

RWorksheet_Condag#3a

Angel Blase

2024-10-01

#USING VECTORS

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.

Based on the above vector LETTERS:

- a. You need to 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"
```

Based on the above vector letters:

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

```
last_five <- tail(letters, 5)
last_five
```

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

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

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

```
## [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. Creating 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. Creating a numeric vector for the average temperatures.

```
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 temp

```
city_temp <- data.frame(city, temp)
city_temp
```

```
##           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. Rename the columns using names() to “City” and “Temperature”.

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

```
##           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(city_temp)
```

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

Description: The output of str(city_temp) reveals that city_temp is a dataframe with 6 observations and 2 variables. The first variable, City, contains character data representing city names, while the second variable, Temperature, holds numeric data for average temperatures. This structure provides a quick overview of the data types and a sample of the contents within the dataframe.

f. From the answer in d, what is the content of row 3 and row 4?

```
city_temp[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.

```
highest_temp <- city_temp[which.max(city_temp$Temperature), ]
highest_temp
```

```
##           City Temperature
## 1 Tuguegarao City          42
lowest_temp <- city_temp[which.min(city_temp$Temperature), ]
lowest_temp
```

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

#USING MATRICES

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?

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

```
##      [,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?

```
matrixB <- matrixA * 2
matrixB
```

```
##      [,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 <- matrixA[2, ]
row_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?

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

```
##      [,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?

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

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

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

```
column_4 <- matrixA[, 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 b. What is its R code and corresponding output?

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

```
##      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 dim(). New dimensions should have 2 columns and 6 rows. What will be the R code and its output?

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

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

#USING ARRAYS

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?

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

```
## , , 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?

```
num_dim <- length(dim(my_array))
num_dim
```

```
## [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?

```
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 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
```