

R interview questions and answers

Q.No.1. Name the different data structures in R? Briefly explain.

Answer: – Organizing the data in a computer for effectively and efficiently used in the future is called Data Structure. It is a particular technique for arranging the data because it reduces our time complexity. It consumes less space also. It can hold multiple values.

R Programming Language is one of the programs, where we can use many tools for holding multiple values. It can also carry one-dimensional and multidimensional data. We can easily work on identical (Homogeneous) and various data types (Heterogeneous).

There are many data structures from which the most important are given below:

1. List- A generic object which has an ordered collection of the objects is called a List. It is heterogeneous. A list can have many data structures in it like a list of vectors, a list of functions, etc.
2. Data frame – It is used to store the tabular data. It is of two dimensions and it can contain heterogeneous data. We can work with multiple types of data on it.
3. Vectors – A collection of basic data types which are in an ordered way is called vectors. It is a one-dimensional data structure for that it is homogeneous also.
4. Array – It is a multidimensional data structure where we can store homogeneous data.
5. Matrices – It is a two-dimensional data structure that has rows and columns in the rectangular set. It is a homogeneous data structure where we can perform multiple operations.
6. Factors – It is used to categorize the data like true/false, male/female, in/out, etc.

Q.No.2. Advantages of using an applied family of functions in R?

Answer: -The applied family of functions is a built-in family which appears with the built-in packages in R. It is already installed in it.

It allows us to manipulate data frames, vectors, arrays, etc. It works more effectively than loops and also gives better performance from them which is faster at the execution level. It reduces the need for explicitly creating a loop in R.

The list of the apply family are as follows: –

1. `apply()` function: – It helps to apply a function on rows or columns of a data frame.

Syntax: – `apply()`

2. `lapply()` function: – It takes a list as an argument and applies a function to each element of the list by looping.

Syntax: – `lapply()`

3. `sapply()` function : – It is more advanced version than `lapply()` however it works same as `lapply()`. It also takes a list as an argument and applies a function to each element of the list by looping. The only difference is in output generalization. Where `lapply()` returns a list as an output every time, `sapply` returns certain algorithms as output.

Syntax: – `sapply()`

4. `tapply()` function: – It can be applied to vectors and factors. The data which contain different subgroup and we have to apply a specific function on each subgroup that time we can use it.

Syntax: – `tapply()`

5. `mapply()` function: – It is a multivariate version of the `sapply()` function where we apply the same function to multiple arguments.

Q.No. 3. What do you mean by shinyR?

Answers: – ShinyR is easy to build interaction between web applications through R Language, Where you can host any standalone application on a webpage or embed or program them in R Markdown documents or build a dashboard. You can also extend the Shiny application with various themes (CSS), widgets (HTML), and actions (JavaScript). It unites the computational power of R with the interactivity of the modern web. It is also very easy to write a program in shinyR. It comes with a variety of built-in input widgets with minimum syntax. We can plot diagrams, tables, and all those things which we can do in an R language.

Q.No. 4. What do you mean by Random Forest? How would you build a

Random Forest in R? Answer: – The Random Forest is used for classification as well as regression of the data. It creates decision trees on data samples from where it gets the prediction from each data. It also selects the best solution through voting. It is a supervised learning algorithm. We know that decision trees are popular for machine learning tasks. The random trees have some overlaps where we can build systems to read the data redundantly with various trees and look for the trends, patterns, and structures that support a given data outcome.

To build a Random Forest in R we have to follow the given steps: –

1. Create a Bootstrapped Data Set

2. Create a Design Tree
3. Predict the outcome of the data point
4. Evaluate the Model

Q.No. 5. What are the functions available in the “dplyr” package?

Answer: – The functions which are available in the “dplyr” package are as follows: –

1. Select() function: -Allows us to rapidly zoom in on a useful subset using operations that usually only work on numeric variable positions.
2. group_by() function : – It allows us to group by a modified columns.
3. mutate() function : -It is useful to add new columns that are functions of previous existing columns.
4. filter() function : -Allows us to select a subset of rows in a data frame.
5. summarize() function :- Allows us to collapses a data frame to a single row.
6. relocate() function : – Allows us to change the column order.
7. slice() function : – Allows us to select, remove and duplicate rows.
8. desc() function : – Allows us to arrange the column in descending order.

Q.No.6. How do you write a custom function in R? Provide an example.

Answer: – There is hundreds of built-in function. Hadley Wickham defined function as “You can do anything with functions that you can do with vectors: You can assign them to variables, store them in lists, pass them as arguments to other functions, create them inside functions and even return them as the result of the function.”

Generally, we can customize function in our own way. We can do everything in arithmetic, logical, command as well as graphical works with it. Here are some examples of the R programming functions.

```
fahrenheit_to_celsius <-function(temp_F) {  
  
temp_C <- (temp_F -32)*5/9  
  
return(temp_C)  
  
}
```

Q.No. 7. What do you understand by the confusion matrix?

Answer: – It is a table that is used to describe the classification model performance on a set of test data for which the true values are known. It is very simple to understand but only the related terms can be confusing. Confusion Matrix allows us to find the measure recall, accuracy, precision, etc. It visualizes the accuracy of a

classifier by comparing the actual and predicted classes. The binary confusion matrix is composed of squares:

True Positive (TP): It predicts values correctly predicted as actual positive.

True Negative (TN): It predicts values correctly predicted as an actual negative.

False Positive (FP): It predicts values incorrectly predicted as actual positive.

False Negative (FN): It predicts values correctly predicted as negative.

Q. No. 8. List packages in R that are used for data imputation.

Answer: – The list of packages in R that are used for data imputation is as follows: –

1. MICE: It stands for Multivariate Imputations through Chained Equations. It is the fastest for imputing values. The methods which are used by this package are as follows:
 1. PMM (Predictive Mean Matching): for numeric variables.
 2. logreg (Logistic Regression): for binary variables.
 3. polyreg (Bayesian polytomous regression): for factor variables
 4. Proportional odds model
2. Amelia: It performs multiple imputations which generate imputed data sets to deal with the missing values. It helps to reduce bias and increase efficiency.
3. Hmisc: It is a multipurpose package that is useful to analyze the data, imputing the missing values, advanced tables makings, linear regression, logistics regression, logistics fitting, high–level graphics, etc. It has wide range of functions as like `impute()`, `areglumpote()` etc.
4. missForest: It uses for the implementation of a random forest algorithm. It is a non-parametric imputation method that applies to various variable types. It builds a random forest model for each variable and then it uses the model to predict the missing values in the variable with the help of values which it is observed.
5. MI: it stands for the multiple imputations. It provides us several features for dealing with missing values and uses the predictive mean matching method. It uses the Bayesian version of the regression model to handle the issues of separation. It also automatically detects the irregularities in data such as high collinearity among the variables.

Q.No.9. How do you build a linear regression model in R?

Answer: To build a linear regression model in R we have to follow the following steps:-

1. Experiment with gathering a sample of observed values.
2. Create a relationship model using the `lm()` function in R.
3. Find the coefficients from the model created.

4. Create the mathematical equation.
5. Find a summary of the relationship model to know the average error in prediction which is also called residuals.
6. Predict the new data from using the predict() function in R.

Q.No. 10. How to install packages in R?

Answer: To install packages in R we have to perform the following steps:

1. Part 1
 1. Type "install.packages("gplots")" and then press the Enter or Return key.
 2. If you have already loaded a package from a server in the R session then R will automatically install the packages. If not the R will automatically prompt you to choose a mirror. Again choose one close to unless you want to watch a loading bar slowly inching its way to fulfilling.
3. Part 2
 1. Type "Library(gplots)" and then press the Enter key.
 2. R will give lots of output because it needs to install other packages as required for gplots.
3. Part 3
 1. You will only need to do the same which is describing in part 1 once time on your computer.
 2. You only need to do part 2 each time you choose and restart R.

Q.No. 11. What do you understand by Rmarkdown?

Answer-It provides us a unified authority of framework for data Science, combining our code, prose commentary, and its results. Documents of R Markdown are fully supported by dozens of output formats i.e. like PDFs, Slideshare, Word files, and many more which we can reproduce many times.

Simply R Markdown is a text-based file format that allows us to include descriptive texts, code blocks, and code output. We can also run the code in R and using a package called Knitr. We can export the text which is formatted .rmd file to a greatly rendered, sharable format like pdf, HTML, etc. When we knit the code is run and so our outputs including plots, graphs, and other figures appear in the rendered document.

Q.No. 12. How can you load the .csv file in R?

Answer- You can load the .csv file in R by following the following steps:

1. The first thing in this process is to getting and setting up the working directory. We need to choose the correct working path of the CSV (comma separated values) file.

2. We can check the default working directory by using `getwd()` function and we can also change the directory by using the function `setwd()`.
3. After the setting of the working path as prescribed earlier, we need to import the data set or a CSV file.
4. After getting the data frame as mentioned above, we can analyze the data. we can also extract the particular information from the data frame.

By this process, you can read the CSV files in R with the use of the `read.csv()` function.

Q.No. 13. How can you do a cross-product of two tables in R?

Answer- We can do a cross-product of two tables in R by using `CJ()` function. It produces data. table out of the two vectors. This function does the Cartesian Product or Cross product of two data. tables.

Q.No. 14. How do you extract a word from a string?

Answer- We extract a word from a string by using the `word()` function in the R language. This function is mainly used for the extracted word from a string that is from the position that is specified as an argument. We can use `String`, `start`, `end`, `sep`, etc. as an argument.

Q.NO. 15. What do you mean by correlation in R?

Answer- To evaluate the association between two or more variables we use Correlation. It has Correlation coefficients which are indicators of the strength of the linear relationship between two different variables say x and y . The correlation coefficient greater than zero indicates that a positive relationship, while a value less than zero indicates that a negative relationship. A negative correlation is also called inverse correlation which is a key concept in the creation of diversified portfolios that can better withstand portfolio volatility.

The most common Correlation coefficient is generated by the Pearson product-moment correlation which is used to measure the linear relationship between two variables. The Pearson Correlation is also called parametric correlation.

Q.No. 16. How do you find out the number of missing values in a particular dataset?

Answer- To find out the number of missing values in a particular dataset we use `is.na()` function which returns a logical vector with `TRUE` in the element location containing missing values is represented by `NA`. The `is.na()` will work on vectors, data frames, matrices, lists, etc.

Q.No. 17. How do you rename a column in a data frame?

Answer- To rename a column in the data frame we can use two functions either `names()` or `colnames()`. For this, we have to perform 2 steps for it which as follows:

1. Get the column names either using the function `names()` or `Colnames()`
2. Change column names where name= xyz

Q.No. 18. How would you do left join and right join in R language?

Answer- Left join will take all of the values from the table as we specify as left and match them to the records from the table on the right. The syntax for the left join is as follows:

```
1 leftJoinDf<-
2 left_join(tableA, tableB, by="Customer.ID")
3 View(leftJoinDf)
```

Right Join is the opposite of a left join. In this function the table specified second within the joint statement will be the one that the new table takes all of its values from.

```
1 rightJoinDf<-
2 right_join(tableA, tableB, by="Customer.ID")
3 View(rightJoinDf)
```

Q.No. 19. How do you make a box-plot using “plotly”?

Answer- We can make a box-plot using “plotly” function by follow the sample synatax for it which are as follows:

```
1 Library(plotly)
2 fig <- plot_ly(y = ~rnorm(50),
3 Type = "box")
4 fig <- fig %>% add_trace
5 (y = ~rnom(50, 1))
6 fig
```

However, we can choose exclusive or inclusive algorithms to compute quartiles and we can also modify the algorithm for computing quartiles.

Q.No. 20. What do you mean by `evaluate_model()` from “statisticalModeling”?

Answer- It is used to find the model outputs for specified inputs. This is identical to the general `predict()` function, except it will choose sensible values by default. This simplifies to get a quick look at the model values. There are several arguments of it like `model`, `data`, `on_training`, `nlevels`, `at`, etc. This function is set up to look easily at typical outputs.

Q.No. 21. What do you understand by the “`initialize()`” function?

Answer- The “initialize()” function is used internally by some imputation algorithms for finding the missing values which are imputed with the mean for vectors of class “numeric”, also with the median for the vector of class “integers” and last but not least the mode for vectors of class “factor”. It initializes the missing values through a rough estimation of missing values in a vector according to its type.

Q.No. 22. How can you find the mean of one column w.r.t. another?

Answer- We can find the mean of one column concerning another by using ColMeans() function along with sapply() function. It is always helpful to find the mean of the multiple columns. We can also find the mean of multiple columns through Dplyr functions. summarise_if() function along with is.numeric() function is used to get the mean of the multiple column. With the help of the summarise_if() function, the mean of numeric columns of the data frame is calculated.

Q.No. 23. What is the PCA model in R? Explain in detail.

Answer- The PCA model stands for “Principal Component Analysis”. It has vast operation because the correlations and covariance are always helpful to extract the result. Principal Component Analysis is widely used and it is a very popular statistical method for reducing data with many dimensions by projecting the data with fewer dimensions using linear combinations of the variables which are known as a principal component. The new projected components are uncorrelated with each other and are ordered so that its first few components retain most of the variation present in the original variables. It is also useful to independent variables which are correlated with each other and can be employed in exploratory data analysis or for making predictive models. It reveals important features of the data such as outliers and departures from a multi-normal distribution.

Q.No. 24. What do you mean by Random Walk Model?

Answer- The Random Walk Model is the integration of a mean zero white noise series. It is also called the Basic Time Series Model which means that the cumulative sum of a mean zero WN (White Noise) series. When a series follows a Random Walk Model, then it is said to be non – stationary. We can rationalize it by taking a first-order difference of the time series, which means Zero Mean White Noise.

Q.No. 25. What is the White noise model?

Answer- All the variables have the same variance and each value has a zero correlation with all other values in the series that is called the White noise model. It is a sequence of random numbers and cannot be predicted. It suggests improvements could be made to the predictive model. This means that a time series is a white noise if the variables are independent and identically distributed with a mean of zero.

Q.No. 26. If you are given a vector of values, how would you convert it into a time series object?

Answer- A vector of values can be converted into a time series object by using the `ts()` function. The syntax is as follows:

```
1 ts(vector, start=, end=, frequency= )
```

Where start is the first and end is the last times of observation and frequency is the number of observations per unit time i.e. 12 for monthly, 6 for half-yearly, 4 for quarterly, and 1 for annually.

Q.No. 27. How do you facet data using the ggplot2 package?

Answer- The facet data using the ggplot2 package is one of the best graphical statistical analysis tools where the graph is partitioned in multiple panels by the levels of the group which we specified.

For splitting in a vertical direction we use syntax like

```
1 bp + facet_grid(xyz ~ .)
```

For splitting in a horizontal direction we can use syntax like

```
1 bp+ facet_grid(. ~ xyz)
```

The above-described syntaxes are used in a single variable now we are describing that syntax that is used in two variables.

Rows are abc and columns are xyz

```
bp + facet_grid(abc ~ xyz)
```

Rows are xyz and columns are abc

```
bp + facet_grid(xyz ~ abc)
```

In `Facet()` function we can use multiple parameters like we can adjust facet scales, can give the labels, and also wrap the graphs through `facet_wrap`.

Q.No. 28. Give examples of the functions in Stringr?

Answer – There are many examples of the functions of Stringr from which the main examples are as follows:

1. `Str_count()`: It count the number of patterns. Syntax= `str_count(x, pattern)`

2. `Str_locate()`: It gives the location or position of the match. Syntax= `str_locate(x, pattern)`
3. `Str_extract()`: It extract the text of the match. Syntax= `Str_extract(x, pattern)`
4. `Str_match()`: It extract parts of the match defined by parenthesis. Syntax = `str_match(x, pattern)`
5. `Str_split()`: It splits a string into multiple pieces. Syntax = `str_split(x, pattern)`

Q. No. 29. What is while and for loop in R? Give examples?

Answer- A while loop is a loop where the statement keeps running until the condition which is specified is satisfied. The syntax for a while loop is following:

```
While (condition){ Exp }
```

We must write a closing condition at some point otherwise it will go on indefinitely.

Example of the While loop program are as follows:

```
#Create a variable with value 1
```

```
Begin <- 1
```

```
#Create the loop
```

```
While (begin<=5){
```

```
  ('This is the loop number', begin)
```

```
  begin <- begin+1
```

```
  Print(begin)
```

```
}
```

For loop: The loop which is used to iterate over a vector in R programming is called for a loop. The syntax for the 'for' loop is as follows;

```
For (val in sequence)
```

```
{
```

Statement

```
}
```

The example of “for” loop is as follows:

```
X <- c(2,5,3,9,8,11,6)
```

```
Count <- 0
```

```
for (val in x) {
```

```
  If (val %% 2 ==0)
```

```
    count = count+1
```

```
  (count)
```

```
}
```

Q.No. 30. Compare R and Python.

Answer- Python is a more general approach to data science while R is used for statistical analysis. The primary objective of python is deployment and production while the primary objective of R is Data analysis and statistics. Python is used by most programmers and Developers while R is used by Research and development scientist and professionals. Python is very easy to learn while R is difficult to learn. Python has many packages and libraries like pandas, scipy, scikit learn, TensorFlow, etc. while R consists of various packages and libraries like caret, zoo, tidyverse, ggplot2, etc.

Q.No. 31. What is the difference between library() and require() functions in R Language?

Answer- If the requested package does not exist then, the library() function gives an error result by default, While the require() function gives a warning message and returns a logical value i.e. false if the requested package is not found and true if the package is loaded in a system.

Q.No. 32. What do you mean by t-test() in R?

Answer- It is used to determine whether the means of two groups are equal to each other. The assumptions for the test of both groups are sampled from normal distributions with equal variances. The null hypotheses(0) of the two means are equal, and their alternatives are that means are not equal. We know that that under the null hypothesis, we can calculate at-statistics that will follow a t-distribution with $n1 + n2 - 2$ degrees of freedom.

The `t.test()` function is available in R for performing the “t-tests”. We can use this function in a function like simulation, we need to know that out how the extract the t-statistic from the output of the `t.test` function. For this function, the R help page has a detailed list of what the object returned by the function contains.

Q.No. 33. How is with() and By() function used in R?

Answer- `with()` function evaluates an R expression in an environment constructed based on a data frame. It takes the variables of our data into account. For example, we can compute the sum of our two and more variables. It can make handling our data much easier, especially when many variables are involved in our expressions.

`by()` function apply a function to each level of a factor or factors. The “`by()`” function is an object-oriented wrapper for `tapply` applied to the data frames. An object of class “`by`”, giving the result for each subset. This is always a list if `simply` is false, otherwise a list or array.

Q.No. 34. How are missing values in R represented?

Answer- The missing values are represented by the symbol `NA`. Impossible values are represented by the symbol `NaN`(Not-a-number). `NA` is used for numeric as well as string data also.

Q.No. 35. What is transpose in R?

Answer:- The conversion of the rows of the matrix in column and column of the matrix in a row is known as transpose. In R we can do it in two ways first by using the `t()` function and by iterating over each value using Loops.

Q.No. 36. Advantages of R?

Answer- The advantages of R language are as follows: –

1. R is an open-source programming language. We can work with R without any need for a license or a fee.
2. R has a vast array of packages. These packages are applied to all areas of the industry.

3. It facilitates quality plotting and graphics. There are many libraries such as ggplot2 and plotly advocate for aesthetic and visually appealing graphs that set R apart from the other programming languages.
4. R is highly compatible and can be paired with many other programming languages also like c, c++, Java, and Python.
5. It can be integrated with technologies like Hadoop and various other database management systems.
6. It provides various facilities for carrying out machine learning operations like regression, classification, and artificial neural networks.
7. R is dominant among the other programming languages for developing statistical tools.

Q.No. 37. Disadvantages of R?

Answer- There are a lot of advantages of R but still some areas of technological system R has some disadvantages also which are as follows:

1. R base package does not have support for 3D graphics.
2. R utilizes more memory as compared with Python because In R, the physical memory stores the objects.
3. R lacks basic security. This is an essential feature of most programming languages like Python, Because of this, there are several restrictions with R.
4. R packages and the R programming languages are much slower than other languages like MATLAB and Python.
5. R is tough to learn to compare to Python.
6. Programmers without knowledge of packages may find it difficult to implement algorithms in R.

Q.No. 38. Which of the function do you use to add datasets in R.

Answer- library() function package provides the infrastructure to make test datasets available within R. They are very large to store within the R package but R prevents them from being included in OSS-licensed packages. If you want to add a new dataset to the text data package the follow the following steps:

1. Create an R file named prefix_*.R in the R/folder, Where *is the name of the dataset.
2. Supported prefixes include
 1. dataset_
 2. lexicon_
3. Inside that file create 3 functions named
 1. download_*
 2. process_*
 3. dataset_*
4. Add the process_*() function to the named list process_function in the file process_functions.R

5. Add the `download_*`() function to the named list `download_function` in the file `download_function.R`
6. Modify the `print_info` list in the `info.R` file.
7. Add `dataset_*.R` to the `@include` tags in `download_function.R`
8. Add the dataset to the table in `README.Rmd`
9. Add the dataset to `_pkgdown.yml`
10. Write a bullet in the `RAVI.md` file.

Q.No. 39. Difference between matrix and data frames?

Answer- Matrix is an $m \times n$ array with a similar data type. It is a homogeneous collection of data sets that are arranged in a two-dimensional rectangular organization. It has a fixed number of rows and columns. We can perform many arithmetic operations on the R matrix. It has great use in Economics, Engineering, and Electronics. It is also useful in probability and statistics.

DataFrames are used for storing data tables. It can contain multiple data types in multiple rows and multiple columns (Field). It just looks like an excel sheet. It has column and row names where each one has a unique number. It can store multiple data types like numeric, character or factor, etc. It is heterogeneous. We can do statistics, processing data, transpose, etc.

Q.No. 40. Difference between `seq(4)` and `seq_along(4)`?

Answer- If the `seq()` is called with the one unnamed numerical argument data of length 1, as a result, it returns an integer sequence from 1 to the value of the argument. In a question `seq(4)` is the command returns the integers 1,2,3,4. While `seq_along(4)` produces the vector of indices of a given factor.

Q.No. 41. Does R have a memory limit? What is it?

Answer- Yes, R has a memory limit. On 32-bit Windows, it cannot exceed up to 3GB and most versions are limited to 2 GB. The minimum is currently 32Mb. If the 32-bit R version is run on a 64-bit version of Windows we get the maximum value of obtainable memory is 4GB. For 64-bit versions of R under the 64-bit version Windows system, the limit is 8TB.

Q.No. 42. Name the sorting algorithms available in R?

Answer- The sorting algorithms available in 'R' are as follows:

1. Quick Sort
2. Selection Sort
3. Merge Sort
4. Bucket Sort
5. Bubble Sort

6. Bin Sort
7. Radix Sort
8. Shell Sort
9. Heapsort

Q.No. 43. How do you export data in R?

Answer- We can export data in R in various applications and programs. We are describing some of them in the following section:

1. R to an Excel
`Library(xlsx)`

`Write.xlsx(my data, "c:/mydata.xlsx")`

2. R to SAS
`#Write out text data file and`

`#an SAS program to read it`

`Library(foreign)`

`Write.foreign(mydata, "C:/mydata.txt", "c:/mydata.sas", package="SAS")`

3. R to Stata
`#export data frame to Stata binary format`

`Library(foreign)`

`Write.dta(mydata, "c:/mydata.dta")`

Q.No. 44. What is coxph()?

Answer- It is the function that is used to calculate the cox proportion hazards regression model in R. It is the time-dependent variables, time-dependent strata, multiple events per subject, and other extensions which are incorporated using the numerous process formulation. The data for a subject is presented as multiple rows or "observations", each of which applies to an interval of observation (start, stop).

Q.No. 45. Define the MATLAB package?

Answer- Matlab and R are the two interactive, high-level programming languages used in scientific computing. The languages have a lot in common but have very

different targets and foci. R is primarily used by the statistical community for advanced data analysis and research in statistical methodology while Matlab is primarily used by engineers for image processing, differential equations, and so on.

The RMatlab package provides a path for R to do Matlab functions and Matlab to do R functions. We can start Matlab from R, or we can embed R within Matlab. The RMatlab package allows us to call R functions from within the Matlab process using the same address space. This makes inter-system communication which is very fast and allows the objects to be shared directly through reference. And since R is embedded within Matlab, it can make calls back to Matlab. It allows for a range of interesting computations and ensures that we can use the two languages back to back and program in the best convenient environment for our tasks.

R can be access Matlab by starting a separate Matlab process where it also sends commands to it. This is the Engine linkage to Matlab.

Q.No. 46. How do you use corrgram() function?

Answer- The corrgram() function produces a graphical display of a correlation matrix. Its cells can be shaded or colored to show the correlation value. In corrgram() function the non-numeric column in the data will be ignored.

Q.No. 47. What is the UIWindow object?

Answer- The presentation of one or more views on a screen is coordinated by the UIWindow object. In iOS application usually only has one window, while View multiple. Windows and Views both are used to present your application's content on the screen. Windows provide a basic container for your application's views but do not have any visible content. The view is a segment of a window where you can fill up with some content.

Q.No. 48. What is the lazy function evaluation in R?

Answer- Lazy evaluation is a programming strategy that allows a symbol to be evaluated only when needed, in another word a symbol can be defined in a function and it will only be evaluated when it is needed. Lazy evaluation is implemented in R as it allows a program to be more efficient when used interactively.

Q. No. 49. Write the difference between “%%” and “%/%”?

Answer – “%%” indicates $x \bmod y$ and “%/%” indicates integer division. Both are arithmetic operators.

Q.No. 50. How is the forecast package packages?

Answer- It provides methods and tools for displaying and analyzing univariate time series forecasts including exponential smoothing through the state-space model and automatic ARIMA modeling. The forecast packages will remain in their current state and maintained with bug fixes only.

Q.No.51. What is auto.arima()?the

Answer- It is a forecasting function for time series. auto.arima() function returns the best ARIMA model according to either AIC, AICC, or BIC value. It searches for all possible models within the order constraints provided.

Q.No. 52. What do you understand by reshaping of data in R?

Answer- It is used in the data frames where Data Reshaping has changed the way data is organized into rows and columns. It helps in extracting data from the rows and columns of the data frame. It is an easy task but there are situations when we need the data frame in a format that is different from the format in which we received it. In R we have so many functions to merge, split and change the rows and columns in a data frame.

Q, No. 53. What is the full form of CFA?

Answer- CFA stands for Confirmatory Factor Analysis.

Q.No. 54. What is the coin package in R?

Answer- It provides a flexible implementation of the abstract framework and a large set of convenience functions to implement classical and non-classical test procedures within the framework. The coin package provides us an implementation of a general framework for conditional inference procedures which is known as permutation tests.

Q.No. 55. What do you mean by workspace in R?

Answer- The workspace is our current R working environment where the objects like vectors, matrices, list, function, etc. are included. At the end of the session, we can save an image of the current workspace that is automatically reloaded the next time in R when it is started. We can give commands interactively at the R user prompt and check the history through arrow keys.

Q.No. 56. How many data structures does R have?

Answer- There are six types of data structure in R which are as follows:

1. Vectors
2. Lists
3. Dataframes
4. Matrices
5. Arrays
6. Factors