

# RWorksheet\_calvario#4a.Rmd

Jolien

2024-11-06

1.

```
sshframe <- 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,  
  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,  
  Gender = c("F", "F", "F", "F", "M", "F", "F", "F", "M", "F", "M", "F", "M", "M", "M", "M", "M", "F", "F", "  
)  
sshframe
```

##	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.

The data contains two sets of observations for shoe size, height, and gender.

b.

```
males <- sshframe[sshframe$Gender == "M", c("Shoe_size", "Height")]
females <- sshframe[sshframe$Gender == "F", c("Shoe_size", "Height")]
```

males

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

females

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

c.

```
mean_shoe_size <- mean(sshframe$Shoe_size)
mean_height <- mean(sshframe$Height)
```

mean\_shoe\_size

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

```
mean_height
```

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

d.

```
correlation <- cor(sshframe$Shoe_size, sshframe$Height)
correlation
```

```
## [1] 0.7766089
```

2.

```
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")
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.

```
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
```

4.

```
directions_vector <- c("East", "West", "North")
frequencies_vector <- c(1, 4, 3)

factor_data <- factor(directions_vector)

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

new_order_data

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

5.

a.

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

b.

```
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.

a.

```
user_input =(readline(prompt = "Enter a random number from 1 to 50: "))

## Enter a random number from 1 to 50:
if(user_input == 20){
  print(TRUE)
}else if(user_input >= 1 && user_input <= 50){
  print(user_input)
}else{
  print("The number selected is beyond the range of 1 to 50")
}
```

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

7.

```
minBills <- function(price) {  
  bills <- c(1000, 500, 200, 100, 50)  
  count <- 0  
  for (bill in bills) {  
    while (price >= bill) {  
      price <- price - bill  
      count <- count + 1  
    }  
  }  
  return(count)  
}  
snack_price <- 300  
cat("Minimum number of bills needed:", minBills(snack_price), "\n")
```

```
## Minimum number of bills needed: 2
```

8.

a.

```
students_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)  
)  
  
print(students_data)
```

```
##      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
```

b.

```
calculate_average <- function(grades) {  
  total <- sum(grades)  
  avg <- total / length(grades)  
  return(avg)  
}  
  
for (i in 1:nrow(students_data)) {  
  grades <- as.numeric(students_data[i, 2:5])  
  avg_grade <- calculate_average(grades)  
  
  if (avg_grade > 90) {
```

```

    cat(students_data$Name[i], "'s average grade this semester is ", avg_grade, "\n", sep = "")
  }
}

```

c.

```

for (j in 2:ncol(students_data)) {
  total <- sum(students_data[, j])
  avg_test <- total / nrow(students_data)

  if (avg_test < 80) {
    cat("The ", j - 1, "nd test was difficult.\n", sep = "")
  }
}

```

```
## The 2nd test was difficult.
```

d.

```

calculate_max <- function(grades) {
  max_grade <- grades[1]
  for (grade in grades) {
    if (grade > max_grade) {
      max_grade <- grade
    }
  }
  return(max_grade)
}

for (i in 1:nrow(students_data)) {
  grades <- as.numeric(students_data[i, 2:5])
  highest_grade <- calculate_max(grades)

  if (highest_grade > 90) {
    cat(students_data$Name[i], "'s highest grade this semester is ", highest_grade, "\n", sep = "")
  }
}

```

```

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

```