

## Experiment 7

Create a data frame with 10 observations and 3 variables and add new rows and columns to it using 'rbind' and 'cbind' function.

### Explanation:

#### Matrices and Dataframes

There are a number of ways to create a matrix and dataframe objects in R. The most common functions are described in the following table. Because matrices and dataframes are just combinations of vectors, each function takes one or more vectors as inputs, and returns a matrix or a dataframe.

<b>cbind(a, b, c)</b>	Combine vectors as columns in a matrix
<b>rbind(a, b, c)</b>	Combine vectors as rows in a matrix
<b>matrix(x, nrow, ncol, byrow)</b>	Create a matrix from a vector x
<b>data.frame()</b>	Create a dataframe from named column

#### cbind and rbind functions:

The name of the **cbind** function in R stands for column-bind and **rbind** stands for row-bind. These functions are used to combine vectors, matrices and/or data frames by columns / rows. The syntax of these functions is

```
cbind(my_data, new_column)
rbind (my_data, new_row)
```

#### Example:

Let us create three vectors of length 4, then we'll combine them into one matrix. The cbind() function will combine the vectors as columns in the final matrix, while the rbind() function will combine them as rows.

```
> cmatrix <- cbind (a,b,c)
> cmatrix
  a b c
[1,] 1 5 9
[2,] 2 6 10
[3,] 3 7 11
```

```
[4,] 4 8 12
> rmatrix <- rbind (a,b,c)
> rmatrix
  [,1] [,2] [,3] [,4]
a   1   2   3   4
b   5   6   7   8
c   9  10  11  12
> cmatrix <- cbind (a,b,c,deparse.level=0)
> cmatrix
  [,1] [,2] [,3]
[1,]  1   5   9
[2,]  2   6  10
[3,]  3   7  11
[4,]  4   8  12
> rmatrix <- rbind (a,b,c,deparse.level=0)
> rmatrix
  [,1] [,2] [,3] [,4]
[1,]  1   2   3   4
[2,]  5   6   7   8
[3,]  9  10  11  12
```

**Solution:**

1. Create a new data frame with 5 rows with emp\_id, emp\_name, salary and start\_date as columns

```
> emp.data <- data.frame(
  emp_id = c (1:5),
  emp_name = c("Rick","Dan","Michelle","Ryan","Gary"),
  salary = c(623.3,515.2,611.0,729.0,843.25),
  start_date = as.Date(c("2012-01-01", "2013-09-23", "2014-11-15", "2014-05-11",
    "2015-03-27")),
  stringsAsFactors = FALSE
)
```

```
> emp.data
  emp_id emp_name salary start_date
1     1    Rick 623.30 2012-01-01
2     2     Dan 515.20 2013-09-23
3     3 Michelle 611.00 2014-11-15
4     4     Ryan 729.00 2014-05-11
5     5     Gary 843.25 2015-03-27
```

2. Create a new vector as dept with department names of the employees

```
> dept <- c("IT","Operations","IT","HR","Finance")
```

3. Add this dept vector as a new column to data frame

```
> emp.data <- cbind(emp.data, dept)
> emp.data
  emp_id emp_name salary start_date dept
1     1    Rick 623.30 2012-01-01   IT
2     2     Dan 515.20 2013-09-23 Operations
3     3 Michelle 611.00 2014-11-15   IT
4     4     Ryan 729.00 2014-05-11   HR
5     5     Gary 843.25 2015-03-27 Finance
```

4. Create a new data frame with 3 more records

```
> emp.newdata <- data.frame(
  emp_id = c(6:8),
  emp_name = c("Rasmi","Pranab","Tusar"),
  salary = c(578.0,722.5,632.8),
  start_date = as.Date(c("2013-05-21","2013-07-30","2014-06-17")),
  dept = c("IT", "Operations", "Finance"),
  stringsAsFactors = FALSE
)
```

5. Add these new records to the existing data frame emp.data

```
> emp.finaldata <- rbind(emp.data, emp.newdata)
```

```
> print (emp.finaldata)
  emp_id emp_name salary start_date dept
1     1    Rick 623.30 2012-01-01   IT
2     2     Dan 515.20 2013-09-23 Operations
3     3  Michelle 611.00 2014-11-15   IT
4     4    Ryan 729.00 2014-05-11   HR
5     5    Gary 843.25 2015-03-27 Finance
6     6   Rasmi 578.00 2013-05-21   IT
7     7  Pranab 722.50 2013-07-30 Operations
8     8   Tusar 632.80 2014-06-17 Finance
```