

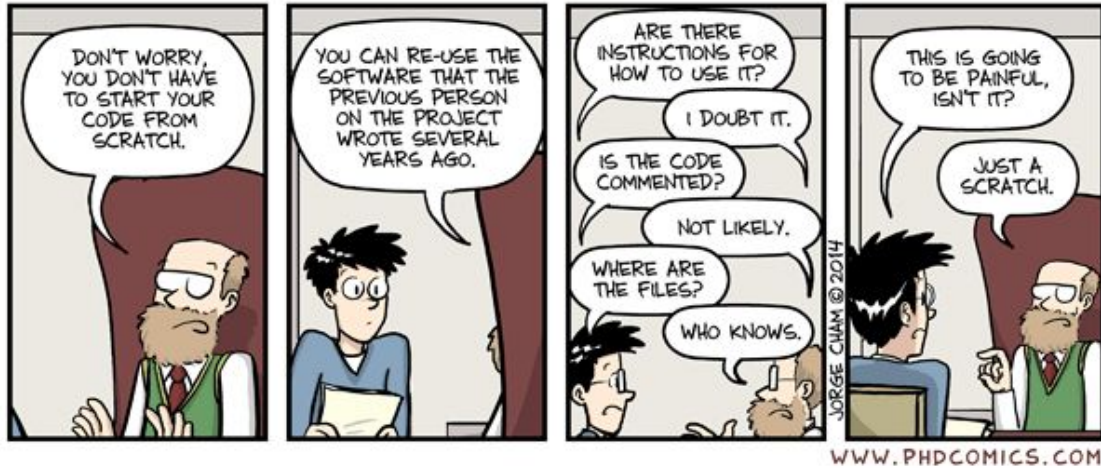


# Introduction to R, RStudio, and RStudio Server

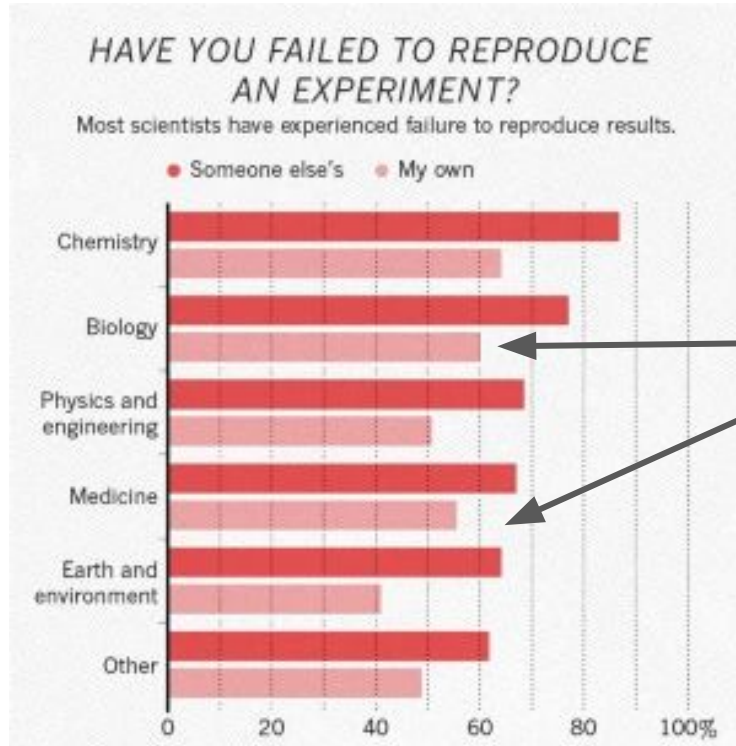
The Data Lab

Powered by Alex's Lemonade Stand Foundation

# Who's been here before?



# Reproducibility in 2016



55% and 60% of biologists and clinicians, respectively, could not reproduce their own results.

Baker, M. 1,500 scientists lift the lid on reproducibility.  
*Nature* 533, 452–454 (2016).  
<https://doi.org/10.1038/533452a>

# Command line vs GUI (graphics user interface)

- An interface is how you interact with a program
- GUI's have buttons you can *click* to do things, but...
- Command-line interfaces (CLI) have you *type* out things to do them



## One in five genetics papers contains errors thanks to Microsoft Excel

By [Jessica Boddy](#) | Aug. 29, 2016, 1:45 PM

What you type	What you see	How Excel stores it
MARCH1	1-MAR	42430
SEPT2	2-SEP	42615

<https://www.sciencemag.org/news/2016/08/one-five-genetics-papers-contains-errors-thanks-microsoft-excel>

Ziemann et al. Genome Biology (2016) 17:177 DOI 10.1186/s13059-016-1044-7

# The problem continues...

NEWS | 13 August 2021 | Correction [25 August 2021](#)

## Autocorrect errors in Excel still creating genomics headache

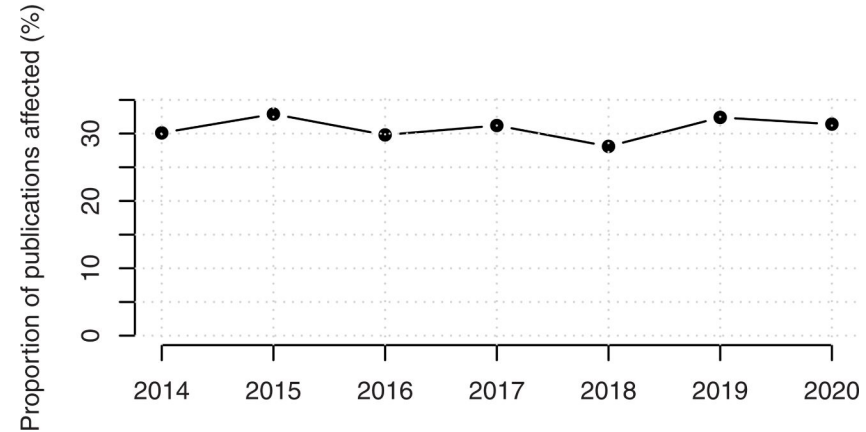
**Despite geneticists being warned about spreadsheet problems, 30% of published papers contain mangled gene names in supplementary data.**

[Dyani Lewis](#)



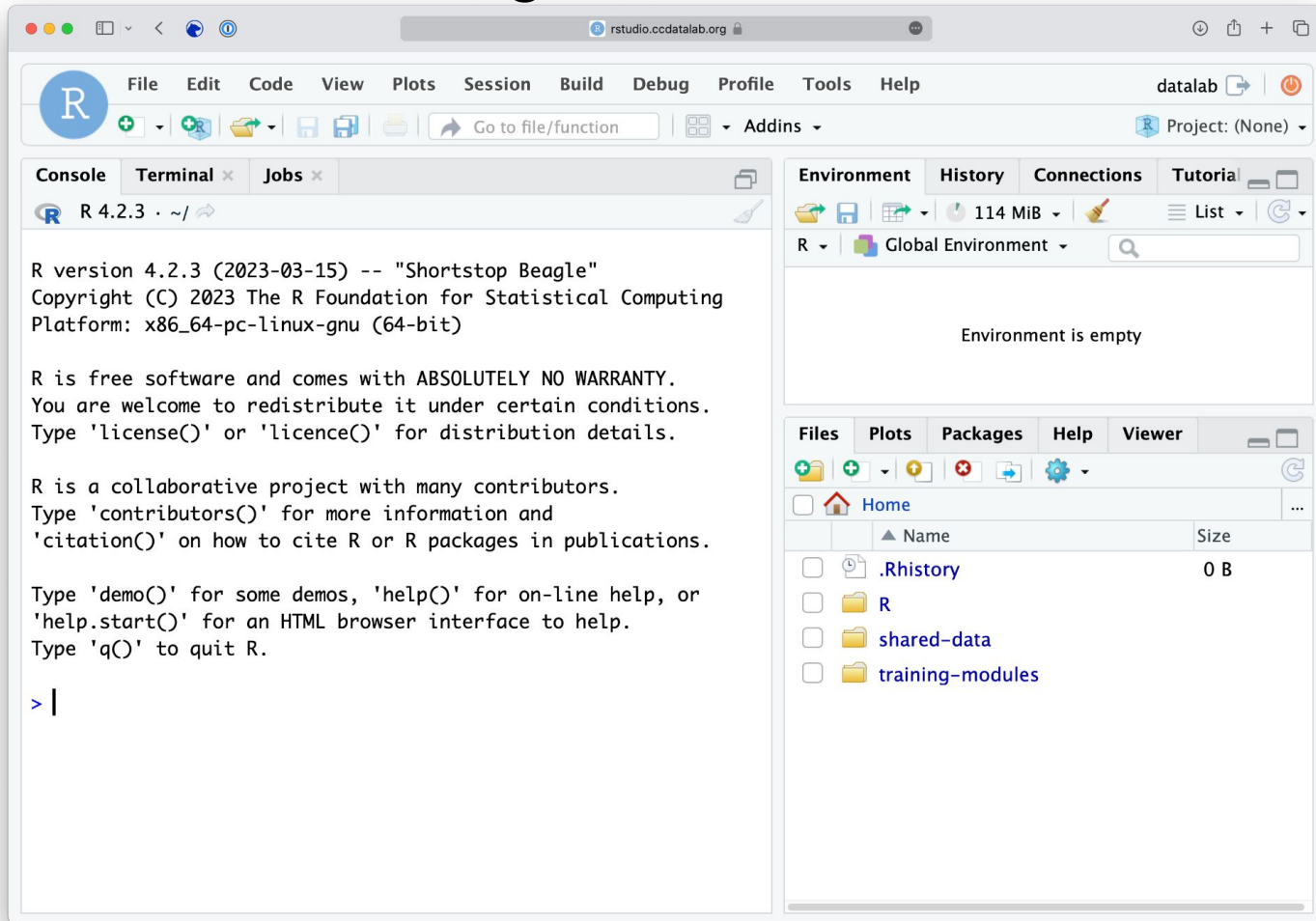
Embarrassing autocorrect mistakes are common fodder for Internet listicles and Twitter threads. But they are also the bane of geneticists using spreadsheet programs such as Microsoft Excel. Five years after a study showed that [autocorrect problems](#) were widespread, the academic literature is still littered with error-riddled spreadsheets, according to an

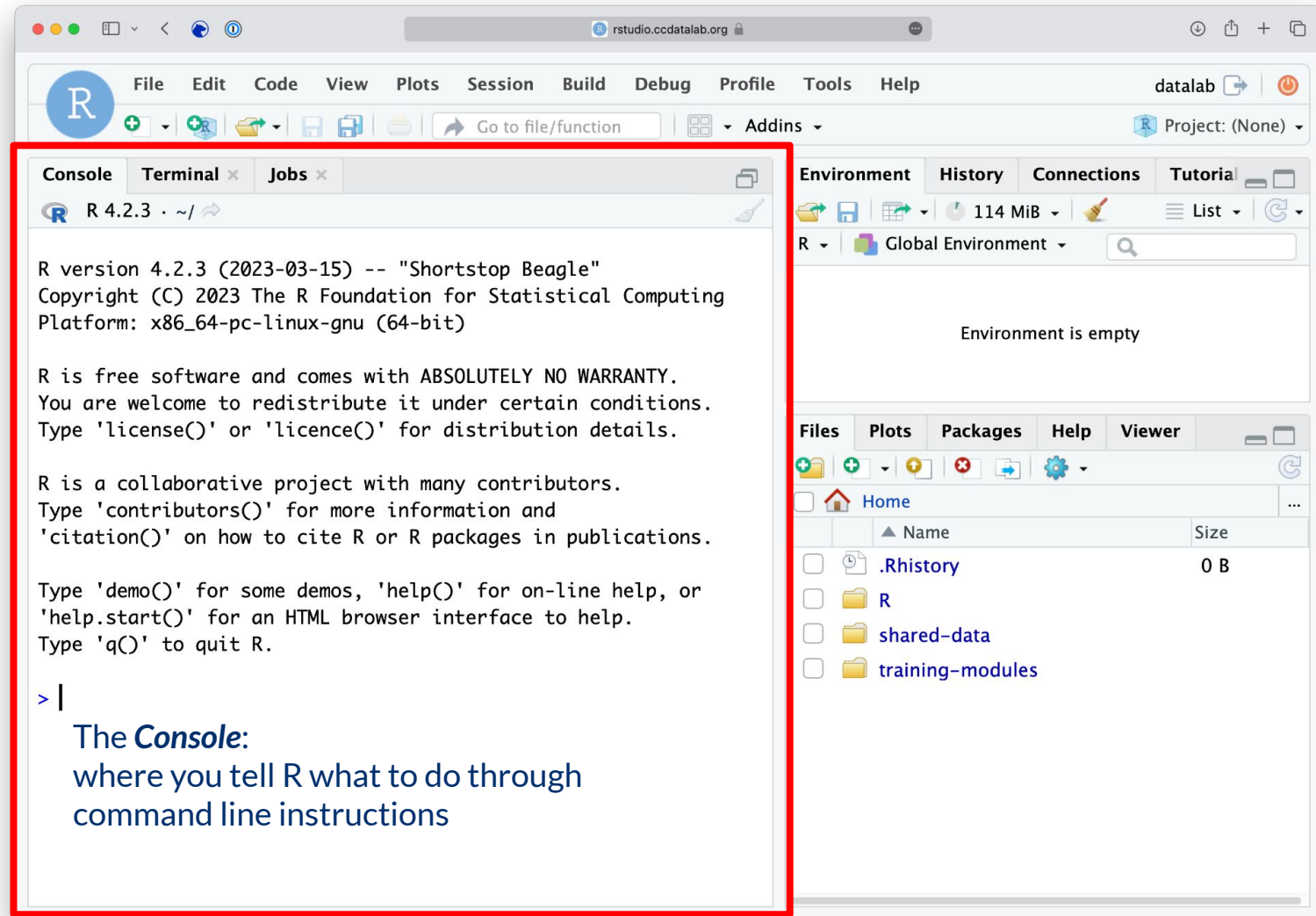
<https://www.nature.com/articles/d41586-021-02211-4>



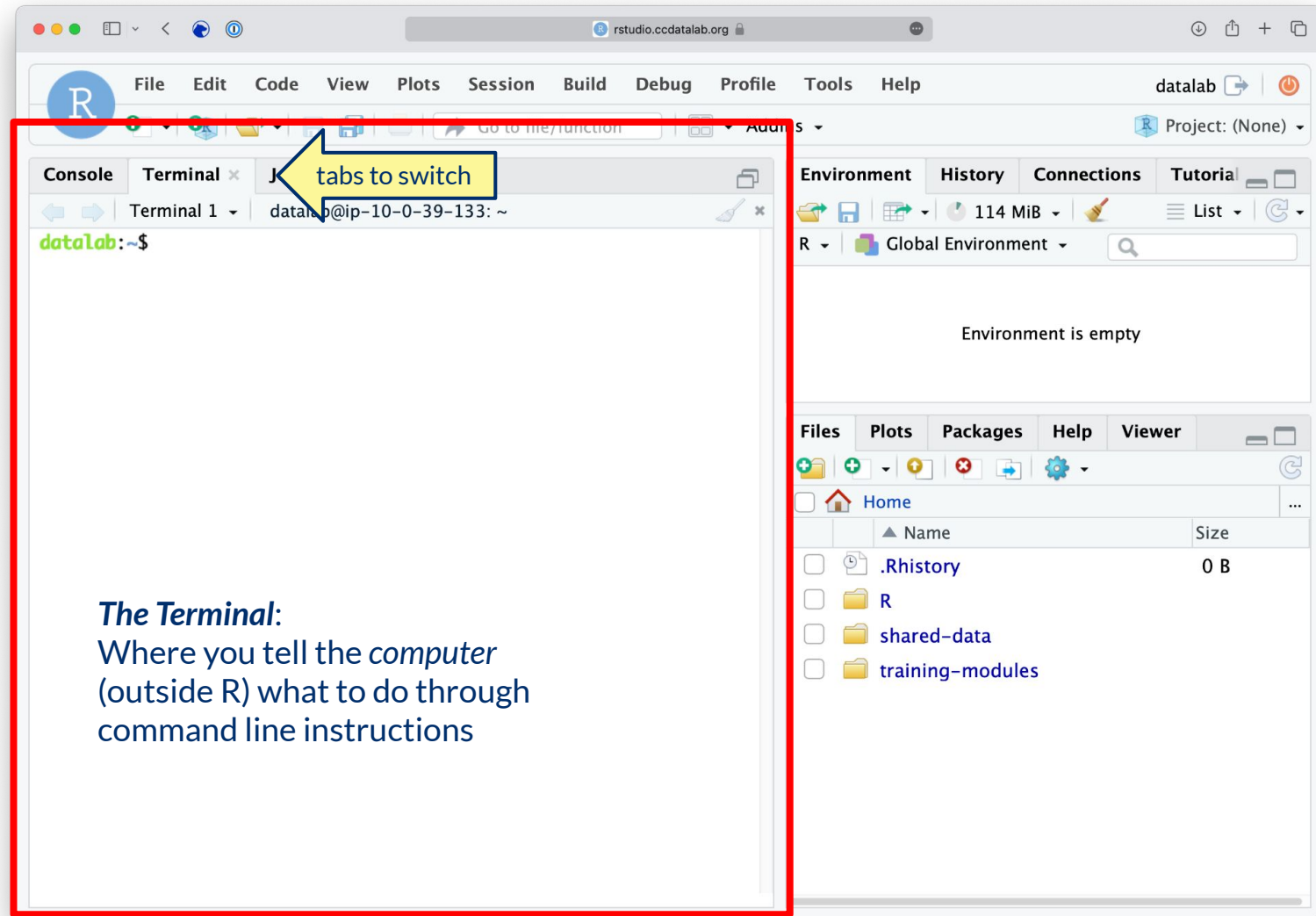
<https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1008984>

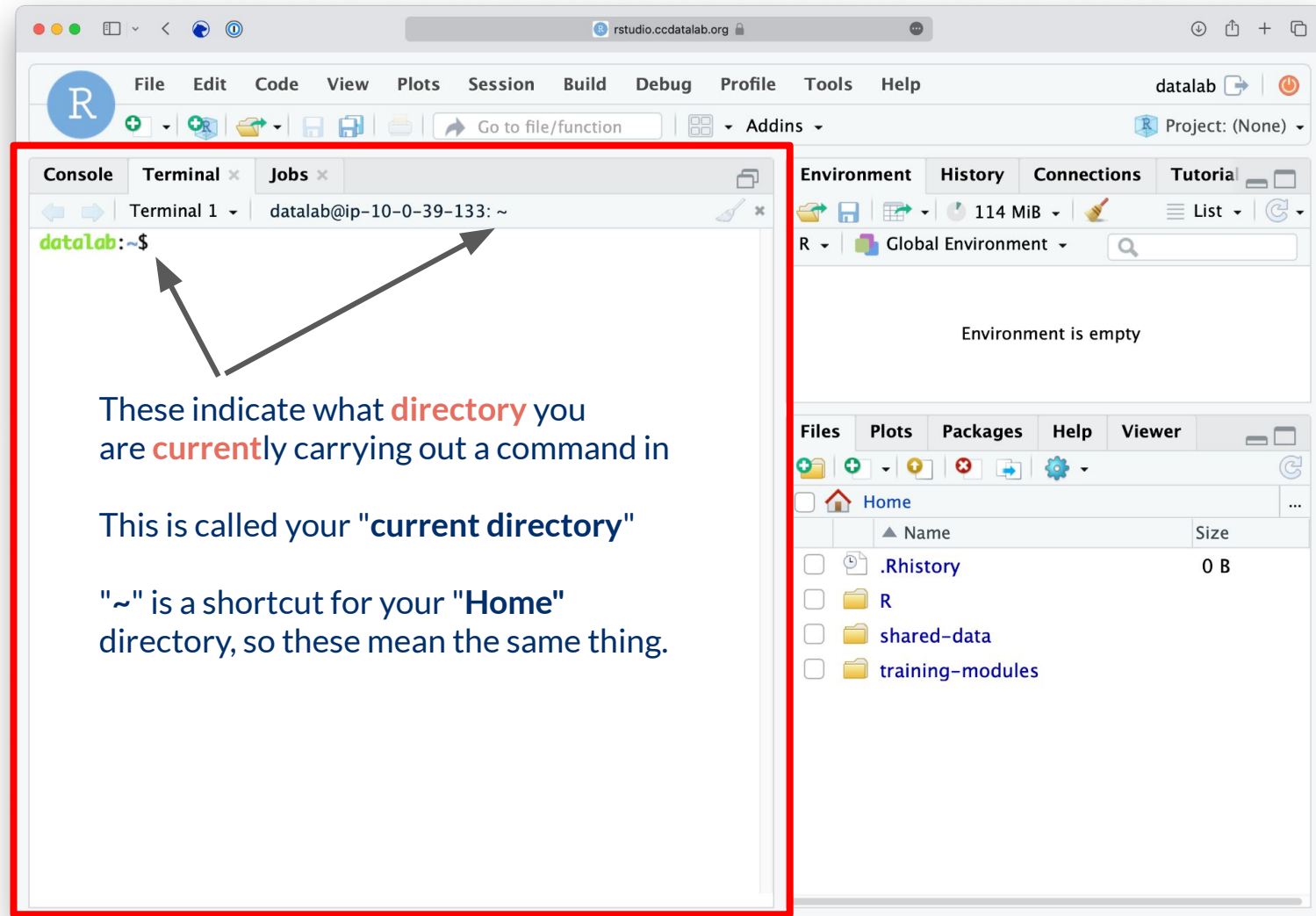
# RStudio Server: A basic guide









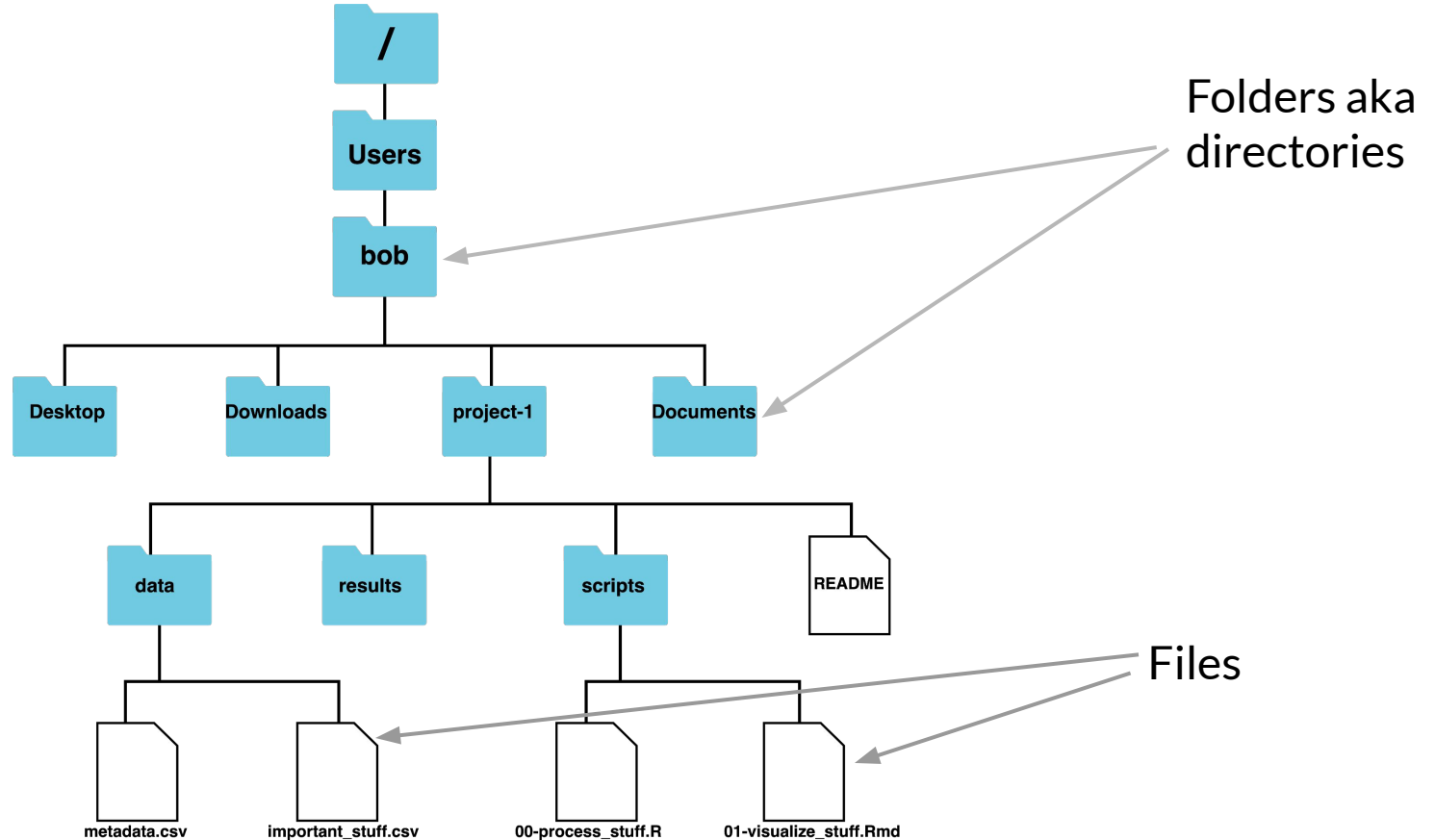


These indicate what **directory** you are **currently** carrying out a command in

This is called your "**current directory**"

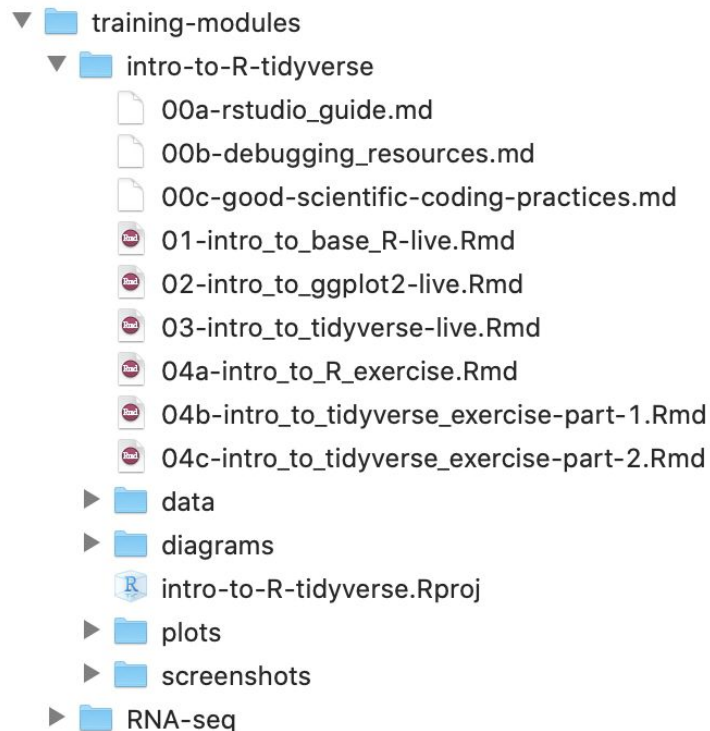
"~" is a shortcut for your "**Home**" directory, so these mean the same thing.

# Example of a filesystem hierarchy



# Our "working directory" for today's *Intro to R* module

When we are working on the command line, we have to keep track of where the files we are using are being kept.



The screenshot shows the RStudio application window. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The top toolbar contains icons for creating a new file, opening a file, saving, and other standard RStudio functions. The top right shows the 'datablab' session name and a power button icon.

The main workspace is divided into four panes:

- Console:** Shows the terminal output for the 'datablab' session. The commands entered are `ls` and `cd training-modules/`. The output of `ls` is `shared-data training-modules`.
- Terminal:** Shows the terminal prompt `datablab@ip-10-0-39-133: ~`.
- Environment:** Shows the current environment. It is empty, with the message 'Environment is empty'.
- Files:** Shows the file explorer. The current directory is 'Home'. The files listed are `.Rhistory`, `R`, `shared-data`, and `training-modules`.

Below the Console pane, there is a text box with the following content:

Some common **Terminal** commands:

- ls** - list the files and folders in a directory (files that start with a '.' are not shown by default)
- cd** - change directories

The screenshot displays the RStudio application window. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The toolbar below the menu contains various icons for file operations and navigation. The main workspace is divided into four panes: Console, Terminal, Jobs, and Environment. The Console pane shows a terminal session with the following commands and output:

```
datalab:~$ ls
R shared-data training-modules
datalab:~$ cd training-modules/
datalab:~/training-modules$ ls
LICENSE.md README.md intro-to-R-tidyverse
module-cheatsheets scrna-seq
datalab:~/training-modules$
```

The words "training-modules" in the last command are highlighted with a red box. Below the terminal output, a text box contains the following explanation:

The words in front of our cursor have changed because we are now "in" the `training-modules` directory

The Environment pane on the right shows the current environment is empty. Below the Environment pane, the Files pane displays the file structure of the current directory:

	Name	Size	Modified
<input type="checkbox"/>	.Rhistory	0 B	Jan 31, 2023
<input type="checkbox"/>	R		
<input type="checkbox"/>	shared-data		
<input type="checkbox"/>	training-modules		

The screenshot displays the RStudio application window. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The toolbar below the menu contains various icons for file operations and navigation. The main workspace is divided into four panes: Console, Terminal, Jobs, and Environment. The Console pane shows a terminal session where the user has navigated to the `training-modules` directory. The Environment pane is empty, indicating no objects are loaded in the current environment. The Files pane at the bottom right shows the file structure of the current project, which is the `Home` directory. The file list includes `.Rhistory`, `R`, `shared-data`, and `training-modules`.

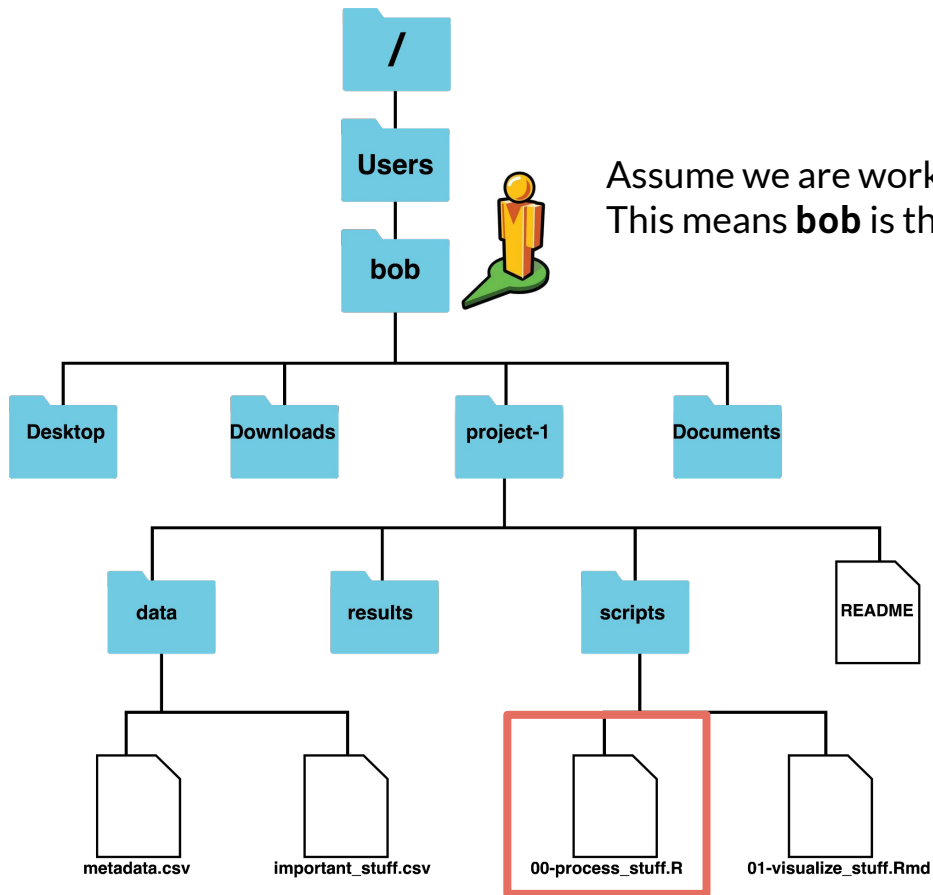
```
datalab:~$ ls
R shared-data training-modules
datalab:~$ cd training-modules/
datalab:~/training-modules$ ls
LICENSE.md README.md intro-to-R-tidyverse
module-cheatsheets scrna-seq
datalab:~/training-modules$
```

The words in front of our cursor have changed because we are now “in” the `training-modules` directory

The files tab over here does **NOT** reflect your current directory or any changes within it

	Name	Size	Modified
<input type="checkbox"/>	.Rhistory	0 B	Jan 31, 2023
<input type="checkbox"/>	R		
<input type="checkbox"/>	shared-data		
<input type="checkbox"/>	training-modules		

# We are always working somewhere!



Assume we are working “from” the **bob** directory..  
This means **bob** is the *current/working directory*

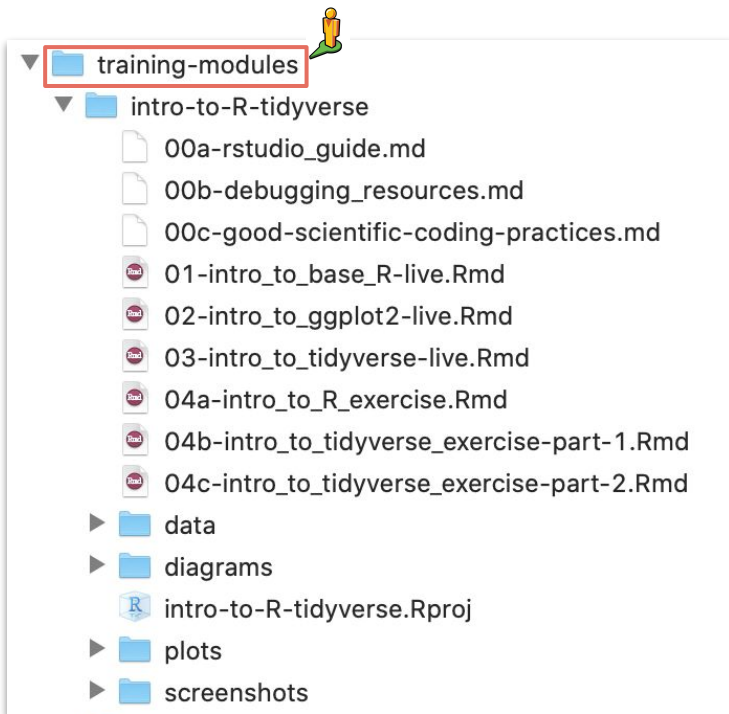
The file we are working on

Relative path: **project-1/scripts/00-process\_stuff.R**



# File paths: Directions to a file or folder

Let's say we want access to `01-intro_to_base_R-live.Rmd`

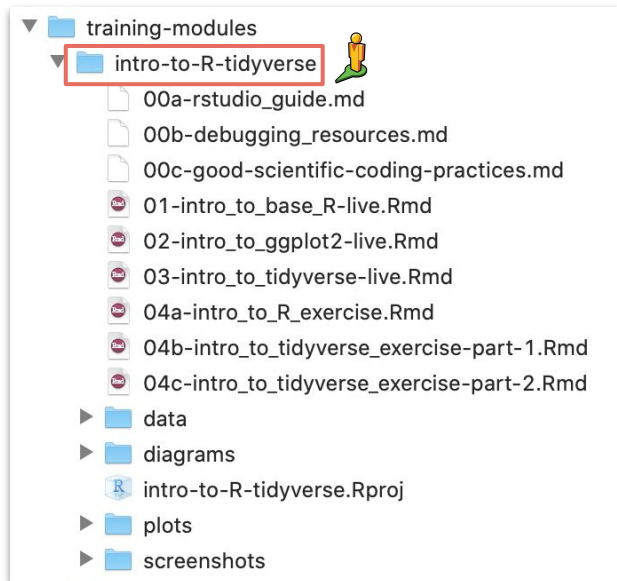
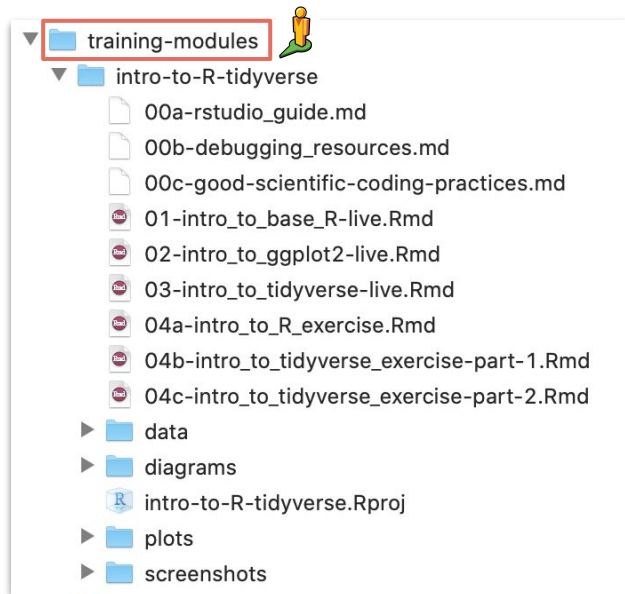



Current directory = `training-modules` 


File path = `intro-to-R-tidyverse/01-intro_to_base_R-live.Rmd`

# Relative file paths

Let's say we want to work with **01-intro\_to\_base\_R-live.Rmd**



 training-modules

 training-modules/intro-to-R-tidyverse

Relative file path =  
**intro-to-R-tidyverse/01-intro\_to\_base\_R-live.Rmd**

Relative file path = **01-intro\_to\_base\_R-live.Rmd**



# Introduction to R

The Data Lab

# R programming

Programming: making executable scripts for accomplishing a task  
(in this case, data analysis is our task)

Scripts allow others to see, step-by-step, what you did.

## Why we use R:

- It's free and open-source
- People make cool packages that do stuff for us
- Many researchers in genomics use it (as well as Python)



# R, RStudio, and RStudio Server

R is a statistical programming language.



RStudio is an IDE for working in R

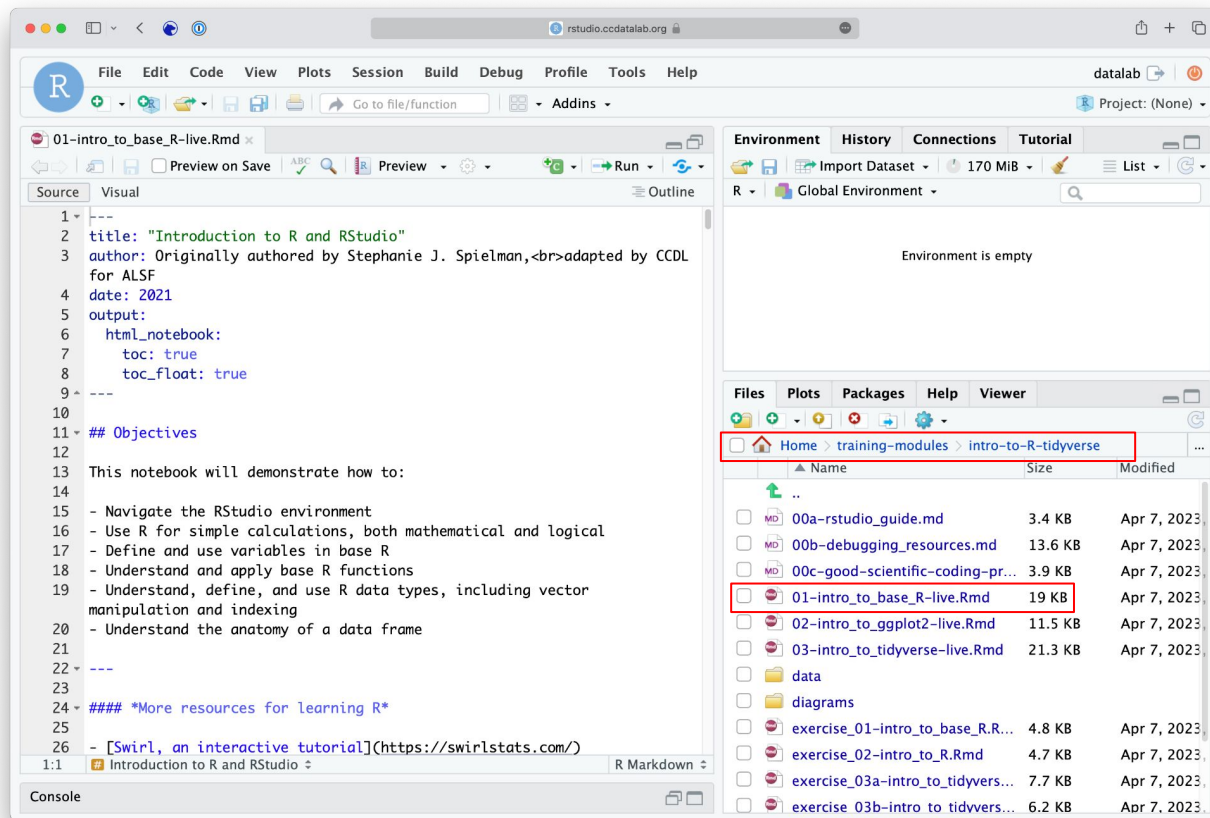
- IDE: Integrated Development Environment
- We write R code using the (free!) RStudio IDE



**RStudio Server** allows us to run the **RStudio IDE** from a browser

# R Notebooks

Use the "Files" tab to open: `training-modules/intro-to-R-tidyverse/01-intro_to_base_R-live.Rmd`

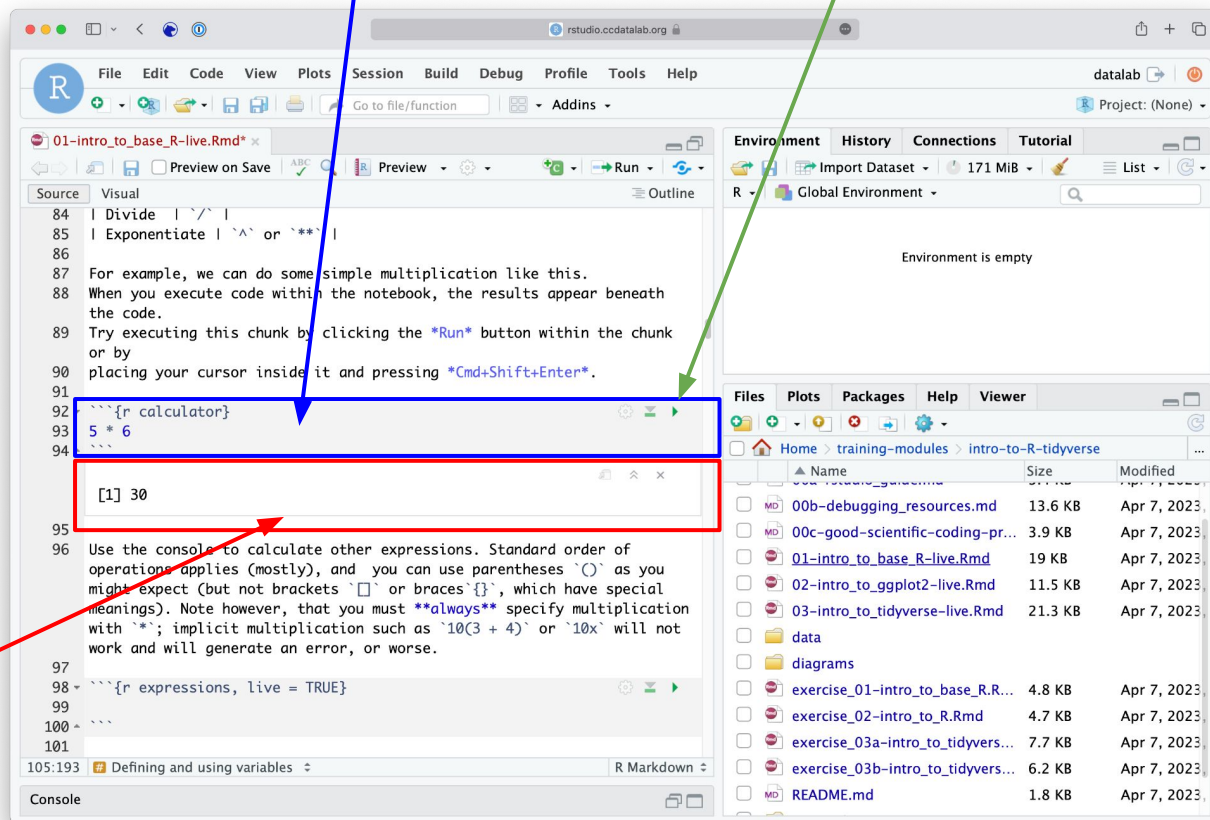


# R Notebooks

R Notebooks allow you to have files that show both your code and results

Executable code chunk

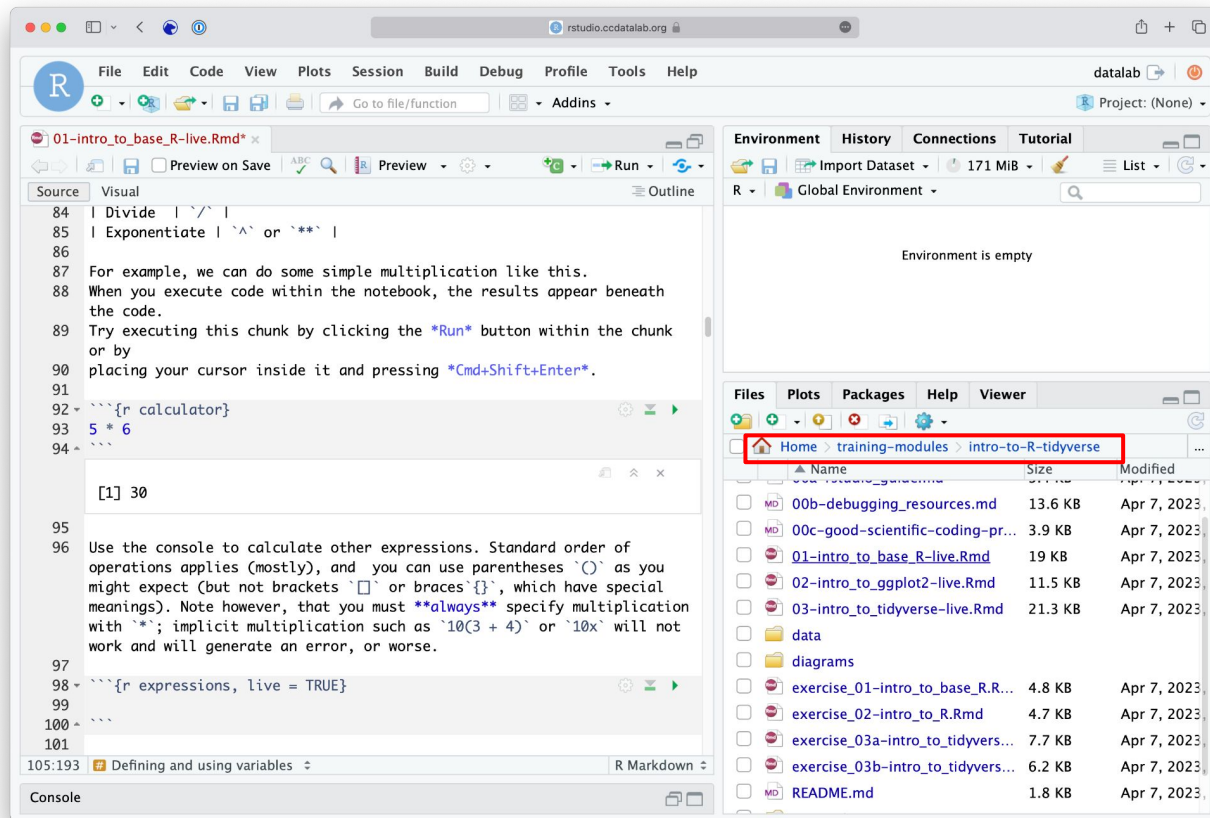
Can click here to run a code chunk



Output from above code chunk

# R Notebooks

- Code that runs in R Notebooks uses wherever the file is saved as its *current directory*
- Warning!** That may not be the directory shown in the files pane or the console!





The screenshot displays the RStudio IDE interface. The top menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The top toolbar contains icons for saving, running, and other functions. The main editor window shows a live R notebook with the following content:

```
84 | Divide | ` / ` |
85 | Exponentiate | ` ^ ` or ` ** ` |
86
87 For example, we can do some simple multiplication like this.
88 When you execute code within the notebook, the results appear beneath
89 the code.
90 Try executing this chunk by clicking the *Run* button within the chunk
91 or by
92 placing your cursor inside it and pressing *Cmd+Shift+Enter*.
93
94 ```{r calculator}
95 5 * 6
96 ```
```

The output of the code chunk is displayed below the code:

```
[1] 30
```

Below the output, there is a text block:

```
97
98 Use the console to calculate other expressions. Standard order of
99 operations applies (mostly), and you can use parentheses `()` as you
100 might expect (but not brackets `[]` or braces `{}`, which have special
101 meanings). Note however, that you must **always** specify multiplication
102 with `*`; implicit multiplication such as `10(3 + 4)` or `10x` will not
103 work and will generate an error, or worse.
104
105 ```{r expressions, live = TRUE}
106
107 ```
```

The bottom status bar shows the current line and column (105:193) and the active mode (R Markdown).

On the right side of the interface, there is a sidebar with several panels:

- Environment**: Shows the current environment (Global Environment) and memory usage (171 MiB).
- History**: Shows a list of executed code chunks.
- Connections**: Shows a list of database connections.
- Tutorial**: Shows a list of tutorial documents.
- Files**: Shows a file explorer view of the current project.

The **Files** panel shows a directory structure with the following files:

Name	Size	Modified
00b-debugging_resources.md	13.6 KB	Apr 7, 2023
00c-good-scientific-coding-pr...	3.9 KB	Apr 7, 2023
01-intro_to_base_R-live.Rmd	19 KB	Apr 7, 2023
02-intro_to_ggplot2-live.Rmd	11.5 KB	Apr 7, 2023
03-intro_to_tidyverse-live.Rmd	21.3 KB	Apr 7, 2023
exercise_03a-intro_to_tidyvers...	6.2 KB	Apr 7, 2023
exercise_03b-intro_to_tidyvers...	6.2 KB	Apr 7, 2023
README.md	1.8 KB	Apr 7, 2023

A red box highlights the **Console** icon in the bottom right corner of the sidebar, with a red arrow pointing to it. A text box next to the arrow says: "Click here to show the Console".

The screenshot displays the RStudio IDE interface. The main editor window shows an R script with a comment explaining the order of operations and a code snippet that assigns the value 5.5 to a variable `x`. The environment pane on the right shows the variable `x` with the value 5.5. The file explorer on the bottom right shows a directory structure with various R Markdown files. The console at the bottom left shows the execution of the code snippet, with the output `[1] 5.5`.

**Source** Visual [1] 30

```
95
96 Use the console to calculate other expressions. Standard order of
    operations applies (mostly), and you can use parentheses `()` as you
    might expect (but not brackets `[]` or braces `{}`, which have special
    meanings). Note however, that you must **always** specify multiplication
    with `*`; implicit multiplication such as `10(3 + 4)` or `10x` will not
    work and will generate an error, or worse.
97
98 ```{r expressions, live = TRUE}
99 x <- 5.5
100
101 x
102 ```
```

[1] 5.5

118:91 # Defining and using variables R Markdown

**Environment** History Connections Tutorial

R Global Environment

Values

Variable	Value
x	5.5

**Files** Plots Packages Help Viewer

Home > training-modules > intro-to-R-tidyverse

Name	Size	Modified
00a-reading_guide.md	3.1 KB	Apr 7, 2023
00b-debugging_resources.md	13.6 KB	Apr 7, 2023
00c-good-scientific-coding-pr...	3.9 KB	Apr 7, 2023
01-intro_to_base_R-live.Rmd	19 KB	Apr 7, 2023
02-intro_to_ggplot2-live.Rmd	11.5 KB	Apr 7, 2023
03-intro_to_tidyverse-live.Rmd	21.3 KB	Apr 7, 2023
data		
diagrams		
exercise_01-intro_to_base_R.R...	4.8 KB	Apr 7, 2023
exercise_02-intro_to_R.Rmd	4.7 KB	Apr 7, 2023
exercise_03a-intro_to_tidyvers...	7.7 KB	Apr 7, 2023
exercise_03b-intro_to_tidyvers...	6.2 KB	Apr 7, 2023
README.md	1.8 KB	Apr 7, 2023

**Console** Terminal Jobs

R 4.2.3 ~ /

```
> x <- 5.5
>
> x
[1] 5.5
> |
```

**R Console:**  
What you are actually telling R to do

The screenshot shows the RStudio IDE interface. A red rectangular box highlights the Source editor area, which contains R code and a text overlay. The text overlay reads: "R Script or Notebook: Where you are writing and editing what you will tell R or Terminal".

The Source editor shows the following code:

```
95  
96 Use the console to calculate other expressions. Standard order of  
   operations applies (mostly), and you can use parentheses `()` as you  
   might expect (but not brackets `[]` or braces `{}`, which have special  
   meanings). Note however, that you must **always** specify multiplication  
   with `*`; implicit multiplication such as `10(3 + 4)` or `10x` will not  
   work and will generate an error, or worse.  
97  
98 ```{r expressions, live = TRUE}  
99 x <- 5.5  
100  
101 x  
102 ```
```

The output of the code is shown in the console:

```
[1] 5.5
```

The console also shows the command prompt and the output of the code:

```
118:91 # Defining and using variables  
R 4.2.3 ~/  
> x <- 5.5  
>  
> x  
[1] 5.5  
> |
```

The Environment pane on the right shows the Global Environment with the variable x set to 5.5.

The Files pane on the right shows the file structure of the project, including the file 01-intro\_to\_base\_R-live.Rmd.





rstudio.ccdataab.org

File Edit Code View Plots Session Build Debug Profile Tools Help

datalab Project: (None)

01-intro\_to\_base\_R-live.Rmd\*

Source Visual [1] 30 Outline

```
95
96 Use the console to calculate other expressions. Standard order of
    operations applies (mostly), and you can use parentheses `()` as you
    might expect (but not brackets `[]` or braces `{}`, which have special
    meanings). Note however, that you must **always** specify multiplication
    with `*`; implicit multiplication such as `10(3 + 4)` or `10x` will not
    work and will generate an error, or worse.
97
98 ```{r expressions, live = TRUE}
99 x <- 5.5
100
101 x
102 ```
```

[1] 5.5

103
104

118:91 # Defining and using variables R Markdown

Console Terminal Jobs

```
R 4.2.3 ~/>
> x <- 5.5
>
> x
[1] 5.5
> |
```

Environment History Connections Tutorial

Import Dataset 172 MiB

R Global Environment

Values

x	5.5
---	-----

Other Assistance Tabs:  
Things that help you in your coding

Files Plots Packages Help Viewer

Home > training-modules > intro-to-R-tidyverse

	Name	Size	Modified
	00a-tidyverse_garden...	3.9 KB	Apr 7, 2023
	00b-debugging_resources.md	13.6 KB	Apr 7, 2023
	00c-good-scientific-coding-pr...	3.9 KB	Apr 7, 2023
	01-intro_to_base_R-live.Rmd	19 KB	Apr 7, 2023
	02-intro_to_ggplot2-live.Rmd	11.5 KB	Apr 7, 2023
	03-intro_to_tidyverse-live.Rmd	21.3 KB	Apr 7, 2023
	data		
	diagrams		
	exercise_01-intro_to_base_R.R...	4.8 KB	Apr 7, 2023
	exercise_02-intro_to_R.Rmd	4.7 KB	Apr 7, 2023
	exercise_03a-intro_to_tidyvers...	7.7 KB	Apr 7, 2023
	exercise_03b-intro_to_tidyvers...	6.2 KB	Apr 7, 2023
	README.md	1.8 KB	Apr 7, 2023

# RStudio Sessions

- On the server, R is running many times at once
  - Each user has their own “**Session**” running, with its own memory and processes
- We will usually want to start new sessions between notebooks to keep the environment clean

Log out of website



End the current session and start new session