# Exercise 1 :Basics of R

**1.Check if a Number is Odd or Even using R Programming.**

a=11

if(a%%2==0)

{ print("even") }else{ print("odd")

}

**2.R Program to Print the Fibonacci Sequence.** fib=numeric(n) fib[1]=0 if(n>1){

fib[2]=1

}

for(i in 3:n){

fib[i]=fib[i-1]+fib[i-2]

}

print(fib)

**3.Write an R program to concatenate 5 strings and print the length.**

s1="My"s2="name" s3="is" s4="Poornima" s5="kr”

newstring=paste(s1,s2,s3,s4,s5) print(newstring)

nchar(newstring)

**4.Write a R program to create Multiplication Table of any number**

a=4. range=1:10

for (i in range) { result=a\*i

print(paste(a,"\*",i,"=",result))

}

**5.Create an R Program to Take Input From the User (readline( ) and scan( ))** n=readline("Enteryour number:") print(num)

# Exercise 2 :R Datastructures

**1.Create an R Program to Find the Minimum and Maximum of a sequence of numbers. (Use any datastructure)**

x=c(2,5,1,7,3,8) print(min(x))

print(max(x))

**2.Create a R program that finds the maximum value in a vector using a for loop.**

myvector=c(5,2,6,1,8) maxval=myvector[1] for(i in myvector){ if(i>maxval){

maxval=i

}

}

print(maxval)

**3.Illustrate the purpose of using which() function in R with the help of suitable programs.** num=c(2,5,1,83,7) indices=which(num>5)

print(indices)

**4.Write a R program to create a list containing strings, numbers, vectors and logical values.**

**(multiple data types in a single list).**

mylist=c(c(2,3),"r",TRUE, 127.7) print(mylist)

**5.Write a R program to list containing a vector, a matrix and a list and give names to the elements in the list. [nested list].**

myvector=c("Apple","Orange","Banana") mymatrix=matrix(c(2,4,1,6),nrow=2,ncol=2) mylist=c("red","green","blue")

mynestedlist=list("fruits"=myvector,"numbers"=mymatrix,"colours"=mylist) print(mynestedlist)

**6.Write a R program to find all elements of a given list that are not in another given list.**

**(**setdiff(list1,list2**))** l1=list(1,2,3,4) l2=list(1,5,2,7,8) print(l1) print(l2) setdiff(l2,l1) setdiff(l1,l2)

**7.Write a R program to convert a given matrix to a 1 dimensional array. (Use the function as.array() and as.vector() as required)**

my\_matrix <- matrix(1:12, nrow = 3) my\_array <- as.array(my\_matrix) print("1D array using as.array():")

print(my\_array)

**8.Write a R program to create an array of two 3x3 matrices each with 3 rows and 3 columns from two given two vectors.**

v1=c(1, 3, 4, 5) v2=c(10, 11, 12, 13, 14, 15) result=array(c(v1, v2), dim = c(3, 3, 2))

print("New array:")

print(result)

**9.Write a R program to create a two-dimensional 5×3 array of sequence of even integers greater than 50.**

even\_integers=seq(52, by = 2, length.out = 15) array\_5x3=array(even\_integers, dim = c(5, 3)) print("5x3 Array of even integers greater than 50:") print(array\_5x3)

**10.R Program to Drop Columns in a Dataframe.**

my\_df=data.frame(

1. = c(1, 2, 3),
2. = c(4, 5, 6),
3. = c(7, 8, 9)

)

my\_df=subset(my\_df, select = -B) print(my\_df)

**11.R Program to Make a List of Dataframes.**

df1 <- data.frame(y1 = c(0, 1, 2), y2 = c(3, 4, 5)) df2 <- data.frame(y1 = c(6, 7, 8), y2 = c(9, 10, 11)) new\_list <- list(df1, df2) print("New list of dataframes:") print(new\_list) print("Dataframe 1:") print(new\_list[[1]]) print("Dataframe 2:")

print(new\_list[[2]])

**12.Write a R program to create a data frame from four given vectors.** name=c('Athu','Aadhi','Sree') age=c(23,22,24)

city=c('kannur','Thalassery','Payyannur') df=data.frame(name,age,city)

print(df)

**13.Write a R program to extract specific column from a data frame using column name.**

df=data.frame( col1=c(1, 2, 3), col2=c("A", "B", "C"), col3=c(10, 20, 30),

col4=c(TRUE, FALSE, TRUE)

)

selected\_df=df[c("col1", "col2", "col4")] print(selected\_df)

**14.Write a R program to sort a given data frame by multiple column(s).(Use order() )** my\_df <- data.frame( col1 = c(10, 5, 8, 12), col2 = c("A", "B", "C", "D"), col3 = c(3, 1, 2, 4)

)

sorted\_df <- my\_df[order(-my\_df$col1, my\_df$col2), ] print(sorted\_df)

**15.Write a R program to replace NA values with 3 in a given data frame. (Use is.na(dataframe) function)**

my\_df <- data.frame( col1 = c(10, NA, 8, 12), col2 = c(NA, "A", "C", NA),

col3 = c(3, 1, NA, 4)

)

print("Original data frame:") print(my\_df) my\_df[is.na(my\_df)] <- 3

print("Data frame after replacing NA values with 3:") print(my\_df)

**16.Write a R program to change a column name of a given data frame. (Try it!!!)** my\_df <- data.frame( Col1= c(100, 120, 150),

Col2= c(36, -33, 14),

Col3= c(136, 87, 164)

)

colnames(my\_df) <- c("score", "mark", "rank") print(my\_df)

**17.Write a R program to compare two data frames to find the row(s) in the first data frame that are not present in the second data frame. (setdiff(dataframe1,dataframe2)**

df\_90 <- data.frame(

"item" = c("item1", "item2", "item3"),

"a1" = c(12, 14, 12),

"a2" = c(11, 12, 15),

"a3" = c(12, 14, 15)

)

df\_91 <- data.frame(

"item" = c("item1", "item2", "item3"),

"a1" = c(12, 14, 12),

"a2" = c(11, 12, 15),

"a3" = c(12, 15, 18)

)

print("Original Dataframes:") print(df\_90) print(df\_91)

print("Row(s) in the first data frame not present in the second data frame:") print(setdiff(df\_90, df\_91))

**18.Write a R program to create a data frame using two given vectors and display the duplicated elements and unique rows of the said data frame. (duplicated() , unique())**

a <- c(10, 20, 10, 10, 40, 50, 20, 30) b <- c(10, 30, 10, 20, 0, 50, 30, 30) ab <- data.frame(a, b) print("Original data frame:") print(ab)

print("Duplicate elements of the data frame:") print(duplicated(ab))

print("Unique rows of the data frame:") print(unique(ab))

**19.Write a R program to change the first level of a factor with another level of a given factor.**

v=c("a", "b", "a") f=factor(v) levels(f)[1]="e" print(f)

**20.Write a R program to concatenate two given factors into a single factor.** fac1=as.factor(letters[1:3]) fac2=as.factor(c(1:4))

combined=factor(c(levels(fac1)[fac1], levels(fac2)[fac2])) print("Combined Factor:") print(combined)