

RWorksheet_anastacio#3a.Rmd

Kyle Johnsen Anastacio

2025-10-14

```
# Uppercase letters A to 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"

#Output [1] "A" "B" "C" "D" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O" "P" "Q" "R" "S"
#[25] "T" "U" "V" "W"

# Lowercase letters a to 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"

#output [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k" "l" "m" "n" "o" "p" "q" "r" "s"
#[25] "t" "u" "v" "w"

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

first_11 <- c(LETTERS[1:11])
first_11

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

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

odd_letters <- c(LETTERS[seq(1, length(LETTERS), 2)])
odd_letters

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

#c. Produce a vector that contains the vowels

vowels <- c(LETTERS[c(1, 5, 9, 15, 21)])
vowels

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

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

```

last_5 <- c(letters[22:26])
last_5

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

#e. Produce a vector that contains letters between 15 to 24 letters in lowercase.
mid_lower <- c(letters[15:24])
mid_lower

## [1] "o" "p" "q" "r" "s" "t" "u" "v" "w" "x"
# output [1] "o" "p" "q" "r" "s" "t" "u" "v" "w" "x"

#2
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"

#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.

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

## [1] 42 39 34 34 30 27

#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

#output [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?
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

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

```

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

```

```

#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 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

```

```

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

```

city_temp[3:4, ]

```

```

##           City Temperature
## 3     Iloilo City        34
## 4      Tacloban          34
#
#           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_temp[which.max(city_temp$Temperature), ]

```

```

##           City Temperature
## 1 Tuguegarao City        42

```

```

city_temp[which.min(city_temp$Temperature), ]

##          City Temperature
## 6 Davao City           27
#output > city_temp[which.max(city_temp$Temperature), ]
#          City Temperature
#1 Tuguegarao City        42
#> city_temp[which.min(city_temp$Temperature), ]
#          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?
matrix_values <- c(1:8, 11:14)
mat <- matrix(matrix_values, ncol = 4, nrow = 3)
mat

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

#[,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?
mat2 <- mat * 2
mat2

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

#[,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[2, ]

## [1] 2 5 8 13
#Output [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[1:2, 3:4]

```

```

##      [,1] [,2]
## [1,]    7   12
## [2,]    8   13
#      [,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[3, 2:3]

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

#f. What is the R code is you want to display only the columns 4? What is its output?
mat[, 4]

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

#g. Name the rows as isa, dalawa, tatlo and columns as uno, dos, tres, quattro
#for the matrix that was created in b.'. What is its R code and corresponding output?
rownames(mat2) <- c("isa", "dalawa", "tatlo")
colnames(mat2) <- c("uno", "dos", "tres", "quattro")
mat2

##      uno dos tres quattro
## isa     2   8   14    24
## dalawa  4  10   16    26
## tatlo   6  12   22    28
#      uno dos tres quattro
#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(mat) <- c(6, 2)
mat

##      [,1] [,2]
## [1,]    1    7
## [2,]    2    8
## [3,]    3   11
## [4,]    4   12
## [5,]    5   13
## [6,]    6   14
#      [,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
values <- c(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?
array_sample <- array(rep(values, 2), dim = c(2, 4, 3))
array_sample

## , , 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
#, , 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?
dim(array_sample)

## [1] 2 4 3
#[1] 2 4 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?

# Assign names for rows and columns
rownames(array_sample) <- letters[1:2]      # lowercase a, b
colnames(array_sample) <- LETTERS[1:4]       # uppercase A, B, C, D

# Assign names for each layer (dimension)
dimnames(array_sample) <- list(
  Row = letters[1:2],
  Column = LETTERS[1:4],
  Dimension = c("1st-Dimensional Array", "2nd-Dimensional Array", "3rd-Dimensional Array")
)

array_sample

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

#     Column
#Row A B C D
#   a 1 3 7 9
#   b 2 6 8 0

#, , Dimension = 2nd-Dimensional Array

#     Column
#Row A B C D
#   a 3 5 1 3
#   b 4 1 2 6

#, , Dimension = 3rd-Dimensional Array

```

```
#   Column
#Row A B C D
#  a 7 9 3 5
#  b 8 0 4 1
```