

# **List and Matrices**

## List

A **list** is a generic vector containing other objects.

## Creating Lists

For example, the following variable x is a list containing copies of three vectors n, s, b, and a numeric value 3.

```
> n = c(2, 3, 5)
> s = c("aa", "bb", "cc", "dd", "ee")
> b = c(TRUE, FALSE, TRUE, FALSE, FALSE)
> x = list(n, s, b, 3) # x contains copies of n, s, b
```

OR

```
x1 <- list("Red", "Blue", c(42,36,01), FALSE, 73.91, 128.6)
print(x1)
```

Output:

```
[[1]]
[1] "Red"
[[2]]
[1] "Blue"
[[3]]
[1] 42 36 01
[[4]]
[1] FALSE
[[5]]
[1] 73.91
[[6]]
[1] 128.6
```

## Naming List Elements

# Create a list containing a vector, a matrix and a list.

```
list_data <- list(c("Jan","Feb","Mar"), matrix(c(3,9,5,1,-2,8), nrow = 2), list("green",12.3))
```

# Give names to the elements in the list.

```
names(list_data) <- c("Month", "Matrix", "Inner list") # Show the list.  
print(list_data)
```

### Output:

```
$`Month`  
[1] "Jan" "Feb" "Mar"  
$Matrix  
[,1] [,2] [,3]  
[1,] 3 5 -2  
[2,] 9 1 8  
$Inner list  
$Inner_list [[1]]  
[1] "green"  
$Inner_list [[2]]  
[1] 12.3
```

## Accessing List Elements

# Create a list containing a vector, a matrix and a list.

```
list_data <- list(c("Jan","Feb","Mar"), matrix(c(3,9,5,1,-2,8), nrow = 2), list("green",12.3))
```

# Give names to the elements in the list.

```
names(list_data) <- c("Month", "Matrix", "Inner list")
```

# Access the first element of the list.

```
print(list_data[1])
```

# Access the third element.

```
print(list_data[3])
```

# Access the list element using the name of the element.

```
print(list_data$Matrix)
```

### Output:

```
$`Month`
```

```
[1] "Jan" "Feb" "Mar"
```

```
$Inner list
```

```
$Inner list[[1]]
```

```
[1] "green"
```

```
$Inner list[[2]]
```

```
[1] 12.3
```

```
[,1] [,2] [,3]
```

```
[1,] 3 5 -2
```

```
[2,] 9 1 8
```

## Manipulating List Elements

- We can **add, delete and update** list elements.
- We can **add and delete** elements **only at the end of a list**.
- we can update any **element**.

### As in above example

**# Add element at the end of the list.**

```
list_data[4] <- "New element"  
print(list_data[4])
```

**# Remove the last element.**

```
list_data[4] <- NULL
```

**# Print the 4th Element.**

```
print(list_data[4])
```

**# Update the 3rd Element.**

```
list_data[3] <- "updated element"  
print(list_data[3])
```

### Output:

```
[[1]]
```

```
[1] "New element"
```

```
$<NA> NULL
```

```
$`Inner list`
```

```
[1] "updated element"
```

## Merging Lists

# Create two lists.

```
list1 <- list(1,2,3)
```

```
list2 <- list("Sun","Mon","Tue")
```

# Merge the two lists.

```
merged.list <- c(list1,list2)
```

# Print the merged list.

```
print(merged.list)
```

**Output:**

```
[[1]]
```

```
[1] 1
```

```
[[2]]
```

```
[1] 2
```

```
[[3]]
```

```
[1] 3
```

```
[[4]]
```

```
[1] "Sun"
```

```
[[5]]
```

```
[1] "Mon"
```

```
[[6]]
```

```
[1] "Tue"
```

## Converting List to Vector

**unlist()**: It takes the list as input and produces a vector.

# Create and print lists.

```
list1 <- list(1:5)
print(list1)
```

```
list2 <- list(10:14)
print(list2)
```

# Convert the lists to vectors.

```
v1 <- unlist(list1)
v2 <- unlist(list2)
print(v1)
print(v2)
```

# Now add the vectors

```
result <- v1+v2
print(result)
```

**Output:**

**(Print list1)**

```
[[1]]
[1] 1 2 3 4 5
```

**(Print list2)**

```
[[1]]
[1] 10 11 12 13 14
```

**(Print v1)**

```
[1] 1 2 3 4 5
```

**(Print v2)**

```
[1] 10 11 12 13 14
```

**(Print result)**

```
[1] 11 13 15 17 19
```

# The *matrix* command

- A **matrix** is a 2-D array
- There is a **fast** method of creating a matrix

## Syntax

The basic syntax for creating a matrix in R is –

**matrix(data, nrow, ncol, byrow, dimnames)**

**Following is the description of the parameters used –**

**data** is the input vector which becomes the data elements of the matrix.

**nrow** is the number of rows to be created.

**ncol** is the number of columns to be created.

**byrow** is a logical clue. If TRUE then the input vector elements are arranged by row.

**dimname** is the names assigned to the rows and columns.

### Example: (Create a matrix)

# Elements are arranged sequentially by row.

```
M <- matrix(c(3:14), nrow = 4, byrow = TRUE)
print(M)
```

# Elements are arranged sequentially by column.

```
N <- matrix(c(3:14), nrow = 4, byrow = FALSE)
print(N)
```

# Define the column and row names.

```
rownames = c("row1", "row2", "row3", "row4")
colnames = c("col1", "col2", "col3")
```

```
P <- matrix(c(3:14), nrow = 4, byrow = TRUE, dimnames = list(rownames, colnames))
print(P)
```

#### **Print(M)**

```
[,1] [,2] [,3]
[1,] 3 4 5
[2,] 6 7 8
[3,] 9 10 11
[4,] 12 13 14
```

#### **print(N)**

```
[,1] [,2] [,3]
[1,] 3 7 11
[2,] 4 8 12
[3,] 5 9 13
[4,] 6 10 14
```

#### **print(P)**

```
col1 col2 col3
row1 3 4 5
row2 6 7 8
row3 9 10 11
row4 12 13 14
```

## Accessing Elements of a Matrix

- by using the column and row index

# Access the element at 3rd column and 1st row. (Byrow=true)

```
print(P[1,3])
```

# Access the element at 2nd column and 4th row.

```
print(P[4,2])
```

# Access only the 2nd row.

```
print(P[2,])
```

# Access only the 3rd column.

```
print(P[:,3])
```

Output:

```
[1] 5
```

```
[1] 13
```

```
col1 col2 col3
```

```
6 7 8
```

```
row1 row2 row3 row4
```

```
5 8 11 14
```

# Matrix Addition & Subtraction

# Create two 2x3 matrices.

```
matrix1 <- matrix(c(3, 9, -1, 4, 2, 6), nrow = 2)  
print(matrix1)
```

```
matrix2 <- matrix(c(5, 2, 0, 9, 3, 4), nrow = 2)  
print(matrix2)
```

# Add the matrices.

```
result <- matrix1 + matrix2  
cat("Result of addition","\n")  
print(result)
```

# Subtract the matrices

```
result <- matrix1 - matrix2  
cat("Result of subtraction","\n")  
print(result)
```

**Output:**

```
[,1] [,2] [,3]  
[1,] 3 -1 2  
[2,] 9 4 6
```

```
[,1] [,2] [,3]  
[1,] 5 0 3  
[2,] 2 9 4
```

Result of addition

```
[,1] [,2] [,3]  
[1,] 8 -1 5  
[2,] 11 13 10
```

Result of subtraction

```
[,1] [,2] [,3]  
[1,] -2 -1 -1  
[2,] 7 -5 2
```

## Matrix Multiplicaton & Division

# Create two 2x3 matrices.

```
matrix1 <- matrix(c(3, 9, -1, 4, 2, 6), nrow = 2)
print(matrix1)
```

```
matrix2 <- matrix(c(5, 2, 0, 9, 3, 4), nrow = 2)
print(matrix2)
```

# Add the matrices.

```
result <- matrix1 * matrix2
cat("Result of addition","\n")
print(result)
```

# Subtract the matrices

```
result <- matrix1 / matrix2
cat("Result of subtraction","\n")
print(result)
```

**Output:**

```
[,1] [,2] [,3]
[1,] 3 -1 2
[2,] 9 4 6
```

```
[,1] [,2] [,3]
[1,] 5 0 3
[2,] 2 9 4
```

Result of multiplication

```
[,1] [,2] [,3]
[1,] 15 0 6
[2,] 18 36 24
```

Result of division

```
[,1] [,2] [,3]
[1,] 0.6 -Inf 0.6666667
[2,] 4.5 0.4444444 1.5000000
```

Note: Element by element multiplication and division

Solution(%\*%) and multiplication condition must satisfied

## Example:

```
matrix1 <- matrix(c(3, 9, -1, 4, 2, 6, 5, 4, 3), nrow = 3)
print(matrix1)
```

```
res= matrix1 * matrix1
```

```
res1=matrix1 %*% matrix1
```

```
print(res)
```

```
print(res1)
```

### Output:

```
>print(matrix1)
```

```
      [,1] [,2] [,3]
[1,]    3    4    5
[2,]    9    2    4
[3,]   -1    6    3
```

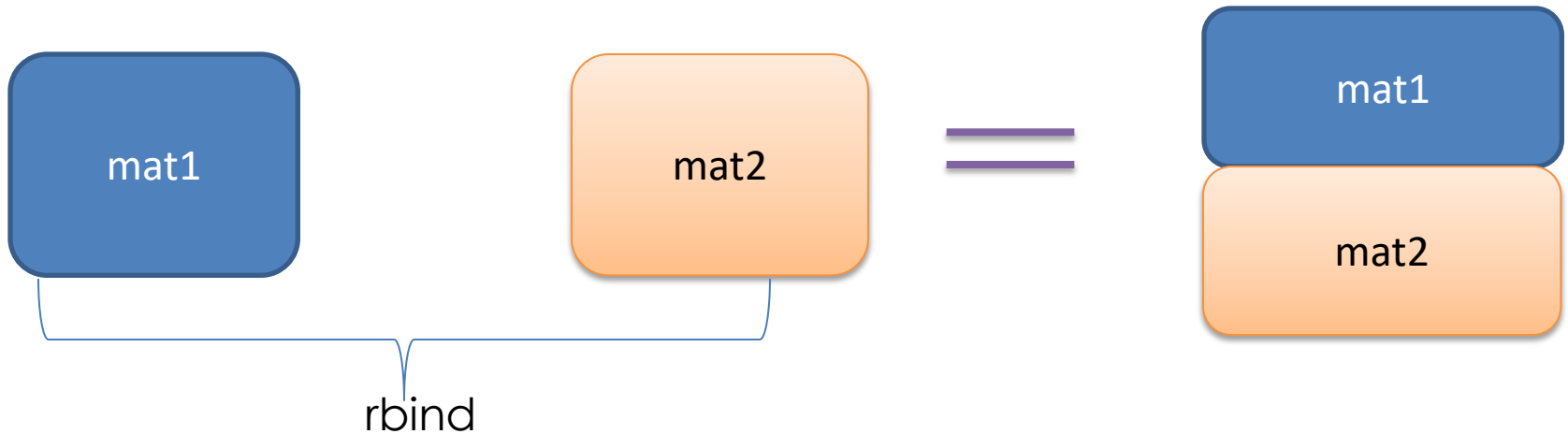
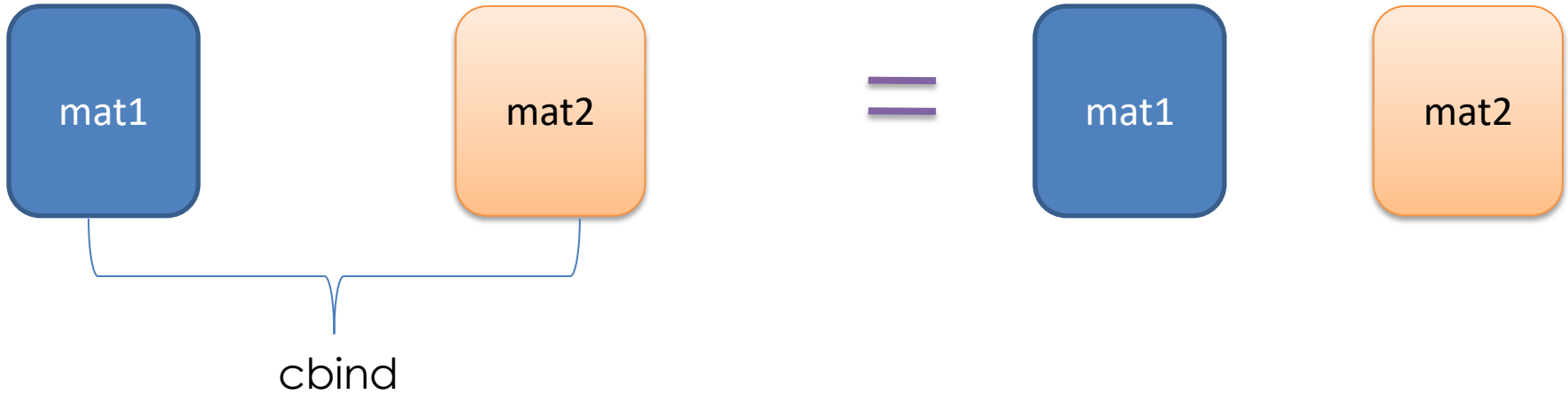
```
> print(res)
```

```
      [,1] [,2] [,3]
[1,]    9   16   25
[2,]   81    4   16
[3,]    1   36    9
```

```
> print(res1)
```

```
      [,1] [,2] [,3]
[1,]   40   50   46
[2,]   41   64   65
[3,]   48   26   28
```

# cbind and rbind



### Example:(cbind and rbind)

#### # Create two matrices

```
m1 <- matrix(c(3, 9, -1, 4, 2, 6), nrow = 2)
m2 <- matrix(c(1, 2, 4, 3, 5, 7), nrow=2)
```

#### # Print m1 and m2

```
print(m1)
print(m2)
```

#### # apply cbind and rbind

```
mcb=cbind(m1[1:2], m2[1:2])
mrb=rbind(m1[1], m2[1])
```

#### # print mca and mrb

```
print(mcb)
print(mrb)
```

#### #Print mcb1

```
mcb1=cbind(m1[1:2,1:2], m2[1:2,1:2])
```

### Output:

```
> print(m1)
```

```
      [,1] [,2] [,3]
[1,]    3   -1    2
[2,]    9    4    6
```

```
> print(m2)
```

```
      [,1] [,2] [,3]
[1,]    1    4    5
[2,]    2    3    7
```

```
> print(mcb)
```

```
      [,1] [,2]
[1,]    3    1
[2,]    9    2
```

```
> print(mrb)
```

```
      [,1]
[1,]    3
[2,]    1
```

```
> mcb1
```

```
      [,1] [,2] [,3] [,4]
[1,]    3   -1    1    4
[2,]    9    4    2    3
```