RWorksheet#4A

Marivic Amuan

2023-10-28

```
#1. The table below shows the data about shoe size and height. Create a data frame.

#1A. Describe the data = AS I input each correspoding data inside the data frame, the data.frame allows

householdData <- data.frame(
  ShoeSize = c(6.5,9.0, 8.5, 8.5, 10.5, 7.0, 9.5, 9.0, 13.0, 7.5, 10.5, 8.5, 12.0, 10.5, 13.0, 11.5, 8.5
  Height = c(66.0, 68.0, 64.5, 65.0, 70.0, 64.0, 70.0, 71.0, 72.0, 64.0, 74.5, 67.0, 71.0, 71.0, 77.0, 7
  Gender = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F", "
)

householdData
```

```
##    ShoeSize Height Gender
## 1       6.5   66.0      F
## 2       9.0   68.0      F
## 3       8.5   64.5      F
## 4       8.5   65.0      F
## 5      10.5   70.0      M
## 6       7.0   64.0      F
## 7       9.5   70.0      F
## 8       9.0   71.0      F
## 9      13.0   72.0      M
## 10      7.5   64.0      F
## 11     10.5   74.5      M
## 12      8.5   67.0      F
## 13     12.0   71.0      M
## 14     10.5   71.0      M
## 15     13.0   77.0      M
## 16     11.5   72.0      M
## 17      8.5   59.0      F
## 18      5.0   62.0      F
## 19     10.0   72.0      M
## 20      6.5   66.0      F
## 21      7.5   64.0      F
## 22      8.5   67.0      M
## 23     10.5   73.0      M
## 24      8.5   69.0      F
## 25     10.5   72.0      M
## 26     11.0   70.0      M
## 27      9.0   69.0      M
## 28     13.0   70.0      M
```

```r
#1B. Create a subset by males and females with their corresponding shoe size and height. What its resul
```

```r
subsetMale <- householdData[householdData$Gender == "M", c("ShoeSize", "Height", "Gender")]
subsetMale
```

```
##    ShoeSize Height Gender
## 5      10.5   70.0      M
## 9      13.0   72.0      M
## 11     10.5   74.5      M
## 13     12.0   71.0      M
## 14     10.5   71.0      M
## 15     13.0   77.0      M
## 16     11.5   72.0      M
## 19     10.0   72.0      M
## 22      8.5   67.0      M
## 23     10.5   73.0      M
## 25     10.5   72.0      M
## 26     11.0   70.0      M
## 27      9.0   69.0      M
## 28     13.0   70.0      M
```

```r
subsetFemale <- householdData[householdData$Gender == "F", c("ShoeSize", "Height", "Gender")]
subsetFemale
```

```
##    ShoeSize Height Gender
## 1       6.5   66.0      F
## 2       9.0   68.0      F
## 3       8.5   64.5      F
## 4       8.5   65.0      F
## 6       7.0   64.0      F
## 7       9.5   70.0      F
## 8       9.0   71.0      F
## 10      7.5   64.0      F
## 12      8.5   67.0      F
## 17      8.5   59.0      F
## 18      5.0   62.0      F
## 20      6.5   66.0      F
## 21      7.5   64.0      F
## 24      8.5   69.0      F
```

```r
#1C. Find the mean of shoe size and height of the respondents. Write the R scripts and its result.
```

```r
meanShoeSize <- mean(householdData$ShoeSize)
meanShoeSize
```

```
## [1] 9.410714
```

```r
#RESULT: [1] 9.410714
```

```r
meanHeight <- mean(householdData$Height)
meanHeight
```

```
## [1] 68.57143
```

```r
#RESULT: [1] 68.57143
```

```r
#1D. Is there a relationship between shoe size and height? Why?
#ANSWER: There could be a relationship between shoe size and height because

correlation <- cor(householdData$ShoeSize, householdData$Height)
correlation
```

```
## [1] 0.7766089
```

```r
#RESULT: [1] 0.7766089

#2. Construct character vector months to a factor with factor() and assign the result to factor_months_

months <- c("March","April","January","November","January", "September","October","September","November"
months
```

```
##  [1] "March"     "April"     "January"   "November"  "January"   "September"
##  [7] "October"   "September" "November"  "August"    "January"   "November"
## [13] "November"  "February"  "May"       "August"    "July"      "December"
## [19] "August"    "August"    "September" "November"  "February"  "April"
```

```r
monthsFactor <- factor(months)
monthsFactor
```

```
##  [1] March     April     January   November  January   September October
##  [8] September November  August    January   November  November  February
## [15] May       August    July      December  August    August    September
## [22] November  February  April
## 11 Levels: April August December February January July March May ... September
```

```r
#3. Then check the summary() of the months_vector and factor_months_vector. Interpret the results of bo
#Are they both equally useful in this case? =

summary(months)
```

```
##    Length     Class      Mode
##        24 character character
```

```r
summary(monthsFactor)
```

```
##     April    August  December  February   January      July     March       May
##         2         4         1         2         3         1         1         1
##  November   October September
##         5         1         3
```

```r
#4. Create a vector and factor for the table below.

direction <- c("East", "West", "North")
direction
```

```
## [1] "East"  "West"  "North"
```

```r
freq <- c(1 ,4, 3)
freq
```

```
## [1] 1 4 3
```

```r
new_order_data <- factor(direction,levels = c("East","West","North") , c(1,4,3))
print(new_order_data)
```

```
## [1] 1 4 3
```

```
## Levels: 1 4 3
```

```
#5. Enter the data below in Excel with file name = import_march.csv

imported <- read.table(file = "/cloud/project/Worksheet#4A/import_march.csv", header = TRUE, sep = ",")
    imported
```

```
##   Students Strategy.1 Strategy.2 Strategy.3
## 1     Male          8         10          8
## 2                   4          8          6
## 3                   0          6          4
## 4   Female         14          4         15
## 5                  10          2         12
## 6                   6          0          9
```

```
#5A.Import the excel file into the Environment Pane using read.table() function. Write the code.

 library(readr)
     import_march <- read_csv("/cloud/project/Worksheet#4A/import_march.csv")
```

```
## Rows: 6 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (1): Students
## dbl (3): Strategy 1, Strategy 2, Strategy 3
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#5B. View the dataset. Write the R scripts and its result.

head(import_march)
```

```
## # A tibble: 6 x 4
##   Students `Strategy 1` `Strategy 2` `Strategy 3`
##   <chr>           <dbl>        <dbl>        <dbl>
## 1 Male                8           10            8
## 2 <NA>                4            8            6
## 3 <NA>                0            6            4
## 4 Female             14            4           15
## 5 <NA>               10            2           12
## 6 <NA>                6            0            9
```

```
#Using Conditional Statements (IF-ELSE)

#6. Full Search

num <- readline(prompt = "Enter number from 1-50: ")
```

```
## Enter number from 1-50:
```

```
paste("You Entered: ", num)
```

```
## [1] "You Entered:  "
```

```
paste("Your entered number is ", num)
```

```
## [1] "Your entered number is  "
```

```r
if(num == 50) {
  paste("The number you selected is beyond the range of 1 to 50")

} else if (num <= 50){
  paste("TRUE")
} else {
  paste(num)
}
```

```
## [1] "TRUE"
```

```r
#7. Change

bills <- c(50,100,200,500,1000)
bills
```

```
## [1]   50  100  200  500 1000
```

```r
minimum <- function (price) {

  bill <- price%/% 50
  paste("The  minimum number of bills:", bill)
}
snackprice <- 250
minimum(snackprice)
```

```
## [1] "The  minimum number of bills: 5"
```

```r
#8. The following is each student's math score for one semester. Based on this, answer the following qu

#8A. Create a dataframe from the above table. Write the R codes and its output.

mathGrades <- data.frame(
  name = c("Annie","Thea","Steve","Hanna"),
  grade1 = c(85,65,75,95),
  grade2 = c(65,75,55,75),
  grade3 = c(85,90,80,100),
  grade4 = c(100,90,85,90)
)
mathGrades
```

```
##    name grade1 grade2 grade3 grade4
## 1 Annie     85     65     85    100
## 2  Thea     65     75     90     90
## 3 Steve     75     55     80     85
## 4 Hanna     95     75    100     90
```

```r
#8B. Without using the rowMean function, output the average score of students whose average math score
#Example Output: Annie's average grade this semester is 88.75.

mathGrades$average <- (mathGrades$grade1 + mathGrades$grade2 + mathGrades$grade3 + mathGrades$grade4) /

highest <- mathGrades[mathGrades$average >= 90,]
highest
```

```
##    name grade1 grade2 grade3 grade4 average
## 4 Hanna     95     75    100     90      90
```

```r
if (nrow(highest) > 0){
  paste0(highest$name, "'s grade  this semester is: ", highest$average)
} else {
    paste("No students have an average math grade over 90.")
}
```

```
## [1] "Hanna's grade  this semester is: 90"
```

```r
#8D. Without using the mean function, output as follows for the tests in which the average score was le

test1 <- sum(mathGrades$grade1) / nrow(mathGrades)
test1
```

```
## [1] 80
```

```r
test2 <- sum(mathGrades$grade2) / nrow(mathGrades)
test2
```

```
## [1] 67.5
```

```r
test3 <- sum(mathGrades$grade3) / nrow(mathGrades)
test3
```

```
## [1] 88.75
```

```r
test4 <- sum(mathGrades$grade4) / nrow(mathGrades)
test4
```

```
## [1] 91.25
```

```r
if(test1 < 80){
  paste("The first test was so difficult!")
} else if (test2 < 80){
  paste("The second test is even more diffucult than the first one!")
} else if (test3 < 80){
  paste("The third test is even difficult!")
} else if (test4 < 80){
  paste("No test had an average grade less than 80")
}
```

```
## [1] "The second test is even more diffucult than the first one!"
```

```r
#8D. Without using the max function, output as follows for students whose highest score for a semester

  #Annie's Scores

if (mathGrades[1,2] > mathGrades[1,3] && mathGrades[1,2] > mathGrades[1,4] && mathGrades[1,2] > mathGra
  Annie <- mathGrades[1,2]
} else if (mathGrades[1,3] > mathGrades[1,4] && mathGrades[1,3] > mathGrades[1,5]) {
  Annie <- mathGrades[1,3]
} else if (mathGrades[1,4] > mathGrades[1,5] && mathGrades[1,2] > mathGrades[1,5]) {
  Annie <- mathGrades[1,4]
} else {
  Annie <- mathGrades[1,5]
}


  #Thea's Scores
```

```r
if (mathGrades[2,2] > mathGrades[2,3] && mathGrades[2,2] > mathGrades[2,4] && mathGrades[2,2] > mathGrad
  Thea <- mathGrades[2,2]
} else if (mathGrades[2,3] > mathGrades[2,4] && mathGrades[2,3] > mathGrades[2,5]) {
  Thea <- mathGrades[2,3]
} else if (mathGrades[2,4] > mathGrades[2,5] && mathGrades[2,2] > mathGrades[2,5]) {
  Thea <- mathGrades[2,4]
} else {
  Thea <- mathGrades[2,5]
}

  #Steve's Scores

if (mathGrades[3,2] > mathGrades[3,3] && mathGrades[3,2] > mathGrades[3,4] && mathGrades[3,2] > mathGrad
  Steve <- mathGrades[3,2]
} else if (mathGrades[3,3] > mathGrades[3,4] && mathGrades[3,3] > mathGrades[3,5]) {
  Steve <- mathGrades[3,3]
} else if (mathGrades[3,4] > mathGrades[3,5] && mathGrades[3,2] > mathGrades[3,5]) {
  Steve <- mathGrades[3,4]
} else {
  Steve <- mathGrades[3,5]
}

  #Hanna's Scores
if (mathGrades[4,2] > mathGrades[4,3] && mathGrades[4,2] > mathGrades[4,4] && mathGrades[4,2] > mathGrad
  Hanna <- mathGrades[4,2]
} else if (mathGrades[4,3] > mathGrades[4,4] && mathGrades[4,3] > mathGrades[4,5]) {
  Hanna <- mathGrades[4,3]
} else if (mathGrades[4,4] > mathGrades[4,5] && mathGrades[4,2] > mathGrades[4,5]) {
  Hanna <- mathGrades[4,4]
} else {
  Hanna <- mathGrades[4,5]
}

mathGrades$highest <- c(Annie, Thea, Steve, Hanna)


abovegradeof90 <- mathGrades[mathGrades$highest >= 90,]


if (nrow(abovegradeof90) > 0) {
  paste(abovegradeof90$name, "'s highest grade this semester is", abovegradeof90$highest)
} else {
  paste("No students have an average math score over 90.")
}
```

```
## [1] "Annie 's highest grade this semester is 100"
## [2] "Thea 's highest grade this semester is 90"
## [3] "Hanna 's highest grade this semester is 100"
```