

American International University-Bangladesh

(AIUB)

Department of Computer Science and Engineering (CSE)

Faculty of Science & Technology (FST)

Spring 2022-23

Course: Introduction to Data Science

Section: B

Mid Term Project

**Submitted by:**

**Name: Sakib-Ul-Ahsan**

**ID: 20-42978-1**

**Department: CSE**

**Submitted to:**

**Tohedul Islam**

**Assistant Professor**

**Department of Computer Science**

**Dataset Description**: The dataset that was provided to us was in csv format. The name of the file is Dataset\_midterm\_Section(B). It contains 302 instances and 10 attributes. The attributes include sex, age, sibsp, parch, fare, embarked, class, who, alone, survived. Here, sibsp means siblings of passenger, parch means parents/child aboard titanic, embarked means port of embarkation. The target attribute in this dataset is named survived which has only two values 0 and 1.

**Importing the dataset:**

mydata<-read.csv("C:/Users/USER/Desktop/Data

Science/Mid project/Dataset\_midterm\_Section(B).csv",header=TRUE,sep=",")

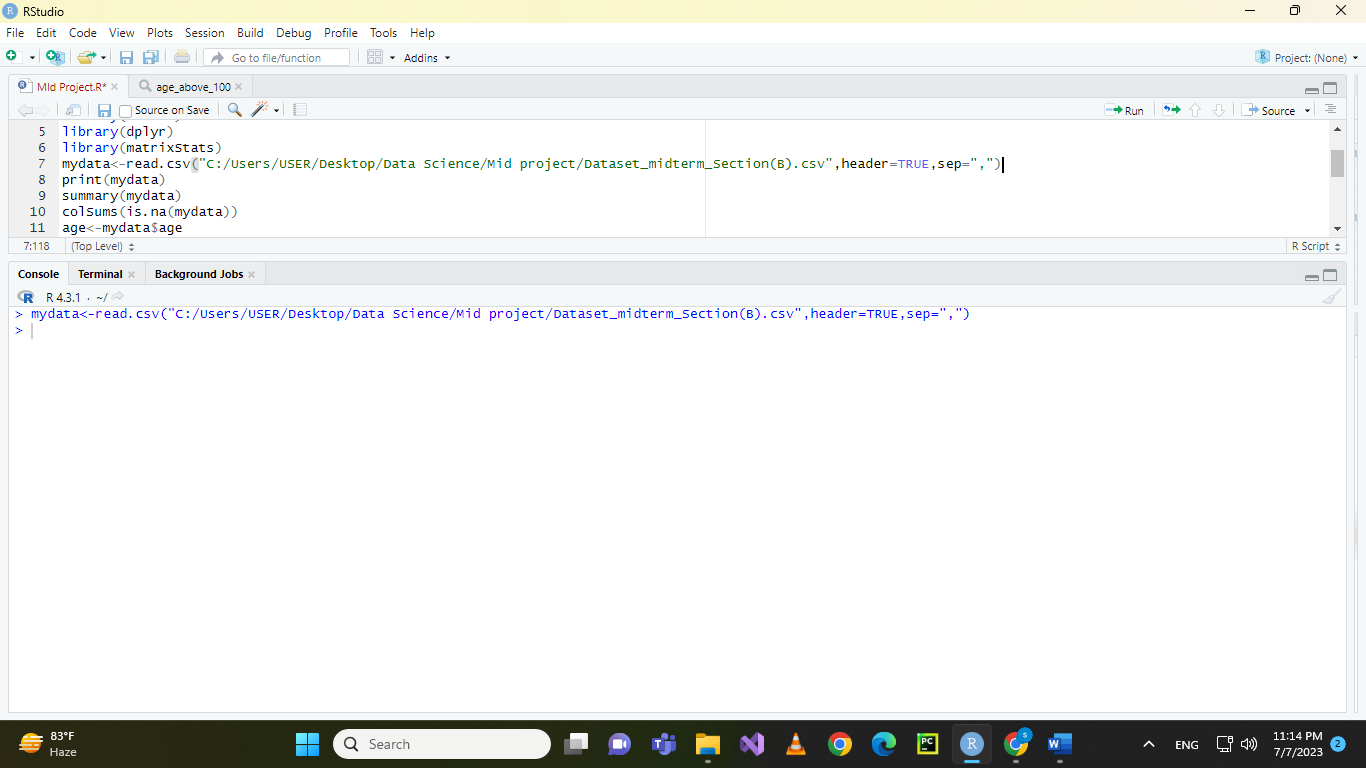


Figure : Importing dataset in RStudio

**Print The dataset:**

mydata

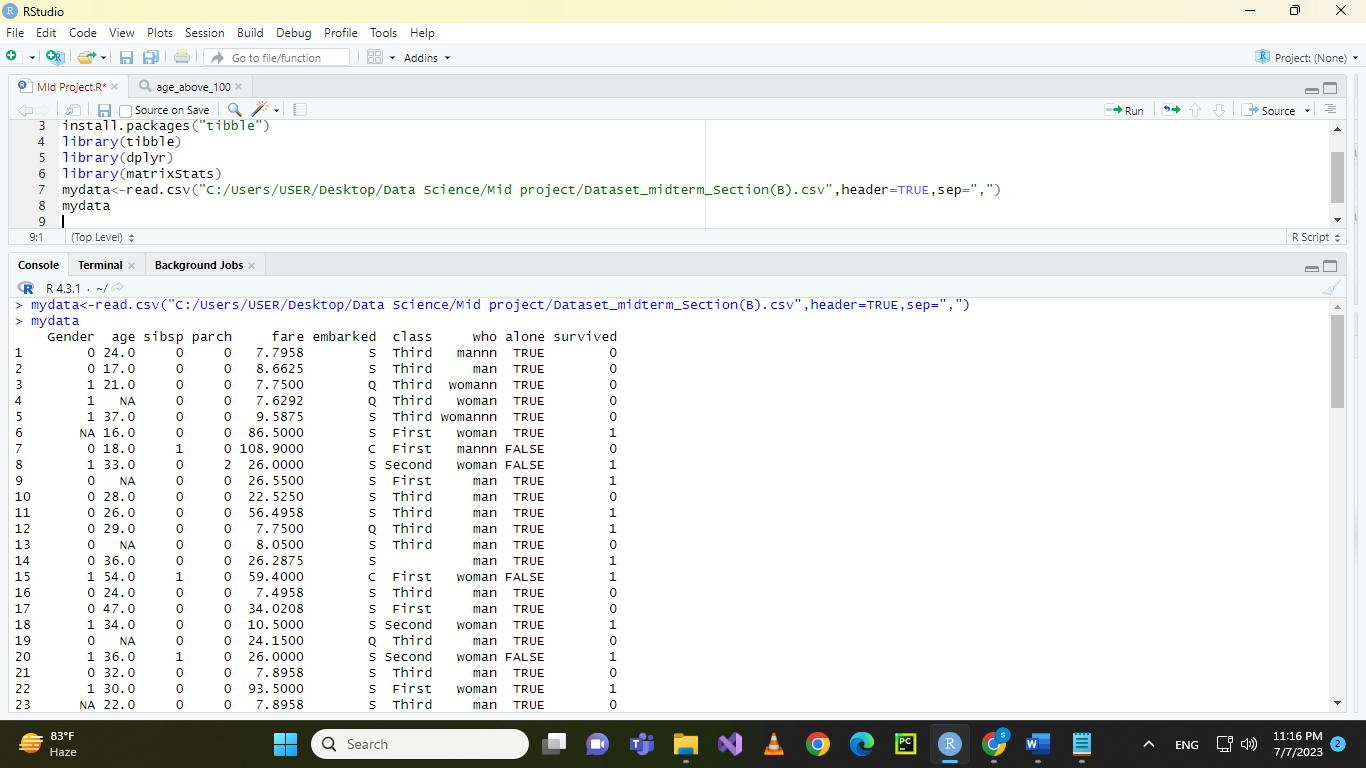


Figure : Printing the dataset

**Names of attributes:**

names(Dataset)

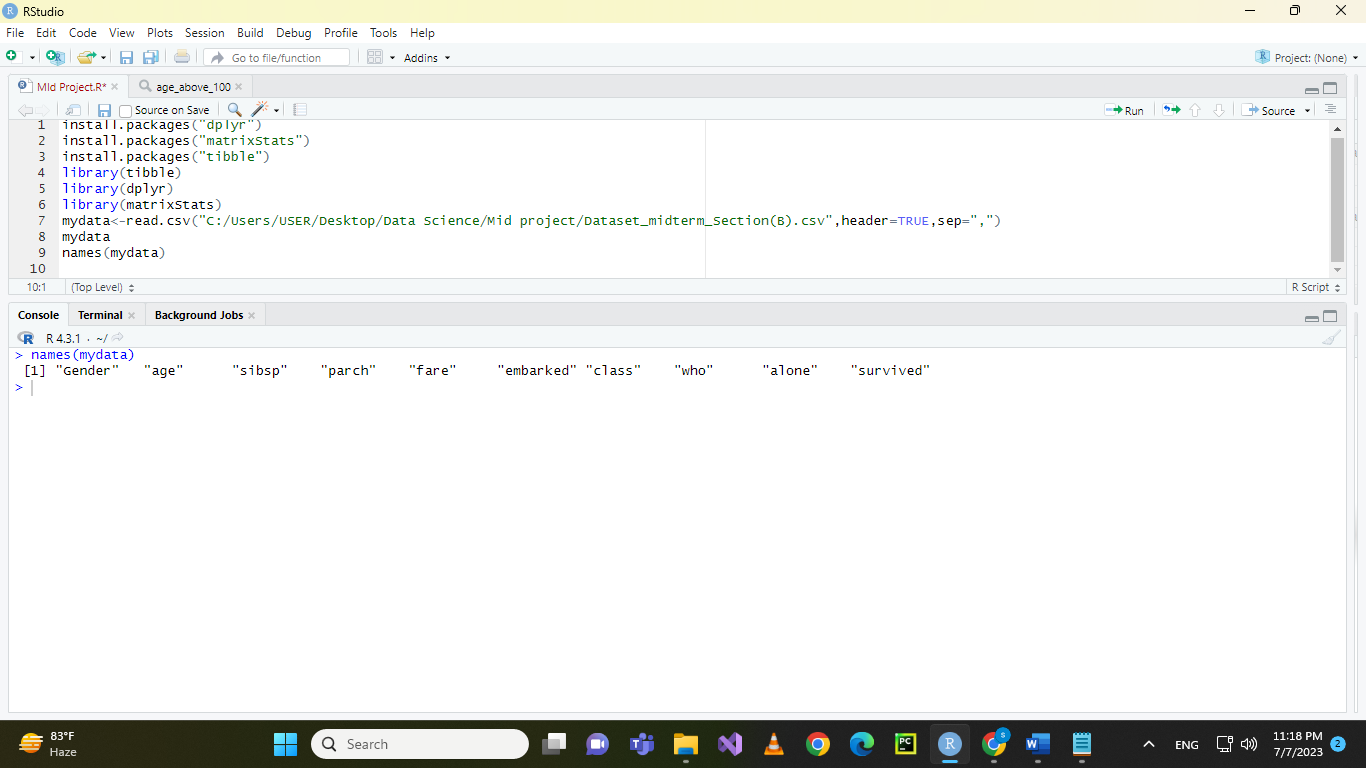


Figure : Names of attributes

**Summary of all the attributes:**

summary(mydata)

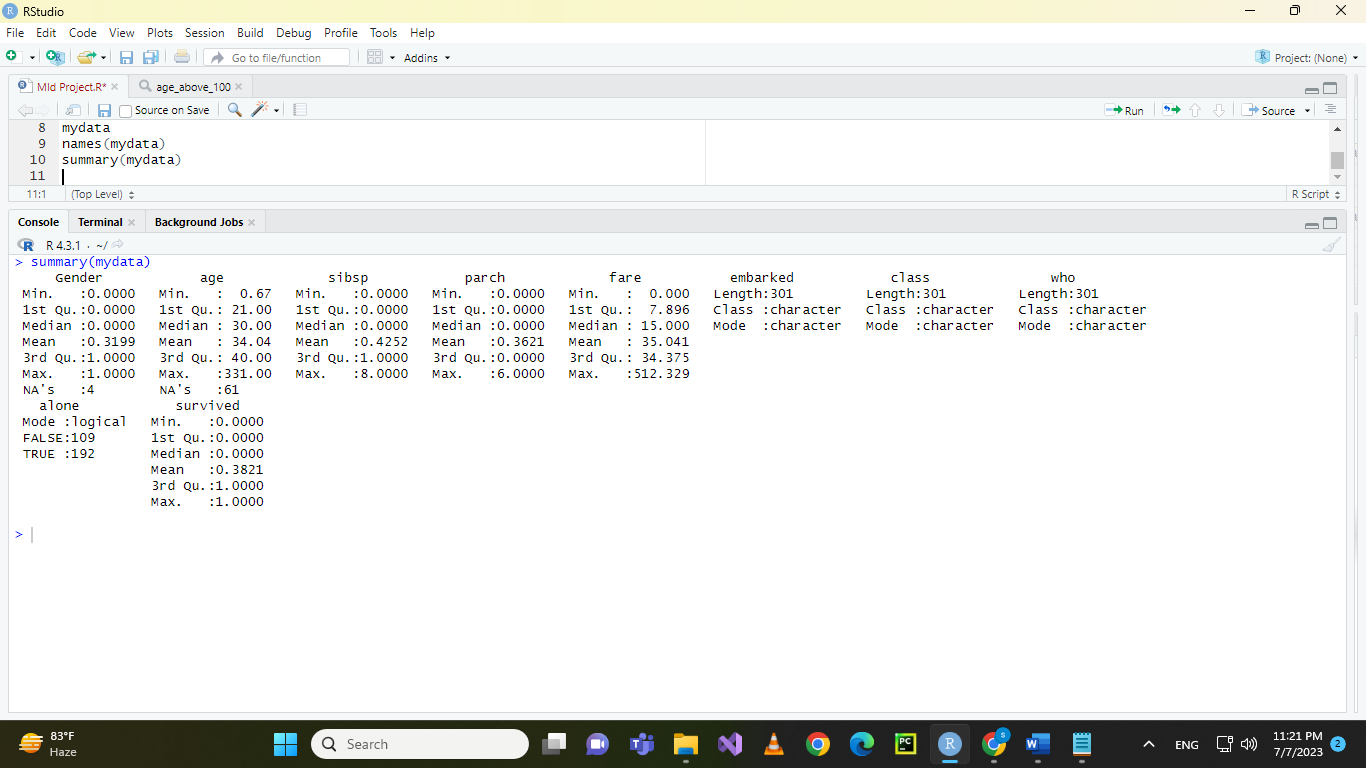


Figure : Summary of the dataset

**Total number of null values:**

**sum(is.na(mydata))**

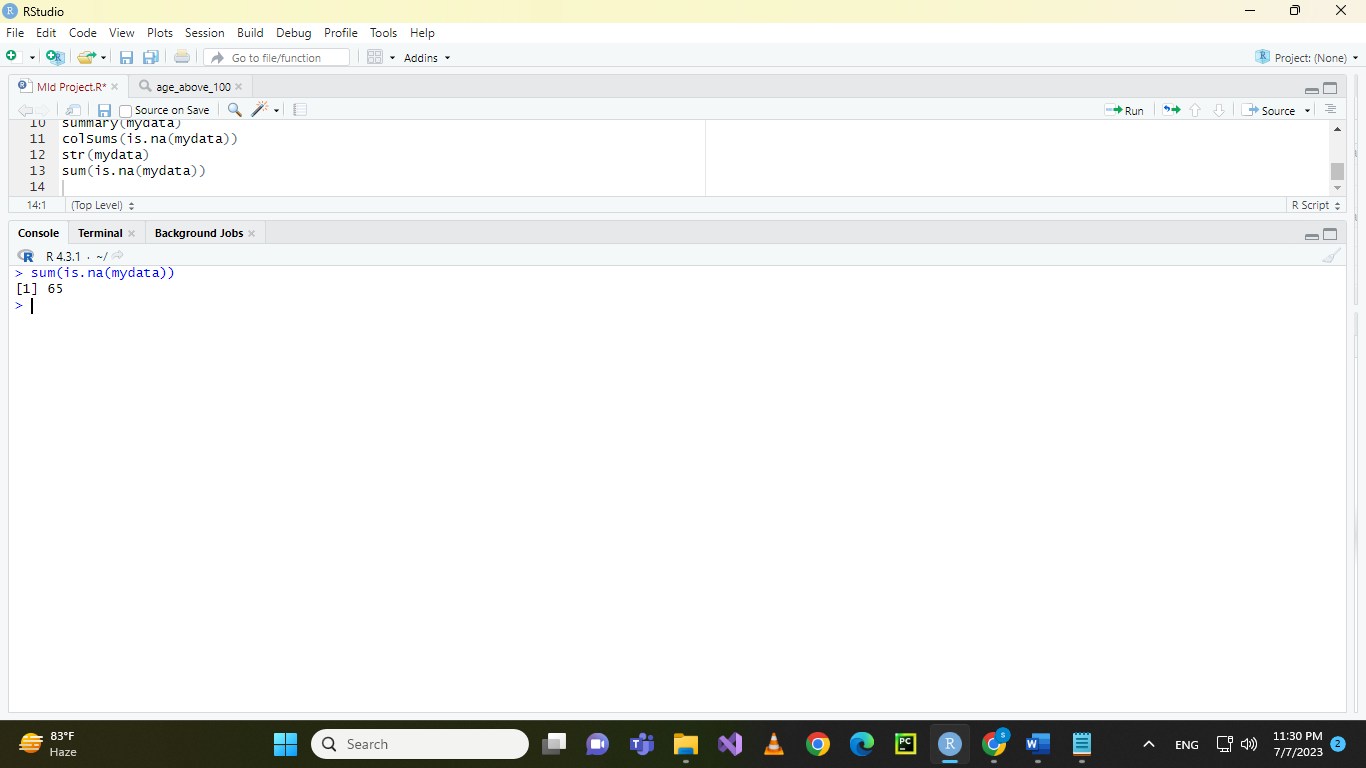


Figure : Total missing values

**Check for null values in exact attributes:**

colSums(is.na(mydata))

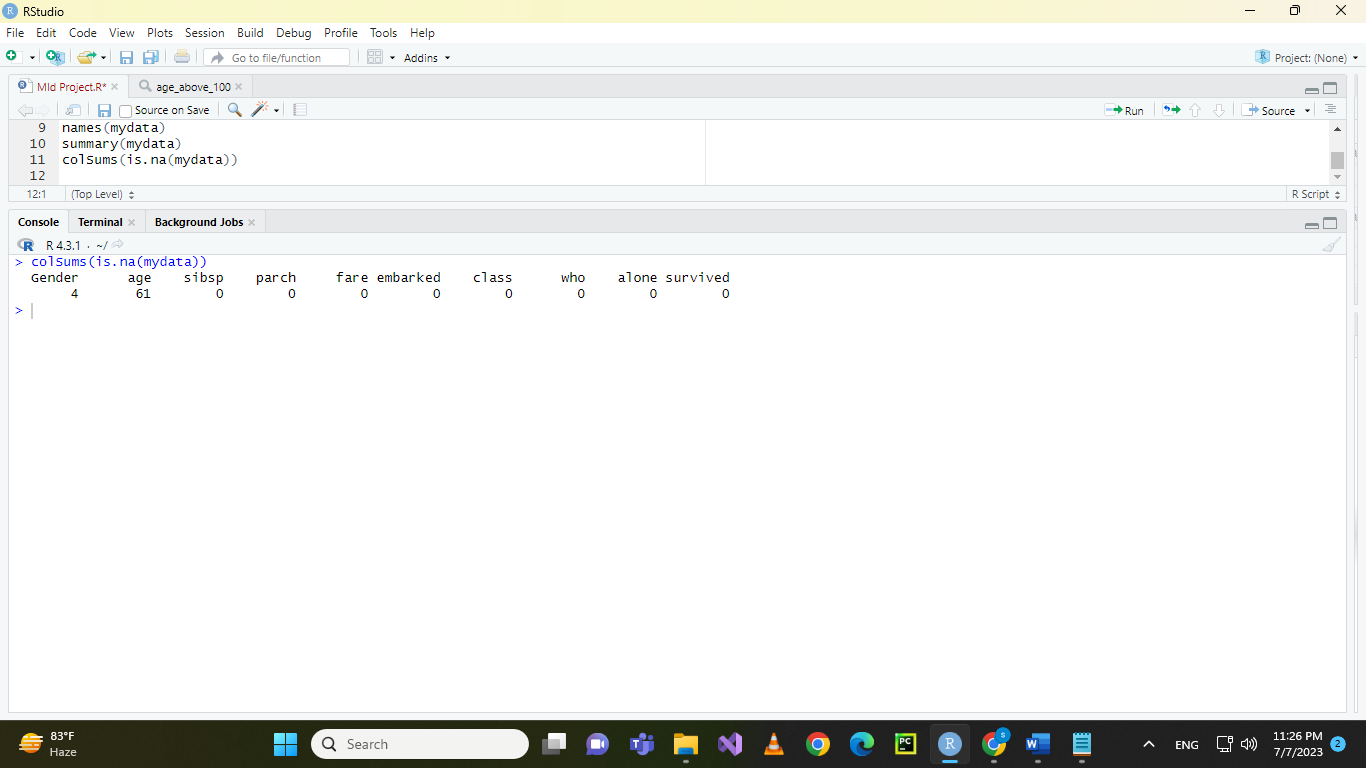


Figure : Missing values in particular attributes

**Types of stributes:**

str(mydata)

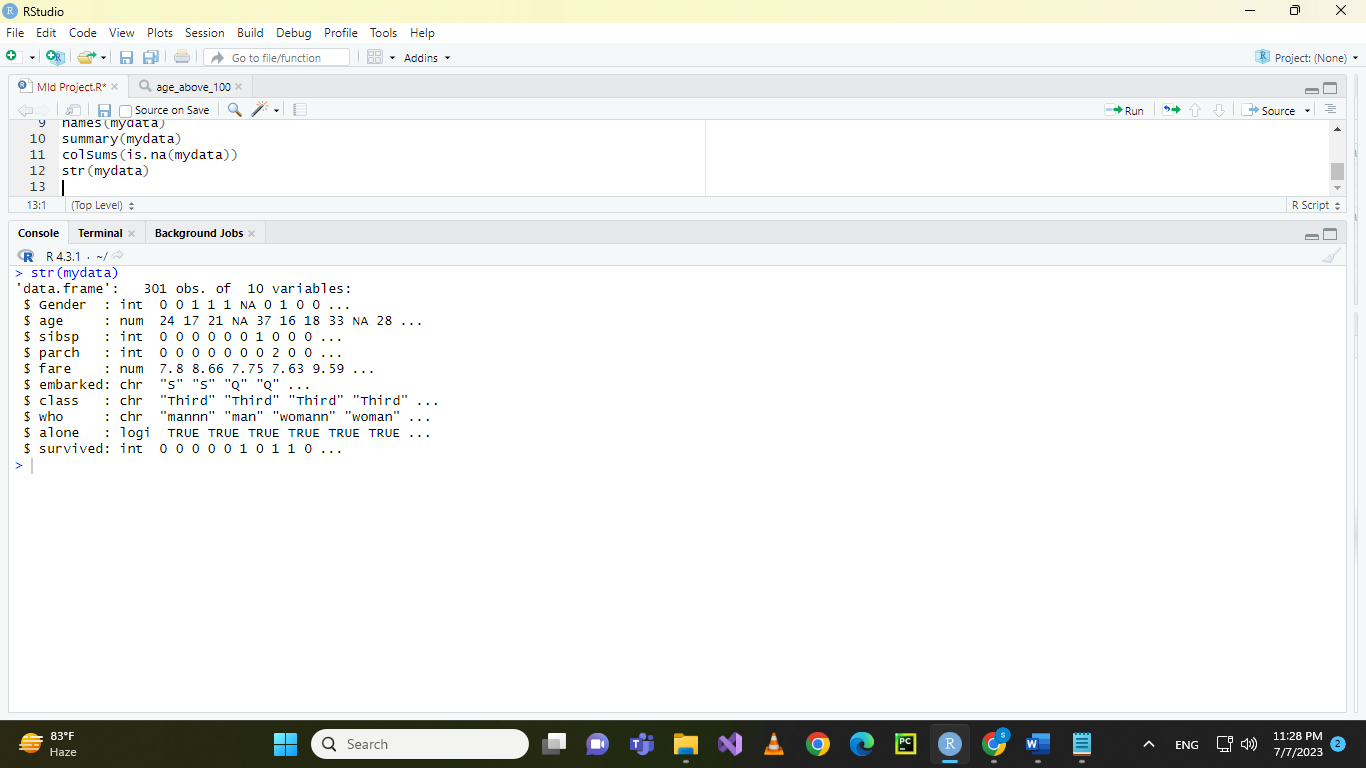


Figure : Types of attributes

**Data Visualization:**

**Age:**

**boxplot(mydata$age, main = "Box Plot of Age", ylab = "Age")**

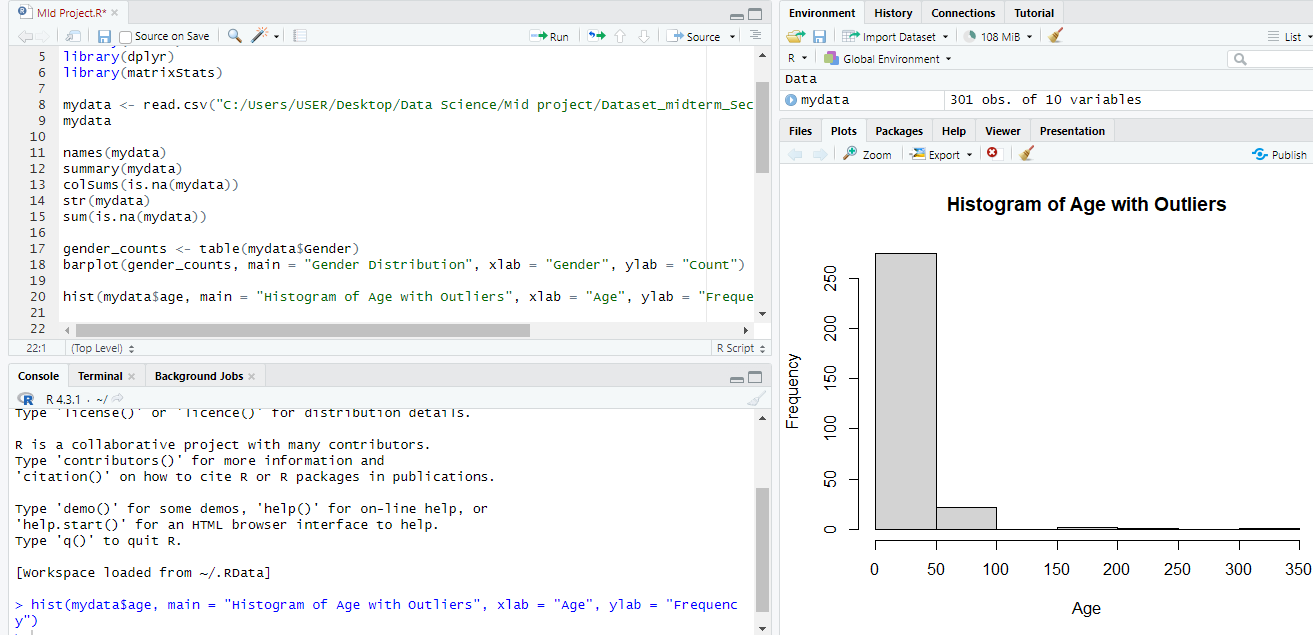


Figure : Histogram of age attribute

In the figure, it visible that some ages have exceeded 100. This means they are clearly outliers as they are significantly higher than the natural age of any person. To deal with this problem the instances with the outliers were removed. After removal of the outliers the histogram with the required code is given below:

data\_filtered <- mydata %>% filter(age < 100)

hist(data\_filtered$age,main="Age without the outliers",xlab = "Age")

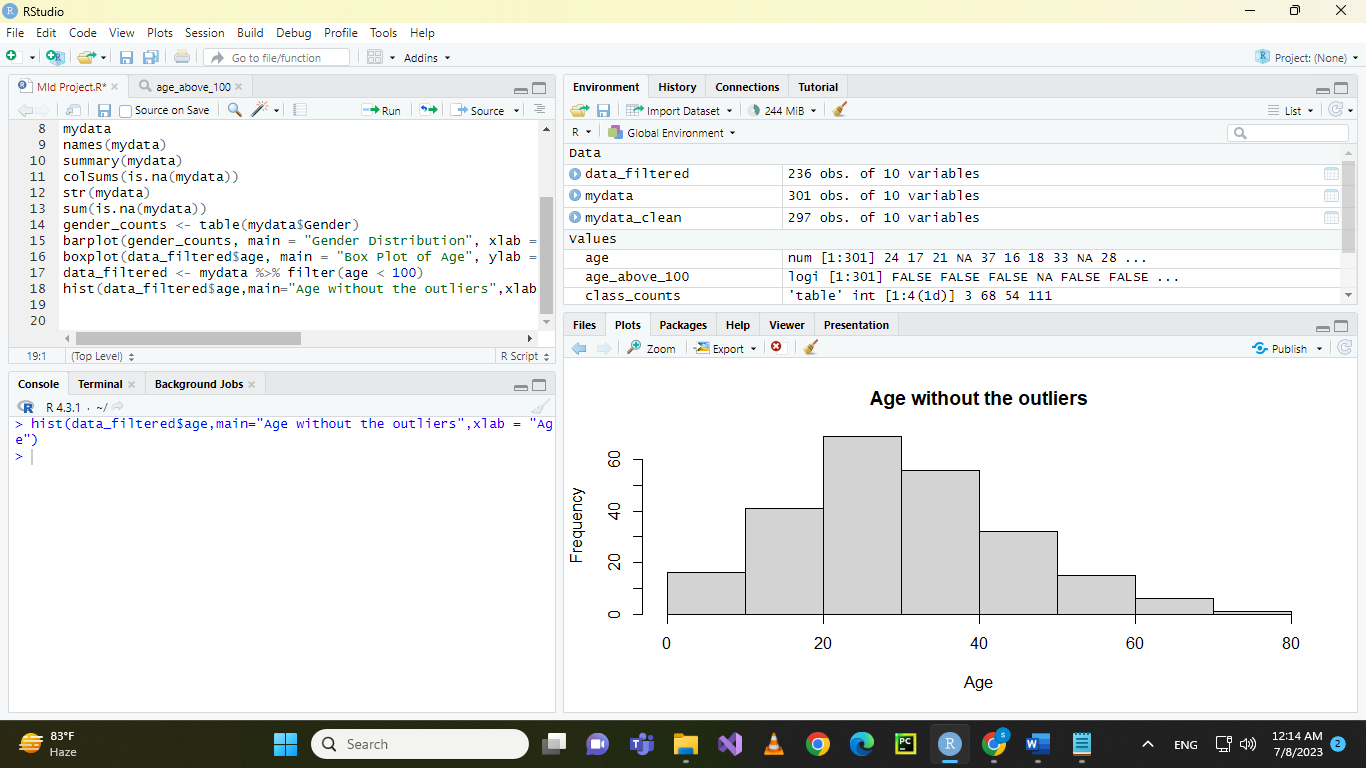


Figure : Histogram of age without outliers

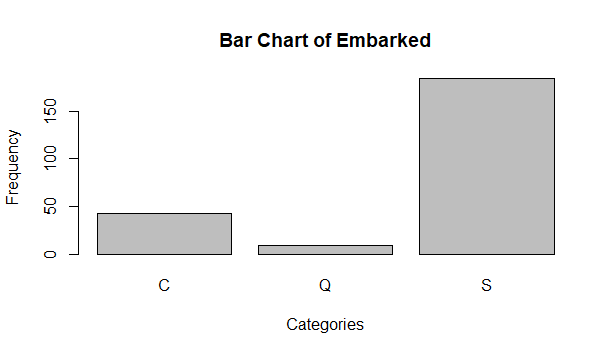
**Embarked**:

Figure : Bar chart of embarkation

**Gender Atrribute:**

barplot(gender\_counts, main = "Gender Distribution", xlab = "Gender", ylab = "Count")

age<-mydata$age

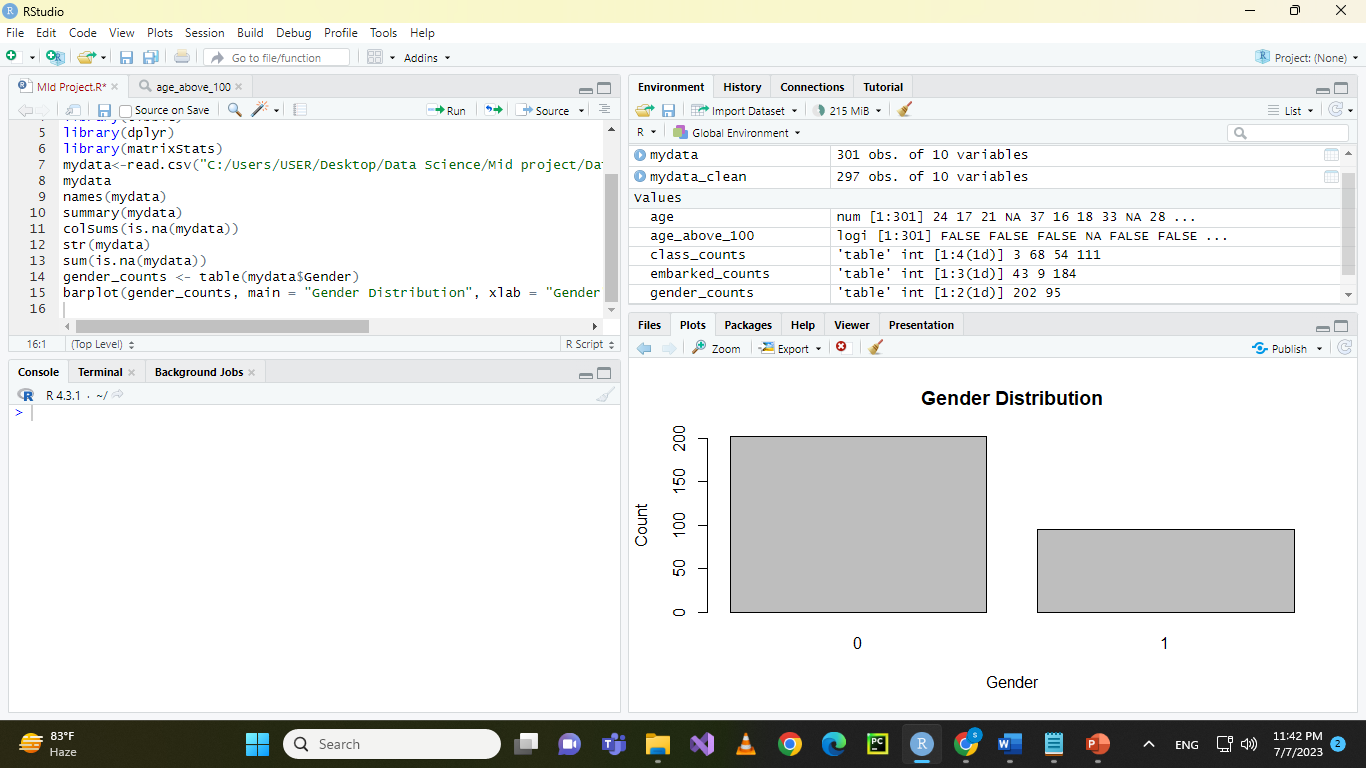


Figure : Barplot of gender

**Alone attribute**:

alone\_counts <- table(mydata$alone)

barplot(alone\_counts,xlab = "Alone", ylab = "Frequency", main = "Bart chart of surviving alone")

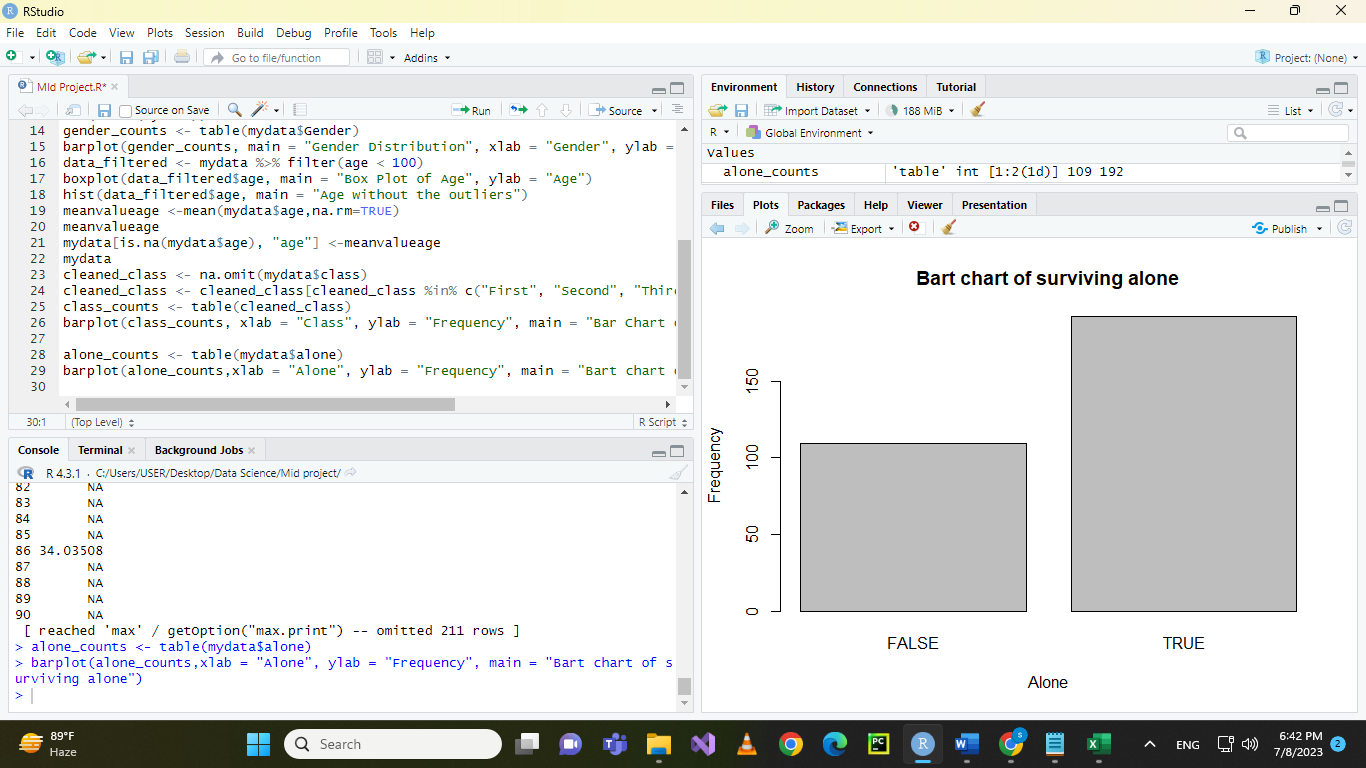


Figure : Barplot of alone attribute

**Recovering missing values:**

In the “age” attribute, 61 instances are missing. So, to deal with this problem, the missing values will be replaced by the mean of the rest of the data.

**Calculating the mean:**

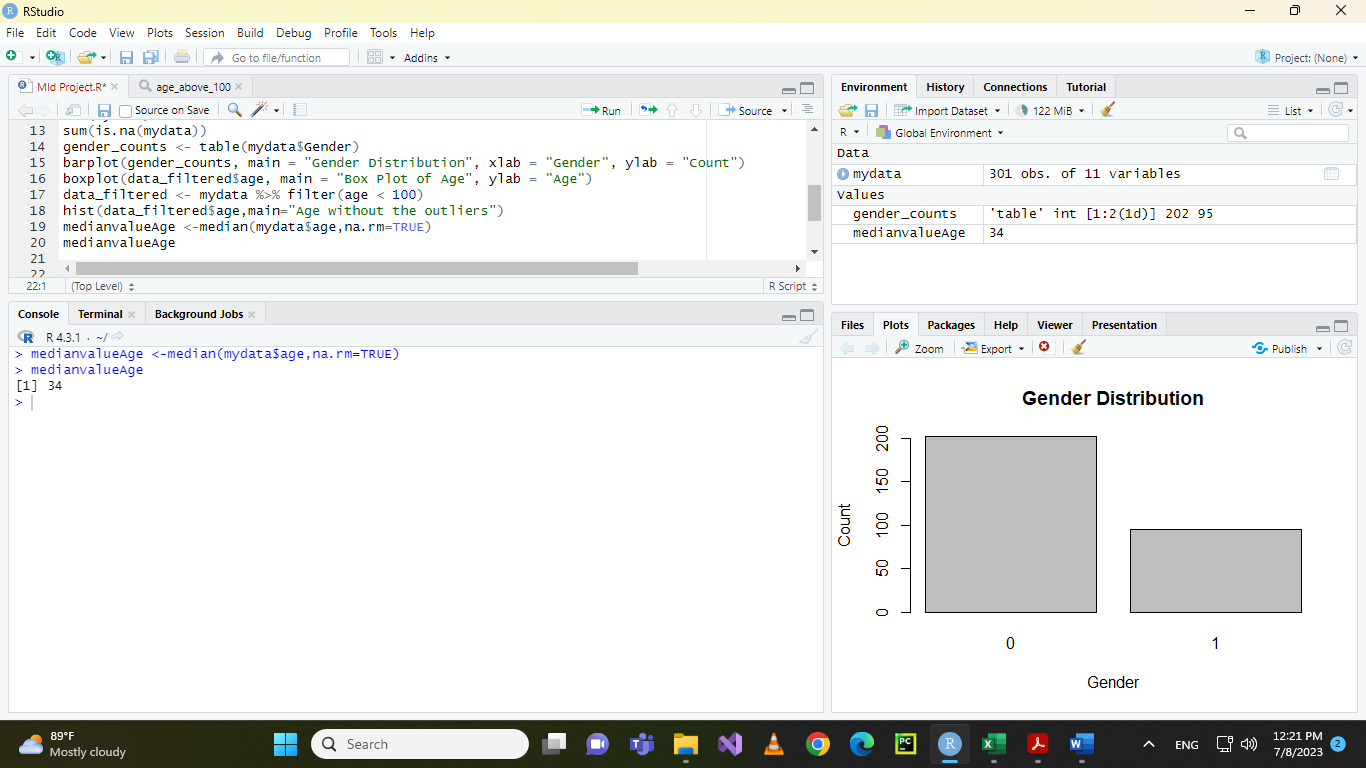
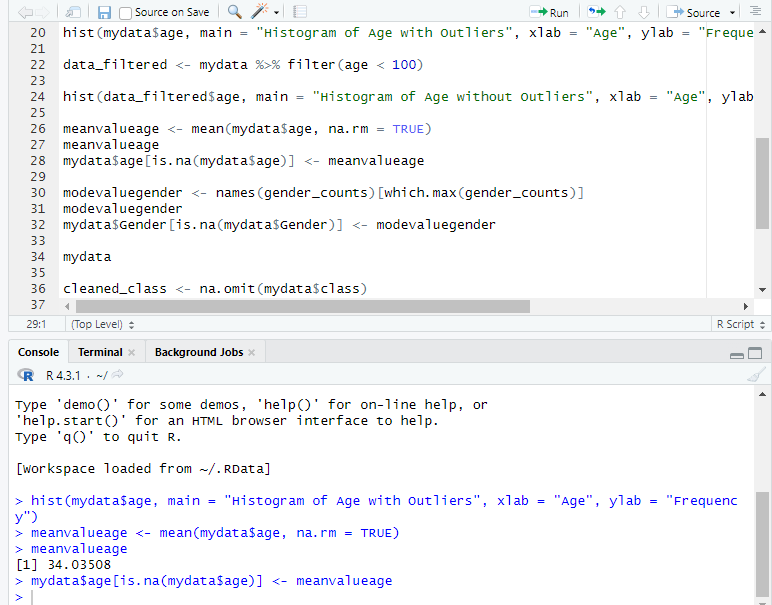


Figure : Calculating mean

**Replacing the missing values with mean:**

mydata[is.na(mydata$age), "Age"] <-medianvalueAge

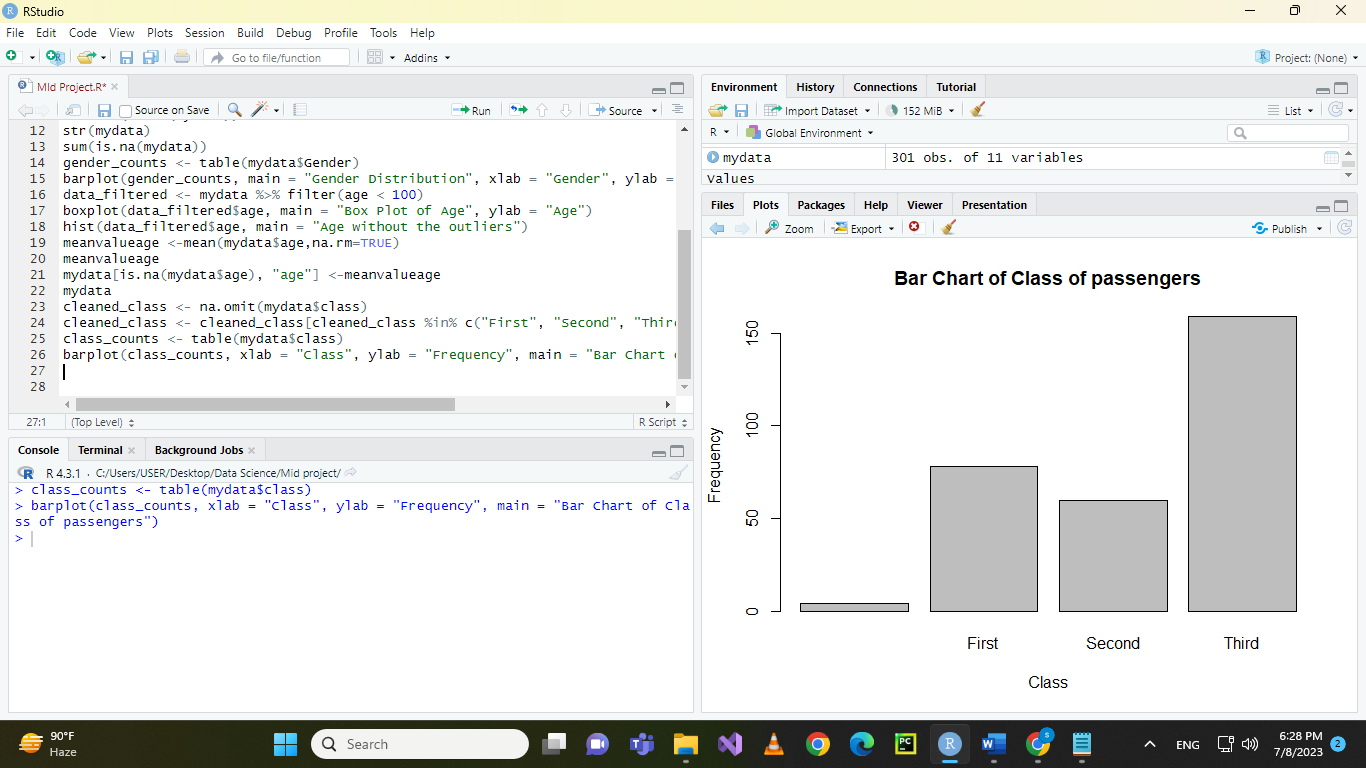
mydata



**Class Attribute**:

class\_counts <- table(mydata$class)

barplot(class\_counts, xlab = "Class", ylab = "Frequency", main = "Bar Chart of Class of passengers")



In the figure above it is seen that there are some unwanted values other than the pre-defined “First”, “Second” or “Third”.

So, these values can either be noisy, invalid values or missing values. These values are omitted by the omit() function.

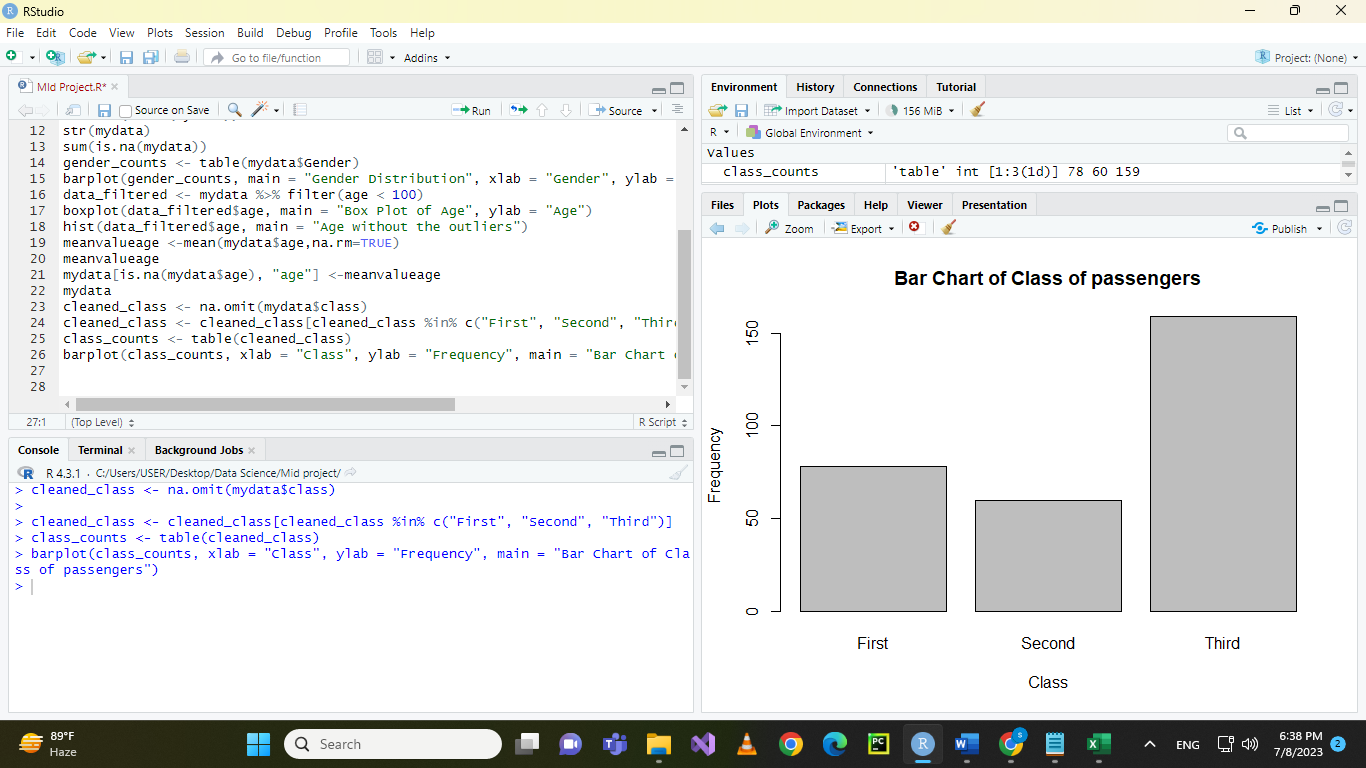
**After removing these values**:

cleaned\_class <- na.omit(mydata$class)

cleaned\_class <- cleaned\_class[cleaned\_class %in% c("First", "Second", "Third")]

class\_counts <- table(cleaned\_class)

barplot(class\_counts, xlab = "Class", ylab = "Frequency", main = "Bar Chart of Class of passengers")



In this screenshot we can see that the unknown values were omitted.

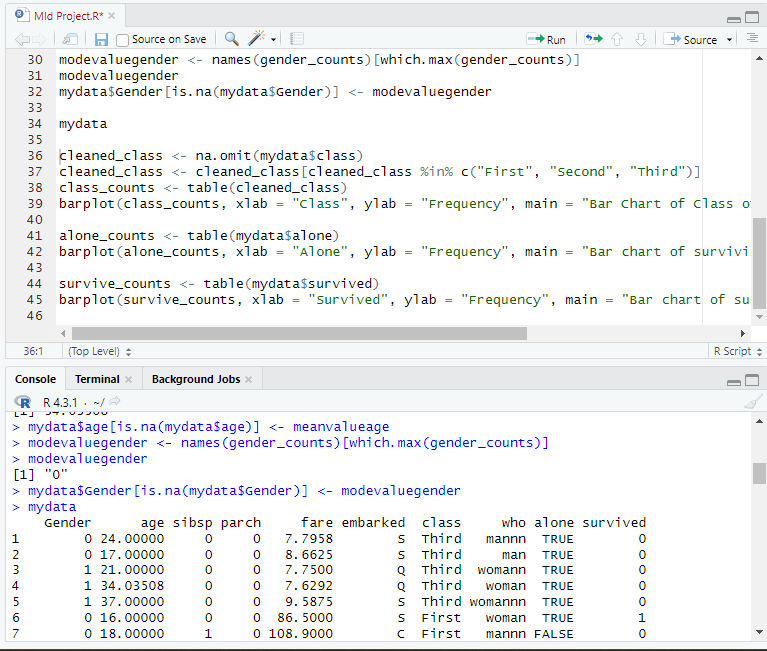
**Gender:**

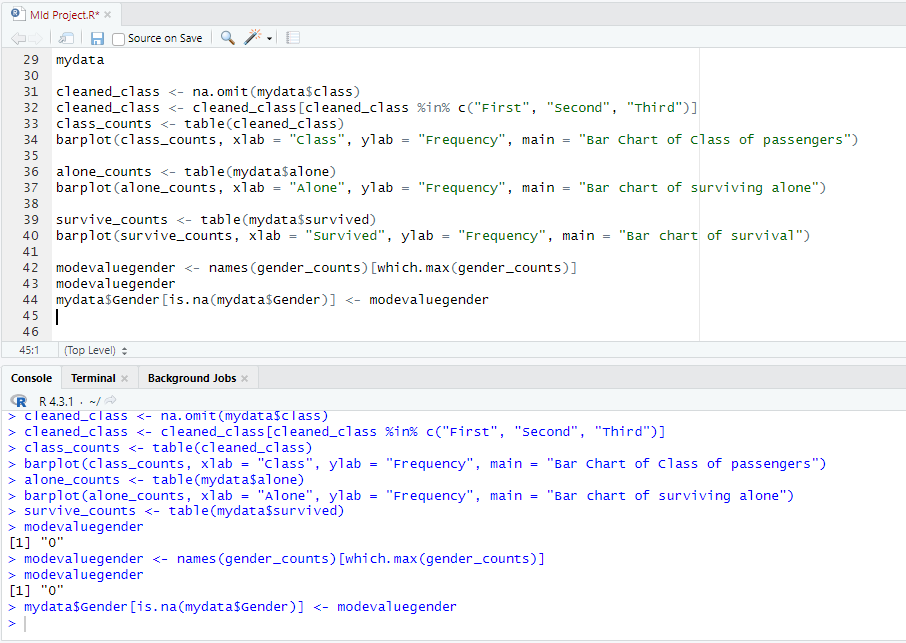
In gender column there are 4 null values. Since the value of gender is categorical, the null values are replaced by mode or most frequent values. The code for this operation is given below:

modevaluegender <- names(gender\_counts)[which.max(gender\_counts)]

modevaluegender

mydata$Gender[is.na(mydata$Gender)] <- modevaluegender





The null values in “gender” attribute are replaced by “0” which is the mode value of this particular attribute.