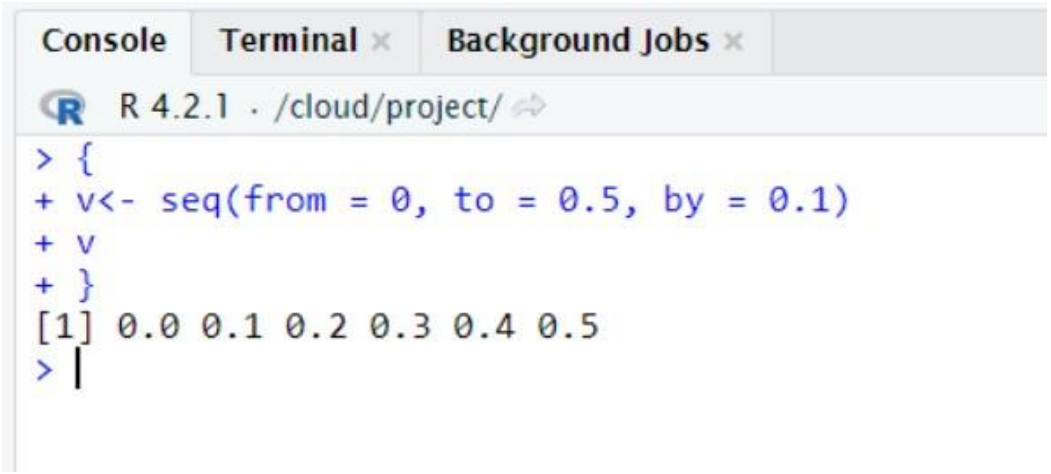
EXP NO: 03	WORKING WITH LIST, MATRICES, ARRAYS, DATA FRAMES AND PLOTS
DATE:	

AIM:

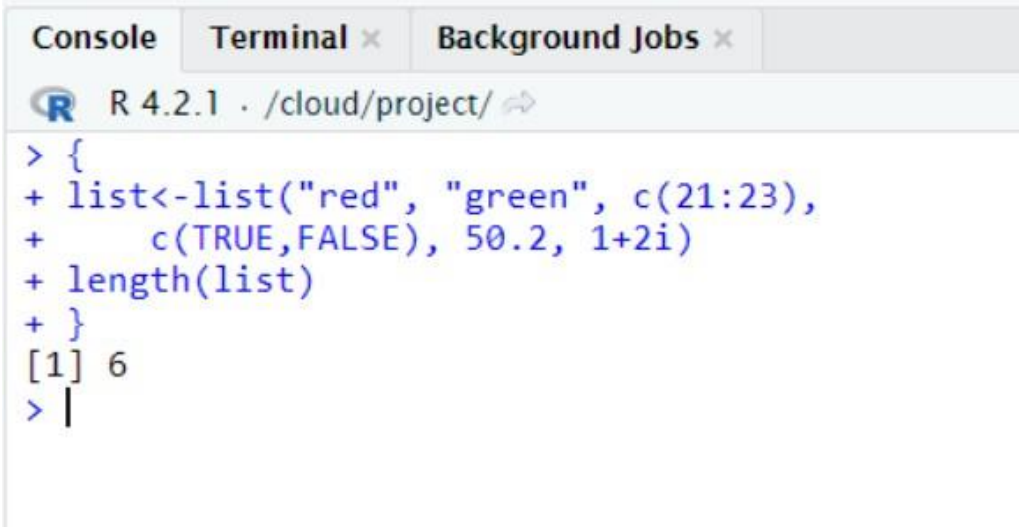
To work with list, matrices, arrays, data frames and plots.

PROGRAM:

1. **Predict the output: v<- seq(from = 0, to = 0.5, by = 0.1)**

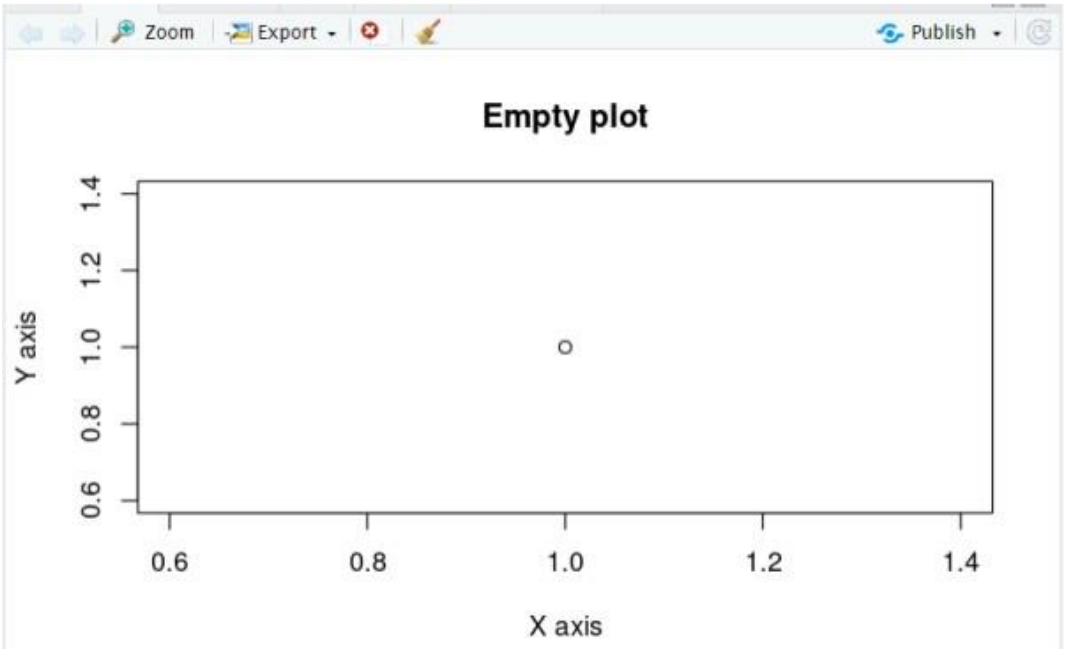


2. **Execute a list which contains the following values {"red", "green", {21:23}, {TRUE,FALSE}, 50.2, 1+2i} and print the count of the list.**



3. **Create an empty plot and give title for axis.**

```
Console  Terminal  Background Jobs
R 4.2.1 · /cloud/project/
> {
+ plot(1,main="Empty plot",xlab="X axis",ylab="Y axis")
+ }
>
```

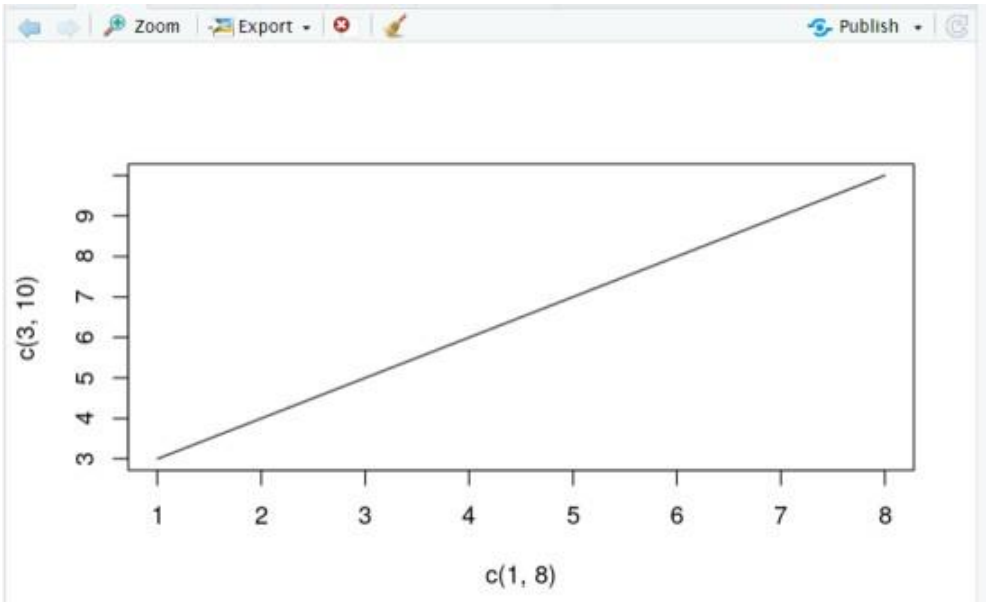


4. **Create vectors for the given information and make it as one vector.**
   V1=(“mike”,”Lucky”,”John”)
   V2=(25,30,45)

```
Console  Terminal  Background Jobs
R 4.2.1 · /cloud/project/
> {
+ V1<-c("mike","Lucky","John")
+ V2<-c(25,30,45)
+ c(V1,V2)
+ }
[1] "mike"  "Lucky" "John"  "25"    "30"    "45"
>
```

5. **Write a simple R code to draw two points in the diagram, one at position (1, 3) and one in position (8, 10) and draw a line to connect the points.**

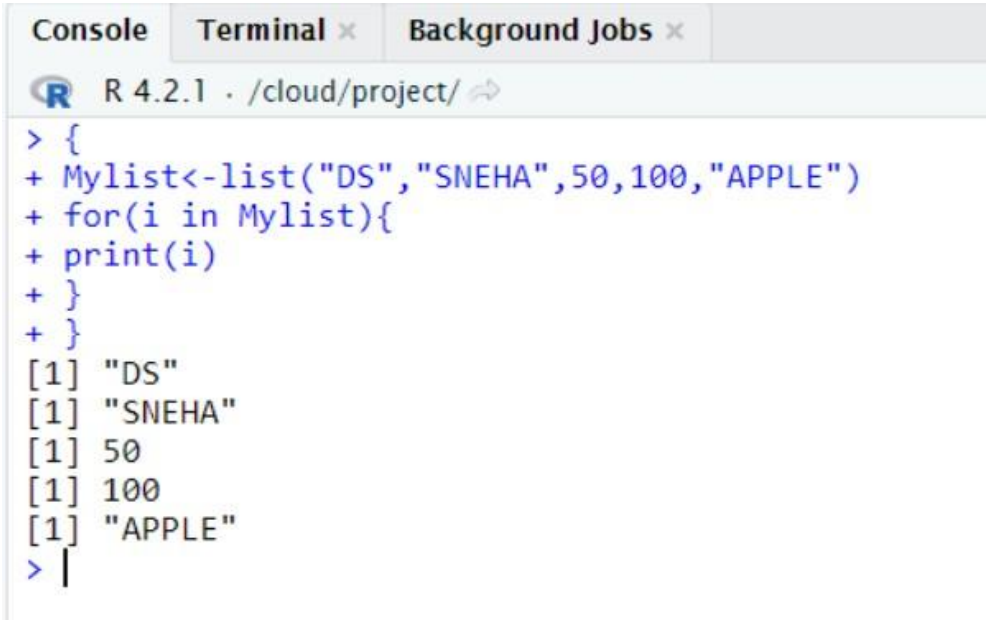
```
Console  Terminal ×  Background Jobs ×
R 4.2.1 · /cloud/project/
> {
+ plot(c(1,8),c(3,10),type="l")
+ }
>
```



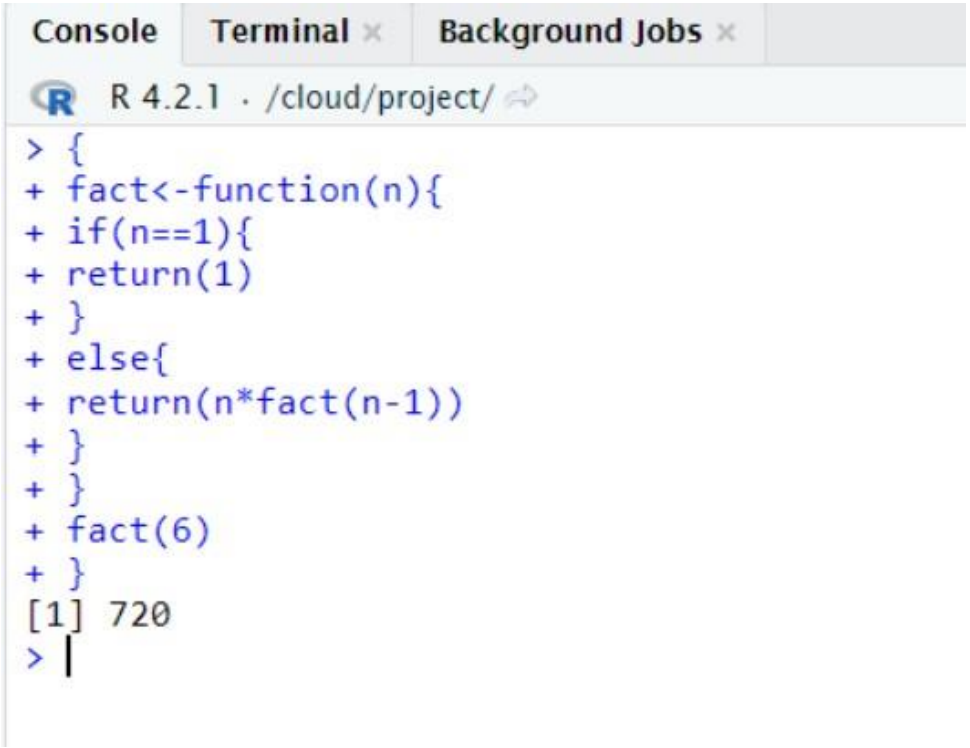
6. **Write a R Program to find the common elements from below Vectors.**
   x = c(10, 20, 30, 20, 20, 25, 29, 26)
   y = c(10, 50, 30, 20, 20, 35, 19, 56)
   z = c(10, 40, 30, 20, 20, 25, 49, 26)

```
Console  Terminal ×  Background Jobs ×
R 4.2.1 · /cloud/project/
> {
+ x <- c(10, 20, 30, 20, 20, 25, 29, 26)
+ y <- c(10, 50, 30, 20, 20, 35, 19, 56)
+ z <- c(10, 40, 30, 20, 20, 25, 49, 26)
+ intersect(intersect(x,y),z)
+ }
[1] 10 20 30
>
```

7. **Create the list of 5 elements and display the list using for loop.**



8. **Write a recursive function in R to find the factorial of a given number.**

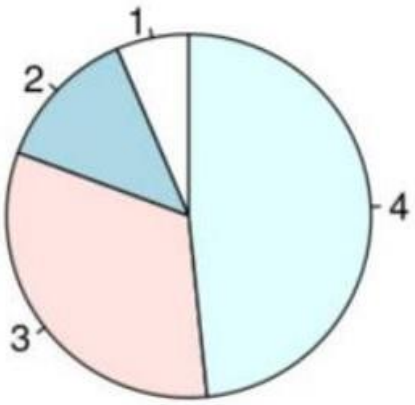


9. **x-axis values = (A,B,C,D) ; y-axis values = (2,4,6,8)**
   Draw a vertical bar chart with the labels.

```
> {
+ x<-c("A","B","C","D")
+ y<-c(2,4,6,8)
+ barplot(y,names.arg=x,col="pink")
+ }
>
```

10. **Create and display your own pie chart and start the first pie at 90 degrees.**

```
> {
+ x<-c(10,20,50,75)
+ pie(x,init.angle=90)
+ }
>
```



11. Create the array of elements 1:36 with 5 rows 3 cols and 2 dimensions and write the code to display the element 27.

```
Console   Terminal ×   Background Jobs ×
R 4.2.0 · /cloud/project/
> {
+ arr<-array(1:36,dim=c(5,3,2))
+ print(arr)
+ arr[2,3,2]
+ }
, , 1

     [,1] [,2] [,3]
[1,]    1    6   11
[2,]    2    7   12
[3,]    3    8   13
[4,]    4    9   14
[5,]    5   10   15

, , 2

     [,1] [,2] [,3]
[1,]   16   21   26
[2,]   17   22   27
[3,]   18   23   28
[4,]   19   24   29
[5,]   20   25   30

[1] 27
>
```

12. Create a vector of elements “banana”, “apple”, “orange”, “mango”, “lemon” and do the following:

```
R 4.2.0 · /cloud/project/
> v<-c("banana","apple","orange","mango","lemon")
> v
[1] "banana" "apple"  "orange" "mango"  "lemon"
> 
```

i) Sort the given vector

```
> sort(v)
[1] "apple"  "banana" "lemon"  "mango"  "orange"
> 
```

ii) Display the vector except the first element

```
> v[-1]
[1] "apple"  "orange" "mango"  "lemon"
> 
```

iii) Access the first and third item.

```
> v[c(1,3)]
[1] "banana" "orange"
> 
```

iv) Repeat the sequence of the vector two times

```
> rep(v,times=2)
 [1] "banana" "apple"  "orange" "mango"  "lemon"  "banana" "apple"  "orange" "mango"
[10] "lemon"
> 
```

v) Update the second item to “Rose”

```
> v[2]="Rose"
> v
[1] "banana" "Rose"   "orange" "mango"  "lemon"
> 
```

vi) Delete the entire vector

```
> v=NULL
> v
NULL
> 
```

13. Create the following data frame.

	Age	Height	Weight	Sex
Alex	25	177	57	F
Lilly	31	163	69	F
Mark	23	190	83	M
Oliver	52	179	75	M
Martha	76	163	70	F
Lucas	49	183	83	M
Caroline	26	164	53	F

```
> names=c("Alex","Lilly","Mark","Oliver","Martha","Lucas","Caroline")
> df<-data.frame(names,"Age"=c(25,31,23,52,76,49,26),"Height"=c(177,163,190,179,163,183,16
4),"Weight"=c(57,69,83,75,70,83,53),"Sex"=c("F","F","M","M","F","M","F"))
> df
     names Age Height Weight Sex
1     Alex  25    177     57   F
2    Lilly  31    163     69   F
3     Mark  23    190     83   M
4   Oliver  52    179     75   M
5   Martha  76    163     70   F
6    Lucas  49    183     83   M
7 Caroline  26    164     53   F
```

i) Extract first two rows from data frame

```
> df[1:2,]
  names Age Height Weight Sex
1  Alex  25    177     57   F
2 Lilly  31    163     69   F
>
```

ii) Extract 1st column and 4th column from data frame.

```
> df[,c(1,4)]
     names Weight
1     Alex     57
2    Lilly     69
3     Mark     83
4   Oliver     75
5   Martha     70
6    Lucas     83
7 Caroline     53
>
```

iii) Add a new column in an existing data frame

```
> df<-cbind(df,"Working"=c("Yes","No","No","Yes","Yes","No","Yes"))
> df
     names Age Height Weight Sex Working
1     Alex  25    177     57   F     Yes
2    Lilly  31    163     69   F      No
3     Mark  23    190     83   M      No
4   Oliver  52    179     75   M     Yes
5   Martha  76    163     70   F     Yes
6    Lucas  49    183     83   M      No
7 Caroline  26    164     53   F     Yes
>
```

iv) Drop the last row from the data frame

```
> head(df,-1)
   names Age Height Weight Sex Working
1   Alex  25    177     57   F     Yes
2  Lilly  31    163     69   F      No
3   Mark  23    190     83   M      No
4 Oliver  52    179     75   M     Yes
5 Martha  76    163     70   F     Yes
6  Lucas  49    183     83   M      No
>
```

v) Change the name of the 2nd column in a given data frame.

```
> colnames(df)[2]<-"NewAge"
> df
     names NewAge Height Weight Sex Working
1     Alex     25    177     57   F     Yes
2    Lilly     31    163     69   F      No
3     Mark     23    190     83   M      No
4   Oliver     52    179     75   M     Yes
5   Martha     76    163     70   F     Yes
6    Lucas     49    183     83   M      No
7 Caroline     26    164     53   F     Yes
>
```

vi) Remove rows and columns in a Data Frame

```
> df[-c(2),-c(1)]
  NewAge Height Weight Sex Working
1     25    177     57   F     Yes
3     23    190     83   M      No
4     52    179     75   M     Yes
5     76    163     70   F     Yes
6     49    183     83   M      No
7     26    164     53   F     Yes
> |
```

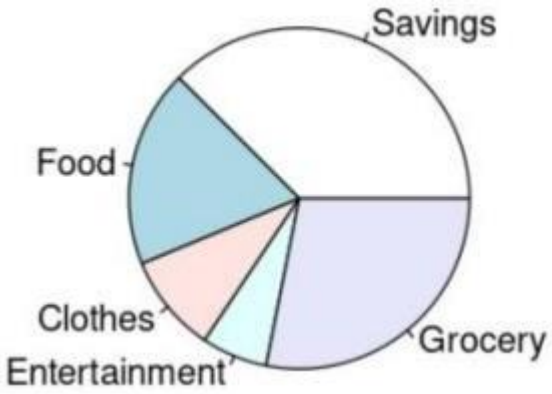
14. Use R to create the following two matrices and do the indicated matrix multiplication. What is the resulting matrix?

$$\begin{bmatrix} 7 & 9 & 12 \\ 2 & 4 & 13 \end{bmatrix} \times \begin{bmatrix} 1 & 7 & 12 & 19 \\ 2 & 8 & 13 & 20 \\ 3 & 9 & 14 & 21 \end{bmatrix}$$

```
> {
+ m1<-matrix(c(7,9,12,2,4,13),nrow=2,ncol=3,byrow=TRUE)
+ m2<-matrix(c(1,2,3,7,8,9,12,13,14,19,20,21),nrow=3,ncol=4)
+ print(m1%*%m2)
+ }
     [,1] [,2] [,3] [,4]
[1,]   61  229  369  565
[2,]   49  163  258  391
> |
```

15. Write R program to create pie chart for the following data.10 Savings -------600; Food --------300; Clothes ---------150; Entertainment---100; Grocery ----- 450

```
> {
+ x<-c(600,300,150,100,450)
+ lab<-c("Savings","Food","Clothes","Entertainment","Grocery")
+ pie(x,label=lab)
+ }
> |
```



16. Illustrate the following Matrix by using R Implement the given Matrix.

Row \ Column	E1	E2	E3
One	1	5	9
Two	2	6	10
Three	3	7	11
Four	4	8	12

```
> row<-c("One","Two","Three","Four")
> col<-c("E1","E2","E3")
> m<-matrix(1:12,nrow=4,ncol=3,dimnames=list(row,col))
> m
      E1 E2 E3
One    1  5  9
Two    2  6 10
Three  3  7 11
Four   4  8 12
>
```

i) **Perform the transpose of the matrix**

```
> t(m)
   One Two Three Four
E1   1   2     3    4
E2   5   6     7    8
E3   9  10    11   12
> 
```

ii) **Display (1,2) cell value**

```
> m[1,2]
[1] 5
> 
```

iii) **Display the first and third row along with all columns**

```
> m[c(1,3),]
      E1 E2 E3
One    1  5  9
Three  3  7 11
> 
```

iv) **Delete the first column**

```
> m[,-c(1)]
      E2 E3
One    5  9
Two    6 10
Three  7 11
Four   8 12
> 
```

v) **Rename the column names to A, B, C**

```
> colnames(m)<-c("A","B","C")
> m
      A B  C
One   1 5  9
Two   2 6 10
Three 3 7 11
Four  4 8 12
> 
```

vi) Replace the value of 11 to 15 in the given matrix

```
> m[m==11]<-15
> m
      A B  C
One   1 5  9
Two   2 6 10
Three 3 7 15
Four  4 8 12
> |
```

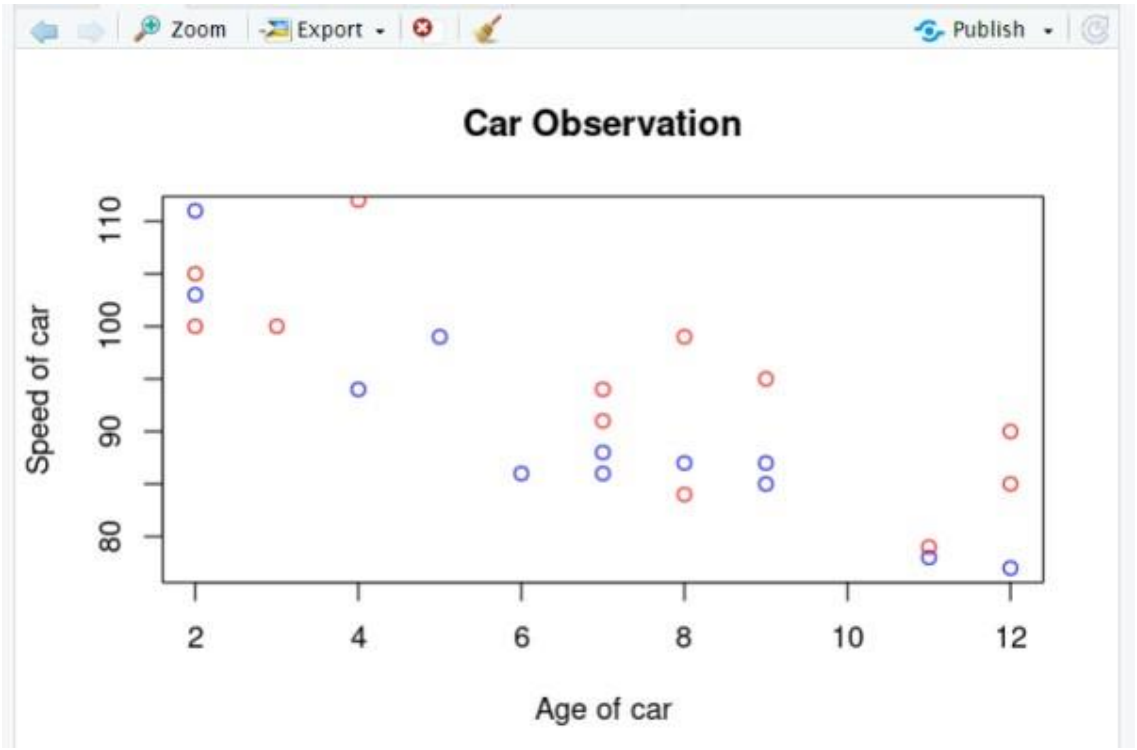
vii) Find the mean of each column and sum of each row.

```
> colMeans(m)
   A    B    C
 2.5  6.5 11.5
> |
```

```
> rowSums(m)
  One   Two Three  Four
   15    18    25    24
> |
```

17. Day 1 Observation:
X1 = c(5,7,8,7,2,2,9,4,11,12,9,6); Y1 = c(99,86,87,88,111,103,87,94,78,77,85,86)
Day 2 Observation:
x2 = c(2,2,8,1,15,8,12,9,7,3,11,4,7,14,12); y2 = c(100, 105, 84, 105, 90, 99, 90, 95, 94, 100, 79, 112, 91, 80, 85)
The x-axis shows how old the car is. The y-axis shows the speed of the car when it passes. Draw and compare the plots. Use different colours and give labelling.

```
> {
+ X1 <- c(5,7,8,7,2,2,9,4,11,12,9,6)
+ Y1 <- c(99,86,87,88,111,103,87,94,78,77,85,86)
+ x2 <- c(2,2,8,1,15,8,12,9,7,3,11,4,7,14,12);
+ y2 <- c(100, 105, 84, 105, 90, 99, 90, 95, 94, 100,
+        79, 112, 91, 80, 85)
+ plot(X1,Y1,main="Car Observation",xlab="Age of car",ylab="Speed of car",col="blue")
+ points(x2,y2,col="red")
+ }
> |
```



- By comparing we can conclude that, the newer the car, the faster it drives.

RESULT:

Thus, list, matrices, arrays, data frames and plots has been implemented and executed.