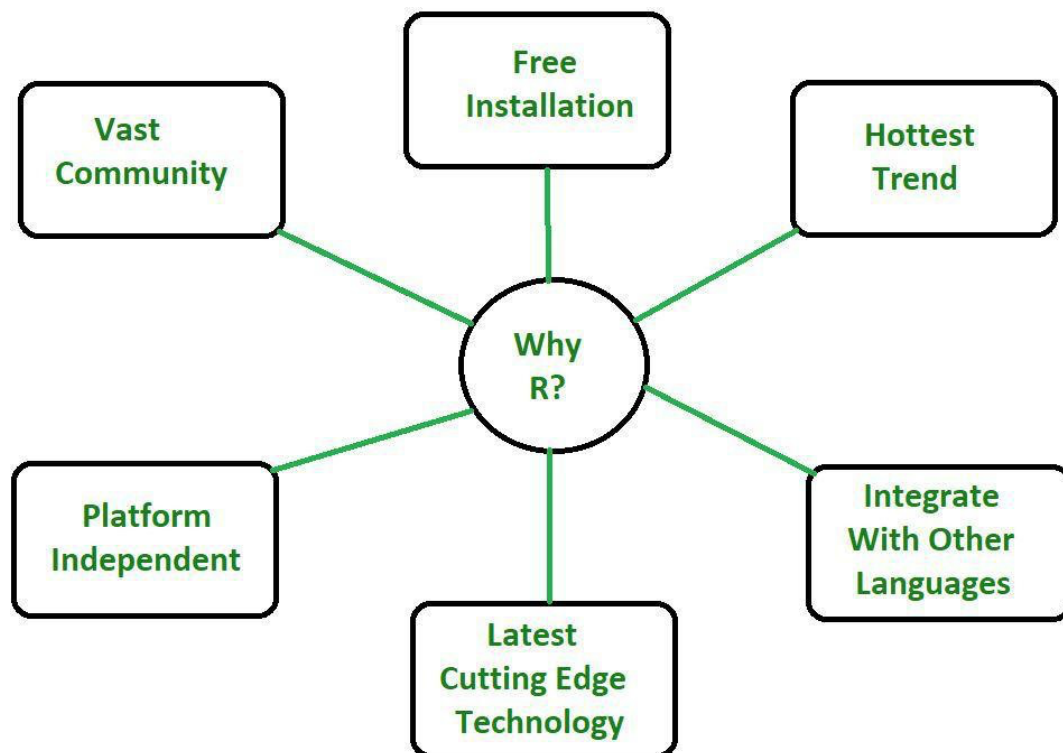


Unit 1: Introduction to R programming

Syllabus Contents: What is R? Basic Features of R, Programming features of R, Installing R and RStudio, RStudio Overview, working in the R Console, Getting Help in R and Quitting RStudio.

What is R?

- The R Language stands out as a powerful tool in the modern era of statistical computing and data analysis.
- Widely embraced by statisticians, data scientists, and researchers, the R Language offers an extensive suite of packages and libraries tailored for data manipulation, statistical modeling, and visualization.
- R programming language is an implementation of the S programming language.
- It also combines with lexical scoping semantics inspired by Scheme. Moreover, the project was conceived in 1992, with an initial version released in 1995 and a stable beta version in 2000.



- R programming is a leading tool for machine learning, statistics, and data analysis, allowing for the easy creation of objects, functions, and packages.
- Designed by Ross Ihaka and Robert Gentleman at the University of Auckland and developed by the R Development Core Team, R

Language is platform-independent and open-source, making it accessible for use across all operating systems without licensing costs.

- Beyond its capabilities as a statistical package, R integrates with other languages like C and C++, facilitating interaction with various data sources and statistical tools. With a growing community of users and high demand in the Data Science job market, R is one of the most sought-after programming languages today.

Basic Features of R/ Programming features of R

The R Language is renowned for its extensive features that make it a powerful tool for data analysis, statistical computing, and visualization. Here are some of the key features of R:

1. Comprehensive Statistical Analysis:

R language provides a wide array of statistical techniques, including linear and nonlinear modeling, classical statistical tests, time-series analysis, classification, and clustering.

2. Advanced Data Visualization:

With packages like ggplot2, plotly, and lattice, R excels at creating complex and aesthetically pleasing data visualizations, including plots, graphs, and charts.

3. Extensive Packages and Libraries:

The Comprehensive R Archive Network (CRAN) hosts thousands of packages that extend R's capabilities in areas such as machine learning, data manipulation, bioinformatics, and more.

4. Open Source and Free:

R is free to download and use, making it accessible to everyone. Its open-source nature encourages community contributions and continuous improvement.

5. Platform Independence:

R is platform-independent, running on various operating systems, including Windows, macOS, and Linux, which ensures flexibility and ease of use across different environments.

6. Integration with Other Languages:

R language can integrate with other programming languages such as C, C++, Python, Java, and SQL, allowing for seamless interaction with various data sources and computational processes.

7. Powerful Data Handling and Storage:

Prepared by Ms. A. A. Khan, Assistant Professor (CSE-AIML), DYP CET

R efficiently handles and stores data, supporting various data types and structures, including vectors, matrices, data frames, and lists.

8. Robust Community and Support:

R has a vibrant and active community that provides extensive support through forums, mailing lists, and online resources, contributing to its rich ecosystem of packages and documentation.

9. Interactive Development Environment (IDE):

RStudio, the most popular IDE for R, offers a user-friendly interface with features like syntax highlighting, code completion, and integrated tools for plotting, history, and debugging.

10. Reproducible Research:

R supports reproducible research practices with tools like R Markdown and Knitr, enabling users to create dynamic reports, presentations, and documents that combine code, text, and visualizations.

Installing R and RStudio


How do I install R and RStudio?

Install R from the CRAN website by selecting the appropriate version for your OS, then download and install RStudio from its official website to get a user-friendly IDE for R.

To Install R and R Studio on Windows we will have to download R and R Studio with the following steps.

Step 1: First, you need to set up an R environment in your local machine. You can download the same from [r-project.org](https://www.r-project.org).

Lecture Notes on Introduction to R programming



Don't want to download or install anything? Get started with RStudio on [Posit Cloud for free](#). If you're a professional data scientist to download RStudio and also need common enterprise features, don't hesitate to [book a call with us](#).

1: Install R

RStudio requires R 3.3.0+. Choose a version of R that matches your computer's operating system.

[DOWNLOAD AND INSTALL R](#)

2: Install RStudio

[DOWNLOAD RSTUDIO DESKTOP FOR WINDOWS](#)

Size: 214.34 MB | SHA-256: FE62B784 | Version: 2023.09.1+494 | Released: 2023-10-17

Install R and R Studio

You have to download both the applications first go with R Base and then install RStudio. after click on install R you will get a new page like this.



CRAN
[Mirrors](#)
[What's new?](#)
[Search](#)
[CRAN Team](#)

About R
[R Homepage](#)
[The R Journal](#)

Software
[R Sources](#)
[R Binaries](#)
[Packages](#)
[Task Views](#)
[Other](#)

Documentation
[Manuals](#)
[FAQs](#)
[Contributed](#)

Donations
[Donate](#)

The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages, **Windows and Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora/Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2023-10-31, Eye Holes) [R-4.3.2.tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- Contributed extension [packages](#)

Questions About R

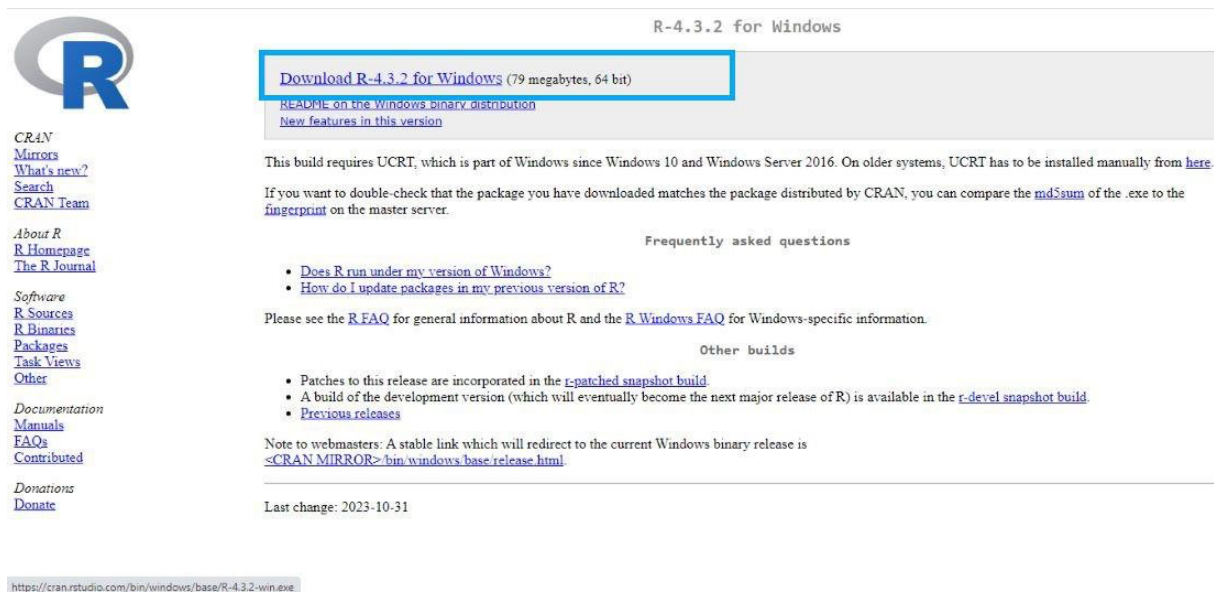
- If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

Supporting CRAN

Install R and R Studio

Here we can select the linux , mac or windows any one according to users system. you have to click on for which you want to install.

Lecture Notes on Introduction to R programming



The screenshot shows the CRAN website for R-4.3.2 for Windows. On the left is a sidebar with links: CRAN, Mirrors, What's new?, Search, CRAN Team, About R, R Homepage, The R Journal, Software, R Sources, R Binaries, Packages, Task Views, Other, Documentation, Manuals, FAQs, Contributed, Donations, and Donate. The main content area has the R logo and the title 'R-4.3.2 for Windows'. A blue box highlights the 'Download R-4.3.2 for Windows (79 megabytes, 64 bit)' link. Below it are links for 'README on the Windows binary distribution' and 'New features in this version'. A paragraph states: 'This build requires UCRT, which is part of Windows since Windows 10 and Windows Server 2016. On older systems, UCRT has to be installed manually from [here](#). If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [md5sum](#) of the .exe to the [fingerprint](#) on the master server.' There are sections for 'Frequently asked questions' and 'Other builds'. A note to webmasters provides a stable link: '<CRAN MIRROR>/bin/windows/base/R-4.3.2-win.exe'. The last change is noted as '2023-10-31'.

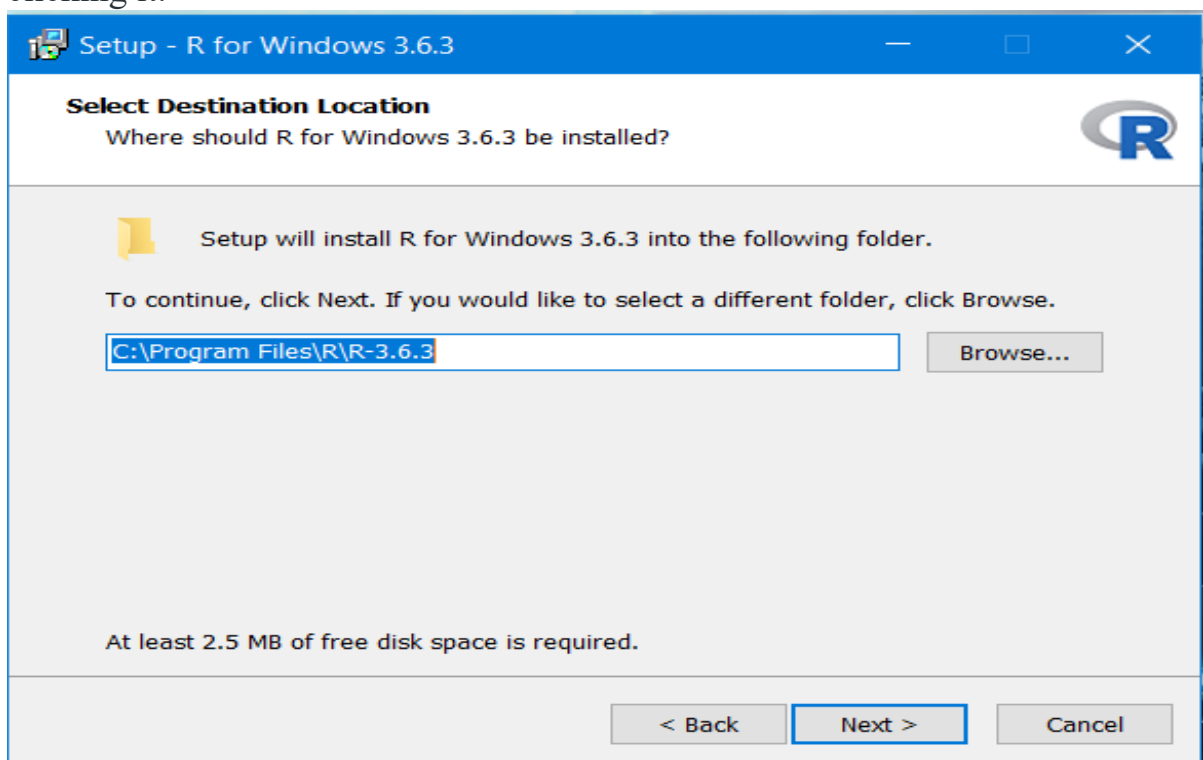
<https://cran.rstudio.com/bin/windows/base/R-4.3.2-win.exe>

Install R and R Studio

now click on the link show above in image so R base start downloading and after again go to main page and download and click on Install RStudio.

Steps to Install R and R Studio

Step 1: After downloading R for the Windows platform, install it by double-clicking it.



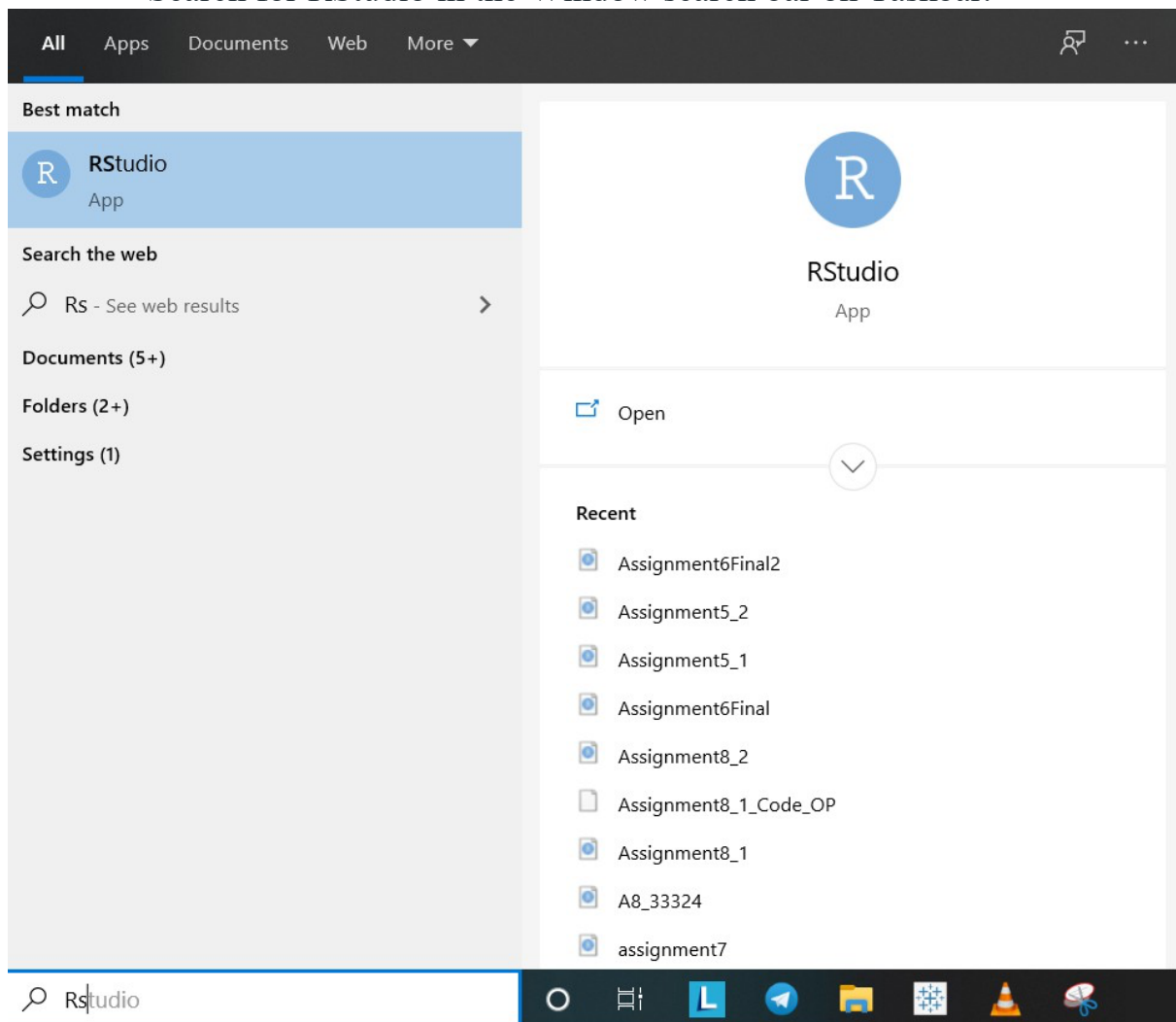
Step 2: Download R Studio from their official page. Note: It is free of cost (under AGPL licensing).

Step 3: After downloading, you will get a file named “RStudio-1.x.xxxx.exe” in your Downloads folder.

Step 4: Double-click the installer, and install the software.

Step 5: Test the R Studio installation

- Search for RStudio in the Window search bar on Taskbar.

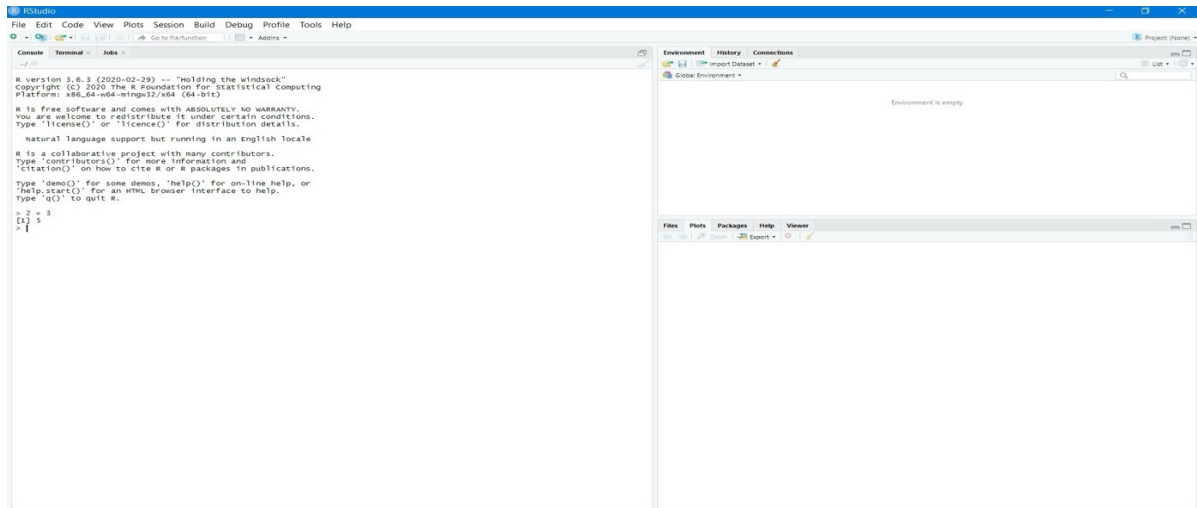


- Start the application.
- Insert the following code in the console.

Input : `print("Hello world!")`

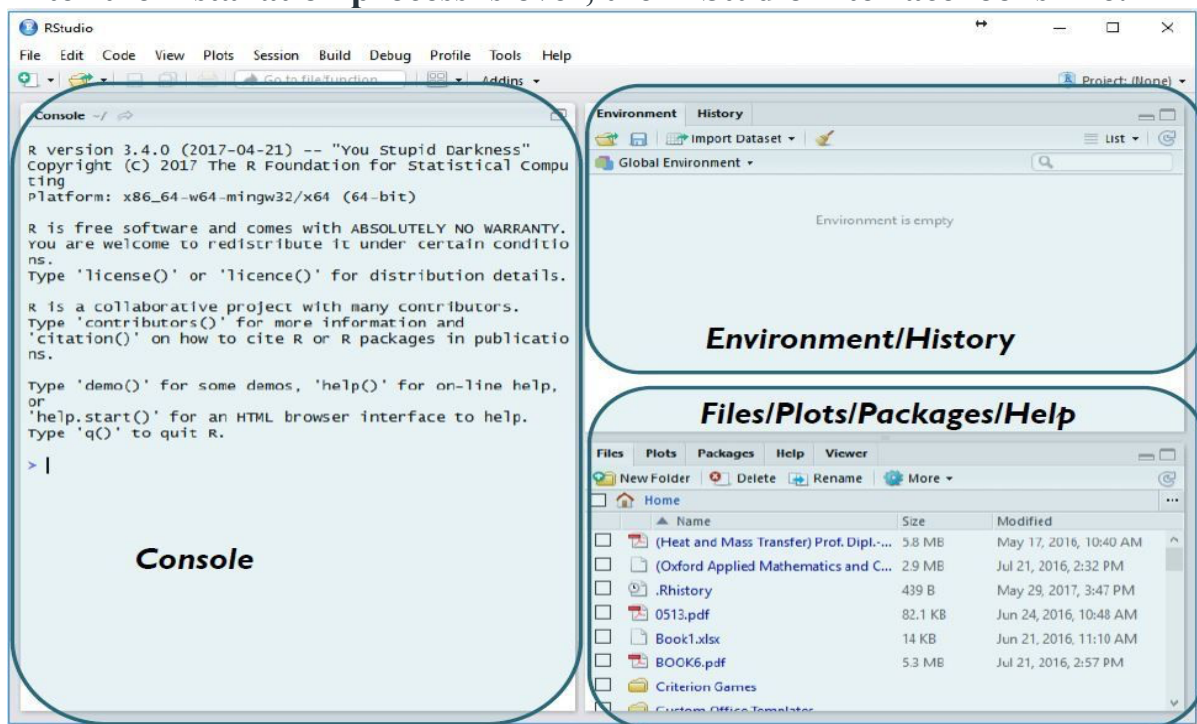
Output : `[1] "Hello world!"`

Lecture Notes on Introduction to R programming



RStudio Overview

After the installation process is over, the R Studio interface looks like:



- The console panel(left panel) is the place where R is waiting for you to tell it what to do, and see the results that are generated when you type in the commands.
- To the top right, you have the Environmental/History panel. It contains 2 tabs:
 - **Environment tab:** It shows the variables that are generated during the course of programming in a workspace that is temporary.

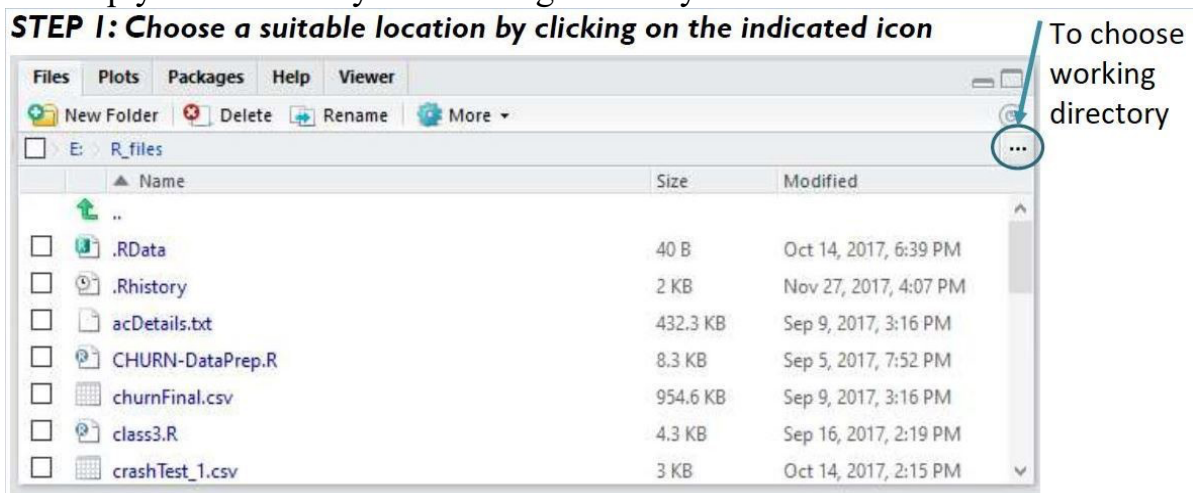
- **History tab:** In this tab, you'll see all the commands that are used till now from the start of usage of R Studio.
- To the right bottom, you have another panel, which contains multiple tabs, such as files, plots, packages, help, and viewer.
 - The **Files tab** shows the files and directories that are available within the default workspace of R.
 - The **Plots tab** shows the plots that are generated during the course of programming.
 - The **Packages tab** helps you to look at what are the packages that are already installed in the R Studio and it also gives a user interface to install new packages.
 - The **Help tab** is the most important one where you can get help from the R Documentation on the functions that are in built-in R.
 - The final and last tab is that the **Viewer tab** which can be used to see the local web content that's generated using R.

working in the R Console and Getting Help in R and Quitting RStudio.

R is always pointed at a directory on our computer. We can find out which directory by running the **getwd()** function.

Note: this function has no arguments. We can set the working directory manually in two ways:

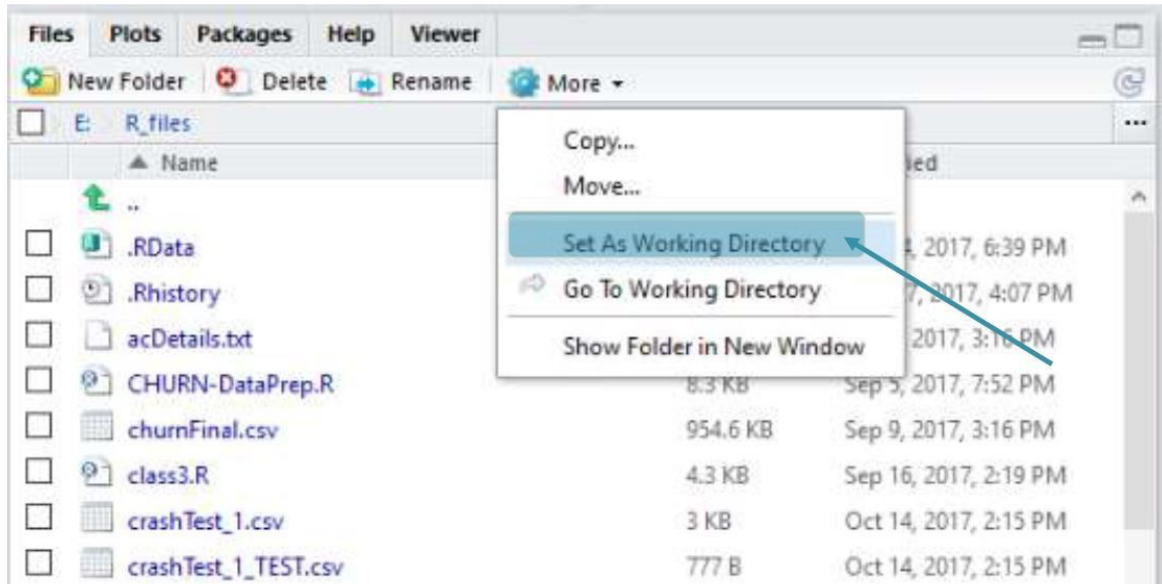
- **The first way is to use the console and using the command `setwd("directorypath")`.**
You can use this function `setwd()` and give the path of the directory which you want to be the working directory for R studio, in the double codes.
- **The second way is to set the working directory from the GUI.**
To set the working directory from the GUI you have to click on this 3 dots button. When you click this, this will open up a file browser, which will help you to choose your working directory.



Lecture Notes on Introduction to R programming

- Once you choose your working directory, you need to use this setting button in the more tab and click it and then you get a popup menu, where you need to select “Set as working directory”.

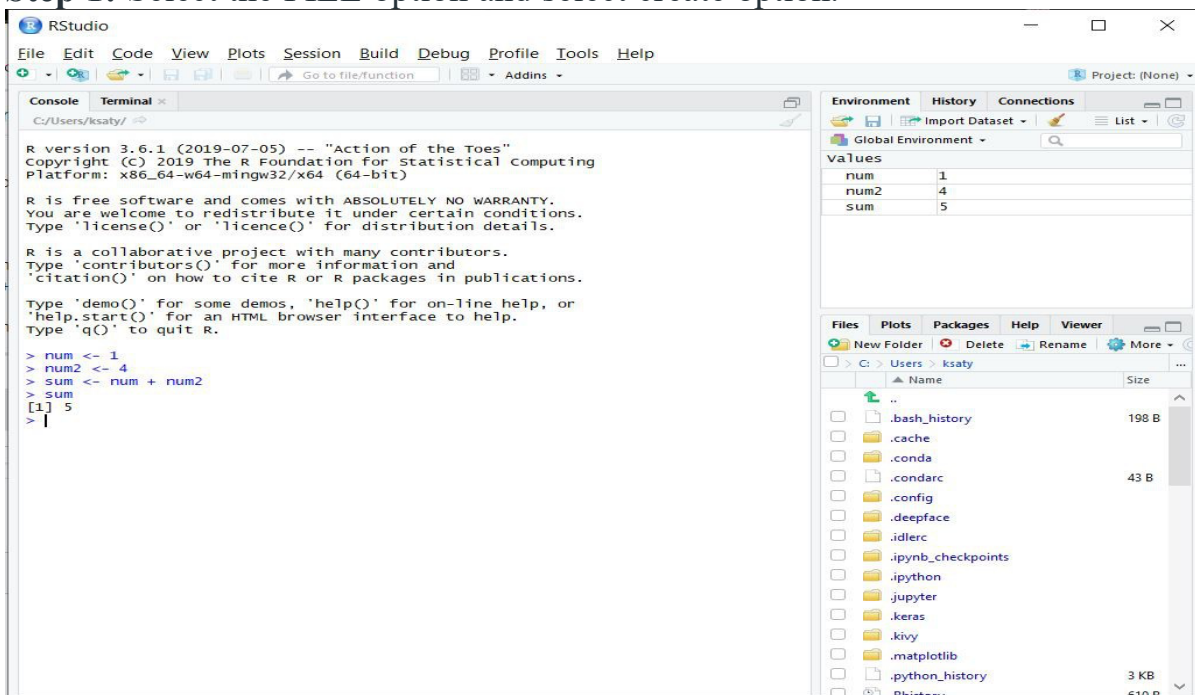
STEP 2: Once directory is chosen, select the more icon and choose “Set as Working Directory”



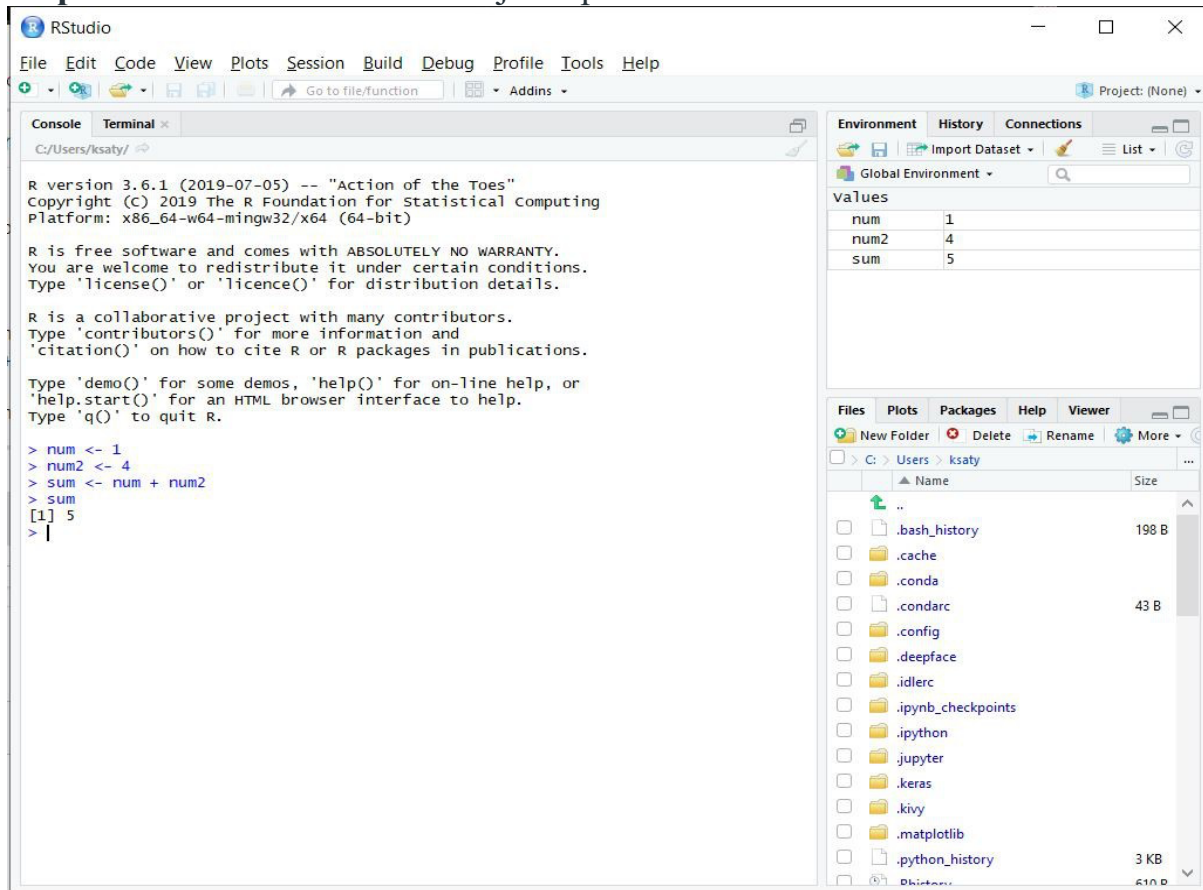
This will select the current directory, which you have chosen using this file browser as your working directory. Once you set the working directory, you are ready to program in R Studio.

Create an RStudio project

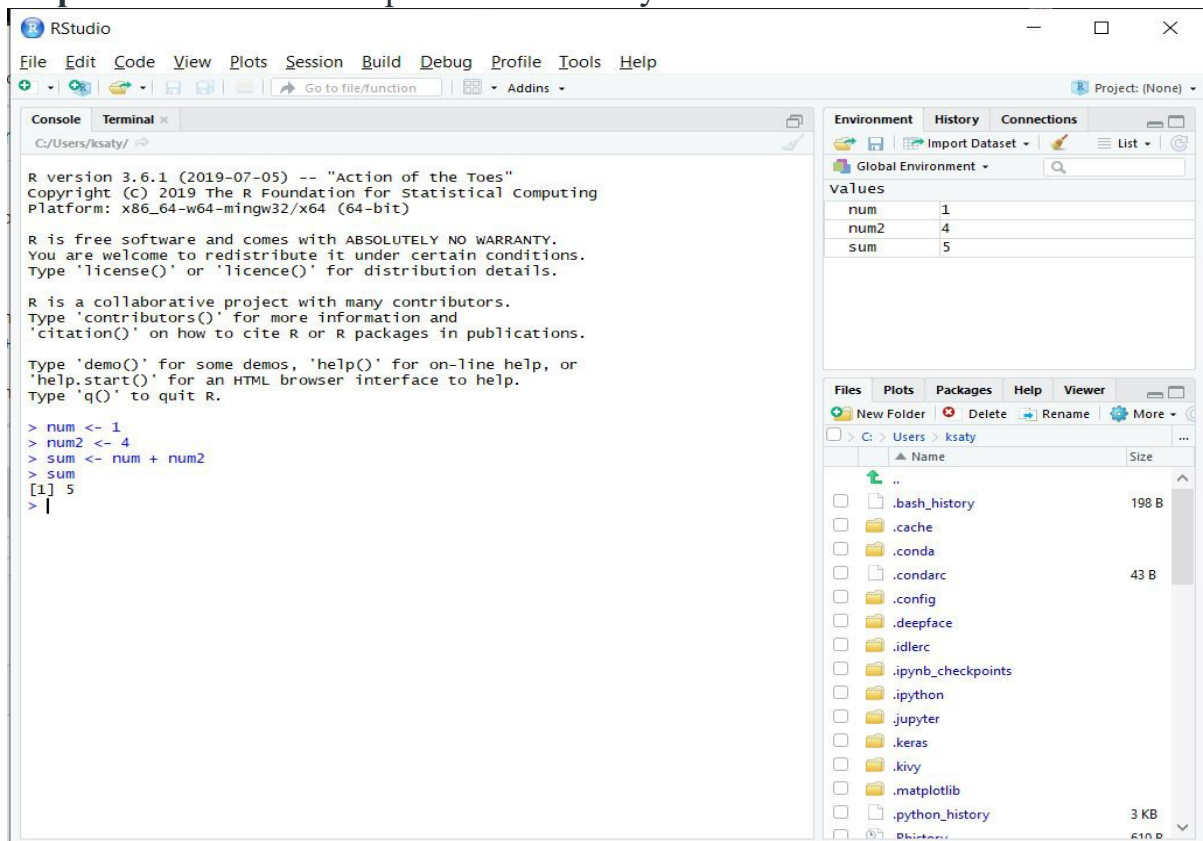
Step 1: Select the FILE option and select create option.



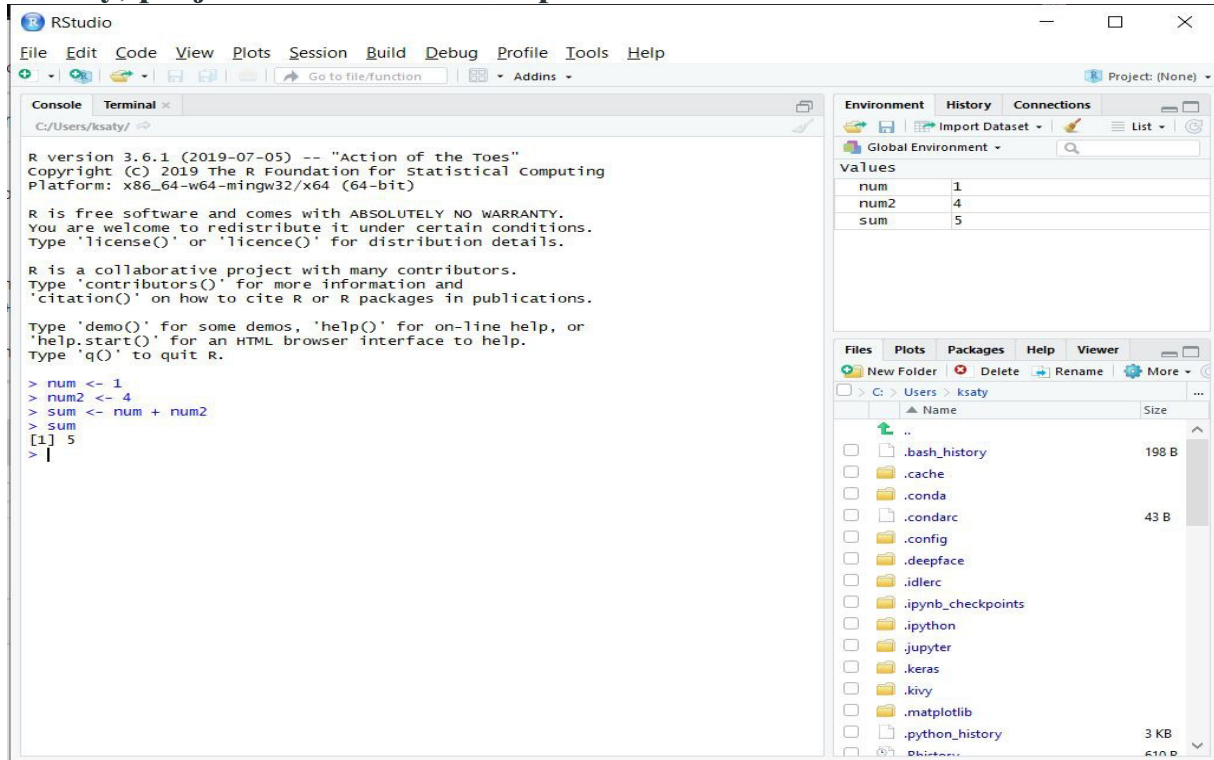
Step 2: Then select the New Project option.



Step 3: Then choose the path and directory name.



Finally, project are created in a specific location:

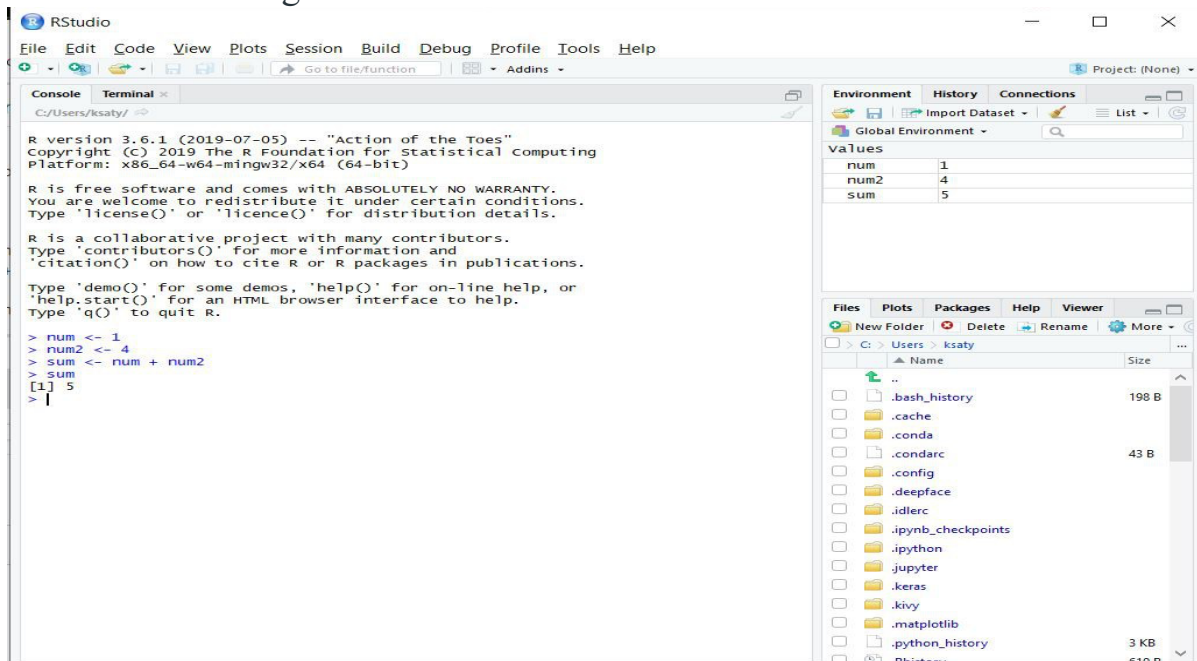


Navigating directories in R studio

- **getwd():** Returns the current working directory.
- **setwd():** Set the working directory.
- **dir():** Return the list of the directory.
- **sessionInfo():** Return the session of the windows.
- **date():** Return the current date.

Creating your first R script

Here we are adding two numbers in R studio.



How to Perform Various Operations in RStudio

We'll see some common tasks, their codes in R Studio

Installing R packages

Syntax:

```
install.packages('package_name')
```

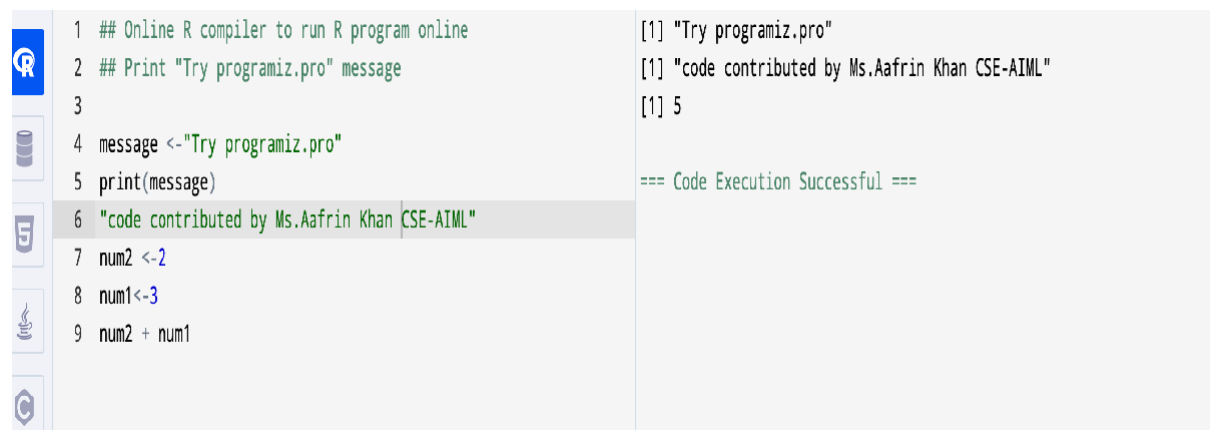
Loading R package

Syntax:

```
library(package_name)
```

Help on an R package

```
help(package_name)
```



The screenshot shows an online R compiler interface. On the left, there is a vertical toolbar with icons for R, a database, a file, a terminal, and a refresh button. The main area is divided into two panes. The left pane contains the following R code:

```
1 ## Online R compiler to run R program online
2 ## Print "Try programiz.pro" message
3
4 message <- "Try programiz.pro"
5 print(message)
6 "code contributed by Ms.Aafrin Khan CSE-AIIML"
7 num2 <- 2
8 num1 <- 3
9 num2 + num1
```

The right pane shows the output of the code execution:

```
[1] "Try programiz.pro"
[1] "code contributed by Ms.Aafrin Khan CSE-AIIML"
[1] 5

=== Code Execution Successful ===
```

Some links on which you can practice

- https://www.w3schools.com/r/r_examples.asp
- <https://www.programiz.com/r/online-compiler/>