PRESIDENCY COLLEGE
(AUTONOMOUS)

AFFILIATED TO BENGALURU CITY UNIVERSITY, APPROVED BY AICTE, DELHI & RECOGNISED BY THE GOVT. OF KARNATAKA
RE-ACCREDITED BY NAAC WITH 'A+' GRADE

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

DATA ANALYTICS LAB MANUAL -2024

BCA V SEMESTER: DATA ANALYTICS LAB MANUAL

1. Write a program to check whether a year (integer) entered by the user is a leap year or not?

2. **Write an R program to find the sum of natural numbers without formula using the if–else statement and the while loop.**

3. Write a program that prints the grades of the students according to the marks obtained. The grading of the marks should be as follows:
   # Store marks of all the subjects in an array
   # using nested if-else

Marks	Grades
800-1000	A+
700 – 800	A
500 – 700	B+
400-500	B
150 – 400	C
Less than 150	D

4. **Write an R program to make a simple calculator that can add, subtract, multiply and divide using switch cases and functions.**

5. **Write a set of instructions to create the following matrix using vectors and rbind() function. Rename the rows to Lang1, Lang2 & Lang3 respectively and use the function to access any one element using row names.**

	Rows	Columns			
		1	2	3	4
MatrixOfTechnology	1	C#	Java	Cobol	.Net
	2	JavaScript	NodeJs	R	Azure
	3	Power BI	ASP.Net	Unity	Block Chain

6. Write a program to perform searching within a list (1 to 50). If the number is found in the list, print that the search is successful otherwise print that the number is not in the list.

7. Create a list and convert to data frame that stores the marks of any three subjects for 10 students. Find out the total marks, average, maximum marks, and minimum marks of every subject.

8. Write the steps to import data from Excel to CSV files and apply data viewer functions like rm(),dim(), head(), tail(), sorting, filtering, searching to view few set of rows.

 # 1. **Write a program to check whether a year (integer) entered by the user is a leap year or not?**

Program

```
1  # 1.  Write a program to check whether a year (integer) entered by the user is a leap year or not?
2
3  year = as.integer(readline( prompt = "Enter year to be checked:  "))
4  if(year%%4 == 0 && year%%100 != 0 || year%% 400 == 0){
5    print(paste(year," is a leap year"))
6  }else{
7    print(paste(year," is not a leap year"))
8  }
9
10 |
```

Output:

```
> source("~/.active-rstudio-document")
Enter year to be checked:  2024
[1] "2024  is a leap year"
> source("~/.active-rstudio-document")
Enter year to be checked:  2021
[1] "2021  is not a leap year"
> source("~/.active-rstudio-document")
```

 # 2. **Write an R program to find the sum of natural numbers without formula**

#using the if–else statement and the while loop.

Program

```r
# 2.  Write an R program to find the sum of natural numbers without formula
#using the if-else statement and the while loop.
num = as.integer(readline(prompt = "Enter a number: "))

if(num < 0) {

  print("Enter a positive number")

} else {
  sum = 0

  # use while loop to iterate until zero
  while(num > 0) {
    sum = sum + num
    num = num - 1
  }

  print(paste("The sum of numbers up to the given limit is", sum))
```

Output:

```
> source("~/.active-rstudio-document")
Enter a number: 5
[1] "The sum of numbers up to the given limit is 15"
> source("~/.active-rstudio-document")
Enter a number: q()
```

3. **Write a program that prints the grades of the students according to the marks obtained. The grading of the marks should be as follows:**

Marks	Grades
800-1000	A+
700 – 800	A
500 – 700	B+
400-500	B
150 – 400	C
Less than 150	D

using nested if-else

#Store marks of all the subjects in an array

sub_marks <- c(57,98,59,89,78,90,57,96,45,75)

marks = sum(sub_marks)

print(paste(marks," is the total marks scored"))

if(marks >= 800 && marks <= 1000){

  grade <-'A+'

}else{

  if(marks >= 700 && marks <= 799){

   grade <-'A'

  }else {

  if(marks >= 500 && marks <= 700){

  grade <-'B+'

  }else{

  if(marks >= 400 && marks <= 500){

  grade <-'B'

  } else{

   if(marks >= 150 && marks <= 400){

```
        grade <-'C'

    } else{

    if(marks < 150){

      grade <-'D'

    }

    }

   }

  }

 }

}
```

print(paste(grade, "is the grade secured by the student"))

<mark>output:</mark>

```
> source("~/.active-rstudio-document")
[1] "744  is the total marks scored"
[1] "A is the grade secured by the student"
```

4. **Write an R program to make a simple calculator that can add, subtract, multiply and divide using switch cases and functions.**

Program make a simple calculator that can add, subtract, multiply and divide using functions

```
add <- function(x, y) {

  return(x + y)

}

subtract <- function(x, y) {

  return(x - y)

}

multiply <- function(x, y) {

  return(x * y)

}

divide <- function(x, y) {

  return(x / y)

}

# take input from the user

print("Select operation.")

print("1.Add")

print("2.Subtract")

print("3.Multiply")

print("4.Divide")

choice = as.integer(readline(prompt="Enter choice[1/2/3/4:] "))

num1 = as.integer(readline(prompt="Enter first number: "))

num2 = as.integer(readline(prompt="Enter second number: "))
```

operator <- switch(choice,"+","-","*","/")

result <- **switch(choice, add(num1, num2), subtract(num1, num2), multiply(num1, num2), divide(num1, num2))**

print(paste(num1, operator, num2, "=", result))

output:

```
[1] "Select operation."
[1] "1.Add"
[1] "2.Subtract"
[1] "3.Multiply"
[1] "4.Divide"
Enter choice[1/2/3/4:] 2
Enter first number: 6
Enter second number: 7
[1] "6 - 7 = -1"
```

5. **Write a set of instructions to create the following matrix using vectors and rbind() function. Rename the rows to Lang1, Lang2 & Lang3 respectively and use the function to access any one element using row names.**

	Rows	Columns			
		1	2	3	4
	1	C#	Java	Cobol	.Net
MatrixOfTechnology	2	JavaScript	NodeJs	R	Azure
	3	Power BI	ASP.Net	Unity	Block Chain

```
# Create vectors for each row
a<-c('c#','java','COBOL','.Net')

b<-c('javascript','Nodejs','R','Azure')
c<-c('Power BI','ASP.Net','Unity','Block chain')

# create matrix using rbind() function
x<-rbind(a,b,c)
print(x)

#rename the rows using rownames() function
rname<-c('Lang1','Lang2','Lang3')
rownames(x)<-rname
print(x)

# Access the 3rd element in the second row
cat("\n\n The third element in the second row is \n")
print(x["Lang2",3])
```

output:

```
> source("~/Prog5.R")
  [,1]          [,2]      [,3]    [,4]
a "c#"          "java"    "COBOL" ".Net"
b "javascript" "Nodejs"  "R"     "Azure"
c "Power BI"    "ASP.Net" "Unity" "Block chain"
```

```
        [,1]          [,2]      [,3]    [,4]
Lang1 "c#"          "java"    "COBOL" ".Net"
Lang2 "javascript" "Nodejs"  "R"     "Azure"
Lang3 "Power BI"    "ASP.Net" "Unity" "Block chain"


 The third element in the second row is
Lang2
  "R"
>
```

6. Write a program to perform searching within a list (1 to 50). If the number is found in the list, print that the search is successful otherwise print that the number is not in the list.

```
mydata<-list(1:50)
ele<-as.integer(readline(prompt = "enter element to be searched"))
data<-unlist(mydata)
if(ele %in% data){
  print(" element is found")
} else {
  print("element is not found")
}
```

```
enter element to be searched55
[1] "element is not found"
> source("~/PROG6.R")
enter element to be searched50
[1] " element is found"
```

7. **Create a list and data frame that stores the marks of any three subjects for 10 students.Find out the total marks, average, maximum marks, and minimum marks of every subject.**

```
# create list
mylist <- list(
  Name = c("John", "Mary", "Bob", "Alice", "David", "Linda", "Sarah", "Tom", "Emily", "Kevin"),
  Marks_sub1 = c(85, 78, 92, 91, 80, 87, 89, 75, 95, 83),
  Marks_sub2 = c(55, 78, 40, 91, 80, 87, 50, 75, 97, 83),
  Marks_sub3 = c(85, 78, 92, 91, 50, 87, 60, 75, 100, 83)
)

# Converting list to dataframe
student <- as.data.frame(mylist)
print(student)

# display max,min,mean and total of a subject Marks

print(paste("Subject1 details","Max",max(student$Marks_sub1),
      "Min",min(student$Marks_sub1),
      "Average",mean(student$Marks_sub1),
      "Total",sum(student$Marks_sub1)))

print(paste("Subject1 details","Max",max(student$Marks_sub2),
      "Min",min(student$Marks_sub2),
      "Average",mean(student$Marks_sub2),
      "Total",sum(student$Marks_sub2)))

print(paste("subject3 details","Max",max(student$Marks_sub3),
      "Min",min(student$Marks_sub3),
      "Average",mean(student$Marks_sub3),
      "Total",sum(student$Marks_sub3)))
```

output:

```
  Name Marks_sub1 Marks_sub2 Marks_sub3
1 John        85         55         85
2 Mary        78         78         78
```

```
3   Bob     92     40     92
4   Alice   91     91     91
5   David   80     80     50
6   Linda   87     87     87
7   Sarah   89     50     60
8   Tom     75     75     75
9   Emily   94     94     94
10  Kevin   83     83     83
```

```
[1] "Subject1 details Max 95 Min 75 Average 85.5 Total 855"
[1] "Subject2 details Max 97 Min 40 Average 73.6 Total 736"
[1] "subject3 details Max 100 Min 50 Average 80.1 Total 801"
```

8. **Write the steps to import data from Excel to CSV files and apply data viewer functions like rm(),dim(), head(), tail(), sorting, filtering, searching to view few set of rows.**

Note: create an excel file by name studentmarks.xlsx

sl.no	studentname	marks
1	VIJAY A	67
2	VISHWAM DOSHI	56
3	VIVEK DUTTA	78
4	YASSIR AHMED	68
5	ZOHRA FATHIMA	99
6	WARIS KHAN	67
7	YASHWANTH BABU J M	56
8	VEDANT KRISHNAKANT SALVI	78
9	VENKATANARIGARI HARIKISHAN	65
10	ZOYA MEHAK D	68

```
# STEPS TO READ AN EXCEL FILE
install.packages("readxl")
library(readxl)
newdat<-read_xlsx("C:/Users/welcome/Desktop/studentmarks.xlsx")
```

View(newdat)

2. # Command to be executed INDIVIDUALY AND WRITE THE OUTPUT
a.dim(newdat)
b.head(newdat)
c.tail(newdat)

3.# FILTERING DATA BASED ON CONDITION MARKS>70
df1 <- newdat[newdat$marks > 70,]
print(df1)

4. # SORTING DATA BASED ON MARKS
df2 <- newdat[order(newdat$marks),]
print(df2)

5. # SORTING DATA BASED ON STUDENTNAME

df3 <- newdat[order(newdat$studentname),]

print(df3)

Output:

2a. `> dim(newdat)`
`[1] 10  3`

2b. `> head(newdat)`

```
1     1        VIJAY A    67
2     2 VISHWAM DOSHI    56
3     3    VIVEK DUTTA    78
4     4   YASSIR AHMED    68
5     5 ZOHRA FATHIMA    99
6     6     WARIS KHAN    67
```

2c. `> tail(newdat)`
```
   sl.no               studentname marks
5     5              ZOHRA FATHIMA    99
6     6                 WARIS KHAN    67
7     7           YASHWANTH BABU J M    56
8     8      VEDANT KRISHNAKANT SALVI    78
9     9 VENKATANARIGARI HARIKISHAN    65
```

PRESIDENCY COLLEGE
(AUTONOMOUS)
AFFILIATED TO BENGALURU CITY UNIVERSITY, APPROVED BY AICTE, DELHI & RECOGNISED BY THE GOVT. OF KARNATAKA
RE-ACCREDITED BY NAAC WITH 'A+' GRADE

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

DATA ANALYTICS LAB MANUAL -2024

```
10    10              ZOYA MEHAK D    68

> df1 <- newdat[newdat$marks > 70, ]
> print(df1)
  sl.no               studentname marks
3     3                VIVEK DUTTA    78
5     5              ZOHRA FATHIMA    99
8     8 VEDANT KRISHNAKANT SALVI    78


df2 <- newdat[order(newdat$studentname), ]
> print(df2)
   sl.no               studentname marks
8      8     VEDANT KRISHNAKANT SALVI    78
9      9 VENKATANARIGARI HARIKISHAN    65
1      1                    VIJAY A    67
2      2              VISHWAM DOSHI    56
3      3                VIVEK DUTTA    78
6      6                WARIS KHAN    67
7      7           YASHWANTH BABU J M    56
4      4                YASSIR AHMED    68
5      5              ZOHRA FATHIMA    99
10    10              ZOYA MEHAK D    68


df3 <- newdat[order(newdat$studentname), ]
> print(df3)
   sl.no               studentname marks
8      8     VEDANT KRISHNAKANT SALVI    78
9      9 VENKATANARIGARI HARIKISHAN    65
1      1                    VIJAY A    67
2      2              VISHWAM DOSHI    56
3      3                VIVEK DUTTA    78
6      6                WARIS KHAN    67
7      7           YASHWANTH BABU J M    56
4      4                YASSIR AHMED    68
5      5              ZOHRA FATHIMA    99
10    10              ZOYA MEHAK D    68
```

PART B PROGRAMS

1. **Create a csv file with 10 students name and marks. Find mean and median for the 10 student marks.**

Solution:
```
setwd('C:/Users/welcome/Desktop')

x<-read.csv("marks.csv")
print(x)

m<-mean(x$marks)
cat("\n\nMean of Student marks is\t\t", m)

med<-median(x$marks)
cat("\n\nMedain of Student marks is\t\t", med)
```

output:

```
    sl.no                  studentname marks
1       1    VEDANT KRISHNAKANT SALVI    67
2       2 VENKATANARIGARI HARIKISHAN    56
3       3                    VIJAY A    78
4       4               VISHWAM DOSHI    68
5       5                VIVEK DUTTA    99
6       6                 WARIS KHAN    67
7       7           YASHWANTH BABU J M    56
8       8               YASSIR AHMED    78

 Mean of Student marks is              70.2

 Medain of Student marks is            67.5
```

2. **Create a csv file which contains 6 product names(product) and the units sold. Draw a pie chart for the data.**

Solution:
```
setwd('E:/Data Analytics')

data<-read.csv('laptopsales.csv')
print(data)

#Draw pie chart
pie(data$unit,data$Product,main='Laptop Sale')
```

#Draw pie chart with percentage
perc<- round(100 * data$unit / sum(data$unit), 1)

pie(data$unit,labels = perc,main='Laptop Sale')

Output:
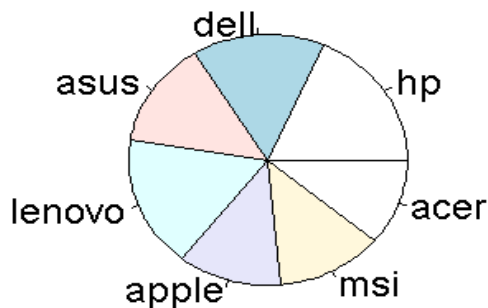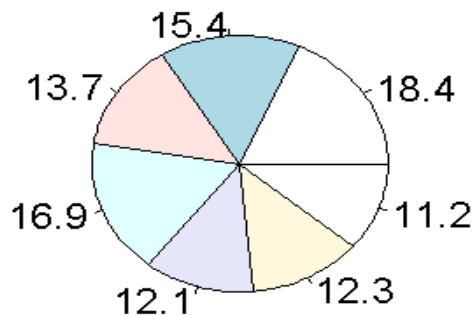
3. **Write a R program to find variance, standard deviation for a given set of data. X=(25,34,22,56,64,46,53,31,26)**

Solution:
```
x<-c(25,34,22,56,64,46,53,31,26)
cat('\nThe list of values are:\n',x)

#calculate variance using var()
v<-var(x)
cat("\n\n Variance:\t\t",v)

#calculate standards deviation using sd()
s<-sd(x)
```

cat("\n Standard Deviation:\t",s)

output:

```
The list of values are:
 25 34 22 56 64 46 53 31 26

 Variance:               237.25
 Standard Deviation:     15.40292
```

4. Write a R program to Program to find the multiplication
 # table (from 1 to 10)
# of a number input by the user

 # take input from the user
num = as.integer(readline(prompt = "Enter a number: "))

 # use for loop to iterate 10 times
for(i in 1:10) {
   print(paste(num,'x', i, '=', num*i))

5. Write a R program to concatenate two strings

create two strings

string1 <- "Programiz"

string2 <- "Pro"

using paste() to concatenate two strings
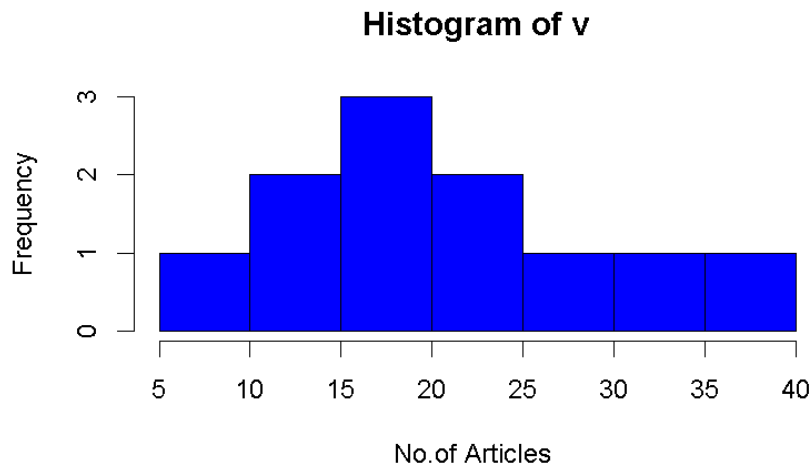
result = paste(string1, string2)

print(result)

6. Write a R program to prepare a Histogram for the given set of data.

Solution:
```
# Create data for the graph.
v <- c(19, 23, 11, 5, 16, 21, 32,
     14, 19, 27, 39)

# Create the histogram.
hist(v, xlab = "No.of Articles ",
    col = "blue", border = "black")
```

Output:

Histogram of v

7. Write a R program to prepare a bar chart for the given set of data. Create a csv file for monthlysales which contains two columns months(Jan,Feb,Mar,Apr,May) and Profit(72,80,87,90,89)

Solution:
```
setwd('E:/Data Analytics')

data<-read.csv('monthlysales.csv')
print(data)

#create a bar chart
barplot(data$profit,names.arg = data$Month,
     xlab = "Month", ylab = "Profit",
     col='blue',
     main ="Monthly Sales")
```
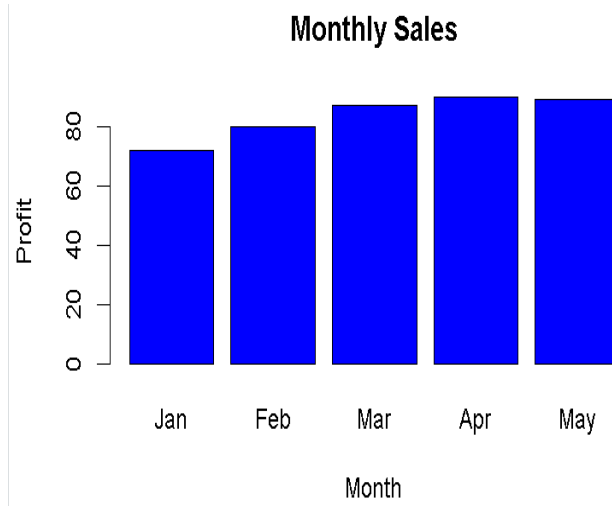
8.Write a R program to prepare a scatter chart for the given set of data by finding correlation of study_hours and exam_sxores and also draw abline

study_hours	5	7	3	8	6	9
exam_scores	80	85	60	90	75	95

```
 # Sample data
study_hours <- c(5, 7, 3, 8, 6, 9)
exam_scores <- c(80, 85, 60, 90, 75, 95)

# Calculate Pearson correlation
correlation <- cor(study_hours, exam_scores)
print(round(correlation, 2))

# Visualize the data and correlation
plot(study_hours, exam_scores, main = "Scatterplot of Study Hours vs. Exam
Scores")

# Add regression line
abline(lm(exam_scores ~ study_hours), col = "red")
text(4, 90, paste("Correlation: ", round(correlation, 2)))
```

Scatterplot of Study Hours vs. Exam Scores

n-gl.com