R Data Frame

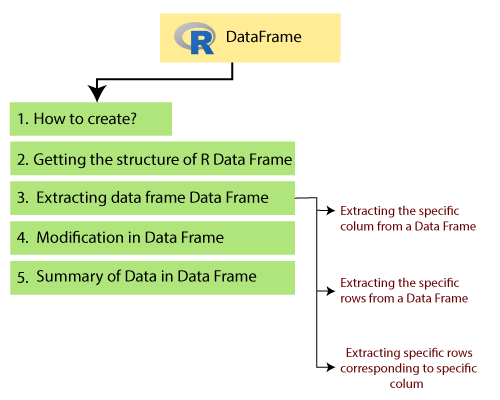
A data frame is a two-dimensional array-like structure or a table in which a column contains values of one variable, and rows contains one set of values from each column. A data frame is a special case of the list in which each component has equal length.

A data frame is used to store data table and the vectors which are present in the form of a list in a data frame, are of equal length.

In a simple way, it is a list of equal length vectors. A matrix can contain one type of data, but a data frame can contain different data types such as numeric, character, factor, etc.

There are following characteristics of a data frame.

* The columns name should be non-empty.
* The rows name should be unique.
* The data which is stored in a data frame can be a factor, numeric, or character type.
* Each column contains the same number of data items.



How to create Data Frame

In R, the data frames are created with the help of frame() function of data. This function contains the vectors of any type such as numeric, character, or integer. In below example, we create a data frame that contains employee id (integer vector), employee name(character vector), salary(numeric vector), and starting date(Date vector).

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,915.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. # Printing the data frame.
12. print(emp.data)

**Output**

employee\_idemployee\_namesalstarting\_date

1 1 Shubham623.30 2012-01-01

2 2 Arpita915.20 2013-09-23

3 3 Nishka611.00 2014-11-15

4 4 Gunjan729.00 2014-05-11

5 5 Sumit843.25 2015-03-27

Getting the structure of R Data Frame

In R, we can find the structure of our data frame. R provides an in-build function called str() which returns the data with its complete structure. In below example, we have created a frame using a vector of different data type and extracted the structure of it.

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. # Printing the structure of data frame.
12. str(emp.data)

**Output**

'data.frame': 5 obs. of 4 variables:

$ employee\_id : int 1 2 3 4 5

$ employee\_name: chr "Shubham" "Arpita" "Nishka" "Gunjan" ...

$ sal : num 623 515 611 729 843

$ starting\_date: Date, format: "2012-01-01" "2013-09-23" ...

Extracting data from Data Frame

The data of the data frame is very crucial for us. To manipulate the data of the data frame, it is essential to extract it from the data frame. We can extract the data in three ways which are as follows:

1. We can extract the specific columns from a data frame using the column name.
2. We can extract the specific rows also from a data frame.
3. We can extract the specific rows corresponding to specific columns.

Let's see an example of each one to understand how data is extracted from the data frame with the help these ways.

Extracting the specific columns from a data frame

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name= c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. # Extracting specific columns from a data frame
12. final **<-** data.frame(emp.data$employee\_id,emp.data$sal)
13. print(final)

**Output**

emp.data.employee\_idemp.data.sal

1 1 623.30

2 2 515.20

3 3 611.00

4 4 729.00

5 5 843.25

Extracting the specific rows from a data frame

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. # Extracting first row from a data frame
12. final **<-** emp.data[1,]
13. print(final)

16. # Extracting last two row from a data frame
17. final **<-** emp.data[4:5,]
18. print(final)

**Output**

employee\_id employee\_name sal starting\_date

1 1 Shubham 623.3 2012-01-01

employee\_id employee\_name sal starting\_date

4 4 Gunjan 729.00 2014-05-11

5 5 Sumit 843.25 2015-03-27

Extracting specific rows corresponding to specific columns

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. # Extracting 2nd and 3rd row corresponding to the 1st and 4th column
12. final **<-** emp.data[c(2,3),c(1,4)]
13. print(final)

**Output**

employee\_id starting\_date

2 2 2013-09-23

3 3 2014-11-15

Modification in Data Frame

R allows us to do modification in our data frame. Like matrices modification, we can modify our data frame through re-assignment. We cannot only add rows and columns, but also we can delete them. The data frame is expanded by adding rows and columns.

We can

1. Add a column by adding a column vector with the help of a new column name using cbind() function.
2. Add rows by adding new rows in the same structure as the existing data frame and using rbind() function
3. Delete the columns by assigning a NULL value to them.
4. Delete the rows by re-assignment to them.

Let's see an example to understand how rbind() function works and how the modification is done in our data frame.

**Example: Adding rows and columns**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. print(emp.data)
13. #Adding row in the data frame
14. x **<-** list(6,"Vaishali",547,"2015-09-01")
15. rbind(emp.data,x)
17. #Adding column in the data frame
18. y **<-** c("Moradabad","Lucknow","Etah","Sambhal","Khurja")
19. cbind(emp.data,Address=y)

**Output**

employee\_id employee\_name sal starting\_date

1 1 Shubham 623.30 2012-01-01

2 2 Arpita 515.20 2013-09-23

3 3 Nishka 611.00 2014-11-15

4 4 Gunjan 729.00 2014-05-11

5 5 Sumit 843.25 2015-03-27

employee\_id employee\_name sal starting\_date

1 1 Shubham 623.30 2012-01-01

2 2 Arpita 515.20 2013-09-23

3 3 Nishka 611.00 2014-11-15

4 4 Gunjan 729.00 2014-05-11

5 5 Sumit 843.25 2015-03-27

6 6 Vaishali 547.00 2015-09-01

employee\_id employee\_name sal starting\_date Address

1 1 Shubham 623.30 2012-01-01 Moradabad

2 2 Arpita 515.20 2013-09-23 Lucknow

3 3 Nishka 611.00 2014-11-15 Etah

4 4 Gunjan 729.00 2014-05-11 Sambhal

5 5 Sumit 843.25 2015-03-27 Khurja

**Example: Delete rows and columns**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. print(emp.data)
13. #Delete rows from data frame
14. emp.data**<-emp.data**[-1,]
15. print(emp.data)
17. #Delete column from the data frame
18. emp.data$starting\_date**<-NULL**
19. print(emp.data)

**Output**

employee\_idemployee\_namesalstarting\_date

1 1 Shubham623.30 2012-01-01

2 2 Arpita515.20 2013-09-23

3 3 Nishka611.00 2014-11-15

4 4 Gunjan729.00 2014-05-11

5 5 Sumit843.25 2015-03-27

employee\_idemployee\_namesalstarting\_date

2 2 Arpita515.20 2013-09-23

3 3 Nishka611.00 2014-11-15

4 4 Gunjan729.00 2014-05-11

5 5 Sumit843.25 2015-03-27

employee\_idemployee\_namesal

1 1 Shubham623.30

2 2 Arpita515.20

3 3 Nishka611.00

4 4 Gunjan729.00

5 5 Sumit843.25

Summary of data in Data Frames

In some cases, it is required to find the statistical summary and nature of the data in the data frame. R provides the summary() function to extract the statistical summary and nature of the data. This function takes the data frame as a parameter and returns the statistical information of the data. Let?s see an example to understand how this function is used in R:

**Example**

1. # Creating the data frame.
2. emp.data**<-** data.frame(
3. employee\_id = c (1:5),
4. employee\_name = c("Shubham","Arpita","Nishka","Gunjan","Sumit"),
5. sal = c(623.3,515.2,611.0,729.0,843.25),
7. starting\_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
8. "2015-03-27")),
9. stringsAsFactors = FALSE
10. )
11. print(emp.data)
13. #Printing the summary
14. print(summary(emp.data))

**Output**

employee\_idemployee\_namesalstarting\_date

1 1 Shubham623.30 2012-01-01

2 2 Arpita515.20 2013-09-23

3 3 Nishka611.00 2014-11-15

4 4 Gunjan729.00 2014-05-11

5 5 Sumit843.25 2015-03-27

employee\_idemployee\_namesalstarting\_date

Min. :1 Length:5 Min. :515.2 Min. :2012-01-01

1st Qu.:2 Class :character 1st Qu.:611.0 1st Qu.:2013-09-23

Median :3 Mode :character Median :623.3 Median :2014-05-11

Mean :3 Mean :664.4 Mean :2014-01-14

3rd Qu.:4 3rd Qu.:729.0 3rd Qu.:2014-11-15

Max. :5 Max. :843.2 Max. :2015-03-27