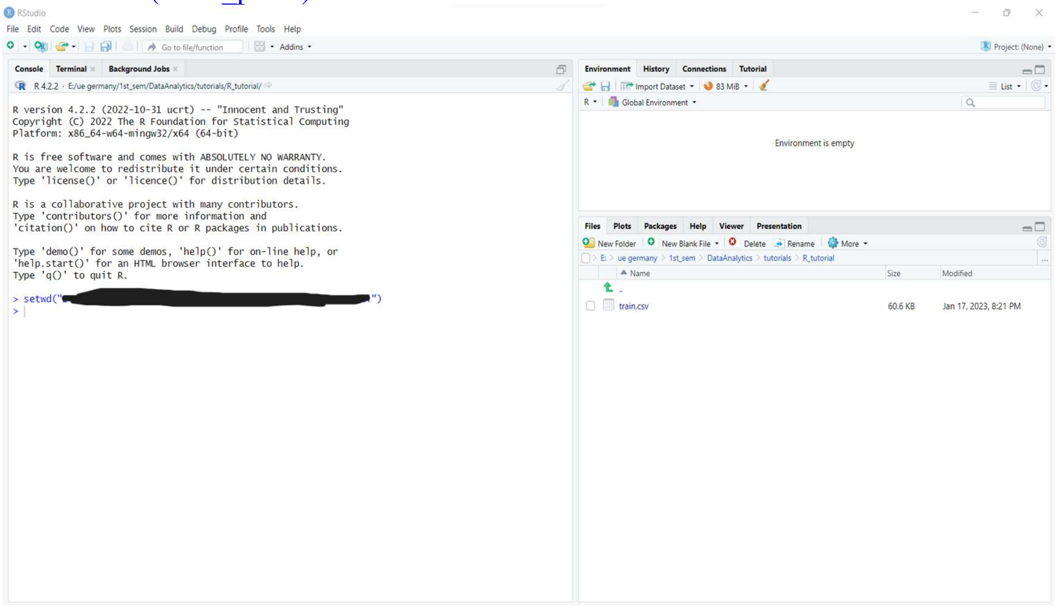
Data Analysis using R: Survivors in Titanic?

Dataset: Kaggle Titanic dataset
Setup: Rstudio
Steps:

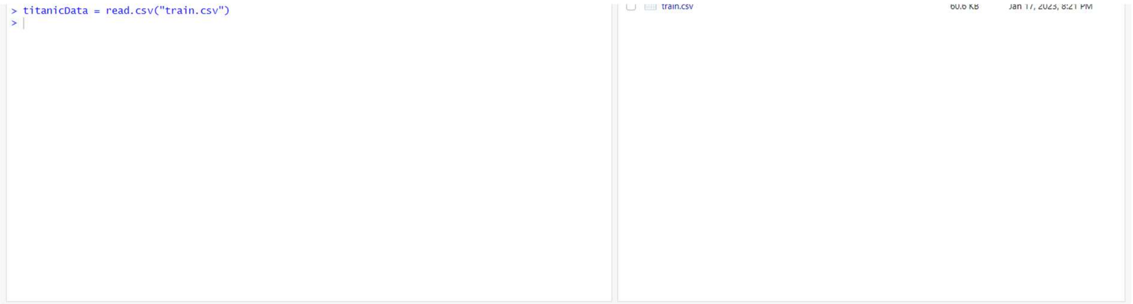
1. Open the Rstudio application.
    - Select Session>Set Working Directory>choose directory.
    - Navigate to the folder where your downloaded dataset is present.
    - It will automatically set it i.e

      > setwd("data_path")



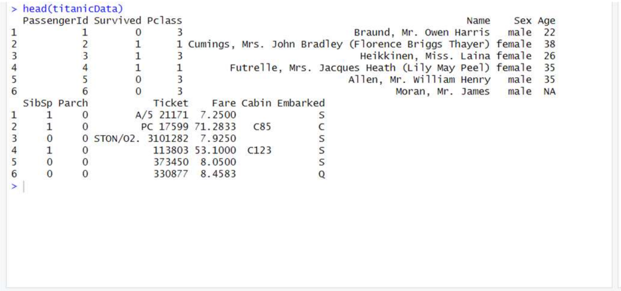
```
R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"
Copyright (C) 2022 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> setwd("[redacted]")
> 
```

2. Load and read the data.



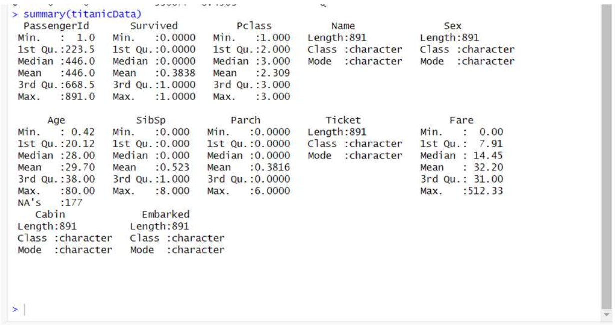
```
> titanicData = read.csv("train.csv")
> 
```

3. Look at the data.



```
> head(titanicData)
  PassengerId Survived Pclass                                                Name    Sex Age
1           1        0      3                             Braund, Mr. Owen Harris   male  22
2           2        1      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38
3           3        1      3                              Heikkinen, Miss. Laina female  26
4           4        1      1        Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35
5           5        0      3                            Allen, Mr. William Henry   male  35
6           6        0      3                                    Moran, Mr. James   male  NA
  SibSp Parch           Ticket    Fare Cabin Embarked
1     1     0        A/5 21171  7.2500                S
2     1     0         PC 17599 71.2833   C85          C
3     0     0 STON/O2. 3101282  7.9250                S
4     1     0           113803 53.1000  C123          S
5     0     0           373450  8.0500                S
6     0     0           330877  8.4583                Q
> |
```

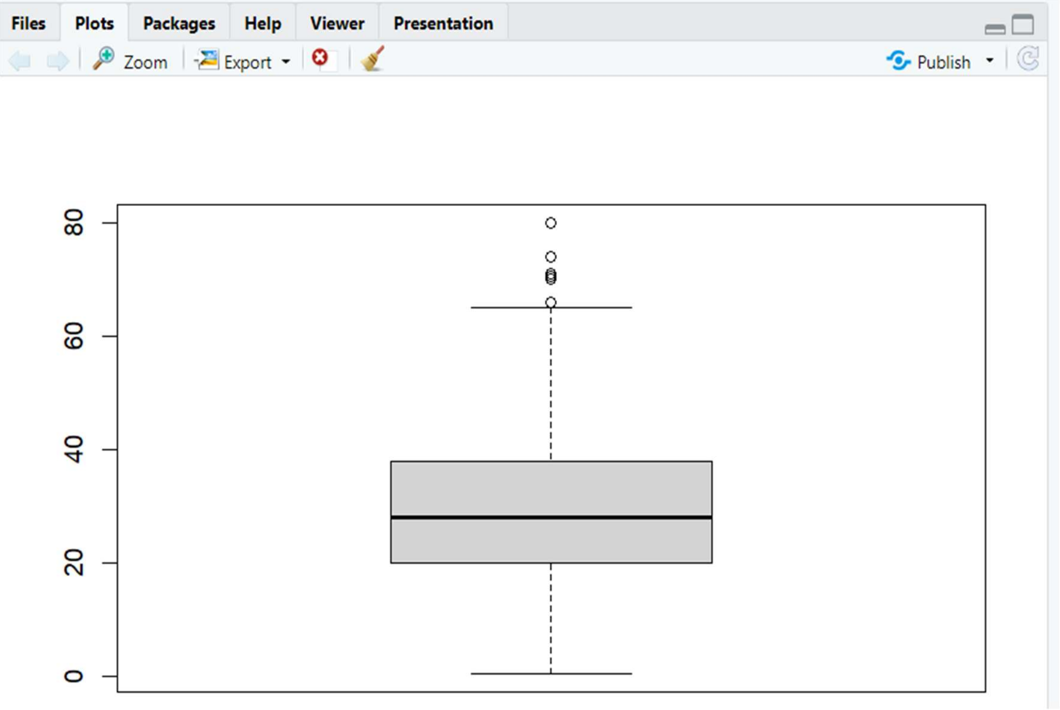
4. Descriptive statistics



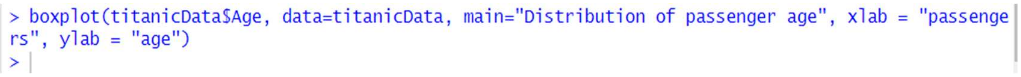
```
> summary(titanicData)
  PassengerId      Survived          Pclass          Name               Sex           
 Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Length:891         Length:891        
 1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Class :character   Class :character  
 Median :446.0   Median :0.0000   Median :3.000   Mode  :character   Mode  :character  
 Mean   :446.0   Mean   :0.3838   Mean   :2.309                                        
 3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000                                        
 Max.   :891.0   Max.   :1.0000   Max.   :3.000                                        

      Age            SibSp           Parch           Ticket               Fare       
 Min.   : 0.42   Min.   :0.000   Min.   :0.0000   Length:891         Min.   :  0.00  
 1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000   Class :character   1st Qu.:  7.91  
 Median :28.00   Median :0.000   Median :0.0000   Mode  :character   Median : 14.45  
 Mean   :29.70   Mean   :0.523   Mean   :0.3816                      Mean   : 32.20  
 3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000                      3rd Qu.: 31.00  
 Max.   :80.00   Max.   :8.000   Max.   :6.0000                      Max.   :512.33  
 NA's   :177                                                                         
    Cabin             Embarked        
 Length:891         Length:891        
 Class :character   Class :character  
 Mode  :character   Mode  :character  


> |
```

5. Distribution of age
    - Simple

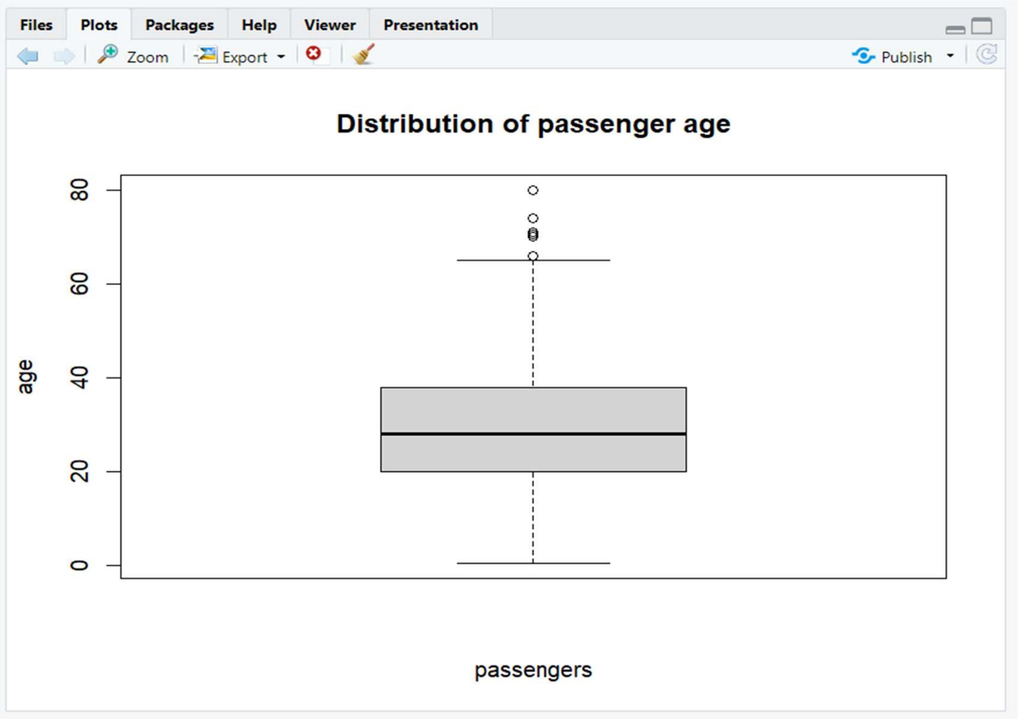
```
> boxplot(titanicData$Age, data=titanicData)
```



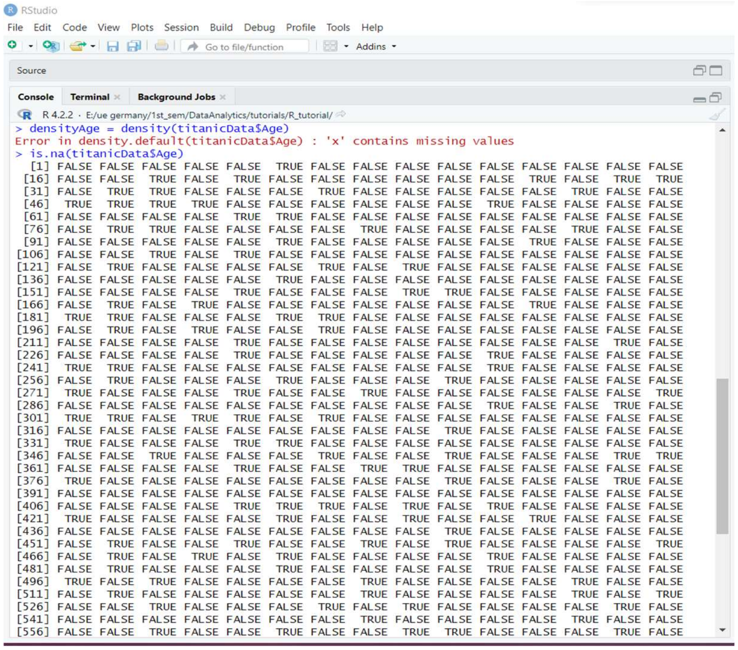
- with labels:



```
> boxplot(titanicData$Age, data=titanicData, main="Distribution of passenger age", xlab = "passengers", ylab = "age")
>
```

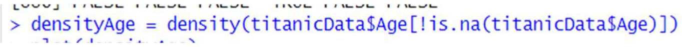


6. Density of age
   - densityAge = density(titanicData$Age) → Produces error message: missing values
   - Use “is.na” - checks if data set contains missing values.



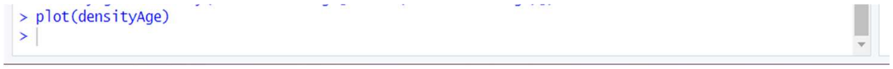
```
> densityAge = density(titanicData$Age)
Error in density.default(titanicData$Age) : 'x' contains missing values
> is.na(titanicData$Age)
  [1] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [16] FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE
 [31] FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
 [46]  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
 [61] FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [76] FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
 [91] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
[106] FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[121] FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[136] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[151] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE
[166] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
[181]  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[196] FALSE  TRUE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[211] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[226] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[241]  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[256] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[271]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
[286] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE
[301]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[316] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[331]  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[346] FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
[361] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[376]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE
[391] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[406] FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE
[421]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE
[436] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[451] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
[466] FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[481] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[496]  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[511] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE
[526] FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE
[541] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[556] FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE
```

- Extended calculation:

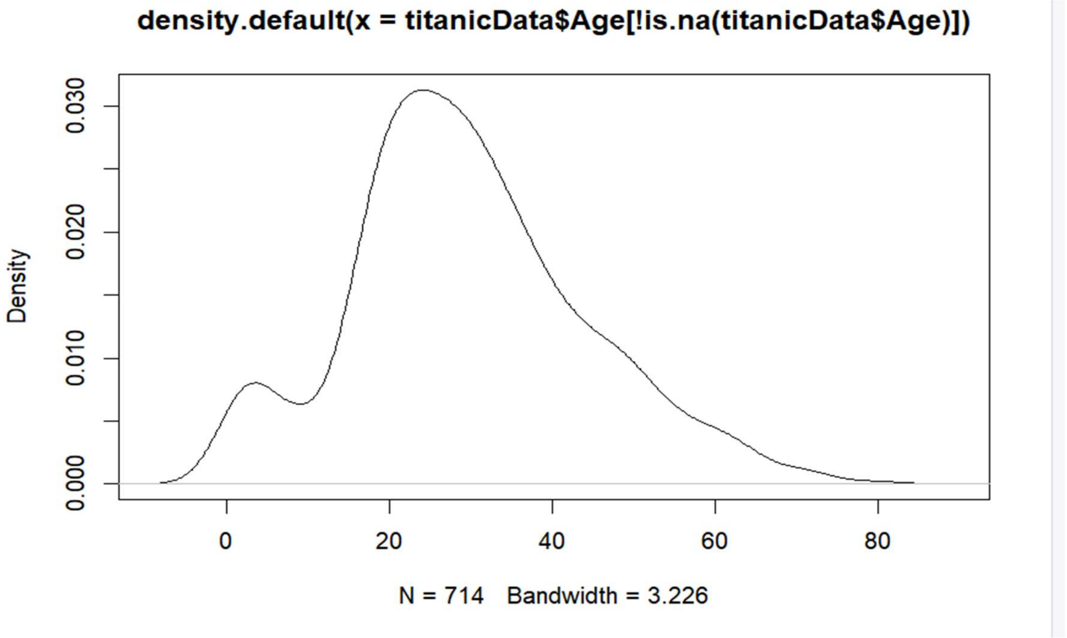


```
> densityAge = density(titanicData$Age[!is.na(titanicData$Age)])
```

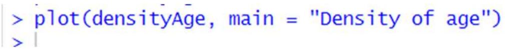
- Simple plot:



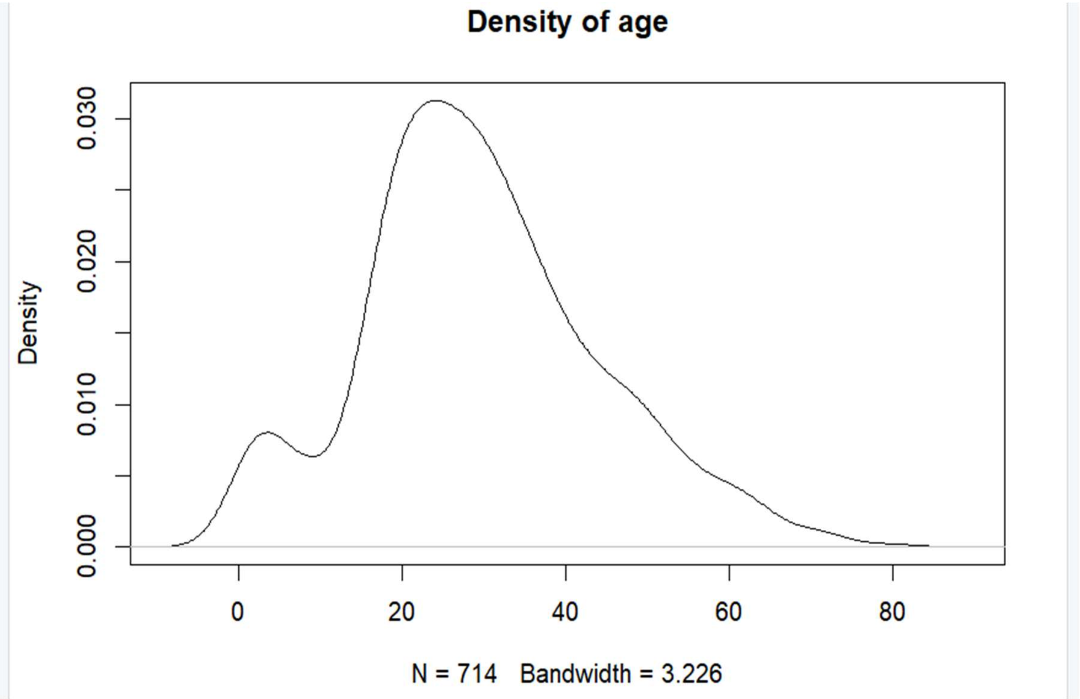
```
> plot(densityAge)
> 
```



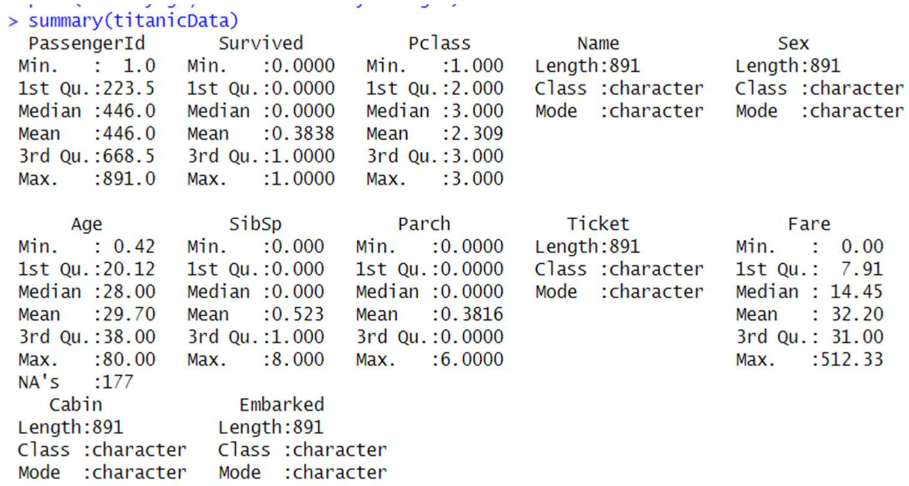
- With labels:



```
> plot(densityAge, main = "Density of age")
> 
```

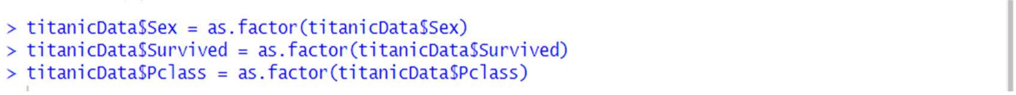


7. Convert categorical values to factors
   - summary shows that Sex, Survived, and PClass are categorical values
   - Sex: female, male, Survived: 0 (no), 1(yes) , PClass: 1, 2 and 3



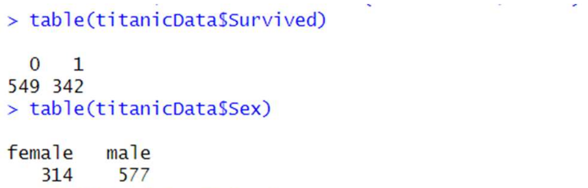
```
> summary(titanicData)
  PassengerId       Survived          Pclass          Name               Sex
 Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Length:891         Length:891
 1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Class :character   Class :character
 Median :446.0   Median :0.0000   Median :3.000   Mode  :character   Mode  :character
 Mean   :446.0   Mean   :0.3838   Mean   :2.309
 3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
 Max.   :891.0   Max.   :1.0000   Max.   :3.000

      Age            SibSp           Parch            Ticket               Fare
 Min.   : 0.42   Min.   :0.000   Min.   :0.0000   Length:891         Min.   :  0.00
 1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000   Class :character   1st Qu.:  7.91
 Median :28.00   Median :0.000   Median :0.0000   Mode  :character   Median : 14.45
 Mean   :29.70   Mean   :0.523   Mean   :0.3816                      Mean   : 32.20
 3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000                      3rd Qu.: 31.00
 Max.   :80.00   Max.   :8.000   Max.   :6.0000                      Max.   :512.33
 NA's   :177
    Cabin             Embarked
 Length:891         Length:891
 Class :character   Class :character
 Mode  :character   Mode  :character
```

   - Factors are used to represent categorical data. Necessary for plotting and analysis.
   - "as.factor(x)" converts a vector of values to a factor:

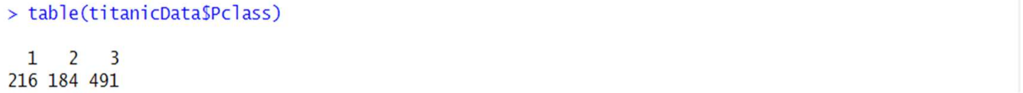


```
> titanicData$Sex = as.factor(titanicData$Sex)
> titanicData$Survived = as.factor(titanicData$Survived)
> titanicData$Pclass = as.factor(titanicData$Pclass)
```

8. Count the number of occurrences
   - The function table() uses factors to build a contingency table and counts the factor levels:

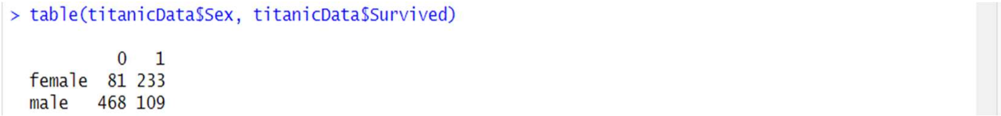


```
> table(titanicData$Survived)

  0   1
549 342
> table(titanicData$Sex)

female   male
   314    577
```



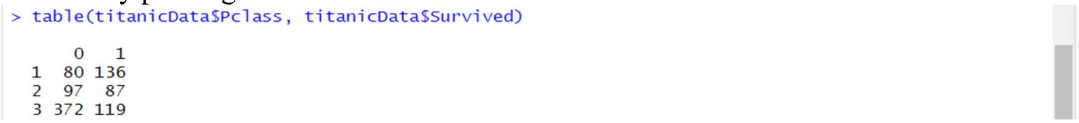
```
> table(titanicData$Pclass)

  1   2   3
216 184 491
```

   - With more factors, the function table() shows the number of occurrences for every combination of each factor levels.
   - How many female and male passengers survived?



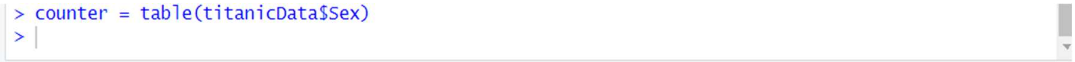
```
> table(titanicData$Sex, titanicData$Survived)

          0   1
  female  81 233
  male   468 109
```

   - How many passengers survived in the classes?

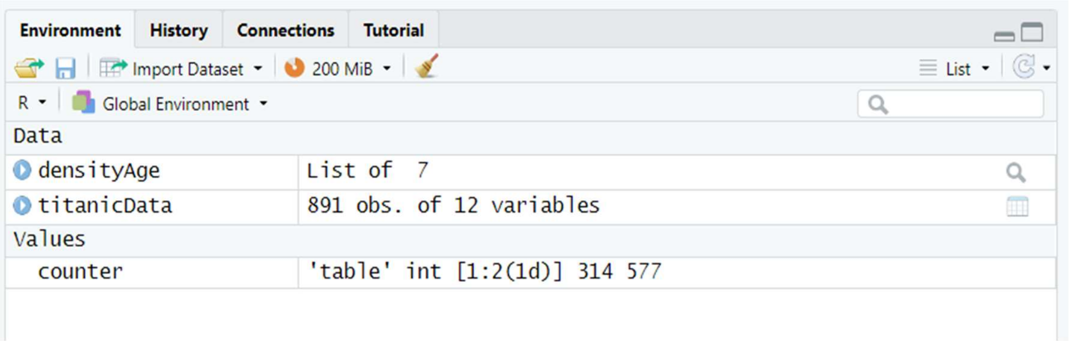


```
> table(titanicData$Pclass, titanicData$Survived)

      0   1
  1  80 136
  2  97  87
  3 372 119
```

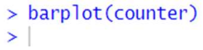
9. Plot the distributions in bar plots
   - Create variable for table:



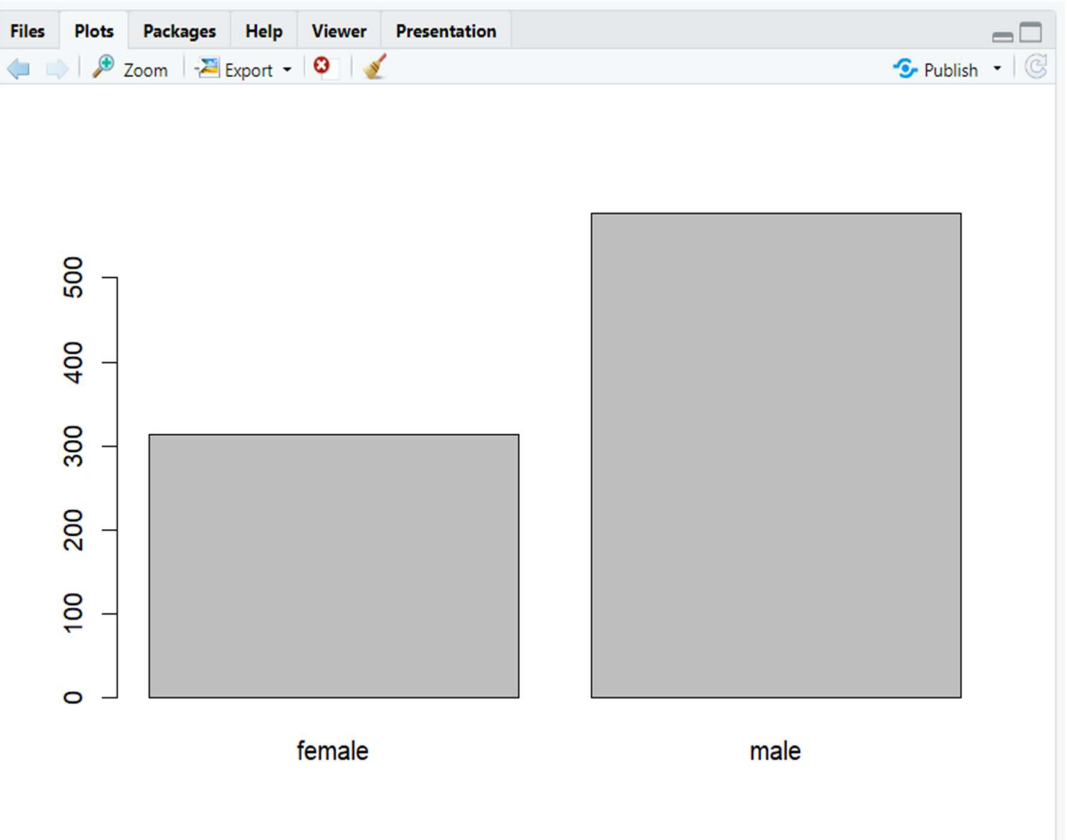
```
> counter = table(titanicData$Sex)
> 
```



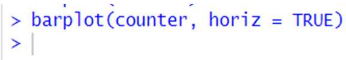
```
Data
densityAge      List of  7
titanicData     891 obs. of 12 variables
Values
counter         'table' int [1:2(1d)] 314 577
```



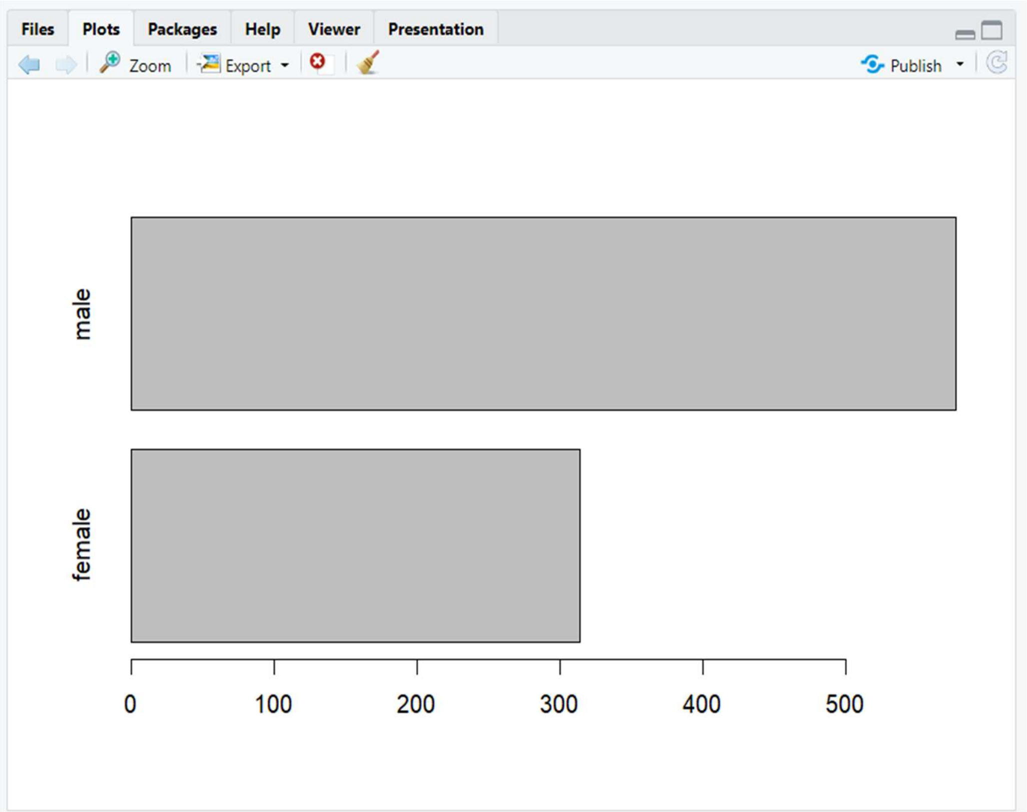
```
> barplot(counter)
> 
```



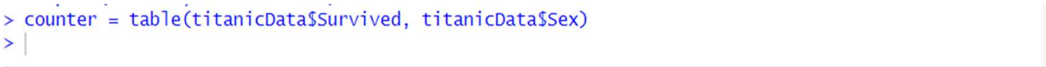
   - Adapt plots as needed:
     a) horizontal barplot with "horiz=" TRUE or FALSE



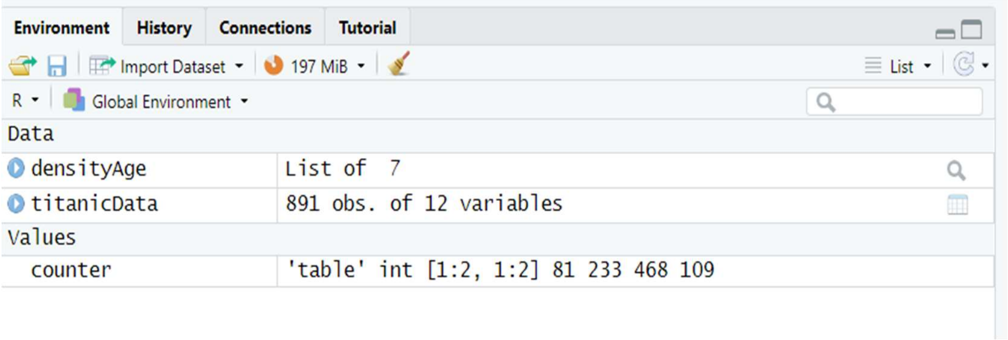
```
> barplot(counter, horiz = TRUE)
> 
```



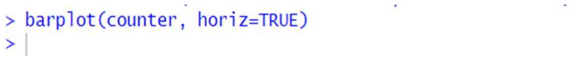
b) Split for survived (1) or not (0):



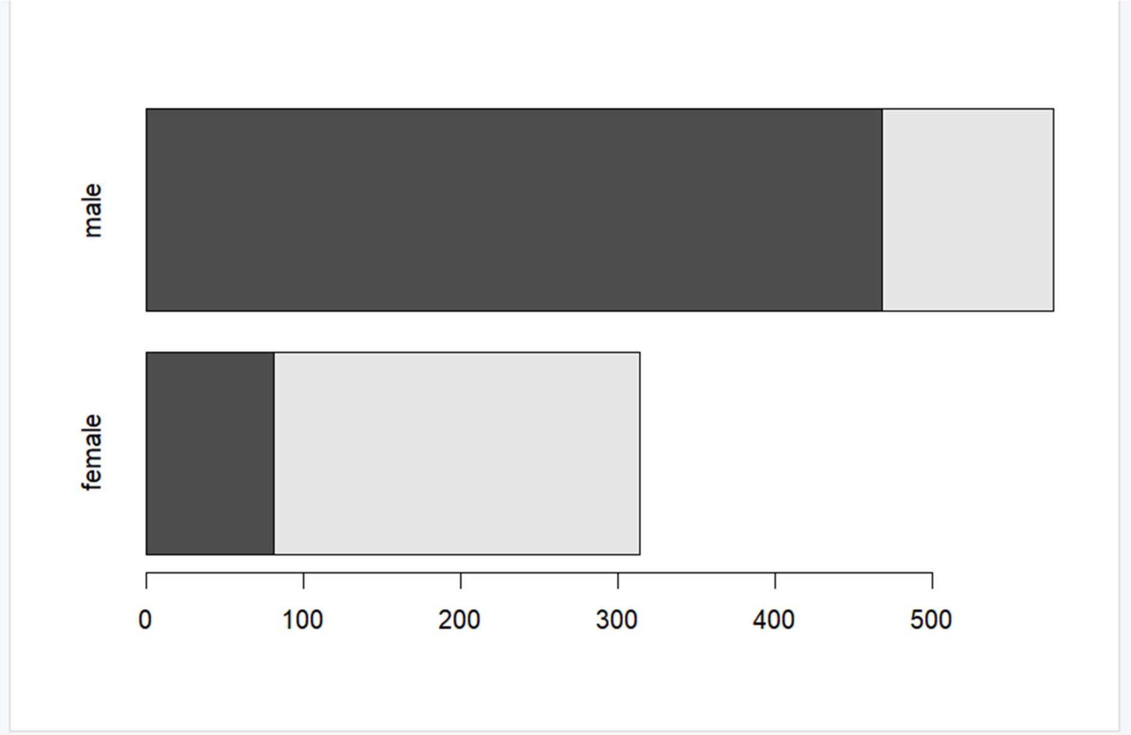
```
> counter = table(titanicData$Survived, titanicData$Sex)
> 
```



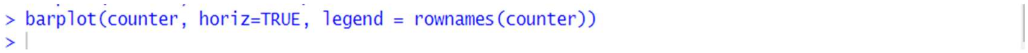
```
Data
densityAge      List of  7
titanicData     891 obs. of 12 variables
Values
  counter       'table' int [1:2, 1:2] 81 233 468 109
```



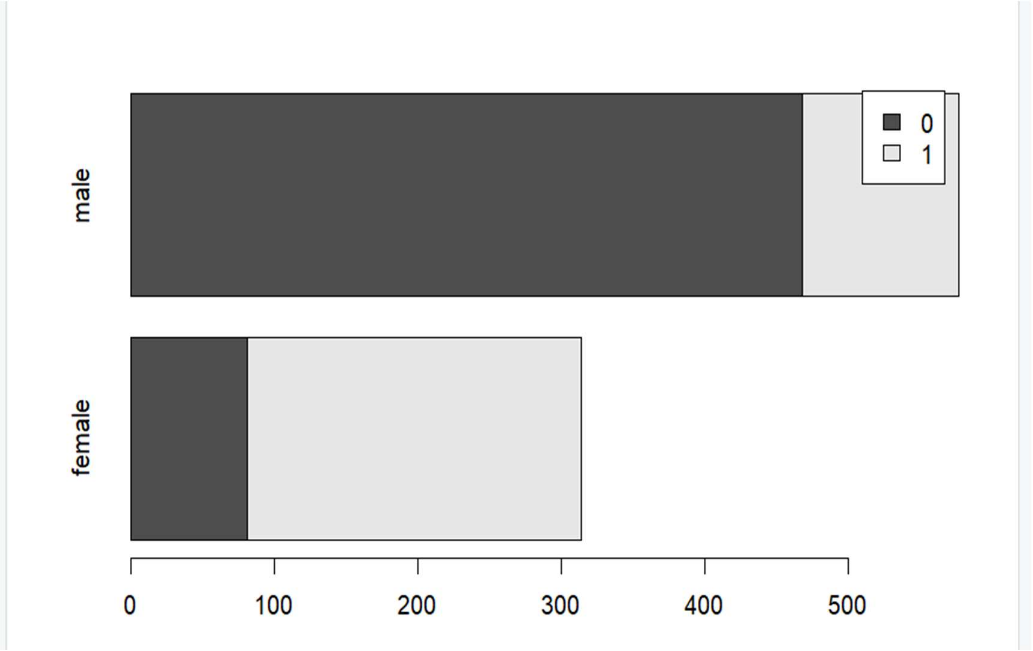
```
> barplot(counter, horiz=TRUE)
> 
```



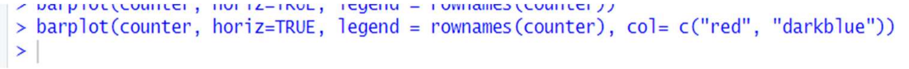
c) add legend:



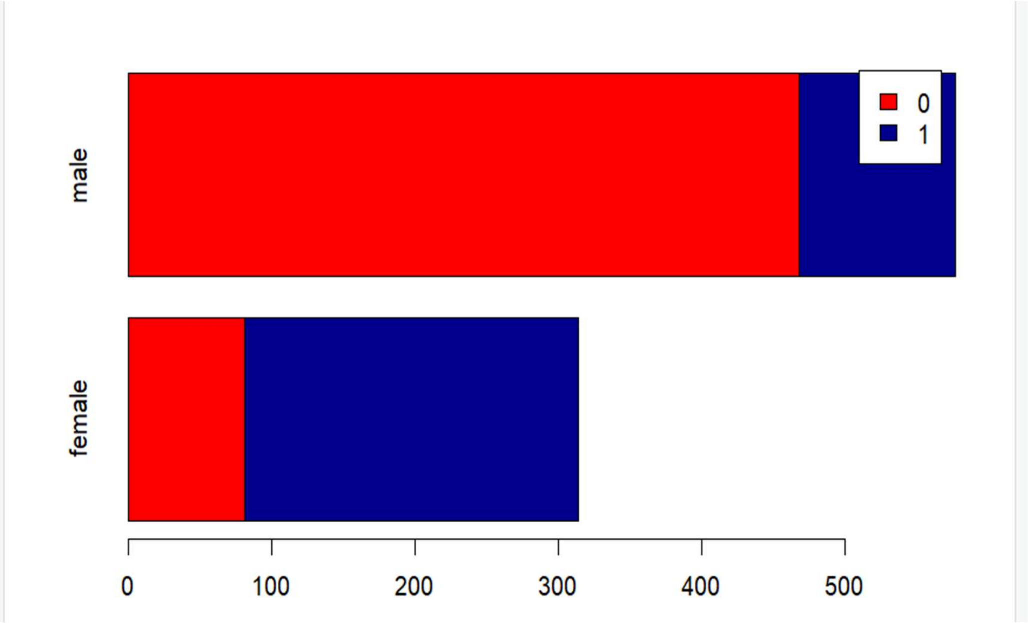
```
> barplot(counter, horiz=TRUE, legend = rownames(counter))
> |
```



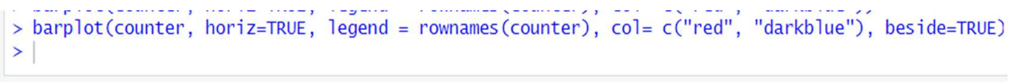
d) change colors:



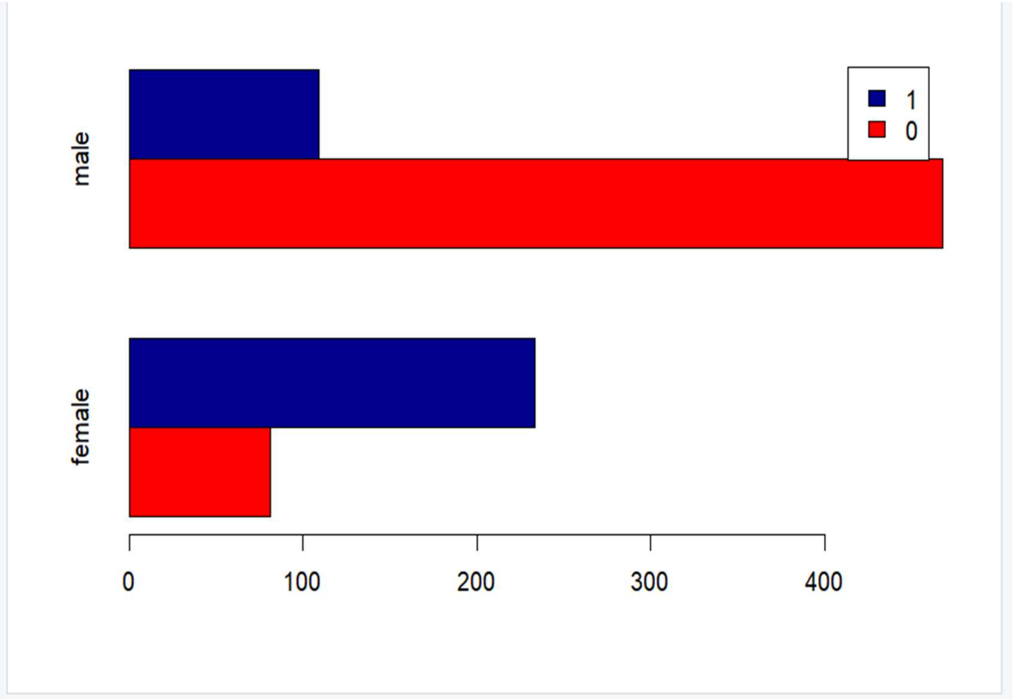
```
> barplot(counter, horiz=TRUE, legend = rownames(counter))
> barplot(counter, horiz=TRUE, legend = rownames(counter), col= c("red", "darkblue"))
> |
```



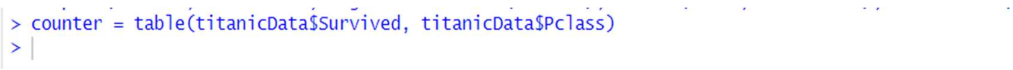
e) Put bars in plot beside each other



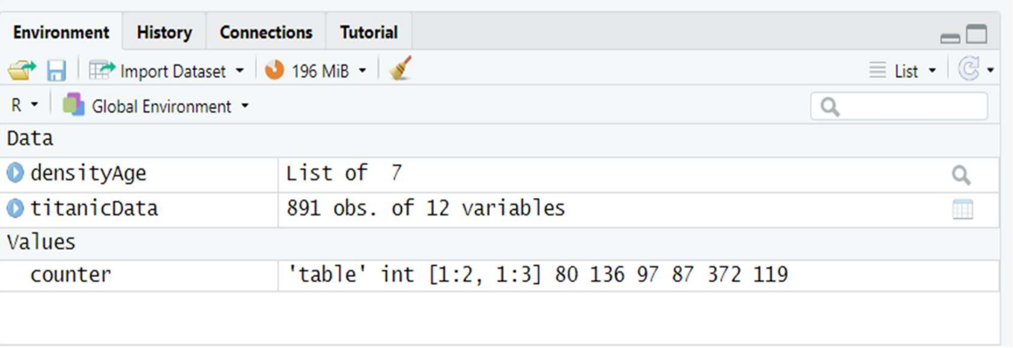
```
> barplot(counter, horiz=TRUE, legend = rownames(counter), col= c("red", "darkblue"))
> barplot(counter, horiz=TRUE, legend = rownames(counter), col= c("red", "darkblue"), beside=TRUE)
> |
```



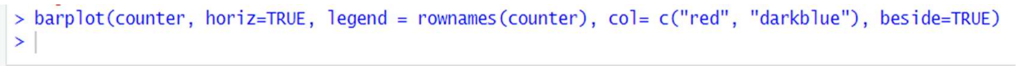
f) plot class vs survived



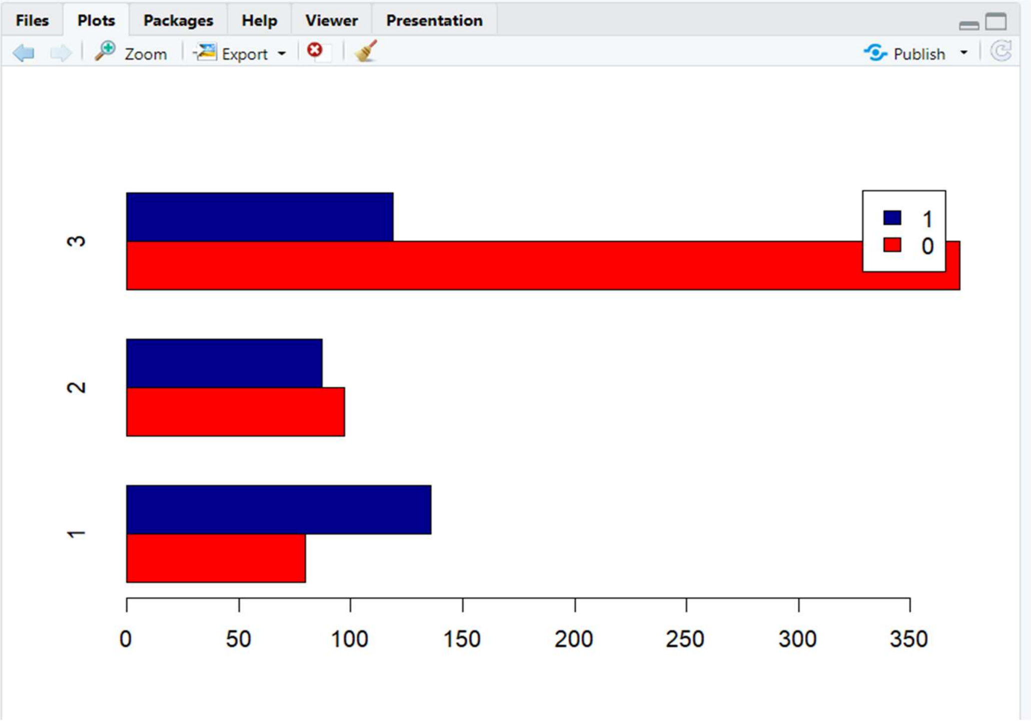
```
> counter = table(titanicData$Survived, titanicData$Pclass)
> |
```



Data	
densityAge	List of 7
titanicData	891 obs. of 12 variables
Values	
counter	'table' int [1:2, 1:3] 80 136 97 87 372 119

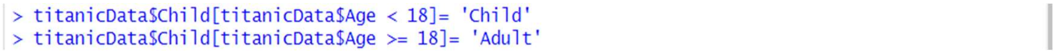


```
> barplot(counter, horiz=TRUE, legend = rownames(counter), col= c("red", "darkblue"), beside=TRUE)
> |
```



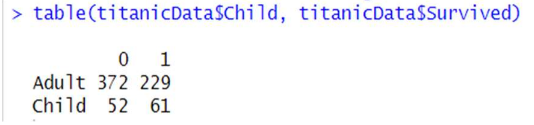
10. Survival of children vs adults?

a. Label child and adult, depending on Age



```
> titanicData$Child[titanicData$Age < 18]= 'Child'
> titanicData$Child[titanicData$Age >= 18]= 'Adult'
```

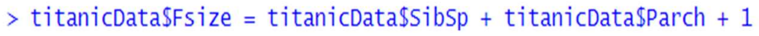
b. Show occurrences of survived and not survived adults and children



```
> table(titanicData$Child, titanicData$Survived)

         0   1
  Adult 372 229
  Child  52  61
```

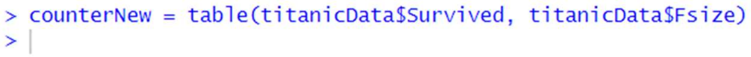
c. Did big families survive?

- New variable family size: Fsize
- Parch - number of parents / children aboard the titanic
- SibSp - number of siblings / spouses aboard the titanic
- Create Fsize - family size (new variable):

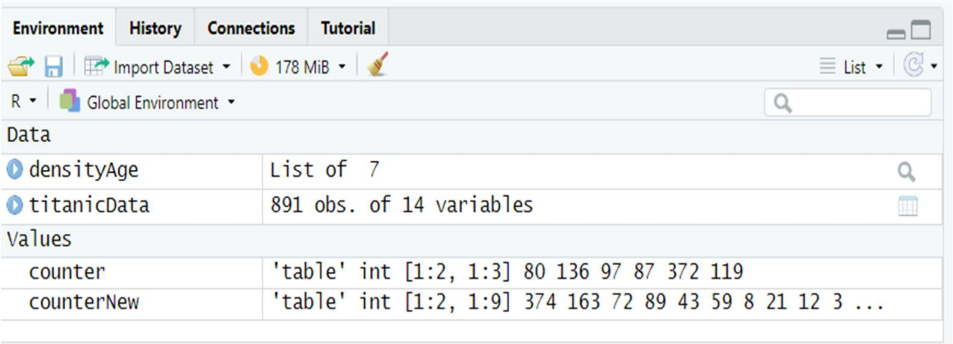


```
> titanicData$Fsize = titanicData$SibSp + titanicData$Parch + 1
```

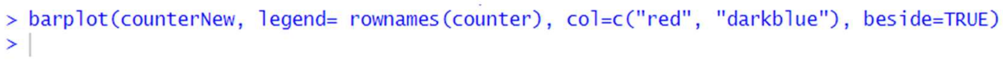
- Create table with counts and plot



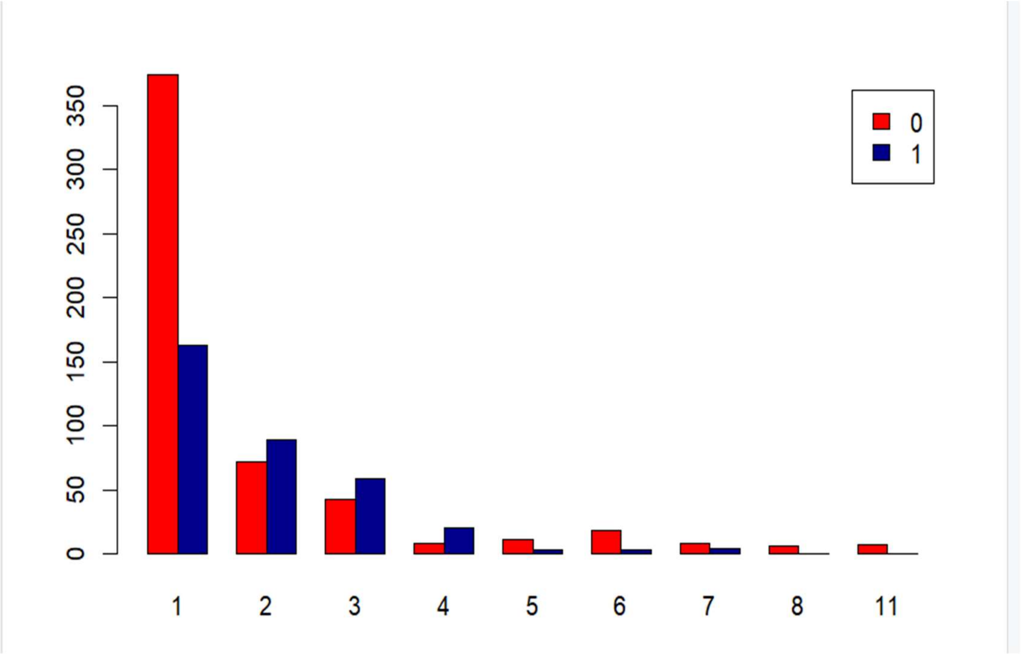
```
> counterNew = table(titanicData$Survived, titanicData$Fsize)
> 
```



```
Data
densityAge      List of  7
titanicData     891 obs. of 14 variables
Values
  counter       'table' int [1:2, 1:3] 80 136 97 87 372 119
  counterNew    'table' int [1:2, 1:9] 374 163 72 89 43 59 8 21 12 3 ...
```



```
> barplot(counterNew, legend= rownames(counter), col=c("red", "darkblue"), beside=TRUE)
> 
```



0
50
100
150
200
250
300
350
🟥 0
🟦 1
1
2
3
4
5
6
7
8
11