RWorksheet_Lumauag#4a

Matt Andrei Lumauag

2024-10-15

```
#1
#a
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, 71
gender <- c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "F", "F", "M
householdData <- data.frame(shoesize, height, gender)
names(householdData) <- c("Shoesize", "Height", "Gender")
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
```

```
#b
householdDataMale <- subset(householdData, Gender == "M")
householdDataMale
```

```
##    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
```

```
householdDataFemale <- subset(householdData, Gender == "F")
householdDataFemale
```

```
##    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
```

```
#c
mean(shoesize)
```

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

```
mean(height)
```

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

```
#d
#The relationship could exist as larger feet are often associated with taller individuals.
```

```
#2
#a
Months <- c("March","April","January","November","January",
"September","October","September","November","August",
```

```
"January","November","November","February","May","August","July","December","August","August","Septembe

Factor_Months <- factor(Months)
Factor_Months
```

```
##  [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
summary(Months)
```

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

```
summary(Factor_Months)
```

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

```
#Factor_Months is more useful, especially for analysis that involve categorical data, as it allows to t
```

```
#4
direction <- c("East", "West", "North")
frequency <- c(1, 4, 3)

neworderdata <- factor(direction,levels = c("East","West","North"))
neworderdata
```

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

```
#5
options(repos = c(CRAN = "https://cloud.r-project.org"))
install.packages("readxl")
```

```
## package 'readxl' successfully unpacked and MD5 sums checked
```

```
## Warning: cannot remove prior installation of package 'readxl'
```

```
## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying
## C:\Users\Matteeuu\AppData\Local\Programs\R\R-4.4.1\library\00LOCK\readxl\libs\x64\readxl.dll
## to
## C:\Users\Matteeuu\AppData\Local\Programs\R\R-4.4.1\library\readxl\libs\x64\readxl.dll:
## Permission denied
```

```
## Warning: restored 'readxl'
```

```
##
## The downloaded binary packages are in
##  C:\Users\Matteeuu\AppData\Local\Temp\RtmpAfqClI\downloaded_packages
```

```
data <- read.table("ExcelData.csv", header = TRUE, sep = ",", stringsAsFactors =  FALSE)
data
```

```
##   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
```

```
#6
user_input <- readline(prompt = "Please enter a number between 1 and 50: ")
```

```
## Please enter a number between 1 and 50:
```

```
paste("The number you selected is:", user_input)
```

```
## [1] "The number you selected is: "
```

```
  if(is.na(user_input) < 1 && user_input > 50) {
    cat("The number selected is beyond the range of 1 to 50\n")
  } else if (user_input == 20) {
    cat("TRUE\n")
  } else {
    cat("You selected: ", user_input, "\n")
  }
```

```
## You selected:
```

```
#7
min_bills <- function(price) {
  bills <- c(1000, 500, 200, 100, 50)


  count <- 0

  for (bill in bills) {
    if (price <= 0) {
      break
    }
    count <- count + floor(price / bill)

    price <- price %% bill
```

```
  }

  return(count)
}


snack_price <- as.integer(readline(prompt = "Enter the price of the snack (divisible by 50): "))
```

```
## Enter the price of the snack (divisible by 50):
```

```
if (is.na(snack_price) %% 50 == 0) {
  cat("Minimum number of bills needed:", min_bills(snack_price), "\n")
} else {
  cat("The price must be divisible by 50.\n")
}
```

```
## The price must be divisible by 50.
```

```
#8
#a
Grades <- 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)
                      )
Grades
```

```
##    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
```

```
#b
for (i in 1:nrow(Grades)) {
  avg_score <- sum(Grades[i, 2:5]) / 4
  if (avg_score > 90) {
    cat(Grades$Name[i], "'s average grade this semester is", avg_score, "\n")
  }
}
```

```
#c
for (j in 2:ncol(Grades)) {
  test_avg <- sum(Grades[, j]) / nrow(Grades)
  if (test_avg < 80) {
    cat("The", colnames(Grades)[j], "test was difficult.\n")
  }
}
```

```
## The Grade_2 test was difficult.
```

```r
#d
for (i in 1:nrow(Grades)) {
  highest_grade <- Grades[i, 2]
  for (j in 3:5) {
    if (Grades[i, j] > highest_grade) {
      highest_grade <- Grades[i, j]
    }
  }
  if (highest_grade > 90) {
    cat(Grades$Name[i], "'s highest grade this semester is", highest_grade, "\n")
  }
}
```

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