RWorksheet_Hallares#3a

Marc Christian P. Hallares

2025-10-13

Using Vectors Based on LETTERS

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

first_11_letters <- LETTERS[1:11] first_11_letters

Output:

[1] “A” “B” “C” “D” “E” “F” “G” “H” “I” “J” “K”

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

odd_numbered_letters <- LETTERS[c(1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25)] odd_numbered_letters # Output: # [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(which(LETTERS %in% c(“A”, “E”, “I”, “O”, “U”)))] vowels # Output: # [1] “A” “E” “I” “O” “U”

Using Vectors Based on letters

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

last_5_letters <- letters[21:26] # letters has 26 elements last_5_letters # Output: # [1] “v” “w” “x” “y” “z”

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

letters_15_to_24 <- letters[15:24] letters_15_to_24 # Output: # [1] “o” “p” “q” “r” “s” “t” “u” “v” “w” “x”

3. Creating and Manipulating Vectors and Dataframes for Average Temperatures

a. Create a character vector for the cities.

city <- c(“Tuguegarao City”, “Manila”, “Iloilo City”, “Tacloban”, “Samal Island”, “Davao City”) city # Output: # [1] “Tuguegarao City” “Manila” “Iloilo City” “Tacloban”
# [5] “Samal Island” “Davao City”

b. Create a numeric vector for the temperatures.

temp <- c(42, 39, 34, 34, 30, 27) temp # Output: # [1] 42 39 34 34 30 27

c. Create a dataframe combining city and temp.

df <- data.frame(city, temp) df # Output:
# 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 of the dataframe to “City” and “Temperature”.

names(df) <- c(“City”, “Temperature”) df # Output:
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(). Describe the output.

str(df) # Output: # ‘data.frame’: 6 obs. of 2 variables: # $ City : chr “Tuguegarao City” “Manila” “Iloilo City” “Tacloban” ... # $ Temperature: num 42 39 34 34 30 27

f. Display the content of row 3 and row 4.

df[3:4,] # Output:
City Temperature # 3 Iloilo City 34 # 4 Tacloban 34

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

City with highest temperature

city_with_max_temp <- df$City[which.max(df$Temperature)] city_with_max_temp

City with lowest temperature

city_with_min_temp <- df$City[which.min(df$Temperature)] city_with_min_temp # Output: # [1] “Tuguegarao City” # [1] “Davao City”

4. Using Matrices

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

my_matrix <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) my_matrix # Output:
# [,1] [,2] [,3] [,4] # [1,] 1 3 5 11 # [2,] 2 4 6 12 # [3,] 7 8 11 13
# [4,] 8 11 12 14

b. Multiply the matrix by two.

multiplied_matrix <- my_matrix * 2 multiplied_matrix # Output:
# [,1] [,2] [,3] [,4] # [1,] 2 6 10 22 # [2,] 4 8 12 24 # [3,] 14 16 22 26 # [4,] 16 22 24 28

c. What is the content of row 2?

my_matrix[2,] # Output: # [1] 2 4 6 12

d. Display column 3 and column 4 in row 1 and row 2.

my_matrix[1:2, 3:4] # Output:
[,1] [,2] # [1,] 5 11 # [2,] 6 12

e. Display only columns 2 and 3 in row 3.

my_matrix[3, 2:3] # Output: # [1] 8 11

f. Display only column 4.

my_matrix[, 4] # Output: # [1] 11 12 13 14

g. Name the rows and columns for the matrix from b.

rownames(multiplied_matrix) <- c(“isa”, “dalawa”, “tatlo”, “apat”) colnames(multiplied_matrix) <- c(“uno”, “dos”, “tres”, “quatro”) multiplied_matrix # Output:
uno dos tres quatro # isa 2 6 10 22 # dalawa 4 8 12 24 # tatlo 14 16 22 26 # apat 16 22 24 28

h. Reshape the original matrix from a with new dimensions (2 columns and 6 rows).

dim(my_matrix) <- c(6, 2)
my_matrix # Output:
[,1] [,2] # [1,] 1 5 # [2,] 3 11 # [3,] 2 6 # [4,] 4 12 # [5,] 7 11 # [6,] 8 13

5. Using Arrays

a. Create a three-dimensional array with the given values, each repeated twice.

my_array <- array(c(1,2,3,6,7,8,9,0,3,4,5,1,1,2,3,6,7,8,9,0,3,4,5,1), dim = c(2, 4, 3)) my_array # Output: # , , 1 # [,1] [,2] [,3] [,4] # [1,] 1 3 4 1 # [2,] 2 6 5 1

, , 2

[,1] [,2] [,3] [,4]

[1,] 3 9 3 1

[2,] 6 0 4 2

, , 3

[,1] [,2] [,3] [,4]

[1,] 7 3 5 3

[2,] 8 4 1 6

b. How many dimensions does the array have?

dim(my_array) # Output: # [1] 2 4 3

c. Name the rows, columns, and dimensions.

rownames(my_array) <- letters[1:2] # Lowercase letters for rows colnames(my_array) <- LETTERS[1:4] # Uppercase letters for columns dimnames(my_array) <- list(c("1st-Dimensional Array", "2nd-Dimensional Array", "3rd-Dimensional Array")) my_array # Output: # $1st-Dimensional Array # A B C D # a 1 3 4 1 # b 2 6 5 1

$2nd-Dimensional Array

A B C D

a 3 9 3 1

b 6 0 4 2

$3rd-Dimensional Array

A B C D

a 7 3 5 3

b 8 4 1 6