

RWorksheet_Quillo#4a

Carl

2023-10-25

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

#1. a

```
dfHouseholdData <- data.frame("Shoe size" =c(6.5,9.0,8.5,8.5,10.5,7.0,9.5,9.0,13.0,7.5,10.5,8.5,12.0,10.5,11.5,13.0,11.5,8.5,5.0,10.0,6.5,7.5,8.5,10.5,8.5,12.0,10.5,11.0,9.0,13.0),
                              "Height" = c(66.0,68.0,64.5,65.0,70.0,64.0,70.0,71.0,72.0,64.0,74.5,67.0,71.0,71.0,77.0,72.0,59.0,62.0,72.0,66.0,64.0,67.0,73.0,69.0,72.0,70.0,69.0,70.0),
                              "Gender" = c("F","F","F","F","M","F","F","F","M","F","M","F","M","M","M","M","F","F","M","F","F","M","M","M","M","M","M","M","M","M","M"))
```

dfHouseholdData

##	Shoe.size	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

*# 1. B.Create a subset by males and females with their corresponding shoe size and height.
What its result? Show the R scripts.*

```
subMaleShoeSize <- subset(dfHouseholdData, Gender == 'M')
```

```
subMaleShoeSize
```

```
##      Shoe.size 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
```

```
subFemaleShoeSize <- subset(dfHouseholdData, Gender == 'F')
subFemaleShoeSize
```

```
##      Shoe.size 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
```

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

```
shoeSizeandHeight <- mean(dfHouseholdData$Shoe.size & dfHouseholdData$Height)
shoeSizeandHeight
```

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

#1.D Is there a relationship between shoe size and height? Why?

if the height is small the shoe size it also small

2 Construct character vector months to a factor with factor() and assign the result to # factor_months_vector. Print out factor_months_vector and assert that R prints out # the factor levels below the actual values.

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

```
## [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"
```

```
factor_months_vector <- factor(months_vector)
factor_months_vector
```

```
## [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
```

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

```
sumofmonths <- summary(months_vector)
sumofmonths
```

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

```
sumoffacmonths <- summary(factor_months_vector)
sumoffacmonths
```

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

4 Create a vector and factor

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

```
direction_factor <- factor(direction_vector, levels = c("East", "West", "North"))
```

```
print(direction_factor)
```

```
## [1] East West West West North North North
## Levels: East West North
```

#5

```
excelData<- read.csv("import_match.csv")
```

*#6a Create an R Program that allows the User to randomly select numbers from 1 to 50.
Then display the chosen number. If the number is beyond the range of the selected choice,
it will have to display a string "The number selected is beyond the range of 1 to 50". If
number 20 is inputted by the User, it will have to display "TRUE", otherwise display the input number.*

```
inputnum <- readline(paste("Eneter a number from 1 to 50"))
```

```
## Eneter a number from 1 to 50
```

```
if(inputnum > 50){
  paste("You Enetered ", inputnum)
```

```

}else{
  paste("The number selected is beyond the range of 1 to 50")
}

```

```
## [1] "The number selected is beyond the range of 1 to 50"
```

7 Write a function that prints the minimum number of bills that must be paid, given the price of the snack. Input: Price of snack (a random number divisible by 50) Output: Minimum number of bills needed to purchase the snack

```
priceofsnack <- as.numeric(readline(paste("Enter the price of the snacks: ")))
```

```
## Enter the price of the snacks:
```

```

numofbills <- function(priceofsnack){

  minBills <- priceofsnack %/% 50
  paste("The minimum number of bills is:", minBills)

}

```

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

```

dfstudents <- data.frame("Name" = c("Annie", "Thea", "Steve", "Hanna"),
  "Grade 1" = c(85, 65, 75, 95),
  "Grade 2" = c(65, 75, 55, 75),
  "Grade 3" = c(85, 90, 80, 100),
  "Grade 4" = c(100, 90, 85, 90))

```

```
dfstudents
```

```

##      Name Grade.1 Grade.2 Grade.3 Grade.4
## 1 Annie      85      65      85      100
## 2 Thea       65      75      90      90
## 3 Steve      75      55      80      85
## 4 Hanna      95      75     100      90

```

#8b Without using the rowMean function, output the average score of students whose average math score over the semester is greater than or equal to 90

```

test_averages <- rowSums(dfstudents[, -1]) / ncol(dfstudents[, -1])
test_averages

```

```
## [1] 83.75 80.00 73.75 90.00
```

```
high_scorers <- dfstudents[test_averages >= 90, ]
```

```

if(nrow(high_scorers) > 0){
  paste(dfstudents$Name, "'s average grade this semester is", test_averages)
}else{
  paste("No student has an average math score over 90 points during the semester")
}

```

```

## [1] "Annie 's average grade this semester is 83.75"
## [2] "Thea 's average grade this semester is 80"
## [3] "Steve 's average grade this semester is 73.75"
## [4] "Hanna 's average grade this semester is 90"

```

#8c Without using the mean function, output as follows for the tests in which the average score was less than 80

```
avegrade1 <- sum(dfstudents$Grade.1) /4
```

```

avegrade2 <- sum(dfstudents$Grade.2) /4

avegrade3 <- sum(dfstudents$Grade.3) /4

avegrade4 <- sum(dfstudents$Grade.4) /4

test_averages <- c(avegrade1, avegrade2, avegrade3, avegrade4)

difficult_tests <- which(test_averages < 80)

if (length(difficult_tests) > 0) {
  cat(paste("The grade", paste(difficult_tests, collapse = ", "), "test(s) were difficult.\n"))
} else {
  cat("No test was difficult.\n")
}

## The grade 2 test(s) were difficult.
#8d Without using the max function, output as follows for students whose highest score for a semester e

df1strow <- dfstudents[1,2:5]
df2ndrow <- dfstudents[2,2:5]
df3rdrow <- dfstudents[3,2:5]
df4throw <- dfstudents[4,2:5]

if(df1strow[1] > 90){
  paste("Annie highest grade this semester is",df1strow[1],"")
}else if(df1strow[2] >90){
  paste("Annie highest grade this semester is",df1strow[2],"")
}else if(df1strow[3] >90){
  paste("Annie highest grade this semester is",df1strow[3],"")
}else if(df1strow[4] >90){
  paste("Annie highest grade this semester is",df1strow[4],"")
}else{
  paste("Annie has no grade that is higher than 90")
}

## [1] "Annie highest grade this semester is 100 "

if(df2ndrow[1] > 90){
  paste("Thea highest grade this semester is",df2ndrow[1],"")
}else if(df2ndrow[2] >90){
  paste("Thea highest grade this semester is",df2ndrow[2],"")
}else if(df2ndrow[3] >90){
  paste("Thea highest grade this semester is",df2ndrow[3],"")
}else if(df2ndrow[4] >90){
  paste("Thea highest grade this semester is",df2ndrow[4],"")
}else{
  paste("Thea has no grade that is higher than 90")
}

## [1] "Thea has no grade that is higher than 90"

if(df3rdrow[1] > 90){
  paste("Steve highest grade this semester is",df3rdrow[1],"")
}else if(df3rdrow[2] >90){

```

```

    paste("Steve highest grade this semester is",df3rdrow[2],"")
  }else if(df3rdrow[3] >90){
    paste("Steve highest grade this semester is",df3rdrow[3],"")
  }else if(df3rdrow[4] >90){
    paste("Steve highest grade this semester is",df1strow[4],"")
  }else{
    paste("Steve has no grade that is higher than 90")
  }

```

```
## [1] "Steve has no grade that is higher than 90"
```

```

if(df4throw[1] > 90){
  paste("Hanna highest grade this semester is",df4throw[3],"")
}else if(df4throw[4] >90){
  paste("Hanna highest grade this semester is",df4throw[2],"")
}else if(df4throw[3] >90){
  paste("Hanna highest grade this semester is",df4throw[3],"")
}else if(df4throw[4] >90){
  paste("Hanna highest grade this semester is",df4throw[4],"")
}else{
  paste("Hanna has no grade that is higher than 90")
}

```

```
## [1] "Hanna highest grade this semester is 100 "
```