Worksheet3

2023-10-08

#Number 1: #A. LETTERS letters first_11_letters <- LETTERS[1:11] first_11_letters #B. odd_numbers <- seq(1, 26, by = 2) odd_letters <- LETTERS[odd_numbers] odd_letters #C. vowels <- LETTERS[LETTERS %in% c("A", "E", "I", "O", "U")] vowels #D. last_5_low <- tail(letters, 5) last_5_low #E. letters15to24 <- letters[15:24] letters15to24

#Number 2:

#A. The result is that city is a vector with specified city. city <- c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban", "Samal Island", "Davao City") city #B. The result is that temp is a vector with specified temperatures temp <- c(42, 39, 34, 34, 30, 27) temp #C. The result is now a data frame which it combines the city and temp data <- data.frame(City = city, Temperature = temp) data #D. The column names now changed to City and Temperature names(data) <- c("City", "Temperature") names(data)

#Number 3: #E The result the str()function will display the structure of datafram "data" str(data) #F it will retrieve the content of row 3 and row 4 data[3,] data[4,] #G The result will Display that City with the highest and lowest temperature max_temp_city <- data[data$Temperature == max(data$Temperature), "City"] min_temp_city <- data[data$Temperature == min(data$Temperature), "City"] max_temp_city min_temp_city

USING MATRIX

#Number 2: #row = 2 dab <-matrix(c(5,6,7,4,3,2,1,2,3,7,8,9),nrow = 2) dab #row = 3 and column = 2 matrix(data = c(3,4,5,6,7,8),3,2) matrix (data) #creating a diagonal matrix where value will always be 1

diag(1,nrow=6,ncol = 5) diag(6) #Number 2: #A. matrix(c(1:8, 11:14), nrow = 3, ncol = 4) #RESULT:[,1] [,2] [,3] [,4] #[1,] 1 4 7 12 #[2,] 2 5 8 13 #[3,] 3 6 11 14 #B. matrix(c(1:8, 11:14), nrow = 3, ncol = 4) * 2 #REsult is value is multiplied by 2: # [,1] [,2] [,3] [,4] #[1,] 2 8 14 24 #[2,] 4 10 16 26 #[3,] 6 12 22 28 #C: REsult content of row to = 2 5 8 13 matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) matrix_data[2,] #D matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) matrix_data[1:2, 3:4] #OUTPUT: [,1] [,2] # [1,] 7 12 # [2,] 8 13 #E matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) matrix_data[3, 2:3] #OUTPUT: 6 11 #F matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) matrix_data[, 4] #OUTPUT: 12 13 14 #G matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) matrix_data #OUTPUT: [,1] [,2] [,3] [,4] # [1,] 1 4 7 12 # [2,] 2 5 8 13 # [3,] 3 6 11 14 rownames(matrix_data) <- c("isa", "dalawa", "tatlo") rownames(matrix_data) #OUTPUT: "isa" "dalawa" "tatlo" colnames(matrix_data) <- c("uno", "dos", "tres", "quatro") colnames(matrix_data) #OUTPUT: "uno" "dos" "tres" "quatro" #H library(dplyr) matrix_data <- matrix(c(1:8, 11:14), nrow = 3, ncol = 4) new_matrix <- matrix_data %>% t() %>% as.vector() %>% matrix(ncol = 2)

#USING ARRAYS

#Number 3: # B: result array has 3 dimensions: 2 rows, 4 columns, and 2 "layers" (depth). So, it is a three-dimensional array. # Given numeric values values <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)

matrix_data <- matrix(rep(values, each = 2), nrow = 2)

array_data <- array(matrix_data, dim = c(2, 4, 2))

rownames(array_data) <- c("a", "b") colnames(array_data) <- c("A", "B", "C", "D")

Assign names to the dimensions

dimnames(array_data) <- list(“1st-Dimensional Array” = rownames(array_data), “2nd-Dimensional Array” = colnames(array_data), “3rd-Dimensional Array” = NULL) print(array_data)