MET
Bhujbal Knowledge City
MET INSTITUTE OF COMPUTER SCIENCE

Advanced DBMS Lab
ACADEMIC YEAR 2025-2026

Assignment No	7
Title	Advance Datatypes in R
Objective	Lists, Matrix, Factor, Dataframe
Roll No	MCA2511

```
> myList <- list(name = "Mukund", age = 20, city = 'Mumbai')
> print(myList)
$name
[1] "Mukund"

$age
[1] 20

$city
[1] "Mumbai"

>
> empId = c(1,2,3,4)
> empName = c("XYZ", "ABC", "PQR", "MNO")
> numberOfEmp = 4
>
> empList = list("ID" = empId, "Name" = empName, "TotalEmp" = numberOfEmp)
> print(empList$Name)
[1] "XYZ" "ABC" "PQR" "MNO"
> print(empList$Name[2])
[1] "ABC"
>
> #Accessing the numberofEmp column and will display data, i.e.4
> print(empList[3][1])
$TotalEmp
[1] 4

>
>
>
```

```
> empId = c(1,2,3,4)
> empName = c("XYZ", "ABC", "PQR", "MNO")
> numberOfEmp = 4
>
> empList1 = list("ID" = empId, "Name" = empName, "TotalEmp" = numberOfEmp)
> empList$TotalEmp=5
> empList1[[1]][5] = 5
> empList1[[2]][5] = "DEF"
> print(empList1)
$ID
[1] 1 2 3 4 5

$Name
[1] "XYZ" "ABC" "PQR" "MNO" "DEF"

$TotalEmp
[1] 4


>
> empList1[[2]][1] = "RAM"
> empList1$Name
[1] "RAM" "ABC" "PQR" "MNO" "DEF"
>
> #Integer Conatenation
> list1 = list(1,3,5,7,9)
> list2 = list(2,4,6,8,10)
> print(list1)
[[1]]
[1] 1

[[2]]
[1] 3

[[3]]
[1] 5

[[4]]
[1] 7

[[5]]
[1] 9

> print(list2)
```

```
[[1]]
[1] 2

[[2]]
[1] 4

[[3]]
[1] 6

[[4]]
[1] 8

[[5]]
[1] 10

> combined <-c(list1, list2)
> print(combined)
[[1]]
[1] 1

[[2]]
[1] 3

[[3]]
[1] 5

[[4]]
[1] 7

[[5]]
[1] 9

[[6]]
[1] 2

[[7]]
[1] 4

[[8]]
[1] 6

[[9]]
[1] 8
```

```
[[10]]
[1] 10

>
> #String Concatenation
> list3 = list("ABC")
> list4 = list("DEF")
> print(list3)
[[1]]
[1] "ABC"

> print(list4)
[[1]]
[1] "DEF"

> combined1 <- c(list3, list4)
> print(combined1)
[[1]]
[1] "ABC"

[[2]]
[1] "DEF"

> m <- matrix(c(11,12,13,55,60, 65, 66,72,78), nrow=3, ncol=3)
> m
     [,1] [,2] [,3]
[1,]   11   55   66
[2,]   12   60   72
[3,]   13   65   78
>
> dim(m)
[1] 3 3
>
> #By default elements in matrix are filled by column
> #"byrow" attributes of matrix can be used.
> m <- matrix(c(11,12,13,55,60, 65, 66,72,78), nrow=3, ncol=3, byrow=TRUE)
> m
     [,1] [,2] [,3]
[1,]   11   12   13
[2,]   55   60   65
[3,]   66   72   78
>
```

```
> #By default elements in matrix are filled by column
> #"byrow" attributes of matrix can be used.
> m <- matrix(c(11,12,13,55,60, 65, 66,72,78), nrow=3, ncol=3, byrow=TRUE)
> m
     [,1] [,2] [,3]
[1,]   11   12   13
[2,]   55   60   65
[3,]   66   72   78
>
> #cbind-ing and rbind-ing
> #By using cbind() and rbind() functions
> x<-c(1,2,3)
> y<-c(11,12,13)
> z<-cbind(x,y)
> z
     x  y
[1,] 1 11
[2,] 2 12
[3,] 3 13
> z<-rbind(x,y)
> z
  [,1] [,2] [,3]
x    1    2    3
y   11   12   13
>
> #Matrix operation/Scalar
> #Addition, Subtraction an Multiplication of two matrices
> #Transpose, determinant of a matrices,etc
>
> #Multiplication by scalar
> p <- 3*m
> p
     [,1] [,2] [,3]
[1,]   33   36   39
[2,]  165  180  195
[3,]  198  216  234
>
> n <- matrix(c(4,5,6,14,15,16,24,25,26), nrow = 3, ncol=3)
> #Addition of two matrices
> q <- m+n
> q
     [,1] [,2] [,3]
[1,]   15   26   37
```

```
[2,]   60   75   90
[3,]   72   88  104
>
> o<-matrix(c(4,5,6,14,15,16,24,25,26),nrow=3,ncol=3)
> o
     [,1] [,2] [,3]
[1,]    4   14   24
[2,]    5   15   25
[3,]    6   16   26
>
> #Matrix multiplication by using %*%
> r <- m%*%o
> r
     [,1] [,2] [,3]
[1,]  182  542  902
[2,]  910 2710 4510
[3,] 1092 3252 5412
>
> #Transpose of Matrix
> mdash <-t(m)
> mdash
     [,1] [,2] [,3]
[1,]   11   55   66
[2,]   12   60   72
[3,]   13   65   78
> #Creating a vector
> x<-c("female","male","male","female")
> print(x)
[1] "female" "male"   "male"   "female"
>
> #converting the vector x into a factor
> #names gender
> gender<-factor(x)
> print(gender)
[1] female male   male   female
Levels: female male
>
> #Creating a factor with levels defined by programmer
> gender <-factor(c("female", "male", "male", "female"),
+         levels = c("female", "transgender","male"));
> gender
[1] female male   male   female
Levels: female transgender male
```

```
>
> #Checking for a Factor in R
> gender <-factor(c("female","male","male","female"));
> print(is.factor(gender))
[1] TRUE
>
> #Accessing elements of a Factor in R
> gender<-factor(c("female","male","male","female"));
> gender[3]
[1] male
Levels: female male
>
> #More than one element can be accessed at a time
> gender <- factor(c("female","male","male","female"));
> gender[c(2,4)]
[1] male   female
Levels: female male
>
> #Subtract one element at a time
> gender <-factor(c("female","male","male","female"));
> gender[-3]
[1] female male   female
Levels: female male
>
> #Modification of a factor in R
> gender <-factor(c("female","male","male","female"));
> gender[2] <- "female"
> gender
[1] female female male   female
Levels: female male
>
>
> #R Program to create a dataframe
> #creating  a dataframe
> friend.data <- data.frame(
+   friend_id = c(1:5),
+   friend_name = c("Mukund","Nimish",  "Altamash", "Aniket", "Sumit"),
+   stringsAsFactors = FALSE
+ )
>
> #prinr the dataframe
> print(friend.data)
  friend_id friend_name
```

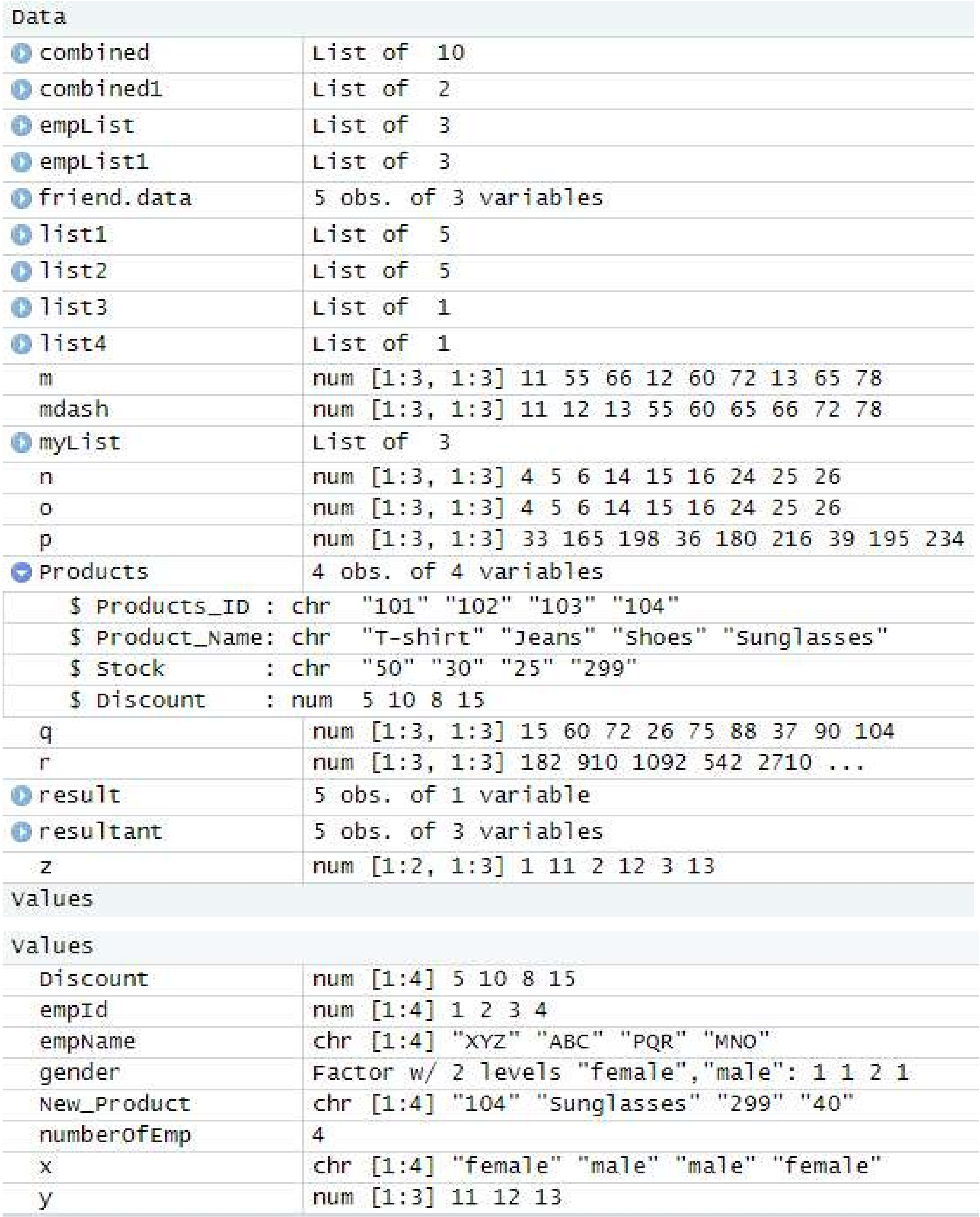
```
1         1     Mukund
2         2     Nimish
3         3   Altamash
4         4     Aniket
5         5      Sumit
>
> #R Program to get the structure of the datafraem
> #using str()
> print(str(friend.data))
'data.frame':	5 obs. of  2 variables:
 $ friend_id  : int  1 2 3 4 5
 $ friend_name: chr  "Mukund" "Nimish" "Altamash" "Aniket" ...
NULL
>
> #R Program to get the summary of the dataframe
> #using summary()
> print(summary(friend.data))
   friend_id friend_name
 Min.   :1   Length:5
 1st Qu.:2   Class :character
 Median :3   Mode  :character
 Mean   :3
 3rd Qu.:4
 Max.   :5
>
> #R program to extract data from the dataframe
> #Extracting friend_name column
> result <- data.frame(friend.data$friend_name)
> print(result)
  friend.data.friend_name
1                  Mukund
2                  Nimish
3                Altamash
4                  Aniket
5                   Sumit
>
> #R program to expand the dataframe
> #Expanding dataframe
> friend.data$location <- c("Kalyan","Badlapur","Nagpur","Nalasupara","Ghatkoper")
> resultant <-friend.data
>
> #Print the modified data frame
> print(resultant)
```

```
  friend_id friend_name   location
1         1      Mukund     Kalyan
2         2      Nimish   Badlapur
3         3    Altamash     Nagpur
4         4      Aniket Nalasupara
5         5       Sumit  Ghatkoper
>
> #print items using[]
> friend.data[1]
  friend_id
1         1
2         2
3         3
4         4
5         5
>
> #access items using [[]]
> friend.data[["friend_name"]]
[1] "Mukund"   "Nimish"   "Altamash" "Aniket"   "Sumit"
>
> #access items using $
> friend.data$friend_id
[1] 1 2 3 4 5
>
> #find out the number of rows and columns
> dim(friend.data)
[1] 5 3
>
> #add Rows and Columns in R Dataframe
> #Creating a dataframe representing products in a store
> Products <-data.frame(
+   Products_ID = c(101,102,103),
+   Product_Name=c("T-shirt","Jeans","Shoes"),
+   Stock=c(50,30,25)
+ )
>
> #Print the existing datframe
> cat("Existing dataframe(Products) : \n")
Existing dataframe(Products) :
> print(Products)
  Products_ID Product_Name Stock
1         101      T-shirt    50
2         102        Jeans    30
```

```
3        103       Shoes    25
> 
> #Adding a new row for a new product
> New_Product <- c(104,"Sunglasses",299,40)
> Products<-rbind(Products,New_Product)
> 
> #Print the updated dataframe after adding the new product
> cat("\n Updated dataframe after adding a new product: \n")

 Updated dataframe after adding a new product: 
> print(Products)
  Products_ID Product_Name Stock
1        101     T-shirt    50
2        102       Jeans    30
3        103       Shoes    25
4        104   Sunglasses   299
> 
> #Adding a new column for 'Discount' to the dataframe
> Discount <-c(5,10,8,15)
> Products <- cbind(Products,Discount)
> 
> #Rename the added column
> colnames(Products)[ncol(Products)] <- "Discount"
> 
> #Print the updated dataframe after adding the new Column
> cat("\n Updated dataframe after adding a new Discount: \n")

 Updated dataframe after adding a new Discount: 
> print(Products)
  Products_ID Product_Name Stock Discount
1        101     T-shirt    50        5
2        102       Jeans    30       10
3        103       Shoes    25        8
4        104   Sunglasses   299       15
```



```
Data
combined          List of  10
combined1         List of  2
empList           List of  3
empList1          List of  3
friend.data       5 obs. of 3 variables
list1             List of  5
list2             List of  5
list3             List of  1
list4             List of  1
m                 num [1:3, 1:3] 11 55 66 12 60 72 13 65 78
mdash             num [1:3, 1:3] 11 12 13 55 60 65 66 72 78
myList            List of  3
n                 num [1:3, 1:3] 4 5 6 14 15 16 24 25 26
o                 num [1:3, 1:3] 4 5 6 14 15 16 24 25 26
p                 num [1:3, 1:3] 33 165 198 36 180 216 39 195 234
Products          4 obs. of 4 variables
    $ Products_ID : chr  "101" "102" "103" "104"
    $ Product_Name: chr  "T-shirt" "Jeans" "Shoes" "Sunglasses"
    $ Stock       : chr  "50" "30" "25" "299"
    $ Discount    : num  5 10 8 15
q                 num [1:3, 1:3] 15 60 72 26 75 88 37 90 104
r                 num [1:3, 1:3] 182 910 1092 542 2710 ...
result            5 obs. of 1 variable
resultant         5 obs. of 3 variables
z                 num [1:2, 1:3] 1 11 2 12 3 13
Values

Values
Discount          num [1:4] 5 10 8 15
empId             num [1:4] 1 2 3 4
empName           chr [1:4] "XYZ" "ABC" "PQR" "MNO"
gender            Factor w/ 2 levels "female","male": 1 1 2 1
New_Product       chr [1:4] "104" "Sunglasses" "299" "40"
numberOfEmp       4
x                 chr [1:4] "female" "male" "male" "female"
y                 num [1:3] 11 12 13
```