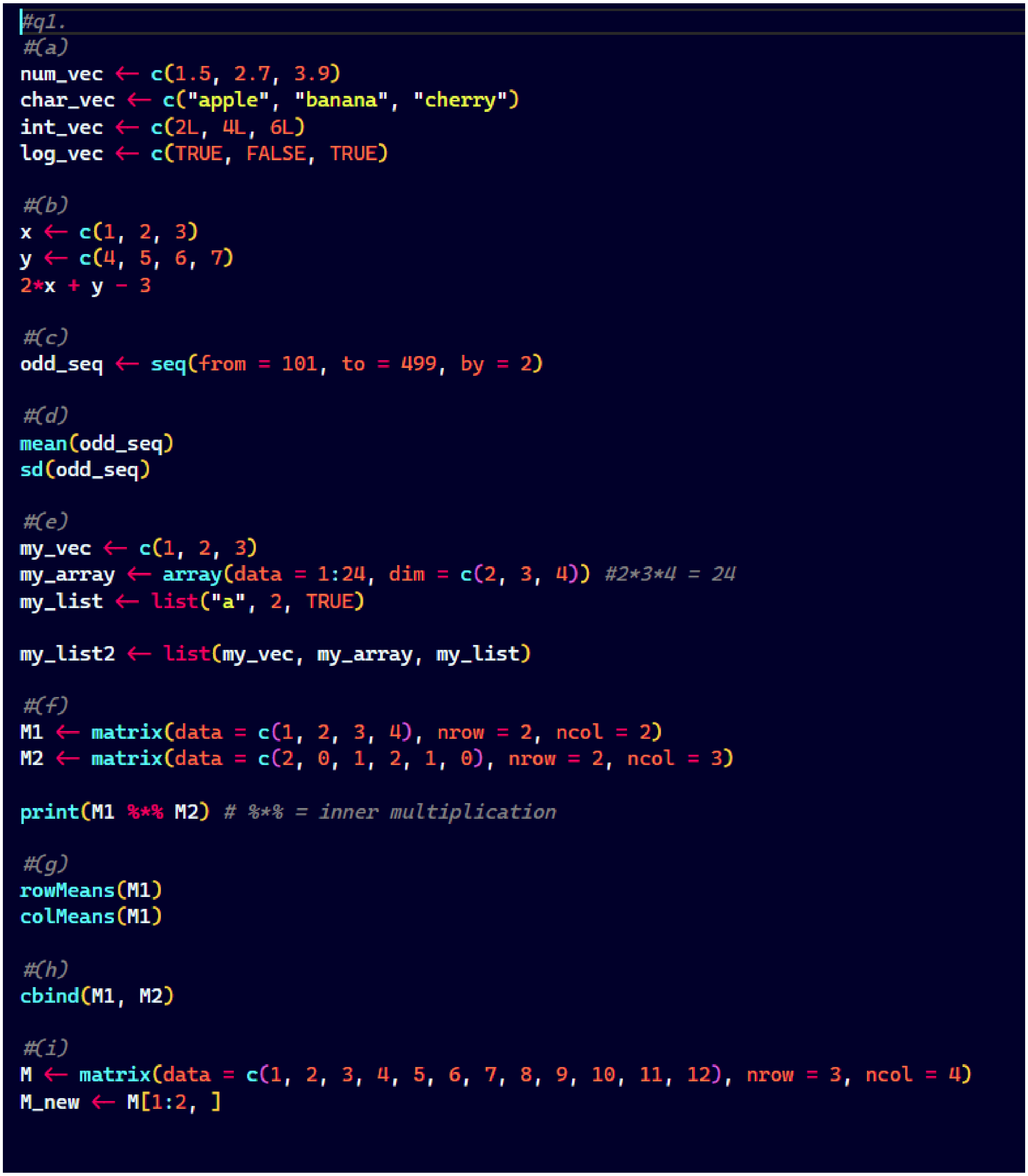
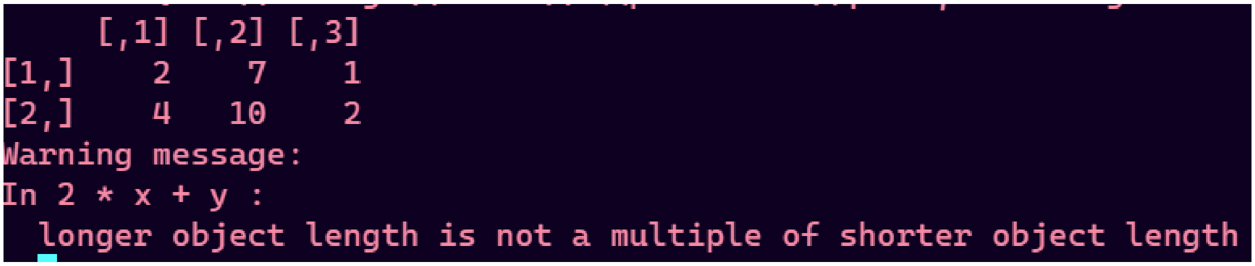
R Assignment

Aryan Sethi(21514)

Q1)

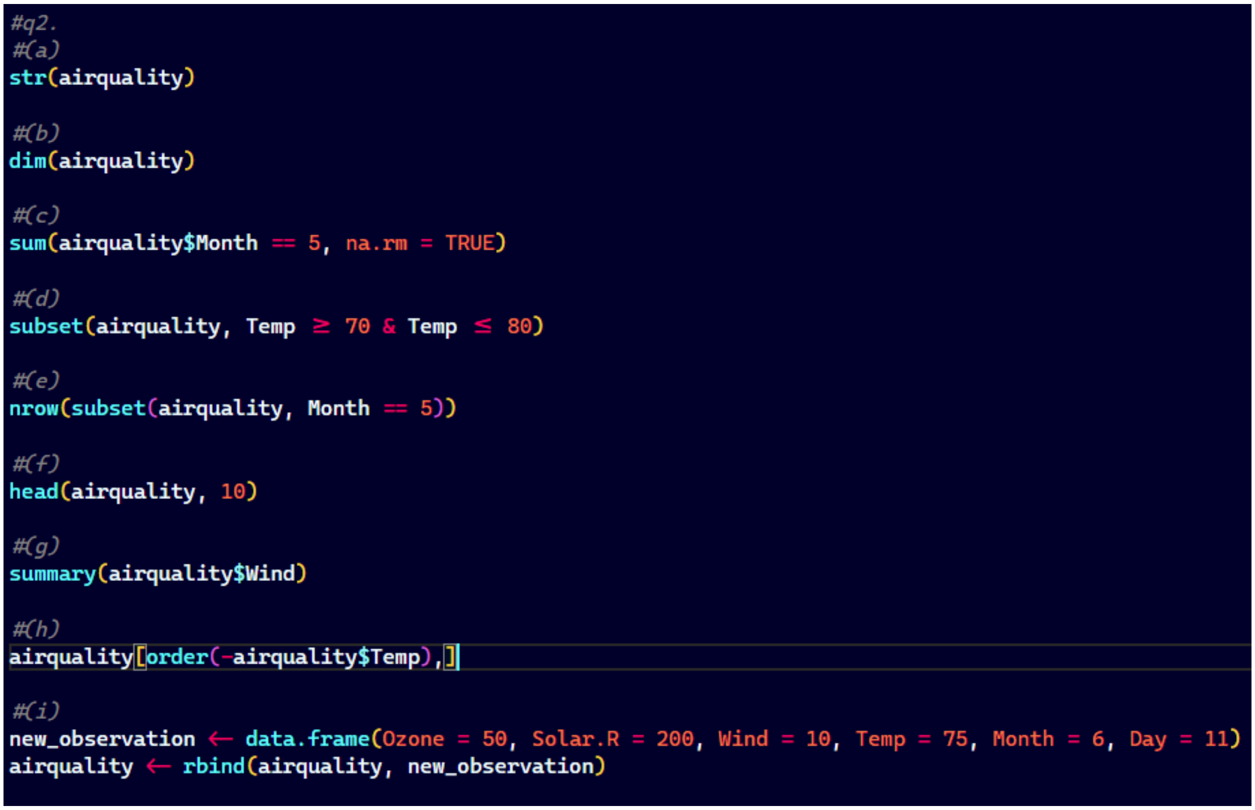


```r
#q1.
#(a)
num_vec <- c(1.5, 2.7, 3.9)
char_vec <- c("apple", "banana", "cherry")
int_vec <- c(2L, 4L, 6L)
log_vec <- c(TRUE, FALSE, TRUE)

#(b)
x <- c(1, 2, 3)
y <- c(4, 5, 6, 7)
2*x + y - 3

#(c)
odd_seq <- seq(from = 101, to = 499, by = 2)

#(d)
mean(odd_seq)
sd(odd_seq)

#(e)
my_vec <- c(1, 2, 3)
my_array <- array(data = 1:24, dim = c(2, 3, 4)) #2*3*4 = 24
my_list <- list("a", 2, TRUE)

my_list2 <- list(my_vec, my_array, my_list)

#(f)
M1 <- matrix(data = c(1, 2, 3, 4), nrow = 2, ncol = 2)
M2 <- matrix(data = c(2, 0, 1, 2, 1, 0), nrow = 2, ncol = 3)

print(M1 %*% M2) # %*% = inner multiplication

#(g)
rowMeans(M1)
colMeans(M1)

#(h)
cbind(M1, M2)

#(i)
M <- matrix(data = c(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12), nrow = 3, ncol = 4)
M_new <- M[1:2, ]
```

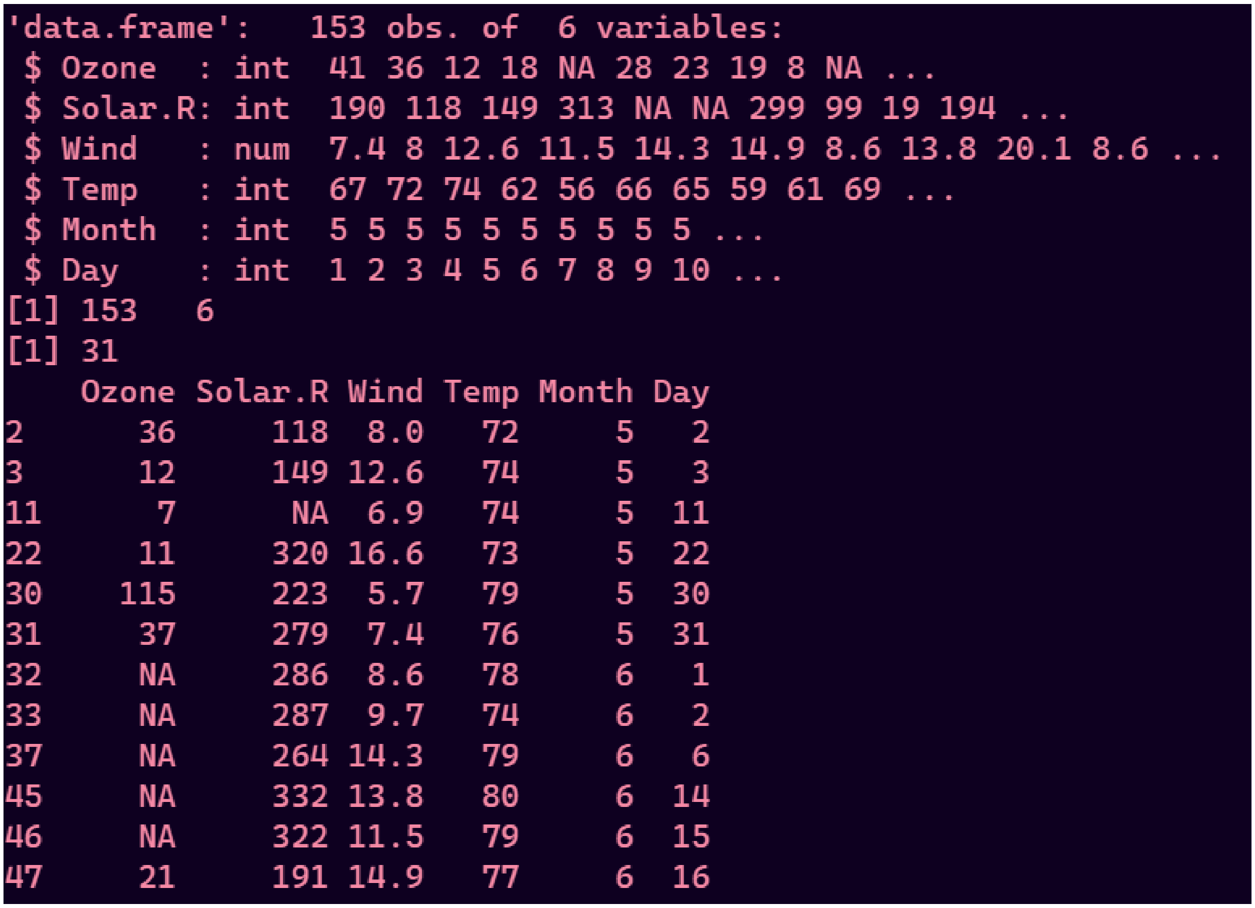


```
     [,1] [,2] [,3]
[1,]    2    7    1
[2,]    4   10    2
Warning message:
In 2 * x + y :
  longer object length is not a multiple of shorter object length
```

Q2)

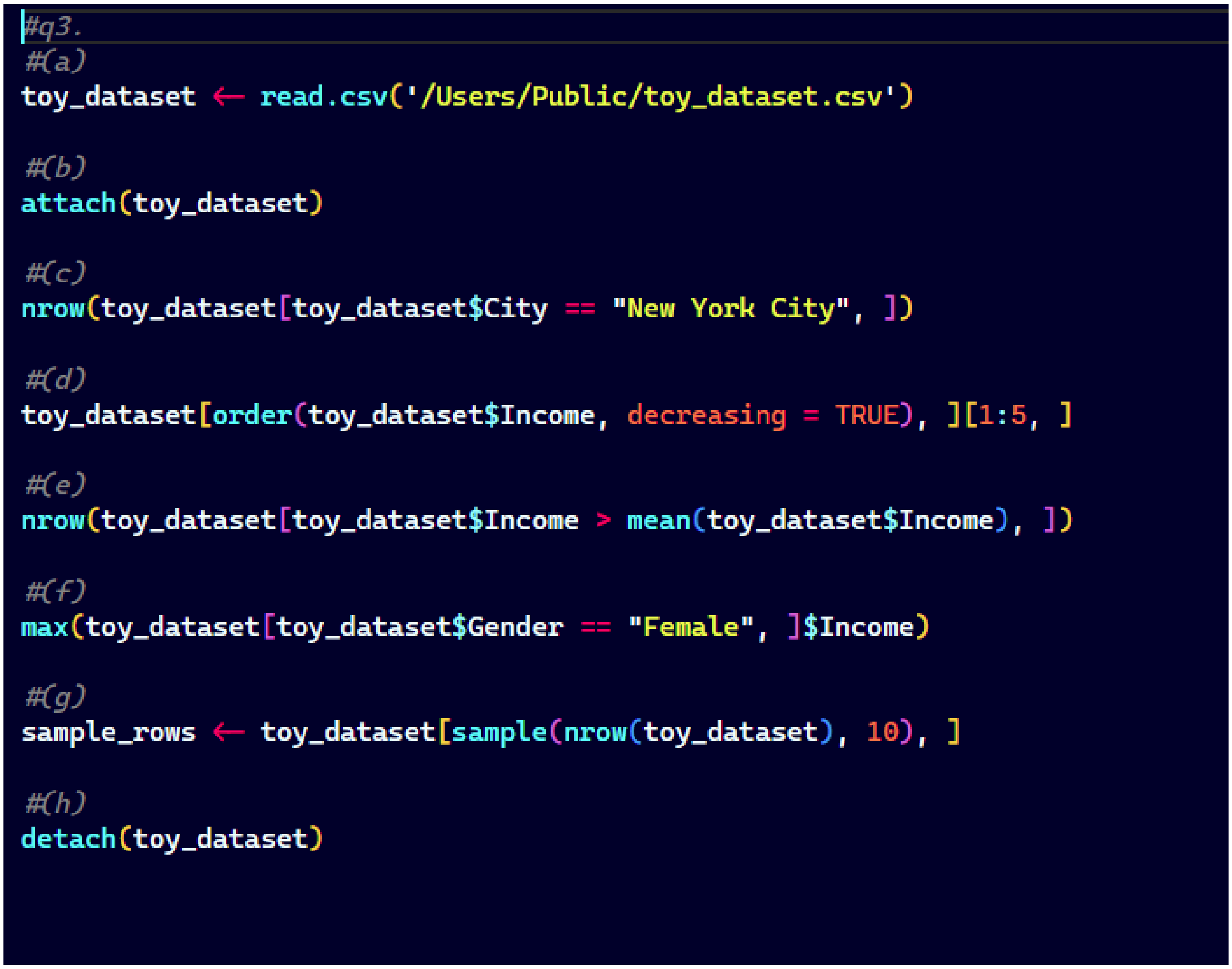


```
#q2.
#(a)
str(airquality)

#(b)
dim(airquality)

#(c)
sum(airquality$Month == 5, na.rm = TRUE)

#(d)
subset(airquality, Temp >= 70 & Temp <= 80)

#(e)
nrow(subset(airquality, Month == 5))

#(f)
head(airquality, 10)

#(g)
summary(airquality$Wind)

#(h)
airquality[order(-airquality$Temp),]

#(i)
new_observation <- data.frame(Ozone = 50, Solar.R = 200, Wind = 10, Temp = 75, Month = 6, Day = 11)
airquality <- rbind(airquality, new_observation)
```

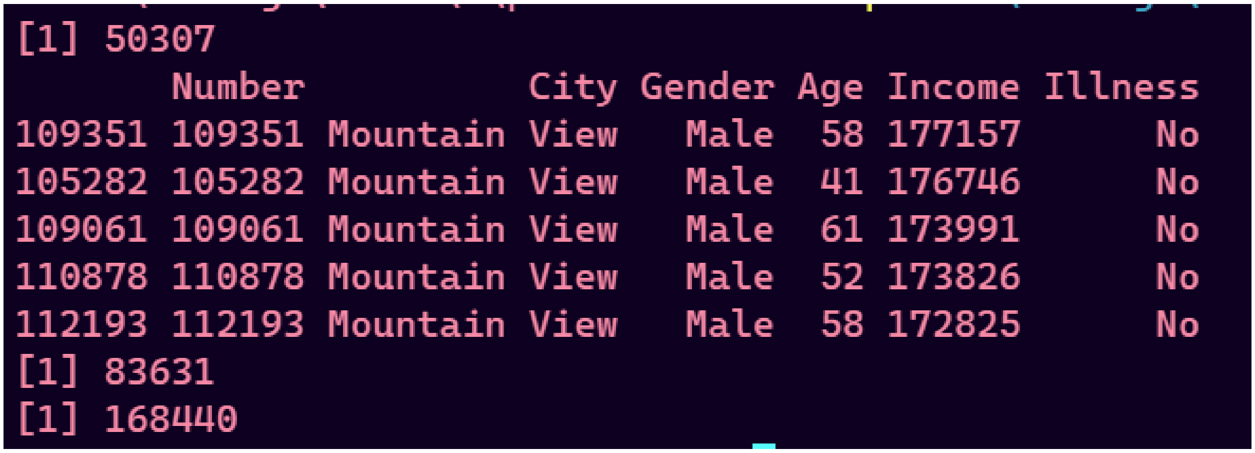


```
'data.frame':   153 obs. of  6 variables:
 $ Ozone  : int  41 36 12 18 NA 28 23 19 8 NA ...
 $ Solar.R: int  190 118 149 313 NA NA 299 99 19 194 ...
 $ Wind   : num  7.4 8 12.6 11.5 14.3 14.9 8.6 13.8 20.1 8.6 ...
 $ Temp   : int  67 72 74 62 56 66 65 59 61 69 ...
 $ Month  : int  5 5 5 5 5 5 5 5 5 5 ...
 $ Day    : int  1 2 3 4 5 6 7 8 9 10 ...
[1] 153   6
[1] 31
    Ozone Solar.R Wind Temp Month Day
2      36     118  8.0   72     5   2
3      12     149 12.6   74     5   3
11      7      NA  6.9   74     5  11
22     11     320 16.6   73     5  22
30    115     223  5.7   79     5  30
31     37     279  7.4   76     5  31
32     NA     286  8.6   78     6   1
33     NA     287  9.7   74     6   2
37     NA     264 14.3   79     6   6
45     NA     332 13.8   80     6  14
46     NA     322 11.5   79     6  15
47     21     191 14.9   77     6  16
```

Q3)

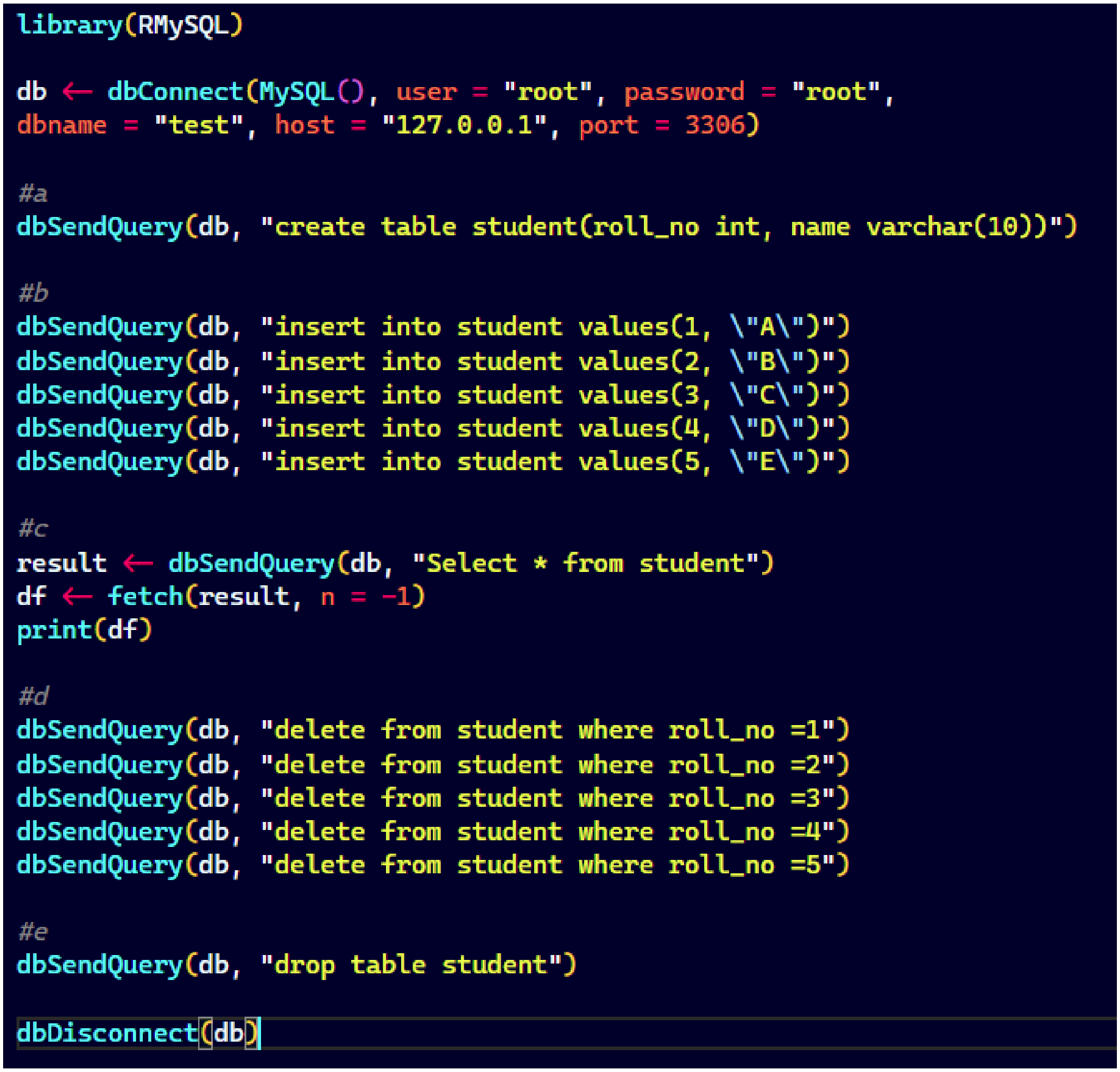


```r
#q3.
#(a)
toy_dataset <- read.csv('/Users/Public/toy_dataset.csv')

#(b)
attach(toy_dataset)

#(c)
nrow(toy_dataset[toy_dataset$City == "New York City", ])

#(d)
toy_dataset[order(toy_dataset$Income, decreasing = TRUE), ][1:5, ]

#(e)
nrow(toy_dataset[toy_dataset$Income > mean(toy_dataset$Income), ])

#(f)
max(toy_dataset[toy_dataset$Gender == "Female", ]$Income)

#(g)
sample_rows <- toy_dataset[sample(nrow(toy_dataset), 10), ]

#(h)
detach(toy_dataset)
```



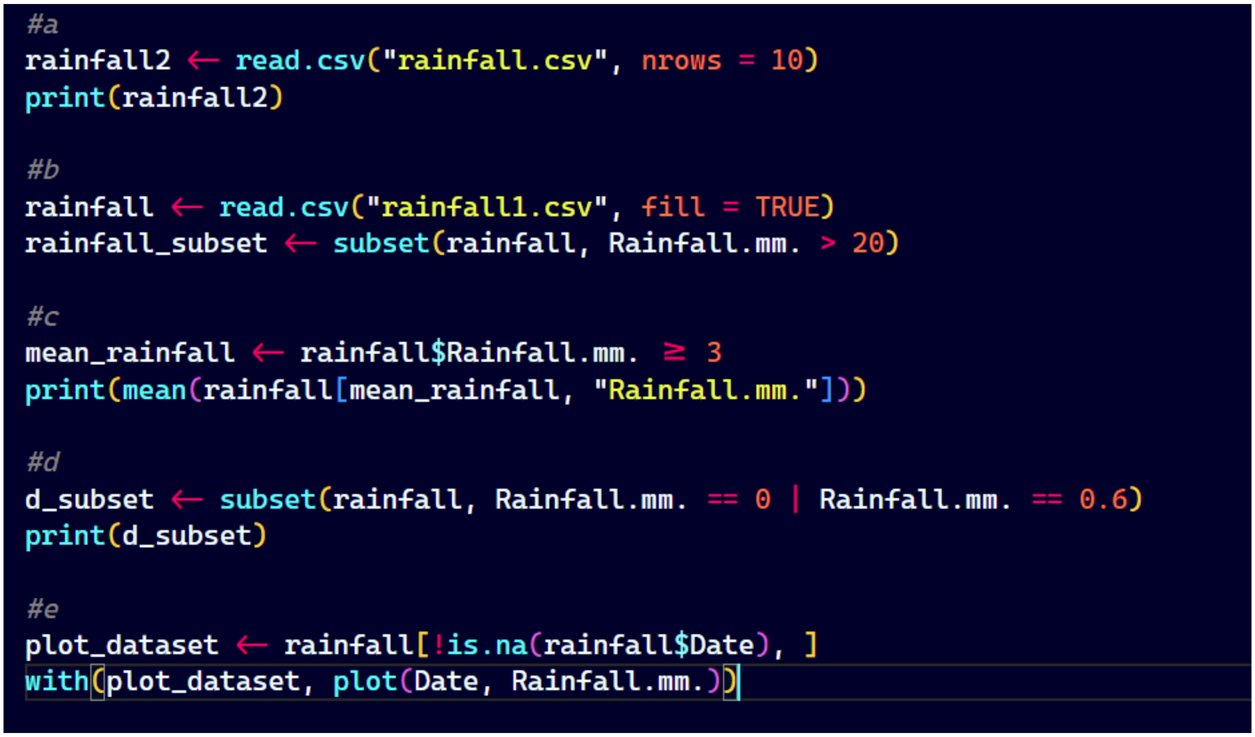
```
[1] 50307
       Number          City Gender Age Income Illness
109351 109351 Mountain View   Male  58 177157      No
105282 105282 Mountain View   Male  41 176746      No
109061 109061 Mountain View   Male  61 173991      No
110878 110878 Mountain View   Male  52 173826      No
112193 112193 Mountain View   Male  58 172825      No
[1] 83631
[1] 168440
```

Q4)

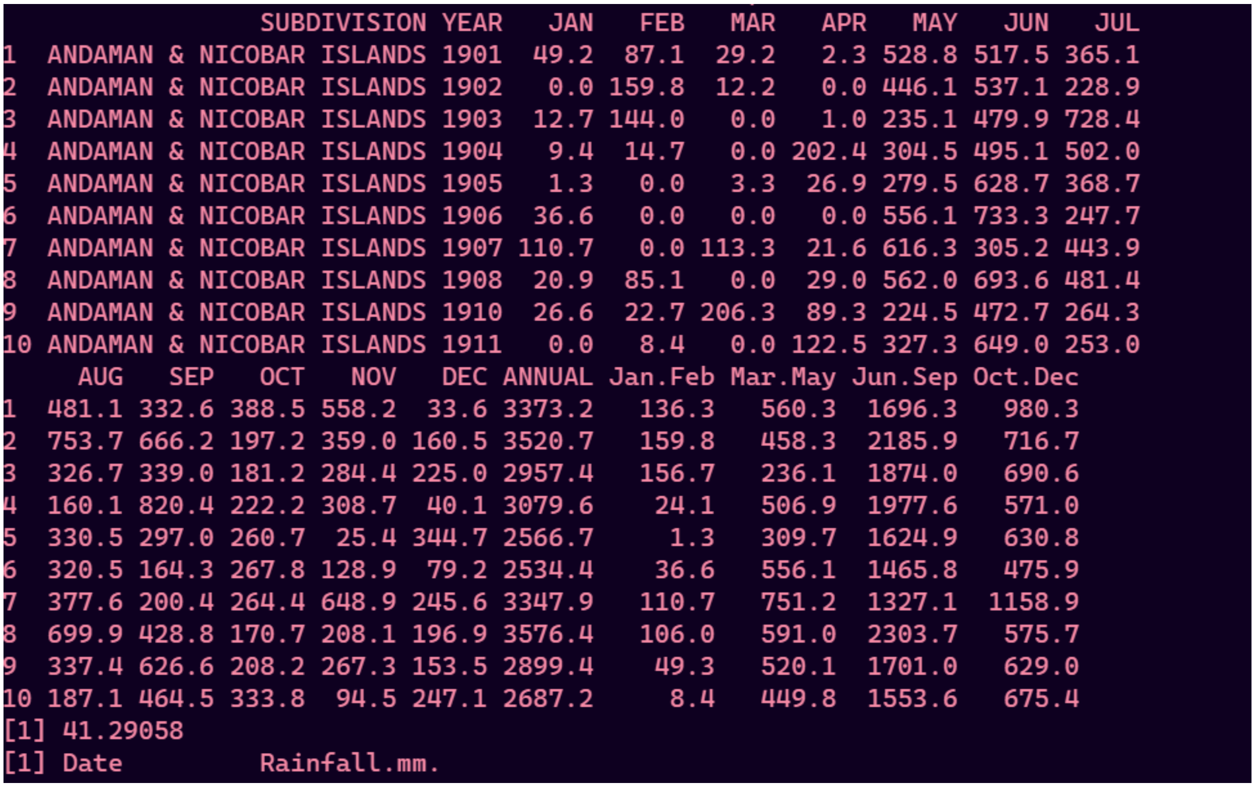


```
library(RMySQL)

db <- dbConnect(MySQL(), user = "root", password = "root",
dbname = "test", host = "127.0.0.1", port = 3306)

#a
dbSendQuery(db, "create table student(roll_no int, name varchar(10))")

#b
dbSendQuery(db, "insert into student values(1, \"A\")")
dbSendQuery(db, "insert into student values(2, \"B\")")
dbSendQuery(db, "insert into student values(3, \"C\")")
dbSendQuery(db, "insert into student values(4, \"D\")")
dbSendQuery(db, "insert into student values(5, \"E\")")

#c
result <- dbSendQuery(db, "Select * from student")
df <- fetch(result, n = -1)
print(df)

#d
dbSendQuery(db, "delete from student where roll_no =1")
dbSendQuery(db, "delete from student where roll_no =2")
dbSendQuery(db, "delete from student where roll_no =3")
dbSendQuery(db, "delete from student where roll_no =4")
dbSendQuery(db, "delete from student where roll_no =5")

#e
dbSendQuery(db, "drop table student")

dbDisconnect(db)
```

Q5)

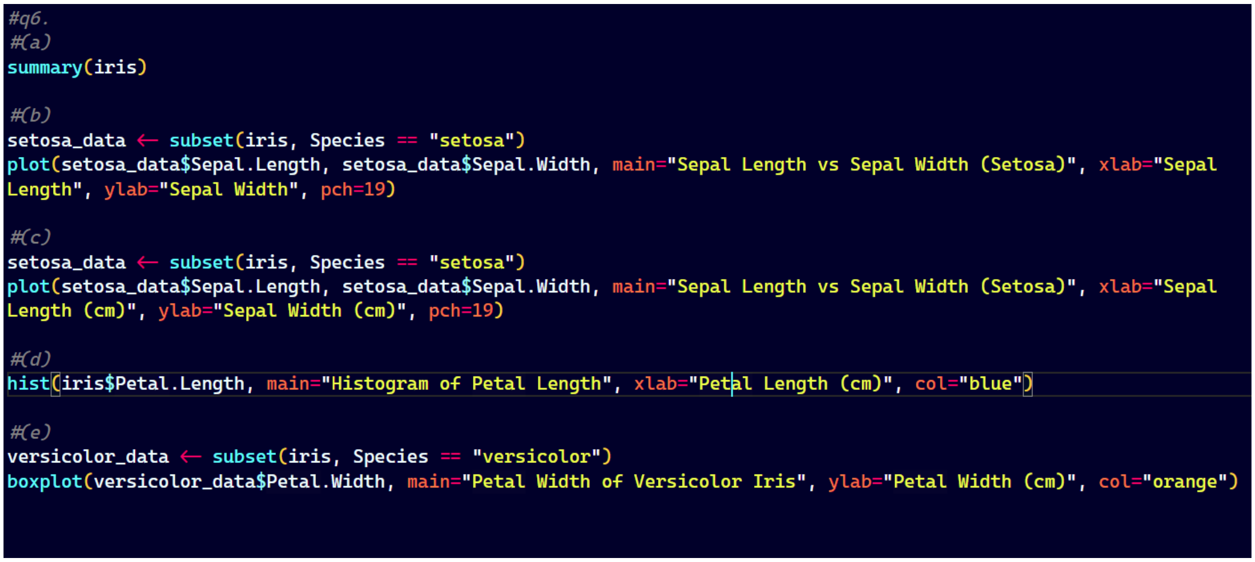


```r
#a
rainfall2 <- read.csv("rainfall.csv", nrows = 10)
print(rainfall2)

#b
rainfall <- read.csv("rainfall1.csv", fill = TRUE)
rainfall_subset <- subset(rainfall, Rainfall.mm. > 20)

#c
mean_rainfall <- rainfall$Rainfall.mm. >= 3
print(mean(rainfall[mean_rainfall, "Rainfall.mm."]))

#d
d_subset <- subset(rainfall, Rainfall.mm. == 0 | Rainfall.mm. == 0.6)
print(d_subset)

#e
plot_dataset <- rainfall[!is.na(rainfall$Date), ]
with(plot_dataset, plot(Date, Rainfall.mm.))
```

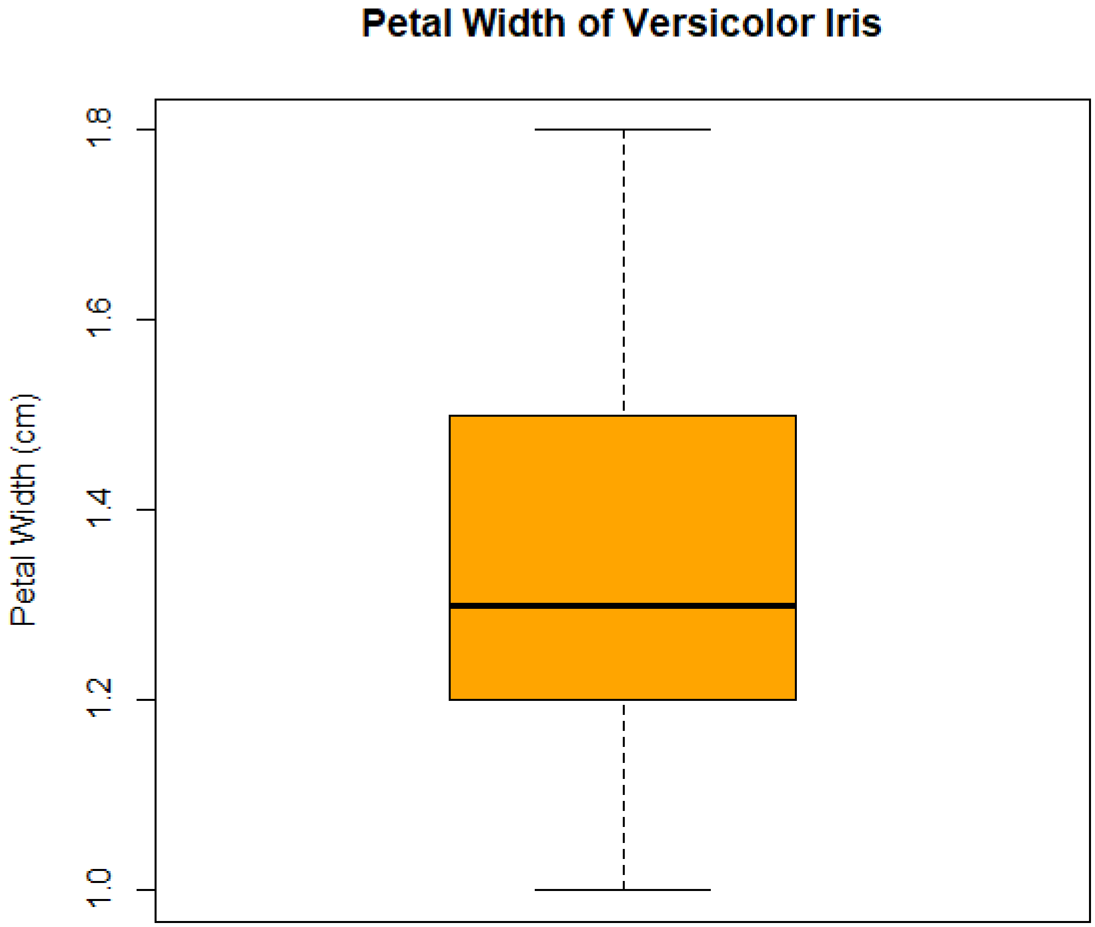


```
                 SUBDIVISION YEAR   JAN   FEB   MAR   APR   MAY   JUN   JUL
1  ANDAMAN & NICOBAR ISLANDS 1901  49.2  87.1  29.2   2.3 528.8 517.5 365.1
2  ANDAMAN & NICOBAR ISLANDS 1902   0.0 159.8  12.2   0.0 446.1 537.1 228.9
3  ANDAMAN & NICOBAR ISLANDS 1903  12.7 144.0   0.0   1.0 235.1 479.9 728.4
4  ANDAMAN & NICOBAR ISLANDS 1904   9.4  14.7   0.0 202.4 304.5 495.1 502.0
5  ANDAMAN & NICOBAR ISLANDS 1905   1.3   0.0   3.3  26.9 279.5 628.7 368.7
6  ANDAMAN & NICOBAR ISLANDS 1906  36.6   0.0   0.0   0.0 556.1 733.3 247.7
7  ANDAMAN & NICOBAR ISLANDS 1907 110.7   0.0 113.3  21.6 616.3 305.2 443.9
8  ANDAMAN & NICOBAR ISLANDS 1908  20.9  85.1   0.0  29.0 562.0 693.6 481.4
9  ANDAMAN & NICOBAR ISLANDS 1910  26.6  22.7 206.3  89.3 224.5 472.7 264.3
10 ANDAMAN & NICOBAR ISLANDS 1911   0.0   8.4   0.0 122.5 327.3 649.0 253.0
     AUG   SEP   OCT   NOV   DEC ANNUAL Jan.Feb Mar.May Jun.Sep Oct.Dec
1  481.1 332.6 388.5 558.2  33.6 3373.2   136.3   560.3  1696.3   980.3
2  753.7 666.2 197.2 359.0 160.5 3520.7   159.8   458.3  2185.9   716.7
3  326.7 339.0 181.2 284.4 225.0 2957.4   156.7   236.1  1874.0   690.6
4  160.1 820.4 222.2 308.7  40.1 3079.6    24.1   506.9  1977.6   571.0
5  330.5 297.0 260.7  25.4 344.7 2566.7     1.3   309.7  1624.9   630.8
6  320.5 164.3 267.8 128.9  79.2 2534.4    36.6   556.1  1465.8   475.9
7  377.6 200.4 264.4 648.9 245.6 3347.9   110.7   751.2  1327.1  1158.9
8  699.9 428.8 170.7 208.1 196.9 3576.4   106.0   591.0  2303.7   575.7
9  337.4 626.6 208.2 267.3 153.5 2899.4    49.3   520.1  1701.0   629.0
10 187.1 464.5 333.8  94.5 247.1 2687.2     8.4   449.8  1553.6   675.4
[1] 41.29058
[1] Date         Rainfall.mm.
```

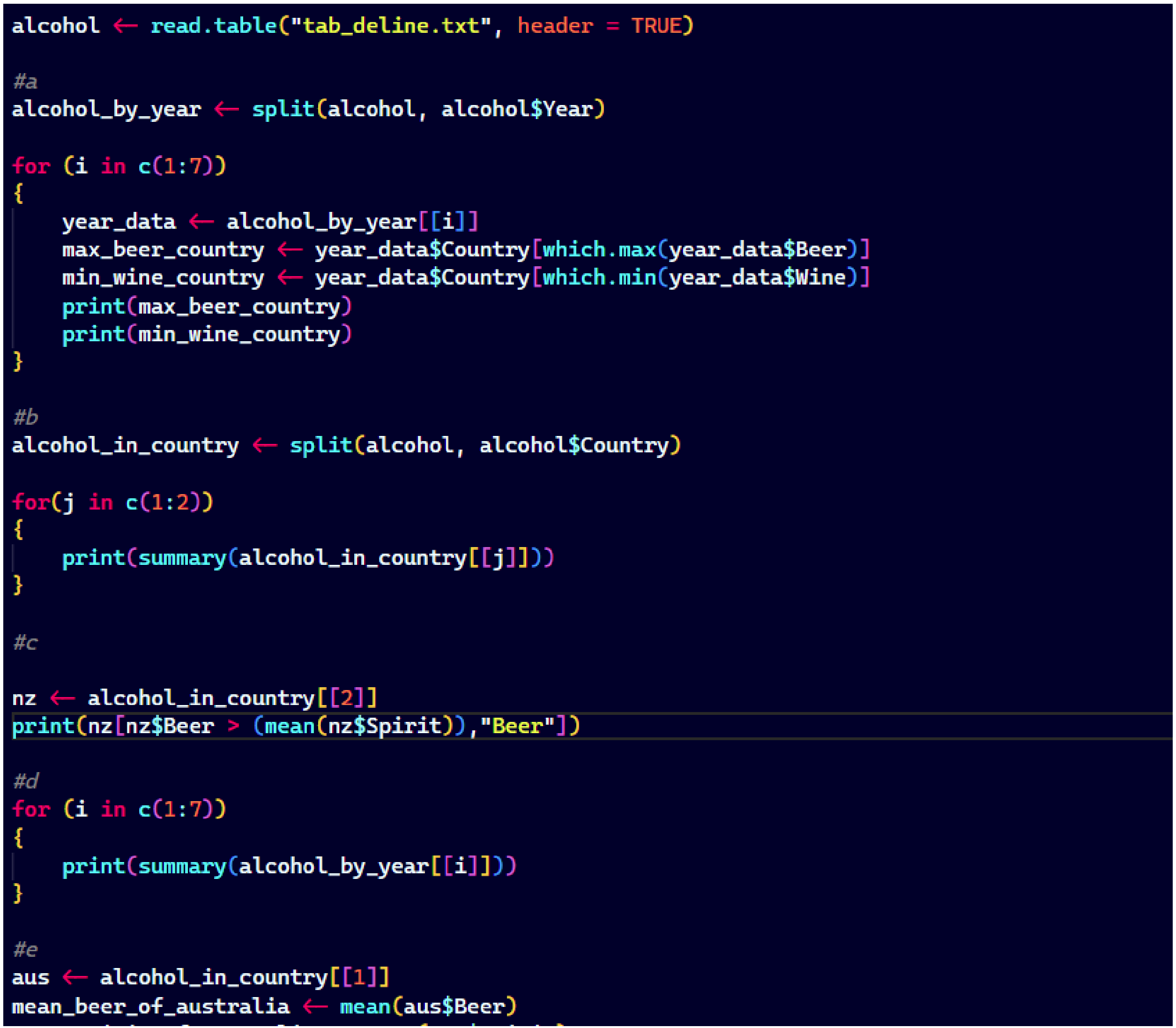
Q6)



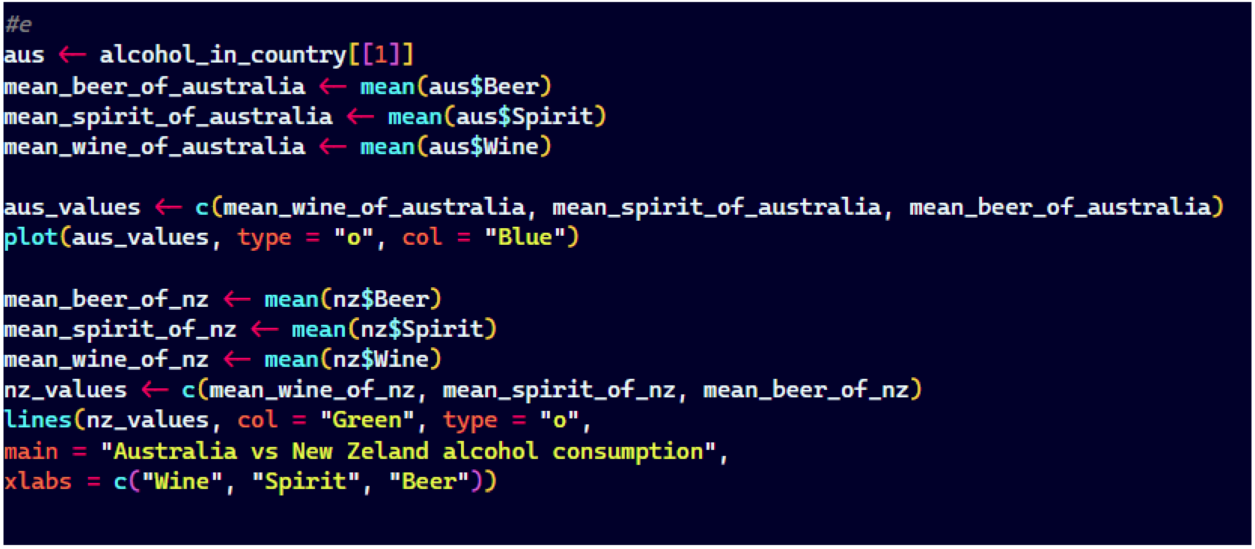
```r
#q6.
#(a)
summary(iris)

#(b)
setosa_data <- subset(iris, Species == "setosa")
plot(setosa_data$Sepal.Length, setosa_data$Sepal.Width, main="Sepal Length vs Sepal Width (Setosa)", xlab="Sepal
Length", ylab="Sepal Width", pch=19)

#(c)
setosa_data <- subset(iris, Species == "setosa")
plot(setosa_data$Sepal.Length, setosa_data$Sepal.Width, main="Sepal Length vs Sepal Width (Setosa)", xlab="Sepal
Length (cm)", ylab="Sepal Width (cm)", pch=19)

#(d)
hist(iris$Petal.Length, main="Histogram of Petal Length", xlab="Petal Length (cm)", col="blue")

#(e)
versicolor_data <- subset(iris, Species == "versicolor")
boxplot(versicolor_data$Petal.Width, main="Petal Width of Versicolor Iris", ylab="Petal Width (cm)", col="orange")
```


Petal Width of Versicolor Iris
Petal Width (cm)
1.0
1.2
1.4
1.6
1.8

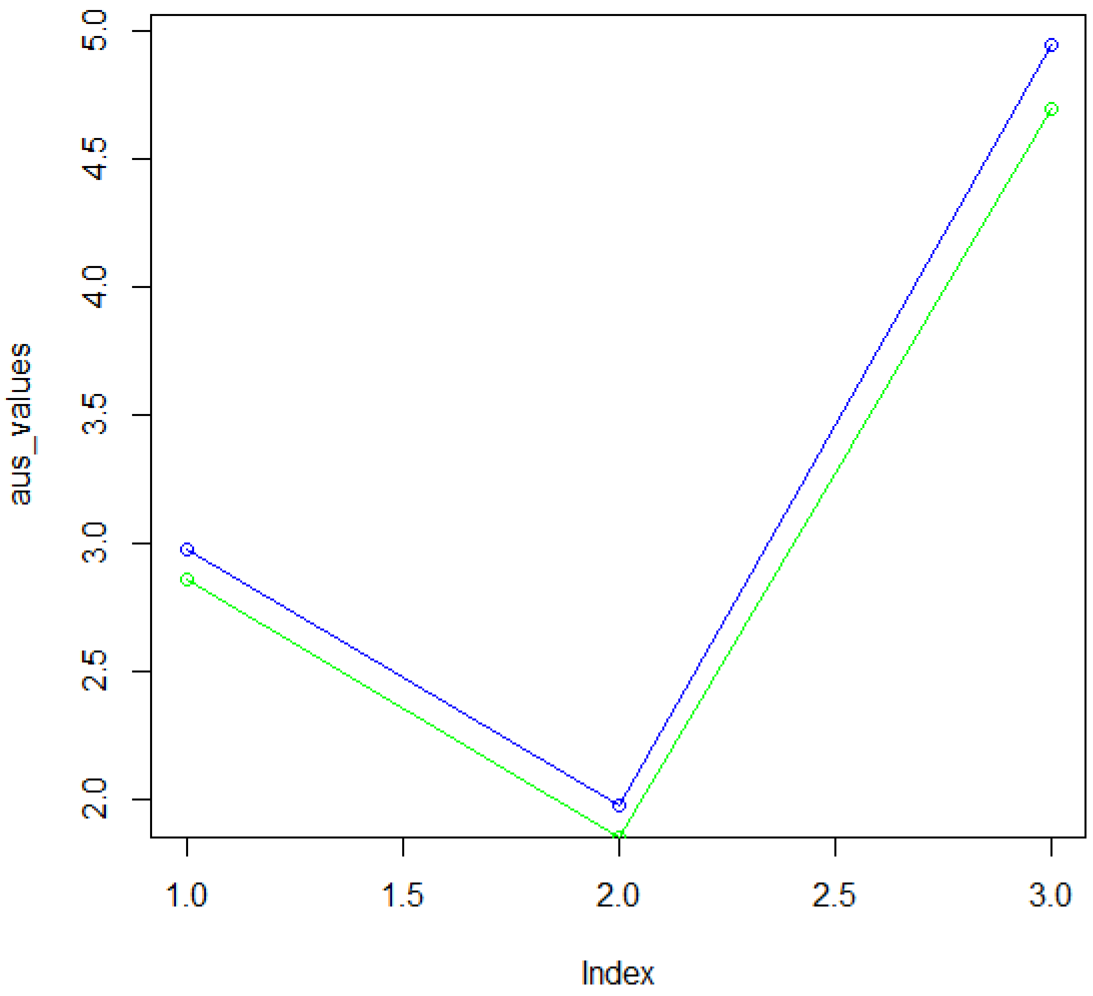
Q7)



```
alcohol <- read.table("tab_deline.txt", header = TRUE)

#a
alcohol_by_year <- split(alcohol, alcohol$Year)

for (i in c(1:7))
{
    year_data <- alcohol_by_year[[i]]
    max_beer_country <- year_data$Country[which.max(year_data$Beer)]
    min_wine_country <- year_data$Country[which.min(year_data$Wine)]
    print(max_beer_country)
    print(min_wine_country)
}

#b
alcohol_in_country <- split(alcohol, alcohol$Country)

for(j in c(1:2))
{
    print(summary(alcohol_in_country[[j]]))
}

#c

nz <- alcohol_in_country[[2]]
print(nz[nz$Beer > (mean(nz$Spirit)),"Beer"])

#d
for (i in c(1:7))
{
    print(summary(alcohol_by_year[[i]]))
}

#e
aus <- alcohol_in_country[[1]]
mean_beer_of_australia <- mean(aus$Beer)
```

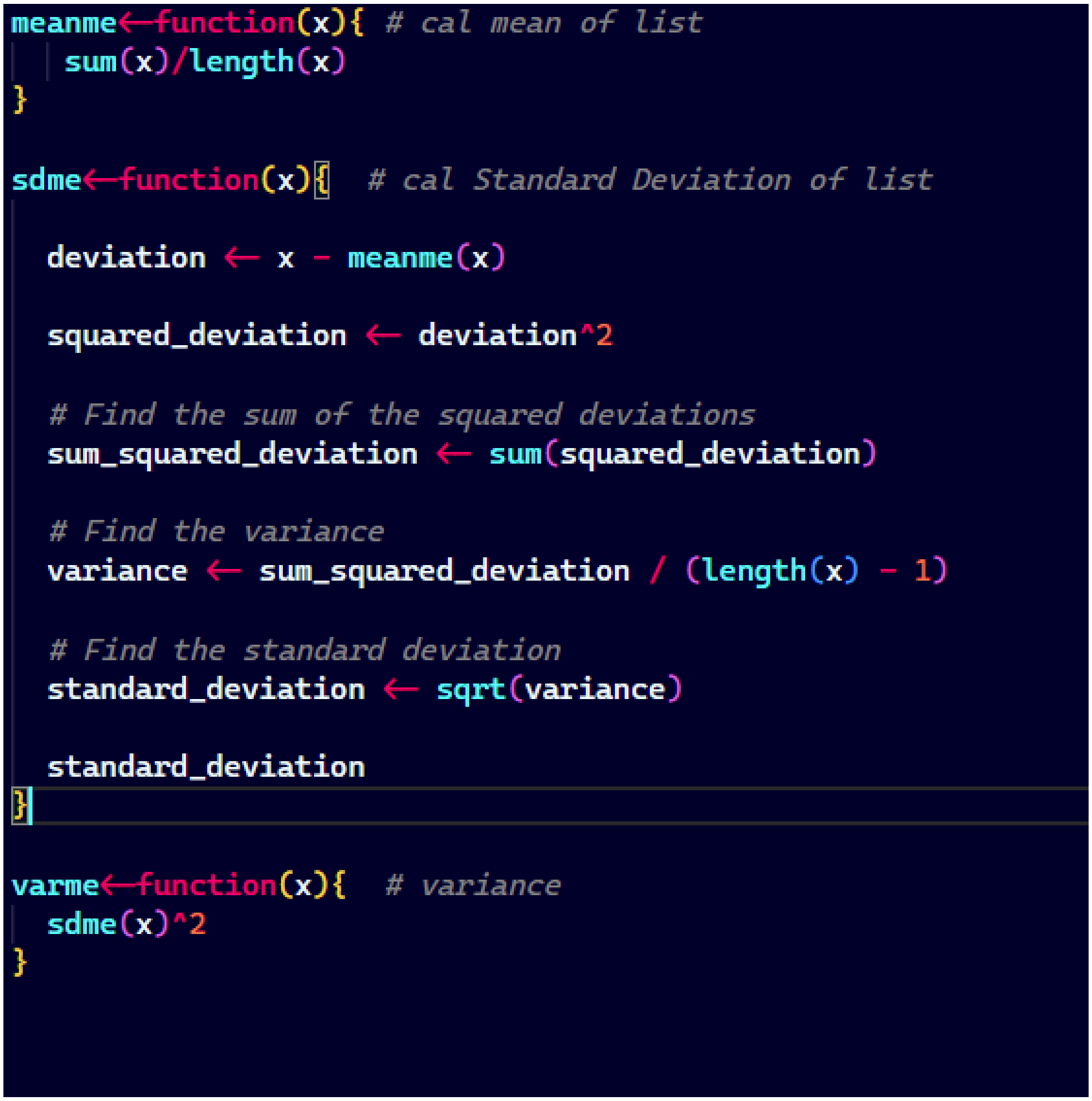


```
#e
aus <- alcohol_in_country[[1]]
mean_beer_of_australia <- mean(aus$Beer)
mean_spirit_of_australia <- mean(aus$Spirit)
mean_wine_of_australia <- mean(aus$Wine)

aus_values <- c(mean_wine_of_australia, mean_spirit_of_australia, mean_beer_of_australia)
plot(aus_values, type = "o", col = "Blue")

mean_beer_of_nz <- mean(nz$Beer)
mean_spirit_of_nz <- mean(nz$Spirit)
mean_wine_of_nz <- mean(nz$Wine)
nz_values <- c(mean_wine_of_nz, mean_spirit_of_nz, mean_beer_of_nz)
lines(nz_values, col = "Green", type = "o",
main = "Australia vs New Zeland alcohol consumption",
xlabs = c("Wine", "Spirit", "Beer"))
```


aus_values
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.0
1.5
2.0
2.5
3.0
Index

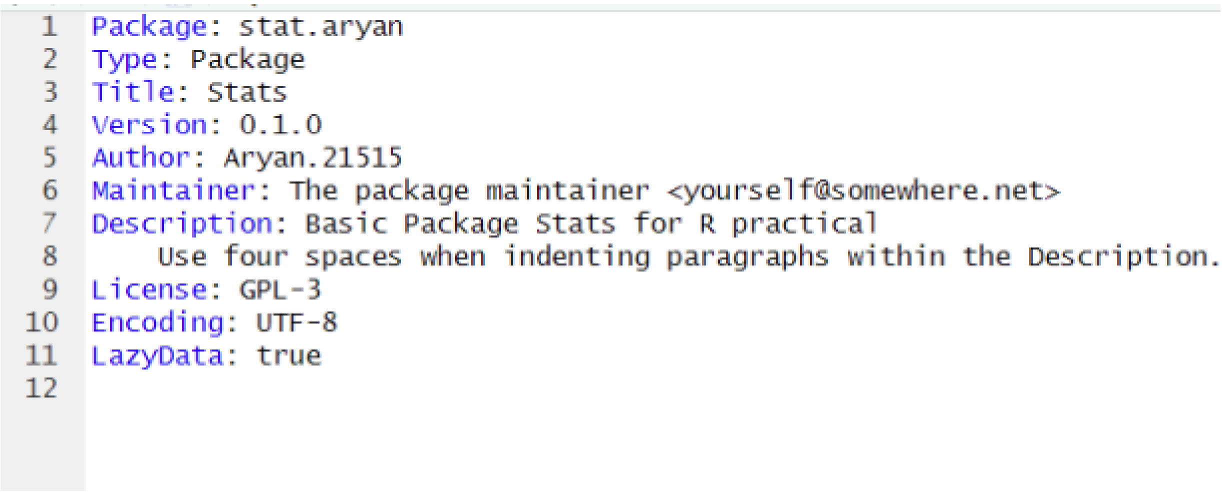
Q8)

package



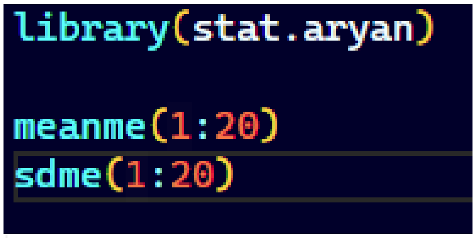
```r
meanme<-function(x){ # cal mean of list
    sum(x)/length(x)
}

sdme<-function(x){  # cal Standard Deviation of list

  deviation <- x - meanme(x)

  squared_deviation <- deviation^2

  # Find the sum of the squared deviations
  sum_squared_deviation <- sum(squared_deviation)

  # Find the variance
  variance <- sum_squared_deviation / (length(x) - 1)

  # Find the standard deviation
  standard_deviation <- sqrt(variance)

  standard_deviation
}

varme<-function(x){  # variance
  sdme(x)^2
}
```

Description

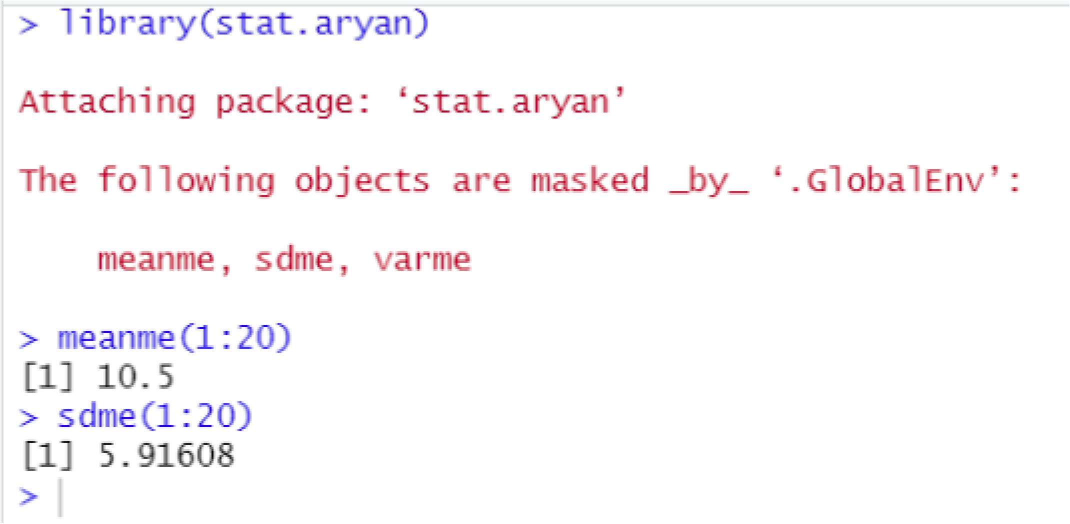


```
Package: stat.aryan
Type: Package
Title: Stats
Version: 0.1.0
Author: Aryan.21515
Maintainer: The package maintainer <yourself@somewhere.net>
Description: Basic Package Stats for R practical
    Use four spaces when indenting paragraphs within the Description.
License: GPL-3
Encoding: UTF-8
LazyData: true
```

usage

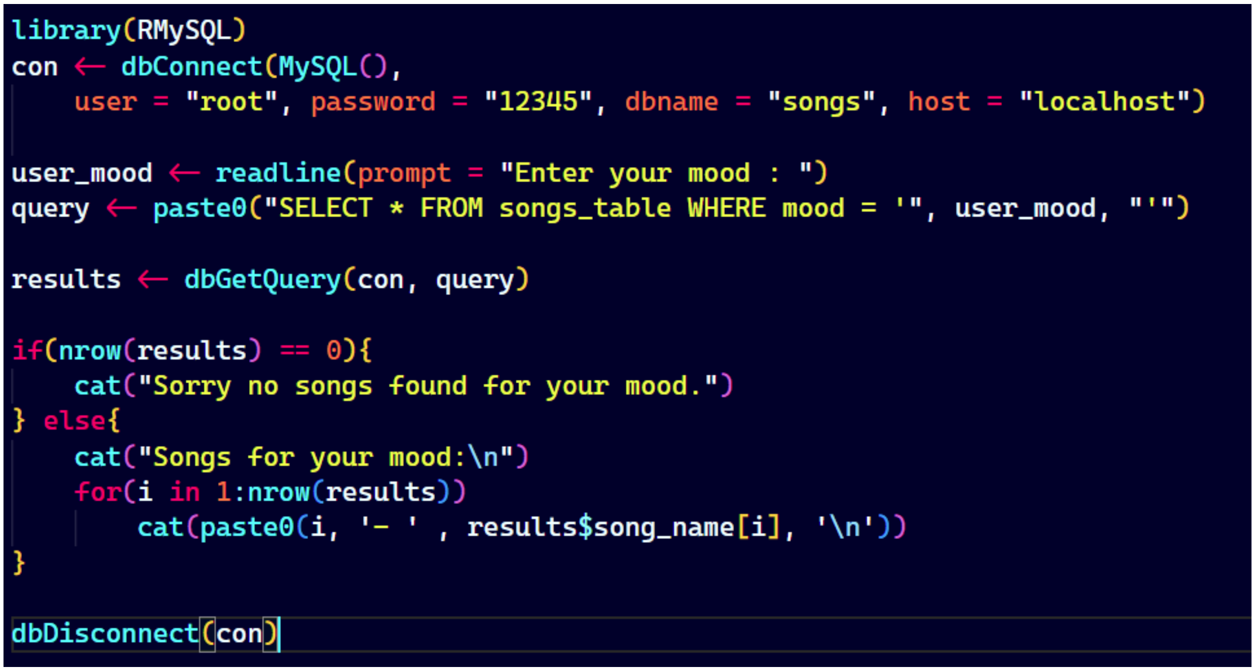


```
library(stat.aryan)

meanme(1:20)
sdme(1:20)
```



```
> library(stat.aryan)

Attaching package: ‘stat.aryan’

The following objects are masked _by_ ‘.GlobalEnv’:

    meanme, sdme, varme

> meanme(1:20)
[1] 10.5
> sdme(1:20)
[1] 5.91608
> 
```

Q9)



```
library(RMySQL)
con <- dbConnect(MySQL(),
    user = "root", password = "12345", dbname = "songs", host = "localhost")

user_mood <- readline(prompt = "Enter your mood : ")
query <- paste0("SELECT * FROM songs_table WHERE mood = '", user_mood, "'")

results <- dbGetQuery(con, query)

if(nrow(results) == 0){
    cat("Sorry no songs found for your mood.")
} else{
    cat("Songs for your mood:\n")
    for(i in 1:nrow(results))
        cat(paste0(i, '- ' , results$song_name[i], '\n'))
}

dbDisconnect(con)
```