplot(admissioned_table[3, ], main = "Hospitalized Per Age-group", xlab
= "Age-group", ylab = "Hospitalized", cex.axis = 1.5, cex.names = 1.5,
cex.main = 1.5, cex.lab = 1.5, legend = c(colnames(admissioned_table)),
col = c("Green", "Blue", "Purple", "Red", "Yellow", "Orange"), ylim =
c(0, 9000))

#Table for Cases
cases_table <- table(data_without_errors$Agegroup)

#barplot for cases per Agegroup
barplot(cases_table, main = "Cases Per Age-group", xlab = "Age-group",
ylab = "Cases", cex.axis = 1.5, cex.names = 1.5, cex.main = 1.5, cex.lab
= 1.5, ylim = c(0, 1450000), col = c("Green", "Blue", "Purple", "Red",
"Yellow", "Orange"))

#####################################################################

# Exercise 3

#Calculate Hospitalized Rate
data_without_errors$hospitalized_rate <-
sum(data_without_errors$Hospital_admission ==
3)/sum(data_without_errors$Day)

#Aggregation
x <- aggregate(hospitalized_rate ~ Agegroup, data_without_errors, sum)

#Barplot for hospitalized Rate
barplot(hospitalized_rate ~ Agegroup, data = x, main = "Hospitalized Rate
Per Age-group", xlab = "Age-group", ylab = "Hospitalized Rate", cex.axis
= 1.5, cex.names = 1.5, cex.main = 1.5, cex.lab = 1.5, ylim = c(0, 70),
col = c("Green", "Blue", "Purple", "Red", "Yellow", "Orange"))

#Calculate Deceased Rate
data_without_errors$deceased_rate <- sum(data_without_errors$Deceased ==
3)/sum(data_without_errors$Day)

#Aggregation
y <- aggregate(deceased_rate ~ Agegroup, data_without_errors, sum)

#Barplot for Deceased Rate
barplot(deceased_rate ~ Agegroup, data = y, main = "Deceased Rate Per
Age-group", xlab = "Age-group", ylab = "Deceased Rate", cex.axis = 1.5,
```

```
cex.names = 1.5, cex.main = 1.5, cex.lab = 1.5, ylim = c(0, 40), col =
c("Green", "Blue", "Purple", "Red", "Yellow", "Orange"))


#####################################################################

# Exercise 4

#Table cases
incidences_table <- table(data_without_errors$Day)

#Line plot cases over days
plot(incidences_table, type = 'l', ylim = c(0, 12000), main = "Cases
Since February 27 2020", xlab = "Cases", ylab = "Days", cex.lab = 1.2,
cex.axis = 1)

#Tables for hospitalized over days
admission_line <- table(data_without_errors$Day,
data_without_errors$Hospital_admission == 3)

#Table for deceased over days
deceased_line <- table(data_without_errors$Day,
data_without_errors$Deceased == 3)

#Line plot Hospitalized over days
plot(admission_line[, 2], type = 'l', main = "Deceased & Hospitalized
since February 27 2020", ylab = "Deceased & Hospitalized", xlab = "Days",
xlim = c(0, 550), ylim = c(0, 600), cex.lab = 1.2, cex.axis = 1, lwd = 2)

#To add line for deceased
lines(deceased_line[, 2], col = "Red", lwd = 2)
#To add a legend
legend ("topright", inset = 0.02, legend = c("Hospitalized", "Deceased"),
col = c("Black", "Red"), lty = 1)

#####################################################################

# Exercise 5

#Creating a column with only 1s
data_without_errors$none <- 1

#Cumulative number cases
cumsum_cases <- cumsum(data_without_errors$none)

#Cumulative line plot for cases
plot(data_without_errors$Day, cumsum_cases, type = 'l', main = "Cases
until 09/09/2021 ", ylab = "Cases", xlab = "Days", xlim = c(0, 550), ylim
= c(0, 2000000), cex.lab = 1.2, cex.axis = 1, lwd = 2)

#Cumulative number hospitalized
cumsum_hospitalized <- cumsum(data_without_errors$Hospital_admission ==
3)


#Cumulative number deceased
cumsum_deceased <- cumsum(data_without_errors$Deceased == 3)
```

```
#To add a line plot for hospitalized
plot(cumsum_hospitalized ~ data_without_errors$Day, type = 'l', main =
"Deceased & Hospitalized until 09/09/2021 ", ylab = "Deceased &
Hospitalized", xlab = "Days", xlim = c(0, 550), ylim = c(0, 40000),
cex.lab = 1.2, cex.axis = 1, lwd = 2)

#To add line for deceased
lines(cumsum_deceased ~ data_without_errors$Day, col = "Red", lwd = 2)

#To add a legend
legend ("bottomright", inset = 0.02, legend = c("Hospitalized",
"Deceased"), col = c("Black", "Red"), lty = 1)

################################################################

# Exercise 6

#Zuid Holland = most deceased
#Flevoland = least deceased
#Sme for cases

#Table for deceased
province_deceased_table <- aggregate(Deceased == 3 ~ Province,
data_without_errors, sum)

#Changing column names
colnames(province_deceased_table) <- c("Province", "Deceased")


#sorted table
tb <- sort(table(data_without_errors$Province,
data_without_errors$Deceased == 3), decreasing = TRUE)


#Barplot
barplot(tb[13:24], col = c("Black", "Red", "Pink", "Green", "Purple",
"Orange", "Yellow", "Darkgreen", "lightgreen", "Blue", "Lightblue",
"Grey"), main = "Deceased Per Province", xlab = "Province", ylab =
"Deceased", cex.axis = 1.2, cex.main = 1, cex.lab = 1.2, names.arg =
c("ZH", "NB", "NH", "GE", "LI", "UT", "OV", "FR", "DR", "GR", "ZE",
"FL"), cex.names = 1.2, ylim = c(0, 5000))

#Legned
legend ("topright", inset = 0.02, legend = c("Zuid Holland", "North
Brabant", "Noord-Holland", "Gelderland", "Limburg", "Utrect",
"Overijssel", "Frysland", "Drenthe", "Groningen", "Zeeland",
"Flevoland"), col = c("Black", "Red", "Pink", "Green", "Purple",
"Orange", "Yellow", "Darkgreen", "lightgreen", "Blue", "Lightblue",
"Grey"), lty = 1)

#Sorted Table for province cases
province_cases <- sort(table(data_without_errors$Province), decreasing =
TRUE)

#Barplot for province cases
barplot(province_cases, col = c("Black", "Red", "Pink", "Green",
"Purple", "Orange", "Yellow", "Darkgreen", "lightgreen", "Blue",
```

```
"Lightblue", "Grey"), main = "Cases Per Province", xlab = "Province",
ylab = "Cases", cex.axis = 1.2, cex.main = 1, cex.lab = 1.2, names.arg =
c("ZH", "NB", "NH", "GE", "LI", "UT", "OV", "FR", "DR", "GR", "ZE",
"FL"), cex.names = 1.2, ylim = c(0, 500000))

#Legend
legend ("topright", inset = 0.02, legend = c("Zuid Holland", "North
Brabant", "Noord-Holland", "Gelderland", "Limburg", "Utrect",
"Overijssel", "Frysland", "Drenthe", "Groningen", "Zeeland",
"Flevoland"), col = c("Black", "Red", "Pink", "Green", "Purple",
"Orange", "Yellow", "Darkgreen", "lightgreen", "Blue", "Lightblue",
"Grey"), lty = 1)

#####################################################################

# Exercise 7

#Table for cases
table_1 <- table(data_without_errors$Day)

table_2 <- table(data_without_errors$Day)

#Table for sequenced cases
table_2 <- table_2[seq(from = 1, to = 557)]

#Table for 4 0s
table_3 <- as.table(c(0, 0, 0, 0))

#Changing column names for the table_3
names(table_3) <- c("1", "2", "3", "4")

#Combining Tables
table_4 <- c(table_3, table_2)
table_4 <- as.table(table_4)

#Calculation for Rt
df_1 <- table_1/table_4

df_1 <- data.frame(df_1)
df_1 <- df_1[, 2][seq(from = 5, to = 561)]

#Plot for Rt
plot(df_1, type = 'l', main = "Reproduction Rate (Rt) by Cases", ylab =
"Reproduction Rate", xlab = "Days", xlim = c(0, 550), ylim = c(0, 6),
cex.lab = 1.2, cex.axis = 1, lwd = 2)

#####################################################################

# Exercise 8

#I would argue Netherlands is out of the most recent wave

#Table for hospitalized
t_1 <- table(data_without_errors$Day,
data_without_errors$Hospital_admission == 3)
t_1 <- as.table(t_1[,1])

#Table for sequenced hospitalized
```

```
t_2 <- t_1
t_2 <- t_2[seq(from = 1, to = 557)]

#Table for 4 0s
t_3 <- as.table(c(0, 0, 0, 0))

#Changing column names for the t_3
names(t_3) <- c("1", "2", "3", "4")

#Combining Tables
t_4 <- as.table(c(t_3, t_2))

#Calculation for Rt
df_2 <- t_1/t_4

df_2 <- data.frame(df_2)
df_2 <- df_2[, 2][seq(from = 5, to = 561)]


#Plot for Rt
plot(df_2, type = 'l', main = "Reproduction Rate (Rt) by Hospitalized",
ylab = "Reproduction Rate", xlab = "Days", xlim = c(0, 550), ylim = c(0,
6), cex.lab = 1.2, cex.axis = 1, lwd = 2)


#####################################################################

# Exercise 9

#Creating a column with only 1s
data_without_errors$none <- 1

#Adding a horizontal line
lines(data_without_errors$none, col = "Red", lwd = 2)


#####################################################################

# Exercise 10
```