Marwadi University	**Marwadi University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Introduction to R and R Studio (01CT0106)	**Aim: Introduction to R Programming and R Studio**	
Experiment: 01	**Date: 14/02/2023**	**Enrollment No: 92200133030**

Aim: Introduction to R Programming and R Studio

IDE: R Studio

Theory:

History of R:

R is a programming language and free software environment for statistical computing and graphics that is supported by the R Foundation for Statistical Computing. The R language is widely used among statisticians and data miners for developing statistical software and data analysis.

R is an implementation of the 5-programming language combined with lexical saving semantics inspired by Scheme. S was created by John Chambers in 1976, while at Bell Labs. There are some important differences, but much of the code written for runs unaltered

R was created by Ross Ihaka anti Robert Gentleman at the University of Auckland, New Zealand, and is currently developed by the R Development Core Team, of which Chambers is a member. R is named partly after the first names of the first two R authors and partly as a play on the name of S. The project was conceived in 1992, with an initial version released in 1995 and a stable beta version in 2000

R and its libraries implement a wide variety of statistical and graphical techniques} including linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, and others. R is easily extensible through functions and extensions, and the R community is noted for its active contributions in terms of packages, Many of R's standard functions are written in R itself, which makes it easy for users to follow the algorithmic choices made.

R is an interpreted language; users typically access it through a command-line interpreter. If a user types 2+2 at the R command prompt and presses enter, the computer replies with 4, as shown below:

```
>2+2

[1] 4
```

Features of R

As stated earlier, R is a programming language arid software environment for statistical analysis, graphics representation and reporting. The following are the important features of R

- R is a well-developed, simple and effective programming language which includes conditionals, loops, user defined recursive functions and input and output facilities,
- R has an effective data handling and storage facility,
- R provides a suite of operators for calculations on arrays, lists, vectors and matrices.
- R provides a large, coherent and integrated collection of tools for data analysis.

- R provides graphical facilities for data analysis and display either directly at the computer or printing at the papers.

To Install R and R Packages

1. Open an in tern et browser and go to wwwir-project.org.
2. Click the "download R" link in the middle of the page under "Getting Started."
3. Select a CRAN location (a mirror site) and click the corresponding link.
4. Click on the "Download Ft for WINDOWS" link at the top of the page,
5. Click on the file containing the latest version of R under "Files."
6. Save the .pkg file, double-click it to open, and follow the installation instructions.
7. Now that R is installed, you need to download and install RStudio.

To Install RStudio

1. Go to www.rstud io.com and dick on the "Download RStudio" button,
2. Click on "Download RStudio Desktop.'"
3. Click on the version recommended for your system, or the latest Mac version, save the idling file on your computer, double-click it to open, and then drag and drop it to your applications folder.

To Install R Packages

The capabilities of R. are extended through user-created packages, which allow specialized statistical techniques, graphical devices. import/export capabilities, reporting tools, etc. These packages are developed primarily in R., and sometimes in Java, C, C++, and Fortran. The R packaging system is also used by researchers to create compendia to organize research data code and report files in a systematic way for sharing and public archiving.

A core set of packages is included with the installation of R, with more than 12,500 additional packages (as of May 20113[update]) available at the Comprehensive R Archive Network (CRAN).

Packages are collections of R functions, data, and compiled code in a well defined format The directory where packages are stored is called the library. R comes with a standard set of packages. Others are available for download and installation. Once installed, they have to be loaded into the session to be used.

Installing Packages

The most common place to get packages from is CRAN. To install packages from CRAN you use install.packages("package name"). For instance, if you want to install the ggplot2 package, which is a very popular visualization package, you would type the following in the console:

Syntax
# install package from CRAN install.packages("ggplot2")

Loading Packages
Once the package is downloaded to your computer you can access the functions and resources provided by the package in two different ways:
# load the package to use in the current R session library(packagename)

Getting Help on Packages
For more direct help on packages that are installed on your computer you can use the help and vignette functions. Here we can get help on the ggplot2 package with the following: help(package = "ggplot2") # provides details regarding contents of a package vignette(package = "ggplot2") # list vignettes available for a specific package vignette("ggplot2-specs") # view specific vignette

Comments
Comments are like helping text in your r program and they are ignored by the interpreter while executing your actual program. single comment is written using # in the beginning of the statement as follows −
# My first program in R Programming
R does not support multi-line comments but you can perform a trick which is something as follows −

Variables in R
Variables are used to store data, whose value can be changed according to our need. Unique name given to variable (function and objects as well) is identifier.
Rules for writing Identifiers in R
1. Identifiers can be a combination of letters, digits, period (.) and underscore (_).
2. It must start with a letter or a period. If it starts with a period, it cannot be followed by a digit.
3. Reserved words in R cannot be used as identifiers.

Valid identifiers in R
total, Sum, .fine.with.dot, this_is_acceptable, Number5
Invalid identifiers in R

tot@l, 5um, _fine, TRUE, .0ne

Observation and Learning: