Program-4

P-4 Create a vector of all those values from 1 to 100 that are divisible by 5 and do the following operations on vector x.
a). Find the length of vector x
b). Print the value stored at the 10th,20th location of vector x.
c). Find the sum,mean,meadian and standard deviation of vector x
d). Create another vector with name "color" and print its values
e). Process 7th and 8th location of vector color
f). Repeat values in color vector excatly twice
g). Access multiple elements simultenously in color vector

```
#Name : Jyoti Devi
#Roll no. : 200010130051
#Class : B.tech(cse),g3

#Code:-
# A sequence of numbers from 1 to 100 that are divisible by 5
x<-seq(from=5,to=100,by=5)
x

#a. Find the length of vector x
len<-length(x)
len

#b. Print the value stored at the 10th,20th location of vector x.
x[10]
x[20]

#c. Find the sum,mean,meadian and standard deviation of vector x
s<-sum(x)
s
m<-mean(x)
m
md<-median(x)
md
sd<-sd(x)
sd

#d. Create another vector with name "color" and print its values
color<-c("red","blue","black","green","white","grey","pink","orange")
color

#e. Process 7th and 8th location of vector color
color[7]
color[8]

#g Repeat values in color vector excatly twice
a<-rep(color,each=2)
a
```

```
#h. Access multiple elements simultenously in color vector
color[2]
color[4]

for(x in color){
  print(x)
}
```

Output:-

Console | Terminal | Jobs

R 4.2.2 · ~/

```
> #Name : Jyoti Devi
> #Roll no. : 200010130051
> #Class : B.tech(cse),g3
>
>
> #sequence of numbers from 1 to 100 that are divisible by 5
> x<-seq(from=5,to=100,by=5)
> x
 [1]   5  10  15  20  25  30  35  40  45  50  55  60  65  70  75  80  85  90  95 100
> #a. Find the length of vector x
> len<-length(x)
> len
[1] 20
>
> #b. Print the value stored at the 10th,20th location of vector x.
> x[10]
[1] 50
> x[20]
[1] 100
>
> #c. Find the sum,mean,meadian and standard deviation of vector x
> s<-sum(x)
> s
[1] 1050
> m<-mean(x)
> m
[1] 52.5
> md<-median(x)
> md
[1] 52.5
> sd<-sd(x)
> sd
[1] 29.5804
>
> #d. Create another vector with name "color" and print its values
> color<-c("red","blue","black")
>          color
[1] "red"   "blue"  "black"
>
>          #e. Process 7th and 8th location of vector color
>          color[7]
```

```
> #d. Create another vector with name "color" and print its values
> color<-c("red","blue","black")
>          color
[1] "red"   "blue"  "black"
>
>          #e. Process 7th and 8th location of vector color
>          color[7]
[1] NA
>          color[8]
[1] NA
>
>          #f. Repeat values in color vector excatly twice
>          a<-rep(color,each=2)
>          a
[1] "red"   "red"   "blue"  "blue"  "black" "black"
>
>          #g. Access multiple elements simultenously in color vector
>          color[2]
[1] "blue"
>          color[4]
[1] NA
>          for(x in color){
+            print(x)
+          }
[1] "red"
[1] "blue"
[1] "black"
>
```

Program-5

P-5 Create a list of students name and perform the following operations:

a). Access the elements of the list
b). Change the item value of 2nd and 3rd location
c). Find out the length of list
d). Check if item is exists or not
e). Add new student name in list
f). Access the element of a list through loop
g). Make another list of student and merge it with original list

```
#Name : Jyoti Devi
#Roll no. : 200010130051
#Class : B.tech(cse),g3

#Code:-
l1<-list("jyoti","meenu","kirti","sahil","rohit")
l2<-list("sonam","deepak","rahul","aman","sonia")
l1
l2

#a. Access the elements of the list
l1[1] #accessing the first element of l1
l2[3] #accessing the third element of l2

#b. Change the item value of 2nd and 3rd location
l1[2]="monia"
l1[3]="sheetal"
l1

l2[2]="seeta"
l2[3]="kanchan"
l2

#c. Find out the length of list
len1<-length(l1)
len1
len2<-length(l2)
len2

#d. Check if item is exists or not
"jyoti" %in% l1 #true
"kanchan" %in% l2 #true

#e. Add new student name in list
append(l1,"sunil",after=2)
l1

#f Access the element of a list through loop
```

```
for(x in l1)
{
  print(x)
}

#g Make another list of student and merge it with original list
l3<-list("joya","mini","priya","khusi")
#merging l1 with new list
comboList1<-c(l1,l3)
comboList1

#merging l2 with new list
combolist2<-c(l2,l3)
combolist2
```

Output:-

```
Console  Terminal  Jobs
R 4.2.2 · ~/
> #Name : Jyoti devi
> #Roll no. : 200010130051
> #Class : B.tech(cse),G-3
>
>
> #P-5 Create a list of student name and perform the following operations.
> l1<-list("jyoti","meenu","kirti","sahil","rohit")
> l2<-list("sonam","deepak","rahul","aman","sonia")
> l1
[[1]]
[1] "jyoti"

[[2]]
[1] "meenu"

[[3]]
[1] "kirti"

[[4]]
[1] "sahil"

[[5]]
[1] "rohit"

> l2
[[1]]
[1] "sonam"

[[2]]
[1] "deepak"

[[3]]
[1] "rahul"

[[4]]
[1] "aman"

[[5]]
[1] "sonia"

>
```

```
>
> #a. Access the elements of the list
> l1[1] #accessing the first element of l1
[[1]]
[1] "jyoti"

> l2[3] #accessing the third element of l2
[[1]]
[1] "rahul"

>
> #b. Change the item value of 2nd and 3rd location
> l1[2]="monia"
> l1[3]="sheetal"
> l1
[[1]]
[1] "jyoti"

[[2]]
[1] "monia"

[[3]]
[1] "sheetal"

[[4]]
[1] "sahil"

[[5]]
[1] "rohit"

>
> l2[2]="seeta"
> l2[3]="kanchan"
> l2
[[1]]
[1] "sonam"

[[2]]
[1] "seeta"
```

```
[[2]]
[1] "seeta"

[[3]]
[1] "kanchan"

[[4]]
[1] "aman"

[[5]]
[1] "sonia"

>
> #c. Find out the length of list
> len1<-length(l1)
> len1
[1] 5
> len2<-length(l2)
> len2
[1] 5
>
> #d. Check if item is exists or not
> "jyoti" %in% l1 #true
[1] TRUE
> "kanchan" %in% l2 #true
[1] TRUE
>
> #e. Add new student name in list
> append(l1,"sunil",after=2)
[[1]]
[1] "jyoti"

[[2]]
[1] "monia"

[[3]]
[1] "sunil"

[[4]]
[1] "sheetal"
```

```
[[5]]
[1] "sahil"

[[6]]
[1] "rohit"

> l1
[[1]]
[1] "jyoti"

[[2]]
[1] "monia"

[[3]]
[1] "sheetal"

[[4]]
[1] "sahil"

[[5]]
[1] "rohit"

>
> #f Access the element of a list through loop
> for(x in l1)
+ {
+   print(x)
+ }
[1] "jyoti"
[1] "monia"
[1] "sheetal"
[1] "sahil"
[1] "rohit"
>
> #g Make another list of student and merge it with original list
> l3<-list("joya","mini","priya","khusi")
> #merging l1 with new list
> comboList1<-c(l1,l3)
> comboList1
[[1]]
```

Program-6

P-6 Use plot(iris) function and interpret the output write down your findings about the data set.

#Name : Jyoti Devi
#Roll no. : 200010130051
#Class : B.tech(cse),g3

The iris dataset is one of the most popular datasets used in machine learning and data science, which contains measurements of the sepal length, sepal width, petal length, and petal width of three different species of iris flowers: Setosa, Versicolor, and Virginica.

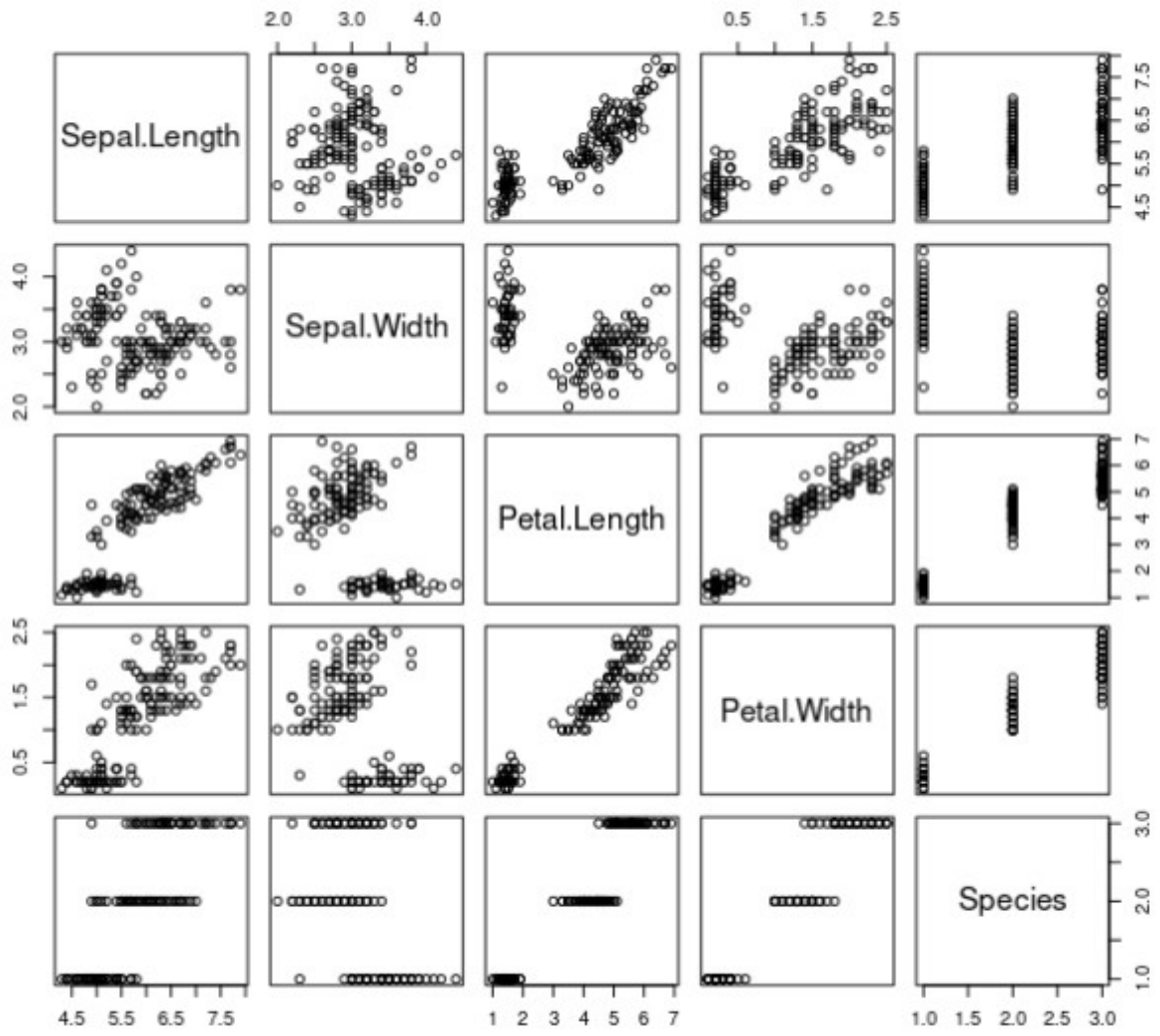
Here are some facts about the iris dataset in R language:

1. The iris dataset is built into R and can be loaded using the command `data(iris)`.
2. The iris dataset consists of 150 observations and 5 variables (4 predictor variables and 1 response variable).
3. The 4 predictor variables in the dataset are: Sepal Length, Sepal Width, Petal Length, and Petal Width.
4. The response variable is the species of the iris flower: Setosa, Versicolor, or Virginica.
5. The dataset is often used for classification tasks in machine learning, where the goal is to train a model to predict the species of an iris flower based on its measurements.
6. The iris dataset is a well-known example of linearly separable data, which means that it is possible to draw a straight line or hyperplane that can separate the three species with a high degree of accuracy.
7. The iris dataset is commonly used for teaching and learning purposes in data science and machine learning, as it is relatively small, easy to understand, and provides a good introduction to basic data analysis and visualization techniques.
8. The iris dataset has been widely used in research on pattern recognition, statistical classification, and machine learning algorithms.

#Code:-

```
plot(iris)
```

Output:-


Sepal.Length
Sepal.Width
Petal.Length
Petal.Width
Species