Contents

How To Use This Book

The creation of this book has been for me, Mike McGrath, an exciting opportunity to update my previous R programming book to discover how the R programming language can be used today for data analysis and the production of beautiful data visualizations. Example code listed in this book describes how to produce R Scripts in easy steps – and the screenshots illustrate the actual results. I sincerely hope you enjoy discovering the exciting possibilities of R programming and have as much fun with it as I did in writing this book.

In order to clarify the code listed in the steps given in each example, I have adopted certain colorization conventions. Components and keywords of the R programming language are colored blue, programmer-specified names and variable names are colored red, literal numeric values and literal character string values are colored black, and comments are colored green:

```
# Write the traditional greeting.
greeting = "Hello World!"
print( greeting )
```

In order to readily identify each source code file described in the steps, a file icon and file name appears in the margin alongside the steps:

Script.R   Data.csv

For convenience I have placed source code files from the examples featured in this book into a single ZIP archive. You can obtain the complete archive by following these easy steps:

1. Browse to **www.ineasysteps.com** then navigate to **Free Resources** and choose the **Source code downloads and other book resources** section

2. Find **R for Data Analysis in easy steps, 2nd edition** in the list, then click on the hyperlink entitled **All Code Examples** to download the archive

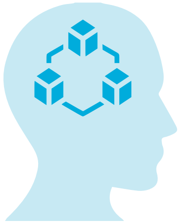
3. Next, extract the **MyRScripts** folder to a convenient location on your system

4. Now, follow the steps to call upon the R program interpreter and see the output

> If you don't achieve the result illustrated in any example, simply compare your code to that in the original example files you have downloaded to discover where you went wrong.

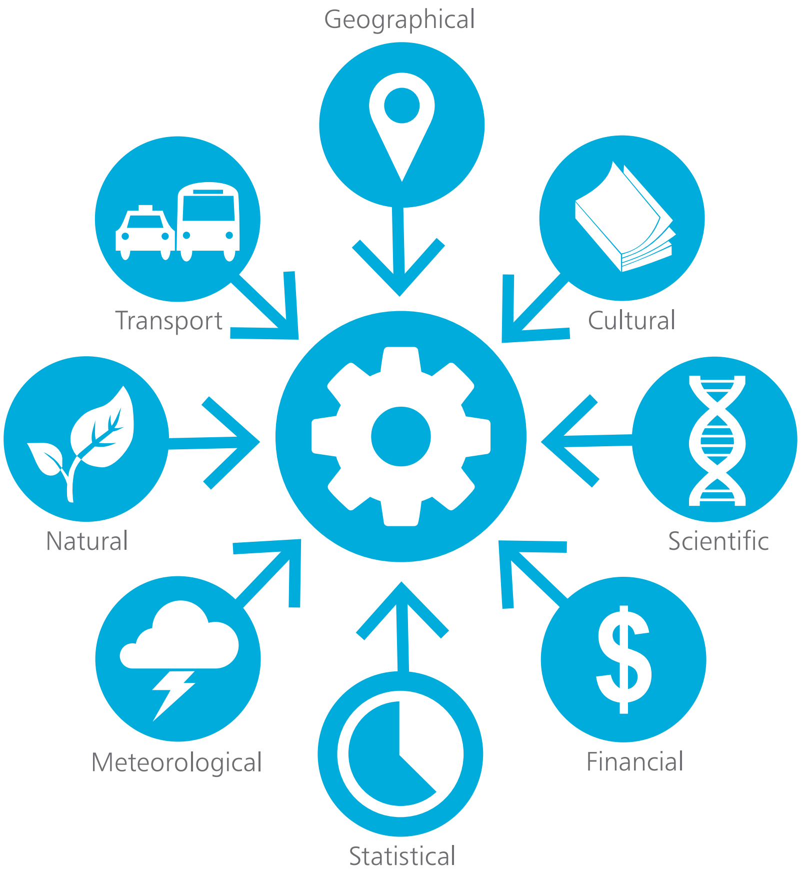
1 Getting started

Welcome to the exciting world of R programming. This chapter describes how to set up an R environment and demonstrates how to create a first R program.



Understanding data

The term "data" refers to items of information that describe a (qualitative) status or a (quantitative) measure of magnitude. Various types of data are collected from a huge range of sources and reported for analysis to reveal pattern and trend insights:



Don't forget

This illustration depicts only some of the many data types that can be reported for analysis.

Hot tip

At the time of writing, there are around 13 billion devices connected to the internet – and the number increases each day.

Data is increasingly being collected by devices that are able to report measurements for analysis via the internet ("The Cloud"). For example, devices that have temperature and humidity sensors can report measurements for instant analysis of climate conditions. The recent rapid decline in the cost of device sensors has given rise to the "Internet of Things" (IoT) that can easily and cheaply report vast amounts of data – this is often referred to as "big data". Big data consists of extremely large data sets that can best be analyzed by computer to reveal pattern and trend insights.

Data analysis (a.k.a. "data analytics") is the practice of converting collected data into information that is useful for decision-making. The collected "raw" data will, however, typically undergo two initial procedures before it can be explored for insights:

- **Data processing** – the raw data must be organized into a structured format. For example, it may be arranged into rows and columns in a table format for use in a spreadsheet.
- **Data cleaning** – the organized data must be stripped of incomplete, duplicated, and erroneous items – for, example, by the removal of duplicated rows in a spreadsheet.

After the data has been processed and cleaned it can be explored to discover its main characteristics. This may require further data cleaning to refine the data to specific areas of interest, or may require additional data to better understand its messages. Descriptive statistics, such as average values, might be calculated to understand the data. Algorithms might be used to identify associations within the data. Data visualization might also be used to produce a graphical representation of the data for examination.

After the data has been analyzed, the results can be communicated using data visualization to present tables, plots, or charts that clearly and efficiently convey the key messages within the data. Tables provide information in which the user can look up a specific number, whereas plots and charts provide information in a way that encourages the eye to make comparisons.

"R" is an interpreted programming language and software environment that is widely used for data analysis and visualization. The "RStudio" Integrated Development Environment (IDE) is often used with R, as RStudio provides a code editor, debugging features, and visualization tools that make R easier to use. The popularity of R has grown rapidly in recent years as the increase in big data has made data analysis more important than ever.

The R programming software and RStudio IDE are both available for Windows, Linux, and macOS operating systems, and both are used throughout this book to demonstrate R for data analysis.

Hot tip

"Data Science" is the study of how data can be turned into a valuable resource.

Hot tip

"Data Mining" is the process of searching large data sets to identify patterns.

Hot tip

"Data Product" is digital information that can be purchased.

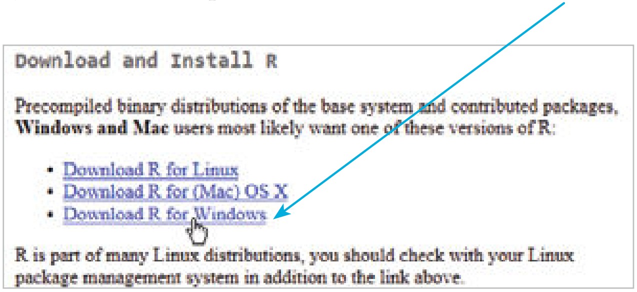
Installing R

The R programming language and software environment is freely available open source software that you can install onto your computer from the Comprehensive R Archive Network (CRAN):

If you are having difficulty downloading R, click the CRAN **Mirrors** link at **cran.r-project.org** then choose a server near to your location.

1. Open a web browser and visit **cran.r-project.org**

2. Select the link appropriate for your computer operating system. For example, click **Download R for Windows**



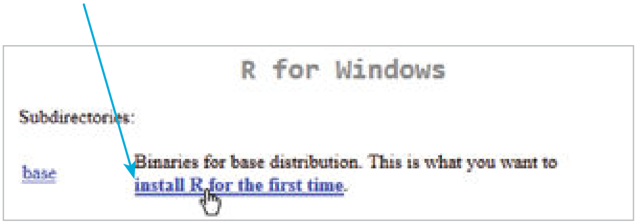
Download and Install R

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
- Download R for (Mac) OS X
- 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.

3. Next, select the link for the **base** R distribution

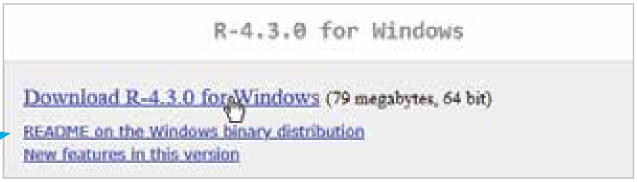


R for Windows

Subdirectories:

base Binaries for base distribution. This is what you want to install R for the first time.

4. Now, select the link to **download** the R installer



R-4.3.0 for Windows

Download R-4.3.0 for Windows (79 megabytes, 64 bit)
README on the Windows binary distribution
New features in this version

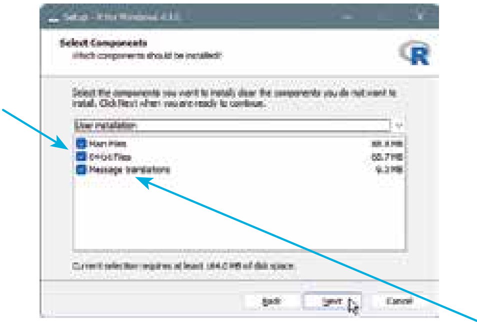
You can click this link for more help with installation.

5. When the download has completed, run the installer then accept "Run anyway" and "Yes" – then click the **Next** button to open the **R Setup Wizard**

6. Read the **License** information, then click the **Next** button to continue

7 Accept the suggested installation location, then click the **Next** button to continue

8 Choose to install for **User installation** and check all the options, then click the **Next** button to continue



Select Components
Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

User installation

☑ Main Files
☑ 64-bit Files
☑ Message translations

Current selection requires at least [illegible] MB of disk space.

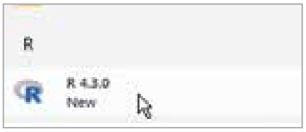
Install **Message translations** for error messages, warning messages, and menu labels in languages other than English.

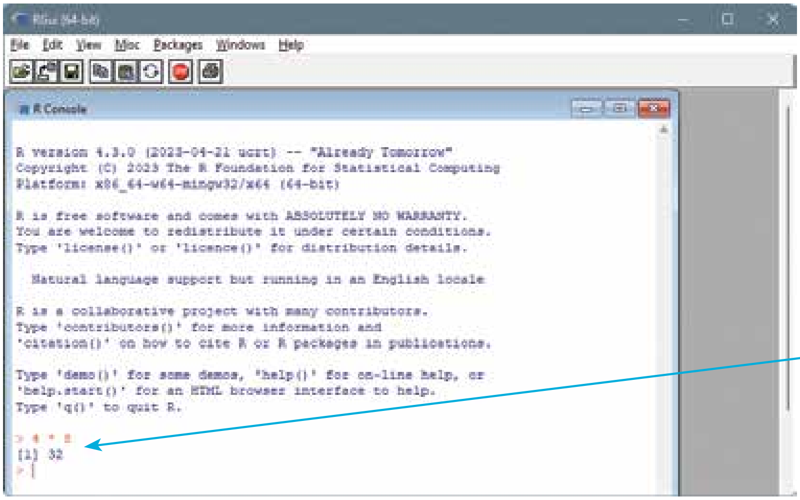
9 Choose **No (accept defaults)** to not customize startup options, then click the **Next** button to continue

10 Enter a name for a Start menu folder (such as “R”), then click the **Next** button to continue

11 Choose additional tasks (such as **Create a desktop shortcut**), then click **Next** to begin the installation

12 When installation has completed, launch the **R** environment from the Start menu folder you named (“R”)





```
R version 4.3.0 (2023-04-21 ucrt) -- "Already Tomorrow"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

  Natural language support but running in an English locale

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.

> 4 * 8
[1] 32
> |
```

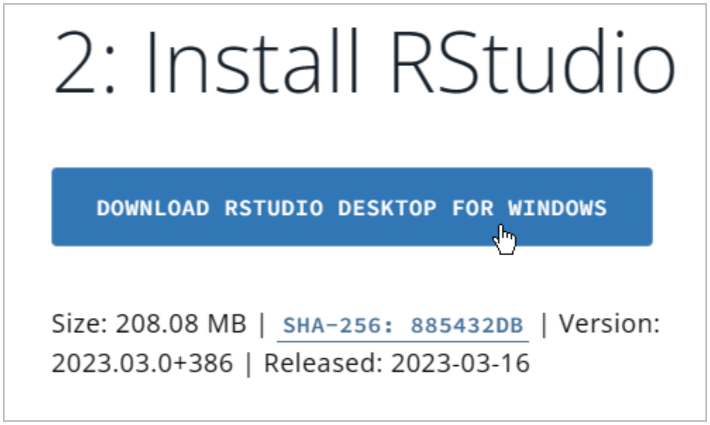
You can type expressions in the R Console to see their result – but the RStudio IDE is a much more effective programming environment.

Installing RStudio

The RStudio IDE has a freely available open source edition that you can install onto your computer from the RStudio website:

1. Open a web browser and visit the RStudio downloads page at **rstudio.com/products/rstudio/download**

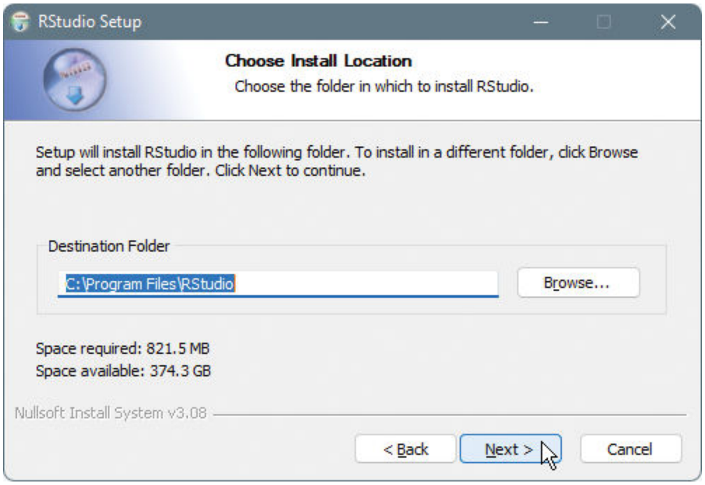
2. Scroll down the page and select the **Installer** download link appropriate for your computer operating system. For example, click the edition for **Windows**



2: Install RStudio

Size: 208.08 MB | SHA-256: 885432DB | Version: 2023.03.0+386 | Released: 2023-03-16

3. When the download has completed, run the installer then accept "Run anyway" and "Yes" – then click the **Next** button to open the **RStudio Setup Wizard**



RStudio Setup

Choose Install Location

Choose the folder in which to install RStudio.

Setup will install RStudio in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder

C:\Program Files\RStudio

Space required: 821.5 MB
Space available: 374.3 GB

Nullsoft Install System v3.08

The RStudio software is available in Desktop and Server versions with Open Source Licenses and Commercial Licenses for each version – be sure to download the Desktop version with the Open Source License to try the examples in this book for free.

Hot tip

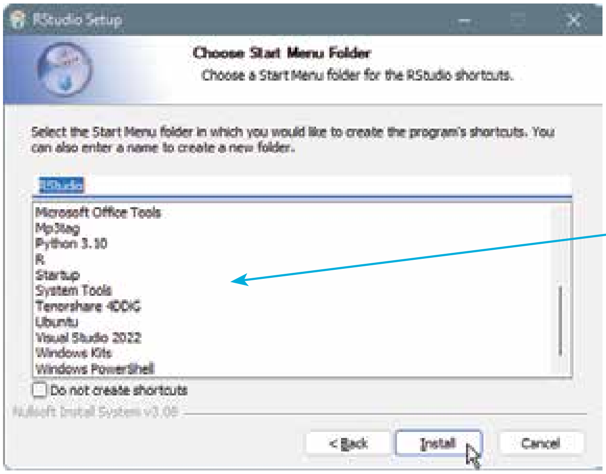
Item 1 on the RStudio download page provides an alternative download for the R installation.

Beware

You must have R installed before you install RStudio. See pages 10-11 for the R software installation procedure.

4 Accept the suggested installation location and click the **Next** button to continue

5 Accept the suggested Start menu folder name "RStudio" and click the **Install** button to continue



RStudio Setup

Choose Start Menu Folder

Choose a Start Menu folder for the RStudio shortcuts.

Select the Start Menu folder in which you would like to create the program's shortcuts. You can also enter a name to create a new folder.

RStudio

Microsoft Office Tools
Mp3tag
Python 3.10
R
Startup
System Tools
Tenorshare 4DDiG
Ubuntu
Visual Studio 2022
Windows Kits
Windows PowerShell

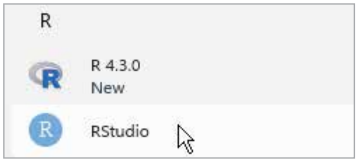
□ Do not create shortcuts

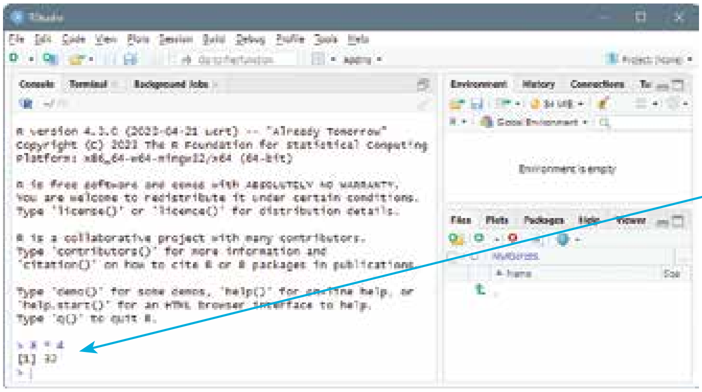
Nullsoft Install System v3.08

The items listed in this dialog box are the names of your existing Start menu folders and will vary according to what you have installed on your computer.

6 When the installation has completed, click the **Finish** button to close the Setup Wizard

7 Launch **RStudio** from the Start menu and choose to **Use your machine's default 64-bit version of R**, then click **OK** to open the RStudio IDE





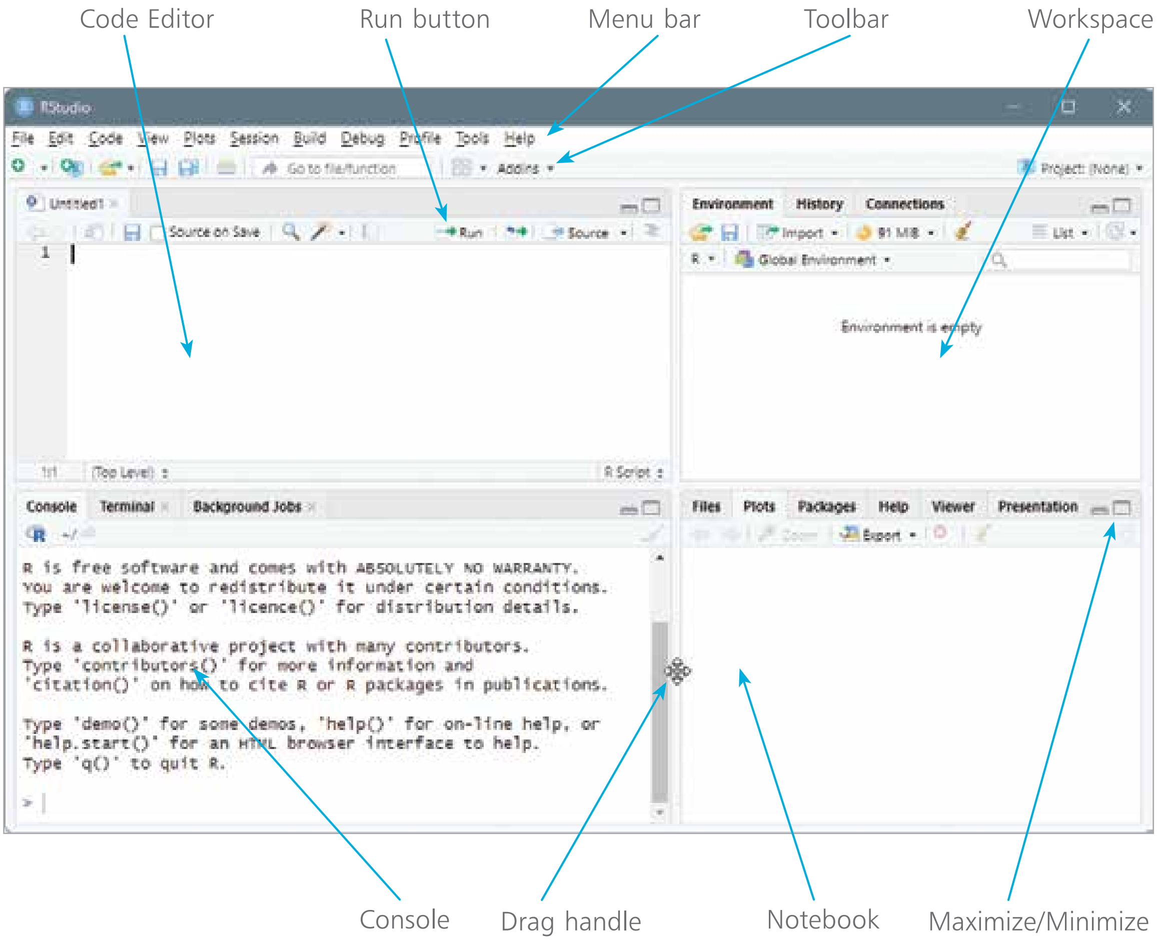
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Console Terminal Background Jobs
Environment History Connections
Global Environment
Environment is empty
Files Plots Packages Help Viewer
MyDocs
Name
Size

```
R version 4.3.0 (2023-04-21 ucrt) -- "Already Tomorrow"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

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.

> 8 * 4
[1] 32
> 
```

You can type expressions in the RStudio Console to see their result, just as you can in the R Console – but the RStudio IDE can do so much more.

Exploring RStudio

The RStudio interface consists of a menu bar and toolbar positioned at the top of the window, and four main panes whose position can be adjusted to suit your preference. When you launch RStudio, only three panes may be visible until you select **File**, **New File**, **RScript** on the menu bar to open the "Code Editor" pane. The default layout positions the four panes as shown below:



```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

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.

> 
```

When the mouse pointer is placed on the border between any two panes, the pointer changes to a four-pointed "drag handle". This allows you to drag the vertical border to adjust the width of the left and right panes, and to drag the horizontal border to adjust the height of the top and bottom panes. The size of each pane can also be adjusted by clicking the Maximize and Minimize buttons.

Each RStudio pane can contain multiple tabs, and it is useful to initially explore each RStudio pane to understand its purpose:

Code Editor pane

The Code Editor is where you type or edit R Script code, and you see it automatically colored to highlight syntax – click this pane's **Run** button to see the script output appear in the Console pane.

Console pane

- **Console tab** – This is where you can directly enter commands for immediate execution by the R interpreter.
- **Terminal tab** – This is where you can directly enter commands for execution by the operating system shell.

Workspace pane

- **Environment tab** – This is where you will see available objects such as variables and datasets.
- **History tab** – This is a list of your past commands executed by the R interpreter in the Console pane.
- **Connections tab** – This tab enables you to connect to databases to explore the objects and data inside the connection.

Notebook pane

- **Files tab** – This is a file browser, which by default lists all the files in your working directory.
- **Plots tab** – This exciting tab is where your plots, graphs, and charts will appear as output from an R Script.
- **Packages tab** – This tab lists available packages that you can install to extend RStudio's functionality.
- **Help tab** – This is where you can seek assistance on the R language and RStudio IDE.
- **Viewer tab** – This is where you can see local HTML content that has been written to the session temporary directory.

Hot tip

R Script code can be saved as a file for later use, and multiple R Script files can be open on separate tabs in the Code Editor pane.

Hot tip

You can click on a data set listed in the Environment tab to open a spreadsheet of that data in the Code Editor pane.

Hot tip

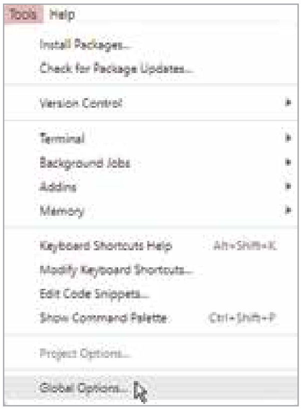
You can click on an R Script file in the Files tab to open that file in the Code Editor pane.

Setting preferences

RStudio is highly customizable, and it is worth setting up its features to better enjoy your R programming environment:

1. Create a new directory on your computer in which to save the R Scripts you will write. For example, on Windows you might create a directory of **C:\MyRScripts**

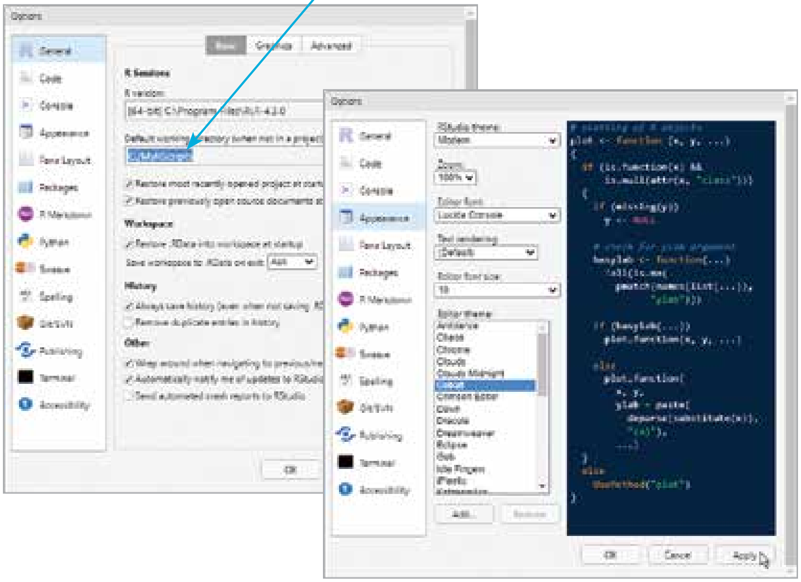
2. Launch RStudio then select **Tools**, **Global Options...** on the menu bar – to open the "Options" dialog



3. Select **General** in the left panel of the "Options" dialog, then enter the path to the directory you created into the **Default working directory** box

Don't forget

Your home directory is set as the default working directory until you specify an alternative.


Options
General
Code
Console
Appearance
Pane Layout
Packages
R Markdown
Python
Sweave
Spelling
Git/SVN
Publishing
Terminal
Accessibility
Basic
Graphics
Advanced
R Sessions
R version:
[64-bit] C:\Program Files\R\R-4.2.0
Default working directory (when not in a project):
C:\MyRScripts
Restore most recently opened project at startup
Restore previously open source documents at startup
Workspace
Restore .RData into workspace at startup
Save workspace to .RData on exit: Ask
History
Always save history (even when not saving .RData)
Remove duplicate entries in history
Other
Wrap around when navigating to previous/next tab
Automatically notify me of updates to RStudio
Send automated crash reports to RStudio
OK
Options
RStudio theme:
Modern
Zoom:
100%
Editor font:
Lucida Console
Text rendering:
Default
Editor font size:
10
Editor theme:
Ambiance
Chaos
Chrome
Clouds
Clouds Midnight
Cobalt
Crimson Editor
Dawn
Dracula
Dreamweaver
Eclipse
Gob
Idle Fingers
iPlastic
Katzenmilch
Add...
Remove...
OK
Cancel
Apply

```
# plotting of R objects
plot <- function (x, y, ...)
{
  if (is.function(x) &&
    is.null(attr(x, "class")))
  {
    if (missing(y))
      y <- NULL

    # check for ylab argument
    hasylab <- function(...)
      !all(is.na(
        pmatch(names(list(...)),
              "ylab")))

    if (hasylab(...))
      plot.function(x, y, ...)

    else
      plot.function(
        x, y,
        ylab = paste(
          deparse(substitute(x)),
          "(x)"),
        ...)
  }
  else
    UseMethod("plot")
}
```

Hot tip

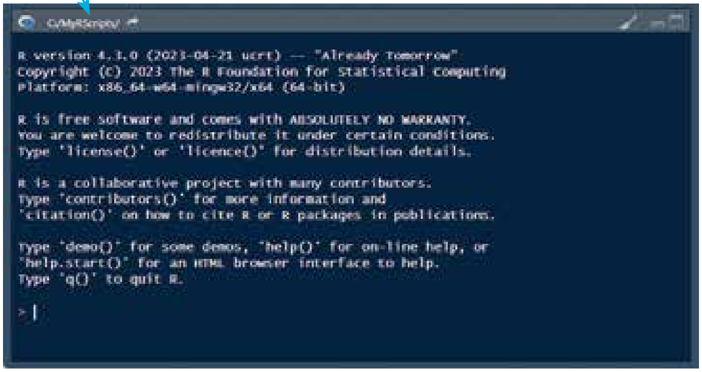
Themes with dark backgrounds (such as the "Cobalt" theme shown here) are often considered to be more restful on your eyes than those with white backgrounds.

4. Next, select **Appearance** in the left panel, then click items in the **Editor theme** box to preview possible color themes

5. Use the **Editor font** and **Editor font size** drop-down menus to choose your font preferences

6. Click the **Apply** button to change the RStudio settings

7. Click the **OK** button to close the "Options" dialog, then restart RStudio to see your preferences have been applied – the working directory appears on the Console title bar



C:/MyRScripts/

```
R version 4.3.0 (2023-04-21 ucrt) -- "Already Tomorrow"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

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.

> |
```

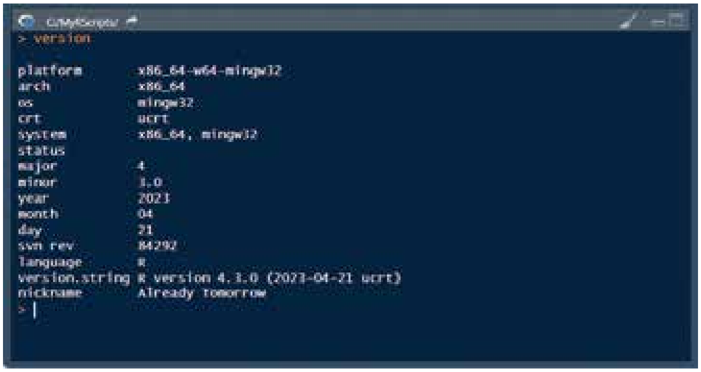
Hot tip

You can click the ➦ arrow button on the Console pane title bar to reveal the working directory's files in the Files pane.

8. You next need to select a pane to work with in RStudio – click on the Console pane to select it

9. Click the 🖌 brush button on the Console pane's title bar, or press the **Ctrl** + **L** keys, to clear existing Console content

10. Now, type **version** at the Console prompt, then hit **Enter** to run the command – see the R interpreter output version details in the Console window



C:/MyRScripts/

```
> version

platform       x86_64-w64-mingw32
arch           x86_64
os             mingw32
crt            ucrt
system         x86_64, mingw32
status
major          4
minor          3.0
year           2023
month          04
day            21
svn rev        84292
language       R
version.string R version 4.3.0 (2023-04-21 ucrt)
nickname       Already Tomorrow
> |
```

Beware

Commands typed at the Console prompt must be entered again to run the command once more – whereas commands typed in the Code Editor can be run repeatedly.

Dark background themes are great for on-screen viewing but all ensuing screenshots throughout this book use a white background theme (TextMate) for better on-page clarity.

Creating an R Script

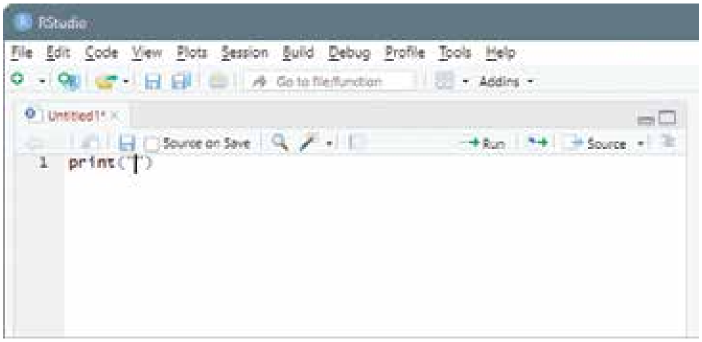
Unless you simply want to test a snippet of code directly at a Console prompt, you should always create an R Script using the Code Editor – so that your code can be run whenever required:

Hello.R

1. Launch RStudio, then click **File**, **New File**, **R Script** on the menu bar to open the **Code Editor** pane

2. Click on the **Code Editor** pane to select it and see a blinking cursor appear – here, type the command **print( )**

3. Type a **"** double-quote character between the command's parentheses, and see RStudio automatically add a second character after the cursor – so you cannot forget the final double-quote that is required to enclose a text string



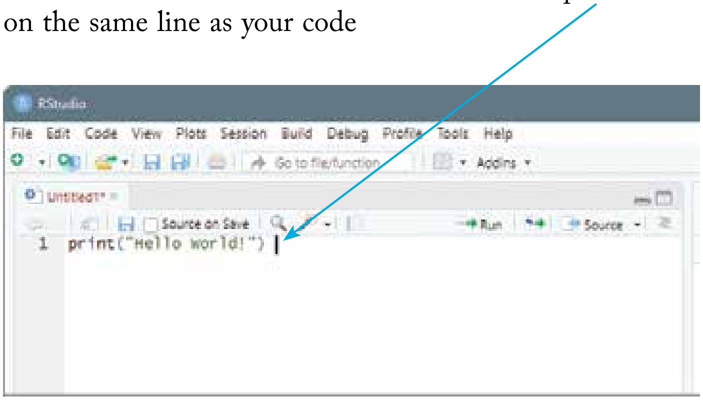
```
print("")
```

Don't forget

The command here is calling the built-in R **print( )** function. The R language is case-sensitive, so typing the command as **Print( )** or **PRINT( )** will simply produce an error.

4. Next, type the traditional program greeting **Hello World!** text string between the double-quotes

5. IMPORTANT: Ensure that the cursor is now positioned on the same line as your code

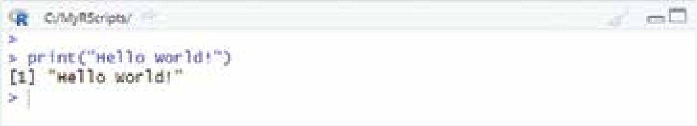


```
print("Hello World!")
```

Beware

The R interpreter will only run code on the line containing the cursor or multiple lines that you have selected (highlighted) by dragging the cursor over them.

6 Click the Run button in the **Code Editor**, or press the **Ctrl** + **Enter** keys, to run the code – see the R interpreter repeat the code and display its output in the **Console** pane

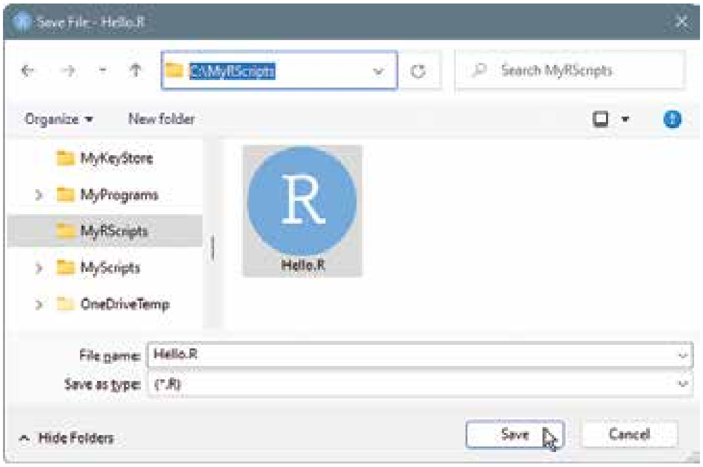


```
>
> print("Hello world!")
[1] "Hello world!"
>
```

The bracketed number [1] that appears before the output indicates that the line begins with the first value of the result. Some results may have multiple values that fill several lines, so this indicator is occasionally useful but can generally be ignored.

7 Click the **Save** button in the **Code Editor**, or press the **Ctrl** + **S** keys, to open the "Save File" dialog

8 Save the R Script as a file named "Hello.R" in the current working directory

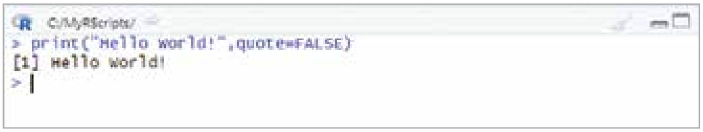


File name: Hello.R

Save as type: (*.R)

9 Edit the command in the **Code Editor** by adding a second argument between the parentheses to become **print("Hello World!" , quote = FALSE)**

10 Run the code again – see the R interpreter repeat the code and display its output with quotes now suppressed



```
> print("Hello world!",quote=FALSE)
[1] Hello world!
> |
```

Click the Open button in the Code Editor, or press the **Ctrl** + **O** keys to open the "Open File" dialog, then choose a saved R Script file to reopen in the Code Editor. Click the arrow button beside the **Open** button to see a list of recently opened files that you can select to quickly reopen.

Summary

- Data is items of information that describe a qualitative status or a quantitative measure of magnitude.
- Devices that are connected to the internet are able to report sensor measurements for analysis in The Cloud.
- The decline in the cost of device sensors has given rise to the Internet of Things that can report on vast amounts of data.
- Big data consists of large data sets that can best be analyzed by computer to reveal pattern and trend insights.
- Data analysis is the practice of converting collected raw data into information that is useful for decision-making.
- Before analysis, raw data must be organized into a structured format and cleaned to remove incomplete, duplicated, and erroneous items.
- After data has been analyzed, the results can be communicated using data visualization to present tables, plots, or charts that efficiently convey the messages within the data.
- R is an interpreted programming language and software environment for data analysis and data visualization.
- RStudio is an Integrated Development Environment for R that provides a code editor, debugger, and visualization tools.
- The RStudio interface consists of a menu bar and toolbar, plus Code Editor, Console, Workspace, and Notebook panes.
- R Script code typed into the Code Editor can be run to see its output appear in the Console.
- Code snippets can be typed at the Console prompt for immediate execution by the R interpreter.
- RStudio's Global Options let you choose colorization themes, font settings, and default working directory.
- R Script in the Code Editor can be saved as a file with a **.R** file extension so the code can be re-run whenever required.