Worksheet_#4a

Kyla Ysabelle De la Cruz

2023-11-15

```
#1.
library(readxl)
HouseholdData <- read_excel("C:\\Users\\ASUS\\Documents\\Worksheet#4\\Worksheet#4\\HouseholdData.xlsx")
HouseholdData
```

```
## # A tibble: 28 x 3
##    ‘Shoe Size‘ Height Gender
##          <dbl>  <dbl> <chr>
##  1         6.5   66   F
##  2         9     68   F
##  3         8.5   64.5 F
##  4         8.5   65   F
##  5        10.5   70   M
##  6         7     64   F
##  7         9.5   70   F
##  8         9     71   F
##  9        13     72   M
## 10         7.5   64   F
## # i 18 more rows
```

```
#1.a. In the data frame, it shows three columns; Shoe size, Height, and Gender along
# with corresponding data within the rows.
```

```
#1.b

males <-  HouseholdData[HouseholdData$Gender == "M",]
males
```

```
## # A tibble: 14 x 3
##    ‘Shoe Size‘ Height Gender
##          <dbl>  <dbl> <chr>
##  1        10.5   70   M
##  2        13     72   M
##  3        10.5   74.5 M
##  4        12     71   M
##  5        10.5   71   M
##  6        13     77   M
##  7        11.5   72   M
##  8        10     72   M
##  9         8.5   67   M
## 10        10.5   73   M
## 11        10.5   72   M
## 12        11     70   M
## 13         9     69   M
## 14        13     70   M
```

```
females <-  HouseholdData[HouseholdData$Gender == "F",]
females
```

```
## # A tibble: 14 x 3
##    ‘Shoe Size‘ Height Gender
##          <dbl>  <dbl> <chr>
##  1         6.5   66   F
##  2         9     68   F
##  3         8.5   64.5 F
##  4         8.5   65   F
##  5         7     64   F
##  6         9.5   70   F
##  7         9     71   F
##  8         7.5   64   F
##  9         8.5   67   F
## 10         8.5   59   F
## 11         5     62   F
## 12         6.5   66   F
## 13         7.5   64   F
## 14         8.5   69   F
```

```
#1.c

colMeans(HouseholdData[c(1, 2)])
```

```
## Shoe Size    Height
##  9.410714 68.571429
```

```
# Shoe Size    Height
#  9.410714 68.571429
```

```
#1.d
# Yes, because it the person's height corresponds to their shoe size.
```

```
#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(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 
```

```
summary(months_vector)
```

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

```
# Yes, they are both useful in this case since factoring the factor_months_vector shows you the
# amount of times a month had been repeated/stated within the vector.
# While in the months_vector, it shows the length, class, and mode of the variables inside the object.
```

```
#4.
factor_data <- c("East", "West", "West", "West", "West", "North","North","North")
new_order_data <- factor(factor_data, levels = c("East","West","North"))
new_order_data
```

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

```
#5.
excel_read <- read.csv("import_march.csv")
excel_read
```

```
##   Students Strategy.1 Strategy.2 Strategy.3
## 1     Male          8         10          8
## 2     Male          4          8          6
## 3     Male          0          6          4
## 4   Female         14          4         15
## 5   Female         10          2         12
## 6   Female          6          0          9
```

```
#6.

# number <- 1:50
```

```r
# user_input <- as.integer(readline(prompt = "Enter a number 1-50: "))
#
# if (user_input >= 1 && user_input <= 50) {
#   cat(TRUE)
# } else{
#   cat(user_input)
# }

#Turned into comment because it won't knit
```

```r
minimum_bills <- function(price) {
  bills_type <- c(1000, 500, 200, 100, 50)
  total <- 0

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

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

snack_price <- 1350
minimum_bills(snack_price)
```

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

```r
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] 5
```

```
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
```

```
##    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
```

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

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

```
## [1] NAME     Grade1   Grade2   Grade3   Grade4   AvgScore
## <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_Ave <- data[data$NAME == "Annie", "AvgMathScore"]
paste("Annie's average grade this semester is", Annie_Ave)
```

```
## [1] "Annie’s average grade this semester is 83.75"
```

```
Thea_Ave <- data[data$NAME == "Thea", "AvgMathScore"]
paste("Thea's average grade this semester is", Thea_Ave)
```

```
## [1] "Thea’s average grade this semester is 80"
```

```
Steve_Ave <- data[data$NAME == "Steve", "AvgMathScore"]
paste("Steve's average grade this semester is", Steve_Ave)
```

```
## [1] "Steve’s average grade this semester is 73.75"
```

```
Hanna_Ave <- data[data$NAME == "Hanna", "AvgMathScore"]
paste("Hanna's average grade this semester is", Hanna_Ave)
```

```
## [1] "Hanna’s average grade this semester is 90"
```

```
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
```

```
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
```