

Sub : Foundations for Data Analytics

Name: Gudi Varaprasad

Reg.No: 19BCE7048

Slot No: L33+L34

Assignment 1:

Date: 02/02/2021

Q1. Practice Some basic calculations.

Log of 3

> log(3)

```
Console Terminal x Jobs x
E:\VITAP\19BCE7048\Semester_4\Foundation for Data Analytics\LAB\LAB1/
> # Q1. Practice some basic calculations :
>
> # Log of 3 =
> log(3)
[1] 1.098612
> |
```

Square root of 121

> sqrt(121)

```
> # Square root of 121
> sqrt(121)
[1] 11
> |
```

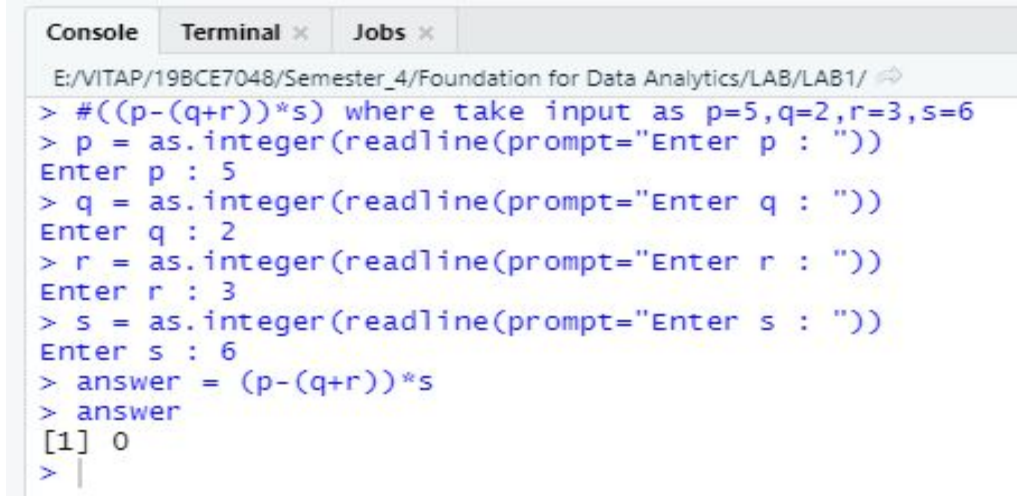
Power : $((p-(q+r))*s)$ where take input as $p=5, q=2, r=3, s=6$

```
> p = as.integer(readline(prompt="Enter p = "))
```

```
> q = as.integer(readline(prompt="Enter q = "))
```

```
> r = as.integer(readline(prompt="Enter r = "))
```

```
> s = as.integer(readline(prompt="Enter s = "))
```

A screenshot of a R console window. The window has tabs for 'Console', 'Terminal', and 'Jobs'. The 'Console' tab is active. The path 'E:/VITAP/198CE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/' is shown at the top. The code being executed is:

```
> #((p-(q+r))*s) where take input as p=5,q=2,r=3,s=6
> p = as.integer(readline(prompt="Enter p : "))
Enter p : 5
> q = as.integer(readline(prompt="Enter q : "))
Enter q : 2
> r = as.integer(readline(prompt="Enter r : "))
Enter r : 3
> s = as.integer(readline(prompt="Enter s : "))
Enter s : 6
> answer = (p-(q+r))*s
> answer
[1] 0
> |
```

Q2. Abhisekh is buying the number of baskets where each basket contains n number of eggs. Take input of the number of baskets, the number of eggs in each basket and the cost of each egg. Write a R program for calculating the cost of the total number of eggs.

```
> numberOfBaskets = as.integer(readline(prompt="Enter Number of Baskets :
"))
```

```
> numberOfEggs = as.integer(readline(prompt="Enter Number of Eggs in
each basket : "))
```

```
> costOfEgg = as.integer(readline(prompt="Enter cost of each Egg : "))
```

```
> totalCostOfEggs = numberOfBaskets * numberOfEggs * costOfEgg
```

```
> print(paste("Total Cost of Eggs = ", totalCostOfEggs))
```

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/ ↗
> numberOfBaskets = as.integer(readline(prompt="Enter Number of Baskets : "))
Enter Number of Baskets : 7
> numberOfEggs = as.integer(readline(prompt="Enter Number of Eggs in each basket : "))
Enter Number of Eggs in each basket : 8
> costOfEgg = as.integer(readline(prompt="Enter cost of each Egg : "))
Enter cost of each Egg : 4
> totalCostOfEggs = numberOfBaskets * numberOfEggs * costOfEgg
> print(paste("Total Cost of Eggs = ", totalCostOfEggs))
[1] "Total Cost of Eggs = 224"
> |
```

Q3. Defining and initializing a vector and calculating Mean, Variance, Standard deviation.

```
> myVector = c(5.04, -1.1, 3, 8.22, -0.7, 9)
```

```
> vectorMean = mean(myVector)
```

```
> print(paste("Mean is = ", vectorMean))
```

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/ ↗
> # Q3. Defining and initializing a vector and calculate Mean, Variance, Standard deviation.
> myVector = c(5.04, -1.1, 3, 8.22, -0.7, 9)
> vectorMean = mean(myVector)
> print(paste("Mean is = ", vectorMean))
[1] "Mean is = 3.91"
> |
```

```
> vectorVariance = (sum((myVector - vectorMean)^2))/length(myVector)
```

```
> print(paste("Variance is = ", vectorVariance))
```

Console	Terminal x	Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/ ↗		
<pre>> vectorVariance = (sum((myVector - vectorMean)^2))/length(myVector) > print(paste("Variance is = ", vectorVariance)) [1] "Variance is = 15.4902333333333" > </pre>		

> vectorStandardDeviation = sqrt(vectorVariance)

> print(paste("Standard deviation is = ", vectorStandardDeviation))

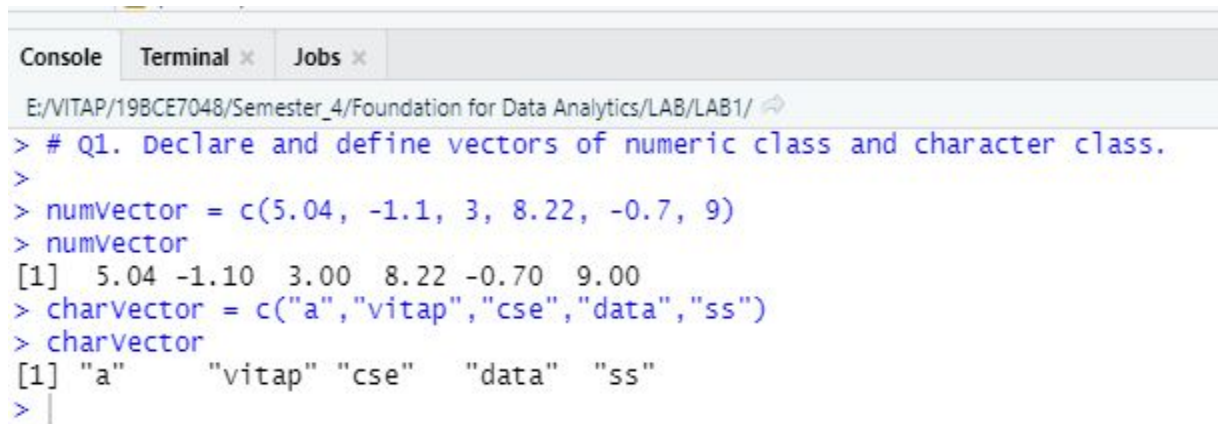
Console	Terminal x	Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/ ↗		
<pre>> vectorStandardDeviation = sqrt(vectorVariance) > print(paste("Standard deviation is = ", vectorStandardDeviation)) [1] "Standard deviation is = 3.935763373646" > </pre>		

Assignment 2:

Date: 09/02/2021

Q1. Declare and define vectors of numeric class and character class.

```
> numVector = c(5.04, -1.1, 3, 8.22, -0.7, 9)
> numVector
> charVector = c("a","vitap","cse","data","ss")
> charVector
```

A screenshot of an R console window. The window has tabs for 'Console', 'Terminal', and 'Jobs'. The 'Console' tab is active. The path 'E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/' is shown at the top. The console contains the following R code and its output:

```
> # Q1. Declare and define vectors of numeric class and character class.
>
> numVector = c(5.04, -1.1, 3, 8.22, -0.7, 9)
> numVector
[1] 5.04 -1.10 3.00 8.22 -0.70 9.00
> charVector = c("a","vitap","cse","data","ss")
> charVector
[1] "a"      "vitap"  "cse"    "data"   "ss"
> |
```

Q2. Declare two vectors of complex class of same length. Find the sum of the two vectors.

```
> vt1 = c(5+1.2*i, -3-0.1*i, 0-7*i, -6*i)
> vt2 = c(0.3*i, 2.2+5*i, -8.2-0.77*i, 1+i)
> print("Sum of the two vectors = ")
> vt1+vt2
```

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB1/
> # Q2. Declare two vectors of complex class of same length. Find the sum of the two vectors.
>
> vt1 = c(5+1.2*i, -3-0.1*i, 0-7*i, -6*i)
> vt2 = c(0.3*i, 2.2+5*i, -8.2-0.77*i, 1+i)
> print("Sum of the two vectors = ")
[1] "Sum of the two vectors = "
> vt1+vt2
[1] 5.0+1.50i -0.8+4.90i -8.2-7.77i 1.0-5.00i
> |
```

Q3. Declare two vectors of numeric class of same length. Find the sum of the two vectors and assign to another vector. Find the average of the resultant vector.

```
> vect1 = c(5.04, -1.1, 3, -0.7, 9)
> vect2 = c(-1, -0.33, 2.26, 7, -3.1)
> vect3 = vect1+vect2
> print("Sum of the two vectors = ")
> vect3
> avgvect3 = sum(vect3)/length(vect3)
> print(paste("Average of resultant vector = ", avgvect3))
```

```

>
> vect1 = c(5.04, -1.1, 3, -0.7, 9)
> vect2 = c(-1, -0.33, 2.26, 7, -3.1)
>
> vect3 = vect1+vect2
> print("Sum of the two vectors = ")
[1] "Sum of the two vectors = "
> vect3
[1] 4.04 -1.43 5.26 6.30 5.90
>
> avgvect3 = sum(vect3)/length(vect3)
> print(paste("Average of resultant vector = ", avgvect3))
[1] "Average of resultant vector = 4.014"
>

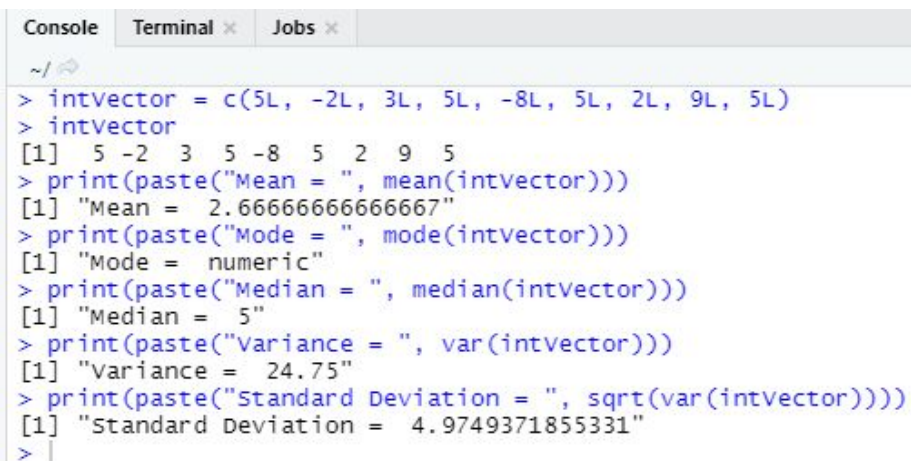
```

Q4. Declare and define vectors of integer and numeric class. Find the mean, median, mod, variance and standard deviation for both the classes separately.

```

> intVector = c(5L, -2L, 3L, 5L, -8L, 5L, 2L, 9L, 5L)
> intVector
> print(paste("Mean = ", mean(intVector)))
> print(paste("Mode = ", mode(intVector)))
> print(paste("Median = ", median(intVector)))
> print(paste("Variance = ", var(intVector)))
> print(paste("Standard Deviation = ", sqrt(var(intVector))))

```



```

Console Terminal x Jobs x
~/
> intvector = c(5L, -2L, 3L, 5L, -8L, 5L, 2L, 9L, 5L)
> intvector
[1] 5 -2 3 5 -8 5 2 9 5
> print(paste("Mean = ", mean(intvector)))
[1] "Mean = 2.66666666666667"
> print(paste("Mode = ", mode(intvector)))
[1] "Mode = numeric"
> print(paste("Median = ", median(intvector)))
[1] "Median = 5"
> print(paste("Variance = ", var(intvector)))
[1] "variance = 24.75"
> print(paste("Standard Deviation = ", sqrt(var(intvector))))
[1] "Standard Deviation = 4.9749371855331"
>

```



```
> numVector = c(3, 5.04, -1.1, 3, 8.22, 3, -0.7, 9, 3)
> numVector
> print(paste("Mean = ", mean(numVector)))
> print(paste("Mode = ", mode(numVector)))
> print(paste("Median = ", median(numVector)))
> print(paste("Variance = ", var(numVector)))
> print(paste("Standard Deviation = ", sqrt(var(numVector))))
```

```
~/
> numVector = c(3, 5.04, -1.1, 3, 8.22, 3, -0.7, 9, 3)
> numVector
[1] 3.00 5.04 -1.10 3.00 8.22 3.00 -0.70 9.00 3.00
> print(paste("Mean = ", mean(numVector)))
[1] "Mean = 3.60666666666667"
> print(paste("Mode = ", mode(numVector)))
[1] "Mode = numeric"
> print(paste("Median = ", median(numVector)))
[1] "Median = 3"
> print(paste("Variance = ", var(numVector)))
[1] "Variance = 11.8247"
> print(paste("Standard Deviation = ", sqrt(var(numVector))))
[1] "Standard Deviation = 3.4387061520287"
> |
```


Assignment 3:

Date: 16/02/2021

Q1. Declare and define list and find the mode of it.

```
Console Terminal x Jobs x
~/
> mylist = list(1,5,-4,3.5,0)
> mylist
[[1]]
[1] 1

[[2]]
[1] 5

[[3]]
[1] -4

[[4]]
[1] 3.5

[[5]]
[1] 0

> mode(mylist)
[1] "list"
> |
```

Q2. Declare two vectors and convert it to a matrix.

```
Console Terminal x Jobs x
~/
> mylist = c(7:22)
> roww = c("row 1", "row 2", "row 3", "row 4")
> col = c("column 1", "column 2", "column 3", "column 4")
> m = matrix(mylist, nrow = 4, byrow = TRUE, dimnames = list(roww, col))
> print(m)
      column 1 column 2 column 3 column 4
row 1       7       8       9      10
row 2      11      12      13      14
row 3      15      16      17      18
row 4      19      20      21      22
> |
```

Q3. Declare two matrices. Find the sum and multiplication of the two matrices and store it in the 3rd matrix.

```
Console Terminal x Jobs x
~/
> matrix1 <- matrix(7:22, nrow=4)
> matrix1
      [,1] [,2] [,3] [,4]
[1,]    7   11   15   19
[2,]    8   12   16   20
[3,]    9   13   17   21
[4,]   10   14   18   22
> matrix2 <- matrix(3:18, nrow=4)
> matrix2
      [,1] [,2] [,3] [,4]
[1,]    3    7   11   15
[2,]    4    8   12   16
[3,]    5    9   13   17
[4,]    6   10   14   18
>
> matrix3 = matrix1 %*% matrix2 # %*% matrix multiplication operator
> print(matrix3)
      [,1] [,2] [,3] [,4]
[1,]  254  462  670  878
[2,]  272  496  720  944
[3,]  290  530  770 1010
[4,]  308  564  820 1076
> |
```

Q4. Declare and define the matrix of integer class. Find the mean, median, mod, variance and standard deviation with respect to row and col.

```
>
> mylist = c(5L, -2L, 5L, 0L)
> roww = c("row1", "row2")
> col = c("column1", "column2")
> intMatrix = matrix(mylist, nrow=2, byrow=TRUE, dimnames = list(roww, col))
> intMatrix
      column1 column2
row1         5      -2
row2         5       0
>
```

Mean :

```
>
> print(paste("Mean of 1st row = ", mean(intMatrix[1,])))
[1] "Mean of 1st row = 1.5"
> print(paste("Mean of 2nd row = ", mean(intMatrix[2,])))
[1] "Mean of 2nd row = 2.5"
>
> print(paste("Mean of 1st column = ", mean(intMatrix[,1])))
[1] "Mean of 1st column = 5"
> print(paste("Mean of 2nd column = ", mean(intMatrix[,2])))
[1] "Mean of 2nd column = -1"
>
```

Mode :

```
>
> print(paste("Mode of 1st row = ", class(intMatrix[1,])))
[1] "Mode of 1st row = integer"
> print(paste("Mode of 2nd row = ", class(intMatrix[2,])))
[1] "Mode of 2nd row = integer"
>
> print(paste("Mode of 1st column = ", class(intMatrix[,1])))
[1] "Mode of 1st column = integer"
> print(paste("Mode of 2nd column = ", class(intMatrix[,2])))
[1] "Mode of 2nd column = integer"
>
```

Median :

```
>
> print(paste("Median of 1st row = ", median(intMatrix[1,])))
[1] "Median of 1st row = 1.5"
> print(paste("Median of 2nd row = ", median(intMatrix[2,])))
[1] "Median of 2nd row = 2.5"
>
> print(paste("Median of 1st column = ", median(intMatrix[,1])))
[1] "Median of 1st column = 5"
> print(paste("Median of 2nd column = ", median(intMatrix[,2])))
[1] "Median of 2nd column = -1"
>
```

Variance :

```
>
> print(paste("Variance of 1st row = ", var(intMatrix[1,])))
[1] "Variance of 1st row = 24.5"
> print(paste("Variance of 2nd row = ", var(intMatrix[2,])))
[1] "Variance of 2nd row = 12.5"
>
> print(paste("Variance of 1st column = ", var(intMatrix[,1])))
[1] "Variance of 1st column = 0"
> print(paste("Variance of 2nd column = ", var(intMatrix[,2])))
[1] "Variance of 2nd column = 2"
>
```

Standard deviation :

```
>
> print(paste("Standard Deviation of 1st row = ", sd(intMatrix[1,])))
[1] "Standard Deviation of 1st row = 4.94974746830583"
> print(paste("Standard Deviation of 2nd row = ", sd(intMatrix[2,])))
[1] "Standard Deviation of 2nd row = 3.53553390593274"
>
> print(paste("Standard Deviation of 1st column = ", sd(intMatrix[,1])))
[1] "Standard Deviation of 1st column = 0"
> print(paste("Standard Deviation of 2nd column = ", sd(intMatrix[,2])))
[1] "Standard Deviation of 2nd column = 1.4142135623731"
> |
```

Assignment 4:

Date: 23/02/2021

Q1. Declare a data frame named as Stu_Result of 15 students with attributes Reg.No, Name, M1, M2, M3, M4 and M5.

```
Console Terminal Jobs
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB3/

> # Q1. Declare a data frame named as Stu_Result of 15 students with attributes Reg.No, Name, M1, M2, M3, M4 and M5.
>
> RegNo = c(101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115)
> Name = c("Ramesh", "Sakshi", "Krishna", "Siri", "Alex", "Ramya", "Kiran", "Karthik", "Suresh", "Neha", "Rekha", "Mohan", "Prasad", "Radha", "Charan")
> M1 = c(87, 75, 83, 68, 62, 93, 85, 97, 78, 91, 93, 82, 100, 56, 51)
> M2 = c(53, 80, 41, 97, 66, 70, 61, 91, 82, 39, 55, 93, 98, 74, 77)
> M3 = c(66, 38, 44, 60, 50, 72, 78, 74, 97, 47, 33, 30, 96, 83, 53)
> M4 = c(75, 41, 97, 68, 55, 32, 30, 60, 80, 91, 62, 57, 94, 78, 56)
> M5 = c(39, 85, 53, 88, 51, 67, 94, 33, 37, 68, 72, 76, 100, 40, 88)
> Stu_Result = data.frame(RegNo, Name, M1, M2, M3, M4, M5)
> names(Stu_Result) = c('RegNo', 'Name', 'M1', 'M2', 'M3', 'M4', 'M5')
> Stu_Result
  RegNo   Name  M1 M2 M3 M4 M5
1   101 Ramesh  87 53 66 75 39
2   102  Sakshi  75 80 38 41 85
3   103  Krishna  83 41 44 97 53
4   104    Siri  68 97 60 68 88
5   105   Alex  62 66 50 55 51
6   106  Ramya  93 70 72 32 67
7   107   Kiran  85 61 78 30 94
8   108 Karthik  97 91 74 60 33
9   109  Suresh  78 82 97 80 37
10  110   Neha  91 39 47 91 68
11  111  Rekha  93 55 33 62 72
12  112  Mohan  82 93 30 57 76
13  113  Prasad 100 98 96 94 100
14  114  Radha  56 74 83 78 40
15  115  Charan  51 77 53 56 88
```

(a) Calculate the Average of Marks in another vector and append it to the Stu_Result.

```
> AverageMark = c(mean(Stu_Result$M1), mean(Stu_Result$M2), mean(Stu_Result$M3), mean(Stu_Result$M4), mean(Stu_Result$M5))
> Stu_Result$AverageMark <- AverageMark
> Stu_Result
  RegNo   Name  M1 M2 M3 M4 M5 AverageMark
1   101 Ramesh  87 53 66 75 39      80.06667
2   102  Sakshi  75 80 38 41 85      71.80000
3   103  Krishna  83 41 44 97 53      61.40000
4   104    Siri  68 97 60 68 88      65.06667
5   105   Alex  62 66 50 55 51      66.06667
6   106  Ramya  93 70 72 32 67      80.06667
7   107   Kiran  85 61 78 30 94      71.80000
8   108 Karthik  97 91 74 60 33      61.40000
9   109  Suresh  78 82 97 80 37      65.06667
10  110   Neha  91 39 47 91 68      66.06667
11  111  Rekha  93 55 33 62 72      80.06667
12  112  Mohan  82 93 30 57 76      71.80000
13  113  Prasad 100 98 96 94 100      61.40000
14  114  Radha  56 74 83 78 40      65.06667
15  115  Charan  51 77 53 56 88      66.06667
```


(b) Then append another column as Grade with respect to the table given below. The resultant data frame name should be Updated_Stu_Result

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB5/ ↗
> j=1
> Grades=0
> for (i in Stu_Result$AverageScore){
+   if(i>90){
+     Grades[j]='S'
+   }
+   else if(i>=81 && i<=90){
+     Grades[j]='A'
+   }
+   else if(i>=71 && i<=80){
+     Grades[j]='B'
+   }
+   else if(i>=61 && i<=70){
+     Grades[j]='C'
+   }
+   else if(i>=51 && i<=60){
+     Grades[j]='P'
+   }
+   else{
+     Grades[j]='F'
+   }
+   j=j+1
+ }
> Updated_Stu_Result=Stu_Result
> Updated_Stu_Result$Grades=Grades
> Updated_Stu_Result
```

	RegNo	Name	M1	M2	M3	M4	M5	AverageScore	Grades
1	101	Ramesh	87	53	66	75	39	64.0	C
2	102	Sakshi	75	80	38	41	85	63.8	C
3	103	Krishna	83	41	44	97	53	63.6	C
4	104	Siri	68	97	60	68	88	76.2	B
5	105	Alex	62	66	50	55	51	56.8	P
6	106	Ramya	93	70	72	32	67	66.8	C
7	107	Kiran	85	61	78	30	94	69.6	C
8	108	Karthik	97	91	74	60	33	71.0	B
9	109	Suresh	78	82	97	80	37	74.8	B
10	110	Neha	91	39	47	91	68	67.2	C
11	111	Rekha	93	55	33	62	72	63.0	C
12	112	Mohan	82	93	30	57	76	67.6	C
13	113	Prasad	100	98	96	94	100	97.6	S
14	114	Radha	56	74	83	78	40	66.2	C
15	115	Charan	51	77	53	56	88	65.0	C

```
> |
```

(c) Create another data frame named as New_Stu_Result with the attributes Reg.No, Name, M1, M2, M3, M4, M5, Average, and Grade.

```
> New_Stu_Result = data.frame(RegNo, Name, M1, M2, M3, M4, M5, AverageScore, Grades)
> names(New_Stu_Result) = c('RegNo', 'Name', 'M1', 'M2', 'M3', 'M4', 'M5', 'Average', 'Grade')
> New_Stu_Result
```

	RegNo	Name	M1	M2	M3	M4	M5	Average	Grade
1	101	Ramesh	87	53	66	75	39	64.0	C
2	102	Sakshi	75	80	38	41	85	63.8	C
3	103	Krishna	83	41	44	97	53	63.6	C
4	104	Siri	68	97	60	68	88	76.2	B
5	105	Alex	62	66	50	55	51	56.8	P
6	106	Ramya	93	70	72	32	67	66.8	C
7	107	Kiran	85	61	78	30	94	69.6	C
8	108	Karthik	97	91	74	60	33	71.0	B
9	109	Suresh	78	82	97	80	37	74.8	B
10	110	Neha	91	39	47	91	68	67.2	C
11	111	Rekha	93	55	33	62	72	63.0	C
12	112	Mohan	82	93	30	57	76	67.6	C
13	113	Prasad	100	98	96	94	100	97.6	S
14	114	Radha	56	74	83	78	40	66.2	C
15	115	Charan	51	77	53	56	88	65.0	C

Q2. Create a data frame as given below

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.4	3.4	1.5	0.4	setosa
5.2	4.1	1.5	0.1	setosa
5.8	2.6	4	1.2	versicolor
5.1	3.5	1.4	0.3	setosa
6.3	2.5	4.9	1.5	versicolor
5.7	3.8	1.7	0.3	setosa
6.7	3.1	5.6	2.4	virginica
5.4	3.4	1.7	0.2	setosa
6.7	3.1	4.4	1.4	versicolor
5.5	3.5	1.3	0.2	setosa

Output :

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB3/ <
> Sepal.Length = c(5.4, 5.2, 5.8, 5.1, 6.3, 5.7, 6.7, 5.4, 6.7, 5.5)
> Sepal.Width = c(3.4, 4.1, 2.6, 3.5, 2.5, 3.8, 3.1, 3.4, 3.1, 3.5)
> Petal.Length = c(1.5, 1.5, 4, 1.4, 4.9, 1.7, 5.6, 1.7, 4.4, 1.3)
> Petal.Width = c(0.4, 0.1, 1.2, 0.3, 1.5, 0.3, 2.4, 0.2, 1.4, 0.2)
> Species = c('setosa', 'setosa', 'versicolor', 'setosa', 'versicolor', 'setosa', 'virginica', 'setosa', 'versicolor', 'setosa')
>
> irisData = data.frame(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, Species)
> names(irisData) = c('Sepal.Length', 'Sepal.Width', 'Petal.Length', 'Petal.Width', 'Species')
> irisData
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.4          3.4          1.5          0.4   setosa
2          5.2          4.1          1.5          0.1   setosa
3          5.8          2.6          4.0          1.2 versicolor
4          5.1          3.5          1.4          0.3   setosa
5          6.3          2.5          4.9          1.5 versicolor
6          5.7          3.8          1.7          0.3   setosa
7          6.7          3.1          5.6          2.4  virginica
8          5.4          3.4          1.7          0.2   setosa
9          6.7          3.1          4.4          1.4 versicolor
10         5.5          3.5          1.3          0.2   setosa
```

(a) What are the mean and median of the column “Sepal.Length”?

```
> print(paste("Mean of Sepal.Length = ", mean(irisData$Sepal.Length)))
[1] "Mean of Sepal.Length = 5.78"
> print(paste("Median of Sepal.Length = ", median(irisData$Sepal.Length)))
[1] "Median of Sepal.Length = 5.6"
> |
```

(b) What is the mode of the column “Species”?

```
> # (b) What is the mode of the column “Species”?
> print(paste("Mode of Species = ", mode(irisData$Species)))
[1] "Mode of Species = character"
> |
```

(c) What are standard deviation and variance of the column “Petal.Width”?

```
> # (c) What are standard deviation and variance of the column “Petal.Width”?
> print(paste("Variance of Petal.Width = ", var(irisData$Petal.Width)))
[1] "Variance of Petal.Width = 0.604444444444444"
> print(paste("Standard Deviation of Petal.Width = ", sd(irisData$Petal.Width)))
[1] "Standard Deviation of Petal.Width = 0.77746025264604"
> |
```

- (d) What is the normalized value of 2.5 in the column "Sepal.Width" using min-max normalization having new minimum value as 11 and new maximum value is 13.

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗
> # x' = (x - min(x)) / (max(x) - min(x))
> print(paste("normalized value = ", (Sepal.Width[5] - min(11)) / (max(13) - min(11))))
[1] "normalized value = -4.25"
> |
```

- (e) What is the normalized value of 2.5 in the column "Petal.length" using z-score normalization?

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗
> irisData$Petal.Length
[1] 1.5 1.5 4.0 1.4 4.9 1.7 5.6 1.7 4.4 1.3
> print("There is no 2.5 value in the column Petal.Length. Hence we cannot normalize the data.")
[1] "There is no 2.5 value in the column Petal.Length. Hence we cannot normalize the data."
> |
```

Assignment 5:

Date: 02/03/2021

Q1. Write a R program to enter any number and calculate its factorial.

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/
> myFactorial = function(n){
+   fact = 1
+   if(n < 0) {
+     print("Factorial does not exist for Negative numbers")
+   } else if(n == 0) {
+     print("The factorial of 0 is 1")
+   } else {
+     for(i in 1:n) {
+       fact = fact * i
+     }
+     print(paste("The factorial of", n ,"is = ",fact))
+   }
+ }
> n = as.integer(readline(prompt="Enter a number: "))
Enter a number: 7
> myFactorial(n)
[1] "The factorial of 7 is = 5040"
> |
```

Q2. Write a R program to print all Armstrong numbers between 1 to n.

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB5/
> myAmstrong = function(n){
+   sum = 0
+   flag=1
+   while(flag<=n)
+   {
+     temp = flag
+     sum=0
+     while(temp > 0)
+     {
+       digit = temp %% 10
+       sum = sum + (digit^3)
+       temp = floor(temp / 10)
+     }
+     if(flag == sum) {
+       print(flag)
+     }
+     flag=flag+1;
+   }
+ }
> n = as.integer(readline(prompt="Enter a number: "))
Enter a number: 1000
> print(paste("Armstrong numbers between 1 to", n,"are :"))
[1] "Armstrong numbers between 1 to 1000 are :"
> myAmstrong(n)
[1] 1
[1] 153
[1] 370
[1] 371
[1] 407
> |
```

Q3. Create a data frame givenData

```
Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗
> EmployeeID = seq(1:20)
> ManagerID = c(0, 1, 2, 3, 3, 3, 3, 4, 4, 4, 3, 4, 4, 3, 3, 1, 2, 2, 2, 2)
>
> Title = c("Chief Executive Officer", "Vice President of Engineering", "Engineering Manager",
+ "Senior Tool Designer", "Design Engineer", "Design Engineer", "Research and Development Manager",
+ "Research and Development Engineer", "Research and Development Engineer", "Research and Development Manager",
+ "Senior Tool Designer", "Tool Designer", "Tool Designer", "Senior Design Engineer", "Design Engineer",
+ "Marketing Manager", "Marketing Assistant", "Marketing Specialist", "Marketing Assistant", "Marketing Assistant")
>
> BirthDate = as.Date(c("1/29/1969", "8/1/1971", "11/12/1974", "12/23/1974", "9/27/1952", "3/11/1959", "2/24/1987",
+ "6/5/1986", "1/21/1979", "11/30/1984", "1/17/1978", "7/29/1959", "5/28/1989", "6/16/1979",
+ "5/2/1961", "3/19/1975", "5/3/1987", "3/6/1978", "1/29/1978", "3/17/1975"), format = "%m/%d/%Y")
>
> MaritalStatus = c('S', 'S', 'M', 'S', 'M', 'M', 'M', 'S', 'M', 'M', 'S', 'M', 'M', 'S', 'M', 'S', 'S', 'S', 'S', 'M')
> Gender = c('M', 'F', 'M', 'M', 'F', 'M', 'M', 'F', 'F', 'M', 'M', 'M', 'F', 'M', 'F', 'M', 'M', 'M', 'F', 'F')
>
> givenData = data.frame(EmployeeID, ManagerID, Title, BirthDate, MaritalStatus, Gender)
> names(givenData) = c('EmployeeID', 'ManagerID', 'Title', 'BirthDate', 'MaritalStatus', 'Gender')
> givenData
  EmployeeID ManagerID Title BirthDate MaritalStatus Gender
1          1         0 Chief Executive Officer 1969-01-29      S      M
2          2         1 Vice President of Engineering 1971-08-01      S      F
3          3         2 Engineering Manager 1974-11-12      M      M
4          4         3 Senior Tool Designer 1974-12-23      S      M
5          5         3 Design Engineer 1952-09-27      M      F
6          6         3 Design Engineer 1959-03-11      M      M
7          7         3 Research and Development Manager 1987-02-24      M      M
8          8         4 Research and Development Engineer 1986-06-05      S      F
9          9         4 Research and Development Engineer 1979-01-21      M      F
10         10         4 Research and Development Manager 1984-11-30      M      M
11         11         3 Senior Tool Designer 1978-01-17      S      M
12         12         4 Tool Designer 1959-07-29      M      M
13         13         4 Tool Designer 1989-05-28      M      F
14         14         3 Senior Design Engineer 1979-06-16      S      M
15         15         3 Design Engineer 1961-05-02      M      F
16         16         1 Marketing Manager 1975-03-19      S      M
17         17         2 Marketing Assistant 1987-05-03      S      M
18         18         2 Marketing Specialist 1978-03-06      S      M
19         19         2 Marketing Assistant 1978-01-29      S      F
20         20         2 Marketing Assistant 1975-03-17      M      F
> |
```


i) Append another column as Age by calculating the age from date of birth.

```

Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗
> Age = floor(age_calc(givenData$BirthDate, enddate = Sys.Date(), units = "years"))
> givenData$Age = Age
> givenData
  EmployeeID ManagerID Title BirthDate MaritalStatus Gender Age
1          1         0 Chief Executive Officer 1969-01-29      S      M  52
2          2         1 Vice President of Engineering 1971-08-01      S      F  49
3          3         2 Engineering Manager 1974-11-12      M      M  46
4          4         3 Senior Tool Designer 1974-12-23      S      M  46
5          5         3 Design Engineer 1952-09-27      M      F  68
6          6         3 Design Engineer 1959-03-11      M      M  61
7          7         3 Research and Development Manager 1987-02-24      M      M  34
8          8         4 Research and Development Engineer 1986-06-05      S      F  34
9          9         4 Research and Development Engineer 1979-01-21      M      F  42
10         10         4 Research and Development Manager 1984-11-30      M      M  36
11         11         3 Senior Tool Designer 1978-01-17      S      M  43
12         12         4 Tool Designer 1959-07-29      M      M  61
13         13         4 Tool Designer 1989-05-28      M      F  31
14         14         3 Senior Design Engineer 1979-06-16      S      M  41
15         15         3 Design Engineer 1961-05-02      M      F  59
16         16         1 Marketing Manager 1975-03-19      S      M  45
17         17         2 Marketing Assistant 1987-05-03      S      M  33
18         18         2 Marketing Specialist 1978-03-06      S      M  42
19         19         2 Marketing Assistant 1978-01-29      S      F  43
20         20         2 Marketing Assistant 1975-03-17      M      F  45
> |

```

ii) Calculate the mean, median, variance and SD of age

```

Console Terminal x Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗
> print(paste("Mean = ", mean(givenData$Age)))
[1] "Mean = 45.55"
>
> print(paste("Median = ", median(givenData$Age)))
[1] "Median = 44"
>
> print(paste("Variance = ", var(givenData$Age)))
[1] "Variance = 105.418421052632"
>
> print(paste("Standard Deviation = ", sd(givenData$Age)))
[1] "Standard Deviation = 10.2673473230738"
> |

```

iii) Display the details of Gender=Female(F)

Console

Terminal x

Jobs x

E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗

```
> print(givenData[givenData$Gender == 'F',])
```

EmployeeID	ManagerID	Title	BirthDate	MaritalStatus	Gender	Age
2	2	1 Vice President of Engineering	1971-08-01	S	F	49
5	5	3 Design Engineer	1952-09-27	M	F	68
8	8	4 Research and Development Engineer	1986-06-05	S	F	34
9	9	4 Research and Development Engineer	1979-01-21	M	F	42
13	13	4 Tool Designer	1989-05-28	M	F	31
15	15	3 Design Engineer	1961-05-02	M	F	59
19	19	2 Marketing Assistant	1978-01-29	S	F	43
20	20	2 Marketing Assistant	1975-03-17	M	F	45

```
> |
```

iv) Display the details of Gender= Male and who are married(M)

E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗

```
> print(givenData[givenData$Gender == 'M' & givenData$MaritalStatus == 'M',])
```

	EmployeeID	ManagerID	Title	BirthDate	MaritalStatus	Gender	Age
3	3	2	Engineering Manager	1974-11-12		M	M 46
6	6	3	Design Engineer	1959-03-11		M	M 61
7	7	3	Research and Development Manager	1987-02-24		M	M 34
10	10	4	Research and Development Manager	1984-11-30		M	M 36
12	12	4	Tool Designer	1959-07-29		M	M 61

```
> |
```

v) Count no of person having Title= Market Assistant and Marital status as Single(S)

Console	Terminal x	Jobs x
E:/VITAP/19BCE7048/Semester_4/Foundation for Data Analytics/LAB/LAB4/ ↗		
> nrow(givenData[givenData\$Title == "Marketing Assistant" & givenData\$MaritalStatus == 'S',])		
[1] 2		
>		