**Overview of R and RStudio**

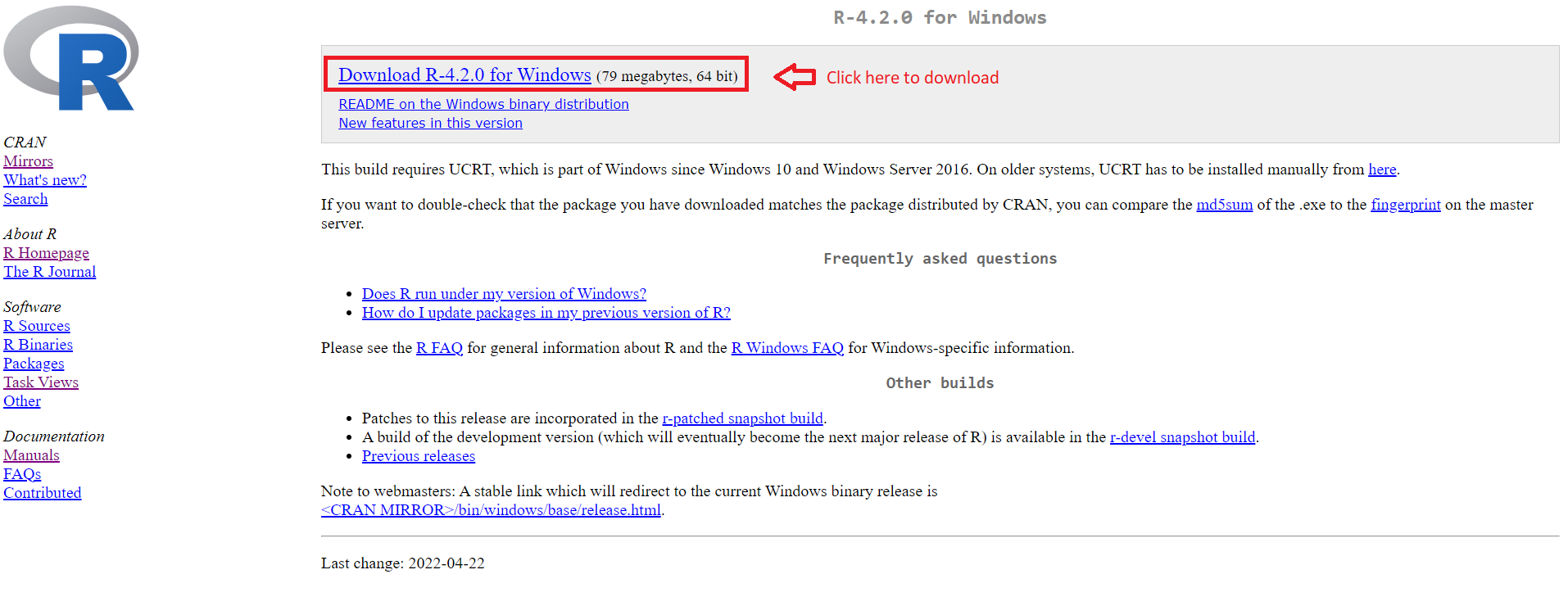
There are several cloud based data science tools that make team collaboration accessible. At times it is useful to work directly on your desktop.

R is a command-line interface; there are several graphical front-ends available. RStudio is an IDE (integrated development environment) for R. It includes the environmental tab, which shows the generated variables. In the history tab, you can see the commands used since starting, and there are other tabs such as files, plots, packages, help, and viewer. It has binaries available for major platforms, including Windows, Linux, and MacOS. This lab includes instructions for downloading and installing R and RStudio on Windows. Mac OS users can download the appropriate .pkg file from <https://cran.r-project.org/bin/macosx/> and follow the instructions.

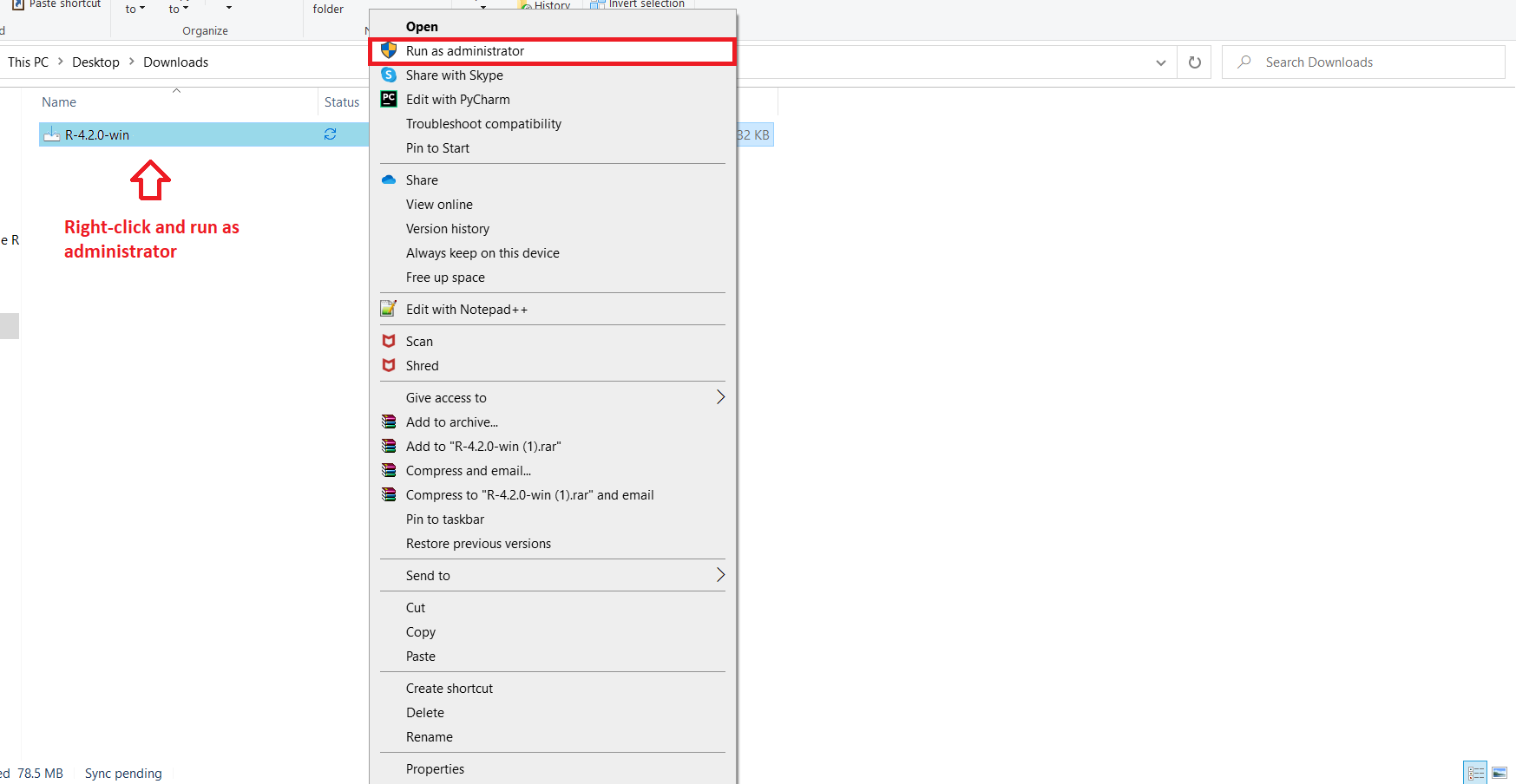
**Exercise 1: Download & Install R on Windows**

**Step 1**: The **latest version** of R can be downloaded by clicking the link.

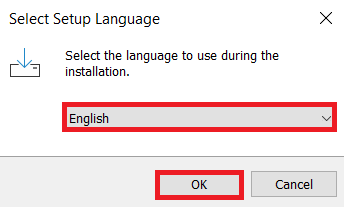
Windows: <https://cran.r-project.org/bin/windows/base/>



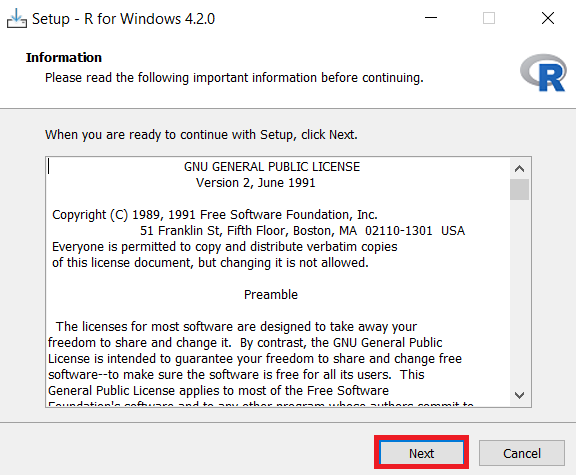
**Step 2**: Once the download completes, **right-click** the downloaded file, and click **Run as administrator**.



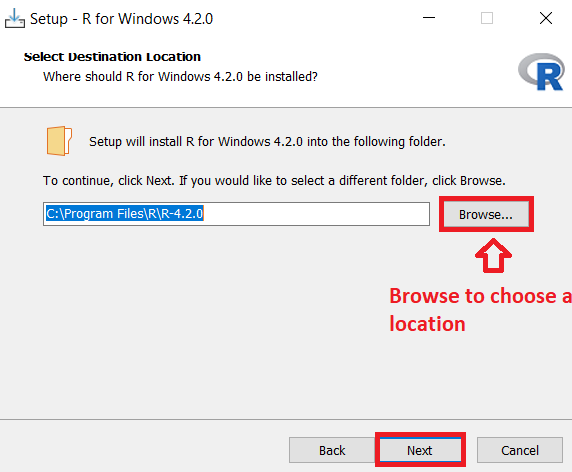
**Step 3**: Select your preferred installation **language**, and click **OK**.



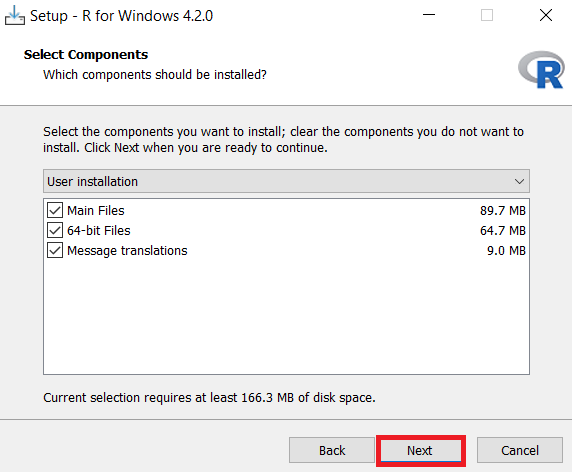
**Step 4**: Read and accept the license and click **Next**.



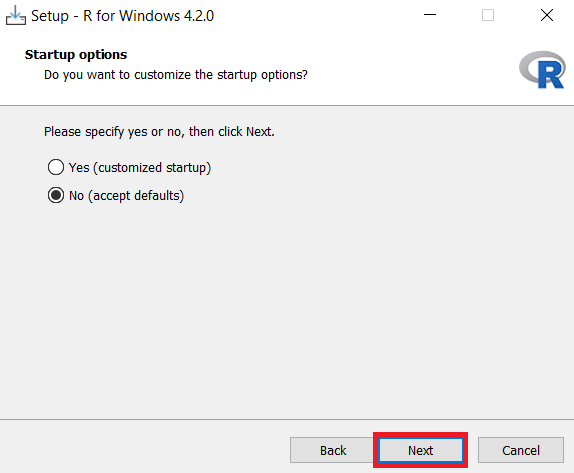
**Step 5**: Select the **Folder** where you would like to install R, or use the **Default** location, and click **Next**.



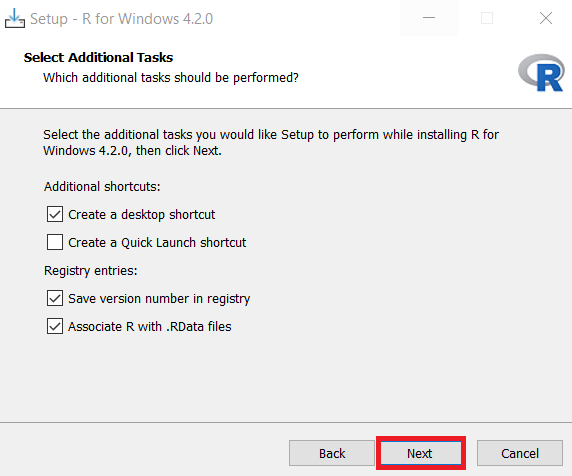
**Step 6**: Select the **Components** you want to install and click **Next**.



**Step 7**: In the **Startup options**, select the **Default** option and click **Next**.



**Step 8**: In the **Select Additional Tasks** window, retain **Default** and click **Next**.



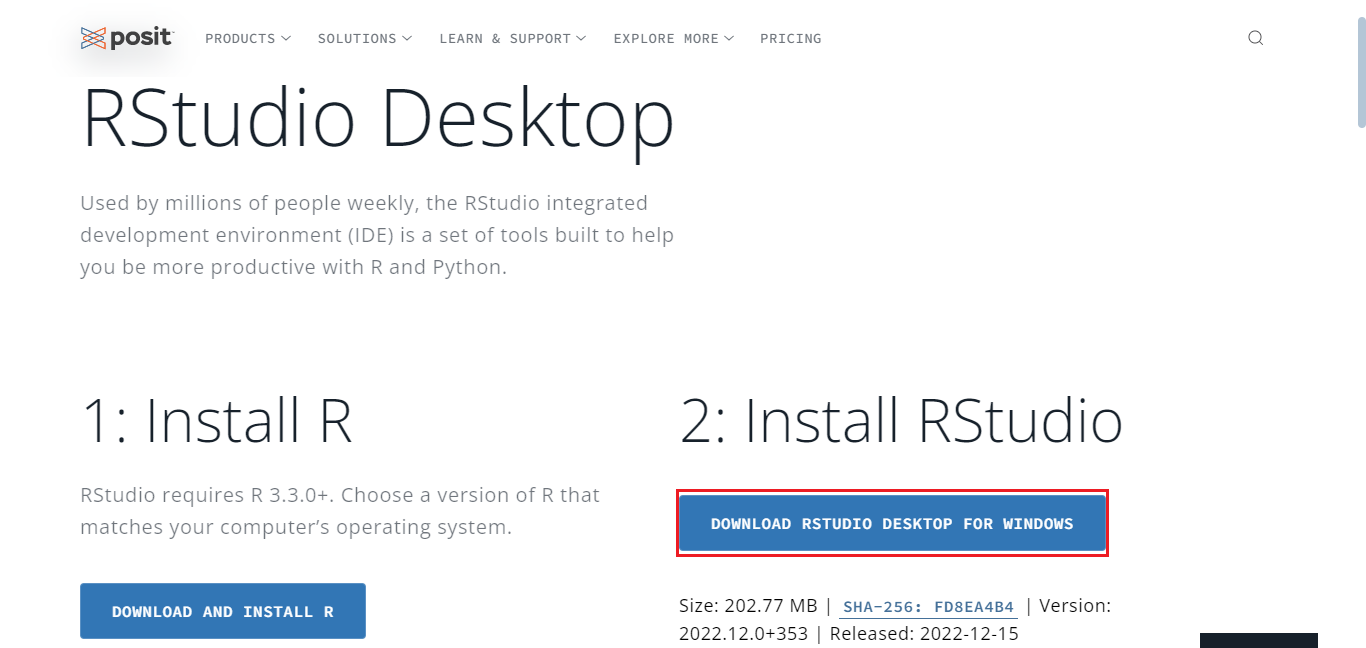
**Step 9**: Once installation is successful, click **Finish** to close the setup.

**Exercise 2: Download & Install RStudio**

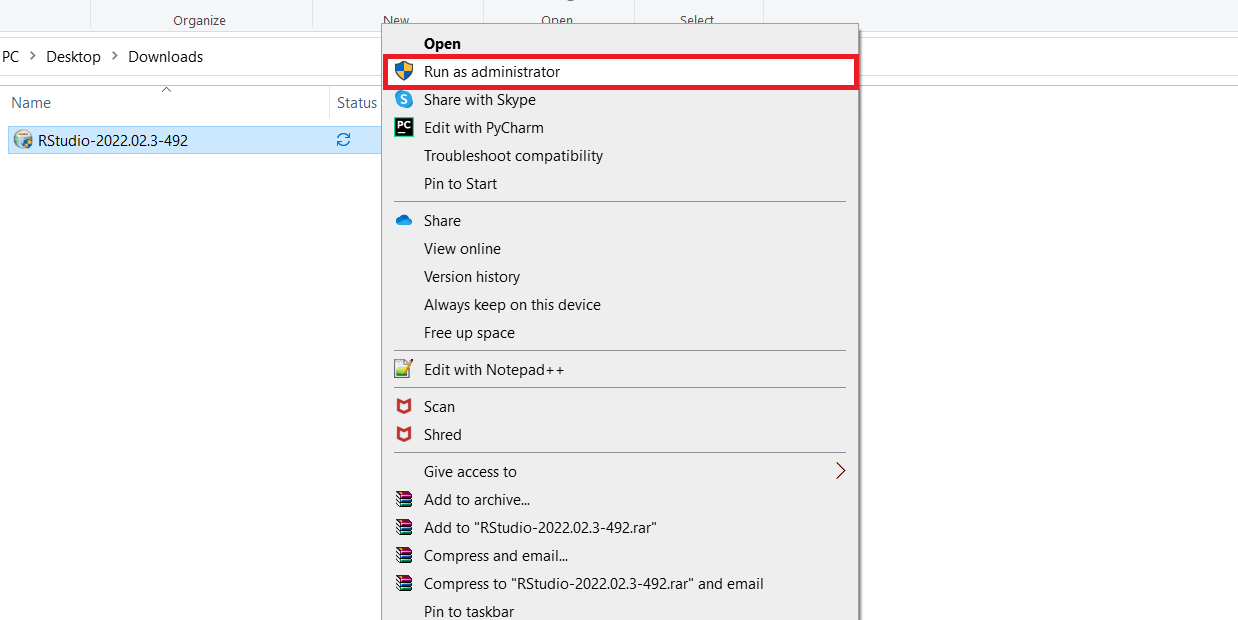
**Step 1**: Use the link below to download **RStudio Desktop** on your local machine.

Link for Download RStudio for windows and mac: <https://posit.co/download/rstudio-desktop/>

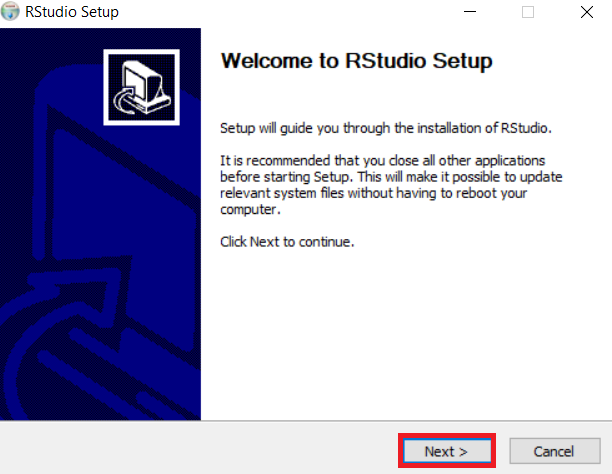
**Step 2**: Click **Download RStudio desktop For Windows**, and your download will start.



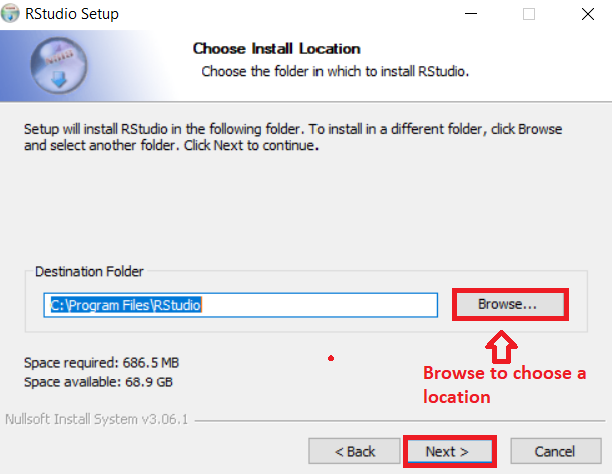
**Step 3**: Once the download completes, **right-click** the setup file, and click **Run as administrator**.



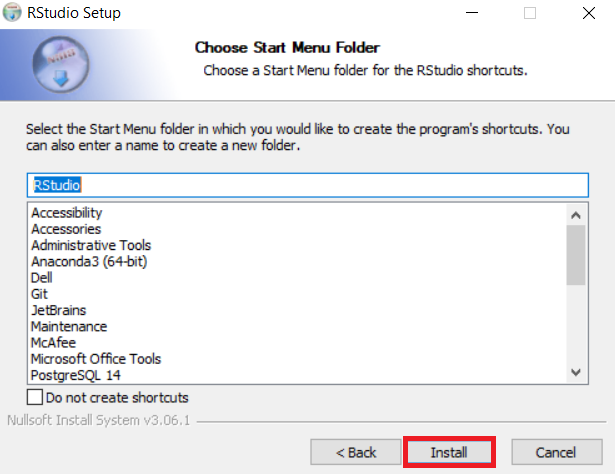
**Step 4**: In the RStudio setup window, click **Next**.



**Step 5**: Select the folder where you would like to install RStudio, or retain the **Default** installation location and click **Next**.



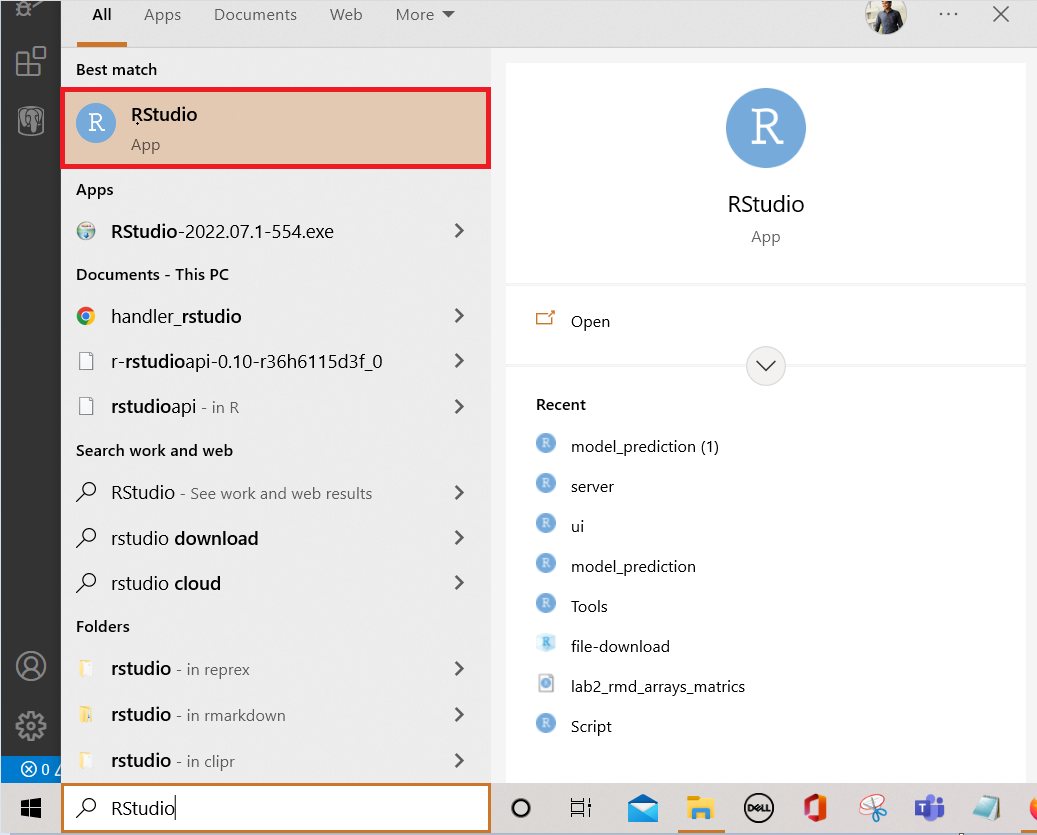
**Step 6**: In the Start menu window, click **Install** to install RStudio.



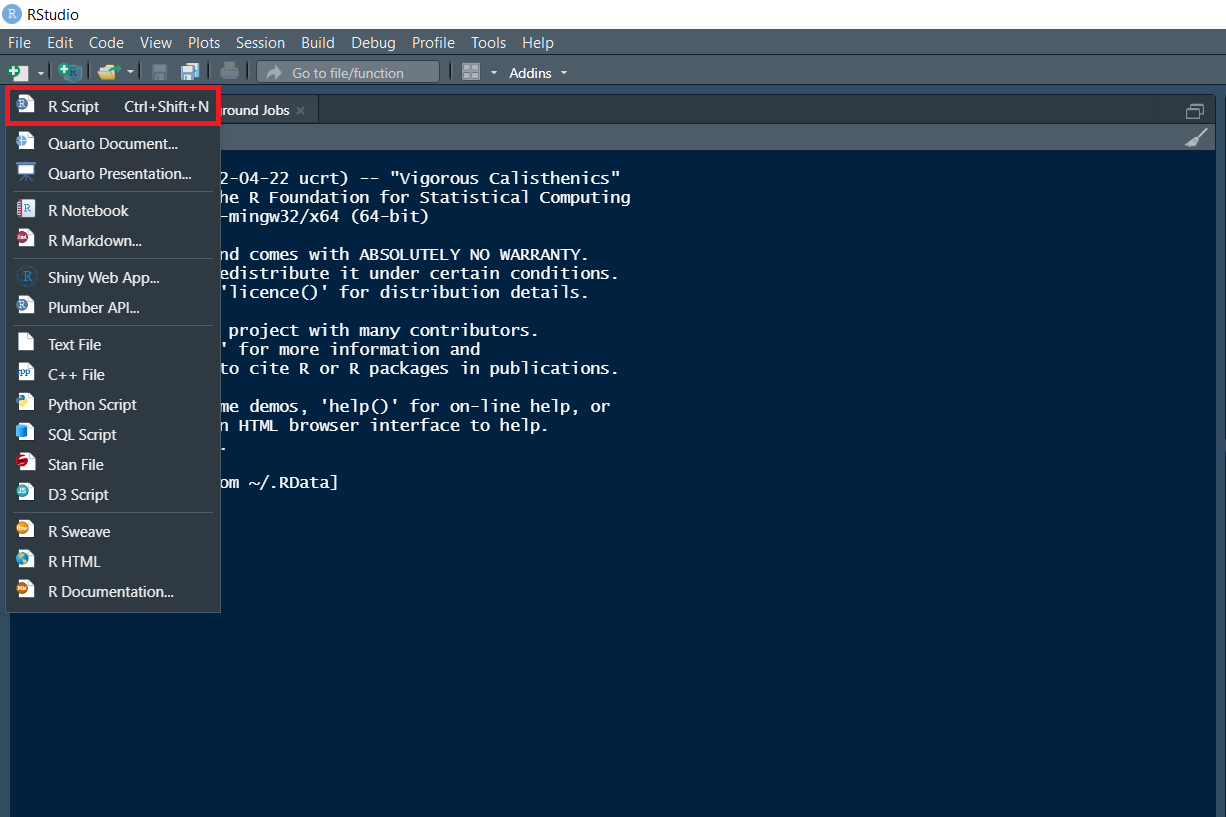
**Step 7**: Once installation completes, click **Finish** to close the window.

**Exercise 3: Execute R code in RStudio**

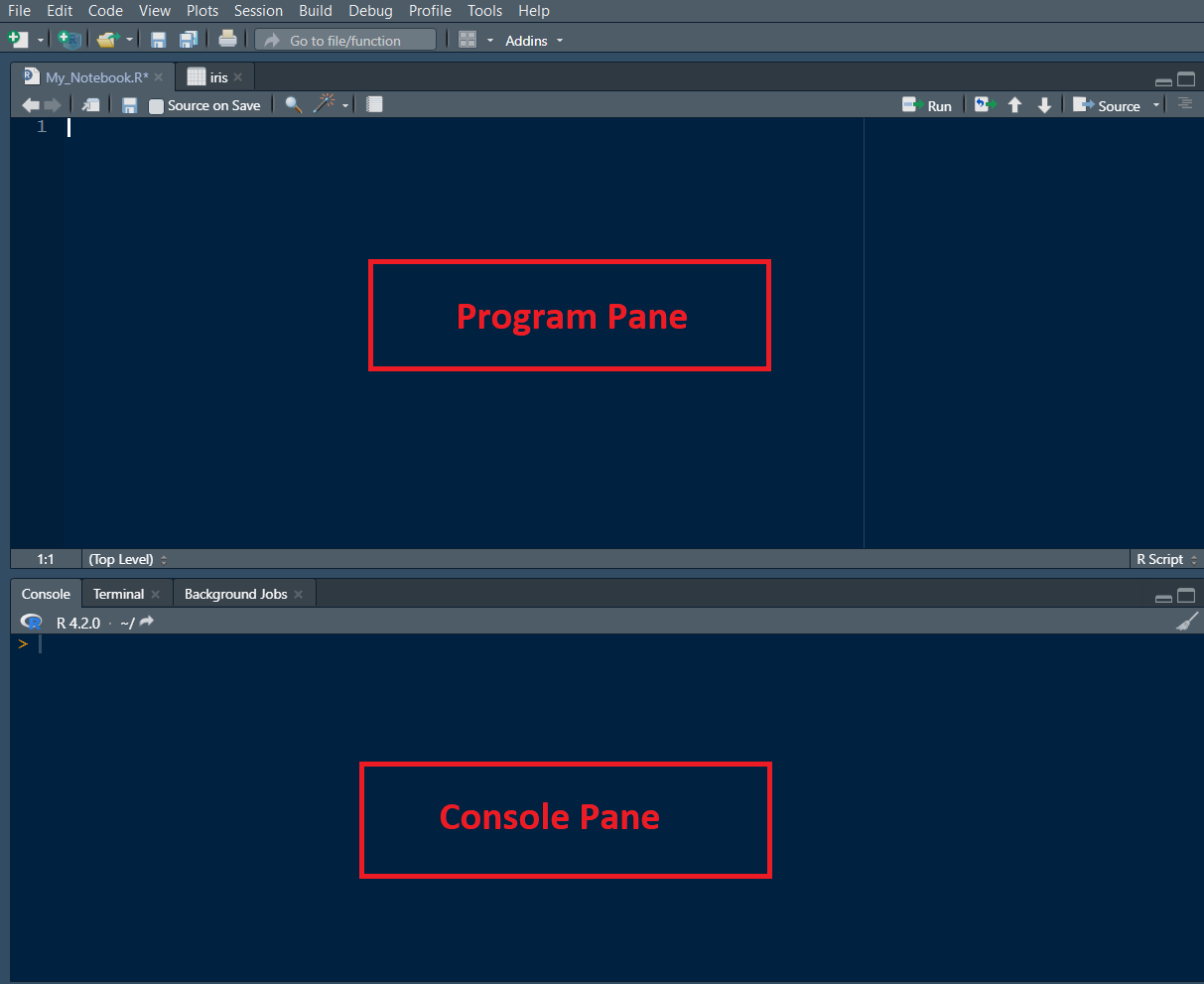
1. Open **RStudio** from the Windows start menu.



1. Click the **plus symbol** on the top left and select **R Script**.

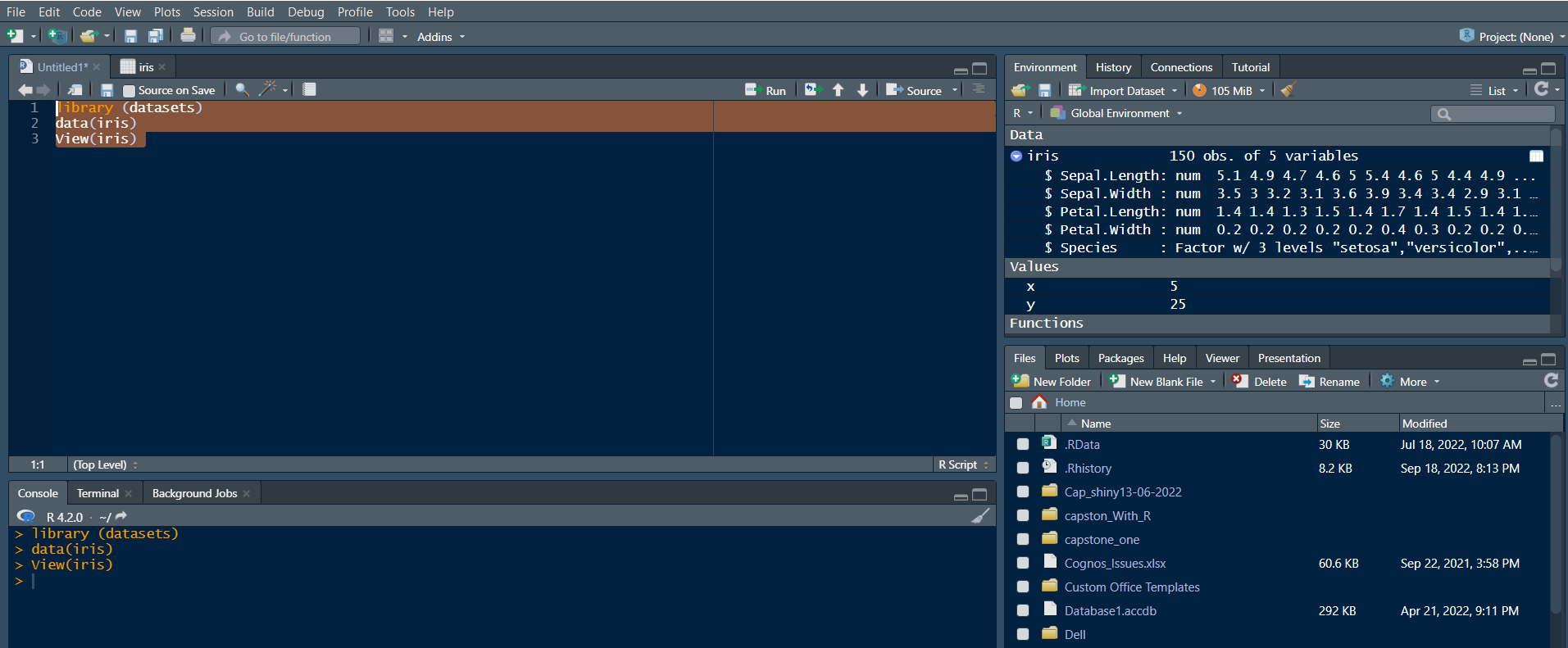


1. An **untitled** R Script panel opens. It would look as follows.

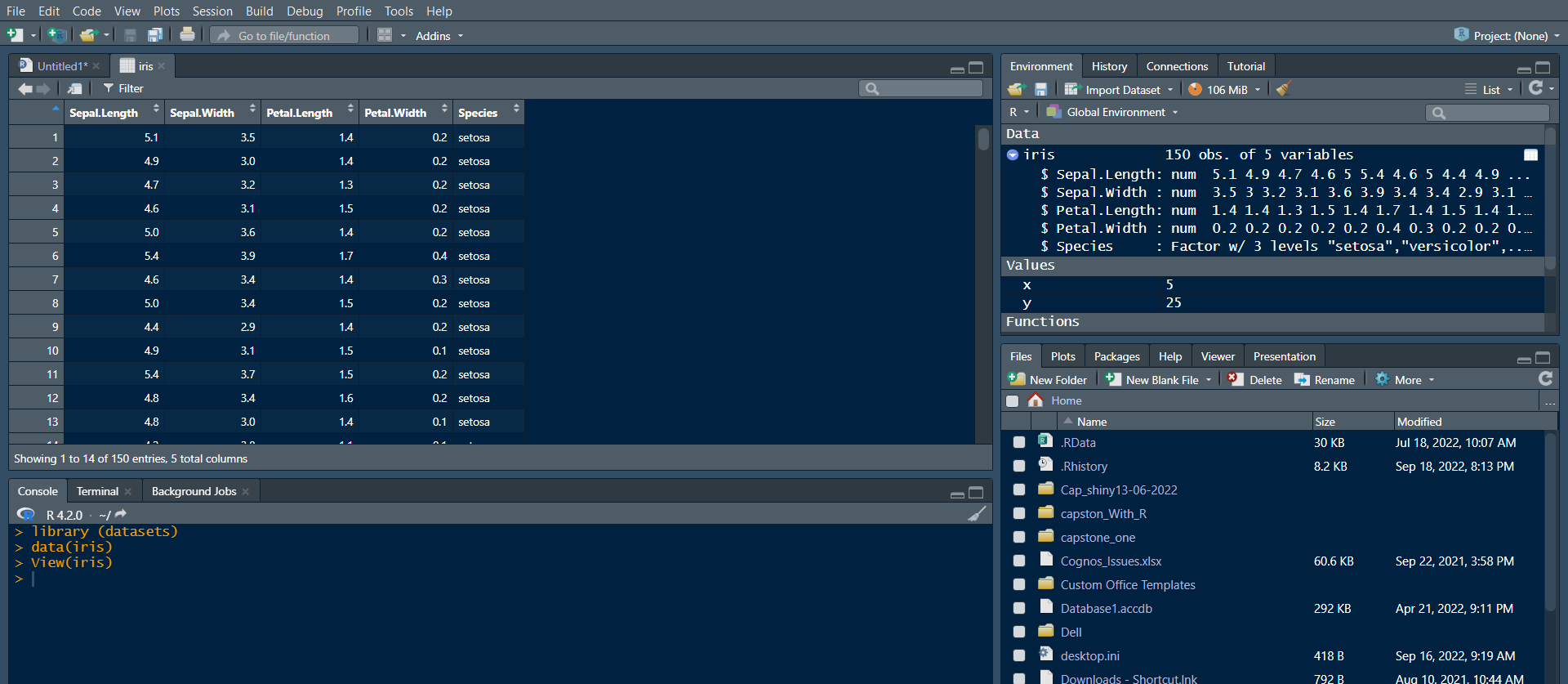


1. Now, load the **iris dataset**. Enter the following **lines** into the **Editor window** which appears. Next, select all of them. Then click the **Run icon** just above the Editor window.
2. 1
3. 2
4. 3
5. library (datasets)
6. data(iris)
7. View(iris)

Copied!

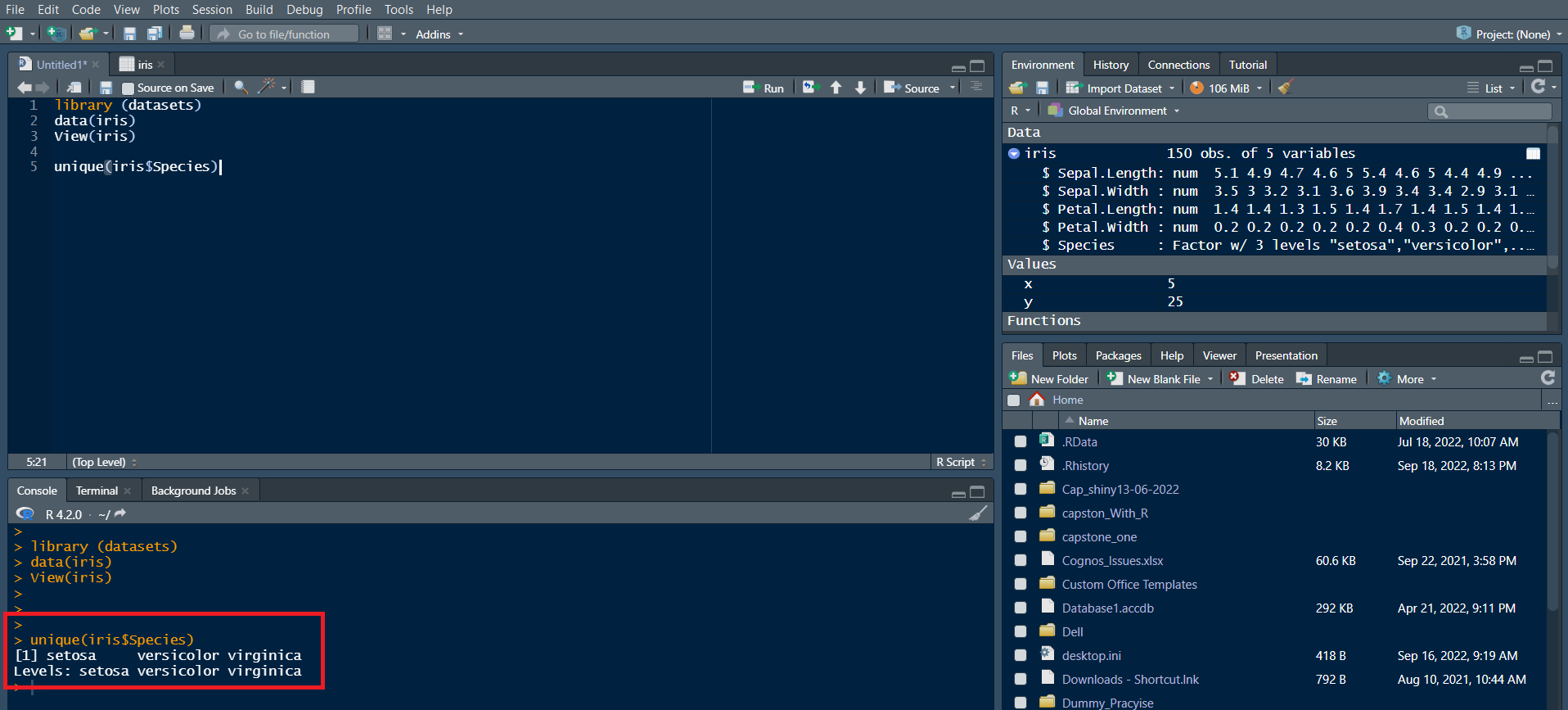
Output

1. You are taken directly to the data view tab to inspect your dataset. You can see five columns in this data set, the first four are floating point, and the last one is the label of the data type string, which contains the category value of your data set. You can see that there are total of 150 entries.



1. Now let’s find how many **different species** are present in the data set. Type the following command in the **Editor window** and **click Run**.
2. 1
3. unique(iris$Species)

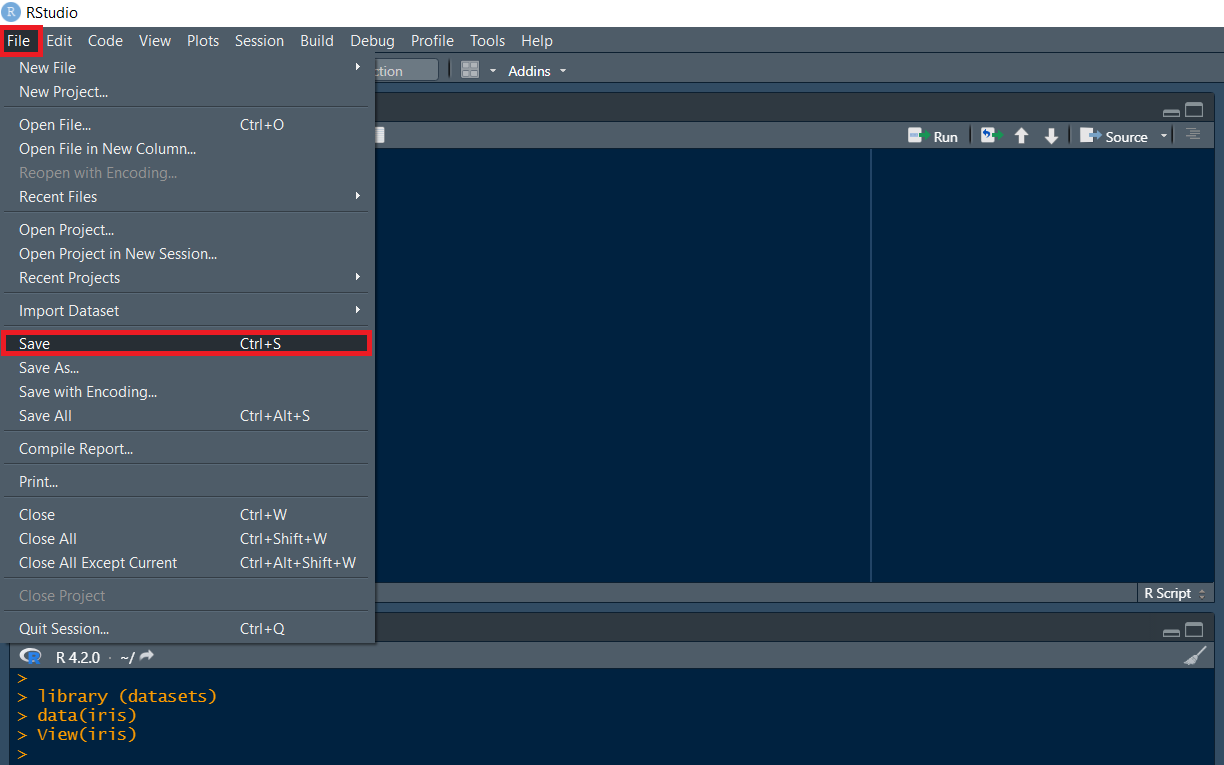
Copied!

Output

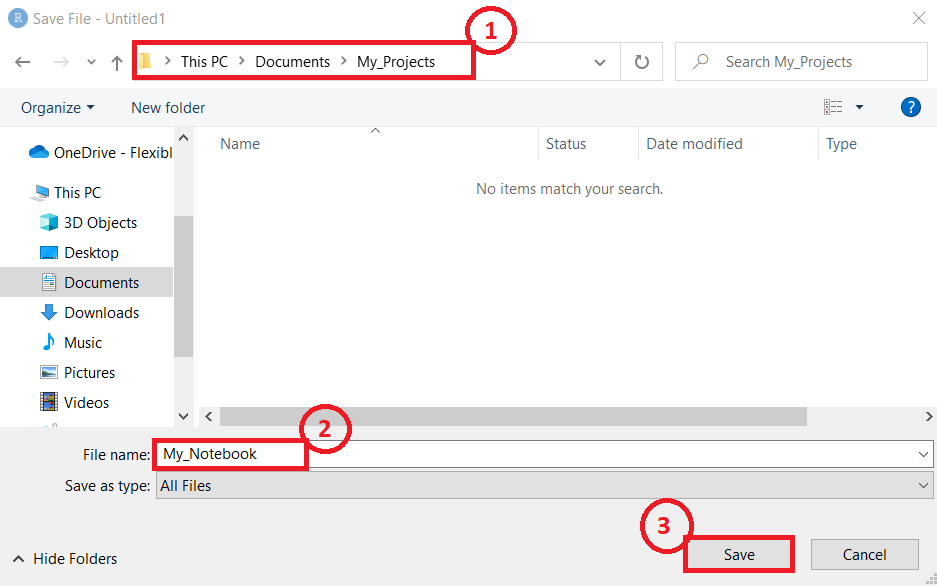
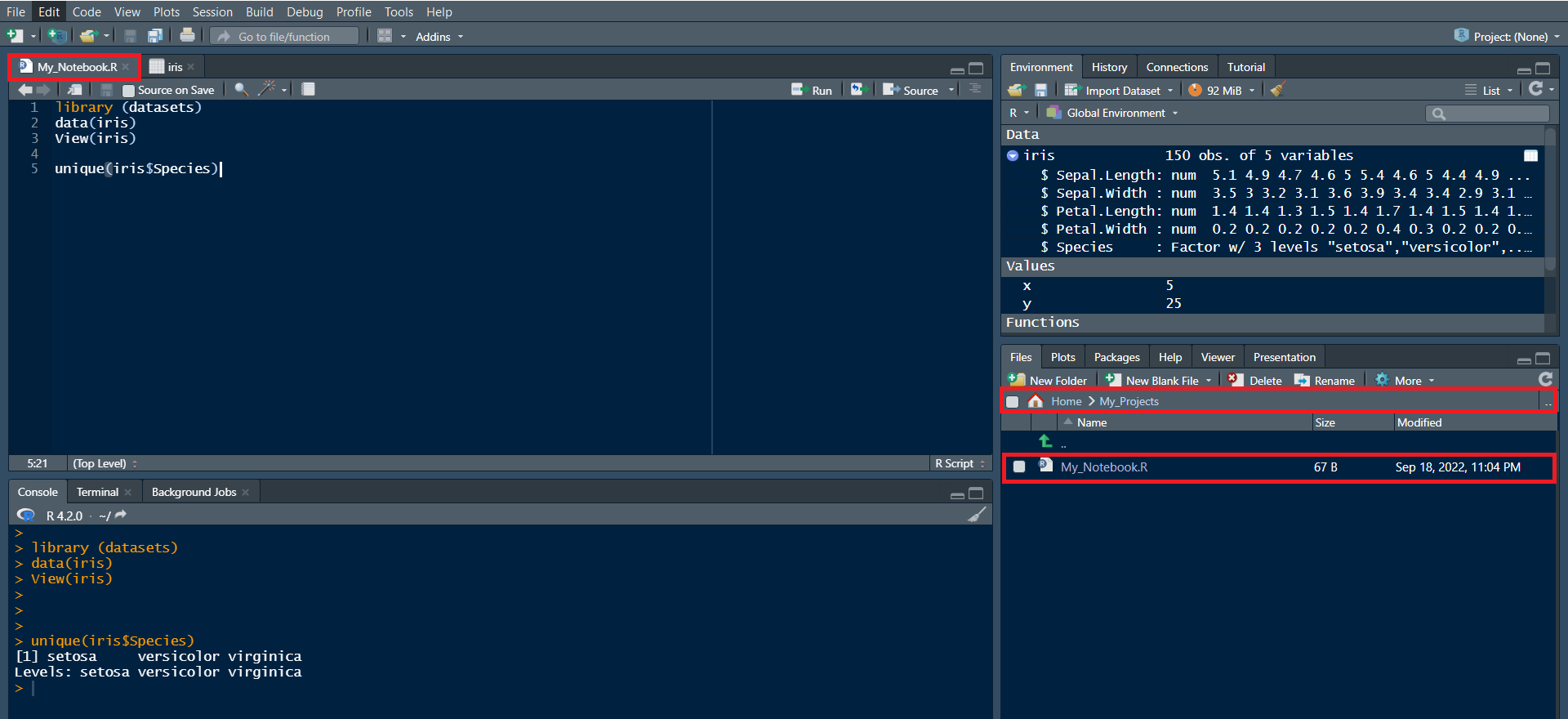
***Note****: In the Console window at the bottom, you will see the result of the executed command and know that only three different species are present in the data set.*

1. **Save & provide a name to your Notebook**.

* To save the notebook, click **Save** or **Save as** in the **File** menu.



* Select the working folder to rename your notebook to ***My\_Notebook***.

We will create different data visualizations using the ggplot package using the inbuilt dataset in R called mtcars

1. Click on the + symbol on the top left and choose R Script from the menu to open a new R edit window in RStudio:

A screenshot of a computer

Description automatically generated

1. Read and view the first 5 rows of the Data using the following:

* library(datasets)
* #Load Data
* data(mtcars)
* #View first 5 rows
* head(mtcars, 5)

Copied!

1. Type this ?mtcars to get information about the variables. This will print the information at the bottom right panel, on the Help tab
2. Copy and paste the following code to load the ggplot package and create a scatterplot of disp and mpg.

* #load ggplot package
* library(ggplot2)
* #create a scatterplot of displacement (disp) and miles per gallon (mpg)
* ggplot(aes(x=disp,y=mpg,),data=mtcars)+geom\_point()

1. Use the following code to add a title.

* #Add a title
* ggplot(aes(x=disp,y=mpg,),data=mtcars)+geom\_point()+ggtitle("displacement vs miles per gallon")

1. Use the following code to change the name of the x-axis and y-axis

* #change axis name
* ggplot(aes(x=disp,y=mpg,),data=mtcars)+geom\_point()+ggtitle("displacement vs miles per gallon") + labs(x = "Displacement", y = "Miles per Gallon")

1. Use the following to create a boxplot of the the distribution of mpg for the individual Engine types vs Engine (0 = V-shaped, 1 = straight)  
   To do this you have to make vs a string or factor.

* #make vs a factor
* mtcars$vs <- as.factor(mtcars$vs)
* #create boxplot of the distribution for v-shaped and straight Engine
* ggplot(aes(x=vs, y=mpg), data = mtcars) + geom\_boxplot()

1. Add color to the boxplots to help differentiate:

* ggplot(aes(x=vs, y=mpg, fill = vs), data = mtcars) +
* geom\_boxplot(alpha=0.3) +
* theme(legend.position="none")

1. Finally, let us create the histogram of weight wt.

* ggplot(aes(x=wt),data=mtcars) + geom\_histogram(binwidth=0.5)