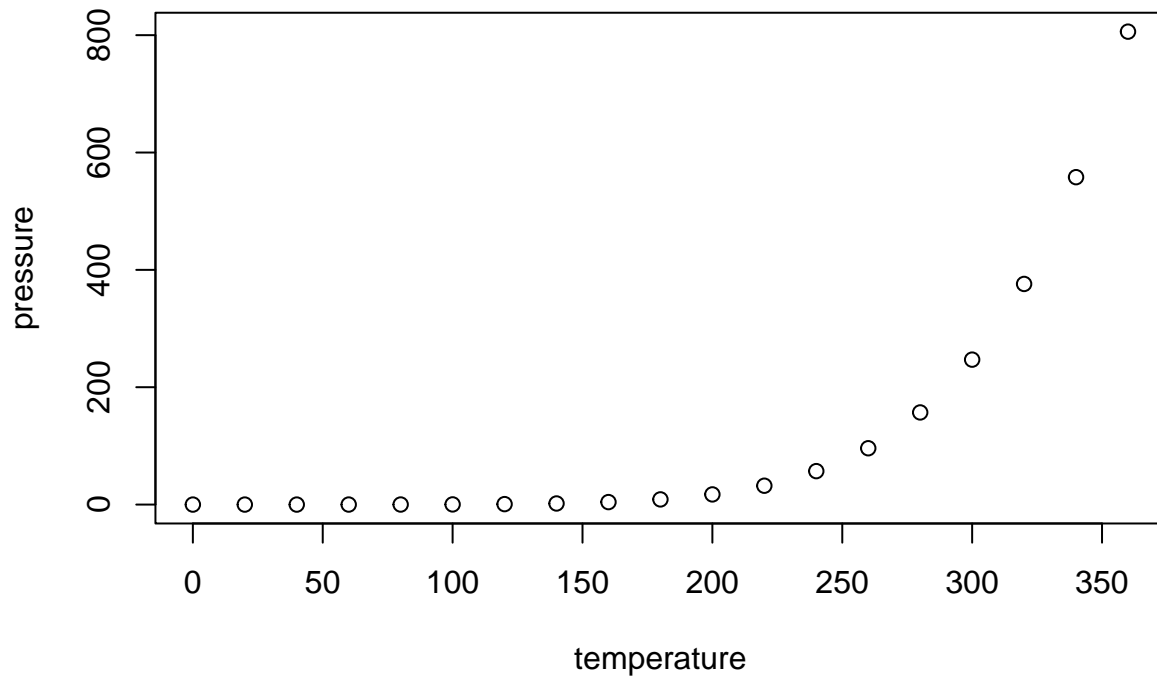


# RWorksheet\_Paclibar#3

Jhon Albert Paclibar

2024-09-30



*# 1. There is a built-in vector LETTERS contains the uppercase letters of the alphabet and letters which*

*# Based on the above vector LETTERS:*

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

```
BLetters <- c("A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P","Q","R","S","T","U","V",
sletters <- c("a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q","r","s","t","u","v",
eleven <- BLetters[1:11]
eleven
```

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

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

```
odd <- BLetters[seq("1","26", by = 2)]
odd
```

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

*#c. Produce a vector that contains the vowels*

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

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

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

```
last5 <- tail(letters,5)
last5
```

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

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

```
between <- sletters[c(15:24)]
between
```

```
## [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 Tugue-garao City, Manila*

```
tempe <- c(42,39,34,34,30,27)
```

*#a. What is the R code and its result for creating a character vector for the city/townof Tuguegarao Ci*

```
place <- c("Tuguegarao City", "Manila", "Iloilo City", "Tacloban","Samal Island", "Davao City")
```

*#b. The average temperatures in Celcius are 42, 39, 34, 34, 30, and 27 degrees.Name the object as temp.*

```
Mtemp <- mean(tempe)
Mtemp
```

```
## [1] 34.33333
```

*#c. Create a dataframe to combine the city and the temp by using 'data.frame()'. Whatthe R code and its*

```
city_temp <- data.frame(tempe,place)
city_temp
```

```
##   tempe      place
## 1    42 Tuguegarao City
## 2    39      Manila
## 3    34   Iloilo City
## 4    34     Tacloban
## 5    30   Samal Island
## 6    27     Davao City
```

*#d. Associate the dataframe you have created in 2.(c) by naming the columns using the names() function.*

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

```
##   Temperature      City
## 1          42 Tuguegarao City
## 2          39      Manila
## 3          34   Iloilo City
## 4          34     Tacloban
## 5          30   Samal Island
## 6          27     Davao City
```

*#e. Print the structure by using str() function. Describe the output.*

```
str(city_temp)
```

```
## 'data.frame':   6 obs. of  2 variables:
```

```
## $ Temperature: num 42 39 34 34 30 27
## $ City : chr "Tuguegarao City" "Manila" "Iloilo City" "Tacloban" ...
```

*# It displayed the 2 variable and 6 objects on data frame.*

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

```
row_content3 <- city_temp[3,]
row_content4 <- city_temp[4,]
```

```
row_content3
```

```
## Temperature City
## 3 34 Iloilo City
```

```
row_content4
```

```
## Temperature City
## 4 34 Tacloban
```

*#g. From the answer in d, display the city with highest temperature and the city with the lowest temperature.*

```
max_index <- max(city_temp$Temperature)
max_index
```

```
## [1] 42
```

```
min_index <- min(city_temp$Temperature)
min_index
```

```
## [1] 27
```

*#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?*

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

```
##      [,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?*

```
multiply <- matrices*2
multiply
```

```
##      [,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?*

```
row2 <- matrices[2,]
row2
```

```
## [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?*

```
display <- matrices[1:2,3:4]
display
```

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

*#e. What is the R code if you want to display only the columns in 2 and 3, row 3? What is its output?*

```
col23 <- matrices[3,2:3]
col23
```

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

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

```
col4 <- matrices[,4]
col4
```

```
## [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.*

```
dimnames(multiply) <- list( c("isa", "dalawa", "tatlo"), c("uno", "dos", "tres", "quatro"))
multiply
```

```
##      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 6 rows and 2 columns.*

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

```
##      [,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*

```
ar <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)
ar
```

```
## [1] 1 2 3 6 7 8 9 0 3 4 5 1
```

```
ra <- c(1, 2, 3, 6, 7, 8, 9, 0, 3, 4, 5, 1)
ra
```

```
## [1] 1 2 3 6 7 8 9 0 3 4 5 1
```

*#a. Create an array for the above numeric values. Each value will be repeated twice, What will be the R code for this?*

```
ay <- array(c(ar,ra),dim=c(2,4,3))
ay
```

```
## , , 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?*

*# 3 dimesions*

*#c. Name the rows as lowercase letters and columns as uppercase letters starting from the A. The array*

```
column.names <- c("A","B","C","D")
```

```
row.names <- c("a","b")
```

```
matrix.names <- c("1st-Dimensional Array","2nd-Dimensional Array","3rd-Dimensional Array")
```

```
ay <- array(c(ar,ra),dim = c(2,4,3),dimnames = list(row.names,column.names,
matrix.names))
ay
```

```
## , , 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
```