Worksheet 3a in R

Erl Syron R. Espadon

```
LETTERS
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S"
## [20] "T" "U" "V" "W" "X" "Y" "Z"
```

#1. VECTORS

#a. Produce a vector that contains the first 11 letters.

```
first_11_letters <- LETTERS[1:11]
first_11_letters
```

```
##  [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K"
```

#b. Produce a vector that contains the odd-numbered letters.

```
odd_letters <- LETTERS[seq(1, length(LETTERS), by=2)]
odd_letters
```

```
##  [1] "A" "C" "E" "G" "I" "K" "M" "O" "Q" "S" "U" "W" "Y"
```

#c. Produce a vector that contains the vowels.

```
vowels <- LETTERS[c(1, 5, 9, 15, 21)]
vowels
```

```
## [1] "A" "E" "I" "O" "U"
```

#d. Produce a vector that contains the last 5 lowercase letters.

```
last_5_lowercase <- letters[22:26]
last_5_lowercase
```

```
## [1] "v" "w" "x" "y" "z"
```

#e. Produce a vector that contains letters between 15 to 24 in lowercase.

```
mid_lowercase <- letters[15:24]
mid_lowercase
```

```
##  [1] "o" "p" "q" "r" "s" "t" "u" "v" "w" "x"
```

#2. Create a vector with the average temperatures in April for Tuguegarao City, Manila, Iloilo City, Tacloban, Samal Island, and Davao City. #a. Character vector for the cities.

```
city <- c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban", "Samal Island", "Davao City")
city
```

```
## [1] "Tuguegarao City" "Manila"          "Iloilo City"     "Tacloban"
## [5] "Samal Island"    "Davao City"
```

#b. Temperature vector.

```r
temp <- c(42, 39, 34, 34, 30, 27)
temp
```

```
## [1] 42 39 34 34 30 27
```

#c. Combine the city and temp vectors into a dataframe.

```r
weather_df <- data.frame(City = city, Temperature = temp)
weather_df
```

```
##              City Temperature
## 1 Tuguegarao City          42
## 2          Manila          39
## 3     Iloilo City          34
## 4        Tacloban          34
## 5    Samal Island          30
## 6      Davao City          27
```

#d. Rename the columns using names() function.

```r
names(weather_df) <- c("City", "Temperature")
weather_df
```

```
##              City Temperature
## 1 Tuguegarao City          42
## 2          Manila          39
## 3     Iloilo City          34
## 4        Tacloban          34
## 5    Samal Island          30
## 6      Davao City          27
```

#e. Print the structure using str() function.

```r
str(weather_df)
```

```
## 'data.frame':    6 obs. of  2 variables:
##  $ City       : chr  "Tuguegarao City" "Manila" "Iloilo City" "Tacloban" ...
##  $ Temperature: num  42 39 34 34 30 27
```

#f. Content of row 3 and row 4.

```r
row_3_4 <- weather_df[3:4, ]
row_3_4
```

```
##          City Temperature
## 3 Iloilo City          34
## 4    Tacloban          34
```

#g. Display the city with the highest and lowest temperature.

```r
highest_temp <- weather_df[which.max(weather_df$Temperature), ]
lowest_temp <- weather_df[which.min(weather_df$Temperature), ]
highest_temp
```

```
##              City Temperature
## 1 Tuguegarao City          42
```

```r
lowest_temp
```

```
##         City Temperature
## 6 Davao City          27
```

#MATRICES #1. a. Create a matrix of numbers one to eight and eleven to fourteen with four columns and three rows.

```
matrix1 <- matrix(c(1:8, 11:14), ncol = 4, byrow = TRUE)
matrix1
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    1    2    3    4
## [2,]    5    6    7    8
## [3,]   11   12   13   14
```

#b. Multiply the matrix by two.

```
matrix_multiplied <- matrix1 * 2
matrix_multiplied
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    2    4    6    8
## [2,]   10   12   14   16
## [3,]   22   24   26   28
```

#wc. Content of row 2.

```
row_2 <- matrix1[2, ]
row_2
```

```
## [1] 5 6 7 8
```

#d. Display columns 3 and 4 in rows 1 and 2.

```
cols_3_4_rows_1_2 <- matrix1[1:2, 3:4]
cols_3_4_rows_1_2
```

```
##      [,1] [,2]
## [1,]    3    4
## [2,]    7    8
```

#e. Display only columns 2 and 3, row 3.

```
cols_2_3_row_3 <- matrix1[3, 2:3]
cols_2_3_row_3
```

```
## [1] 12 13
```

#f. Display only column 4.

```
col_4 <- matrix1[, 4]
col_4
```

```
## [1]  4  8 14
```

#g. Name the rows and columns.

```
dimnames(matrix_multiplied) <- list(c("isa", "dalawa", "tatlo"), c("uno", "dos", "tres", "quatro"))
matrix_multiplied
```

```
##        uno dos tres quatro
## isa      2   4    6      8
## dalawa  10  12   14     16
## tatlo   22  24   26     28
```

#h. Reshape the matrix using dim().

```
dim(matrix1) <- c(6, 2)
matrix1
```

```
##      [,1] [,2]
## [1,]    1    3
## [2,]    5    7
## [3,]   11   13
## [4,]    2    4
## [5,]    6    8
## [6,]   12   14
```

#ARRAYS #a.Create an array with the specified numeric values, repeated twice, and a three-dimensional array with 4 columns and 2 rows.

```
values <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)
array_data <- array(rep(values, each = 2), dim = c(2, 4, 3))
array_data
```

```
## , , 1
##
##      [,1] [,2] [,3] [,4]
## [1,]    1    2    3    6
## [2,]    1    2    3    6
##
## , , 2
##
##      [,1] [,2] [,3] [,4]
## [1,]    7    8    9    0
## [2,]    7    8    9    0
##
## , , 3
##
##      [,1] [,2] [,3] [,4]
## [1,]    3    4    5    1
## [2,]    3    4    5    1
```

#b. Dimensions of the array.

```
dim(array_data)
```

```
## [1] 2 4 3
```

#c. Name the rows and columns.

```
dimnames(array_data) <- list(letters[1:2], LETTERS[1:4], c("1st-Dimensional Array", "2nd-Dimensional Ar
array_data
```

```
## , , 1st-Dimensional Array
##
##   A B C D
## a 1 2 3 6
## b 1 2 3 6
##
## , , 2nd-Dimensional Array
##
##   A B C D
## a 7 8 9 0
```

```
## b 7 8 9 0
##
## , , 3rd-Dimensional Array
##
##   A B C D
## a 3 4 5 1
## b 3 4 5 1
```