

Q. Compare R and Python?

Python	R Programming
Python is a general-purpose language that is used for the deployment and development of various projects. Python has all the tools required to bring a project into the production environment.	R is a statistical language used for the analysis and visual representation of data.
Python is better suitable for machine learning, deep learning, and large-scale web applications.	R is suitable for statistical learning having powerful libraries for data experiment and exploration.
Python has a lot of libraries. However, it can be complex to understand all of them.	R has fewer libraries compared to Python and is easy to know.
Python can be used for various purposes like building a graphical user interface, develop games, etc., despite being an object-oriented language.	Along with object-oriented programming, R can also be used to develop music.
Python has a simple syntax and is easy to learn.	R has a relatively complex syntax and the learning curve is not straightforward.
Python's statistical packages are less powerful.	R's statistical packages are highly powerful.
Python is mainly used when the data analysis needs to be integrated with web applications.	R is generally used when the data analysis task requires standalone computation(analysis) and processing.
Python can be used to build applications from scratch.	R can be used to simplify complex mathematical problems.
There are many python IDE's available to choose from, a few of them are Jupyter Notebook, Spyder, Pycharm, etc.	A few of IDE's for the R language are RStudio, StatET, etc.
Python is more popular and has a vast user base. Primary users of python include developers and programmers.	R is less popular among users. Its users include scientists and Research & Development who frequently rely on data analysis.

Q. What is the main difference between an Array and a matrix and a data frame?

Arrays	Matrices
Arrays can contain greater than or equal to 1 dimensions.	Matrices contain 2 dimensions in a table like structure.
Array is a homogeneous data structure.	Matrix is also a homogeneous data structure.
It is a singular vector arranged into the specified dimensions.	It comprises of multiple equal-length vectors stacked together in a table.
array() function can be used to create matrix by specifying the third dimension to be 1.	matrix() function however can be used to create at most 2-dimensional array.
Arrays are superset of matrices.	Matrices are a subset, special case of array where dimensions is two.
Limited set of collection-based operations.	Wide range of collection operations possible.
Mostly, intended for storage of data.	Mostly, matrices are intended for data transformation.

Define R.

R is a system for statistical computation and graphics. It provides, among other things, a programming language, high level graphics, interfaces to other languages and debugging facilities. This manual details and defines the R language.

Are variable 'H' and 'h' same in R language

Case sensitivity. Technically R is a function language with a very simple syntax. It is case sensitive, so A and a are different variables.

What is the difference between cbind() and rbind() functions?

cbind() and rbind() both create matrices by combining several vectors of the same length. cbind() combines vectors as columns, while rbind() combines them as rows.

What are the packages in r language give some example of visualisation package

R packages are extensions to the R statistical programming language. R packages contain code, data, and documentation in a standardised collection

format that can be installed by users of R, typically via a centralised software repository such as CRAN (the Comprehensive R Archive Network).

- ggplot2. ggplot2 is based on the 'Grammar of Graphics', which is a popular data visualization library. ...
- data. table. ...
- dplyr. ...
- tidyr. ...
- Shiny. ...
- plotly. ...
- knitr. ...
- mlr3.

How many types of loop available in R language?

There are three types of loop in R programming:

- For Loop
- While Loop
- Repeat Loop

List the differences between vector and list.

Vector	List
It has contiguous memory.	While it has non-contiguous memory.
It is synchronized.	While it is not synchronized.
Vector may have a default size.	List does not have default size.
In vector, each element only requires the space for itself only.	In list, each element requires extra space for the node which holds the element, including pointers to the next and previous elements in the list.
Insertion at the end requires constant time but insertion elsewhere is costly.	Insertion is cheap no matter where in the list it occurs.

Vector	List
Vector is thread safe.	List is not thread safe.
Deletion at the end of the vector needs constant time but for the rest it is $O(n)$.	Deletion is cheap no matter where in the list it occurs.
Random access of elements is possible.	Random access of elements is not possible.
Iterators become invalid if elements are added to or removed from the vector.	Iterators are valid if elements are added to or removed from the list.

What is the use of legend() function

legend **creates a legend with descriptive labels for each plotted data series.**

For the labels, the legend uses the text from the DisplayName properties of the data series.

Write applications of R programming

- In Research and academic
- In IT sector
- In finance
- In e-commerce

Write syntax of repeat in R

The basic syntax of the repeat loop is as follows:

1. repeat {
2. commands.
3. if(condition) {
4. break.
5. }
6. }

What do you meant by API key in google map

The API key is a unique identifier that authenticates requests associated with your project for usage and billing purposes. You must have at least one API key associated with your project.

Google Maps API is a set of application programming interfaces that lets us talk to its services. It will allow us to build simple apps to very sophisticated location-based apps for Web, iOS, and Android.

Which function in R language is used to find out whether the means of 2 groups are equal to each other or not?

Explanation: **t. tests ()** function in R language is used to find out whether the means of 2 groups are equal to each other.

What are the Homogeneous and Heterogeneous data structures in R.?

Homogeneous data structures are ones that can only store a single type of data (numeric, integer, character, etc.).

Heterogeneous data structures are ones that can store more than one type of data at the same time.

How do you create log-linear models in R language?

loglm() is a fit log-linear model by iterative proportional scaling. This function provides a front-end of the standard function, **loglin**, to allow log-linear models to be specified and fitted in a manner similar to that of other fitting functions.

What is the purpose of using the Next statement in R language?

The next statement in the R programming language is useful when we want to skip the current iteration of a loop without terminating it. On encountering next, the R parser skips further evaluation and starts the next iteration of the loop.

What are the rules to define a variable name in R programming language?

Rules for R variables are:

- A variable name must start with a letter and can be a combination of letters, digits, period(.), and underscore(_). If it starts with a period(.), it cannot be followed by a digit.
- A variable name cannot start with a number or underscore (_)
- Variable names are case-sensitive (age,, Age and AGE are three different variables)
- Reserved words cannot be used as variables (TRUE, FALSE, NULL, if...)

What will be the output of the following R programming code?

```
var2<- c("I","Love","DeZyre")
```

```
var2
```

Error: unexpected symbol in "var2<- c("I","Love","DeZyre")"

What will be the result of multiplying two vectors in R having different lengths?

When we try to multiply vectors of different lengths in R Studio, it gives a warning message as the longer object length is not a multiple of the shorter object length.

What is the syntax of normal distribution in R.?

rnorm() This function is used to generate random numbers whose distribution is normal. It takes the sample size as input and generates that many random numbers. We draw a histogram to show the distribution of the generated numbers.

What do you mean by the Chi-Square test? How to use it in R also interprets the output.

The Chi-Square test in R is a statistical method used to determine if two categorical variables have a significant correlation between them. The two variables are selected from the same population. Furthermore, these variables are then categorized as Male/Female, Red/Green, Yes/No, etc.

Write the r programme to generate the table of any number.

```
num <- as.integer(readline(prompt = "Enter a number: "))
for (i in 1:10){
  print(num*i)
}
```

Write r programme to calculate the compound amount of home loan.

```
P <- as.double(readline(prompt = "Enter Principal amount:- "))
R <- as.double(readline(prompt = "Enter Annual nominal interest rate as a
percent: "))
r <- R/100
n <- as.double(readline(prompt = "Enter number of compounding periods per
unit of time:- "))
t <- as.double(readline(prompt = "Enter time in decimal year"))
```

```
A<- P*(1+r/n)^n*t
print(A)
```

GST has different rates according to the product purchased. Imagine we have three different kinds of products with different VAT applied: We can write a chain to apply the correct VAT rate to the product a customer bought.

A	Book, magazine, newspaper, etc.	8%
B	Vegetables, meat, beverage, etc.	10%
C	Tee-shirt, jeans, pants, etc.	20%

```
category <- 'A'

price <- 10

if (category == 'A'){

  cat('A vat rate of 8% is applied.', 'The total price is', price * 1.08)

} else if (category == 'B'){

  cat('A vat rate of 10% is applied.', 'The total price is', price * 1.10)

} else {

  cat('A vat rate of 20% is applied.', 'The total price is', price * 1.20)

}
```

Cyclone Fani was the longest-lived cyclone in the Bay of Bengal ever observed. The elongated time period of the storm went on for 11 days in the sea and land put together. What made it even more surprising? We had started observing it as soon as it formed in the equatorial. The parameter that we observed in the Indian Ocean low-pressure area, direction, and intensity of winds close to the ocean surface, wind speed, Area, humidity, and reduction in speed, shows the center of the storm, from where it is started, and finished, what type of pattern it follows in different parameter at different time, losses from the cyclone, a major area of losses, etc. Forecast for the pattern and precaution. A data analyzer wants to visualize

these parameter before the cyclone and after the cyclone and relate them to each other which type of chart graphs and diagrams are useful for visualization draw all charts with syntax in r and explain why it is useful to show parameters. You are free to take many more parameters to visualize the data related to the fani storm in more attractive way.

Next Paper

Why this is named R programming language?

The "R" name is **derived from the first letter of the names of its two developers, Ross Ihaka and Robert Gentleman**, who were associated with the University of Auckland at the time.

What are the main features of R?

Features of R Programming

- Open-source. R is an open-source software environment. ...
- Strong Graphical Capabilities. ...
- Highly Active Community. ...
- A Wide Selection of Packages. ...
- Comprehensive Environment. ...
- Can Perform Complex Statistical Calculations. ...
- Distributed Computing. ...
- Running Code Without a Compiler

What do you understand by R-objects?

R consists of a number of data objects to perform various functions. There are 6 types of objects in R Programming. They include vector, list, matrix, array, factor, and data frame.

What are R Basic Functions?

This package **contains the basic functions which let R function as a language: arithmetic, input/output, basic programming support, etc.** Its contents are available through inheritance from any environment. For a complete list of functions, use `library(help = "base")` .

Example:- `as.function`, `as.factor`, `as.name`, `as.ordered`, `as.integer`, `as.double` etc.

What is a CSV file? Which function is used to read CSV files? can you give some examples?

A **CSV file** is a simple type of plain text file that uses a specific structure to arrange tabular data. The standard format of a CSV file is defined by rows and columns data where a newline terminates each row to begin the next row, and each column is separated by a comma within the row.

read.csv() function in R Language is used to read “comma-separated value” files. It imports data in the form of a data frame.

Example:- `data <- read.csv("CSVFileExample.csv",
header = FALSE, sep = "\t")`

What are the types of control statements in R programming?

In R programming, there are 8 types of control statements as follows:

- if condition
- if-else condition
- for loop
- nested loops
- while loop
- repeat and break the statement
- return statement
- next statement

What is the use of with () and By() functions in R programming?

`with()` function enables us to evaluate an R expression within the function to be passed as an argument. It works on data frames only.

For example:- applying a t-test to a data frame mydata

`with(mydata, t-test(y~group))`

The `by()` function in R is an easy function that allows us to group data within a data set, and perform mathematical functions on it.

For example:- Obtain variable means separately

`by(mydata, mydata$byvar, function(x).mean(x))`

Return a vector containing the letters of the alphabet in reverse order

```
a <- c(LETTERS)
```

```
rev(a)
```

What's the difference between "=" and "<-" in R?

The operator <- can be used anywhere, whereas the operator = is only allowed at the top level (e.g., in the complete expression typed at the command prompt) or as one of the subexpressions in a braced list of expressions.

What is theme in ggplot?

Themes are a powerful way to customize the non-data components of your plots: i.e. titles, labels, fonts, background, gridlines, and legends. Themes can be used to give plots a consistent customized look. Modify a single plot's theme using `theme()`

Why is R programming used instead of MS excel?

1. More powerful data manipulation capabilities
2. Easier automation
3. Faster computation
4. It reads any type of data
5. Easier project organization
6. It supports larger data sets
7. Reproducibility (important for detecting errors)
8. Easier to find and fix errors
9. It's free
10. It's open source
11. Advanced Statistics capabilities
12. State-of-the-art graphics
13. It runs on many platforms
14. Anyone can contribute packages to improve its functionality

Write an R program to create a sequence of numbers from 20 to 50 and find the mean of numbers from 20 to 60 and sum of numbers from 51 to 91.

```
a <- seq(20:50)
b <- mean(20:60)
c <- sum(51:91)
print(paste("Sequence of Number from 20 to 50 is :- "))
print(a)
print(paste("Mean of number from 20 to 60 is:- ", b))
print(paste("Sum of Numbers from 51 to 91:- ", c))
```

Create the following vectors in R. a = (5, 10, 15, 20, ..., 160) b = (87, 86, 85, ..., 56) Use vector arithmetic to multiply these vectors and call the result 'd'. Select subsets of d to identify the following.

- i. What are the 19th, 20th, and 21st elements of d?**
- ii. What are all of the elements of d which are less than 2000?**
- iii. How many elements of d are greater than 6000?**

```
a <- seq(5,160, by= 5)
b <- seq(87,56, by= -1)
d <- a*b
print(d)
i.)  print(d[19])
      print(d[20])
      print(d[21])
ii.)  print(d[d<2000])
iii.) length(d[d>6000])
```

Your cell phone bill (\$) varies from month to month. Suppose your year has the following monthly amounts

46 33 39 37 46 30 48 32 49 35 30 48

Enter this data into a variable called bill. Use commands to find the amount you spent this year on the cell phone. What is the smallest amount you spent in a month? What is the largest? How many months was the amount greater than \$40? What percentage was this?

```
bill <- c(46, 33, 39, 37, 46, 30, 48, 32, 49, 35, 30, 48)
sum(bill)
print(min(bill))
print(max(bill))
(sum(bill>40)/length(bill))*100
```

Following are the runs scored by a batsman in 10 consecutive matches. Test the performance of batsman is consistent 22, 98, 13, 54, 77, 61, 45, 32, 19, 85.

```
score <- c(22,98,13,54,77,61,45,32,19,85)
print(mean(score))
print(var(score))
print(sqrt(var(score)))
print(sd(score)/mean(score) * 100)
```

Heights (in cm) of father and son are given as follows Fit an appropriate regression line predict the height of son.

Father(X)	150	152	155	157	160	161	164	165	
Son(Y)		154	156	158	159	160	162	161	164

```
father <- c(150,152,155,157,160,161,164,165)
son <- c(154,156,158,159,160,162,161,164)
relation <- lm(son~father)
print(relation)
```

Heights (in cm) of father and son are given as follows Fit an appropriate regression line predict the height of son.

Father(X)	150	152	155	157	160	161	164	165	
Son(Y)		154	156	158	159	160	162	161	164

```
father <- c(150,152,155,157,160,161,164,165)
```

```
son <- c(154,156,158,159,160,162,161,164)
```

```
relation <- lm(son~father)
```

```
print(relation)
```

a) Obtain a pattern using mapply() as follows

```
4
3 3
2 2 2
1 1 1
```

```
mapply(rep, 4:1, 1:4)
```

Create a list that contain a reptition of characters as follows

```
A A A A A A
B B B B B
C C C C
D D D
E E
F
```

```
char <- c(LETTERS[1:6])
```

```
count <- 1
```

```
for (i in 6:1){
```

```
  c <- rep(char[count], i)
```

```
  print(c)
```

```
  count <- count + 1
```

```
}
```

Discuss the following:

i. What does P-value signify about the statistical data?

- A p-value is a statistical measurement used to validate a hypothesis against observed data.
- A p-value measures the probability of obtaining the observed results, assuming that the null hypothesis is true.
- The lower the p-value, the greater the statistical significance of the observed difference.
- A p-value of 0.05 or lower is generally considered statistically significant.
- P-value can serve as an alternative to or in addition to preselected confidence levels for hypothesis testing.

What are the disadvantages of the linear model?

- Regression models cannot work properly if the input data has errors (that is poor quality data). If the data preprocessing is not performed well to remove missing values or redundant data or outliers or imbalanced data distribution, the validity of the regression model suffers.
- Regression models are susceptible to collinear problems (that is there exists a strong linear correlation between the independent variables). If the independent variables are strongly correlated, then they will eat into each other's predictive power and the regression coefficients will lose their ruggedness.
- As the number of variables increases the reliability of the regression models decreases. The regression models work better if you have a small number of variables.
- Regression models do not automatically take care of nonlinearity. The user needs to imagine the kind of additional terms that might be needed to be added to the regression model to improve its fit.
- Regression models work with datasets containing numeric values and not with categorical variables. There are ways to deal with categorical variables though by creating multiple new variables with a yes/no value.

What are the possible ways of improving the accuracy of a linear regression model?

- Handling Null/Missing Values.
- Data Visualization.
- Feature Selection and Scaling.
 - Feature Engineering.
 - Feature Transformation.

- Use of Ensemble and Boosting Algorithms.
- Hyperparameter Tuning.

Write the script to sum the exponential series.

```
myfun=function(n){
  mysum <- 0
  while(n!=0){
    mysum = mysum + exp(n)
    n=n-1
  }
  return(mysum)
}
myfun(3)
```

Write a programme to compute your Grade in R programming Subject. Marks and grade is as 35<F , 35- 39 P, 40-49-C , 50-59- B, 60-69- B+, 70-79- A, 80-89- A+ ,90- above -O

```
marks <- as.double(readline(prompt = "Enter the Marks:- "))
if (marks < 35){
  print("F")
}else if (marks > 35 & marks < 40){
  print("P")
}else if (marks > 39 & marks < 50 ){
  print("C")
}else if (marks > 49 & marks < 60){
  print("B")
}else if (marks > 59 & marks < 70){
  print("B+")
}
```

```

}else if (marks > 69 & marks < 80){
  print("A")
}else if (marks > 79 & marks < 90){
  print("A+")
}else if (marks > 89){
  print("O")
}

```

Create a sample of 50 numbers which are incremented by 1. Create the binomial distribution, plot the graph for this sample

```

# Create a sample of 50 numbers which are incremented by 1.
x <- seq(0,50,by = 1)
# Create the binomial distribution.
y <- dbinom(x,50,0.5)
# Plot the graph for this sample.
plot(x,y)

```

Plot the function $g(t) = (t^2 + 1)^{0.5}$ for t between 0 and 5

```

v <- c()
for (i in 1:5){
  v <- append(v, (i^2+1)^.5)
}
plot(v)

```

Suppose, a group of 25 people are surveyed as to their beer-drinking preference. The categories were (1) Domestic can, (2) Domestic bottle, (3) Microbrew and (4) import. The raw data is 3 4 1 1 3 4 3 3 1 3 2 1 2 1 2 3 2 3 1 1 1 1 4 3 1 Let's make Suitable plot of both frequencies and proportions

Let's make a barplot of both frequencies and proportions. First, we use the scan function to read in the data then we plot a bar chart>

```
beer = c(3, 4, 1, 1, 3, 4, 3, 3, 1, 3, 2, 1, 2, 1, 2, 3, 2, 3, 1, 1, 1, 1, 4, 3, 1)
```

```
barplot(beer) # this isn't correct> barplot(table(beer)) # Yes, call with summarized data> barplot(table(beer)/length(beer)) # divide by n for proportions.
```

We did 3 barplots.

The first to show that we don't use barplot with the raw data.

b. The second shows the use of the table command to create summarized data, and the result of this is sent to barplot creating the barplot of frequencies shown.

c. Finally, the command

```
table(beer)/length(beer)
```

With the help of mtcars data plot a heat map.

The mtcars dataset:

```
data <- as.matrix(mtcars)
```

Default Heatmap

```
heatmap(data)
```

You built a scatter plot of the diamonds_ sample dataset, with carat on the x-axis and price on the y-axis. Is used to add a smooth line, but show only the smooth line, no points. Draw multiple line with different colours for the variables.

```
library(ggplot2)
```

```
data("diamonds", package = "ggplot2")
```

```
View(diamonds)
```

```
ggplot(diamonds, aes(x=carat, y=price, color=clarity)) +  
geom_smooth(se=FALSE)
```

**School of Data Science & Forecasting
Devi Ahilya Vishwavidyalaya, Indore
Semester End Examination, M.-Tech. (Data Science) I Sem.**

DS7A/7B: 723 Statistical programming in R

Time : 3 Hours

M.M. 60

Note : Attempt Ten question From Part A . Attempt Eight Questions from Part B

Part A

1. How to install R programming language.

Ans: Steps to install R in windows:

- Download the R installer from official site of R programming with version R-4.2.0 .
- Run the installer. Default settings are fine.
- Once the installation of R has completed successfully (and not before), run the RStudio installer.
- Open RStudio. It should open a window that looks like a window with four partions. Named as R script , R console , Global Environment, visualization window.
- In the left hand window, by the '>' sign, type '4+5'(without the quotes) and hit enter. An output line reading '[1] 9' should appear. This means that R and RStudio are working.

2. What is role of package in R.

Ans: Packages are part of R programming and they are useful in collecting sets of R functions into a single unit. It also contains compiled code and sample data. All of these are kept stored in a directory called the "library" in the R environment.

3. What is a vector? How to create it?

Ans: A **vector** is a basic data structure which plays an important role in R programming.

In R, a sequence of elements which share the same data type is known as vector. A vector supports logical, integer, double, character, complex, or raw data type. The elements which are contained in vector known as **components** of the vector. we use c() function to create a vector.

Example : `vec<-c(1,2,3,4,5,6,7).`

4. Create a vector X of elements 5, 2, -1, 7, 4, 8, 12 and from it create a vector Y containing elements of $x > 4$

Ans: creating vector x:

```
X<-c(5,2,-1,7,4,8,12)
```

Creating Y:

```
X<-c(5,2,-1,7,4,8,12)
```

```
Y<-c()
```

```
for (item in X){
```

```
  if (item>4){
```

```
    Y<-c(Y,item)
```

```
  }
```

```
}
```

```
print(Y)
```

5. Write applications of R programming.

Ans: Application of R programming:

- Finance
- Healthcare
- Social Media
- E-Commerce
- Manufacturing
- R is most widely used for *exploratory data analysis*
- R also allows hypothesis testing to validate statistical models.

6. Write about Arithmetic and Boolean operators in R programming?

Ans: Arithmetic operations simulate various math operations, like addition, subtraction, multiplication, division, and modulo using the specified operator between operands, which may be either scalar values, complex numbers, or vectors.

- Subtraction Operator (-)
- Multiplication Operator (*)
- Division Operator (/)

- Power Operator (^)
- Modulo Operator (%)
- Integer division(/)

Logical operations simulate element-wise decision operations, based on the specified operator between the operands, which are then evaluated to either a True or False boolean value.

- Element-wise Logical AND operator (&)
- Element-wise Logical OR operator (|)
- NOT operator (!)
- Logical AND operator (&&)
- Logical OR operator (||)

7. What is the syntax and use of abline () function?

Ans: A program in R is made up of three things: Variables, Comments, and Keywords. Variables are used to store the data, Comments are used to improve code readability, and Keywords are reserved words that hold a specific meaning to the compiler.

- Variables in R : Used to store a value or values.
- Comments in R: Comments are a way to improve your code's readability and are only meant for the user so the interpreter ignores it.
- Keywords in R: **Keywords** are the words reserved by a program because they have a special meaning thus a keyword can't be used as a variable name, function name, etc

Abline function:

abline():

function in R Language is used to add one or more straight lines to a graph. The abline() function can be used to add vertical, horizontal or regression lines to plot.

abline(a=NULL, b=NULL, h=NULL, v=NULL, ...)

Parameters:

a, b: It specifies the intercept and the slope of the line

h: specifies y-value for horizontal line(s)

v: specifies x-value(s) for vertical line(s)

Returns: a straight line in the plot

8. Why library () function is used?

Ans Library functions are built-in functions that are grouped together and placed in a common location called library.

Each function here performs a specific operation. We can use this library functions to get the pre-defined output.

9. How to plot correlograms in r?

Ans:

```
install.packages("ggcorrplot")
```

```
library(ggcorrplot)
```

```
data(mtcars)
```

```
corr <- round(cor(mtcars), 1)
```

```
ggcorrplot(corr)
```

10.What is a geographical map?

Ans: A geographical map represents **an image of the earth's surface following a specific focus and / or purpose** (e.g. topographic, geological).

The maps library for R is a powerful tool for creating maps of countries and regions of the world. For example, you can create a map of the USA and its states in just three lines of code:

```
library(maps)
```

```
map("state", interior = FALSE)
```

```
map("state", boundary = FALSE, col="gray", add = TRUE)
```

11.How to plot a data set in single command?

Ans: Everyone of us tries doing this step at some point in time. We all look for one command using which we can plot all variables in the data set at once. Here's your answer.

You can use tabplot() package to accomplish this feat.

```
#plot data
```

```
> install.packages("tabplot")
```

```
> library(tabplot)
> tableplot(train)
```

12. Create three vectors x, y, z with integers and each vector has 3 elements. Combine the three vectors to become a 3×3 matrix A where each column represents a vector. Change the row names to a, b, c

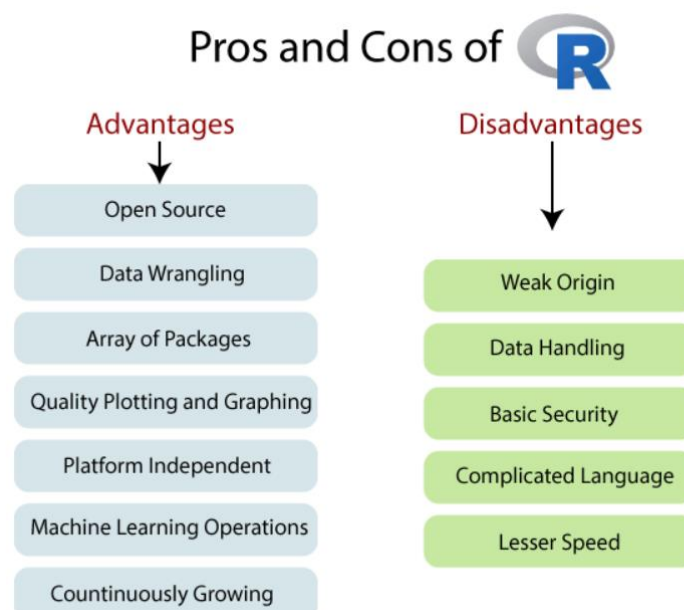
Ans

```
a<-c(1,2,3)
b<-c(4,5,6)
c<-c(7,8,9)
m<-cbind(a,b,c)
print("Content of the said matrix:")
print(m)
rownames(m) <- c("a","b","c")
print(m)
```

Part B

1. Explain the disadvantage and advantages of R language

Ans:



<https://www.javatpoint.com/r-advantages-and-disadvantages>

2. Use the functions `mean()` and `range()` to find the mean and range of:
- (a) The numbers 1, 2, . . . , 21
 - (b) The sample of 50 random normal values that can be generated from a normal distribution with mean 0 and variance 1

Ans: (a)

```
x<-c(1:21)
```

```
result.mean<-mean(X)
```

```
print(result.mean)
```

```
print(range(x))
```

(b) `rnorm(50, mean=0, sd=1)`

3. Write R function to check, count and display the number in a vector is odd. Computed Compound interest where P is the original money lent, A is what it amounts to in n years at R percent per year interest. Write R code to calculate the amount of money owed after n years, where n changes from 1 to 15 in yearly increments, if the money lent originally is 5000 pounds and the interest rate remains constant throughout the period at 11.5%.

4. Write R code to generate first n terms of a Fibonacci series.

```
nterm <- as.integer(readline(prompt= "Enter lenght of series:- "))
```

```
n1 <- 0
```

```
n2 <- 1
```

```
if (nterms <= 0){
```

```
  print("Print Enter a positive Number")
```

```
}else if (nterms == 1){
```

```
  print(n1)
```

```
}else if (nterm > 1){
```

```
  print(n1)
```

```
  print(n2)
```

```

count = 2
while (nterm > count){
  nth <- n1 + n2
  print(nth)
  n1 <- n2
  n2 <- nth
  count <- count + 1
}
}

```

5. Write a R program to calculate the GCD of two numbers using recursive computation.

```

Ans: num1 <- as.integer(readline(prompt = "Enter a number:- "))
    num2 <- as.integer(readline(prompt = "Enter another number:- "))
    if (num1 < num2){
      smaller <- num1
    }else{
      smaller <- num2
    }
    for (i in 1: smaller){
      if ((num1 %% i == 0) & (num2 %% i == 0)){
        hcf = i
      }
    }
    print(paste("HCF of number is ", hcf))

```

6. Write a function for Make a vector from 1 to 100. Make a for-loop which runs through the whole vector. Multiply the elements which are smaller than 5 and larger than 90 with 10 and the other elements with

0.1., so that you can feed it any vector you like (as argument). Use a for-loop in the function to do the computation with each element.

```
Ans v1 <- c(1:100)
v2 <- c(1:5)
v3 <- c(90:100)
v4 <- c()
for(i in v1){
  if ((i %in% v2)|(i %in% v3)){
    v4 <- append(v4,i*10)
  }else{
    v4 <- append(v4,i*.1)
  }
}
print(v4)
```

7. What is Recursion? What is Recursive Function in R?

Ans Recursion in R is when the function **calls itself**. This forms a loop, where every time the function is called, it calls itself again and again.

Recursive functions are functions that use the concept of recursion to perform repetitive or iterative tasks. They call themselves, again and again, this imitates a loop.

```
Example:- recursive.factorial <- function(x) {
  if (x == 0){
    return (1)
  }else{
    return (x * recursive.factorial(x-1))
  }
}
recursive.factorial(5)
```

8. Find the sum of natural numbers using the Recursive function.

```
Factorial <- function(N){  
  if (N == 0){  
    return(1)  
  }else{  
    return( N + Factorial (N-1))  
  }  
}  
Factorial(9)
```

9. How to create a heat map? What is the use of this map?

Ans

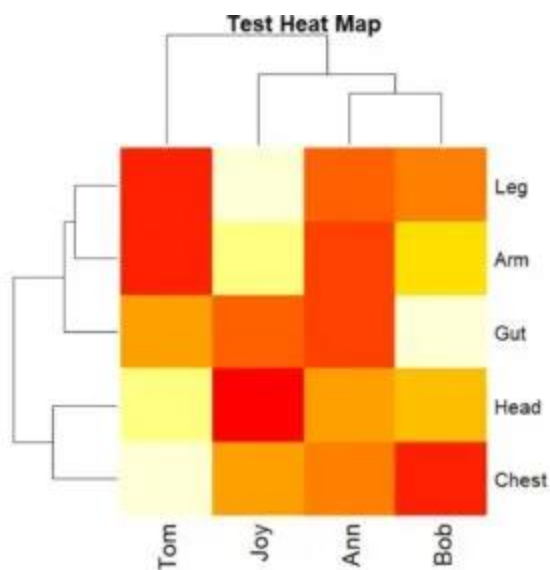
```
ds = data.frame(rnorm(5, 50, 20),rnorm(5, 50, 20),rnorm(5, 50, 20),rnorm(5, 50,  
20))
```

```
rn = c("Arm","Leg","Chest","Gut","Head")
```

```
cn = c("Ann","Bob","Tom","Joy")
```

```
x = data.matrix(ds, rownames.force = FALSE)
```

```
heatmap(x, labRow=cn, labCol=rn, main = "Test Heat Map")
```



Heat Map uses intensity (density) of colors to display relationship between two or three or many variables in a two dimensional image.

Question 1. Explain The Data Import In R Language.?

Answer :R provides to import data in R language. To begin with the R commander GUI, user should type the commands in the command Rcmdr into the console.

Data can be imported in R language in 3 ways such as:

- Select the data set in the dialog box or enter the name of the data set as required.
- Data is entered directly using the editor of R Commander via Data->New Data Set. This works good only when the data set is not too large.
- Data can also be imported from a URL or from plain text file (ASCII), or from any statistical package, or from the clipboard.

Question 2. Explain How To Communicate The Outputs Of Data Analysis Using R Language.?

Answer: Combine the data, code, and analysis results in a single document using knitr for Reproducible research done. Helps to verify the findings, add to them and engage in conversations. Reproducible research makes it easy to redo the experiments by inserting new data values and applying it to different various problems.

Question 3. What Is R?

Answer: R is a programming language that is used for developing statistical software and data analysis.

Question 4. How R Commands Are Written?

Answer: By using # at the starting of the line of code like #division commands are written.

Question 5. What Is T-tests() In R?

Answer :It is used to determine that the means of two groups are equal or not by using t.test() function.

Question 6. What Are The Disadvantages Of R Programming?

Answer :The disadvantages are:-

- Lack of standard GUI
- Not good for big data.
- Does not provide a spreadsheet view of data.

Question 7. What Is The Use Of With () And By () Function In R?

Answer :

with() function applies an expression to a dataset.

```
#with(data,expression)
```

By() function applies a function each level of a factors.

```
#by(data,factorlist,function)
```

Question 8. In R Programming, How Missing Values Are Represented?

Answer: In R missing values are represented by NA which should be in capital letters.

Question 9. What Is The Use Of Subset() And Sample() Function In R?

Answer: Subset() is used to select the variables and observations and sample() function is used to generate a random sample of the size n from a dataset.

Question 10. Explain What Is Transpose.?

Answer : Transpose is used for reshaping the data which is used for analysis. Transpose is performed by t() function.

Question 11. What Are The Advantages Of R?

Answer :The advantages are:-

- It is used for managing and manipulating of data.
- No license restrictions
- Free and open-source software.
- Graphical capabilities of R are good.
- Runs on many Operating system and different hardware and also run on 32 & 64 bit processors etc.

Question 12. What Is The Function Used For Adding Datasets In R?

Answer: For adding two datasets rbind() function is used but the column of two datasets must be same.

Syntax: rbind(x1,x2.....) where x1,x2: vector, matrix, data frames.

Question 13. How You Can Produce Co-relations And Covariances?

Answer:Cor-relations is produced by cor() and covariances is produced by cov() function.

Question 14. What Is Difference Between Matrix And Dataframes?

Answer :Dataframe can contain different type of data but matrix can contain only similar type of data.

Question 15. What Is Difference Between Lapply And Sapply?

Answer :lapply is used to show the output in the form of list whereas sapply is used to show the output in the form of vector or data frame.

Question 16. What Is The Difference Between Seq(4) And Seq_along(4)?

Answer :Seq(4) means vector from 1 to 4 (c(1,2,3,4)) whereas seq_along(4) means a vector of the length(4) or 1(c(1)).

Question 17. Explain How You Can Start The R Commander Gui.?

Answer :rcmdr command is used to start the R commander GUI.

Question 18. What Is The Memory Limit Of R?

Answer :In 32 bit system memory limit is 3Gb but most versions limited to 2Gb and in 64 bit system memory limit is 8Tb.

Question 19. How Many Data Structures R Has?

Answer :There are 5 data structure in R i.e. vector, matrix, array which are of homogenous type and other two are list and data frame which are heterogeneous.

Question 20. Explain How Data Is Aggregated In R.?

Answer :There are two methods that is collapsing data by using one or more BY variable and other is aggregate() function in which BY variable should be in list.

Question 21. How Many Sorting Algorithms Are Available?

Answer :There are 5 types of sorting algorithms are used which are:-

- Bubble Sort
- Selection Sort
- Merge Sort
- Quick Sort
- Bucket Sort

Question 22. How To Create New Variable In R Programming?

Answer :For creating new variable assignment operator '<-' is used For e.g. mydata\$sum <- mydata\$x1 + mydata\$x2.

Question 23. What Are R Packages?

Answer: Packages are the collections of data, R functions, and compiled code in a well-defined format and these packages are stored in the library.

Question 24. What Is The Workspace In R?

Answer: Workspace is the current R working environment which includes any user-defined objects like vectors, lists, etc.

Question 25. What Is The Function Which Is Used For Merging Of Data Frames Horizontally In R?

Answer : Merge() function is used to merge two data frames

Eg: `Sum<-merge(data frame1,data frame 2,by='ID')`

Question 26. What Is The Function Which Is Used For Merging Of Data Frames Vertically In R?

Answer : rbind() function is used to merge two data frames vertically.

Eg. `Sum<- rbind(data frame1,data frame 2)`

Question 27. What Is The Power Analysis?

Answer : It is used for experimental design .It is used to determine the effect of given sample size.

Question 28. Which Package Is Used For Power Analysis In R?

Answer : Pwr package is used for power analysis in R.

Question 29. Which Method Is Used For Exporting The Data In R?

Answer : There are many ways to export the data into another formats like SPSS, SAS , Stata , Excel Spreadsheet.

Question 30. Which Packages Are Used For Exporting Of Data?

Answer : For excel xlsReadWrite package is used and for sas,spss ,stata foreign package is implemented.

Question 31. How Impossible Values Are Represented In R?

Answer : In R NaN is used to represent impossible values.

Question 32. Which Command Is Used For Storing R Object Into A File?

Answer : Save command is used for storing R objects into a file.

Syntax: `>save(z,file='z.Rdata')`

Question 33. Which Command Is Used For Restoring R Object From A File?

Answer : load command is used for storing R objects from a file.

Syntax: `>load('z.Rdata')`

Question 34. What Is The Use Of Coin Package In R?

Answer : Coin package is used to achieve the re randomization or permutation based statistical tests.

Question 35. Which Function Is Used For Sorting In R?

Answer :order() function is used to perform the sorting.

Question 36. What Happens When The Application Object Does Not Handle An Event?

Answer The event will be dispatched to your delegate for processing.

Question 37. Explain App Specific Objects Which Store The App Contents.?

Answer :The app specific objects are Data model objects that store app's contents.

Question 38. Explain The Purpose Of Using Uindow Object?

Answer :UIWindow object coordinates the one or more views presenting on the screen.

Question 39. How To Create Axes In The Graph?

Answer :Using axes() function custom axes are created.

Question 40. What Is The Use Of Abline() Function?

Answer :abline() function is add the reference line to a graph.

Syntax:-

abline(h=yvalues, v=xvalues)

Question 41. Why Vcd Package Is Used?

Answer :vcd package provides different methods for visualizing multivariate categorical data.

Question 42. What Is Ggobi?

Answer :GGobi is an open source program for visualization for exploring high dimensional typed data.

Question 43. What Is Iplots?

Answer :It is a package which provide bar plots, mosaic plots, box plots, parallel plots, scatter plots and histograms.

Question 44. What Is The Use Of the Lattice Package?

Answer :lattice package is to improve on base R graphics by giving better defaults and it have the ability to easily display multivariate relationships.

Question 45. What Is Fitdistr() Function?

Answer: It is used to provide the maximum likelihood fitting of univariate distributions. It is defined under the MASS package.

Question 46. Which Data Structures Are Used To Perform Statistical Analysis And Create Graphs.?

Answer: Data structures are vectors, arrays, data frames and matrices.

Question 47. What Is The Use Of Sink() Function?

Answer :It defines the direction of output.

Question 48. Why Library() Function Is Used?

Answer :This function is used to show the packages which are installed.

Question 49. Why Search() Function Is Used?

Answer : By this function we see that which packages are currently loaded.

Question 50. On Which Type Of Data Binary Operators Are Worked?

Answer :Binary operators are worked on matrices, vectors and scalars.

Question 51. What Is The Use Of Doby Package?

Answer :It is used to define the desired table using function and model formula.

Question 52. Which Function Is Used To Create Frequency Table?

Answer :Frequency table is created by table() function.

Question 53. What Is The Of Use Matrix Package?

Answer :Matrix package includes those function which support sparse and dense matrices like Lapack, BLAS etc.

Question 54. Define Matlab Package.?

Answer :Matlab package includes those wrapper functions and variable which are used to replicate matlab function calls.

Question 55. What We Use Party Package?

Answer :It is used to provide a non-parametric regression for ordinal, nominal, censored and multivariate responses.

Question 56. Define Cluster.stats() ?

Answer :It is define in fpc package which provide a method for comparing the similarity of two clusters solution using different validation criteria.

Question 57. What Is Pvclust() Function?

Answer :It comes under the pvclust package which provides p-values for hierarchical clustering.

Question 58. Which Function Performs Classical Multidimensional Scaling?

Answer :cmdscale() function is used to perform classical multidimensional scaling.

Question 59. What Is The Use Of Boot.sem() Function?

Answer :It is used to bootstrap the structural equation model.

Question 60. What Is Factominer?

Answer :It is a package which includes quantitative and qualitative variables. It also includes supplementary variables and observations.

Question 61. What Is Principal() Function?

Answer :It is define in psych package which is used to rotate and extract the principal components.

Question 62. Define Auto.arima().?

Answer :It is used to handle the seasonal as well as non-seasonal ARIMA models.

Question 63. What Is The Use Of Forecast Package?

Answer :It provides the functions which are used for automatic selection of ARIMA and exponential models.

Question 64. Define Lda().?

Answer :lda() is used to print the discriminant functions which is based on centered variable.

Question 65. What Is The Use Of Mass Package?

Answer :MASS functions include those functions which performs linear and quadratic discriminant function analysis.

Question 66. What Is Coxph()?

Answer :It is a function which is used to model the hazard function on the set of predictor variable.

Question 67. Define Survival Analysis.?

Answer :It includes number of techniques which is used for modeling the time to an event.

Question 68. Define Poison Regression.?

Answer :It is used to predict the outcome variable which represents counts from the given set of continuous predictor variable.

Question 69. What Is Logistic Regression?

Answer :Logistic regression is used to predict the binary outcome from the given set of continuous predictor variables.

Question 70. Which Variables Are Represented By Upper Case Letters?

Answer :Categorical factors are represented by upper case letters.

Question 71. Which Variables Are Represented By Lower Case Letters?

Answer :Numerical variables are represented by lower case letters.

Question 72. Define Barlett.test().?

Answer :Barlett.test() is used to provide a parametric k-sample test of the equality of variances.

Question 73. What Is Fligner.test()?

Answer :It is a function which provides a non-parametric k sample test of the equality of variances.

Question 74. What Is The Use Of Manova?

Answer :By using MANOVA we can test more than one dependent variable simultaneously.

Question 75. Define Plotmeans().?

Answer :It is define under gplots package which includes confidence intervals and it produces mean plot for single factors.

Question 76. What Is Robustbase?

Answer :It is a package which provides basic robust statistics including model selection methods.

Question 77. Why Car Package Is Used?

Answer :It provide a variety of regression including scatter plots, variable plots and it also enhanced diagnostic.

Question 78. Define Relaimpo Package.?

Answer :It is used to measure the relative importance of each of the predictor in the model.

Question 79. Define Stepaic() Function.?

Answer :It is define under the MASS package which performs stepwise model selection under exact AIC.

Question 80. What Is The Use Of Diagnostic Plots?

Answer :It is used to check the normality, heteroscedasticity and influential observations.

Question 81. What Is Npmc?

Answer :It is a package which gives nonparametric multiple comparisons.

School of Data Science & Forecasting
Devi Ahilya Vishwavidyalaya, Indore
Mid Term Test II MBA /M.Sc. : 2nd Semester (Batch 2018-20)
Subject: Statistical Programming in R
Note: Attempt any Five questions contain equals marks.

1. What do you meant by loop in R .explain different types of loop

Ans: . A loop is a control statement that allows multiple executions of a statement or a set of statements. The word ‘looping’ means cycling or iterating
There are three types of loop in R programming:

- For Loop
- While Loop
- Repeat Loop

For Loop in R

It is a type of control statement that enables one to easily construct a loop that has to run statements or a set of statements multiple times. For loop is commonly used to iterate over items of a sequence.

While Loop in R

It is a type of control statement which will run a statement or a set of statements repeatedly unless the given condition becomes false. It is also an entry controlled loop, in this loop the test condition is tested first, then the body of the loop is executed, the loop body would not be executed if the test condition is false.

Repeat Loop in R

It is a simple loop that will run the same statement or a group of statements repeatedly until the stop condition has been encountered. Repeat loop does not have any condition to terminate the loop, a programmer must specifically place a condition within the loop’s body and use the declaration of a break

statement to terminate this loop. If no condition is present in the body of the repeat loop then it will iterate infinitely.

2.What will be the output of the following R programming code?

```
x<-5
if(x%%2==0)
  print("X is an even number")
else
  print("X is an odd number")
```

Ans:

X is odd number

3.Find the sum of an exponential series consist n terms.

```
myfun=function(n){

  mysum <- 0

  while(n!=0){

    mysum = mysum + exp(n)

    n=n-1

  }

  return(mysum)

}
```

myfun(3)

6. Write an R programme to find factor of a number.

Ans:

```
print_factors <- function(x) {
```

```

print(paste("The factors of",x,"are:"))

for(i in 1:x) {

  if((x %% i) == 0) {

    print(i)

  }

}

}

```

4. Write an R function that returns the real roots of the quadratic $ax^2 + bx + c$. The function should take a, b and c as arguments and return appropriate messages if the values entered don't specify a quadratic or if there are no real roots.

Ans:

```

quadraticRoots <- function(a, b, c) {

  print(paste0("You have chosen the quadratic equation ", a, "x^2 + ", b, "x + ",
c, "."))

  discriminant <- (b^2) - (4*a*c)

  if(discriminant < 0) {
    return(paste0("This quadratic equation has no real numbered roots."))
  }
  else if(discriminant > 0) {
    x_int_plus <- (-b + sqrt(discriminant)) / (2*a)
    x_int_neg <- (-b - sqrt(discriminant)) / (2*a)
  }
}

```

```

return(paste0("The two x-intercepts for the quadratic equation are ",
              format(round(x_int_plus, 5), nsmall = 5), " and ",
              format(round(x_int_neg, 5), nsmall = 5), "."))
}
else #discriminant = 0 case
  x_int <- (-b) / (2*a)
  return(paste0("The quadratic equation has only one root. This root is ",
              x_int))
}

```

Q1 = What is the procedure to read the excel file in which data has been entered in the third sheet

Answer: Steps to read the Excel file into R

- **Steps 1 – Install the readxl package**

In the R Console type the command

“install.packages(readxl)”

- **Steps 2 – Prepare your excel file**

Let’s suppose that you have an Excel file with some data and let’s say that the Excel file name is **product_list**, and your goal is to read that file into R.

- **Step 3 – Read the excel file into R**

First, you have to import the library using the command **“library(‘readxl’)”**

Then we use the **“read_excel()”** function to read the file and pass them the file path and sheet name that you want

Command: **read_excel(“Filename.xlsx”, sheet = ‘your sheet name’)**

Q2 = How can we present the missing value in CSV and how it use in R

Answer: In R, missing value are represented by the symbol **NA**(not available)

Use CSV file in R

- **Approach 1 : read.csv**

If your CSV file is small enough, you may simply use Base R's read.csv function to import it.

Command: `data <- read.csv('filename.csv', header=TRUE,
stringsAsFactors=FALSE)
head(data)`

- **Approach 2: read_csv**

You can use the read CSV function from the “readr” package if you're working with larger files.

Command: `library(readr)
data2 <- read_csv('filename.csv')
head(data2)`

- **Approach 3: fread**

If your CSV is exceptionally huge, the fread function from the data is the fastest way to import it into the R.

Command: `library(data.table)
data3 <- fread("filename.csv")
head(data3)`

Q3 You have two dataframe "M" and "N". M has 34 rows and N has 46 rows how will merge the data frame what will the number of rows in the resultant dataframe

Answer:

Command: `df <- rbind(M,N)
print(df)`

In this total number of row is 80

Q5 The student taught by 3 different method gave the following performance(marks)

A = 19, 9, 12, 16, 7, 14, 11

B = 8, 13, 3, 17, 15

C = 14, 11, 10, 9, 15, 16

Calculate the analysis of variance

Q6 Write about the scatter plots and histogram with example and explain its importance

Answer: Scatter plots: Scatter plots are the graphs that present the relationship between two variables in a data set. It represents data points on a two-dimensional plane or on a Cartesian system. The independent variable or attribute is plotted on the X-axis, while the dependent variable is plotted on the Y-axis.

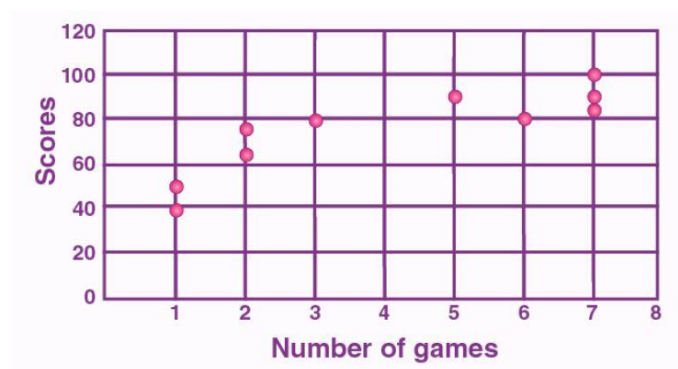
Example : Let us understand how to construct a scatter plot

Draw a scatter plot for the given data that shows the number of games played and scores obtained in each instance.

No. of games	3	5	2	6	7	1	2	7	1	7
Scores	80	90	75	80	90	50	65	85	40	100

X-axis or horizontal axis: Number of games

Y-axis or vertical axis: Scores



Importance of scatter plot

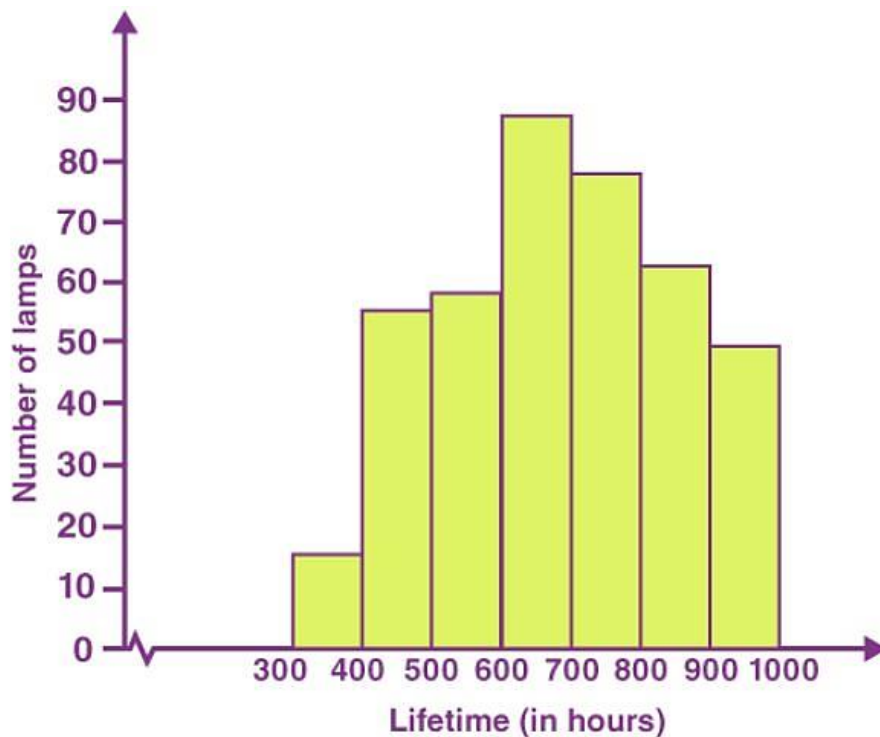
Scatter plots are important in statistics because they can show the extent of correlation, if any, between the values of observed quantities

Histogram: It is represented by a frequency distribution by the mean of rectangles whose widths represent class intervals and whose areas are proportional to the corresponding frequencies

Example:

Question: The following table gives the lifetime of 400 neon lamps. Draw the histogram for the below data.

Lifetime (in hours)	Number of lamps
300 – 400	14
400 – 500	56
500 – 600	60
600 – 700	86
700 – 800	74
800 – 900	62
900 – 1000	48



Importance of Histogram

Creating a histogram provides a visual representation of data distribution. Histograms can display a large amount of data and the frequency of the data values. The median and distribution of the data can be determined by a histogram. In addition, it can show any outliers or gaps in the data.

Q7 How to plot multiple curves in the same graph? Explain with example?

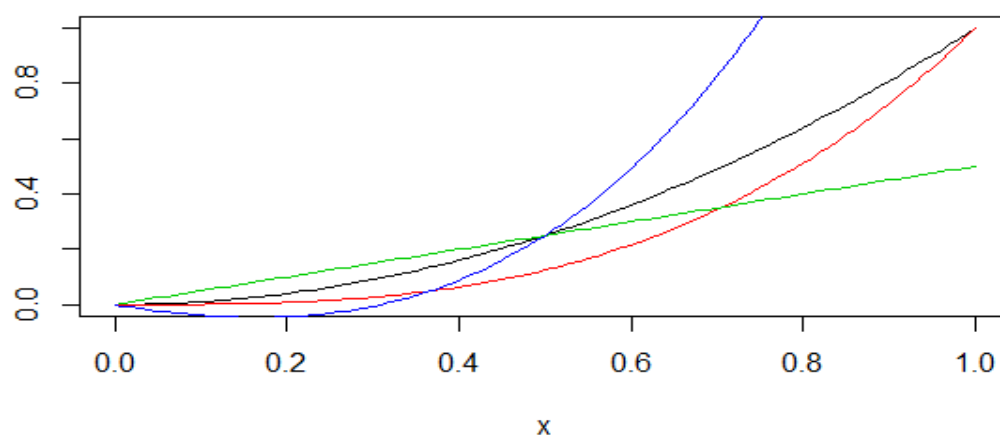
Answer: Base R supports a function `curve()` which can be used to visualize a required function curve. It supports various parameters to edit the curve according to requirements.

To draw multiple curves in one plot, different functions are created separately and the `curve()` function is called repeatedly for each curve function. The call for every other `curve()` function except for the first one should have added an attribute set to `TRUE` so that multiple curves can be added to the same plot. To differentiate among the different colors are used.

Example:

```
function1<- function(x){x ** 2}
function2<-function(x){x ** 3}
function3<-function(x){x / 2}
function4<-function(x){2*(x ** 3)+(x ** 2)-(x / 2)}
curve(function1, col = 1)
curve(function2, col = 2, add = TRUE)
curve(function3, col = 3, add = TRUE)
curve(function4, col = 4, add = TRUE)
```

Output



Q8 Write the Script to calculate the HCF of the number

```
num1 = as.integer(readline(prompt = "Enter first number: "))
num2 = as.integer(readline(prompt = "Enter second number: "))
hcf <- function(x,y){
  if(x<y){
    smaller = x
  }else{
    smaller = y
  }
  for(i in 1:smaller){
    if((x %% i == 0 ) && ( y %% i == 0)){
      hcf = i
    }
  }
  return(hcf)
}
print(paste("The H.C.F. of", num1,"and", num2,"is", hcf(num1, num2)))
```

Q9 Is R have any graphical user interface, If yes write about it

Yes R have graphical user interface

R is an open source programming language and software environment for statistical computing and graphics. It consists of a language together with a run-time environment with a debugger, graphics, access to system functions, and scripting.

Graphical User Interfaces for R

RStudio	Professional software for R with a code editor, debugging & visualization tools
Rattle	R Analytic Tool To Learn Easily: Data Mining using R
StatET for R	Eclipse based IDE (integrated development environment) for R
RKWard	Easy to use and easily extensible IDE/GUI
JGR	Universal and unified graphical user interface for R
R Commander	A Basic-Statistics GUI for R
Deducer	Intuitive, cross-platform graphical data analysis system

Q10 What is the use of corgram() function?

The corgram function produces a graphical display of a correlation matrix, called a correlogram. The cells of the matrix can be shaded or colored to show the correlation value.

Command: corgram(x, type = NULL, order = FALSE, labels, panel = panel.



Q11 Draw a random sample of size 50 geometric distribution of N = 60, M = 15 and n = 6

```
# Specify x-values for dgeom function
x_dgeom <- seq(1, 50, by = 1)

# Apply dgeom function
y_dgeom <- dgeom(x_dgeom, prob = 0.5)

# Plot dgeom values
plot(y_dgeom)
```

Q13 Create a decision tree in R with the help of suitable example?

Answer: Let us now examine this concept with the help of an example, which in this case is the most widely used “readingSkills” dataset by visualizing a decision tree for it and examining its accuracy.

Import required libraries and Load the dataset readingSkills and execute

```
head(readingSkills)
library(datasets)
library(caTools)
library(party)
library(dplyr)
```

```

library(magrittr)

data("readingSkills")
head(readingSkills)

sample_data = sample.split(readingSkills, SplitRatio = 0.8)
train_data <- subset(readingSkills, sample_data == TRUE)
test_data <- subset(readingSkills, sample_data == FALSE)

model<- ctree(nativeSpeaker ~ ., train_data)
plot(model)

# testing the people who are native speakers
# and those who are not
predict_model<-predict(ctree_, test_data)

# creates a table to count how many are classified
# as native speakers and how many are not
m_at <- table(test_data$nativeSpeaker, predict_model)
m_at

# Determining the accuracy of the model developed
ac_Test <- sum(diag(table_mat)) / sum(table_mat)
print(paste('Accuracy for test is found to be', ac_Test))

```

Q14 = what is heat map where it is used?

Answer: A heat map is a two-dimensional representation of data in which values are represented by colors. A simple heat map provides an immediate visual summary of information. More elaborate heat maps allow the viewer to understand complex data sets.

Where it is used

Heatmaps are used in various forms of analytics but are most commonly used to show user behavior on specific webpages or webpage templates. Heatmaps can be used to show where users have clicked on a page, how far they have scrolled down a page or used to display the results of eye-tracking tests.