i) Write a function called kelvin\_to\_celsius() that takes a temperature in Kelvin and returns that temperature in Celsius (Hint: To convert from Kelvin to Celsius you subtract 273.15)

CODE:

kelvin\_to\_celsius <- function(temp\_in\_kelvin){

temp\_in\_celsius <- temp\_in\_kelvin - 273.15

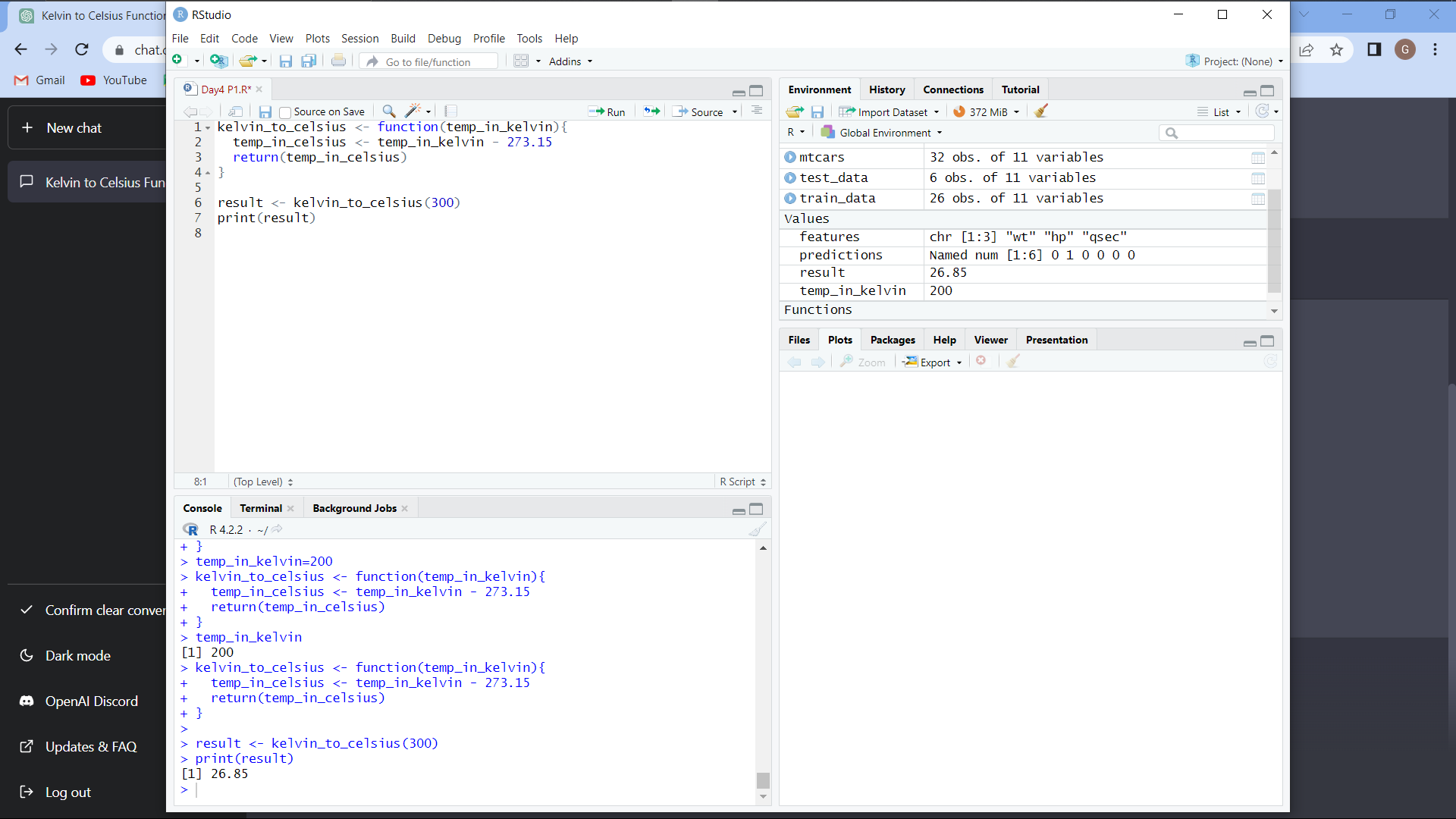
return(temp\_in\_celsius)

}

result <- kelvin\_to\_celsius(300)

print(result)

Output:



(ii) Write suitable R code to compute the mean, median ,mode of the following values c(90, 50, 70, 80, 70, 60, 20, 30,80, 90, 20)

CODE:

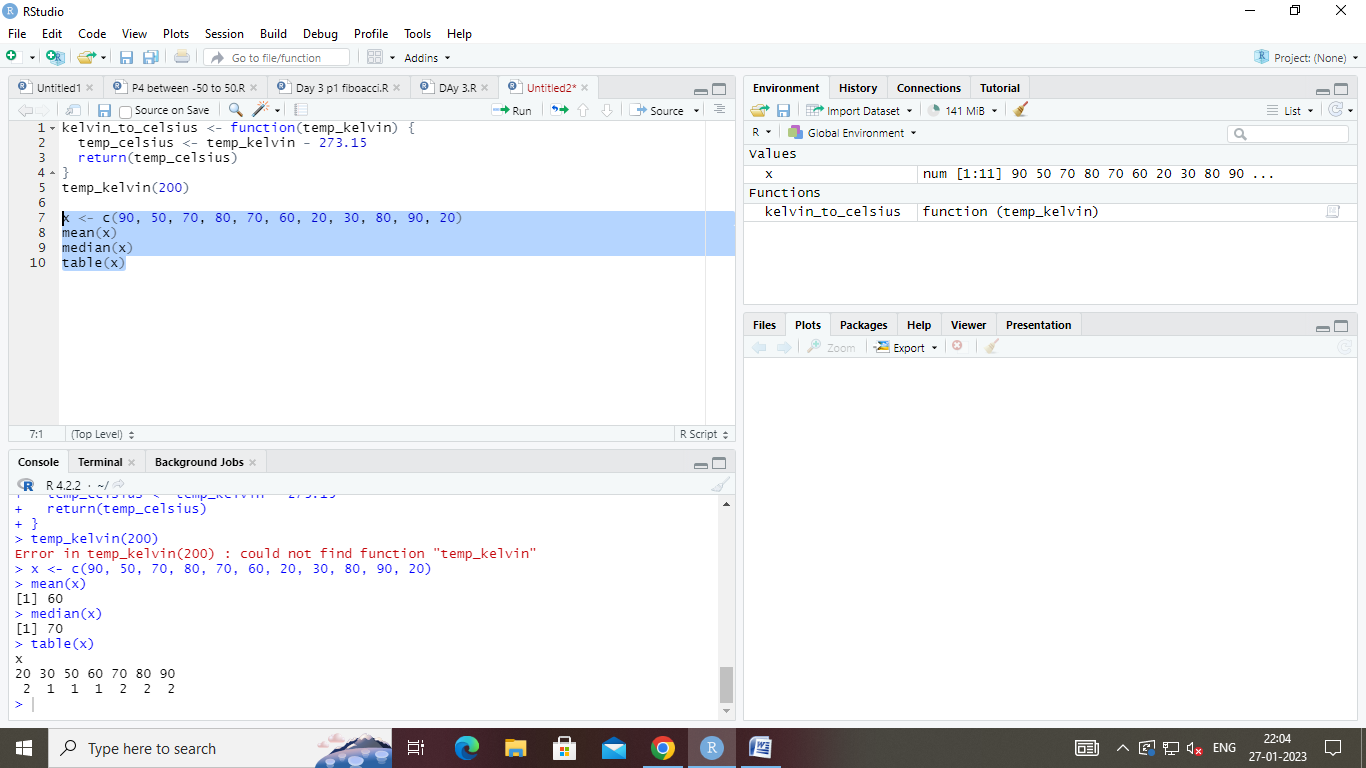
x <- c(90, 50, 70, 80, 70, 60, 20, 30, 80, 90, 20)

mean(x)

median(x)

table(x)

Output:



(iii) Write R code to find 2nd highest and 3rd Lowest value of above problem.

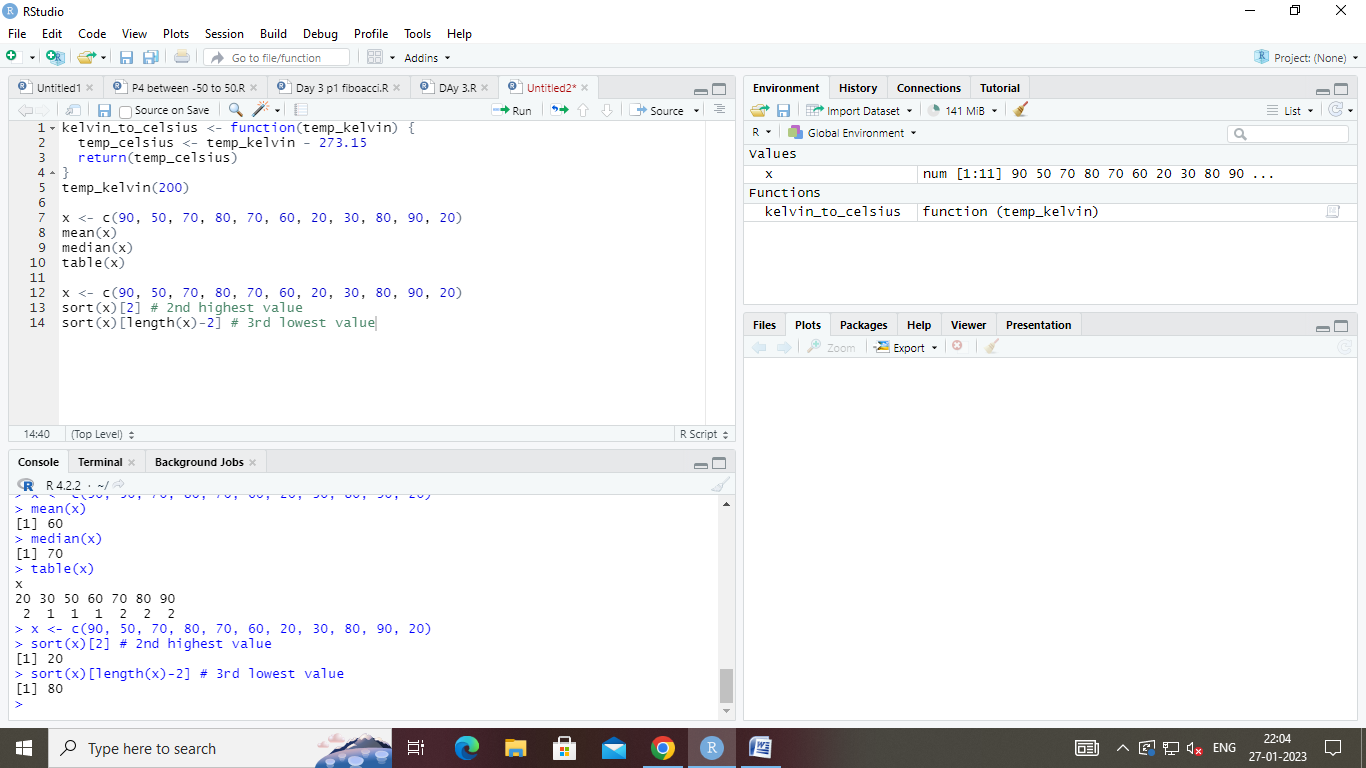
CODE:

x <- c(90, 50, 70, 80, 70, 60, 20, 30, 80, 90, 20)

sort(x)[2] # 2nd highest value

sort(x)[length(x)-2] # 3rd lowest value

Output:



2. Explore the airquality dataset. It contains daily air quality measurements from New York during a period of five months:

• Ozone: mean ozone concentration (ppb),

• Solar.R: solar radiation (Langley),

• Wind: average wind speed (mph),

• Temp: maximum daily temperature in degrees Fahrenheit,

• Month: numeric month (May=5, June=6, and so on),

• Day: numeric day of the month (1-31).

i. Compute the mean temperature(don’t use build in function)

CODE:

airquality <- airquality

sum\_temp <- 0

n <- nrow(airquality)

for(i in 1:n){

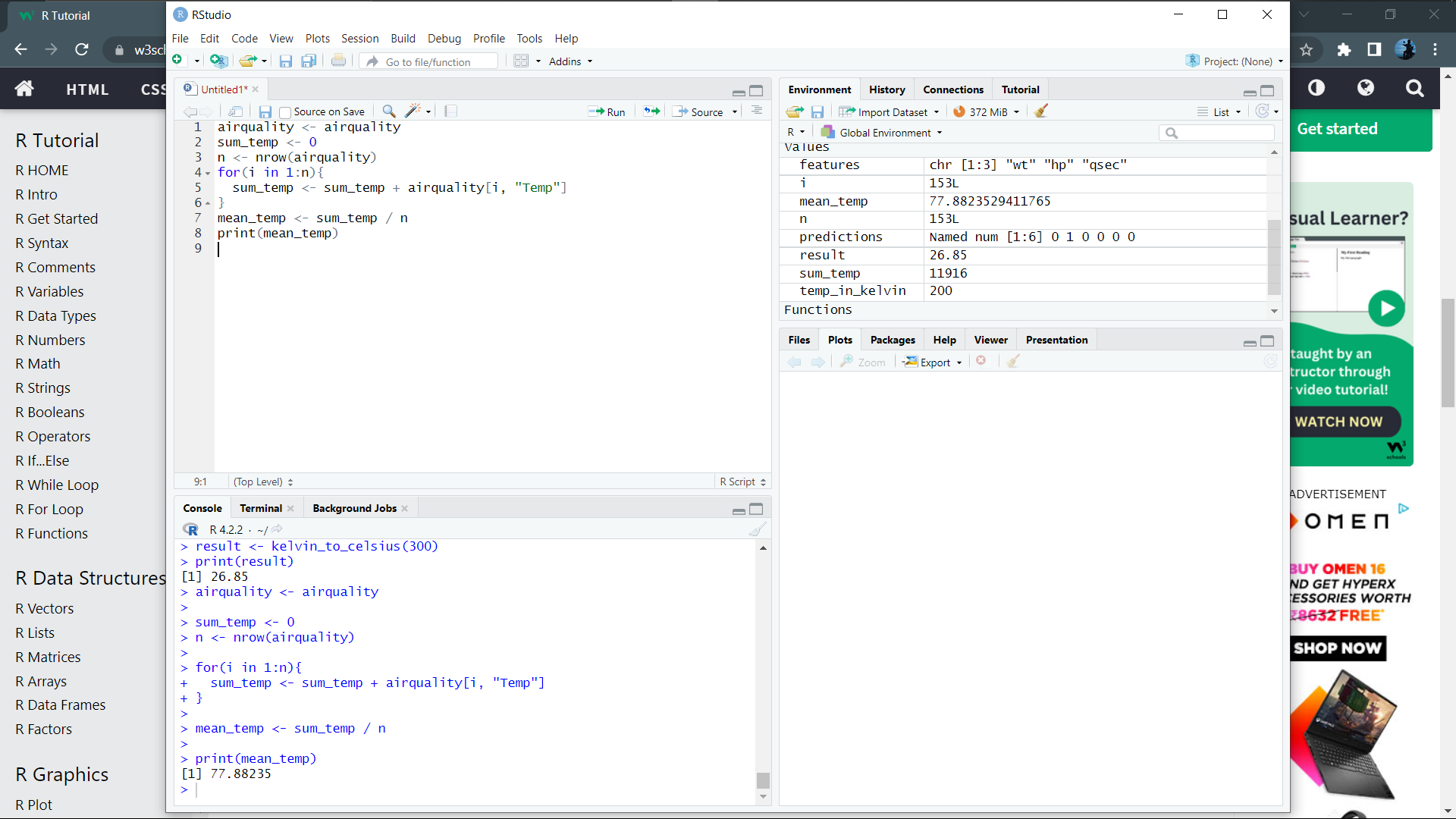
sum\_temp <- sum\_temp + airquality[i, "Temp"]

}

mean\_temp <- sum\_temp / n

print(mean\_temp)

Output:



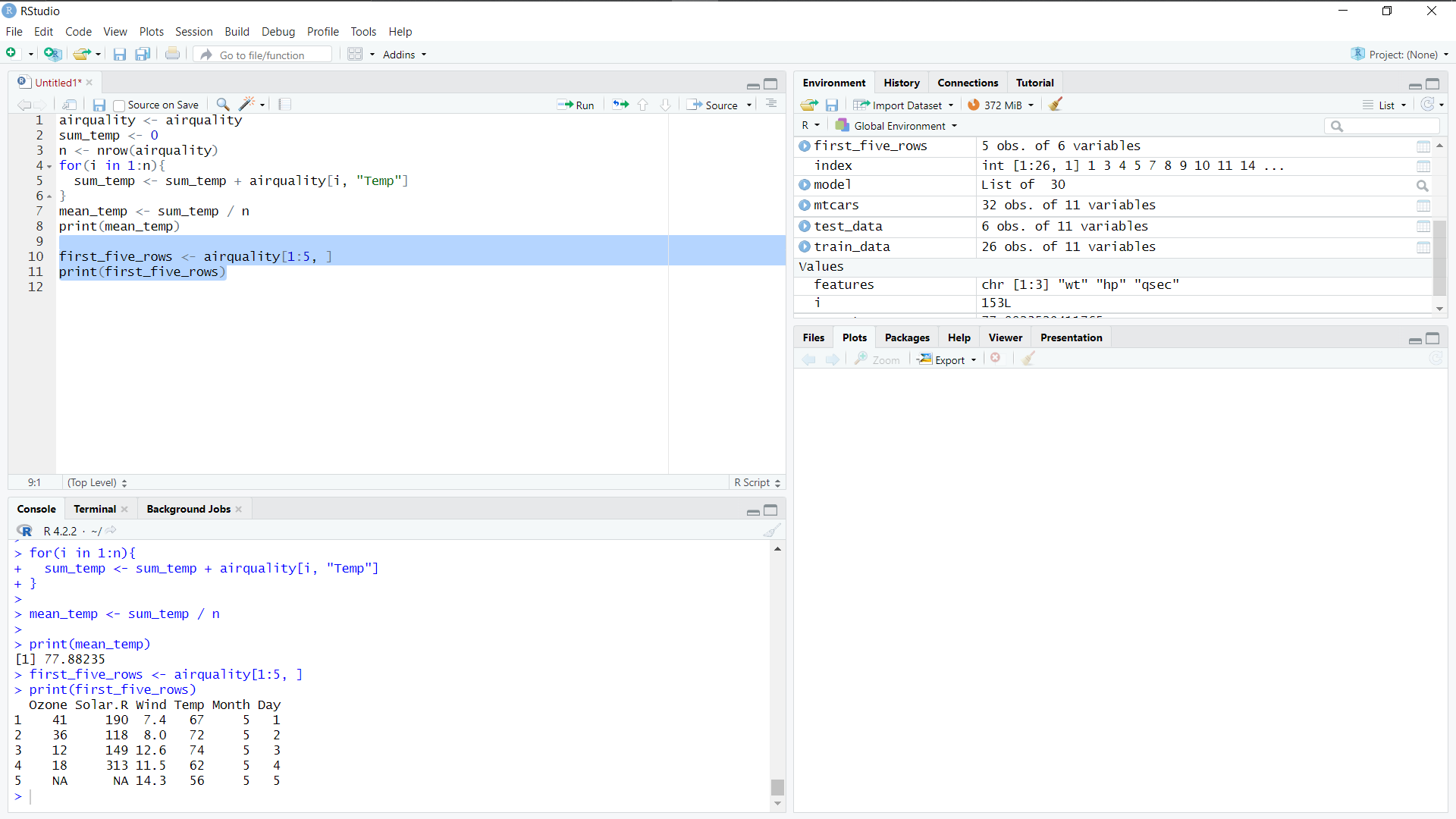
ii. Extract the first five rows from airquality.

CODE:

first\_five\_rows <- airquality[1:5, ]

print(first\_five\_rows)

Output:



iii. Extract all columns from airquality except

Temp and Wind

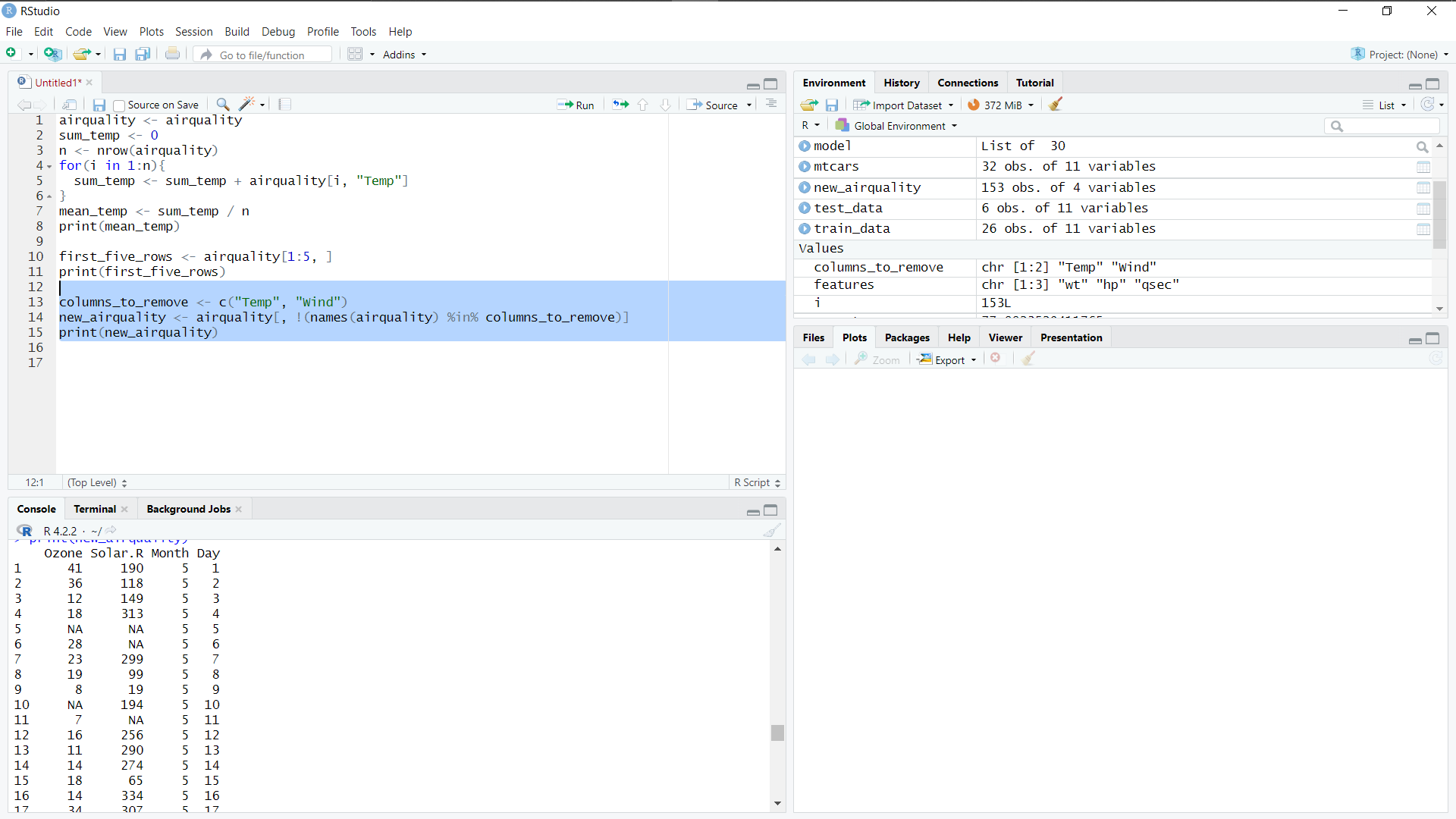
CODE:

columns\_to\_remove <- c("Temp", "Wind")

new\_airquality <- airquality[, !(names(airquality) %in% columns\_to\_remove)]

print(new\_airquality)

Output:



iv.Which was the coldest day during the period?

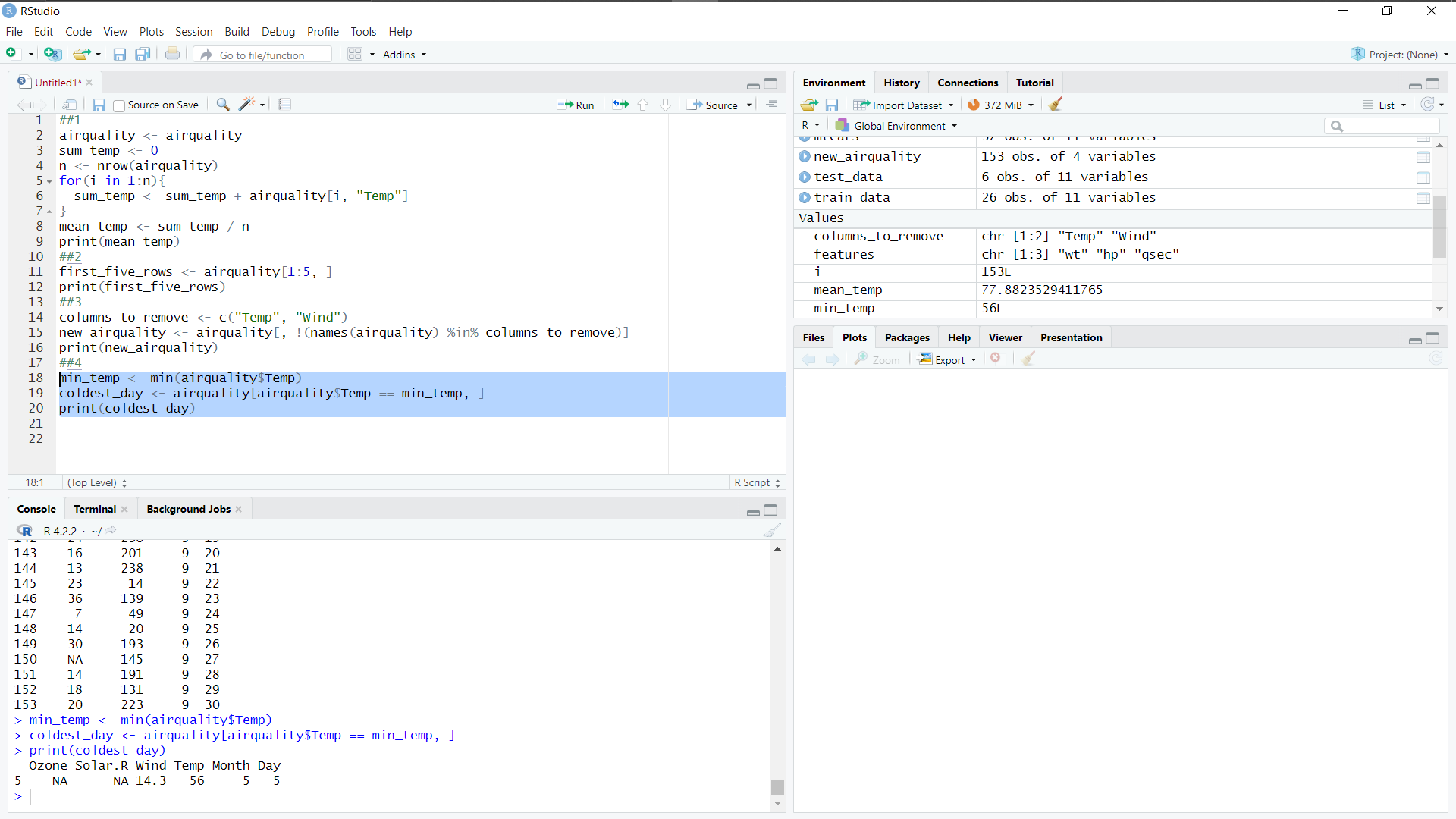
CODE:

min\_temp <- min(airquality$Temp)

coldest\_day <- airquality[airquality$Temp == min\_temp, ]

print(coldest\_day)

Output:



v.How many days was the wind speed greater than 17 mph?

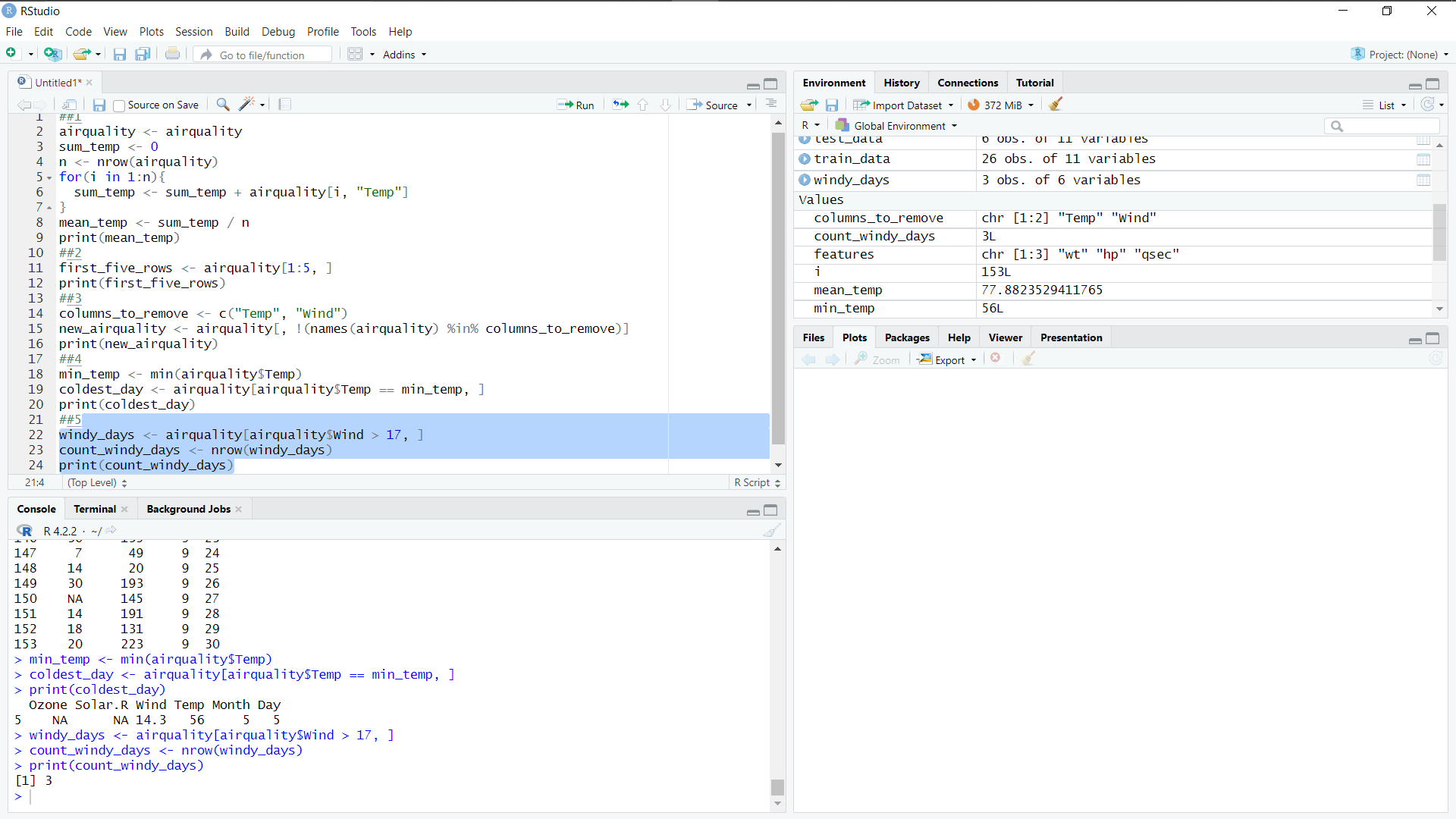
CODE:

windy\_days <- airquality[airquality$Wind > 17, ]

count\_windy\_days <- nrow(windy\_days)

print(count\_windy\_days)

Output:



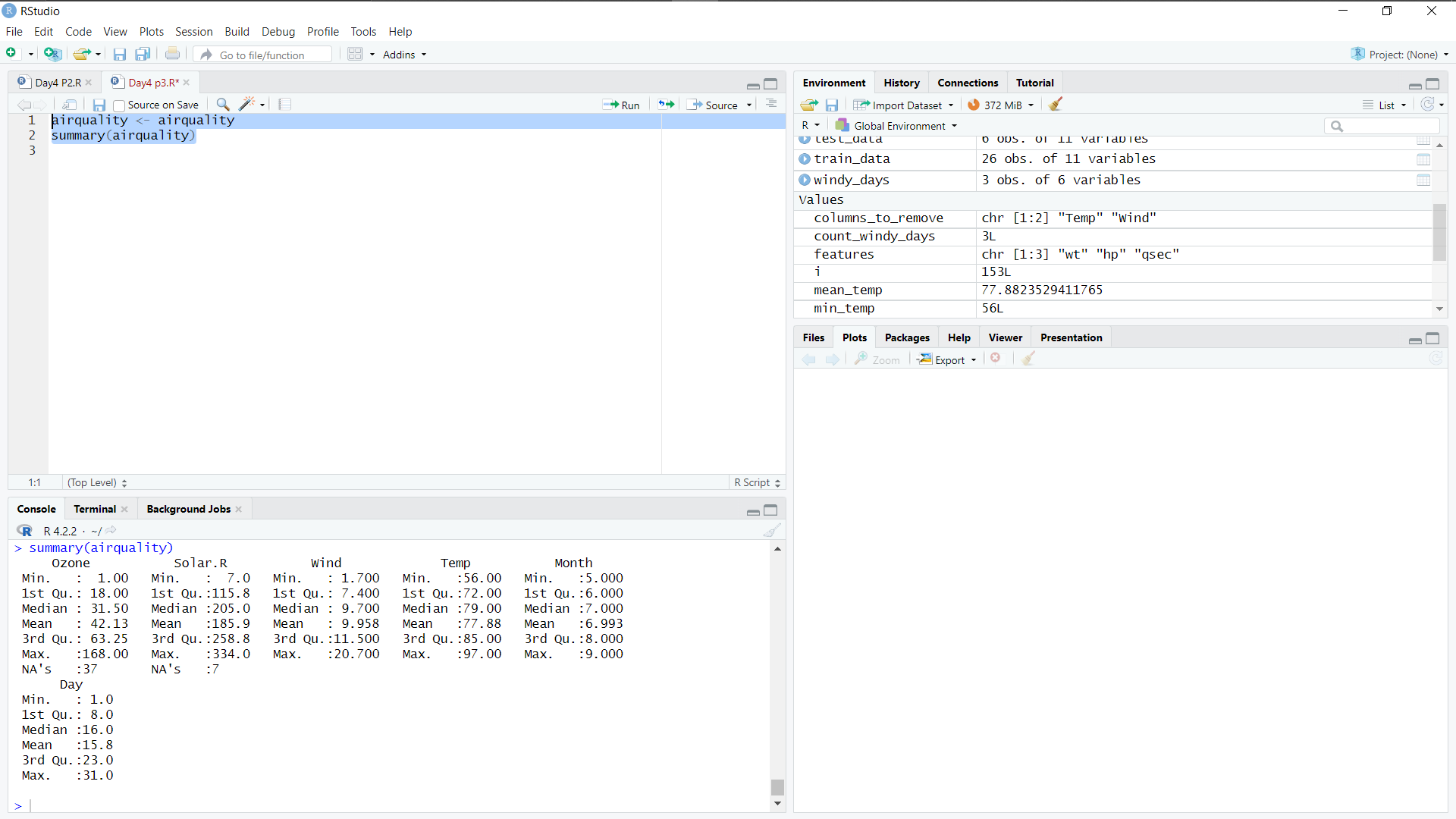
3.(i)Get the Summary Statistics of air quality dataset

CODE:

airquality <- airquality

summary(airquality)

Output:



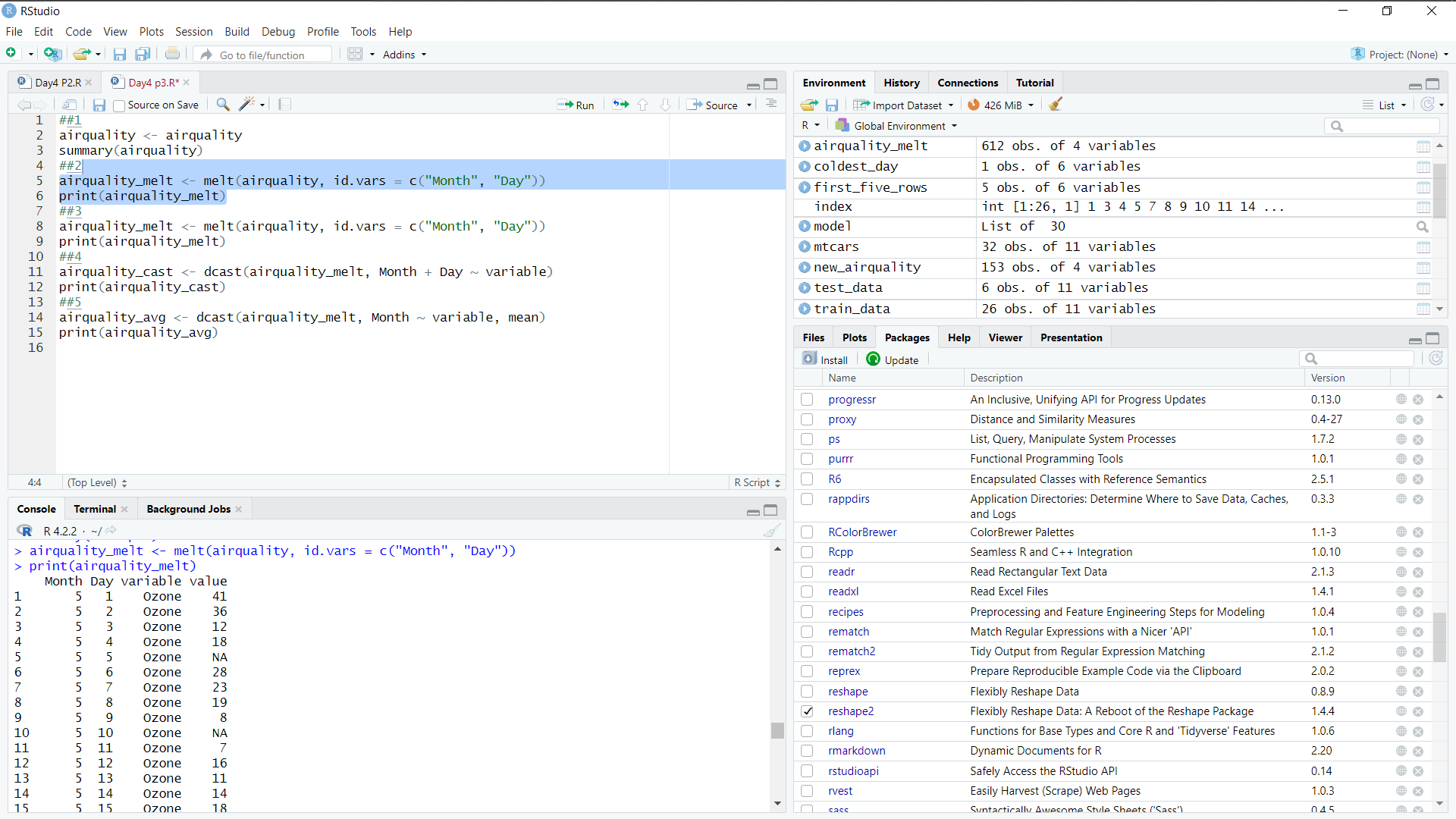
(ii)Melt airquality data set and display as a long – format data?

CODE:

airquality\_melt <- melt(airquality, id.vars = c("Month", "Day"))

print(airquality\_melt)

Output:



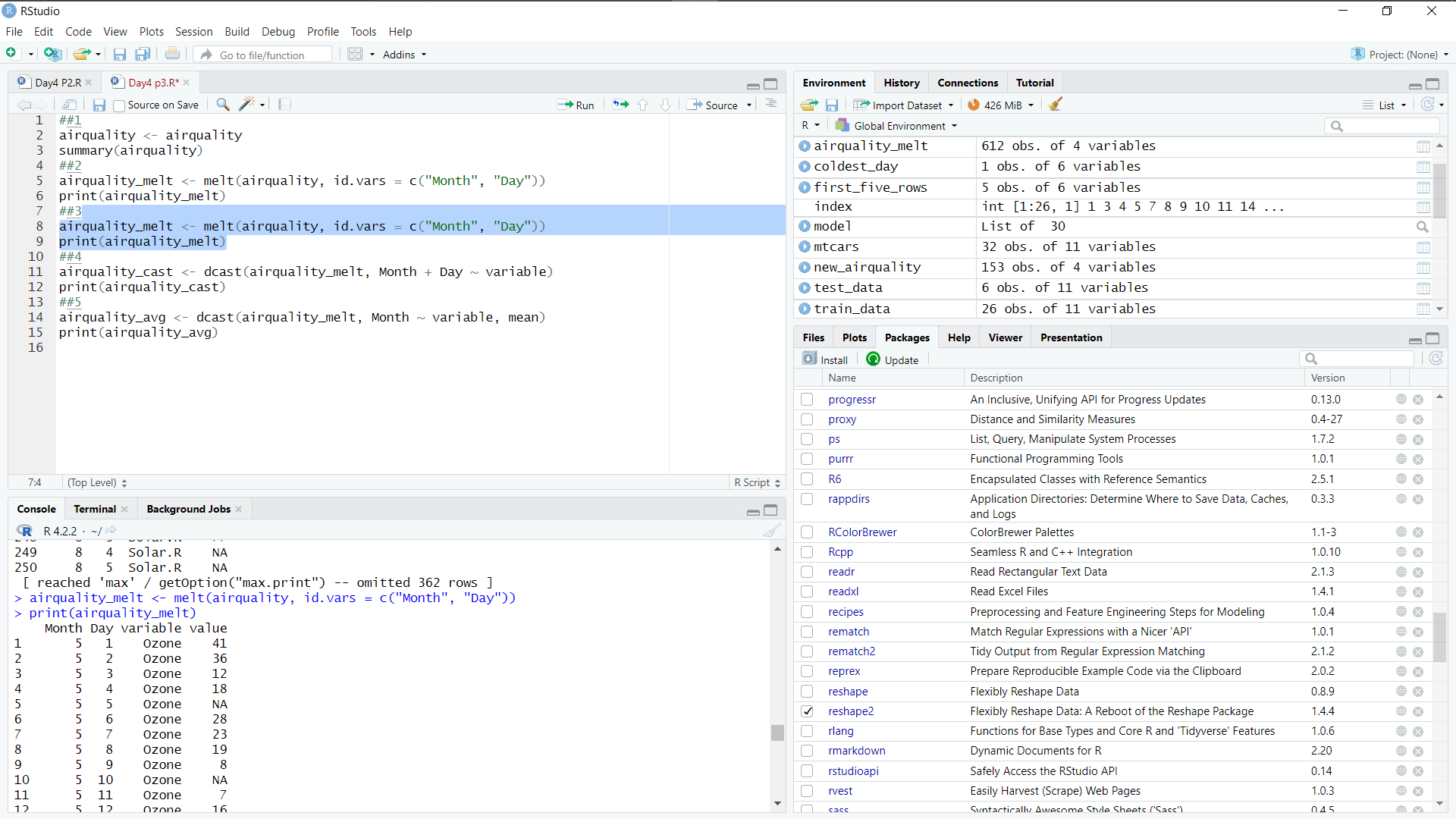
(iii)Melt airquality data and specify month and day to be “ID variables”?

CODE:

airquality\_melt <- melt(airquality, id.vars = c("Month", "Day"))

print(airquality\_melt)

Output:



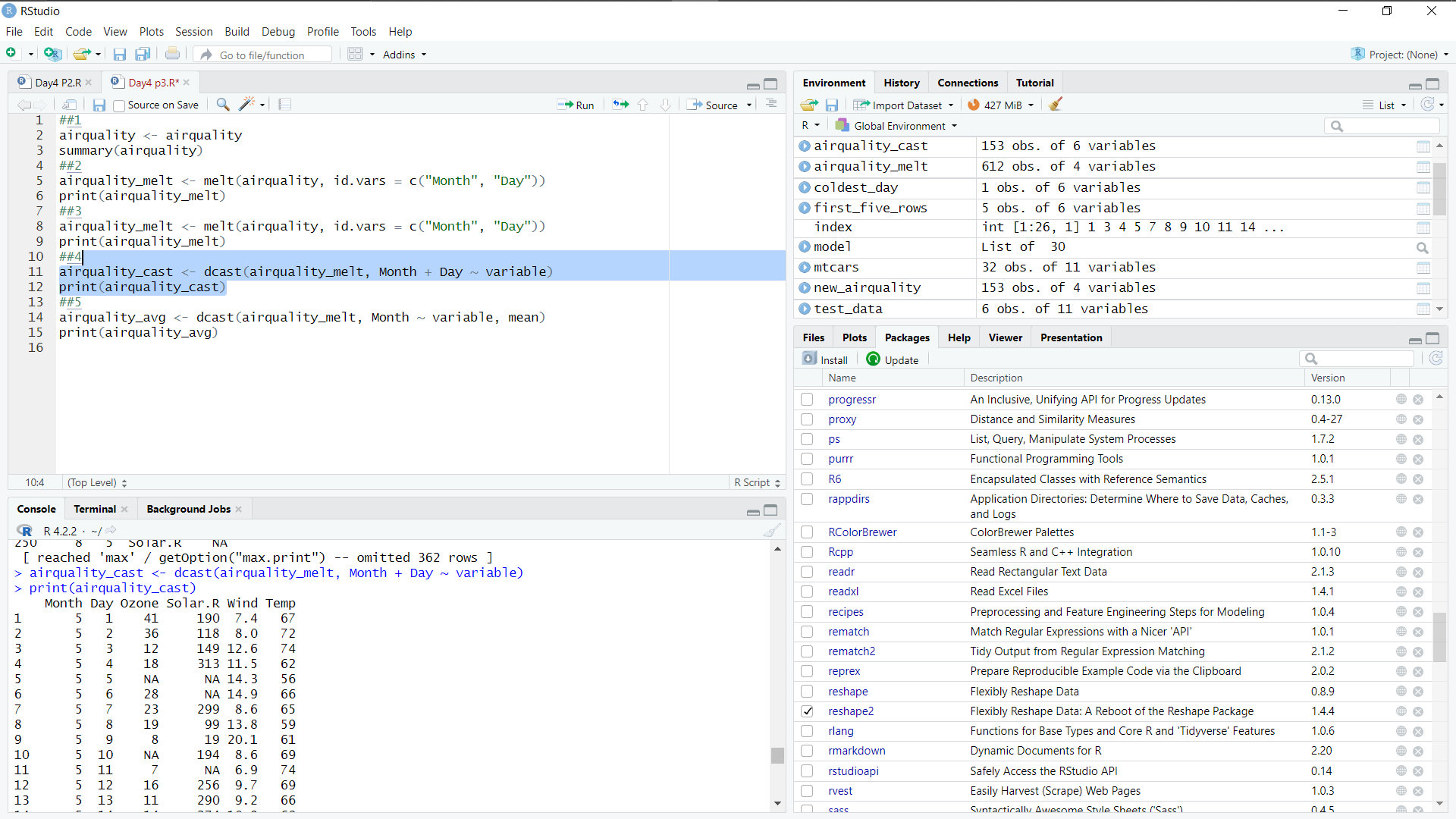
(iv)Cast the molten airquality data set with respect to month and date features

CODE:

airquality\_cast <- dcast(airquality\_melt, Month + Day ~ variable)

print(airquality\_cast)

Output:



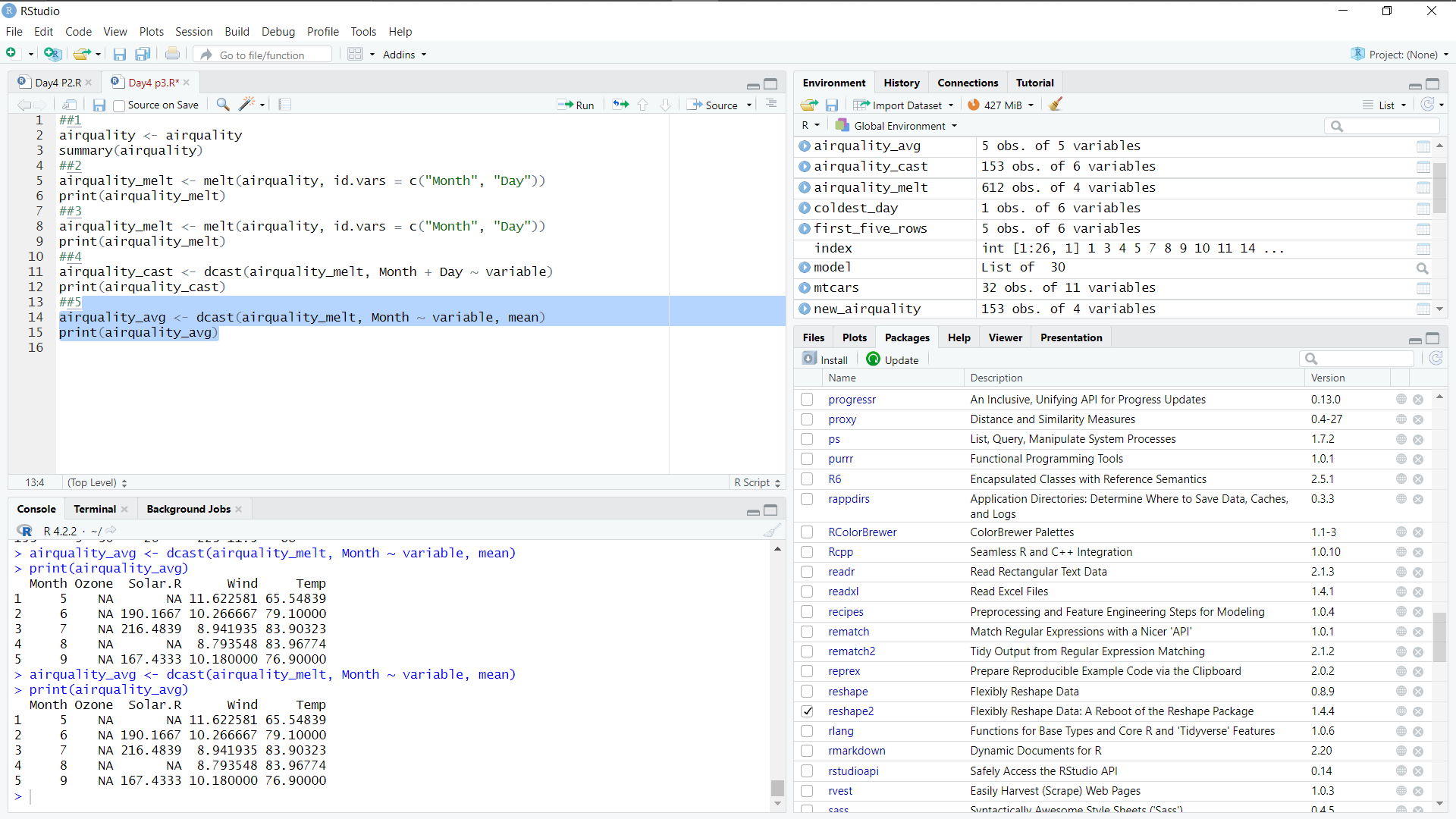
(v) Use cast function appropriately and compute the average of Ozone, Solar.R , Wind and temperature per month?

CODE:

airquality\_cast <- dcast(airquality\_melt, Month + Day ~ variable)

print(airquality\_cast)

Output:



4.(i) Find any missing values(na) in features and drop the missing values if its less than 10% else replace that with mean of that feature.

CODE:

airquality <- airquality

na\_count <- sum(is.na(airquality))

if (na\_count / nrow(airquality) < 0.1) {

airquality <- na.omit(airquality)

} else {

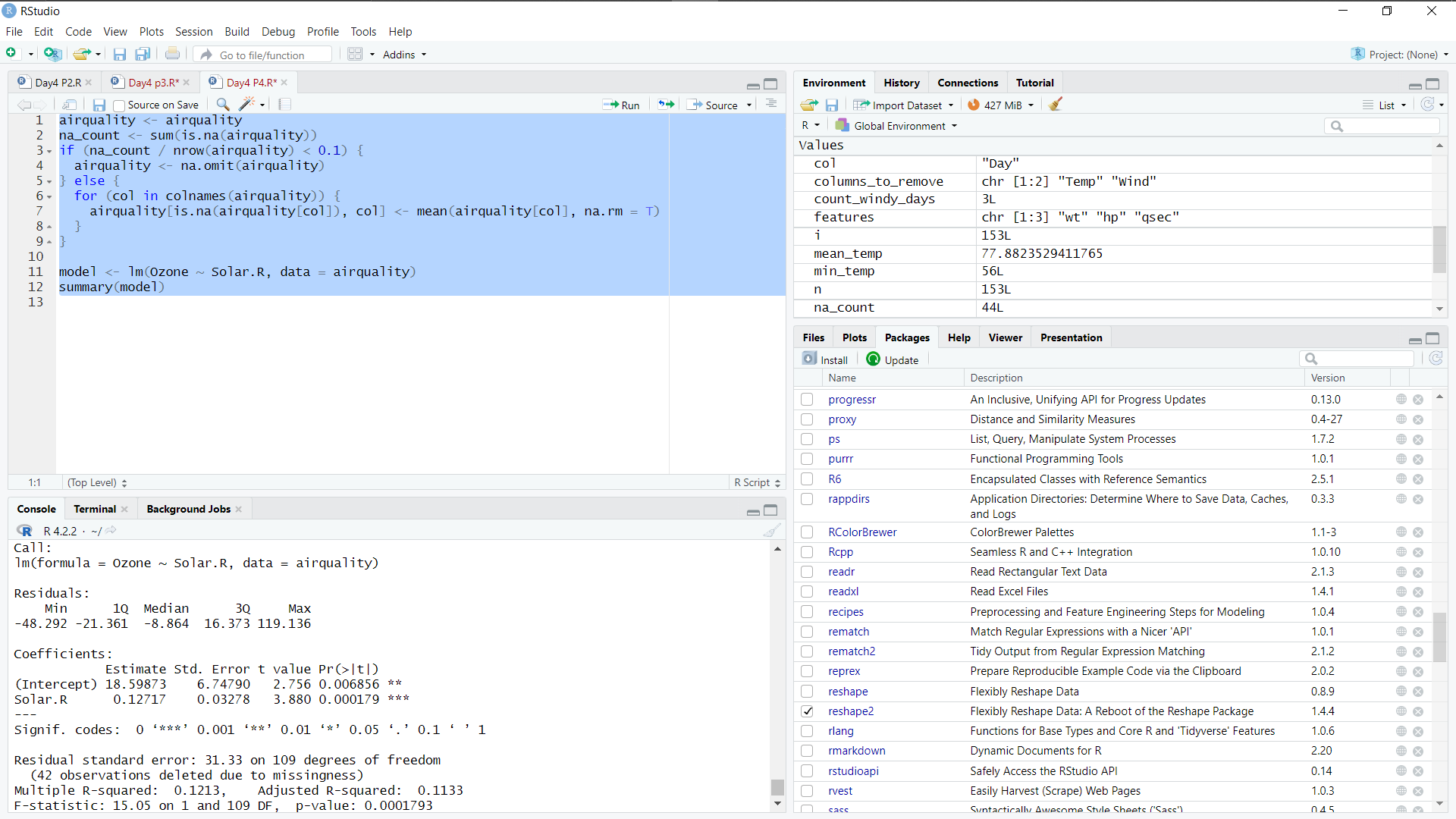
for (col in colnames(airquality)) {

airquality[is.na(airquality[col]), col] <- mean(airquality[col], na.rm = T)

}

}

Output:



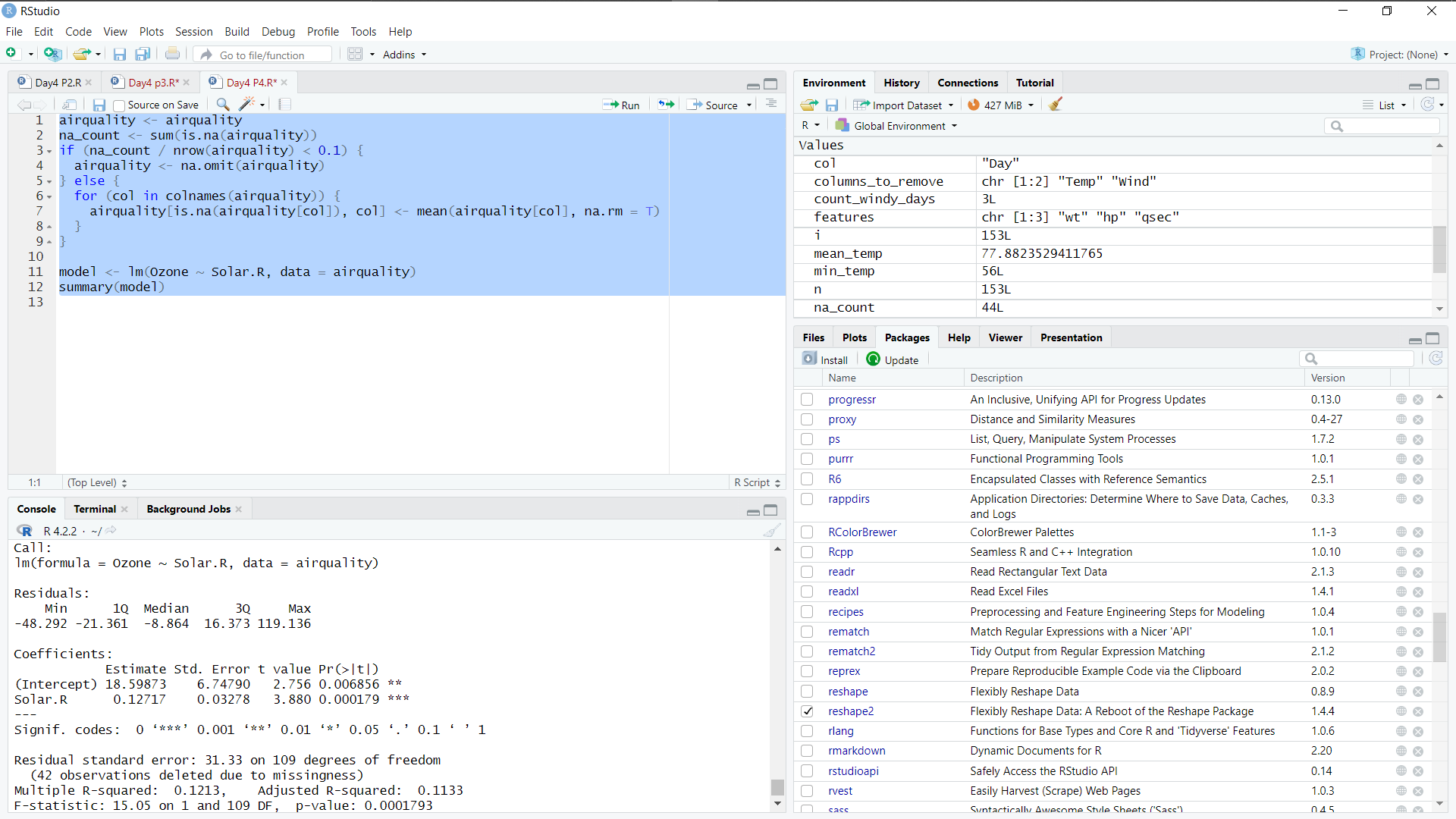
(ii) Apply a linear regression algorithm using Least Squares Method on “Ozone” and “Solar.R”

CODE:

model <- lm(Ozone ~ Solar.R, data = airquality)

summary(model)

Output:



(iii)Plot Scatter plot between Ozone and Solar and add regression line created by above model

CODE:

library(ggplot2)

ggplot(airquality, aes(x = Solar.R, y = Ozone)) +

geom\_point() +

geom\_smooth(method = "lm", formula = y ~ x, se = FALSE) +

ggtitle("Scatter Plot of Ozone vs Solar.R with Regression Line")

Output:

