Arrays in R

Array:

- An array is a data structure that can hold multi-dimensional data
- In arrays, data is stored in the form of matrices, rows, and columns.
- An array is created using the **array()** function.

R Array Syntax

Array_NAME <- array(data, dim = (row_Size, column_Size, matrices, dimnames)

- **data –** Data is an input vector that is given to the array.
- **matrices –** Array in R consists of multi-dimensional matrices.
- **row_Size –** row_Size describes the number of row elements that an array can store.
- **column_Size –** Number of column elements that can be stored in an array.
- **dimnames –** Used to change the default names of rows and columns to the user's preference.

How to create?

There are only two steps to create a matrix which are as follows

- In the first step, we will create two vectors of different lengths.
- Once our vectors are created, we take these vectors as inputs to the array.

Example:

```
vector1 <- c(2,9,3)
vector2 <- c(10,16,17,13,11,15)
# Take these vectors as input to the array.
result <- array(c(vector1,vector2),dim = c(3,3,2))
print(result)
```

Output

```
> vector1 <- c(2,9,3)

> vector2 <- c(10,16,17,13,11,15)

> result <- array(c(vector1,vector2),dim = c(3,3,2))

> 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
```

id1	2L
id2	5L
index	4L
result	num [1:3, 1:3, 1:2] 2 9 3…
val	10
values	int [1:10] 1 2 3 4 5 6 7 …
vector	num [1:6] 10 16 17 13 11 …
vector1	num [1:3] 2 9 3
vector2	num [1:6] 10 16 17 13 11 …
x	num [1:5] 1 2 3 4 5

Different Operations on Rows and Columns

- 1. Naming Columns And Rows

Create two vectors of different lengths.

- **Create two vectors of different lengths.**

```
vector1 <- c(2,9,6)
vector2 <- c(10,15,13,16,11,12)
column.names <- c("COL1","COL2","COL3")
row.names <- c("ROW1","ROW2","ROW3")
matrix.names <- c("Matrix1","Matrix2")
#Taking the vectors as input to the array
result <- array(c(vector1,vector2),dim = c(3,3,2),dimnames = list(row.names,column.names,matrix.names))
```

- **print(result)**

Output

```
> result <- array(c(vector1,vector2),dim = c(3,3,2),dimnames = list(row.names,column.names, matrix.names))

> print(result)
, , Matrix1

     COL1 COL2 COL3
ROW1    2   10   16
ROW2    9   15   11
ROW3    6   13   12

, , Matrix2

     COL1 COL2 COL3
ROW1    2   10   16
ROW2    9   15   11
ROW3    6   13   12

>
```

Files | Plots | Packages | Help | Viewer

2. Accessing R Array Elements

 # Print the third row of the second matrix of the array.

- print(result[3,,2])

 # Print the element in the 1st row and 3rd column of the 1st matrix.

- print(result[1,3,1])

 # Print the 2nd Matrix.

- print(result[,,2])

OUTPUT

```
> print(result[3,,2])
COL1 COL2 COL3
   6   13   12

> print(result[1,3,1])
[1] 16

> print(result[,,2])
     COL1 COL2 COL3
ROW1    2   10   16
ROW2    9   15   11
ROW3    6   13   12
>
```

Files | Plots | Packages | Help | Viewer

3. Manipulating R Array Elements

 # Create two vectors of different lengths.

- vector1 <- c(1,2,3)
- vector2 <- c(3,4,5,6,7,8)

 # Take these vectors as input to the array.

- array1 <- array(c(vector1,vector2),dim = c(3,3,2))

 # Create two vectors of different lengths.

- vector3 <- c(3,2,1)
- vector4 <- c(8,7,6,5,4,3)
- array2 <- array(c(vector1,vector2),dim = c(3,3,2))

 # create matrices from these arrays.

- matrix1 <- array1[,,2]
- matrix2 <- array2[,,2]

 # Add the matrices.

- result <- matrix1+matrix2
- print(result)

output

```
> matrix1 <- array1[,,2]

> matrix2 <- array2[,,2]

> result <- matrix1+matrix2

> print(result)
     [,1] [,2] [,3]
[1,]    2    6   12
[2,]    4    8   14
[3,]    6   10   16
>
```

vector3	num [1:3] 3 2 1
vector4	num [1:6] 8 7 6 5 4 3

Files | Plots | Packages | Help | Viewer

Thank you