

introduction to Sorting in R

A mechanism provided by R programming through which elements of a vector can be arranged in a particular order, usually facilitated by but not just limited to the `order()` function that assists in sorting the elements either in ascending or descending order, as required, with the normal use of `order()` function sorting the result in ascending order(), otherwise sorting the result in descending order, is known as sorting in R.

Sorting Performed in R

There are multiple ways by which data can be sorted in R. It's up to the **data Analyst to consider** the most suitable method based upon the structure of the data. This is because R language has multiple functions and ways to sort the data, such as `sort()`, `order()`, and `dplyr::arrange()` package.

Things to keep in mind before sorting the data.

1. Order in which data needs to be sorted ascending or descending.
2. Multiple columns sorting criteria.

3. Accounting missing and duplicate values during sorting. It's up to the analyst to decide what must be done with the missing and duplicate values. Before removing or replacing null values, the overall impact on the data should be considered.

Sort() function in R

Sort function in R is used to sort a vector. By default, the value is organized in ascending order. Let's take an example of the mark's column of all the students in a classroom.

The syntax to sort the vector is

```
"sort (x, decreasing = FALSE) "
```

Here x refers to the vector and decreasing has to be replaced to TRUE when the sorting has to be done in descending order. The sort function is used in arranging numeric or character vector in the desired order. The major limitation of the sort function is that it cannot be used to sort a data frame. To overcome this limitation Order () function is used.

A basic sorting example using sort() function

```
set.seed(1)
```

```
x <- sample(1:100,10)
```

```
x
```

Output

```
[1] 68 39 1 34 87 43 14 82 59 51
```

```
sort(x)
```

Output

```
[1] 1 14 34 39 43 51 59 68 82 87
```

Sorting data frames can be accomplished with the help of `order()` function.

Variables can be easily sorted in either ascending or descending order

however, the `order` function will sort the variable in ascending by default.

```
> df <- data.frame("Serial_number" = 1:5, "Age" = c(20,
```

```
21, 17, 18, 19), "Name" = c("Johnny", "Dorian", "Linda",
```

```
"Cathy", "Rick"))
```

```
>
```

```
> # Sort by age ascending order
```

```
> newdataAsc <- df[order(df$Age),] > newdataAsc
```

```
  Serial_number Age  Name  
3             3  17  Linda  
4             4  18  Cathy  
5             5  19   Rick  
1             1  20 Johnny  
2             2  21 Dorian  
>
```

```
# sorting is descending order
```

```
> newdataDsc> newdataDsc <- df[order(-df$Age),] >
```

```
newdataAsc
```

```
> newdataDsc  
  Serial_number Age  Name  
2             2  21 Dorian  
1             1  20 Johnny  
5             5  19   Rick  
4             4  18  Cathy  
3             3  17  Linda
```