

**R - Notebook [LAB EXPERIMENTS DEMONSTRATION] -----Prepared by - Asst. Prof. Ashwini Mathur(CSSP)- Jain University**

**Following Tasks to Perform:**

1. Explore assignment operator.
2. Create vectors using `c()`, `seq()`, `rep()`, colon operator.
3. Create different matrices using `matrix()` operator and explore its rows, columns, and diagonals.
4. Perform different basic operation of matrices on above created matrices.
5. Create single and multidimensional arrays with `array()` command.
6. Explore `length()`, `dim()`, `ncol()`, `nrow()` operators on above matrices and arrays.
7. Explore commands for Selecting and extracting elements from above matrices and arrays.
8. Explore logical operators from R programming language.
9. Remove elements from selected positions from a considered matrix.

**Question -1: Explore Assignments Operators**

```
In [4]: x = 20 #assigned value to the variable x
        x

        y = 30 #assigned value to the variable y
        y
```

20

30

**Question 2. Arithmetic Operators : Addition, Substraction, Multiplication and Division**

```
In [5]: a = x+y # addition operator
a

b = x-y # subtraction operator
b

c = x*y # Multiplication operator
c

d = x/y # Division operator
d
```

50

-10

600

0.6666666666666667

**Create vectors using c(), seq(), rep(), colon operator**

```
In [8]: x = c(100,200,300,400,500) #creation of Vector
x #Print x
y = seq(1,10, by=0.5)
y #print y
z = 10:20
z #Print z
q = rep(50,10) #repeat(10 number 10 times)
q #
```

100 200 300 400 500

1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 6.5 7 7.5 8 8.5 9 9.5 10

10 11 12 13 14 15 16 17 18 19 20

50 50 50 50 50 50 50 50 50 50

**Create different matrices using matrix() operator and explore its rows, columns, and diagonals.**

```
In [9]: x = matrix(1:16, nrow = 4, ncol = 4)    # Create a matrix of dimension 4x4  
x
```

A matrix: 4 × 4

of type int

```
 1  5  9 13  
 2  6 10 14  
 3  7 11 15  
 4  8 12 16
```

```
In [13]: y = matrix(1:9, nrow=3, byrow=TRUE)   # fill matrix row-wise  
y
```

A matrix:

3 × 3 of

type int

```
1 2 3  
4 5 6  
7 8 9
```

```
In [14]: z = matrix(1:9, nrow=3, byrow=FALSE)  # fill matrix column-wise  
z
```

A matrix:

3 × 3 of

type int

```
1 4 7  
2 5 8  
3 6 9
```

**Perform different basic operation of matrices on above created matrices**

```
In [16]: x = cbind(c(1,2,3),c(4,5,6))
x
dim(x)
```

A  
matrix:  
3 × 2  
of type  
dbl

```
1 4
2 5
3 6

3 2
```

```
In [17]: y = rbind(c(1,2,3),c(4,5,6))
y
dim(y)
```

A matrix:  
2 × 3 of  
type dbl

```
1 2 3
4 5 6

2 3
```

### Create single and multidimensional arrays with array() command

```
In [18]: # Create two vectors of different lengths.
vector1 <- c(2,9,3)
vector2 <- c(10,16,17,13,11,15)

vector1 #print
vector2
```

```
2 9 3
```

```
10 16 17 13 11 15
```

```
In [25]: # Take these vectors as input to the array.
result <- array(c(vector1,vector2),dim = c(3,3,2)) #Multi-dimension
print(result)
```

```
, , 1
```

```
      [,1] [,2] [,3]
[1,]     2    10    13
[2,]     9    16    11
[3,]     3    17    15
```

```
, , 2
```

```
      [,1] [,2] [,3]
[1,]     2    10    13
[2,]     9    16    11
[3,]     3    17    15
```

```
In [ ]: result <- array(c(vector1,vector2),dim = c(3)) #Single dimension
print(result)
```

**Explore length(), dim(), ncol(), nrow() operators on above matrices and arrays.**

```
In [19]: x = matrix(1:100, nrow = 10, ncol = 10)
x

length(x) #Length of the Matrix

dim(x)    #Dimension of the given matrix
```

A matrix: 10 × 10 of type int

```
 1  11  21  31  41  51  61  71  81  91
 2  12  22  32  42  52  62  72  82  92
 3  13  23  33  43  53  63  73  83  93
 4  14  24  34  44  54  64  74  84  94
 5  15  25  35  45  55  65  75  85  95
 6  16  26  36  46  56  66  76  86  96
 7  17  27  37  47  57  67  77  87  97
 8  18  28  38  48  58  68  78  88  98
 9  19  29  39  49  59  69  79  89  99
10  20  30  40  50  60  70  80  90 100

100

10 10
```

```
In [ ]: ncol(x) #Total number of columns in given matrix
        nrow(x) #Total Number of Rows in Given Matrix
```

**Explore commands for Selecting and extracting elements from above matrices and arrays.**

```
In [21]: x = matrix(1:9, nrow = 3, ncol = 3)
x

x[1,2] #Selecting the element of First row and second column
```

A matrix:

3 × 3 of

type int

```
 1  4  7
 2  5  8
 3  6  9
```

4

```
In [22]: x[3,3] #Selecting the element of third row and third column
```

9

```
In [30]: x[,1] #Selection the 1st column
```

1 2 3

```
In [23]: x[2,] #Selecting the 3rd row
```

2 5 8

```
In [32]: y <- c(10,16,17,13,11,15)
```

```
y[4] #Selecting 4th element
```

```
y[1] #selecting the first element
```

13

10

**Explore logical operators from R programming language.**

```
In [33]: x <- c(TRUE,FALSE,0,6)  
y <- c(FALSE,TRUE,FALSE,TRUE)  
  
!x  #(Complement of x) Logical NOT
```

FALSE TRUE TRUE FALSE

```
In [34]: x&y #Element-wise Logical AND
```

FALSE FALSE FALSE TRUE

```
In [35]: x&&y #Logical AND
```

FALSE

```
In [36]: x|y # Element Wise Logical OR
```

TRUE TRUE FALSE TRUE

```
In [37]: x||y #Logical OR
```

TRUE

**Remove elements from selected positions from a considered matrix.**

```
In [38]: x = matrix(1:9, nrow = 3, ncol = 3)
x
```

A matrix:

3 × 3 of

type int

1 4 7

2 5 8

3 6 9

```
In [39]: x[, -3] #Remove 3rd Column
```

A

matrix:

3 × 2

of type

int

1 4

2 5

3 6

A matrix:

3 × 3 of

type int

1 4 7

2 5 8

3 6 9



```
In [40]: x[-3,] #Remove 3rd Row
```

A matrix:

2 × 3 of

type int

1 4 7

2 5 8