

Introduction to R Software

Swayam Prabha

Lecture 5

Introduction to R Studio

Shalabh

Department of Mathematics and Statistics

Indian Institute of Technology Kanpur

Slides can be downloaded from
<http://home.iitk.ac.in/~shalab/sp>



Command Line versus Scripts

Execution of commands in R is not menu driven.
(Not like Clicking over buttons to get outcome)

We need to type the commands.

Single line and multi line commands are possible to write.

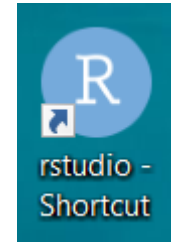
When writing multi-line programs, it is useful to use a text editor rather than execute everything directly at the command line.

Command Line versus Scripts

Option 2:

Use R studio software.

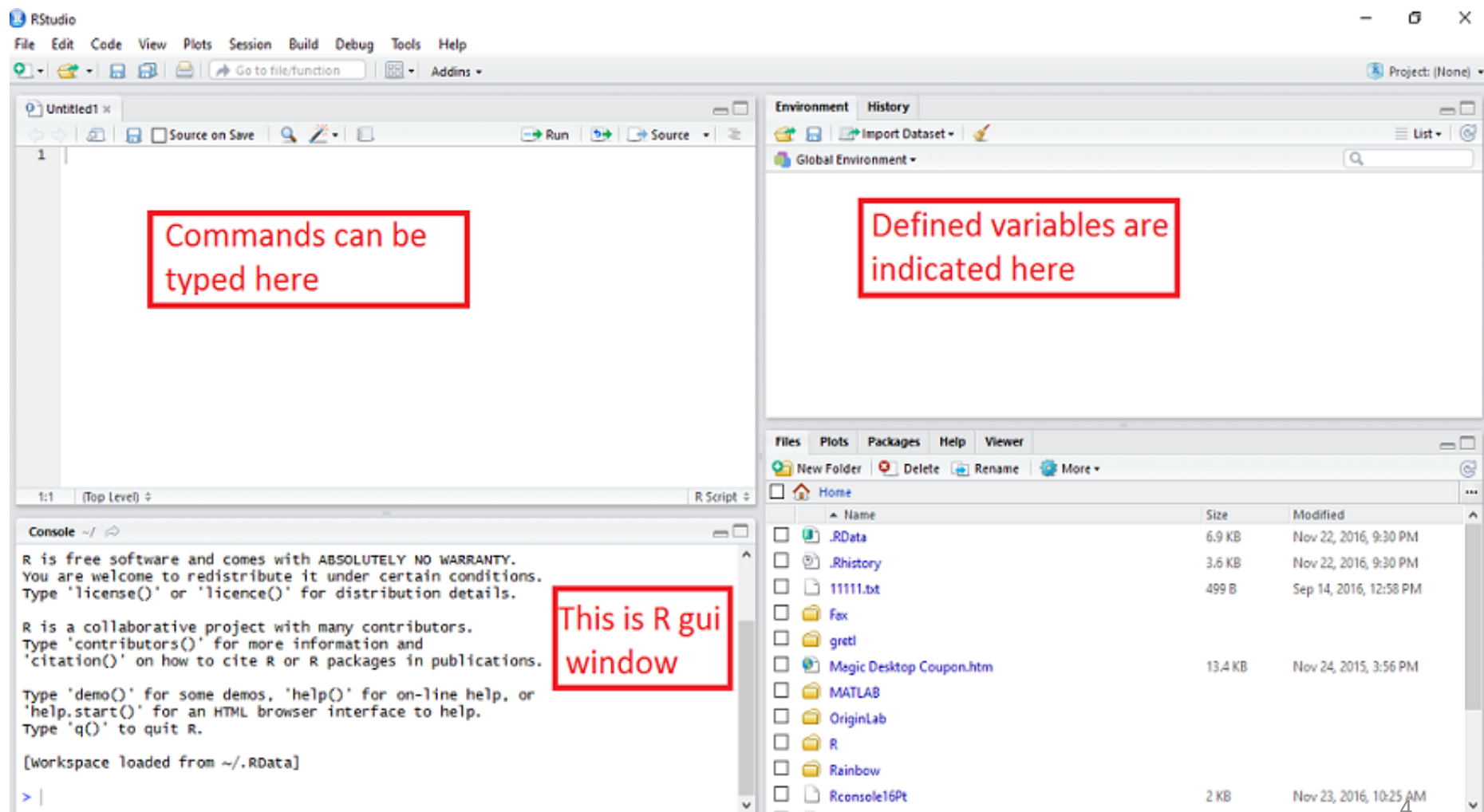
Click at the following icon to begin R Studio



Command Line versus Scripts

Option 2:

Use R studio software.



Command Line versus Scripts

Suppose we want to use following three functions:

Type them.

```
library(MASS)
```

```
attach(bacteria)
```

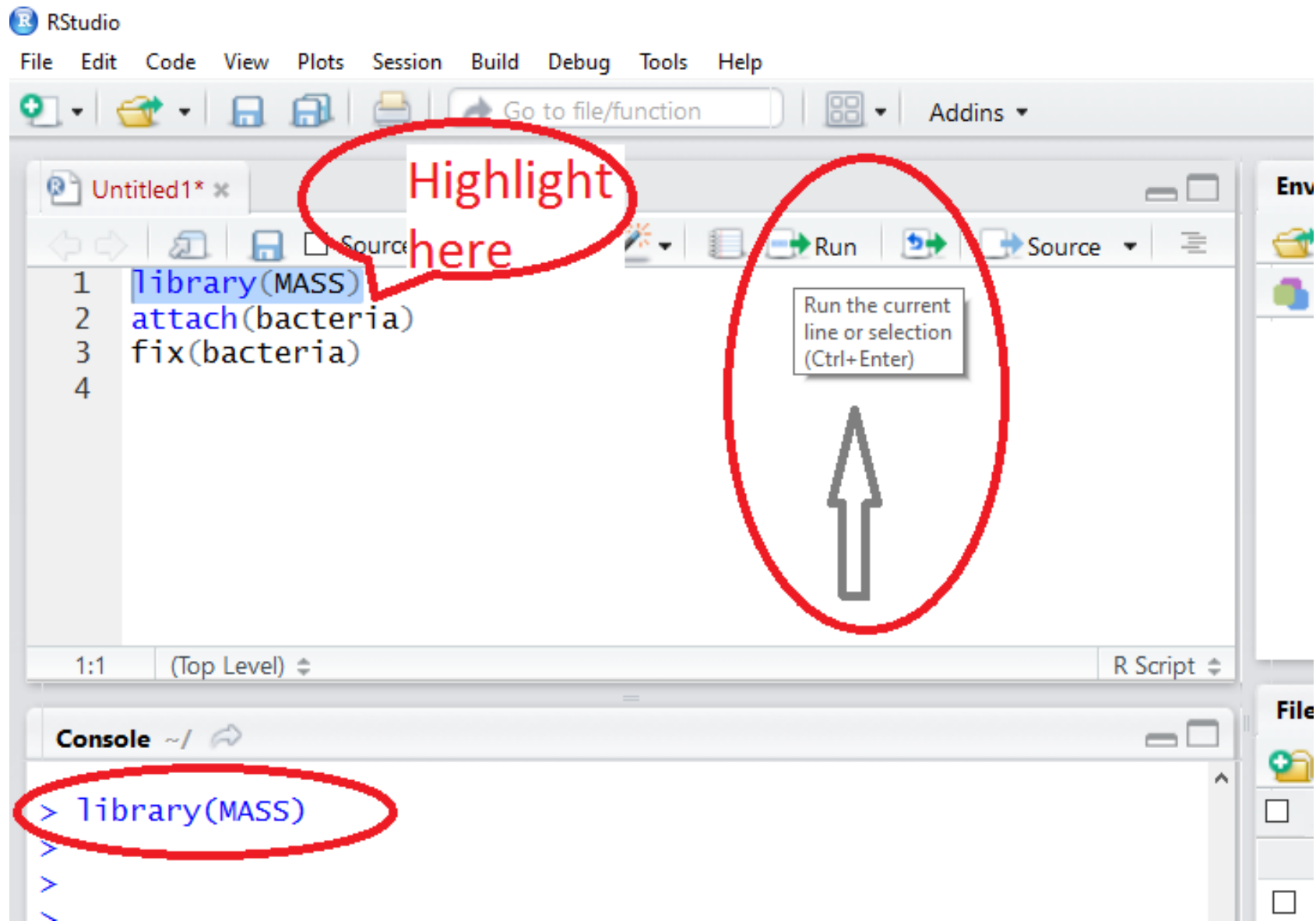
```
fix(bacteria)
```

Suppose we want to run only function: `library(MASS)`

Highlight it and click on `Run`

Then we get....

Command Line versus Scripts



Data Editor

There is a data editor within R that can be accessed from the menu bar by selecting Edit/Data editor.

Provide the name of the matrix or data frame that we want to edit and a **Data Editor** window appears.

Alternatively we can do this from the command line using the **fix** function.

Example:

```
library(MASS)  
attach(bacteria)  
fix(bacteria)
```

Data Editor

We can do it in R Studio as follows :

The screenshot shows the RStudio interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Tools, and Help. The toolbar contains icons for file operations and a search bar. The main editor window shows a script with the following code:

```
1 library(MASS)
2 attach(bacteria)
3 fix(bacteria)
```

A red circle highlights the first line of code with the text "Highlight here". The Run button in the toolbar is highlighted with a red arrow and the text "Click here".

The Data Editor window is open, displaying a table with the following data:

	y	ap	hilo	week	ID	trt
1	y	p	hi	0	X01	placebo
2	y	p	hi	2	X01	placebo
3	y	p	hi	4	X01	placebo
4	y	p	hi	11	X01	placebo
5	y	a	hi	0	X02	drug+
6	y	a	hi	2	X02	drug+
7	n	a	hi	6	X02	drug+
8	y	a	hi	11	X02	drug+
9	y	a	lo	0	X03	drug
10	y	a	lo	2	X03	drug
11	y	a	lo	4	X03	drug
12	y	a	lo	6	X03	drug

A red arrow points to the Data Editor window with the text "We get this window". A grey arrow points to the table content with the text "This is the data in MASS".

Cleaning up the Windows

We assign names to variables when analyzing any data.

It is good practice to remove the variable names given to any data frame at the end each session in R.

This way, variables with same names but different properties will not get in each others way in subsequent work.

`rm()` command removes variable names

For example,

`rm(x, y, z)` removes the variables x, y and z.

Cleaning up the Windows

`detach()` command detaches objects from the Search Path

It removes it from the `search()` path of available R objects.

Usually this is either a `data.frame` which has been attached or a package which was attached by `library`.

To get rid of everything, including data frames, type

```
rm(list=ls())
```

Then we get....

Cleaning up the Windows

R Console

```
> library(splines)  
>  
>  
> detach(package:splines)  
>  
>
```

Loads the package 'splines'

Detaches the package 'splines'

The image shows a screenshot of an R console window with a light blue header. The console contains several lines of red text representing R commands. A green arrow points from the first command, `library(splines)`, to a callout box on the right that says "Loads the package 'splines'". Another green arrow points from the fourth command, `detach(package:splines)`, to a callout box below it that says "Detaches the package 'splines'". There are also several empty prompt lines (`>`) in the console.

Introduction to R Studio

It is an interface between R and us.

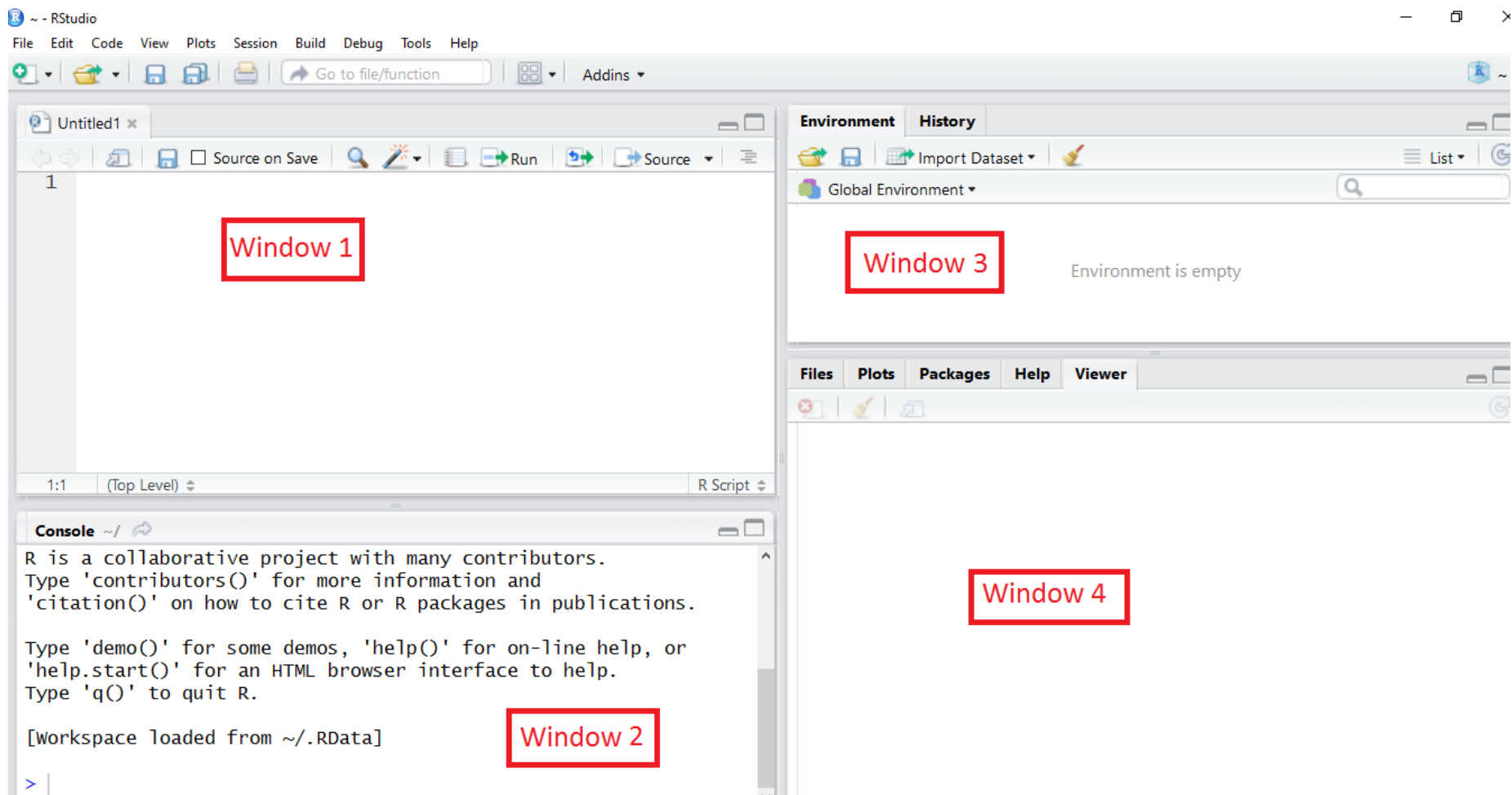
More useful for beginners.

It makes coding easier.

When we start R studio, we see 4 windows

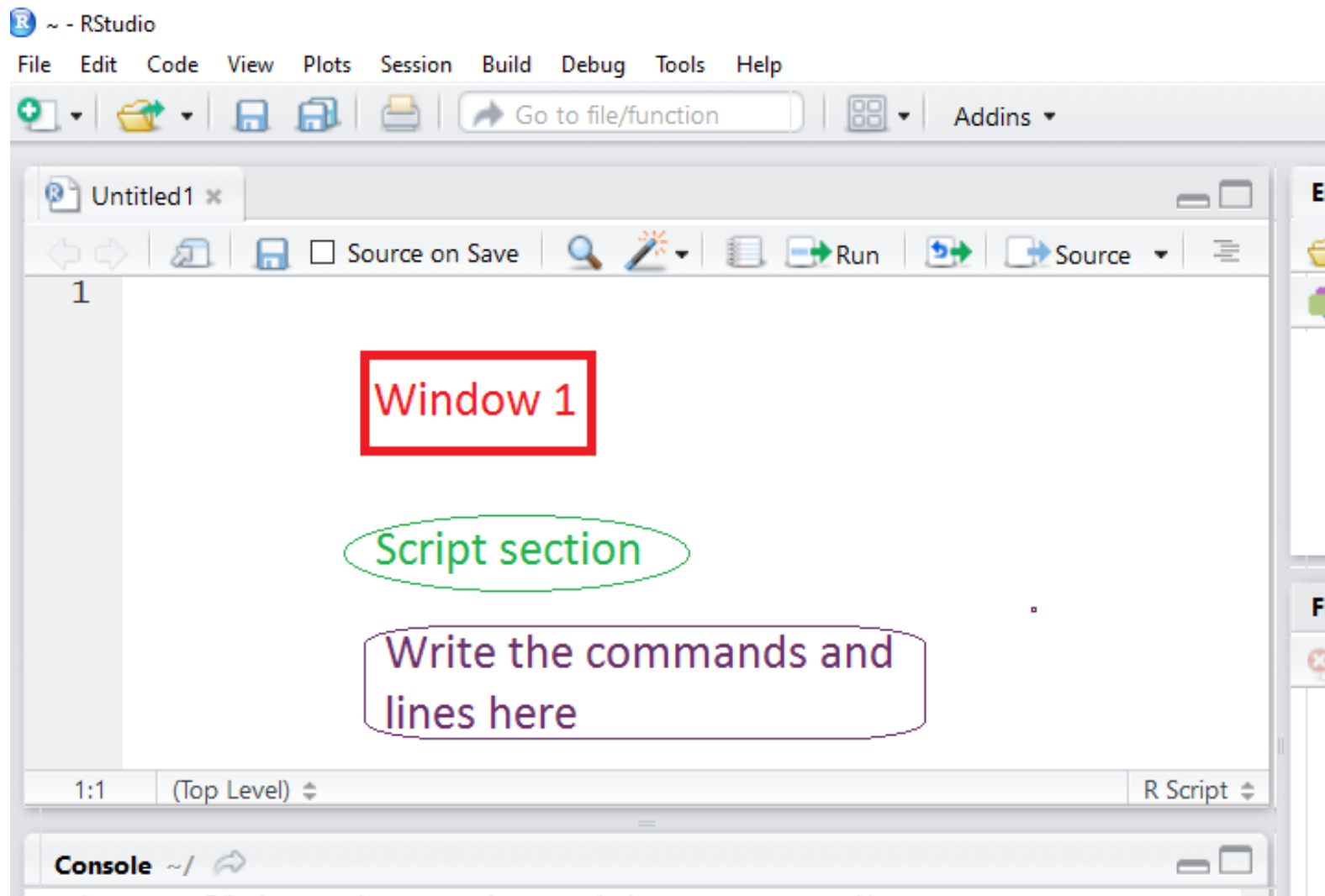
Introduction to R Studio

First opening window of Rstudio is as follows having four windows.



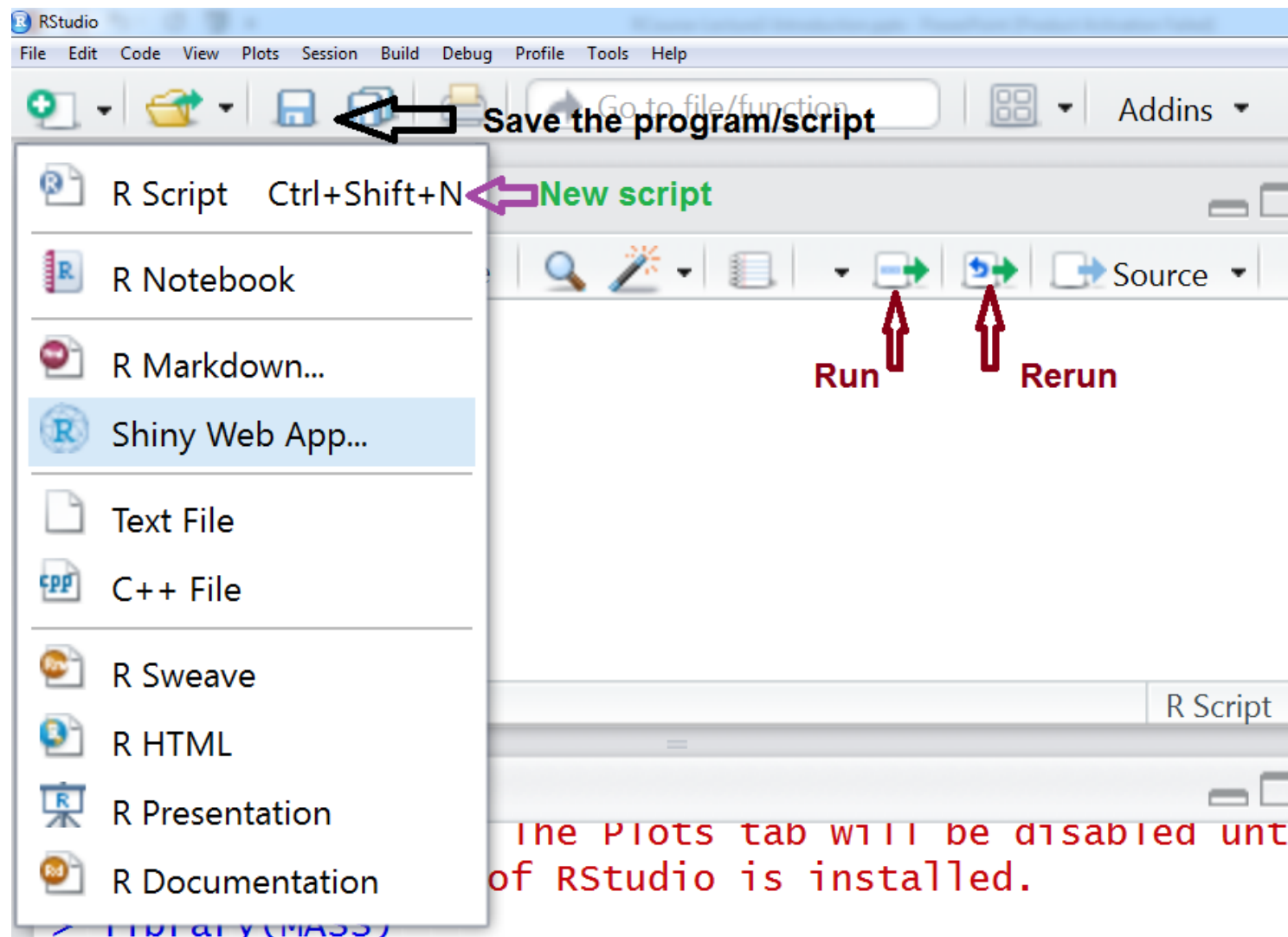
Introduction to R Studio

Description of Window 1



Introduction to R Studio

Description of Window 1



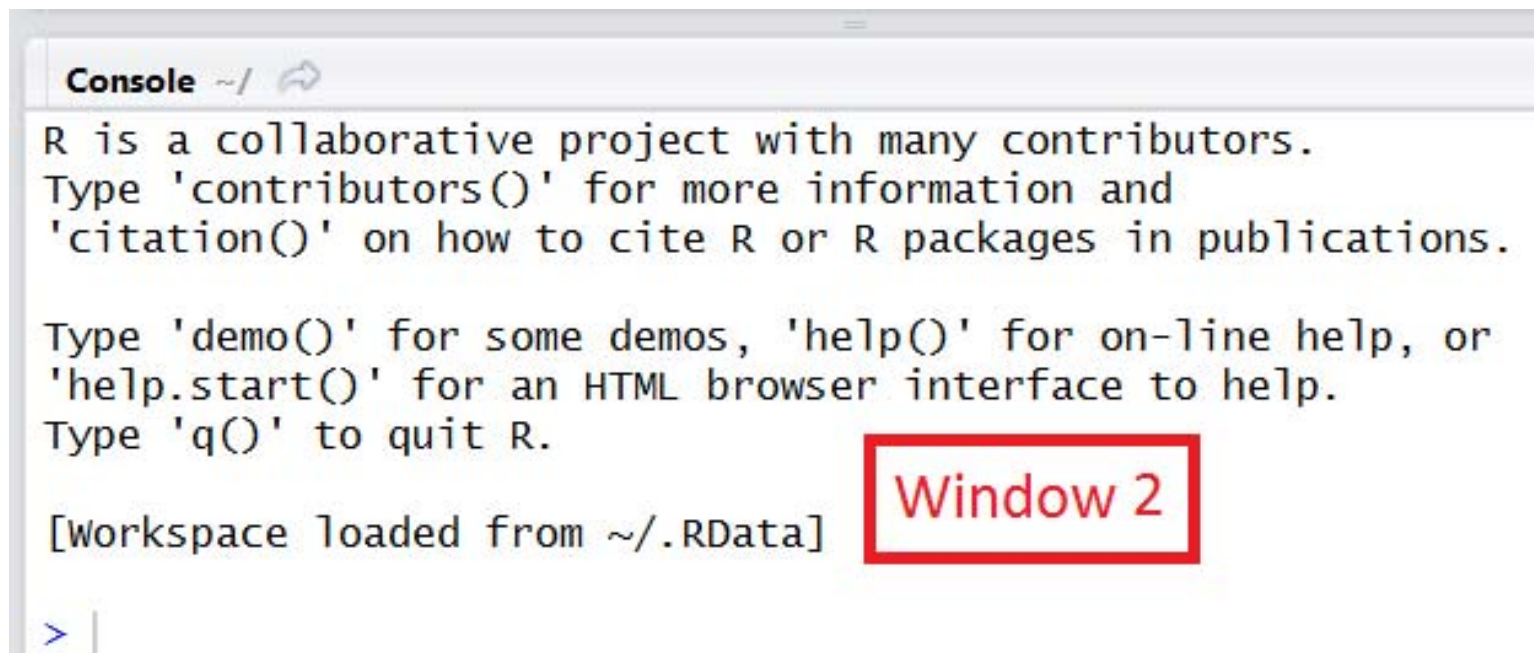
Introduction to R Studio

Description of Window 2 : Console

R program window appears here.

Calculations take place in console window.

One can write programmes in console also but it is hard to make corrections and experiments with the coding.



```
Console ~/ ↻  
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.  
  
Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.  
  
[Workspace loaded from ~/.RData]  
> |
```


Introduction to R Studio

Description of Window 3 : Environment window

All the variables and objects used in the programme appear here. The nature and values of variables and objects also appear here.

The screenshot shows the R Studio Environment window. At the top, there are two tabs: 'Environment' and 'History'. Below the tabs is a toolbar with icons for file operations and a search icon. The main area is titled 'Global Environment' and contains a table of variables. The table has two columns: 'Name' and 'Value'. The first row shows the variable 'x' with the value '1'. There are four callout boxes: 1. A blue box pointing to the 'History' tab with the text 'History tells about the codes used earlier.' 2. A blue box pointing to the 'Import Dataset' button with the text 'Data can be imported from other files by clicking here'. 3. A blue box pointing to the 'x' variable with the text 'Stored value can be erased from here'. 4. A green box pointing to the '1' value with the text 'The stored value x = 1 appears here'. A red box at the bottom left contains the text 'Window 3'.

Environment History

Import Dataset

Global Environment

Name	Value
x	1

History tells about the codes used earlier.

Data can be imported from other files by clicking here

Stored value can be erased from here

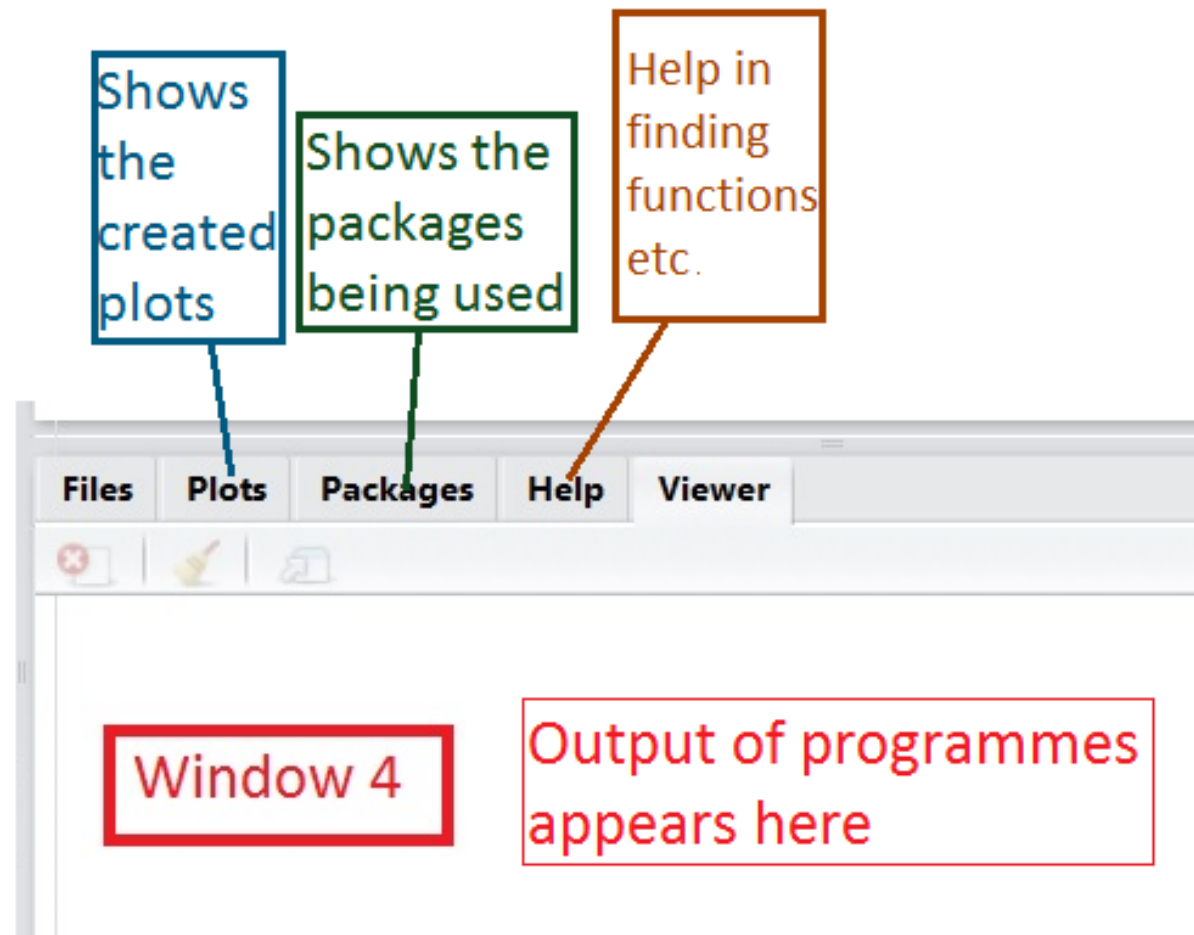
The stored value x = 1 appears here

Window 3

Introduction to R Studio

Description of Window 4 : Output window

The output of programmes appears in this window.



Introduction to R Studio

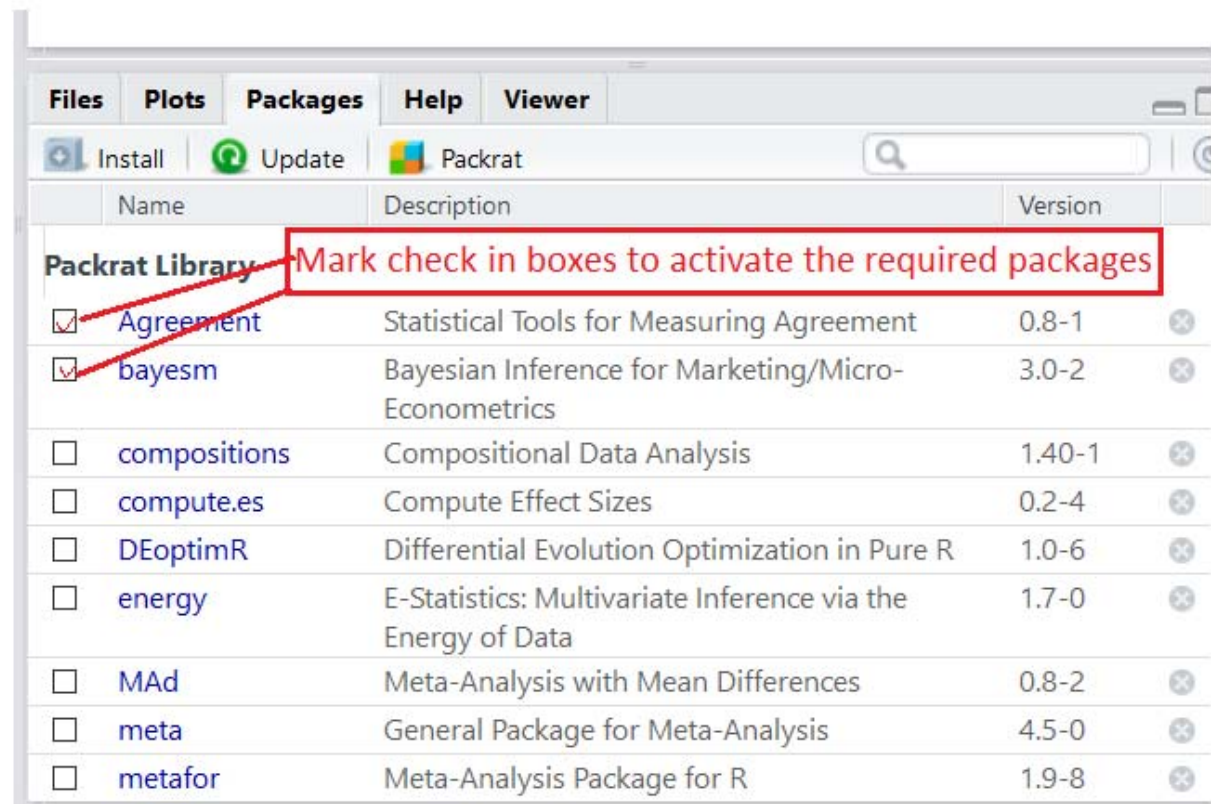
Description of Window 4 : Output window

Packages:

All the packages being installed appear here.

Packages are not active.

Check mark in the boxes to activate them.



Introduction to R Studio

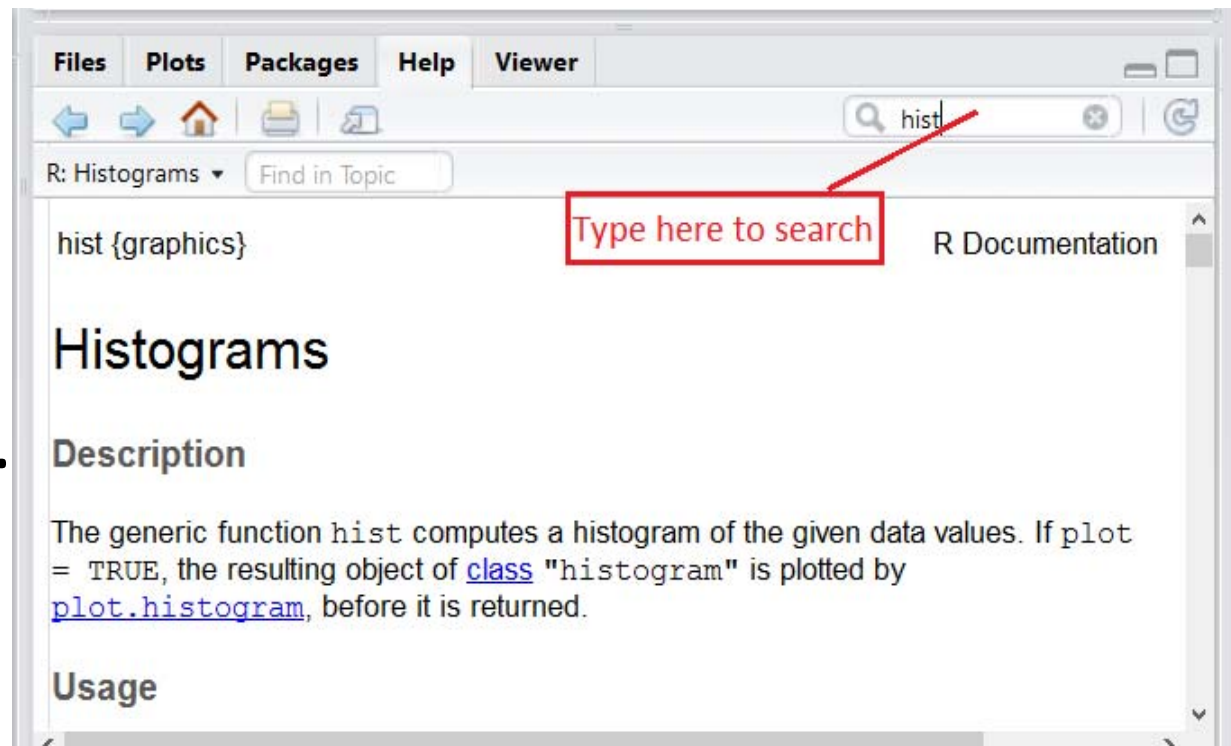
Window 4 : Output window

Help:

Various types of help can be asked.

E.g., to know about histogram,
type **hist**.

Information appears.



Introduction to R Studio

Example:

Histogram of values 1,2,1,1,2,3,1,2,3,1,2,2,3

R studio has following operation and output:

```
hist(c(1,2,1,1,2,3,1,2,3,1,2,2,3))
```

Introduction to R Studio

The screenshot displays the R Studio environment with the following components and annotations:

- Source Editor:** Contains the R code:

```
1 x=c(1,2,1,1,2,3,1,2,3,1,2,2,3)
2 hist(x)
3
```

 - Annotation: "Step 1: Type commands here" points to the code lines.
 - Annotation: "Step 2 :Click on 'Run'" points to the Run button in the toolbar.
- Environment Panel:** Shows the variable `x` as a numeric vector of length 13: `num [1:13] 1 2 1 1 2 3 1 2 3 1 2 2 3`.
 - Annotation: "The nature and values of x appears here" points to the variable `x`.
- Console:** Shows the command `hist(x)` being executed.
 - Annotation: "Step 3: The executed command appear here" points to the console output.
- Plots Panel:** Displays a histogram titled "Histogram of x".
 - Annotation: "Output appears here" points to the histogram plot.
 - Annotation: "Delete graphs" points to the delete icon in the toolbar.
 - Annotation: "Export" points to the export menu, which includes options: "Save as Image...", "Save as PDF...", and "Copy to Clipboard".