

Worksheet #3a

John Rick Estoce

2025-10-13

- a. You need to produce a vector that contains the first 11 letters.

```
LETTERS[1:11]
```

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

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

```
LETTERS[seq(1, 26, 2)]
```

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

- c. Produce a vector that contains the vowels

```
LETTERS[LETTERS %in% c("A", "E", "I", "O", "U")]
```

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

- d. Vector that contains the last 5 lowercase letters

```
LETTERS[22:26]
```

```
## [1] "V" "W" "X" "Y" "Z"
```

- e. Vector that contains letters between 15 to 24 (inclusive) in lowercase

```
LETTERS[15:24]
```

```
## [1] "O" "P" "Q" "R" "S" "T" "U" "V" "W" "X"
```

##2. Create a vector(not a dataframe) with the average temperatures in April for Tugue- garao City, Manila, Iloilo City, Tacloban, Samal Island, and Davao City. The average temperatures in Celcius are 42, 39, 34, 34, 30, and 27 degrees.

- a. What is the R code and its result for creating a character vector for the city/town of Tuguegarao City, Manila, Iloilo City, Tacloban, Samal Island, and Davao City? Name the object as city. The names should follow the same order as in the instruction.

```
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. The average temperatures in Celcius are 42, 39, 34, 34, 30, and 27 degrees. Name the object as temp. Write the R code and its output. Numbers should also follow what is in the instruction.

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

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

- c. Create a dataframe to combine the city and the temp by using 'data.frame()'. What the R code and its result?

```
data.frame(
  city = c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban", "Samal Island", "Davao City"),
  temp = c(42, 39, 34, 34, 30, 27)
)
```

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

- d. Associate the dataframe you have created in 2.(c) by naming the columns using the names() function. Change the column names by using names() function as City and Temperature. What is the R code and its result?

```
df <- data.frame(
  city = c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban", "Samal Island", "Davao City"),
  temp = c(42, 39, 34, 34, 30, 27)
)
```

```
names(df) <- c("City", "Temperature")
```

```
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 by using str() function. Describe the output.

```
str(df)
```

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

Explanation: 'data.frame': 6 obs. of 2 variables: Your object df is a data frame with 6 observations (rows) and 2 variables (columns).

\$ City : chr ... The first column named City is of type character vector (chr). It contains the city names as strings (like "Tuguegarao City", "Manila", etc.).

\$ Temperature : num ... The second column named Temperature is of type numeric (num). It contains the average temperature values (42, 39, 34, etc.).

- f. From the answer in d, what is the content of row 3 and row 4 What is its R code and its output?

```
df[3:4, ]
```

```
##           City Temperature
## 3 Iloilo City           34
```

```
## 4      Tacloban      34
```

- g. From the answer in d, display the city with highest temperature and the city with the lowest temperature. What is its R code and its output?

```
# City with highest temperature
df[df$Temperature == max(df$Temperature), "City"]
```

```
## [1] "Tuguegarao City"
```

```
# City with lowest temperature
df[df$Temperature == min(df$Temperature), "City"]
```

```
## [1] "Davao City"
```

- ##2. Create a matrix of one to eight and eleven to fourteen with four columns and three rows. a. What will be the R code for the #2 question and its result?

```
matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
```

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

- b. Multiply the matrix by two. What is its R code and its result?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
```

```
mat * 2
```

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

- c. What is the content of row 2? What is its R code?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
mat[2, ]
```

```
## [1]  2  5  8 13
```

- d. What will be the R code if you want to display the column 3 and column 4 in row 1 and row 2? What is its output?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
```

```
mat[1:2, 3:4]
```

```
##      [,1] [,2]
## [1,]    7   12
## [2,]    8   13
```

- e. What is the R code is you want to display only the columns in 2 and 3, row 3? What is its output?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)
```

```
mat[3, 2:3]
```

```
## [1]  6 11
```

- f. What is the R code is you want to display only the columns 4? What is its output?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)

mat[, 4]
```

```
## [1] 12 13 14
```

g. Name the rows as isa, dalawa, tatlo and columns as uno, dos, tres, quatro for the matrix that was created in b. What is its R code and corresponding output?

```
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)

rownames(mat) <- c("isa", "dalawa", "tatlo")
colnames(mat) <- c("uno", "dos", "tres", "quatro")

mat
```

```
##          uno dos tres quatro
## isa         1  4   7    12
## dalawa      2  5   8    13
## tatlo       3  6  11    14
```

h. From the original matrix you have created in a, reshape the matrix by assigning a new dimension with dim(). New dimensions should have 2 columns and 6 rows. What will be the R code and its output?

```
# Original matrix
mat <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4)

# Change dimensions to 6 rows and 2 columns
dim(mat) <- c(6, 2)

# Print the reshaped matrix
print(mat)
```

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

##3 An array contains 1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1

a. Create an array for the above numeric values. Each values will be repeated twice What will be the R code if you are to create a three-dimensional array with 4 columns and 2 rows. What will be its output?

```
# Original numeric values
values <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)

# Repeat each value twice
values_repeated <- rep(values, 2)

# Create a 3-dimensional array with 2 rows, 4 columns, and 3 layers (because 24 / (2*4) = 3)
arr <- array(values_repeated, dim = c(2, 4, 3))

# Display the array
arr
```

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

b. How many dimensions do your array have? The array you created has 3 dimensions.

```
length(dim(arr))
```

```
## [1] 3
```

c. Name the rows as lowercase letters and columns as uppercase letters starting from the A. The array names should be “1st-Dimensional Array”, “2nd-Dimensional Array”, and “3rd-Dimensional Array”. What will be the R codes and its output

```
# Create the array (assuming previous code)
values <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)
values_repeated <- rep(values, 2)
arr <- array(values_repeated, dim = c(2, 4, 3))

# Name the rows, columns, and 3rd dimension
rownames(arr) <- letters[1:2] # "a", "b"
colnames(arr) <- LETTERS[1:4] # "A", "B", "C", "D"
dimnames(arr)[[3]] <- c("1st-Dimensional Array", "2nd-Dimensional Array", "3rd-Dimensional Array")

# Print the array
arr
```

```
## , , 1st-Dimensional Array
##
##      A B C D
## a 1 3 7 9
## b 2 6 8 0
##
## , , 2nd-Dimensional Array
##
##      A B C D
## a 3 5 1 3
## b 4 1 2 6
##
## , , 3rd-Dimensional Array
##
##      A B C D
## a 7 9 3 5
```

b 8 0 4 1