

RWorksheet_Josue#3a

Miguel F. Josue Jr.

2024-10-02

1. There is a built-in vector `LETTERS` contains the uppercase letters of the alphabet and letters which contains the lowercase letters of the alphabet.

```
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"
```

```
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"
```

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.

```
odd <- LETTERS[seq(1, 26, by = 2)]
odd
```

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

c. Produce a vector that contains the vowels

```
vowels <- LETTERS[LETTERS %in% c('A', 'E', 'I', 'O', 'U')]
vowels
```

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

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

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

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

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

```
vec1 <- letters[15:24]
vec1
```

```
## [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 Tuguegarao 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
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.
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
tempdata <- data.frame(city, temp)
tempdata
```

```
##           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.
names(tempdata) <- c("City", "Temperature")
tempdata
```

```
##           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(tempdata)
```

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

```
#f. From the answer in d, what is the content of row 3 and row 4 What is its R code and its output?
rows_3_and_4 <- tempdata[3:4, ]
rows_3_and_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
highest_temp <- tempdata[which.max(tempdata$Temperature), ]
lowest_temp <- tempdata[which.min(tempdata$Temperature), ]

print(highest_temp)
```

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

```
print(lowest_temp)
```

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

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

```
##      [,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?
double_matrix <- my_matrix * 2
double_matrix
```

```
##      [,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?

```
row_2_content <- my_matrix[2, ]
row_2_content
```

```
## [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?

```
rows_1_2 <- my_matrix[1:2, 3:4]
rows_1_2
```

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

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

```
row_3_columns_2_3 <- my_matrix[3, 2:3]
print(row_3_columns_2_3)
```

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

#f. What is the R code if you want to display only the column 4? What is its output?

```
column_4 <- my_matrix[, 4]
column_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 the previous question.

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

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

#h. From the original matrix you have created in a, reshape the matrix by assigning a new dimension with 6 rows and 2 columns.

```
rmatrix <- my_matrix
dim(rmatrix) <- c(6, 2)
rmatrix
```

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

```
## , , 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. How many dimensions do your array have?
num_dimensions <- length(dim(my_array))
print(paste("There are", num_dimensions, "dimensions"))
```

```
## [1] "There are 3 dimensions"
```

```
#c. Name the rows as lowercase letters and columns as uppercase letters starting from the A. The array
dimnames(my_array) <- list(c("a", "b"), c("A", "B", "C", "D"), c("1st-Dimensional Array", "2nd-Dimensional Array", "3rd-Dimensional Array"))
my_array
```

```
## , , 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
```