

RWorksheet_SOCO-3

There is a built-in vector LETTERS contains the uppercase letters of the alphabet and letters which c
#Based on the above vector LETTERS:

```
x <- LETTERS  
x
```

```
## [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"
```

```
x <- letters  
x
```

```
## [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.
`print(x[1:11])`

```
## [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j" "k"
```

#B. Produce a vector that contains the odd numbered letters.
`print(x[seq(1, 26, 2)])`

```
## [1] "a" "c" "e" "g" "i" "k" "m" "o" "q" "s" "u" "w" "y"
```

#c. Produce a vector that contains the vowels
`vowels <- x`
`vowels[c(1, 5, 9, 15, 21)]`

```
## [1] "a" "e" "i" "o" "u"
```

#d. Produce a vector that contains the last 5 lowercase letters.
`y <- letters`
`last_5 <- y[22:26]`
`print(last_5)`

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

#e. Produce a vector that contains letters between 15 to 24 letters in lowercase.
`letters_15_to_24 <- y[15:24]`
`print(letters_15_to_24)`

```
## [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

#a. What is the R code and its result for creating a character vector for the city/town of Tuguegarao C

```
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

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

```
##           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(weather) <- c("City", "Temperature")
weather
```

```
##           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(weather)
```

```
## '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?

```
weather[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

```
weather[which.max(weather$Temperature), ]
```

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

```
weather[which.min(weather$Temperature), ]
```

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

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

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

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

```
m2 <- m * 2
m2
```

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

```
m[2, ]
```

```
## [1]  2  5  8 13
```

```
m[1:2, 3:4]
```

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

```
m[3, 2:3]
```

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

```
m[, 4]
```

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

```
rownames(m2) <- c("isa", "dalawa", "tatlo")
colnames(m2) <- c("uno", "dos", "tres", "quatro")
m2
```

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

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

```
##      [,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

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

```
length(dim(my_array))
```

```
## [1] 3
```

```

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

dimnames(my_array) <- list(
  c("a", "b"),           # row names
  c("A", "B", "C", "D"), # column names
  c("1st-Dimensional Array",
    "2nd-Dimensional Array",
    "3rd-Dimensional Array") # layer names
)

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

```