RWorksheet_Infiesto#4a

Infiesto

2024-10-22

1.

```
#1. The table below shows the data about shoe size and height.
#a. Describe the data.
#The table shows information about people's shoe size, height, and gender (M for male, F for female). I

#b. Create a subset by males and females with their corresponding shoe size and height. What its result
# Create the data frame
household_data <- data.frame(
  Shoe_Size = c(6.5, 9.0, 8.5, 8.5, 10.5, 7.0, 9.5, 9.0, 13.0, 10.5, 10.5, 8.5, 10.5, 12.0, 10.5, 13.0,
  Height = c(66.0, 68.0, 64.5, 65.0, 70.0, 64.0, 70.0, 72.0, 72.0, 74.5, 67.0, 71.0, 71.0, 67.0, 71.0, 
  Gender = c('F', 'F', 'F', 'M', 'M', 'F', 'F', 'F', 'M', 'M', 'F', 'F', 'M', 'F', 'M', 'M', 'M', 'F',
)
md <- subset(household_data, Gender == "M")
fd <- subset(household_data, Gender == "F")
md
```

```
##    Shoe_Size Height Gender
## 4        8.5   65.0      M
## 5       10.5   70.0      M
## 9       13.0   72.0      M
## 10      10.5   74.5      M
## 13      10.5   71.0      M
## 15      10.5   71.0      M
## 16      13.0   77.0      M
## 17      11.5   72.0      M
## 20      10.0   72.0      M
## 24      10.5   73.0      M
## 25      10.5   72.0      M
## 26       9.0   69.0      M
## 27      13.0   71.0      M
## 29       9.0   69.0      M
## 30      13.0   70.0      M
```

```
fd
```

```
##    Shoe_Size Height Gender
## 1        6.5   66.0      F
## 2        9.0   68.0      F
## 3        8.5   64.5      F
## 6        7.0   64.0      F
## 7        9.5   70.0      F
## 8        9.0   72.0      F
## 11      10.5   67.0      F
## 12       8.5   71.0      F
```

```
## 14      12.0   67.0      F
## 18       8.5   59.0      F
## 19       5.0   62.0      F
## 21       6.5   66.0      F
## 22       7.5   64.0      F
## 23       8.5   67.0      F
## 28      11.0   69.0      F
```

```
#c. Find the mean of shoe size and height of the respondents. Write the R scripts and its result.
mean_shoe_size <- mean(household_data$Shoe_Size)
mean_height <- mean(household_data$Height)
mean_shoe_size
```

```
## [1] 9.683333
```

```
mean_height
```

```
## [1] 68.83333
```

```
#d. Is there a relationship between shoe size and height? Why?
correlation <- cor(household_data$Shoe_Size, household_data$Height)
correlation
```

```
## [1] 0.6790149
```

2.

```
#2. Construct character vector months to a factor with factor() and assign the result to factor_months_
# Create a character vector of months
months <- c("March", "April", "January", "November", "January", "September", "October", "September", "N
factor_months_vector <- factor(months)
print(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
```

```
levels(factor_months_vector)
```

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

```
#3. Then check the summary() of the months_vector and factor_months_vector. | Inter-pret the results of
summary(months)
```

```
##    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
```

4.

```
#4. Create a vector and factor for the table below.
factor_data <- rep(c("East", "West", "North"), c(1, 4, 3))
```

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

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

5.

```r
#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.
data <- read.table("import_march.csv", header = TRUE, sep = ",",na.strings = "")
print(data)
```

```
##   Students Strategy.1 Strategy.2 Strategy.3
## 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
```

6.

```r
#6. Full Search

#Exhaustive search is a methodology for finding an answer by exploring all possible cases. When trying

#a. Create an R Program that allows the User to randomly select numbers from 1 to 50. Then display the

a <- function() {
  num <- as.integer(readline(prompt = "Enter a number between 1 and 50: "))

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

a()
```

```
## Enter a number between 1 and 50:
## [1] "Invalid"
```

7.

```r
#7. Change

#At ISATU University's traditional cafeteria, snacks can only be purchased with bills. A long-standing

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

```r
min_bills <- function(price) {
  bills <- c(1000, 500, 200, 100, 50)
  count <- 0

  if (price %% 50 != 0) return("The price must be a multiple of 50.")

  for (bill in bills) {
    count <- count + price %/% bill
    price <- price %% bill
  }

  return(paste("Minimum number of bills needed:", count))
}

min_bills(2700)
```

```
## [1] "Minimum number of bills needed: 4"
```

8.

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

#a. Create a dataframe from the above table. Write the R codes and its output.
students <- 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)
)
print(students)
```

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

students$Average <- rowMeans(students[2:5])
above90 <- students[students$Average > 90, "Average"]

if (length(above90) > 0) {
  cat(students$Name[students$Average > 90], "'s average grade this semester is", round(above90, 2), "\n
}


#c. Without using the mean function, output as follows for the tests in which the average score was les
#Example output: The nth test was difficult.

average_scores <- colMeans(students[2:5])

difficult_tests <- which(average_scores < 80)
```

```r
if (length(difficult_tests) > 0) {
  cat("The", difficult_tests, "tests were difficult.\n")
}
```

```
## The 2 tests were difficult.
```

```r
#d. Without using the max function, output as follows for students whose highest score for a semester e
#Example Output: Annie's highest grade this semester is 95.

for (i in 1:nrow(students)) {
  highest_score <- max(students[i, 2:5])
  if (highest_score > 90) {
    cat(students$Name[i], "'s highest grade this semester is", highest_score, "\n")
  }
}
```

```
## Annie 's highest grade this semester is 100
## Hanna 's highest grade this semester is 100
```