

1) Create simple vectors of data ( same size and different size too) and perform operations on it.

```
> #creating vector using colon operator
> x<-2:9
> x
[1] 2 3 4 5 6 7 8 9
> #creating a vector using seq() function
> seq_vec<-seq(1,4,by=0.5)
> seq_vec
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0
> #4 types of atomic vectors:numeric,logical,integer,character
> #numeric vector
> num_v<-c(10.1,10.2,33.2)
> num_v
[1] 10.1 10.2 33.2
> class(num_v)
[1] "numeric"
> int_v<-c(1,2,3,4,5)
> class(int_v)
[1] "numeric"
> int_v<-as.integer(int_v)
> class(int_v)
[1] "integer"
> d<-'hello'
> d
[1] "hello"
> class(d)
[1] "character"
> char_vec<-c(1,2,3,4,5)
> char_vec<-as.character(char_vec)
> class(char_vec)
[1] "character"
> f<-as.integer(5)
> g<-as.integer(6)
> h<-as.integer(7)
> i<-f>g
> i
[1] FALSE
> j<-h>g
> j
[1] TRUE
> class(j)
[1] "logical"
> class(i)
[1] "logical"
> #accessing the elements of the vector
> seq_vec
[1] 1.0 1.5 2.0 2.5 3.0 3.5 4.0
> seq_vec[2]
[1] 1.5
> #indexing starts from 1
> char_vector<-c("Rashi","Vaishali","Arpita")
> char_vector
```

```

[1] "Rashi" "Vaishali" "Arpita"
> charv<-c("suruchi"=32,"sanika"=9,"prachiti"=7)
> charv
suruchi sanika prachiti
    32     9     7
> charv["sanika"]
sanika
    9
> v<-c(1,3,5,7,9)
> v[c(TRUE,FALSE,TRUE,TRUE,FALSE,TRUE)]
[1] 1 5 7 NA
> #p<-c(1,2,4,8)
> p<-c(8,9,4,7)
> q<-c("Arshia","Nishika","Rahi")
> r<-c(p,q)
> r
[1] "8" "9" "4" "7" "Arshia" "Nishika" "Rahi"
> #arithmetic operations
> m1<-c(1,3,5,7)
> m2<-c(2,4,6,8)
> m1+m2
[1] 3 7 11 15
> m1-m2
[1] -1 -1 -1 -1
> m1*m2
[1] 2 12 30 56
> m1%%m2
[1] 1 3 5 7
> p1<-c("purva","manasi","nishtha")
> p2<-c(TRUE,FALSE,TRUE)
> p1[p2]
[1] "purva" "nishtha"
> d<-c("a","b","c","d")
> d[c(2,4,4,3)]
[1] "b" "d" "d" "c"
> d[2:4]
[1] "b" "c" "d"
> d[1:3]
[1] "a" "b" "c"
> d[1:2]
[1] "a" "b"

```

2) Create matrix and perform some operations on it.

```

>#a matrix can store only one type of data
> #creating a matrix
> #arranging the elements sequentially by row
> P<-matrix(c(5,16),nrow=4,byrow=TRUE)
> P
     [,1]
[1,]  5
[2,] 16
[3,]  5
[4,] 16

```

```

> P<-matrix(c(3:14),nrow=4,byrow=TRUE)
> P
      [,1] [,2] [,3]
[1,]   3   4   5
[2,]   6   7   8
[3,]   9  10  11
[4,]  12  13  14
> #arranging the elements sequentially by columns
> Q
      [,1] [,2] [,3]
[1,]   7  11  15
[2,]   8  12   7
[3,]   9  13   8
[4,]  10  14   9
> R<-matrix(c(7:22),nrow=4,byrow=FALSE)
> R
      [,1] [,2] [,3] [,4]
[1,]   7  11  15  19
[2,]   8  12  16  20
[3,]   9  13  17  21
[4,]  10  14  18  22
> R[2,3]
[1] 16
> R[4,2]
[1] 14
> R[1,2]
[1] 11
> R[1,2]<-9
> R
      [,1] [,2] [,3] [,4]
[1,]   7   9  15  19
[2,]   8  12  16  20
[3,]   9  13  17  21
[4,]  10  14  18  22
> R[1,2]
[1] 9
> row_names=c("row1","row2","row3","row4")
> col_names=c("col1","col2","col3")
> ma<-matrix(c(5:16),nrow=4,byrow=TRUE,dimnames = list(row_names,col_names))
> ma
      col1 col2 col3
row1    5   6   7
row2    8   9  10
row3   11  12  13
row4   14  15  16
> ma[ma==12]<-0
> ma
      col1 col2 col3
row1    5   6   7
row2    8   9  10
row3   11   0  13
row4   14  15  16
> r_nme=c("row1","row2","row3","row4")

```

```

> c_nme=c("col1","col2","col3","col4")
> m3<-matrix(c(1:16),nrow=4,byrow=TRUE,dimnames=list(r_nme,c_nme))
> m3
  col1 col2 col3 col4
row1   1   2   3   4
row2   5   6   7   8
row3   9  10  11  12
row4  13  14  15  16
> #adding rows
> rbind(m3,c(1,2,3,4))
  col1 col2 col3 col4
row1   1   2   3   4
row2   5   6   7   8
row3   9  10  11  12
row4  13  14  15  16
  1   2   3   4
> #adding a column
> cbind(m3,c(5,0,9,7))
  col1 col2 col3 col4
row1   1   2   3   4 5
row2   5   6   7   8 0
row3   9  10  11  12 9
row4  13  14  15  16 7
> #transpose of matrix by using t() function
> t(m3)
  row1 row2 row3 row4
col1   1   5   9  13
col2   2   6  10  14
col3   3   7  11  15
col4   4   8  12  16
> #matrix operations
> #addition,subtraction,multiplication,division
> M1<-matrix(c(5:16),nrow=4,ncol=3)
> M2<-matrix(c(1:12),nrow=4,ncol=3)
> #Addition
> sum<-M1+M2
> sum
  [,1] [,2] [,3]
[1,]   6  14  22
[2,]   8  16  24
[3,]  10  18  26
[4,]  12  20  28
> M1
  [,1] [,2] [,3]
[1,]   5   9  13
[2,]   6  10  14
[3,]   7  11  15
[4,]   8  12  16
> M2
  [,1] [,2] [,3]
[1,]   1   5   9
[2,]   2   6  10
[3,]   3   7  11
[4,]   4   8  12

```

```

> #Subtraction
> sub<-M1-M2
> sub
  [,1] [,2] [,3]
[1,]  4  4  4
[2,]  4  4  4
[3,]  4  4  4
[4,]  4  4  4
> #Mutliplication
> mu<-M1*M2
> mu
  [,1] [,2] [,3]
[1,]  5 45 117
[2,] 12 60 140
[3,] 21 77 165
[4,] 32 96 192
> #Multiplication by a constant
> muc<-M1*2
> muc
  [,1] [,2] [,3]
[1,] 10 18 26
[2,] 12 20 28
[3,] 14 22 30
[4,] 16 24 32
> #division
> div<-M1/M2
> div
  [,1] [,2] [,3]
[1,] 5.000000 1.800000 1.444444
[2,] 3.000000 1.666667 1.400000
[3,] 2.333333 1.571429 1.363636
[4,] 2.000000 1.500000 1.333333

```

3) Create a data frame of different types of data, access its rows,columns. Add one row and one colum after frame creation. Also perform some basic operations on data frames

```

> #creating a dataframe
> name<-c('Rutvi','Pranjal','Madhura')
> language<-c('Python','Java','C++')
> age<-c(20,19,20)
> df<-data.frame(name,language,age)
> df
  name language age
1 Rutvi  Python  20
2 Pranjal   Java  19
3 Madhura   C++  20
> #accessing first and second row
> df[1:2,]
  name language age
1 Rutvi  Python  20
2 Pranjal   Java  19
> #accessing the first and second column

```

```

> df[,1:2]
  name language
1 Rutvi  Python
2 Pranjal  Java
3 Madhura  C++
> #selecting subset of dataframe
> newdf=subset(df,name=='Rutvi'|age==19)
> newdf
  name language age
1 Rutvi  Python  20
2 Pranjal  Java  19
> df
  name language age
1 Rutvi  Python  20
2 Pranjal  Java  19
3 Madhura  C++  22
> #adding a column to dataframe
> #adding a new column using cbind()
> Newdf=cbind(df,Rank=c(3,5,1))
> Newdf
  name language age Rank
1 Rutvi  Python  20   3
2 Pranjal  Java  19   5
3 Madhura  C++  22   1
> #adding a new row
> Newdf=rbind(df,data.frame(name='Anushka',language='C#',age=30))
> Newdf
  name language age
1 Rutvi  Python  20
2 Pranjal  Java  19
3 Madhura  C++  22
4 Anushka  C#  30
> #deleting rows and columns
> #delete the third row and the second column
> newwdf=df[-3,-2]
> newwdf
  name age
1 Rutvi 20
2 Pranjal 19

```

4) Create a factor and perform basic operations like table creation for the factor.

```

#creating a factor
> #creating a vector as the input
> data<-c('Sanika','Shirish','Jagdeep','Romil','Rakesh','Priyanshi','Sahil')
> data
[1] "Sanika" "Shirish" "Jagdeep" "Romil" "Rakesh" "Priyanshi"
[7] "Sahil"
> is.factor(data)
[1] FALSE
> #converting the vector into a factor named factor_data

```

```

> factor_data=factor(data)
> factor_data
[1] Sanika  Shirish Jagdeep Romil  Rakesh  Priyanshi Sahil
Levels: Jagdeep Priyanshi Rakesh Romil Sahil Sanika Shirish
> is.factor(factor_data)
[1] TRUE
> #factors can be used to create tables in R
> Q1=factor(c("Sometimes","Sometimes","Never","Always","Always","Sometimes","Sometimes"))
> Q2=factor(c("Maybe","Maybe","Yes","Maybe","No","Yes","No"))
> t=table(Q1,Q2)
> t

```

	Q2			
Q1	Maybe	No	Yes	
Always	1	1	0	
Never	0	0	1	
Sometimes	2	1	1	

```

> #tables can be made from one,two or many factors
> #extract new table for columns 2 and 3 of t
> t[,2:3]

```

	Q2		
Q1	No	Yes	
Always	1	0	
Never	0	1	
Sometimes	1	1	

```

> #only 3rd row from table
> t[3,]

```

	Maybe	No	Yes
	2	1	1

5) Create a function to calculate sum of first n numbers, and call that function too.

```

> #create a function of first n numbers and call that function
> new.function <- function(n) {
+ result=sum(1:n)
+ print(result)
+ }
> new.function(10)
[1] 55
> new.function(5)
[1] 15
> new.function(40)
[1] 820

```

6) Use a builtin data set , convert it into frame and read that dataset, its coulmns , rows etc.

```

> library(datasets)
>
> #see the list of datasets availbale with R
> data()
> #learn more about a dataset
> help("women")

```

```

> data("women")
> women
  height weight
1    58   115
2    59   117
3    60   120
4    61   123
5    62   126
6    63   129
7    64   132
8    65   135
9    66   139
10   67   142
11   68   146
12   69   150
13   70   154
14   71   159
15   72   164

```

7) Read csv file and store file data in frame and view few data rows.

```

> #input.csv is the csv file to be read
> data<-read.csv("input.csv")
> print(is.data.frame(data))
[1] TRUE
> print(ncol(data))
[1] 5
> print(nrow(data))
[1] 8
> data
  id  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   Nina 578.00 2013-05-21      IT
7  7  Simon 632.80 2013-07-30 Operations
8  8   Guru 722.50 2014-06-17  Finance
> data<-read.csv("input.csv")
> print(is.data.frame(data))
[1] TRUE
> print(ncol(data))
[1] 5
> print(nrow(data))
[1] 8
> data
  id  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   Nina 578.00 2013-05-21      IT

```



```
7 7 Simon 632.80 2013-07-30 Operations
8 8 Guru 722.50 2014-06-17 Finance
> #get maximum salary
> sal <- max(data$salary)
> sal
[1] 843.25
> #get all people working in the IT department
> retval <- subset( data, dept == "IT")
> print(retval)
  id  name salary start_date dept
1  1  Rick  623.3 2012-01-01  IT
3  3 Michelle 611.0 2014-11-15  IT
6  6  Nina  578.0 2013-05-21  IT
> info <- subset(data, salary > 600 & dept == "IT")
> print(info)
  id  name salary start_date dept
1  1  Rick  623.3 2012-01-01  IT
3  3 Michelle 611.0 2014-11-15  IT
```