

```
#USING VECTORS
```

```
uppercase <- LETTERS[1:11]
uppercase
```

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

```
oddletters<-LETTERS[seq(1,25,2)]
oddletters
```

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

```
letvowels<-c(letters[(letters %in% c("a","e","i","o","u"))], LETTERS [(LETTERS %in% c("A","E","I","O",
letvowels
```

```
## [1] "a" "e" "i" "o" "u" "A" "E" "I" "O" "U"
```

```
last5lowercase<-tail(letters,5)
last5lowercase
```

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

```
btwnlowercase<-letters[15:24]
btwnlowercase
```

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

```
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"
```

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

```
## [1] 42 39 34 34 30 27
```

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

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

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

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

```
str(df)
```

```
## 'data.frame': 6 obs. of 2 variables:
## $ City : chr "Tuguegarao City" "Manila" "Iloilo City" "Tacloban" ...
## $ Temperature: num 42 39 34 34 30 27
```

```
df[3:4,]
```

```
##           City Temperature
## 3 Iloilo City           34
## 4  Tacloban           34
```

```
highest.temp.city<-df$City[which.max(df$Temperature)]
highest.temp.city
```

```
## [1] "Tuguegarao City"
```

```
lowest.temp.city<-df$City[which.min(df$Temperature)]
lowest.temp.city
```

```
## [1] "Davao City"
```

```
#USING MATRIX
```

```
matrix1<-matrix(c(1:8,11:14),ncol=4,nrow=3)
matrix1
```

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

```
#it will display 4 columns and 3 rows with the numbers from 1 to 14
```

```
matrix(c(1:8,11:14),ncol=4,nrow=3) * 2
```

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

```
#it doubles the value of the numbers from 1 to 14
```

```
matrix(c(1:8,11:14),ncol=4,nrow=3)[2,]
```

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

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

```
matrix(c(1:8,11:14),ncol=4,nrow=3)[1:2,c(3,4)]
```

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

```
matrix(c(1:8,11:14),ncol=4,nrow=3)[3, c(2,3)]
```

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

```
matrix(c(1:8,11:14),ncol=4,nrow=3)[,4]
```

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

```
matrix2<-matrix(c(1:8,11:14),ncol=4,nrow=3) * 2
rownames(matrix2) <- c("isa", "dalawa", "tatlo")
```

```
colnames(matrix2) <- c("uno", "dos", "tres", "quatro")
matrix2
```

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

```
dim(matrix1) <- c(6,2)
matrix1
```

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

```
#USING ARRAYS
```

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

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

```
threeDarray <- array(numbers, dim = c(2, 4, 3))
threeDarray
```

```
## , , 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
```

```
#My array has 3 dimensions.
```

```
dimnames(threeDarray) <- list(c("a","b"), LETTERS[1:4], c("1st-Dimensional Array", "2nd-Dimensional Array"))
threeDarray
```

```
## , , 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
```