Experiment 6: Demonstrate List in R

Aim: To create a list, access and manipulate the elements of a List.

- Vectors, matrices, and arrays are that each of these types of objects may only contain one type of data.
- **For example**, a vector may contain all numeric data or all character data.
- A list is a special type of object that can contain **data of multiple types**.
- Lists are characterized by the fact that their elements do not need to be of the same object type.

Experiment 6: Demonstrate List in R

Aim: To create a list, access and manipulate the elements of a List.

- Vectors, matrices, and arrays are that each of these types of objects may only contain one type of data.

- **For example**, a vector may contain all numeric data or all character data.

- A list is a special type of object that can contain **data of multiple types**.

- **Lists are characterized by** the fact that their elements do not need to be of the same object type.

Experiment 6: Demonstrate List in R

- A list can contain all numeric or characters or a mix of the two or data frames or recursively other lists.

- Lists can contain elements of different types so that the list elements may have different **modes**.

- **Mode :** Every object has a mode.

  - The mode indicates how the object is stored in memory : as a

    ✓ Number,

    ✓ Character string ,

    ✓ List of points to the objects,

    ✓ Functions etc.

- List is created using **list()** function

Experiment 6: Demonstrate List in R

Object	Example	Mode
Number	1,2,3,45,1.234	numeric
Vector of Numbers	C(2,3,4,5)	numeric
Character String	“India”	character
Vector of Character Strings	C(“India”,”USA”)	character
Factor	factor(c(“UP”,”MP”))	Numeric
List	list(“India”,”USA”)	list
Data Frame	data.frame(x=1:2,y=c(“India”,”USA”))	list
Function	print	function

Experiment 6: Demonstrate List in R

Mode: mode function gives us such information.

Syntax: mode()

Example:

```
 >mode(2.34)
  [1] "numeric“
 > mode(c(1,2,3,4))
[1] "numeric“

> mode("India")
[1] "character“

> mode(c("India","USA"))
[1] "character"
```

Experiment 6: Demonstrate List in R

Example:

```
> mode(factor(c("UP","MP")))
[1] "numeric“

> mode(list("India","USA"))
[1] "list“

> mode(data.frame(x=1:2,y=c("India","USA")))
[1] "list“

> mode(print)
[1] "function"
```

Experiment 6: Demonstrate List in R

Difference between Data Frame and List in R:

A **Data Frame** is the term in R for a spreadsheet style of data: a grid of rows and columns.

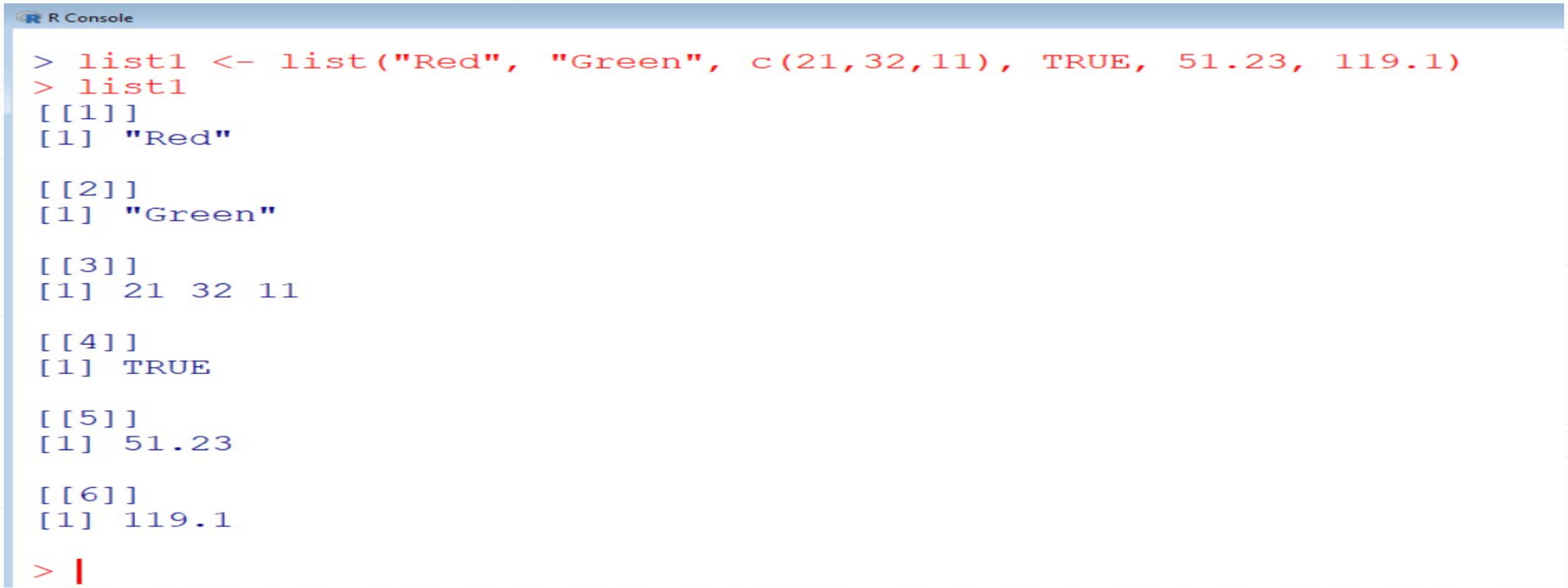
Lists are the R objects which contain elements of different types like – numbers, strings, vectors and another list inside it.

Experiment 6: Demonstrate List in R

Creating a list:

#Create a list containing strings, numbers, vectors and a logical values:

>list1 <- list("Red", "Green", c(21,32,11), TRUE, 51.23, 119.1)
>list1



```
> list1 <- list("Red", "Green", c(21,32,11), TRUE, 51.23, 119.1)
> list1
[[1]]
[1] "Red"

[[2]]
[1] "Green"

[[3]]
[1] 21 32 11

[[4]]
[1] TRUE

[[5]]
[1] 51.23

[[6]]
[1] 119.1

> 
```

Experiment 6: Demonstrate List in R

Example 1:

```
> x1=matrix(data=1:4,nrow=2,ncol=2,byrow=T)
> x2=matrix(data=5:8,nrow=2,ncol=2,byrow=T)
> x1
     [,1] [,2]
[1,]    1    2
[2,]    3    4
> x2
     [,1] [,2]
[1,]    5    6
[2,]    7    8
> x1+x2
     [,1] [,2]
[1,]    6    8
[2,]   10   12
```

Experiment 6: Demonstrate List in R

Example 2:

Now we will try to replace / Manipulate the one of the element in x1 matrix by some character.

```
> x1[2,1]<-"hello"

> x1
      [,1]       [,2]
[1,]  "1"        "2"
[2,]  "hello“    "4"

> x1+x2
Error in x1 + x2 : non-numeric argument to binary operator
```

Experiment 6: Demonstrate List in R

Example 3:

Lists can contain any kind of objects as well as objects of different types. For example, lists can contain matrices as objects.

```
> x1=matrix(data=1:4,nrow=2,ncol=2,byrow=T)
> x2=matrix(data=5:8,nrow=2,ncol=2,byrow=T)
> matlist=list(x1,x2)
> matlist
[[1]]
         [,1]  [,2]
[1,]     1     2
[2,]     3     4

[[2]]
         [,1]  [,2]
[1,]     5     6
[2,]     7     8
>
```

Experiment 6: Demonstrate List in R

Example 4: Extract the elements from a list

```
#Extracting the first element from the list

> matlist[1]
[[1]]
        [,1]  [,2]
[1,]     1     2
[2,]     3     4

#Extracting the Second element from the list

> matlist[2]

[[1]]
        [,1]   [,2]
[1,]     5      6
[2,]     7      8
>
```

Experiment 6: Demonstrate List in R

Example 5: An example list that contains different object types:

```
>z=list(c("water","juice","lemonade"),rep(1:4,each=2),matrix(data=5:8,nrow=2,ncol=2,byrow=T))
>z
[[1]]
[1] "water"   "juice"   "lemonade"

[[2]]
[1] 1 1 2 2 3 3 4 4

[[3]]
     [,1] [,2]
[1,]    5    6
[2,]    7    8
```

Experiment 6: Demonstrate List in R

Access the elements of a list using the operator [[]].

Example 6:

```
> z[[1]]
[1] "water"    "juice"    "lemonade“
```

suppose we want to Extract “**juice**” the command

```
> z[1][2]                          # Notice the position of the brackets
[[1]]
NULL
```

It returns **NULL** instead of “juice”, while

```
> z[[1]][2]                        # Notice the position of the brackets
[1] "juice"
```

Experiment 6: Demonstrate List in R

Merging Lists:

We can merge many lists into one list by placing all the lists inside one list() function.

```
# 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)
```