RWorksheet_Lapso#4a.Rmd

Darlene Erl Lapso

2023-10-25

R Markdown

```
Size_height <- data.frame(
  Shoe_Size_1 = c(6.5, 9, 8.5, 8.5, 10.5, 7, 9.5, 9, 13, 7.5, 10.5, 8.5, 12, 10.5),
  Height_1 = c(66, 68, 64.5, 65, 70, 64, 70, 71, 72, 64, 74.5, 67, 71, 71),
  Gender_1 = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M"),
  Shoe_Size_2 = c(13, 11.5, 8.5, 5, 10, 6.5, 7.5, 8.5, 10.5, 8.5, 10.5, 11, 9, 13),
  Height_2 = c(77, 7, 59, 62, 72, 66, 64, 67, 73, 69, 72, 70, 69, 70),
  Gender_2 = c("M", "M", "F", "F", "M", "F", "F", "M", "M", "F", "M", "M", "M", "M")
)

Size_height
```

```
##    Shoe_Size_1 Height_1 Gender_1 Shoe_Size_2 Height_2 Gender_2
## 1          6.5     66.0        F        13.0       77        M
## 2          9.0     68.0        F        11.5        7        M
## 3          8.5     64.5        F         8.5       59        F
## 4          8.5     65.0        F         5.0       62        F
## 5         10.5     70.0        M        10.0       72        M
## 6          7.0     64.0        F         6.5       66        F
## 7          9.5     70.0        F         7.5       64        F
## 8          9.0     71.0        F         8.5       67        M
## 9         13.0     72.0        M        10.5       73        M
## 10         7.5     64.0        F         8.5       69        F
## 11        10.5     74.5        M        10.5       72        M
## 12         8.5     67.0        F        11.0       70        M
## 13        12.0     71.0        M         9.0       69        M
## 14        10.5     71.0        M        13.0       70        M
```

```
males_subset <- subset(Size_height, Gender_1 == "M" | Gender_2 ==
                         "M", select = c(Shoe_Size_1, Height_1, Shoe_Size_2, Height_2))

print("Males Subset:")
```

```
## [1] "Males Subset:"
```

```
print(males_subset)
```

```
##    Shoe_Size_1 Height_1 Shoe_Size_2 Height_2
## 1          6.5     66.0        13.0       77
## 2          9.0     68.0        11.5        7
## 5         10.5     70.0        10.0       72
## 8          9.0     71.0         8.5       67
## 9         13.0     72.0        10.5       73
```

```
## 11        10.5     74.5        10.5       72
## 12         8.5     67.0        11.0       70
## 13        12.0     71.0         9.0       69
## 14        10.5     71.0        13.0       70
```

```
females_subset <- subset(Size_height, Gender_1 == "F" | Gender_2 ==
                           "F", select = c(Shoe_Size_1, Height_1, Shoe_Size_2, Height_2))

print("Females Subset:")
```

```
## [1] "Females Subset:"
```

```
print(females_subset)
```

```
##    Shoe_Size_1 Height_1 Shoe_Size_2 Height_2
## 1          6.5     66.0        13.0       77
## 2          9.0     68.0        11.5        7
## 3          8.5     64.5         8.5       59
## 4          8.5     65.0         5.0       62
## 6          7.0     64.0         6.5       66
## 7          9.5     70.0         7.5       64
## 8          9.0     71.0         8.5       67
## 10         7.5     64.0         8.5       69
## 12         8.5     67.0        11.0       70
```

```
mean_shoe_size <- mean(c(Size_height$Shoe_Size_1, Size_height$Shoe_Size_2))
mean_height <- mean(c(Size_height$Height_1, Size_height$Height_2))

mean_shoe_size
```

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

```
mean_height
```

```
## [1] 66.25
```

```
cat("Mean Shoe Size:", mean_shoe_size, "\n")
```

```
## Mean Shoe Size: 9.410714
```

```
cat("Mean Height:", mean_height, "\n")
```

```
## Mean Height: 66.25
```

```
#Yes, as the shoe sizes increases, the height increases
```

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

"January","November","November","February","May","August", "July","December","August","August",
"September","November","February",
"April")

factor_months <- factor(months_vector)
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
```

```
assign("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
```

```
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
```

```
#Yes of course, so it is easy accessible the data basic information of the table given such as;
#its length, class, mode, and how many are the each elements.
```

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

factor_data <- factor(direction_vector)

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
```

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

```
#the 'read' function isn't working on this R script.
```

```
#number <- 1:50
#chosen <- as.integer(readline(prompt = "Enter a number 1-50: "))
#chosen


#if (chosen >= 1 && chosen <= 50) {
#  cat("TRUE\n")
```

```
#} else{
#  cat("The number you entered", chosen, "is beyond the expected.")
#}

#result

## > chosen <- as.integer(readline(prompt = "Enter a number 1-50: "))
##Enter a number 1-50: 50
##> chosen <- as.integer(readline(prompt = "Enter a number 1-50: "))
##Enter a number 1-50: 50
##> sen >= 1 && chosen <= 50) {
##Error: unexpected ')' in "sen >= 1 && chosen <= 50)"
##> if (chosen >= 1 && chosen <= 50) {
##+   cat("TRUE\n")
##+ } else{
##+   cat("The number you entered", chosen, "is beyond the ##expected.")
##+ }
##TRUE

#>>> I made this code as a comment since it cant be knitted
#>> when using a readline.
```

```
minimum_bills <- function(price) {
  bill_denominations <- c(1000, 500, 200, 100, 50)
  total_bills <- 0

  if (price %% 50 == 0) {
    for (bill in bill_denominations) {
      bill_count <- price %/% bill
      total_bills <- total_bills + bill_count
      price <- price %% bill
    }

    cat("Minimum number of bills needed: ", total_bills, "\n")
  } else {
    cat("Price must be divisible by 50\n")
  }
}

price_of_snack <- 1350
minimum_bills(price_of_snack)
```

```
## Minimum number of bills needed:  4
```

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

  num_bills <- 0


  for (bill in bills) {
    while (price >= bill) {
      price <- price - bill
```

```
      num_bills <- num_bills + 1
    }
  }

  return(num_bills)
}

price <- sample(1000:5000, size = 1)

num_bills <- min_bills(price)

print(num_bills)
```

```
## [1] 6
```

```
data <- 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)
)
data$AvgMathScore <- (data$Grade1 + data$Grade2 + data$Grade3 + data$Grade4) / 4

students_over_90 <- data[data$AvgMathScore > 90, ]
students_over_90
```

```
## [1] NAME         Grade1       Grade2       Grade3       Grade4
## [6] AvgMathScore
## <0 rows> (or 0-length row.names)
```

```
data <- 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)
)
data$AvgMathScore <- (data$Grade1 + data$Grade2 + data$Grade3 + data$Grade4) / 4

#ANNIE

Annie_Ave <- data[data$NAME == "Annie", "AvgMathScore"]

Annie_Ave
```

```
## [1] 83.75
```

```
#THEA

Thea_Ave <- data[data$NAME == "Thea", "AvgMathScore"]

Thea_Ave
```

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

```r
#STEVE

Steve_Ave <- data[data$NAME == "Steve", "AvgMathScore"]

Steve_Ave
```

```
## [1] 73.75
```

```r
#HANNA

Hanna_Ave <- data[data$NAME == "Hanna", "AvgMathScore"]

Hanna_Ave
```

```
## [1] 90
```

```r
data <- 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)
)

data$AvgScore <- (data$Grade1 + data$Grade2 + data$Grade3 + data$Grade4) / 4

students_below_80 <- data[data$AvgScore < 80, ]
students_below_80
```

```
##    NAME Grade1 Grade2 Grade3 Grade4 AvgScore
## 3 Steve     75     55     80     85    73.75
```

```r
students <- c("Annie", "Thea", "Steve", "Hanna")

for (student in students) {
  max_score <- max(data[data$NAME == student, 2:5])
  cat(student, "Max Score:", max_score, "\n")
}
```

```
## Annie Max Score: 100
## Thea Max Score: 90
## Steve Max Score: 85
## Hanna Max Score: 100
```