RWorksheet_Cautivar#4a.Rmd

James Clark Cautivar

2024-10-16

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

```
RespondentsData <- 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, 13.0, 11.5, 8
  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, 71
  Gender = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F", "
)
RespondentsData
```

```
##    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
```

   a. Describe the data. The data shows informations about respondent's shoe size, height, and their gender.

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

```
Males <- subset(RespondentsData, Gender == "M", select = c(Gender, Shoe_Size, Height))
Males
```

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

```
Females <- subset(RespondentsData, Gender == "F", select = c(Gender, Shoe_Size, Height))
Females
```

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

c. Find the mean of shoe size and height of the respondents. Write the R scripts and its result.

```
AllShoeSize <- RespondentsData$Shoe_Size
mean(AllShoeSize)
```

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

```
AllHeight <- RespondentsData$Height
mean(AllHeight)
```

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

d. Is there a relationship between shoe size and height? Why? I would say that there is a relationship between shoe size and height. Because taller people tend to have bigger shoe size while shorter people have smaller shoe size.

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","N
"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?

```
summary(months_vector)
```

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

```
summary(factor_months_vector)
```

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

Personally, i think that they're both useful because they show different results which are helpful if you need a specific information about a vector.

4. Create a vector and factor for the table below.

```
Directions <- c("East", "West", "North")
Frequency <- c(1, 4, 3)

new_order_data <- factor(Directions,levels = c("East","West","North"))
print(new_order_data)
```

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

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

a. Import the excel file into the Environment Pane using read.table() function. Write the code.

```
file <- read.table("import_march.csv", header = TRUE, sep = ",")
```

b. View the dataset. Write the R scripts and its result.

```
print(file)
```

```
##   Students Strategy.1 Strategy.2 Strategy3
## 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
```

6. Full Search

a. 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.

```
num <- readline(prompt="Enter a number: ")
```

```
## Enter a number:
```

```
print(num)
```

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

```
if(num < 1 || num > 50) {
  print("The number selected is beyond the range of 1 to 50")
} else if(num == 20) {
  print("TRUE")
  }else{
    print(num)
  }
```

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

7. Change

a. Write a function that prints the minimum number of bills that must be paid, given the price of the snack.

```
min_bills <- function(price) {

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

  num_bills <- 0

  for (bill in bills) {

    num_bills <- num_bills + price %/% bill

    price <- price %% bill
  }

  return(num_bills)
}

price<- as.numeric(readline(prompt="Price of snack, a random number divisible by 50: "))
```

```
## Price of snack, a random number divisible by 50:
```

```
price <- as.numeric(price)
print(paste("Minimum number of bills needed:", min_bills(price)))
```

```
## [1] "Minimum number of bills needed: NA"
```

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

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

```
studData <- data.frame(
  Name = c("Annie", "Thea", "Steve", "Hanma"),
```

```
  Grade1 = c(85, 65, 75, 95),
  Grade2 = c(65, 75 , 55, 75),
  Grade3 = c(85, 90, 80, 100),
  Grade4 = c(100, 90, 85, 90)
)
print(studData)
```

```
##    Name Grade1 Grade2 Grade3 Grade4
## 1 Annie     85     65     85    100
## 2  Thea     65     75     90     90
## 3 Steve     75     55     80     85
## 4 Hanma     95     75    100     90
```

b. Without using the rowMean function, output the average score of students whose average math score over 90 points during the semester. write R code and its output.

```
for (i in 1:nrow(studData)) {
  avg <- sum(studData[i, 2:5]) / 4
  if (avg > 90) {
    print(paste(studData$Name[i], "'s average grade this semester is", round(avg, 2)))
  }
}
```

c. Without using the mean function, output as follows for the tests in which the average score was less than 80 out of 4 tests.

```
tests <- colnames(studData)[2:5]

for (j in 2:5) {
  test_avg <- sum(studData[, j]) / nrow(studData)
  if (test_avg < 80) {
    print(paste(tests[j - 1], "test was difficult."))
  }
}
```

```
## [1] "Grade2 test was difficult."
```

d. Without using the max function, output as follows for students whose highest score for a semester exceeds 90 points.

```
for (i in 1:nrow(studData)) {
  highest_score <- studData[i, 2]
  for (j in 3:5) {
    if (studData[i, j] > highest_score) {
      highest_score <- studData[i, j]
    }
  }
  if (highest_score > 90) {
    print(paste(studData$Name[i], "'s highest grade this semester is", highest_score))
  }
}
```

```
## [1] "Annie 's highest grade this semester is 100"
## [1] "Hanma 's highest grade this semester is 100"
```