BIG DATA ANALYTICS LAB

Assignment-3

Name- Ajit Singhal

Course-B.Tech(CS)

Section-A1

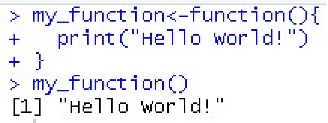
Roll No.- 04

University Roll No.-201500052

Functions, Data Frames & R Factors using R Programming Language

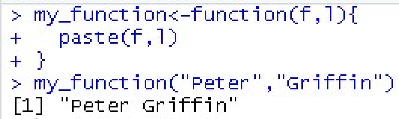
1) Write a R program to perform below operations using R User-Defined Functions –

(a) Creating and Calling User-Define Function



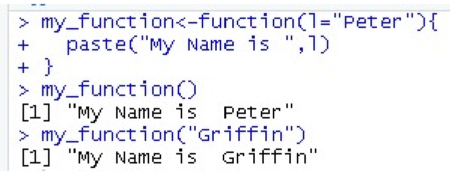
```
> my_function<-function(){
+   print("Hello world!")
+ }
> my_function()
[1] "Hello world!"
```

(b) Number of Arguments



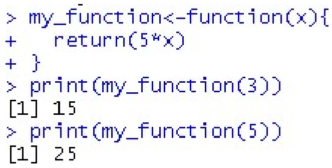
```
> my_function<-function(f,l){
+   paste(f,l)
+ }
> my_function("Peter","Griffin")
[1] "Peter Griffin"
```

(c) Default Arguments



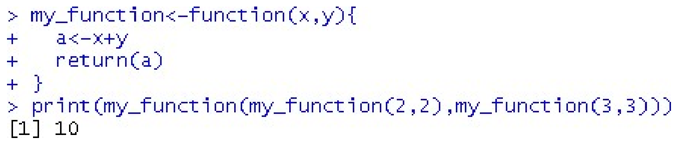
```
> my_function<-function(l="Peter"){
+   paste("My Name is ",l)
+ }
> my_function()
[1] "My Name is  Peter"
> my_function("Griffin")
[1] "My Name is  Griffin"
```

(d) Return Values



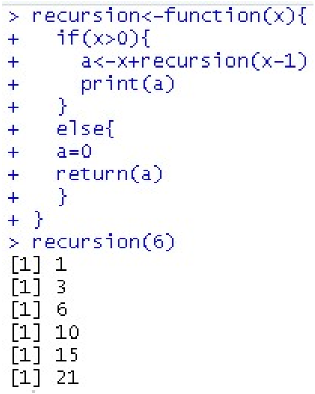
```
> my_function<-function(x){
+   return(5*x)
+ }
> print(my_function(3))
[1] 15
> print(my_function(5))
[1] 25
```

(e) Nested Functions



```
> my_function<-function(x,y){
+   a<-x+y
+   return(a)
+ }
> print(my_function(my_function(2,2),my_function(3,3)))
[1] 10
```

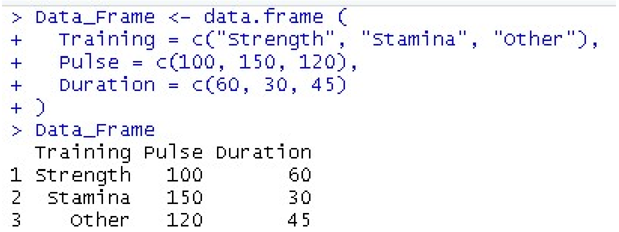
(f) Recursion



```
> recursion<-function(x){
+   if(x>0){
+     a<-x+recursion(x-1)
+     print(a)
+   }
+   else{
+   a=0
+   return(a)
+   }
+ }
> recursion(6)
[1] 1
[1] 3
[1] 6
[1] 10
[1] 15
[1] 21
```

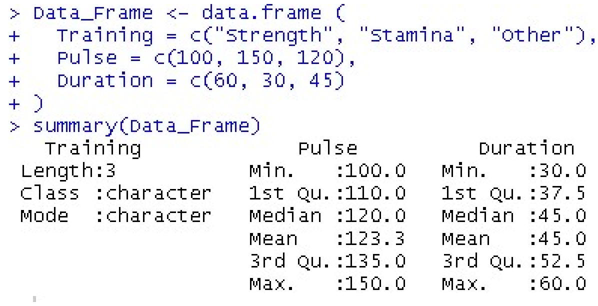
2) Write a R program to perform below operations using Data Frames –

(a) Create Data Frame



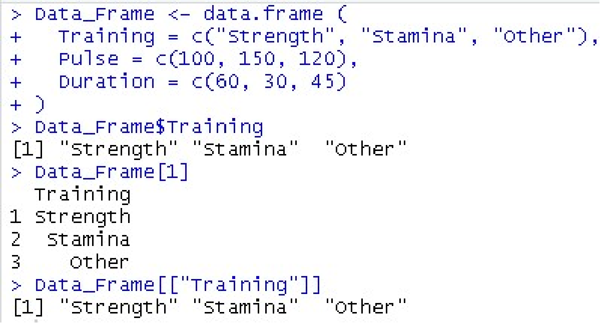
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> Data_Frame
  Training Pulse Duration
1 Strength   100       60
2  Stamina   150       30
3    Other   120       45
```

(b) Summarize the Data



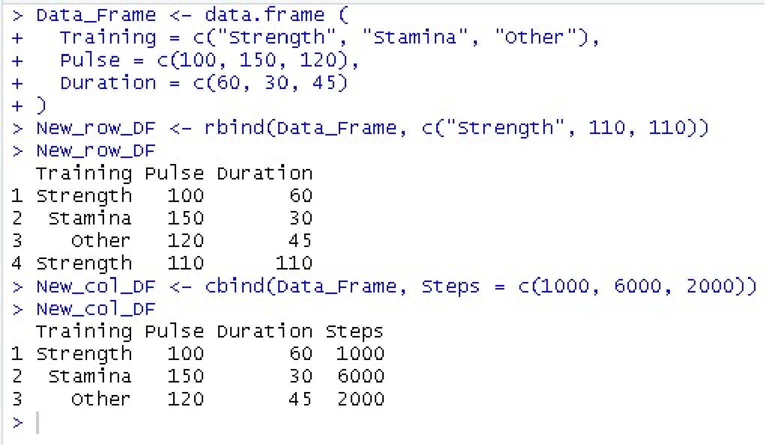
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> summary(Data_Frame)
   Training             Pulse          Duration   
 Length:3           Min.   :100.0   Min.   :30.0  
 Class :character   1st Qu.:110.0   1st Qu.:37.5  
 Mode  :character   Median :120.0   Median :45.0  
                    Mean   :123.3   Mean   :45.0  
                    3rd Qu.:135.0   3rd Qu.:52.5  
                    Max.   :150.0   Max.   :60.0  
```

(c) Access Items



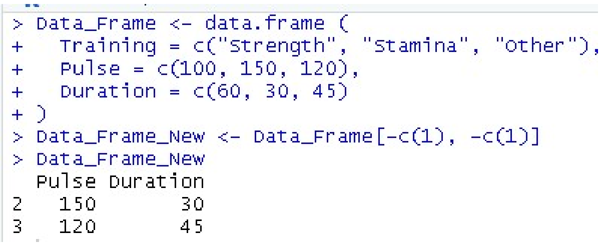
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> Data_Frame$Training
[1] "Strength" "Stamina"  "Other"
> Data_Frame[1]
  Training
1 Strength
2  Stamina
3    Other
> Data_Frame[["Training"]]
[1] "Strength" "Stamina"  "Other"
```

(d) Add Rows & Columns



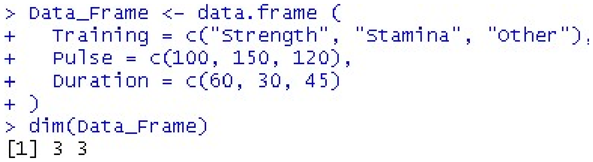
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> New_row_DF <- rbind(Data_Frame, c("Strength", 110, 110))
> New_row_DF
  Training Pulse Duration
1 Strength   100       60
2  Stamina   150       30
3    Other   120       45
4 Strength   110      110
> New_col_DF <- cbind(Data_Frame, Steps = c(1000, 6000, 2000))
> New_col_DF
  Training Pulse Duration Steps
1 Strength   100       60  1000
2  Stamina   150       30  6000
3    Other   120       45  2000
> 
```

(e) Remove Rows and Columns



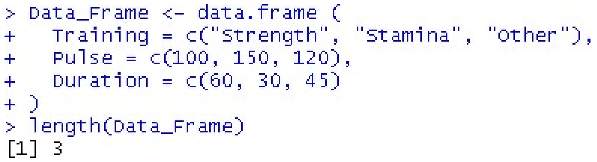
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> Data_Frame_New <- Data_Frame[-c(1), -c(1)]
> Data_Frame_New
  Pulse Duration
2   150       30
3   120       45
```

(f) Amount of Rows and Columns



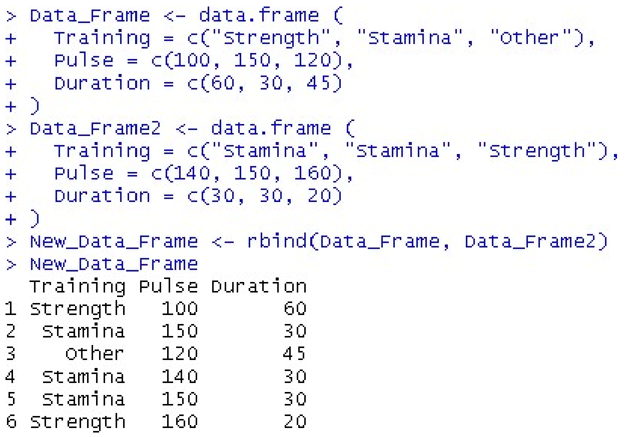
```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> dim(Data_Frame)
[1] 3 3
```

(g) Data Frame Length



```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> length(Data_Frame)
[1] 3
```

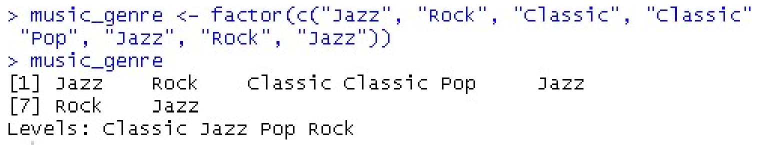
(h) Combining Data Frames



```
> Data_Frame <- data.frame (
+   Training = c("Strength", "Stamina", "Other"),
+   Pulse = c(100, 150, 120),
+   Duration = c(60, 30, 45)
+ )
> Data_Frame2 <- data.frame (
+   Training = c("Stamina", "Stamina", "Strength"),
+   Pulse = c(140, 150, 160),
+   Duration = c(30, 30, 20)
+ )
> New_Data_Frame <- rbind(Data_Frame, Data_Frame2)
> New_Data_Frame
  Training Pulse Duration
1 Strength   100       60
2  Stamina   150       30
3    Other   120       45
4  Stamina   140       30
5  Stamina   150       30
6 Strength   160       20
```

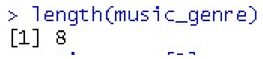
3) Write a R program to perform below operations using R Factors –

(a) Create a factor



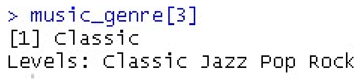
```
> music_genre <- factor(c("Jazz", "Rock", "Classic", "Classic"
 "Pop", "Jazz", "Rock", "Jazz"))
> music_genre
[1] Jazz    Rock    Classic Classic Pop     Jazz
[7] Rock    Jazz
Levels: Classic Jazz Pop Rock
```

(b) Factor Length



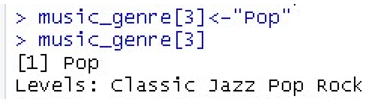
```
> length(music_genre)
[1] 8
```

(c) Access Factors



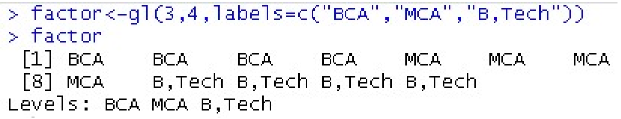
```
> music_genre[3]
[1] Classic
Levels: Classic Jazz Pop Rock
```

(d) Change Item Value



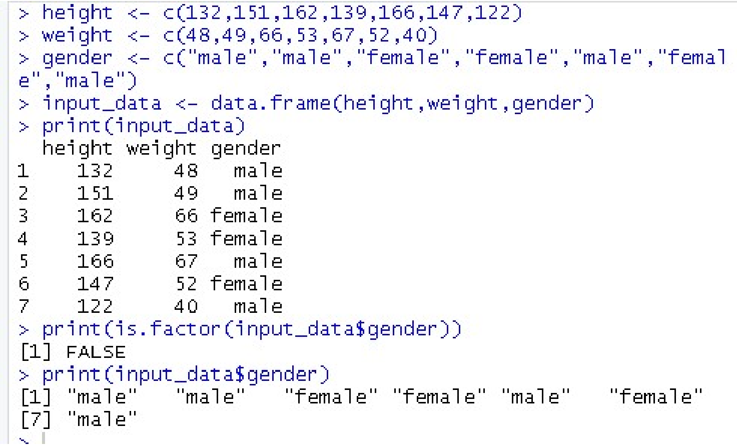
```
> music_genre[3]<-"Pop"
> music_genre[3]
[1] Pop
Levels: Classic Jazz Pop Rock
```

(e) Print the levels using levels( ) function



```
> factor<-gl(3,4,labels=c("BCA","MCA","B,Tech"))
> factor
 [1] BCA    BCA    BCA    BCA    MCA    MCA    MCA
 [8] MCA    B,Tech B,Tech B,Tech B,Tech
Levels: BCA MCA B,Tech
```

4) Create R Factors in Data Frame that prints data with a column of text into categorical form using R Factor.



```
> height <- c(132,151,162,139,166,147,122)
> weight <- c(48,49,66,53,67,52,40)
> gender <- c("male","male","female","female","male","femal
e","male")
> input_data <- data.frame(height,weight,gender)
> print(input_data)
  height weight gender
1    132     48   male
2    151     49   male
3    162     66 female
4    139     53 female
5    166     67   male
6    147     52 female
7    122     40   male
> print(is.factor(input_data$gender))
[1] FALSE
> print(input_data$gender)
[1] "male"   "male"   "female" "female" "male"   "female"
[7] "male"
>
```