

RWorksheet_Redosendo#4a

Kyle Edward B. Redosendo

2025-11-16

#1. Create a data frame.

```
shoe_data <- data.frame(  
  ShoeSize = c(6.5, 9.0, 8.5, 8.5, 10.5, 7.0, 9.5, 7.0, 7.5, 7.5, 8.5, 10.5,  
  13.0, 11.5, 8.5, 5.0, 10.0, 6.5, 7.5, 10.5, 8.0, 11.0, 9.0, 13.0),  
  Height = c(66.0, 68.0, 64.5, 65.0, 70.0, 70.0, 71.0, 72.0, 64.0, 64.0, 67.0, 71.0,  
  77.0, 72.0, 59.0, 60.0, 72.0, 66.0, 64.0, 69.0, 67.0, 70.0, 69.0, 70.0),  
  Gender = c('F', 'F', 'F', 'F', 'M', 'F', 'F', 'F', 'F', 'F', 'F', 'M',  
  'M', 'M', 'F', 'F', 'M', 'F', 'F', 'M', 'F', 'M', 'M', 'M')  
)  
shoe_data
```

##	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	70.0	F
## 7	9.5	71.0	F
## 8	7.0	72.0	F
## 9	7.5	64.0	F
## 10	7.5	64.0	F
## 11	8.5	67.0	F
## 12	10.5	71.0	M
## 13	13.0	77.0	M
## 14	11.5	72.0	M
## 15	8.5	59.0	F
## 16	5.0	60.0	F
## 17	10.0	72.0	M
## 18	6.5	66.0	F
## 19	7.5	64.0	F
## 20	10.5	69.0	M
## 21	8.0	67.0	F
## 22	11.0	70.0	M
## 23	9.0	69.0	M
## 24	13.0	70.0	M

#a. Describe the data.

```
summary(shoe_data)
```

##	ShoeSize	Height	Gender
## Min.	: 5.000	Min. :59.00	Length:24

```
## 1st Qu.: 7.500    1st Qu.:64.88    Class :character
## Median : 8.500    Median :68.50    Mode  :character
## Mean   : 8.917    Mean   :67.81
## 3rd Qu.:10.500    3rd Qu.:70.25
## Max.   :13.000    Max.    :77.00
```

#b. Create a subset by males and females with their corresponding shoe size and height. What is the result?

#females

```
female_data <- subset(shoe_data, Gender == "F", select = c(ShoeSize, Height))
female_data
```

```
##      ShoeSize Height
## 1         6.5   66.0
## 2         9.0   68.0
## 3         8.5   64.5
## 4         8.5   65.0
## 6         7.0   70.0
## 7         9.5   71.0
## 8         7.0   72.0
## 9         7.5   64.0
## 10        7.5   64.0
## 11        8.5   67.0
## 15        8.5   59.0
## 16         5.0   60.0
## 18        6.5   66.0
## 19        7.5   64.0
## 21        8.0   67.0
```

#males

```
male_data <- subset(shoe_data, Gender == "M", select = c(ShoeSize, Height))
male_data
```

```
##      ShoeSize Height
## 5         10.5    70
## 12        10.5    71
## 13        13.0    77
## 14        11.5    72
## 17        10.0    72
## 20        10.5    69
## 22        11.0    70
## 23         9.0    69
## 24        13.0    70
```

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

Mean of respondents

```
mean_shoe <- mean(shoe_data$ShoeSize)
mean_height <- mean(shoe_data$Height)
```

```
mean_shoe
```

```
## [1] 8.916667
```

```
mean_height
```

```
## [1] 67.8125
```

#d. Is there a relationship between shoe size and height? Why?

Correlation test

```
correlation <- cor(shoe_data$ShoeSize, shoe_data$Height)
correlation
```

```
## [1] 0.6723337
```

#2. Construct character vector months to a factor with factor() and assign the result to factor_months_

```
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")
```

Convert to factor

```
factor_months_vector <- factor(months_vector)
```

Print the factor 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 o

Summary of character vector

```
summary(months_vector)
```

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

Summary of factor vector

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

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

```
frequency <- c(1, 4, 3)
```

Create factor with specific order

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

Print the ordered factor

```
print(factor_data)
```

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

```
## Levels: East West North
```

Combine into a data frame

```
table_data <- data.frame(Direction = factor_data, Frequency = frequency)
```

```
print(table_data)
```

```
## Direction Frequency
## 1      East      1
## 2      West      4
## 3      North     3
```

```
#5. Enter the data below in Excel with file name = import_march.csv
#data <- read.table("import_march.csv", header = TRUE, sep = ",")
#data
```

```
#6.Full Search
```

```
if (interactive()) {
mode <- tolower(trimws(readline("Enter mode ('r' for random, 'm' for manual): ")))
} else {
mode <- "r"
}
```

```
if (mode == "r") {
chosen <- sample(1:50, 1)
cat("Randomly chosen number:", chosen, "\n")
} else if (mode == "m") {
if (interactive()) {
input <- readline("Enter an integer: ")
chosen_num <- suppressWarnings(as.integer(input))
if (is.na(chosen_num)) stop("Invalid input: please enter an integer.")
chosen <- chosen_num
} else {
chosen <- 20
cat("Default number selected for knitting:", chosen, "\n")
}
} else {
stop("Invalid mode. Use 'r' or 'm'.")
}
```

```
## Randomly chosen number: 25
```

```
if (chosen < 1 || chosen > 50) {
cat("The number selected is beyond the range of 1 to 50\n")
} else if (chosen == 20) {
cat("TRUE\n")
} else {
cat("Selected number:", chosen, "\n")
}
```

```
## Selected number: 25
```

```
#7. Change
```

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

```
remaining <- price
```

```
for (b in bills) {
if (remaining >= b) {
```

```

n <- remaining %% b
count <- count + n
remaining <- remaining - n*b
}
}

```

```

return(count)
}

```

```

# RUN PROGRAM

```

```

# generate random price divisible by 50
price <- sample(seq(50, 5000, by = 50), 1)
cat("Price of snack:", price, "\n")

```

```

## Price of snack: 4450

```

```

cat("Minimum bills needed:", min_bills(price), "\n")

```

```

## Minimum bills needed: 7

```

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

```

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)

```

```

df <- data.frame(Name, Grade1, Grade2, Grade3, Grade4)
df

```

```

##      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

```

```

cat("\n--- b. average > 90 ---\n")

```

```

##

```

```

## --- b. average > 90 ---

```

```

# b. without rowMeans

```

```

avg <- (df$Grade1 + df$Grade2 + df$Grade3 + df$Grade4) / 4

```

```

for(i in 1:nrow(df)){
  if(avg[i] > 90){
    cat(df$Name[i], "'s average grade this semester is ", avg[i], ".\n", sep="")
  }
}

```

```

cat("\n--- c. test average < 80 ---\n")

```

```

##

```

```

## --- c. test average < 80 ---

```

```

# c. without mean()
testAvg <- c(
  sum(df$Grade1)/4,
  sum(df$Grade2)/4,
  sum(df$Grade3)/4,
  sum(df$Grade4)/4
)

for(i in 1:4){
  if(testAvg[i] < 80){
    cat("The", i, "th test was difficult.\n")
  }
}

```

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

```
cat("\n--- d. highest > 90 ---\n")
```

```
##
```

```
## --- d. highest > 90 ---
```

```

# d. without max()
for(i in 1:nrow(df)){
  highest <- sort(c(df$Grade1[i], df$Grade2[i], df$Grade3[i], df$Grade4[i]))[4]
  if(highest > 90){
    cat(df$Name[i], "'s highest grade this semester is ", highest, ".\n", sep="")
  }
}

```

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

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