**Roll No:- 48**

**Assignment No:-05**

**Assignment Name:- Write program for validating and exploring data manipulation (Summerizing, Sorting,Subsetting,Merging,Joining).**

**#Validating data:-**

data(cars)

head(cars, 3)

library(validate)

rules <- validator(speed >= 0,

dist >= 0,

speed/dist <= 1.5,

cor(speed, dist)>=0.2)

out <- confront(cars, rules)

summary(out)

**Output: -**

data(cars)

> head(cars, 3)

speed dist

1 4 2

2 4 10

3 7 4

>

>

> library(validate)

> rules <- validator(speed >= 0,

+ dist >= 0,

+ speed/dist <= 1.5,

+ cor(speed, dist)>=0.2)

> out <- confront(cars, rules)

> summary(out)

name items passes fails nNA error warning expression

1 V1 50 50 0 0 FALSE FALSE speed - 0 >= -1e-08

2 V2 50 50 0 0 FALSE FALSE dist - 0 >= -1e-08

3 V3 50 48 2 0 FALSE FALSE speed/dist <= 1.5

4 V4 1 1 0 0 FALSE FALSE cor(speed, dist) >= 0.2

**1)Summarizing:-**

#create a data frame

data1<-data.frame(player=c('A','B','c','D','E'),

runs=c(100,200,105,50,90),

wickets=c(15,20,8,5,8)

)

data1

#summarize method

summarize(data1,sum(runs),mean(runs),mode(wickets))

//summarize(data1)

------------------------------------------------------------------------------------------------------------------------------

**2)Sorting:-**

#creating data frame

dataBook=data.frame(Customers=c("Ruhi","James","Heera","Shubham","Joe","Priya"),

Products=c("ProdA","ProdB","ProdC","ProdD","ProdE","prodF"),

Salary=c(500,600,450,700,300,400))

dataBook

#sorting the data frame in ascending order

arrange(dataBook,Salary)

#sorting the data frame in descending order

dataBook%>%arrange(desc(Salary))

------------------------------------------------------------------------------------------------------------------------------

**3)Subsetting:-**

#Subsetting in R using []operator:

#create vector

x<-1:15

cat("Original vector:",x,"\n")

#subsetting vector:

cat("First 5 values of vector:",x[1:5],"\n")

cat("Without values present at index 1,2and 3",x[-c(1,2,3),"\n"])

#Subsetting in R using [[]]operator:

#create list:

ls<-list(a=1,b=2,c=10,d=20)

cat("Original List:\n")

print(ls)

#select first element of list:

cat("Element of list:",ls[[3]],"\n")

#Subsetting using c() function:

ls2<-list(a=list(x=1,y="students"),b=1:10)

ls2

cat("Using c() function:\n")

//print(ls2[[c(1,2)]])

//print(ls2[[1]][[2]])

#Subsetting Using $ operator:

ls3<-list(a="Roshani",b=1,c="Hello")

ls3

cat("Using $ operator:\n")

print(ls3$a)

------------------------------------------------------------------------------------------------------------------------------------

**4)Merging: -**

#Merge DataFrames by Row Names:-

data\_frame1<-data.frame(No=c(1:5),

Name=letters[1:5],

Salary=c(200,200,300,NA,300)

)

data\_frame1

data\_frame2<-data.frame(No=c(6:8),

Name=letters[8:10],

Salary=c(400,350,NA)

)

data\_frame2

data\_frame\_merge<-merge(data\_frame1,data\_frame2,by='row.names',all=TRUE)

print("Merged DataFrame")

print(data\_frame\_merge)

------------------------------------------------------------------------------------------------------------------------------------

**5)Joining:-**

#Using Inner join:-

data1<-data.frame(ID=c(1:5))

data2<-data.frame(ID=c(4:8))

inner\_join(data1,data2,by="ID")

#Using Left join:-

data1<-data.frame(ID=c(1:5),

Name=c("Rutuja","Lokesh","Ram","Purvi","Nita"))

data2<-data.frame(ID=c(4:8),

Marks=c(70,85,80,90,75))

left\_join(data1,data2,by="ID")

------------------------------------------------------------------------------------------------------------------------------------

**OUTPUT: -**

#1)Summarizing:-

> #create a data frame

> data1<-data.frame(player=c('A','B','c','D','E'),

+ runs=c(100,200,105,50,90),

+ wickets=c(15,20,8,5,8)

+ )

> data1

player runs wickets

1 A 100 15

2 B 200 20

3 c 105 8

4 D 50 5

5 E 90 8

> #summarize method

> summarize(data1,sum(runs),mean(runs),mode(wickets))

sum(runs) mean(runs) mode(wickets)

1 545 109 numeric

> #-------------------------------------------------------------------------------

> #2)Sorting:-

> #creating data frame

> dataBook=data.frame(Customers=c("Ruhi","James","Heera","Shubham","Joe","Priya"),

+ Products=c("ProdA","ProdB","ProdC","ProdD","ProdE","prodF"),

+ Salary=c(500,600,450,700,300,400))

> dataBook

Customers Products Salary

1 Ruhi ProdA 500

2 James ProdB 600

3 Heera ProdC 450

4 Shubham ProdD 700

5 Joe ProdE 300

6 Priya prodF 400

> #sorting the data frame in ascending order

> arrange(dataBook,Salary)

Customers Products Salary

1 Joe ProdE 300

2 Priya prodF 400

3 Heera ProdC 450

4 Ruhi ProdA 500

5 James ProdB 600

6 Shubham ProdD 700

> #sorting the data frame in descending order

> dataBook%>%arrange(desc(Salary))

Customers Products Salary

1 Shubham ProdD 700

2 James ProdB 600

3 Ruhi ProdA 500

4 Heera ProdC 450

5 Priya prodF 400

6 Joe ProdE 300

> #-------------------------------------------------------------------------------

> #3)Subsetting:-

> #Subsetting in R using []operator:

> #create vector

> x<-1:15

> cat("Original vector:",x,"\n")

Original vector: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

> #subsetting vector:

> cat("First 5 values of vector:",x[1:5],"\n")

First 5 values of vector: 1 2 3 4 5

> cat("Without values present at index 1,2and 3:",x[-c(1,2,3)])

Without values present at index 1,2and 3: 4 5 6 7 8 9 10 11 12 13 14 15> #Subsetting in R using [[]]operator:

> #create list:

> ls<-list(a=1,b=2,c=10,d=20)

> cat("Original List:\n")

Original List:

> print(ls)

$a

[1] 1

$b

[1] 2

$c

[1] 10

$d

[1] 20

> #select first element of list:

> cat("Element of list:",ls[[3]],"\n")

Element of list: 10

> #Subsetting using c() function:

> ls2<-list(a=list(x=1,y="students"),b=1:10)

> ls2

$a

$a$x

[1] 1

$a$y

[1] "students"

$b

[1] 1 2 3 4 5 6 7 8 9 10

> cat("Using c() function:\n")

Using c() function:

> print(ls2[[c(1,2)]])

[1] "students"

> print(ls2[[1]][[2]])

[1] "students"

> #Subsetting Using $ operator:

> ls3<-list(a="Roshani",b=1,c="Hello")

> ls3

$a

[1] "Roshani"

$b

[1] 1

$c

[1] "Hello"

> cat("Using $ operator:\n")

Using $ operator:

> print(ls3$a)

[1] "Roshani"

> #----------------------------------------------------------------------------------

> #4)Merging:-

> #Merge DataFrames by Row Names:-

> data\_frame1<-data.frame(No=c(1:5),

+ Name=letters[1:5],

+ Salary=c(200,200,300,NA,300)

+ )

> data\_frame1

No Name Salary

1 1 a 200

2 2 b 200

3 3 c 300

4 4 d NA

5 5 e 300

>

> data\_frame2<-data.frame(No=c(6:8),

+ Name=letters[8:10],

+ Salary=c(400,350,NA)

+ )

> data\_frame2

No Name Salary

1 6 h 400

2 7 i 350

3 8 j NA

>

> data\_frame\_merge<-merge(data\_frame1,data\_frame2,by='row.names',all=TRUE)

> print("Merged DataFrame")

[1] "Merged DataFrame"

> print(data\_frame\_merge)

Row.names No.x Name.x Salary.x No.y Name.y Salary.y

1 1 1 a 200 6 h 400

2 2 2 b 200 7 i 350

3 3 3 c 300 8 j NA

4 4 4 d NA NA <NA> NA

5 5 5 e 300 NA <NA> NA

> #-------------------------------------------------------------------------------

> #5)Joining:-

> #Using Inner join:-

> data1<-data.frame(ID=c(1:5))

> data2<-data.frame(ID=c(4:8))

> inner\_join(data1,data2,by="ID")

ID

1 4

2 5

>

> #Using Left join:-

> data1<-data.frame(ID=c(1:5),

+ Name=c("Rutuja","Lokesh","Ram","Purvi","Nita"))

> data2<-data.frame(ID=c(4:8),

+ Marks=c(70,85,80,90,75))

> left\_join(data1,data2,by="ID")

ID Name Marks

1 1 Rutuja NA

2 2 Lokesh NA

3 3 Ram NA

4 4 Purvi 70

5 5 Nita 85