

These materials adapted by Amelia McNamara from  
the RStudio [CC BY-SA](#) materials Introduction to R  
(2014) and [Master the Tidyverse](#) (2017).

# Introduction to R & RStudio:

## deck 01: Getting started

**Amelia McNamara**

Visiting Assistant Professor of Statistical and Data Sciences

Smith College

**January 2018**

**HELLO**

my name is

**Amelia**

**@AmeliaMN**

**HELLO**

my name is

**Katie**

**Katie Leap**

**HELLO**

my name is

**Kelly**

**Kelly O'Briant**

[@b23kelly](#) [@RLadiesDC](#)

# Your turn

Introduce yourself to your nearest 2-4 neighbors. Tell them:

- Who you are
- What you do with data



# Schedule

snacks and lunch provided!

9:00 - 10:30

Introduction

10:30 - 11:00

Morning break (Seaport Foyer)

11:00 - 12:00

Visualization

12:00 - 1:00

Lunch (Grand Ballroom)

1:00 - 3:00

Data types

3:00 - 3:30

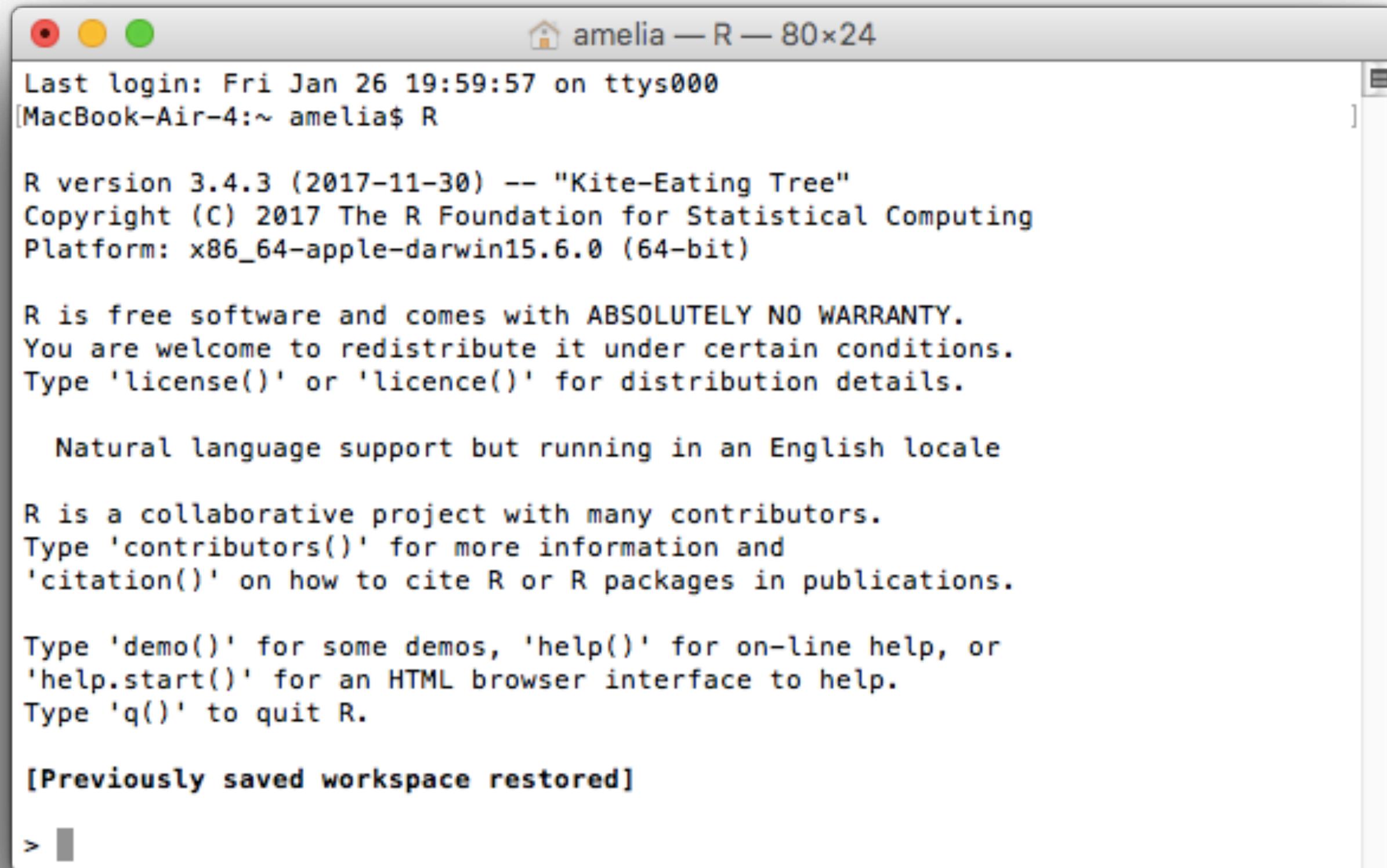
Afternoon break (Seaport Foyer)

3:30 - 5:00

Syntax and subsetting



# R: a computer programming language



The image shows a screenshot of an R terminal window titled "amelia — R — 80x24". The window displays the standard R startup message, which includes the last login information, the R version and copyright details, the platform, the warranty notice, natural language support information, collaborative project details, and help instructions. The message concludes with a note about a previously saved workspace and a prompt at the bottom.

```
Last login: Fri Jan 26 19:59:57 on ttys000
[MacBook-Air-4:~ amelia$ R

R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (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

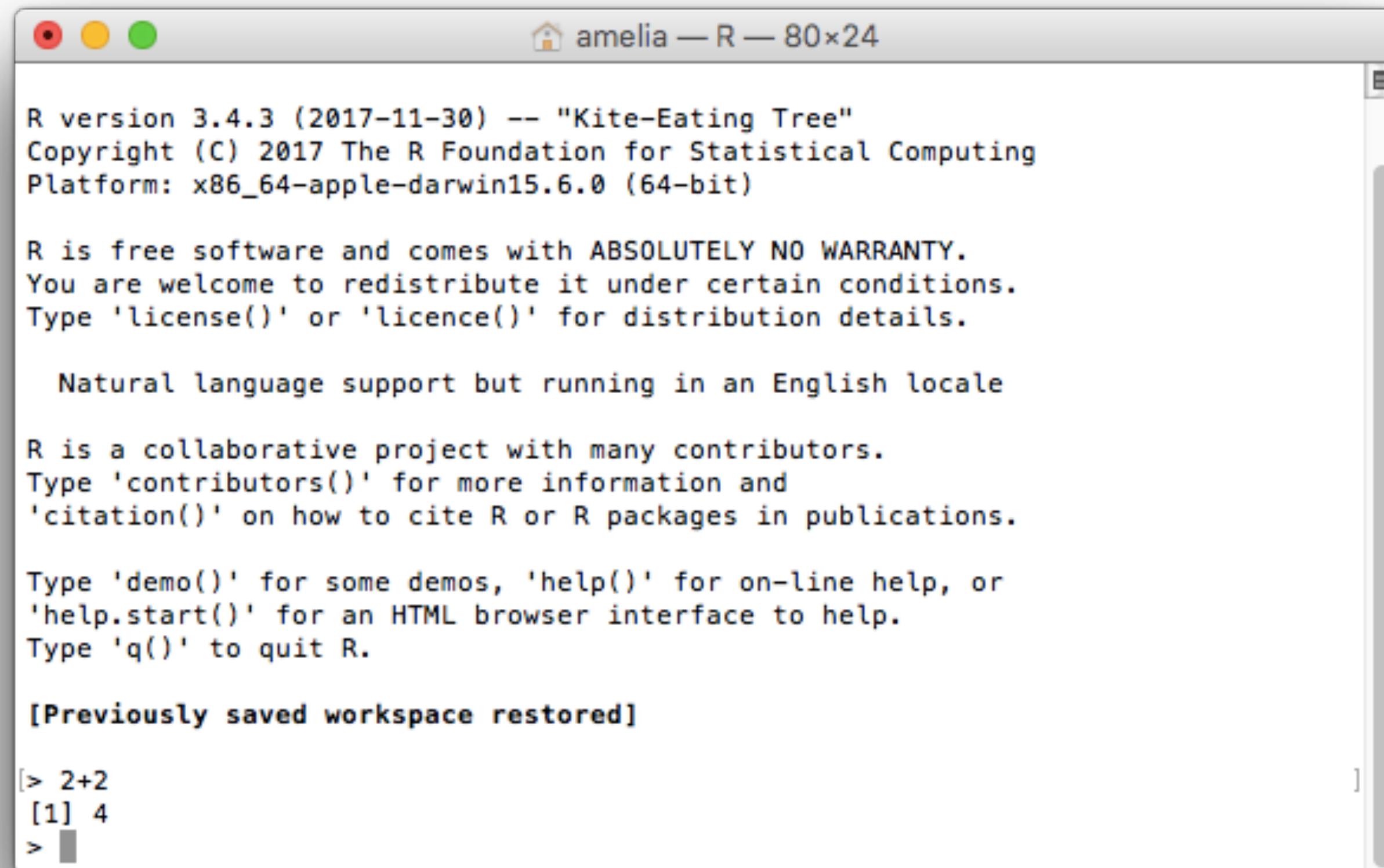
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

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.

[Previously saved workspace restored]

> ]
```

# R: a computer programming language



```
R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (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.

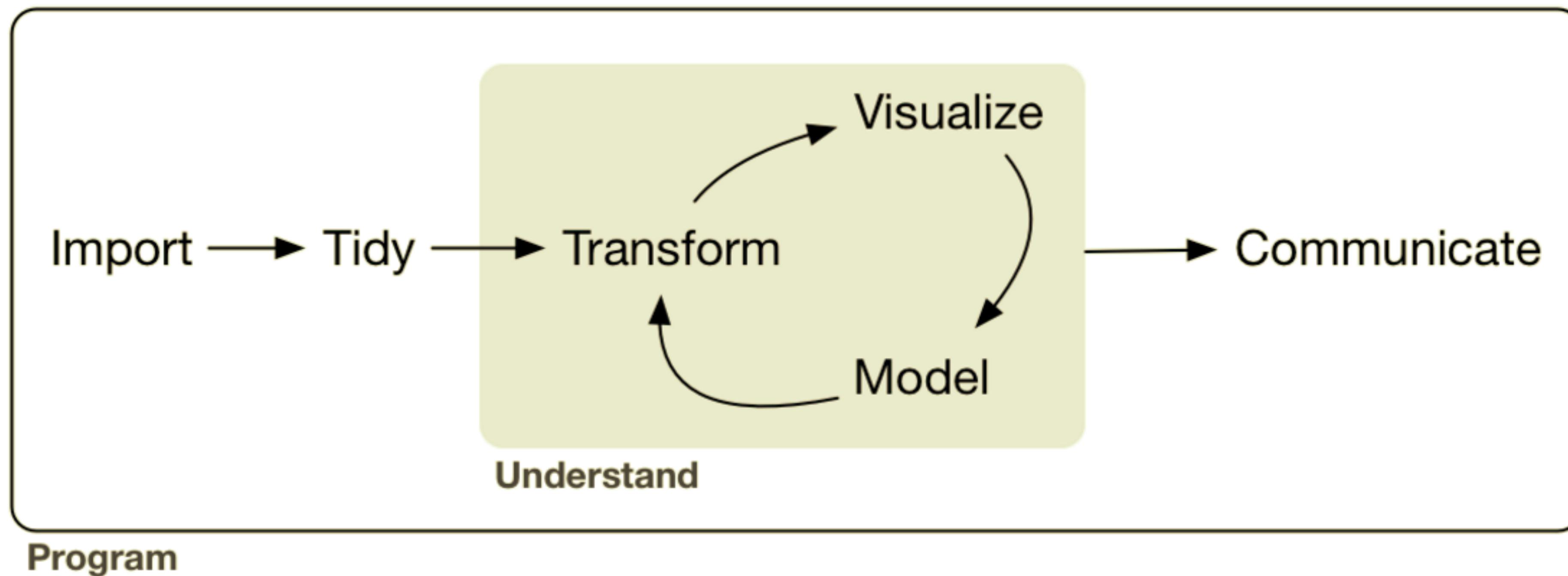
[Previously saved workspace restored]

[> 2+2
[1] 4
> ]
```

# R: a computer programming language

1. Descends from S, Bell Labs
2. Evolved in university environment
3. Full language
4. ...but can be used as a simple application
5. Designed for use with data

# R: designed for data



From *R for Data Science* by Hadley Wickham and Garrett Grolemund.

```
> bechdel
```

```
# A tibble: 1,794 x 15
```

	year	imdb	title	test	clean_test	binary	budget	domgross	intgross
	<int>	<chr>	<chr>	<chr>	<fctr>	<chr>	<int>	<dbl>	<dbl>
1	2009	tt1003034	Perrier's Bounty	nowomen	nowomen	FAIL	6600000	828	828
2	2008	tt1226681	Pontypool	nowomen-disagree	nowomen	FAIL	1500000	3865	31916
3	2012	tt1874789	Supporting Characters	men	men	FAIL	60000	4917	4917
4	2007	tt0861739	Tropa de Elite	ok-disagree	ok	PASS	6537890	8744	14319195
5	2007	tt0964587	St. Trinian's	ok	ok	PASS	11400000	15000	22446568
6	2011	tt1535616	The Divide	ok	ok	PASS	3000000	18000	18000
7	1996	tt0115591	August	dubious	dubious	FAIL	3400000	12636	12636
8	2006	tt0783238	The Dead Girl	ok	ok	PASS	3300000	19875	19875
9	2005	tt0342272	Dear Wendy	notalk	notalk	FAIL	8000000	23106	446438
10	2011	tt1788391	Kill List	dubious	dubious	FAIL	800000	29063	462206

```
# ... with 1,784 more rows, and 6 more variables: code <chr>, budget_2013 <int>, domgross_2013 <dbl>,
```

```
# intgross_2013 <dbl>, period_code <int>, decade_code <int>
```

```
> bechdel %>% skim(domgross_2013)
```

Skim summary statistics

n obs: 1794

n variables: 15

Variable type: numeric

	variable	missing	complete	n	mean	sd	p25	median	p75	hist
domgross_2013		18	1776	1794	9.5e+07	1.3e+08	2.1e+07	5.6e+07	1.2e+08	█

```
> bechdel %>% skim(clean_test)
```

Skim summary statistics

n obs: 1794

n variables: 15

Variable type: factor

	variable	missing	complete	n	n_unique	top_counts	ordered
clean_test		0	1794	1794	5	ok: 803, not: 514, men: 194, dub: 142	FALSE

```
> qplot(budget_2013, domgross_2013, data=bechdel, color = binary)
```



```
> lm(domgross_2013~budget_2013, data=bechdel)
```

Call:

```
lm(formula = domgross_2013 ~ budget_2013, data = bechdel)
```

Residuals:

	Min	1Q	Median	3Q	Max
-256686756	-47529500	-27186696	15143559	1690886212	

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	3.615e+07	3.782e+06	9.559	<2e-16 ***
budget_2013	1.056e+00	4.823e-02	21.896	<2e-16 ***

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11180000 on 1774 degrees of freedom  
(18 observations deleted due to missingness)

Multiple R-squared: 0.2128, Adjusted R-squared: 0.2123  
F-statistic: 479.4 on 1 and 1774 DF, p-value: < 2.2e-16

## Movie explorer

**Filter**

Minimum number of reviews on Rotten Tomatoes

10 40 70 100 130 160 190 220 250 280 300

Year released

1,940 1,948 1,956 1,964 1,972 1,980 1,988 1,996 2,004 2,012 2,014

Minimum number of Oscar wins (all categories)

0 1 2 3 4

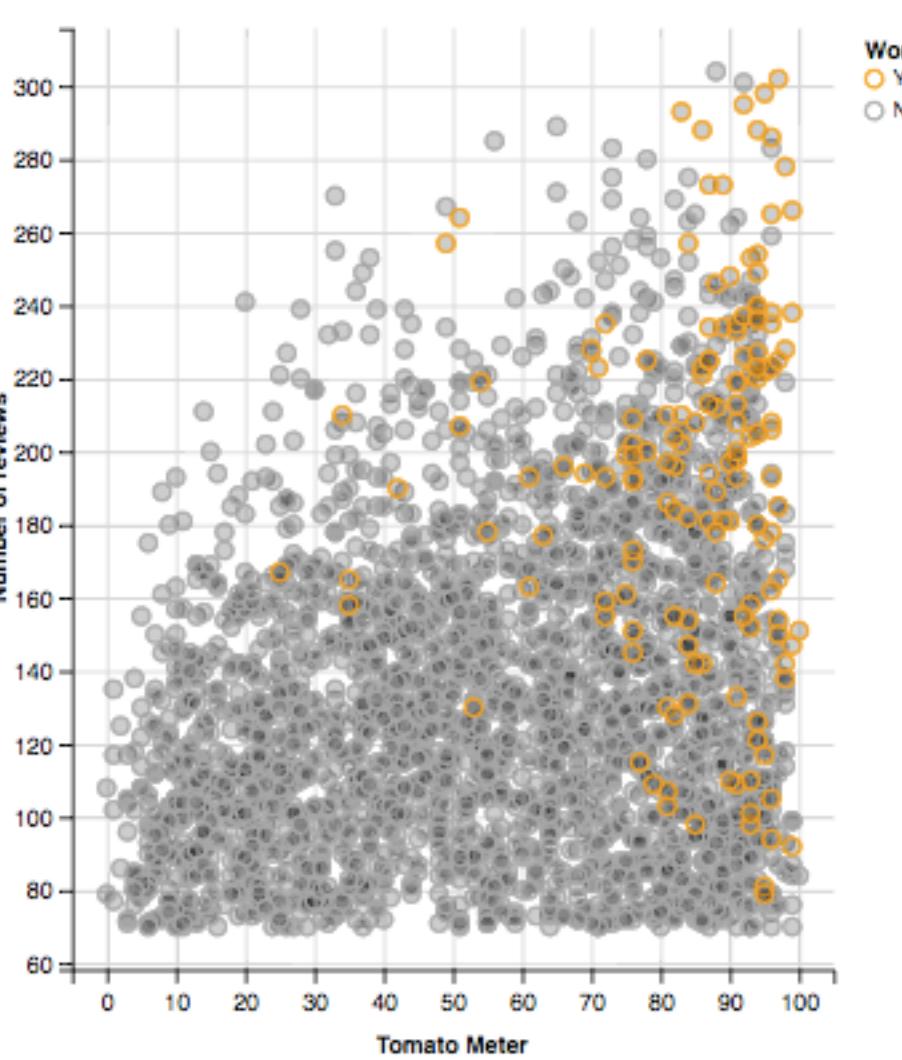
Dollars at Box Office (millions)

0 80 160 320 400 480 560 640 720 800

Genre (a movie can have multiple genres)

Director name contains (e.g., Miyazaki)

Cast names contains (e.g. Tom Hanks)



Number of movies selected:  
2758

X-axis variable

Y-axis variable

Note: The Tomato Meter is the proportion of positive reviews (as judged by the Rotten Tomatoes staff), and the Numeric rating is a normalized 1-10 score of those reviews which have star ratings (for example, 3 out of 4 stars).

The screenshot shows a web application titled "intRo" running on a Mac OS X system. The browser window has a tab labeled "intRo" and a URL bar showing "www.intro-stats.com". The application interface is divided into several sections:

- Left Sidebar (Data):** Contains links for Sources, Transform, Summaries, Graphical, Numerical, Inference, Contingency, Regression, and T test. The "Sources" link is highlighted in orange.
- Central Panel (Choose Dataset):** A dropdown menu is set to "MPG". Below it are options for "Upload Dataset", "Random Subset", "Save Subset", "Reset Data", and a "Download Data" button.
- Right Panel (Data View):** A table titled "Categorical Variable Numeric Variable" displays 10 entries from the MPG dataset. The columns are: manufacturer, model, displ, year, cyl, trans, drv, cty, hwy, fl. The data rows are:

manufacturer	model	displ	year	cyl	trans	drv	cty	hwy	fl
audi	a4	1.8	1999	4	auto(l5)	f	18	29	p
audi	a4	1.8	1999	4	manual(m5)	f	21	29	p
audi	a4	2.0	2008	4	manual(m6)	f	20	31	p
audi	a4	2.0	2008	4	auto(av)	f	21	30	p
audi	a4	2.8	1999	6	auto(l5)	f	16	26	p
audi	a4	2.8	1999	6	manual(m5)	f	18	26	p
audi	a4	3.1	2008	6	auto(av)	f	18	27	p
audi	a4 quattro	1.8	1999	4	manual(m5)	4	18	26	p
audi	a4 quattro	1.8	1999	4	auto(l5)	4	16	25	p
audi	a4 quattro	2.0	2008	4	manual(m6)	4	20	28	p
- Bottom Navigation:** Buttons for "Previous", page numbers 1 through 5, and "Next".

<http://www.intro-stats.com/>



# RStudio: a software program

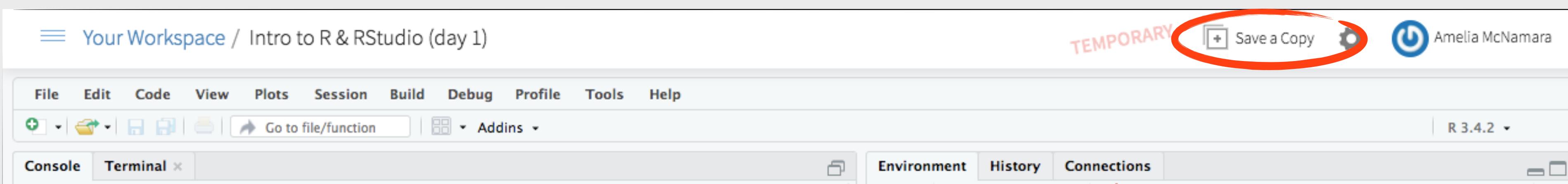
1. like Microsoft Word, Excel, etc.
2. built to help you write R code, run R code, and analyze data with R
3. text editor, version control, keyboard shortcuts, debugging tools, and much more

# Your turn

It's time to log in to RStudio, if you haven't already.  
Go to [http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)

Make an account

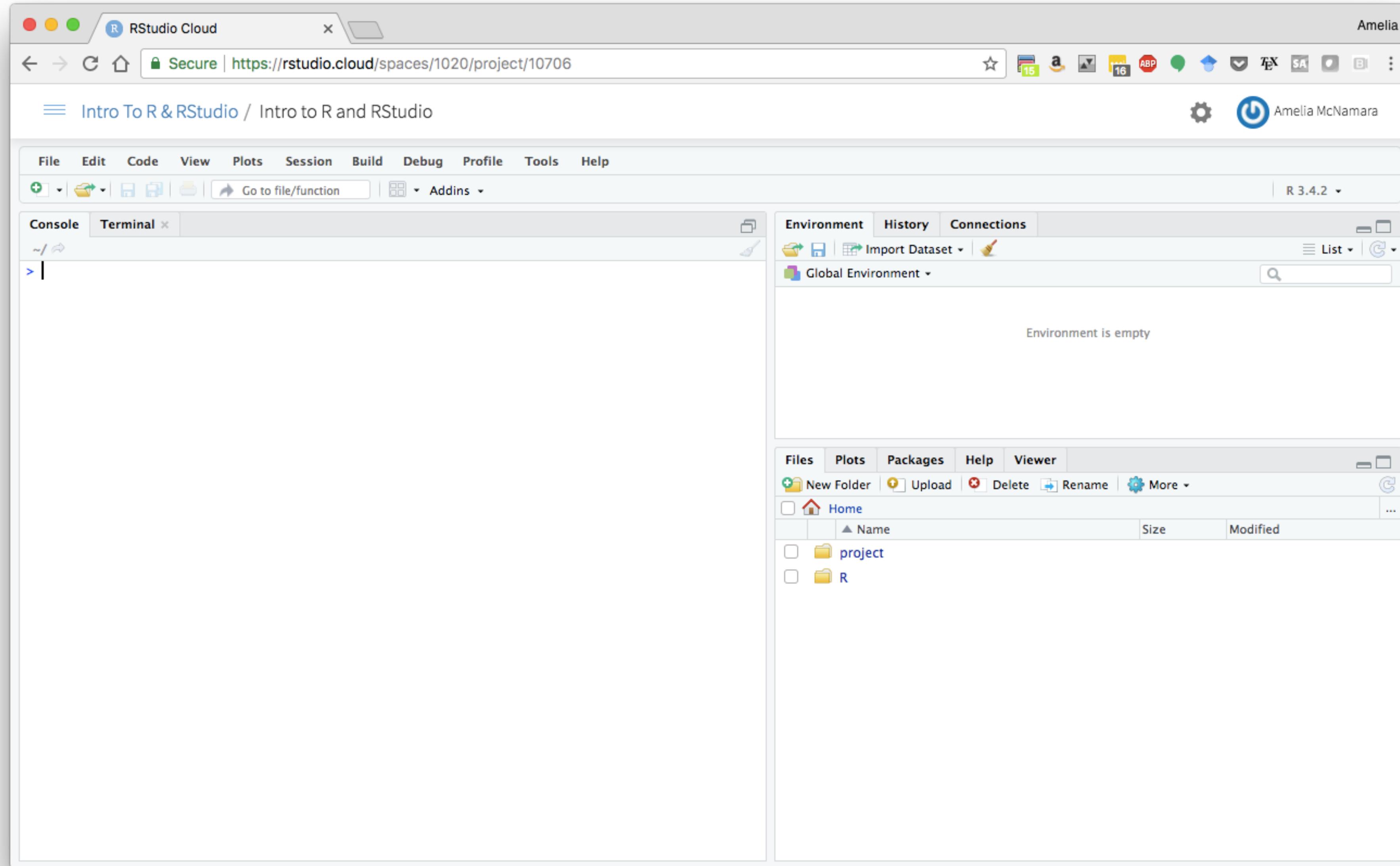
Click Save a Copy



04:00

# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



# RStudio

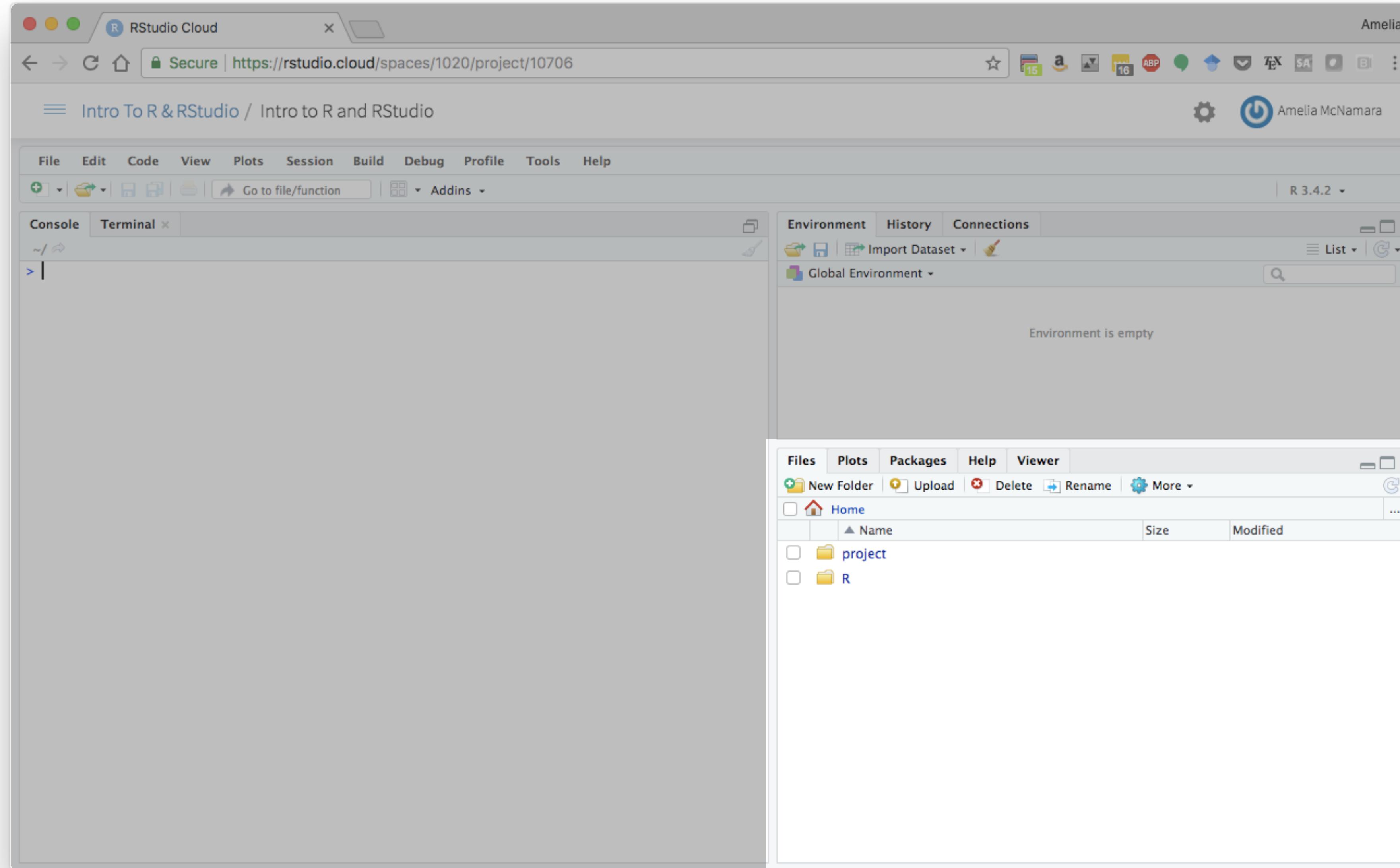
[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)

The screenshot shows the RStudio Cloud interface. The top navigation bar includes tabs for File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. The main area features several panes: a left pane labeled "Intro To R & RStudio / Intro to R and RStudio" containing the text "The console gives you a place to execute commands written in R"; a top-right pane titled "Environment" showing "Global Environment" with "Environment is empty"; a bottom-right pane titled "Files" showing a directory structure with "Home" containing "project" and "R"; and a bottom-left pane titled "Plots" which is currently empty. The browser address bar shows a secure connection to <https://rstudio.cloud/spaces/1020/project/10706>. The overall interface is clean and modern, designed for collaborative data analysis.

The console gives you a place to execute commands written in R

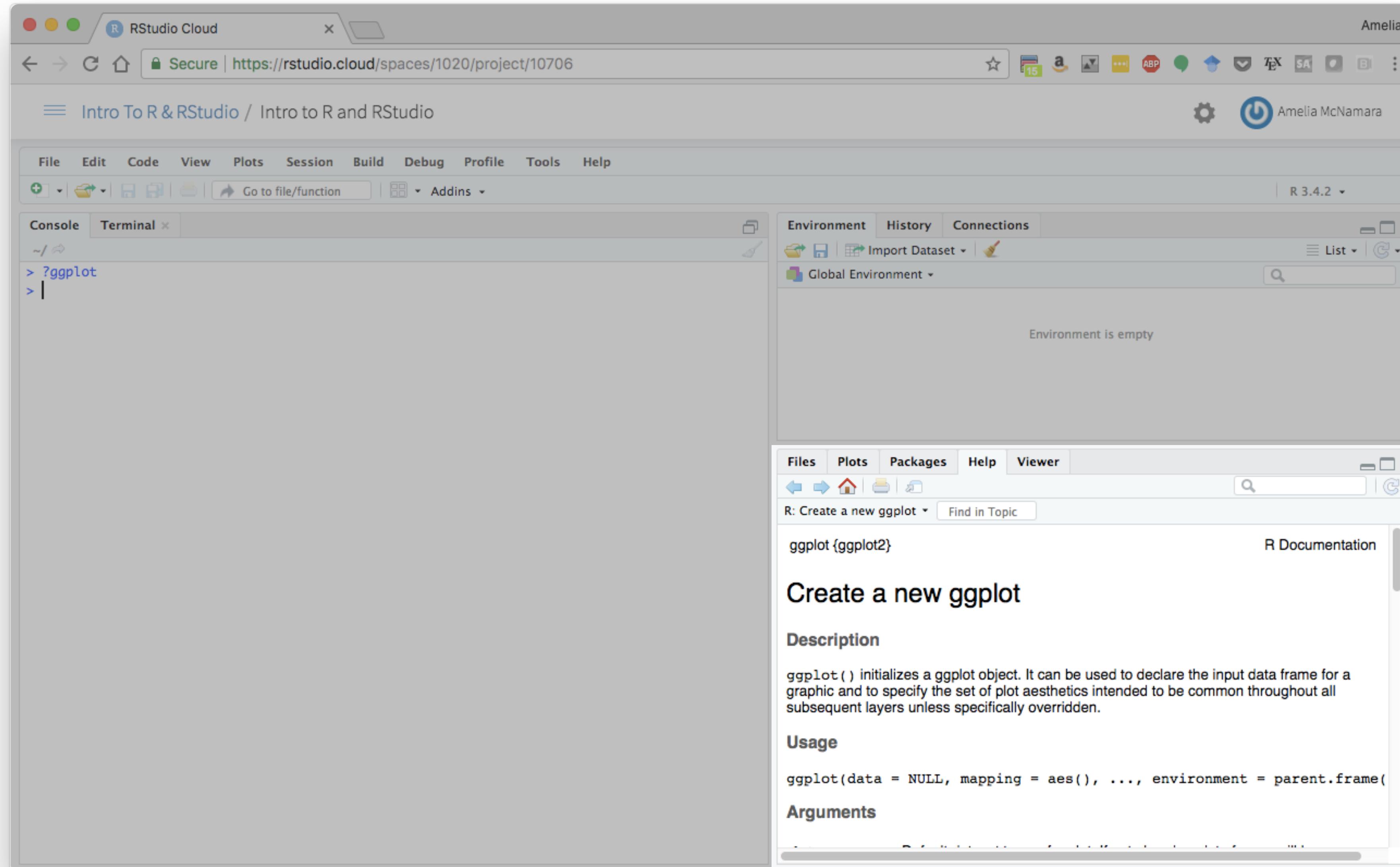
# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



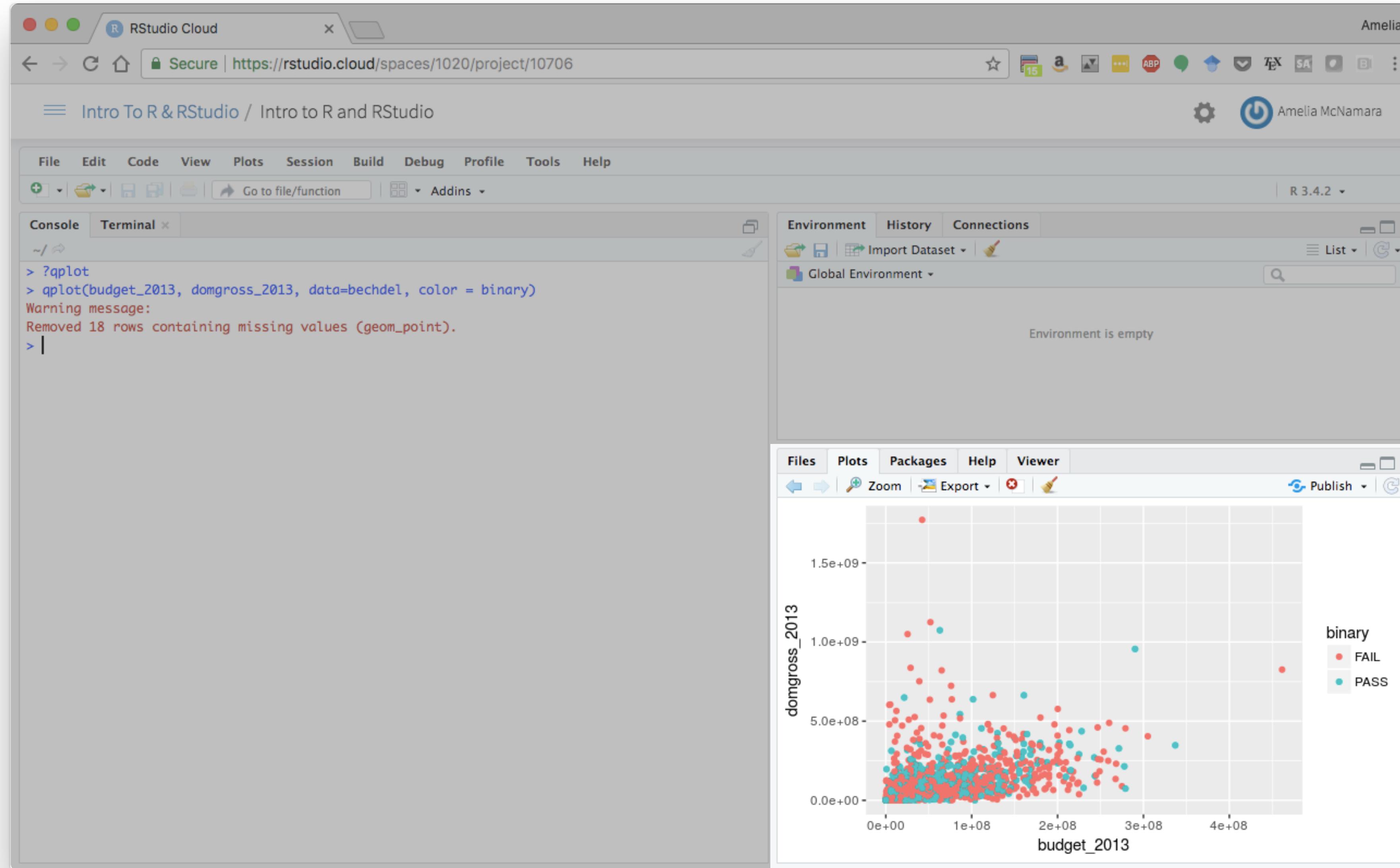
# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



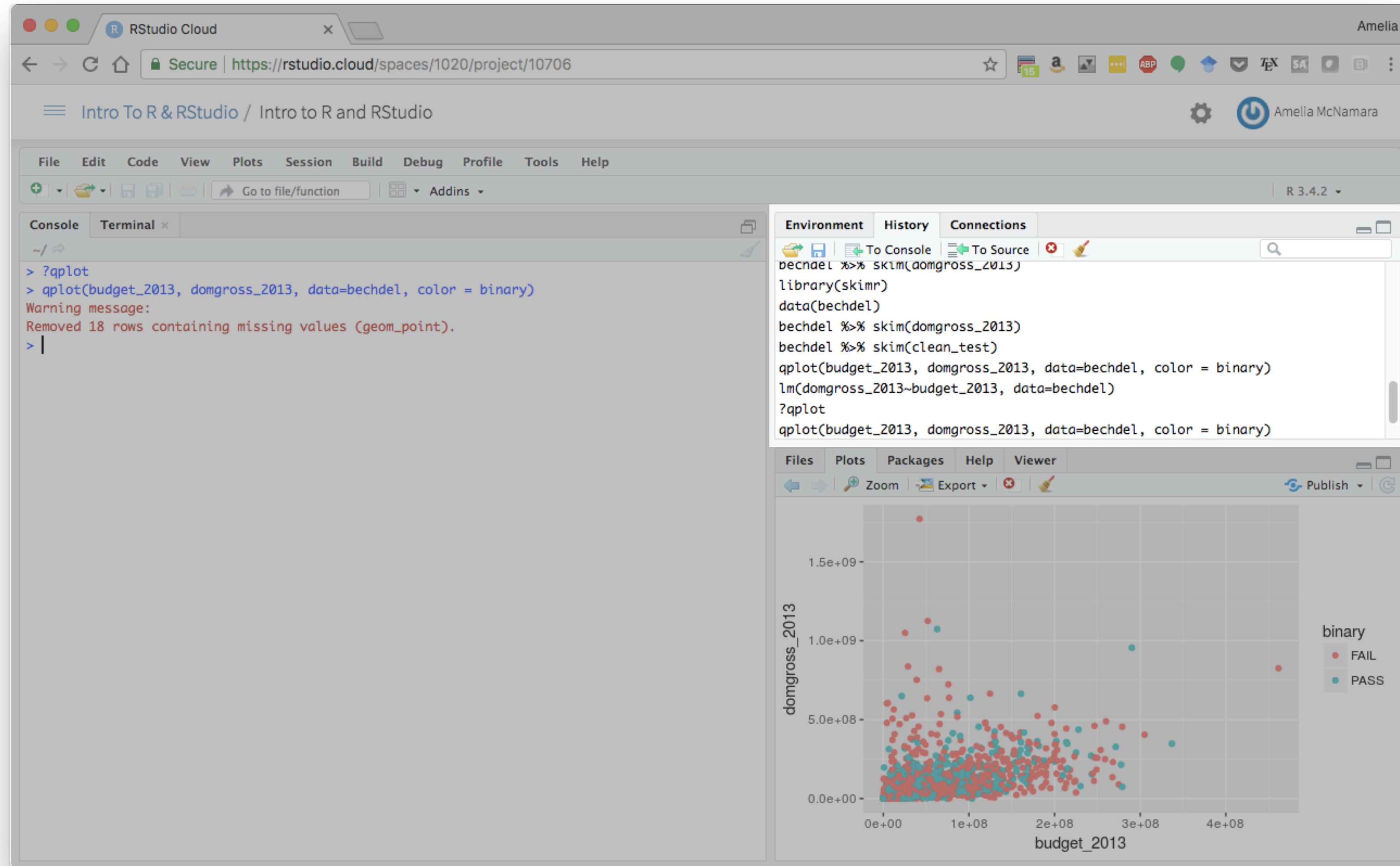
# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



# RStudio

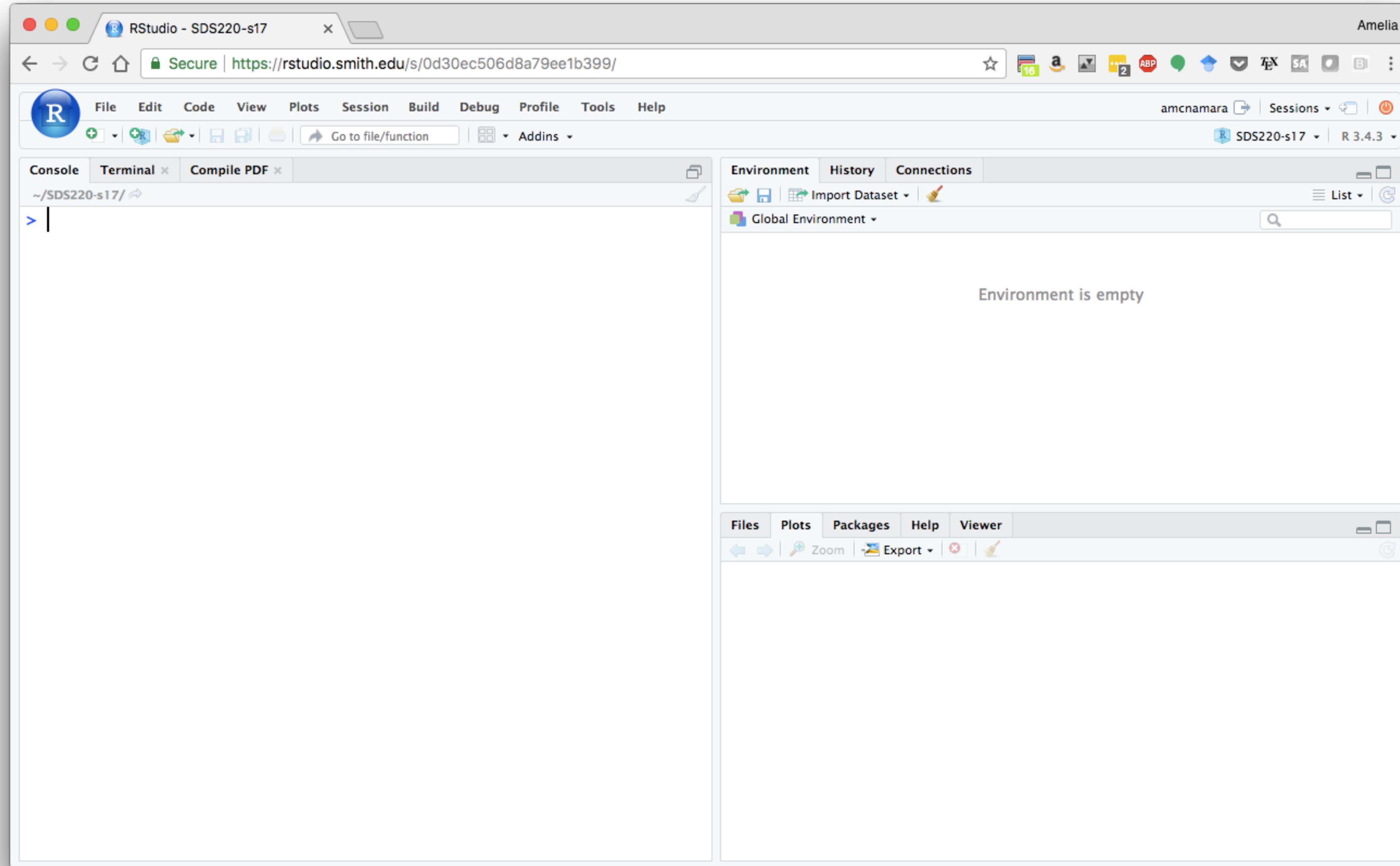
[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)

The screenshot shows the RStudio Cloud interface with the following components:

- Code Editor:** An R Markdown file titled "Untitled1" is open. The code includes setup code for knitr and a section for R Markdown. The text area contains a note about R Markdown and a warning message about missing values.
- Console:** The R console window shows the command `?qplot` followed by `qplot(budget\_2013, domgross\_2013, data=bechdel, color = binary)`. A warning message indicates that 18 rows containing missing values were removed.
- Environment:** The global environment is currently empty.
- Plots:** A scatter plot titled "domgross\_2013" is displayed, showing the relationship between budget\_2013 (x-axis, ranging from 0e+00 to 4e+08) and domgross\_2013 (y-axis, ranging from 0.0e+00 to 1.5e+09). The plot uses a color legend for the "binary" variable, where red dots represent "FAIL" and teal dots represent "PASS".

# RStudio

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)

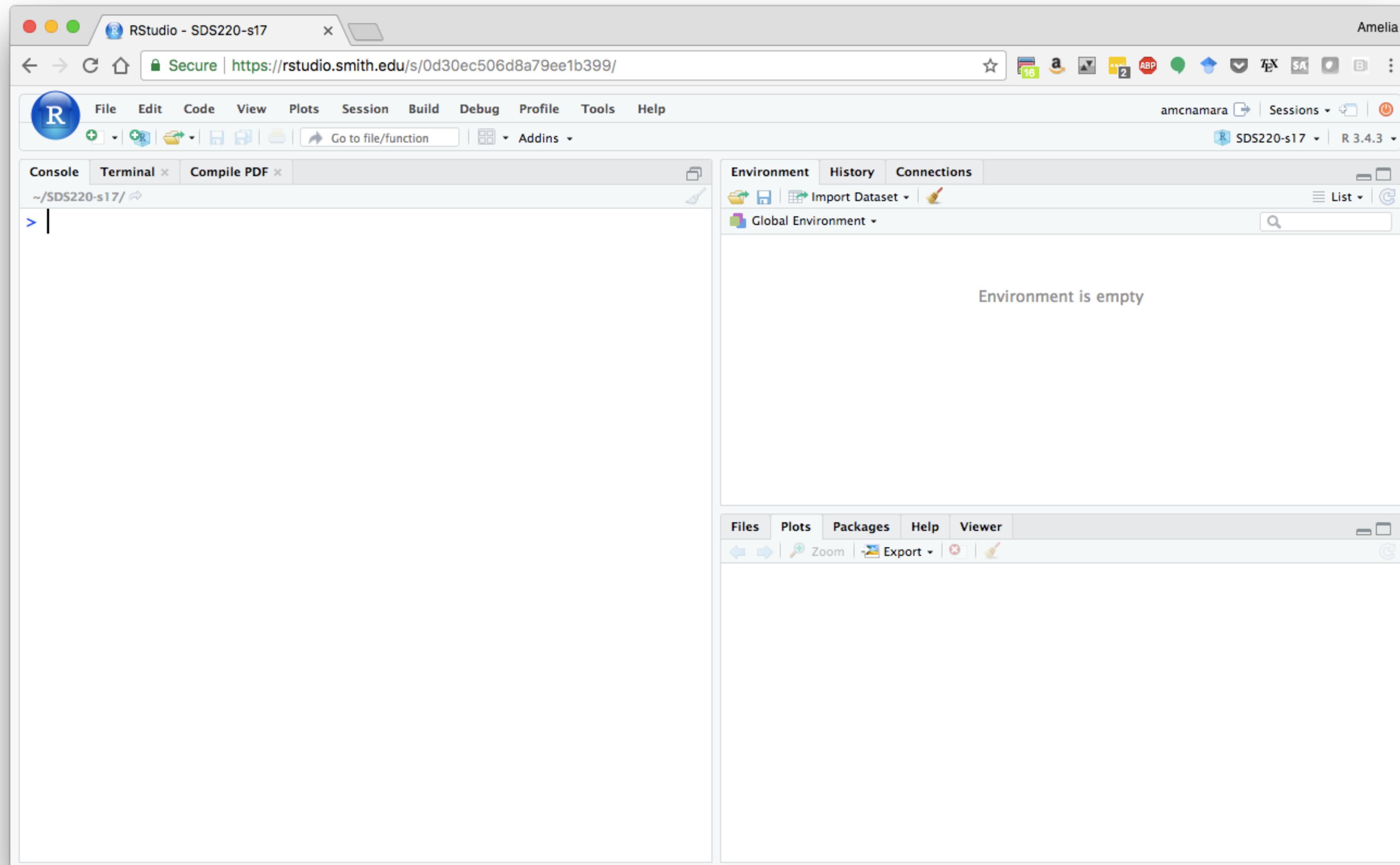


# RStudio: ways to use

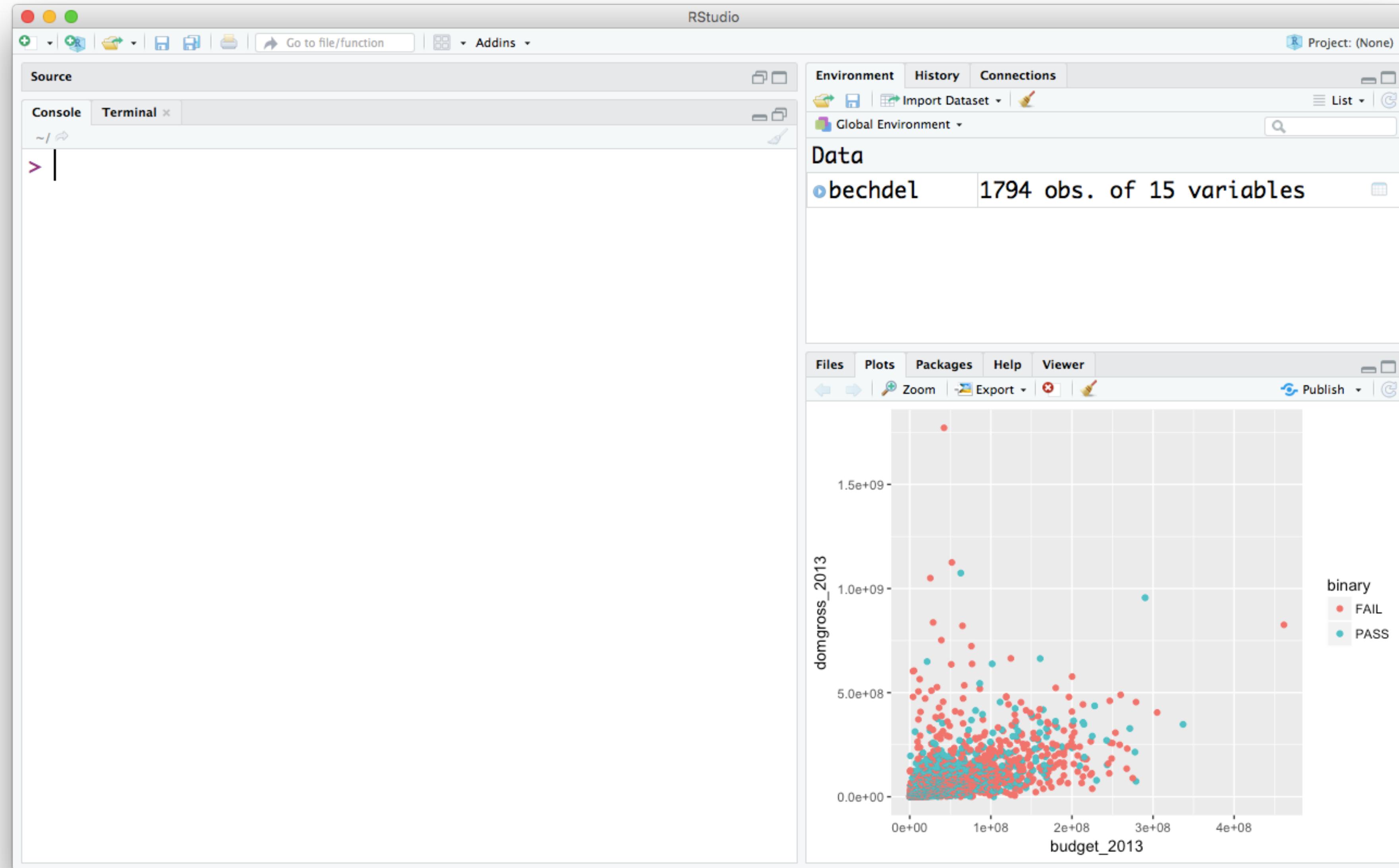
We're using RStudio Cloud, which allows you to log in through a web browser and do your work there.

But, there are other versions of RStudio.

# RStudio: server edition



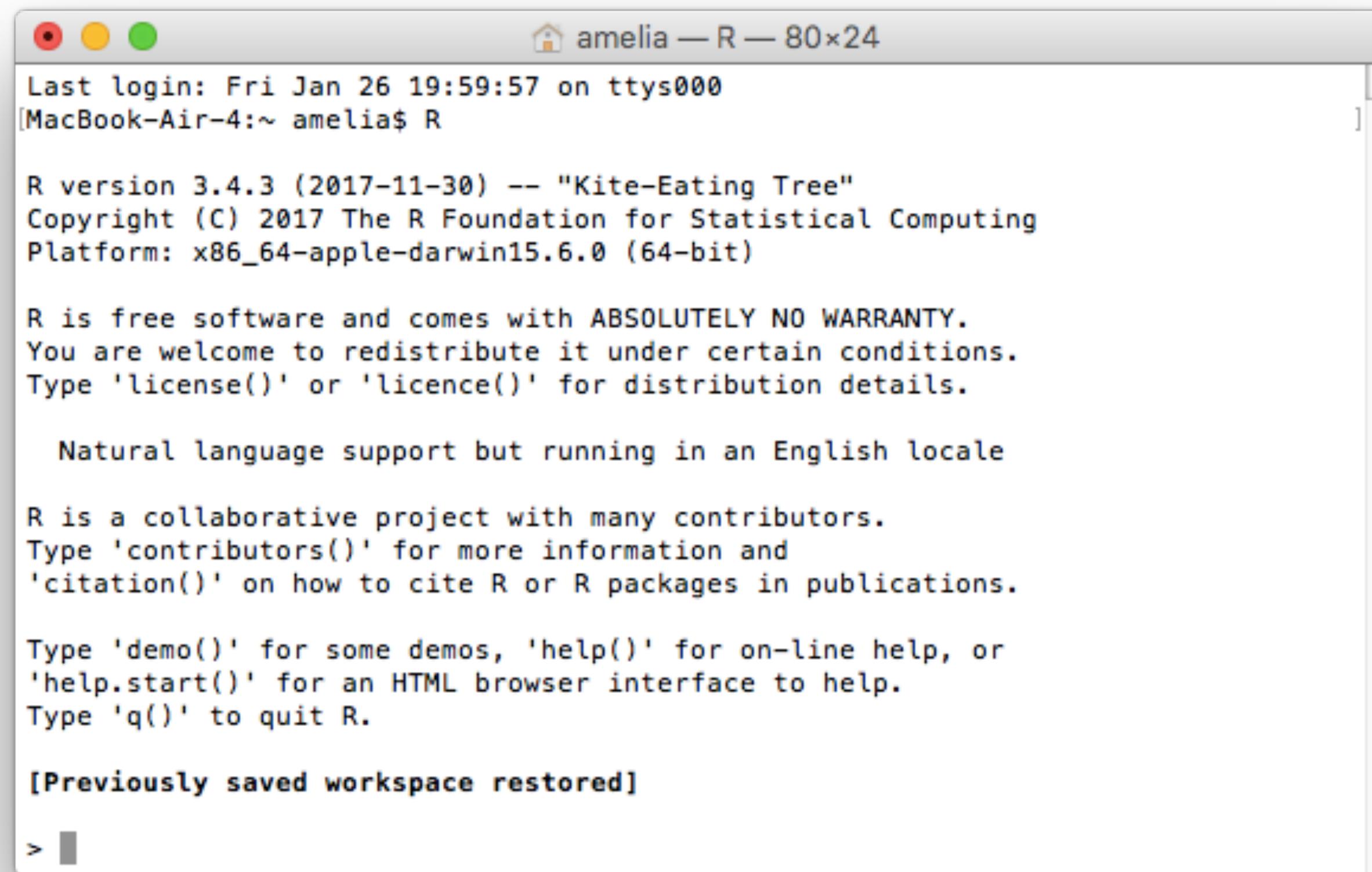
# RStudio: desktop edition



# Transferability

Everything you learn here will work in any version  
of RStudio

It will even work in the basic console version of R



```
Last login: Fri Jan 26 19:59:57 on ttys000
[MacBook-Air-4:~ amelia$ R

R version 3.4.3 (2017-11-30) -- "Kite-Eating Tree"
Copyright (C) 2017 The R Foundation for Statistical Computing
Platform: x86_64-apple-darwin15.6.0 (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.

[Previously saved workspace restored]

> ]
```

# Tips

Learning things can be frustrating!

Ask questions!

Practice!

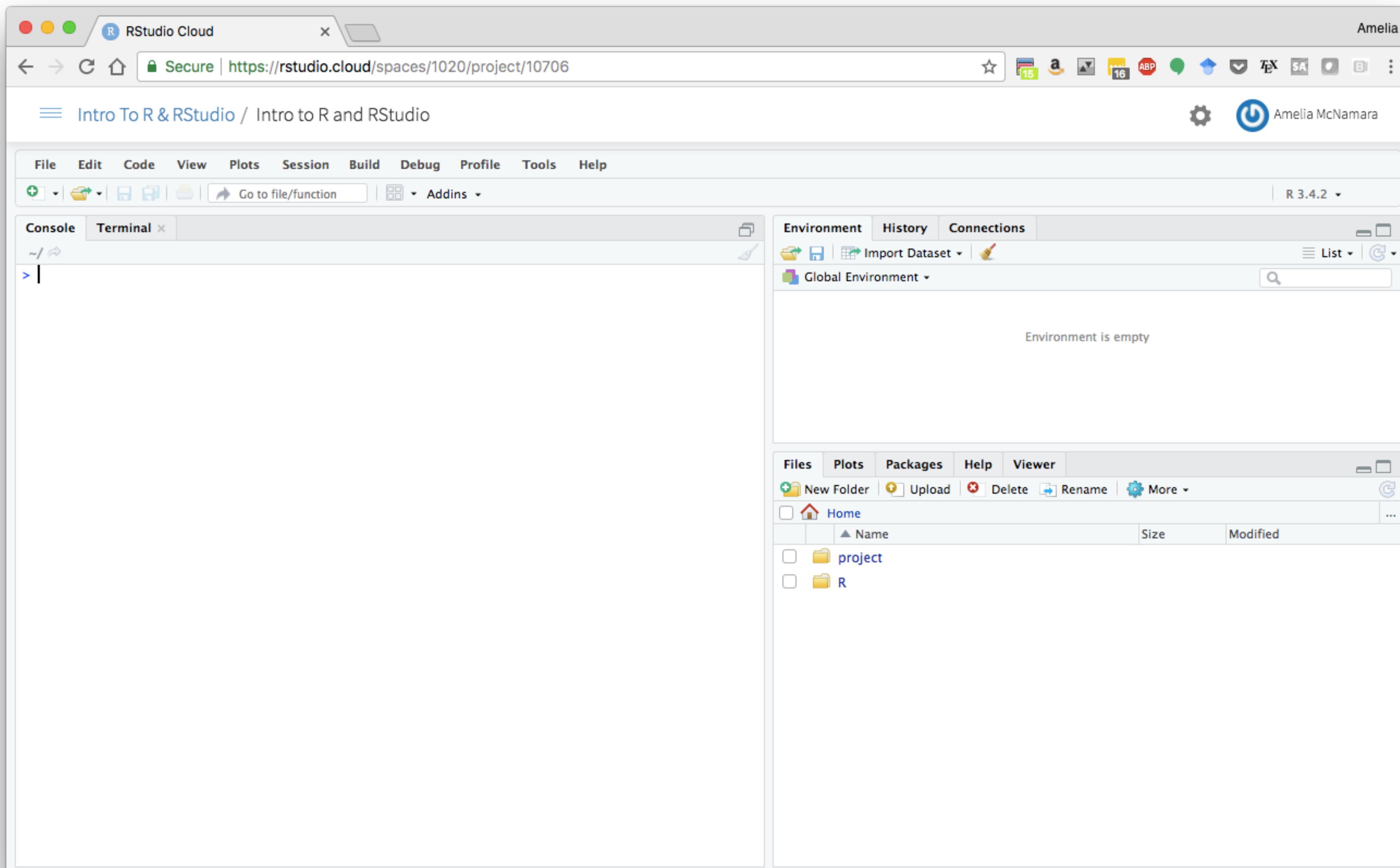
Seriously: practice!

And practice consciously: make a prediction,  
then test it, then reflect.

# RStudio

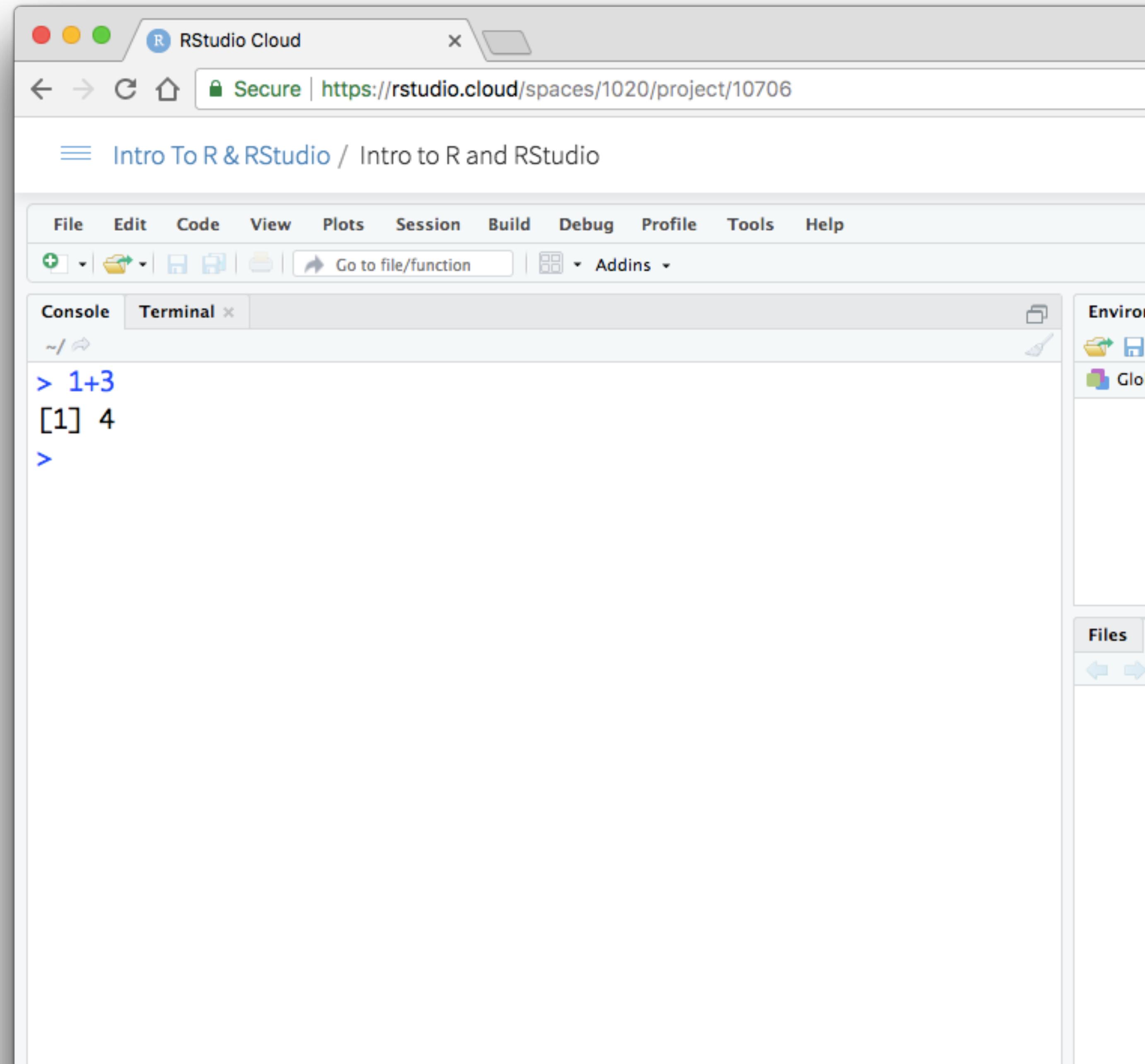
# Getting started

[http://bit.ly/R\\_and\\_RStudio\\_day1](http://bit.ly/R_and_RStudio_day1)



The console gives you a place to execute commands written in R

Type commands on the line that begins with a > sign (known as the prompt)

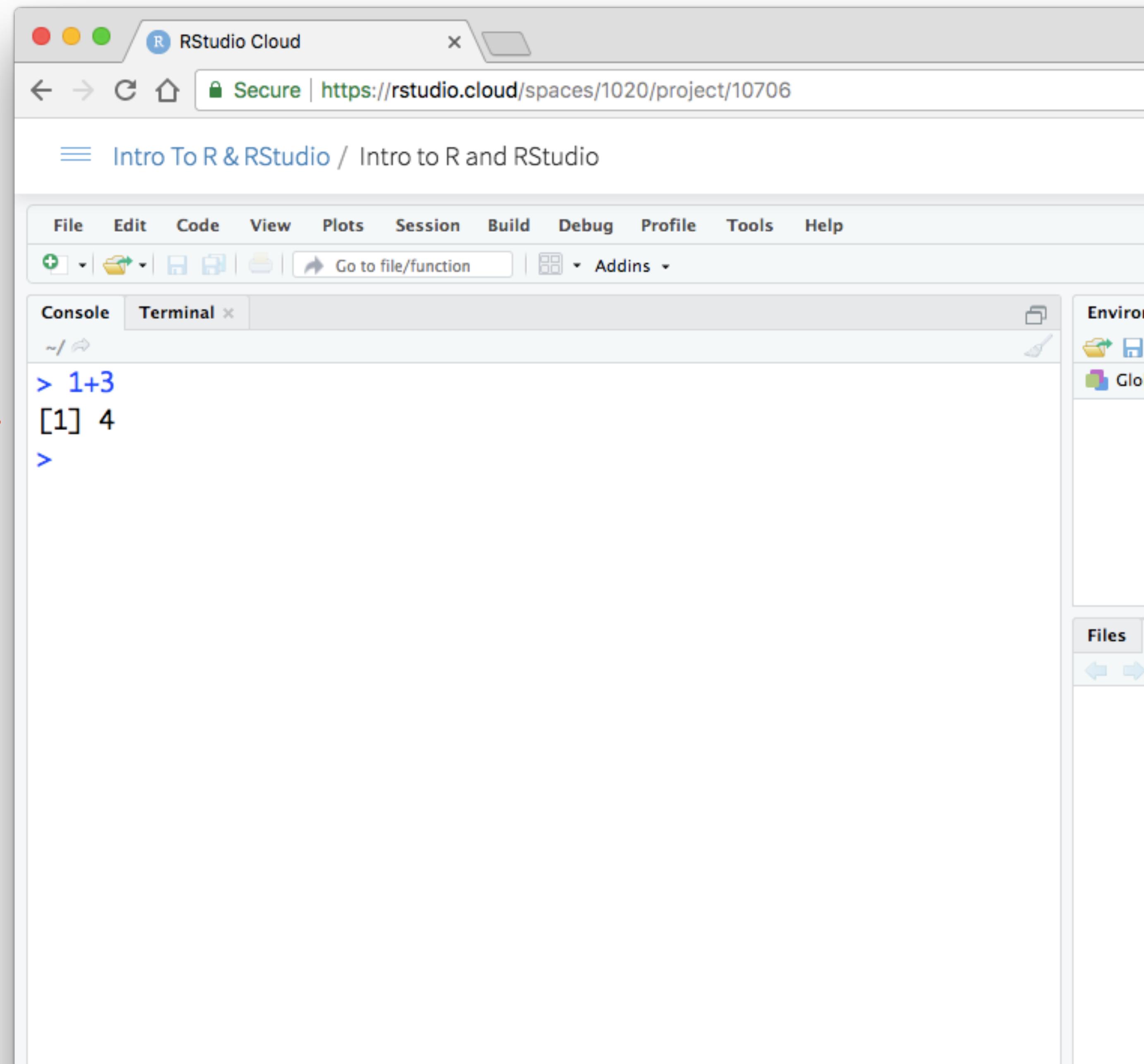


A screenshot of the RStudio Cloud interface. The title bar says "RStudio Cloud". The address bar shows a secure connection to "https://rstudio.cloud/spaces/1020/project/10706". The main window title is "Intro To R & RStudio / Intro to R and RStudio". The menu bar includes File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. Below the menu is a toolbar with various icons. The main area has tabs for "Console" and "Terminal", with "Console" selected. The console window displays the following text:  
> 1+3  
[1] 4  
>

# Output

When you hit enter,  
RStudio will run  
your command and  
display any output  
below it

Output →  
New prompt →



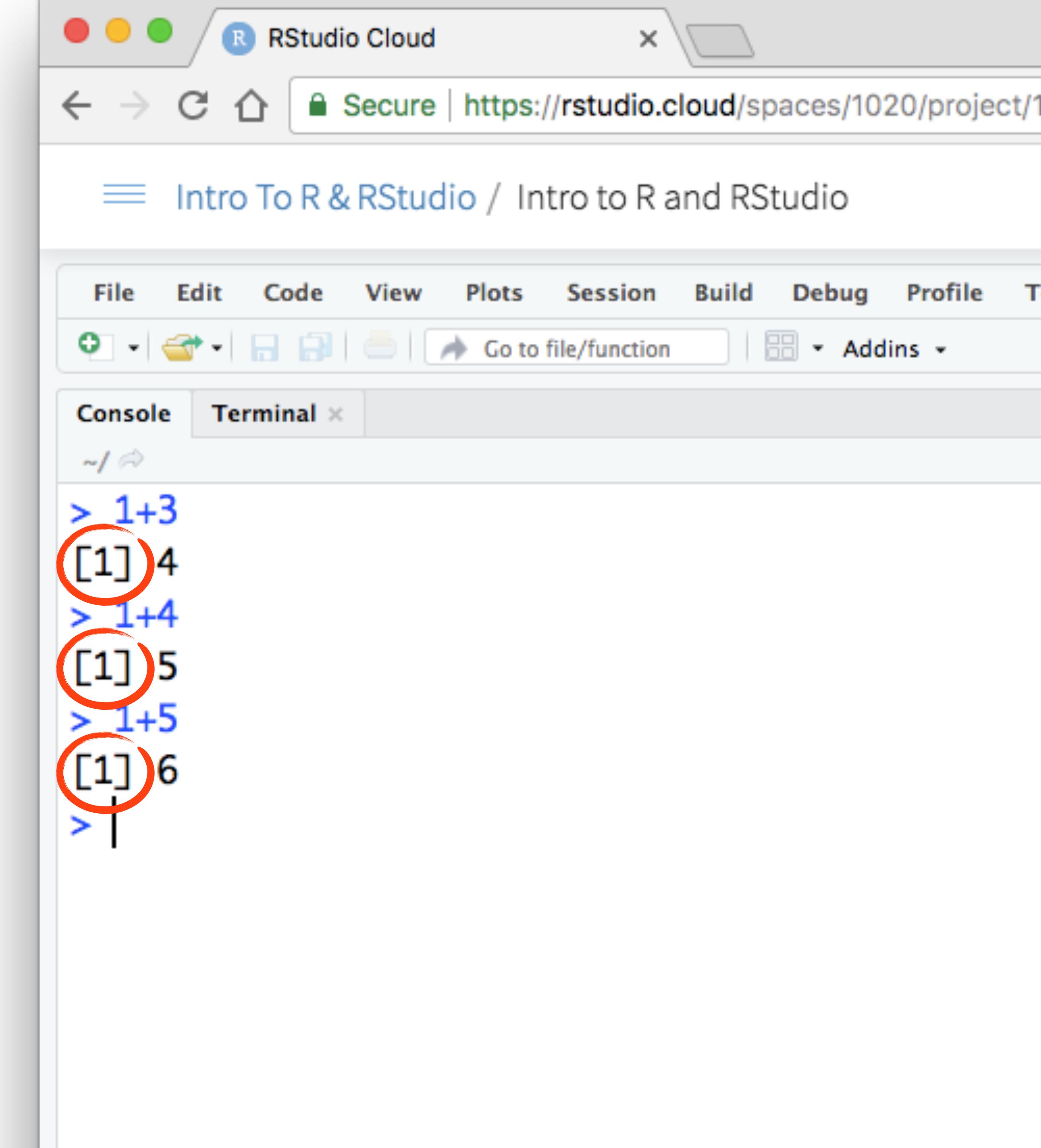
The screenshot shows the RStudio Cloud interface. The title bar says "RStudio Cloud". The address bar is "Secure | https://rstudio.cloud/spaces/1020/project/10706". The main area shows a "Console" tab with the following text:  
> 1+3  
[1] 4  
>  
The "Console" tab is highlighted in blue. To the right of the console are several tabs: "Console", "Terminal", "Plots", "Session", "Build", "Debug", "Profile", "Tools", and "Help". Below the tabs is a toolbar with icons for file operations like "New", "Open", "Save", etc. On the far right, there are panels for "Environment", "Files", and "Global Options".

[1]

R displays an index next to the output.

Just ignore this.

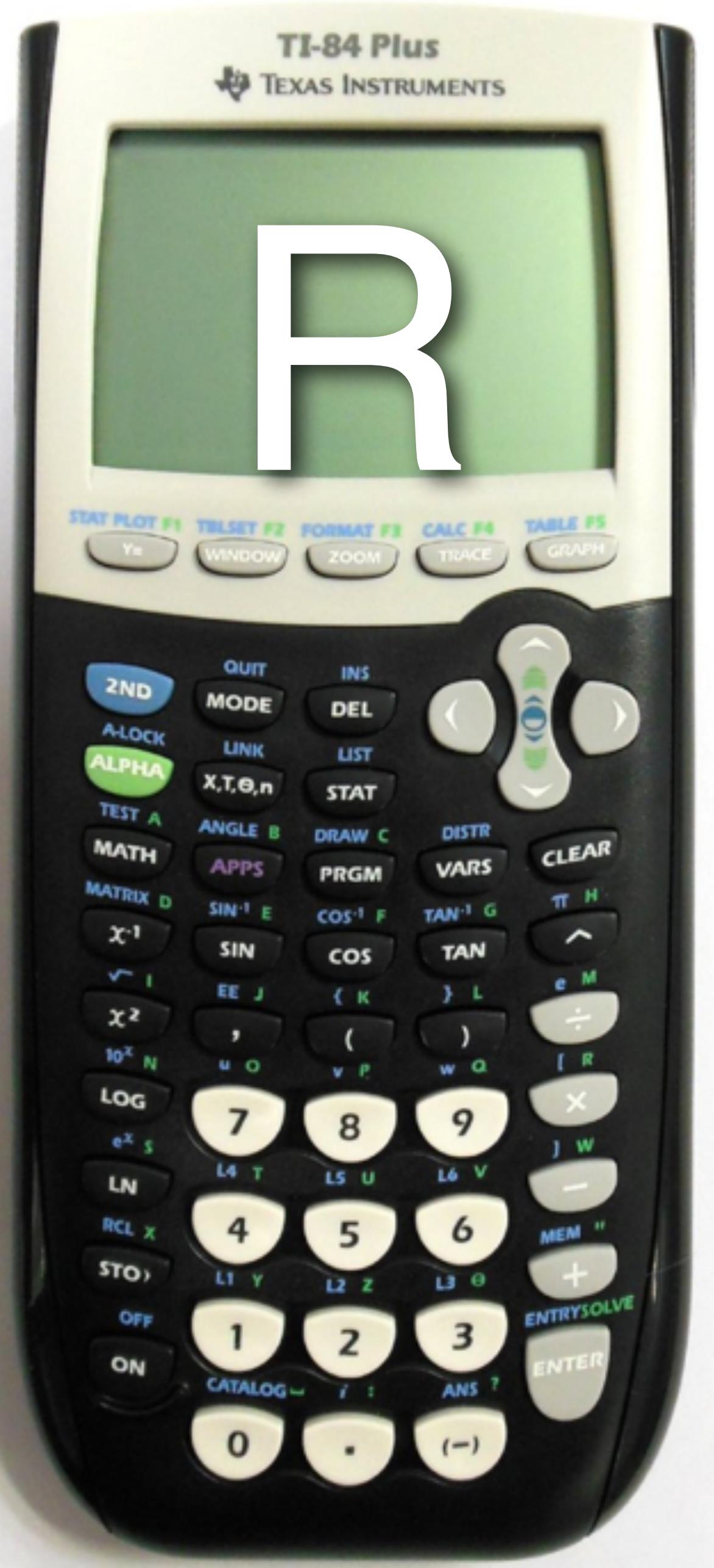
Somewhat helpful when R returns more than one value in the output.



The screenshot shows the RStudio Cloud interface with a browser header at the top. Below the header is a navigation bar with links to 'Intro To R & RStudio' and 'Intro to R and RStudio'. The main area is a code editor with tabs for 'File', 'Edit', 'Code', 'View', 'Plots', 'Session', 'Build', 'Debug', 'Profile', and 'Terminal'. The 'Terminal' tab is selected. The console window displays the following R session:

```
> 1+3  
[1] 4  
> 1+4  
[1] 5  
> 1+5  
[1] 6  
> |
```

The outputs '[1] 4', '[1] 5', and '[1] 6' are circled in red. The first output '[1] 4' corresponds to the large '[1]' shown above. The other two circled outputs correspond to the two smaller '[1]'s shown below.



R is like a fancy calculator  
on your computer

$$5 + 5$$

$$\# 10$$

$$4 - 1$$

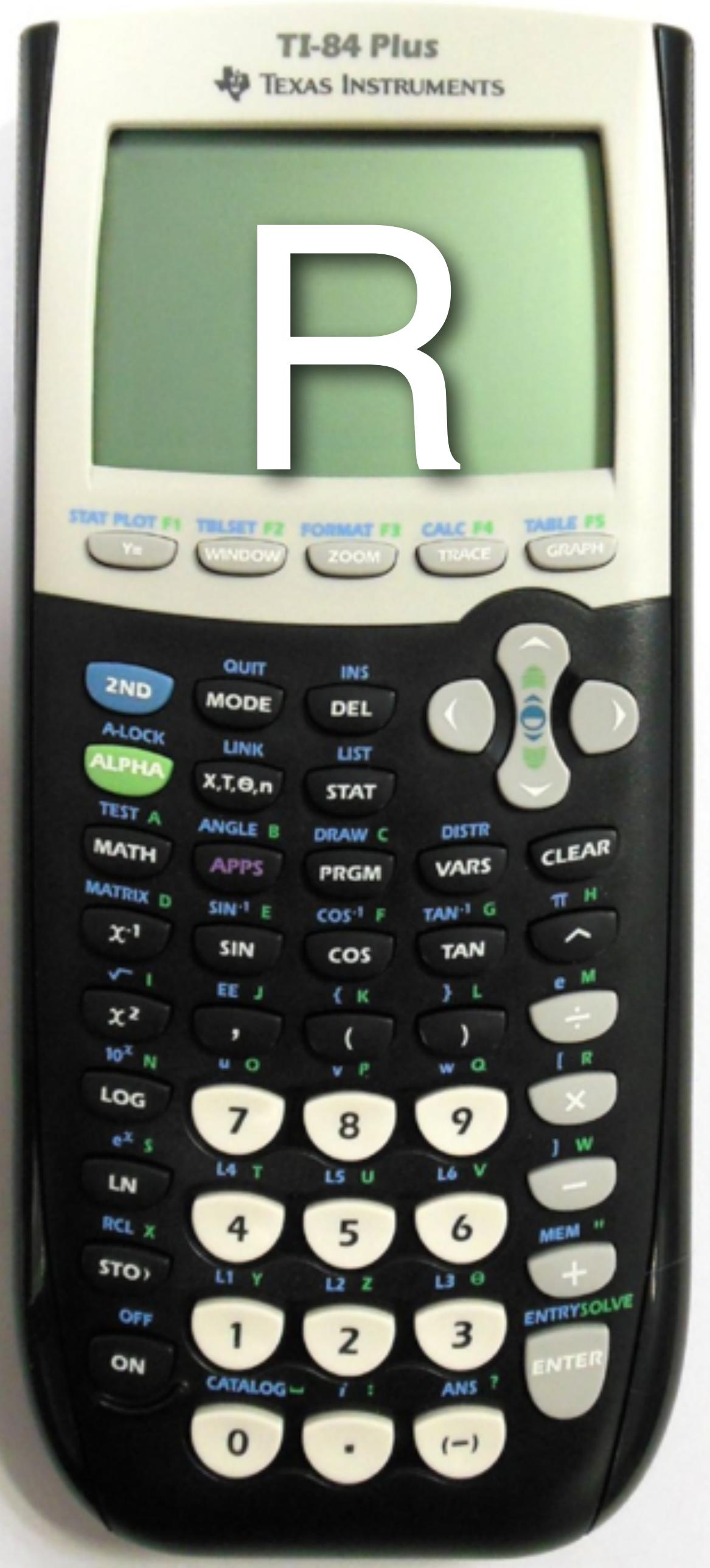
$$\# 3$$

$$1 * 2$$

$$\# 2$$

$$4 ^ 2$$

$$\# 16$$



It can do algebra

$$a <- 1$$

$$b <- 2$$

$$a + b$$

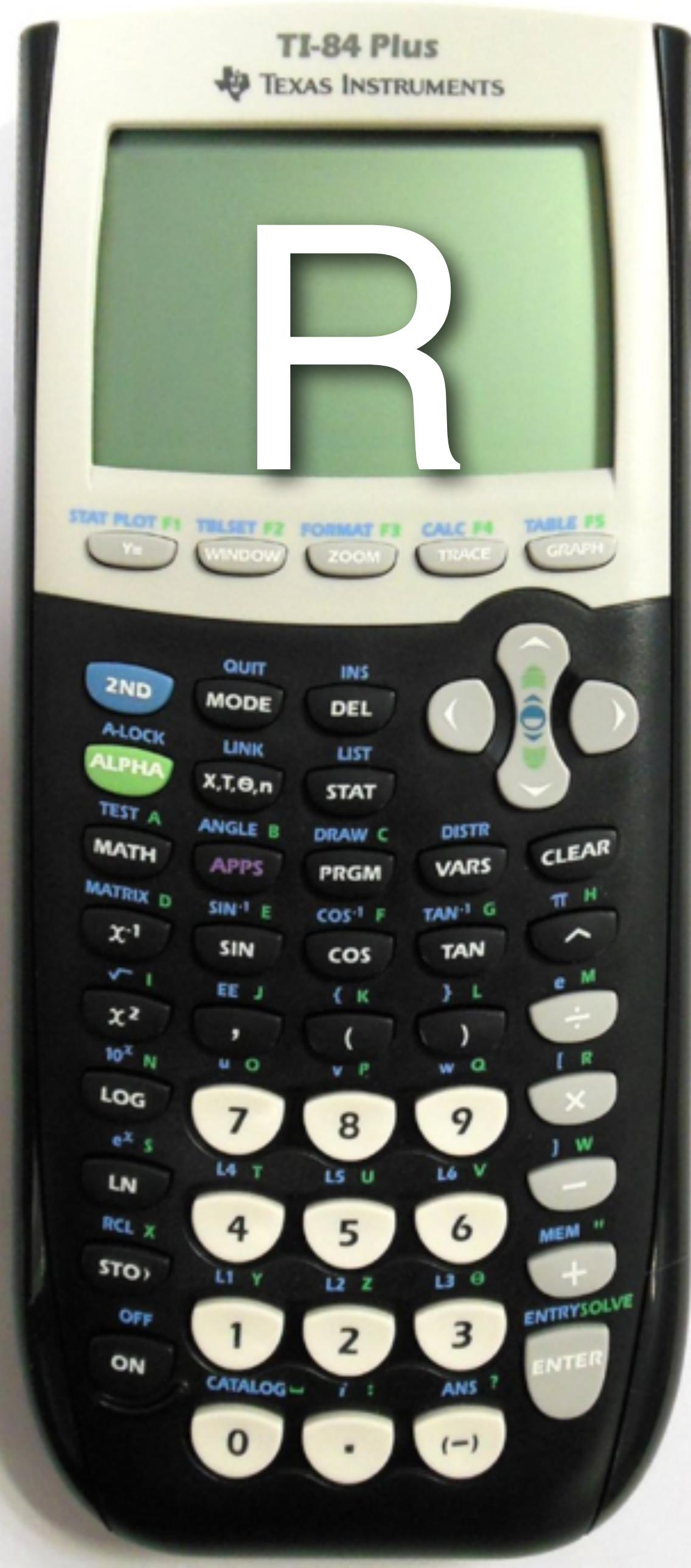
$$\# 3$$

$$A <- 3$$

It cares about capitalization

$$a + b - A$$

$$\# 0$$



And it has functions that let you do more sophisticated manipulations

round(3.1415)

# 3

factorial(3)

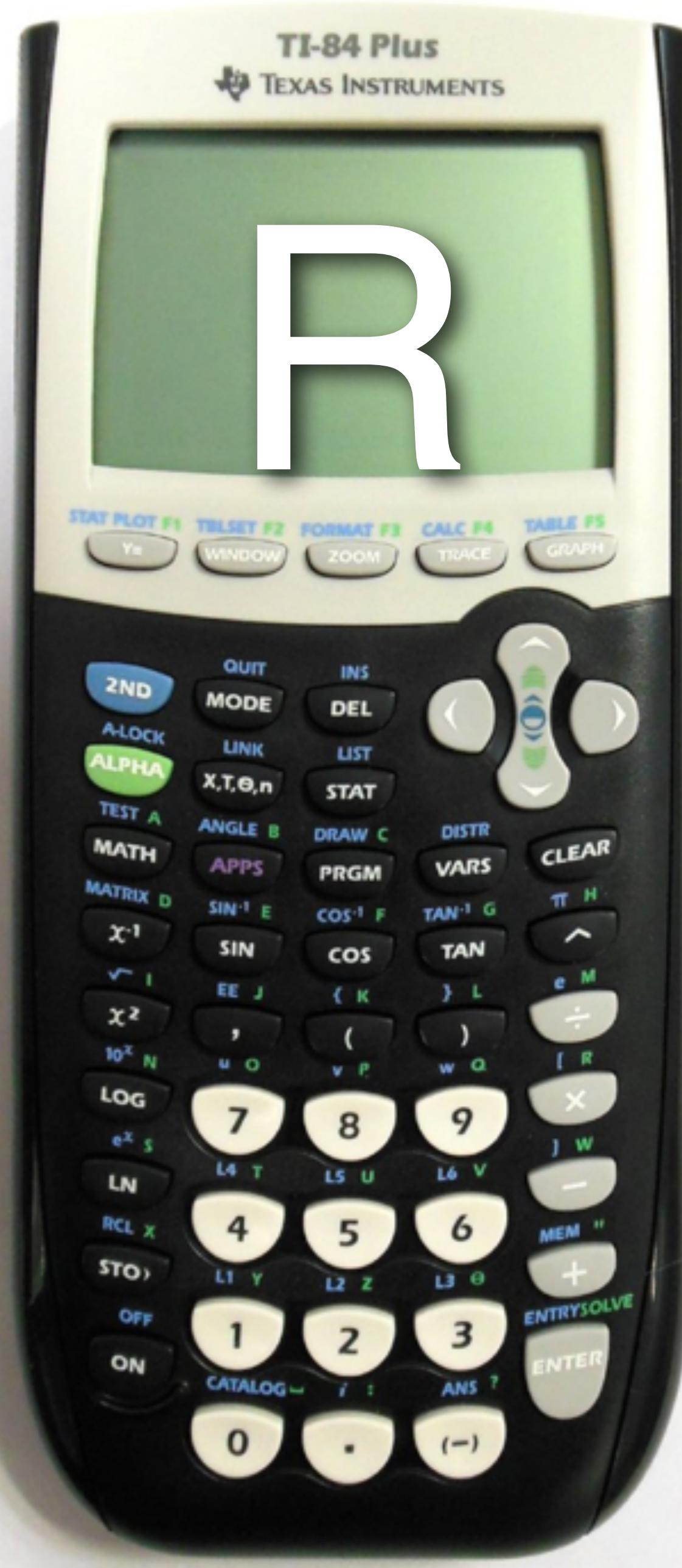
$$3! = 3 \times 2 \times 1$$

# 6

sqrt(9)

square root

# 3



Most of the cool stuff in R comes from functions. Like  $f(x)$  (“f of x”) functions in R have names, parentheses, and arguments

factorial(3)  
# 6

sqrt(9)  
# 3

factorial of 3  
square root of 9

# Your turn

What do you think this will return?

# Your turn

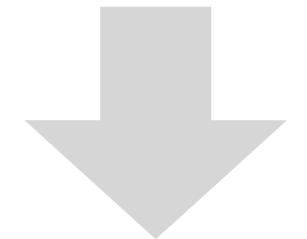
What do you think this will return?

```
factorial(round(2.0015) + 1)
```



R always works from the innermost parenthesis to the outermost (just like a calculator).

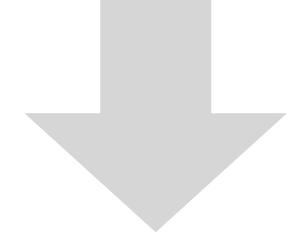
```
factorial(round(2.0015) + 1)
```



```
factorial(2 + 1)
```



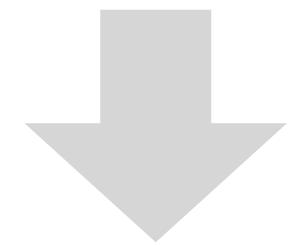
```
factorial(3)
```



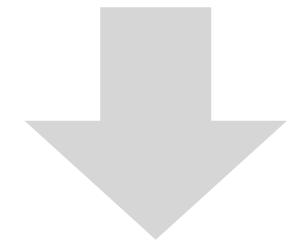
6

R always works from the innermost parenthesis to the outermost (just like a calculator).

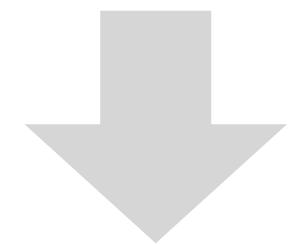
`factorial(round(2.0015) + 1)`



`factorial(2 + 1)`



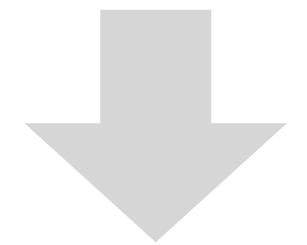
`factorial(3)`



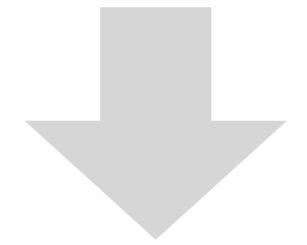
6

R always works from the innermost parenthesis to the outermost (just like a calculator).

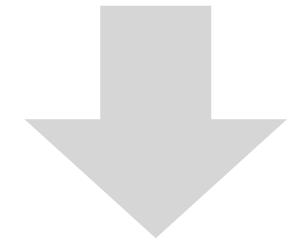
factorial(round(2.0015) + 1)



factorial(2 + 1)



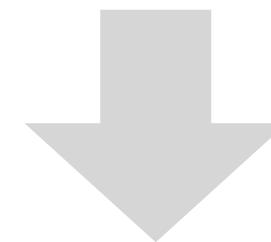
factorial(3)



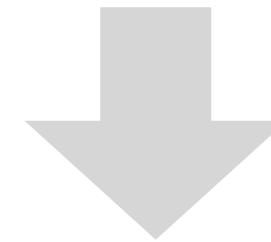
6

R always works from the innermost parenthesis to the outermost (just like a calculator).

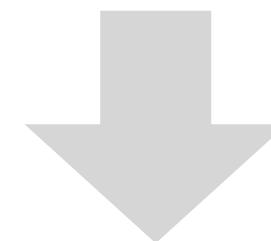
factorial(round(2.0015) + 1)



factorial(2 + 1)



factorial(3)



6

$$3! = 3 \times 2 \times 1$$

# + prompt

If your prompt turns into a "+", R thinks you haven't finished your previous command.

Either finish the command, or press escape.



The screenshot shows a web-based RStudio Cloud interface. The top bar includes the RStudio Cloud logo and a secure connection indicator. Below the header is a navigation bar with links to 'Intro To R & RStudio' and 'Intro to R and RStudio'. The main area features a toolbar with various icons for file operations and a search bar labeled 'Go to file/function'. Below the toolbar is a tab bar with 'Console' selected and 'Terminal' as an option. The console window displays the following R session:

```
> 1+3  
[1] 4  
> 1+4  
[1] 5  
> 1+5  
[1] 6  
> factorial(round(3.1415) + 1  
+
```

A red circle highlights the final '+', indicating it's a partially typed command.

# + prompt

If your prompt turns into a "+", R thinks you haven't finished your previous command.

Either finish the command, or press escape.



The screenshot shows the RStudio Cloud interface. The title bar says "RStudio Cloud". The address bar is secure with the URL <https://rstudio.cloud/spaces/1020/project/10706>. The main area shows a navigation menu with "Intro To R & RStudio / Intro to R and RStudio". Below is a toolbar with File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, and Help. A "Console" tab is selected. The console window displays the following R session:

```
> 1+3
[1] 4
> 1+4
[1] 5
> 1+5
[1] 6
> factorial(round(3.1415) + 1
+ )
[1] 24
>
```

# Your turn

Open RStudio and try the following tasks:

1. Pick a number and add 2 to it
2. Multiply the result by 3
3. Subtract 6 from the result of step 2
4. Divide the result of step 3 by 3

$10 + 2$

# 12

$12 * 3$

# 36

$36 - 6$

# 30

$30 / 3$

# 10

# Workflow

# R notebooks

It is easier to compose your code in an R Notebook than in the command line, and Notebooks allow you to keep text with your code.

We'll begin with a notebook I have started for you, called **01-Intro.Rmd**

Look in the folder **project**, then **code**, then click on

**01-Intro.Rmd**

The screenshot shows the RStudio Cloud interface with a red box highlighting the main workspace area. The workspace contains an R Notebook file named "01-Structures.Rmd". The code chunk at line 10 contains a large red heading "R Notebook" and explanatory text about R Notebooks.

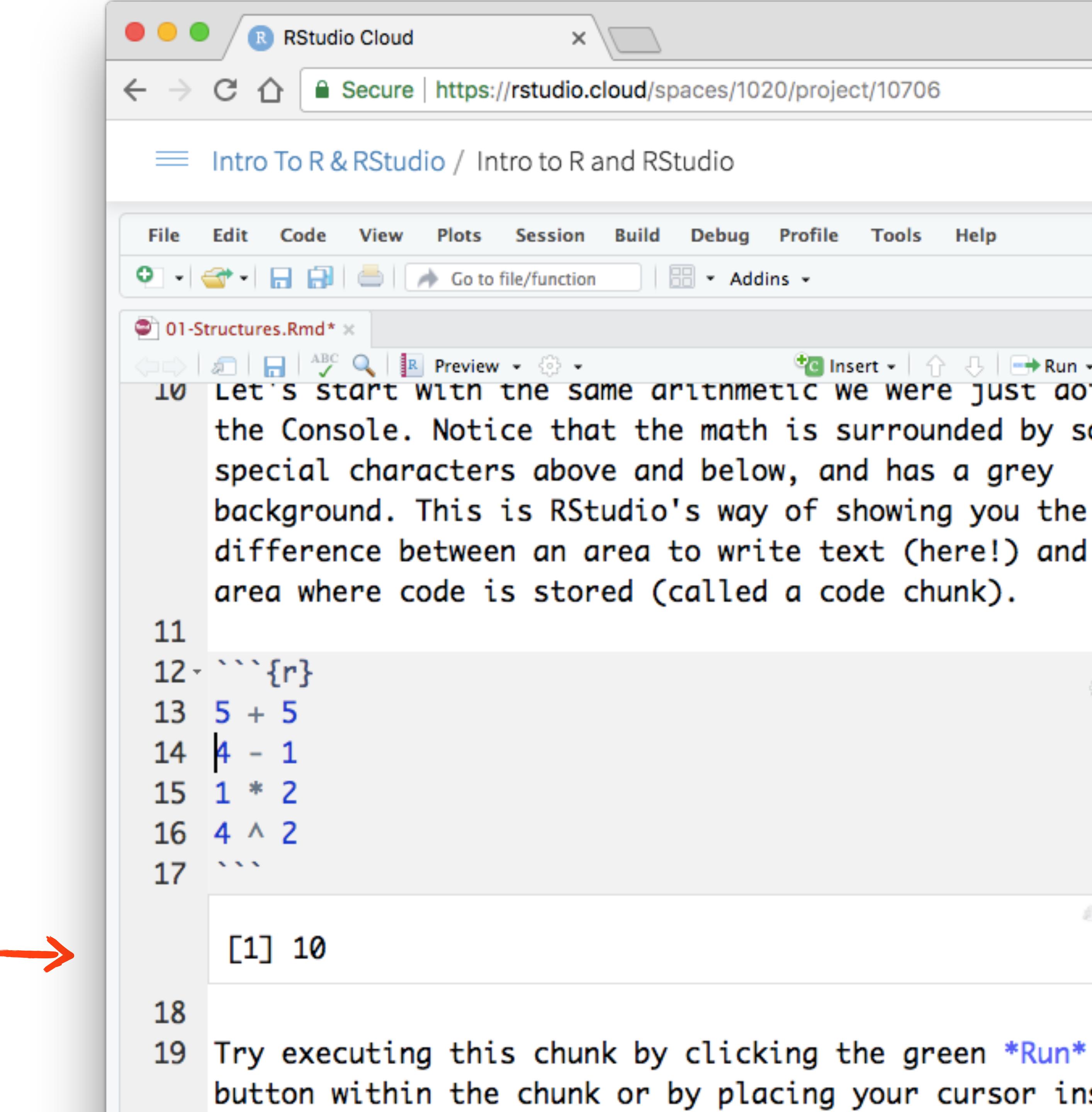
```
1 ---  
2 title: "R Notebook"  
3 output: html_notebook  
4 editor_options:  
5   chunk_output_type: inline  
6 ---  
7  
8 # Arithmetic and algebra  
9  
10 Let's start with the same arithmetic we were just doing in  
the Console. Notice that the math is surrounded by some  
special characters above and below, and has a grey  
background. This is RStudio's way of showing you the  
difference between an area to write text (here!) and an  
area where code is stored (called a code chunk).  
11  
12 ````{r}  
13 5 + 5  
14 4 - 1  
15 1 * 2
```

The R Notebook interface includes a toolbar, a preview pane, and a code editor. To the right of the code editor are the Environment, History, and Connections panes. The Environment pane displays "Environment is empty". Below the code editor is a file browser showing three files: "01-Structures.Rmd", "02-Syntax.Rmd", and "03-Programming.Rmd".

Notice that the Console has automatically minimized itself, to give you room to work in your notebook. From here out, we'll be working almost exclusively in notebooks, but all the code we write would work in the Console as well.

The screenshot shows the RStudio Cloud interface. At the top, there's a toolbar with various icons. Below it is a header bar with the title 'Intro To R & RStudio / Intro to R and RStudio' and a user profile for 'Amelia McNamara'. The main area contains a code editor with R Markdown syntax. A red box highlights the 'Console' tab at the bottom left of the editor. In the background, there's a file browser showing three files: '01-Structures.Rmd', '02-Syntax.Rmd', and '03-Programming.Rmd'. The '01-Structures.Rmd' file is selected. The overall interface is clean and modern, typical of a cloud-based development environment.

**Do what the text  
instructs, and run a line  
of the code. Notice how  
results display  
immediately below the  
chunk, just like they did  
in the Console.**



The screenshot shows the RStudio Cloud interface. The title bar says "RStudio Cloud" and the URL is "Secure | https://rstudio.cloud/spaces/1020/project/10706". The main area shows a file named "01-Structures.Rmd\*". A red arrow points from the text above to the code chunk starting at line 10. The code chunk contains the following R code:

```
10 Let's start with the same arithmetic we were just doing in the Console. Notice that the math is surrounded by special characters above and below, and has a grey background. This is RStudio's way of showing you the difference between an area to write text (here!) and an area where code is stored (called a code chunk).
11
12 ``{r}
13 5 + 5
14 | 4 - 1
15 1 * 2
16 4 ^ 2
17 ``
```

The output of the code chunk is displayed below it:

```
[1] 10
```

Line 18 and 19 provide instructions for executing the chunk:

```
18
19 Try executing this chunk by clicking the green *Run* button within the chunk or by placing your cursor in
```

9  
10 Let's start with the same arithmetic we were just doing in  
the Console. Notice that the math is surrounded by some  
special characters above and below, and has a grey  
background. This is RStudio's way of showing you the  
difference between an area to write text (here!) and an area  
where code is stored (called a code chunk).

11  
12 ````{r}`  
13 `5 + 5`  
14 `4 - 1`  
15 `1 * 2`  
16 `4 ^ 2`  
17 `````  
[1] 10

Files Plots Packages Help

New Folder Delete R

Home > Dropbox > Intro\_to

Name
..
.Rhistory
04-Syntax.nb.html
04-Syntax.Rmd
03-DataTypes.nb.html
03-DataTypes.Rmd
02-Visualization.Rmd
01-Intro.Rmd
solutions

18  
19 Try executing this chunk by clicking the green **\*Run\*** button  
within the chunk or by placing your cursor inside it and  
pressing **\*Cmd+Shift+Enter\***

9

10 Let's start with the same arithmetic we were just doing in the Console. Notice that the math is surrounded by some special characters above and below, and has a grey background. This is RStudio's way of showing you the difference between an area to write text (here!) and an area where code is stored (called a code chunk).

11

```
12 `r`  
13 5 + 5  
14 4 - 1  
15 1 * 2  
16 4 ^ 2  
17 ...
```

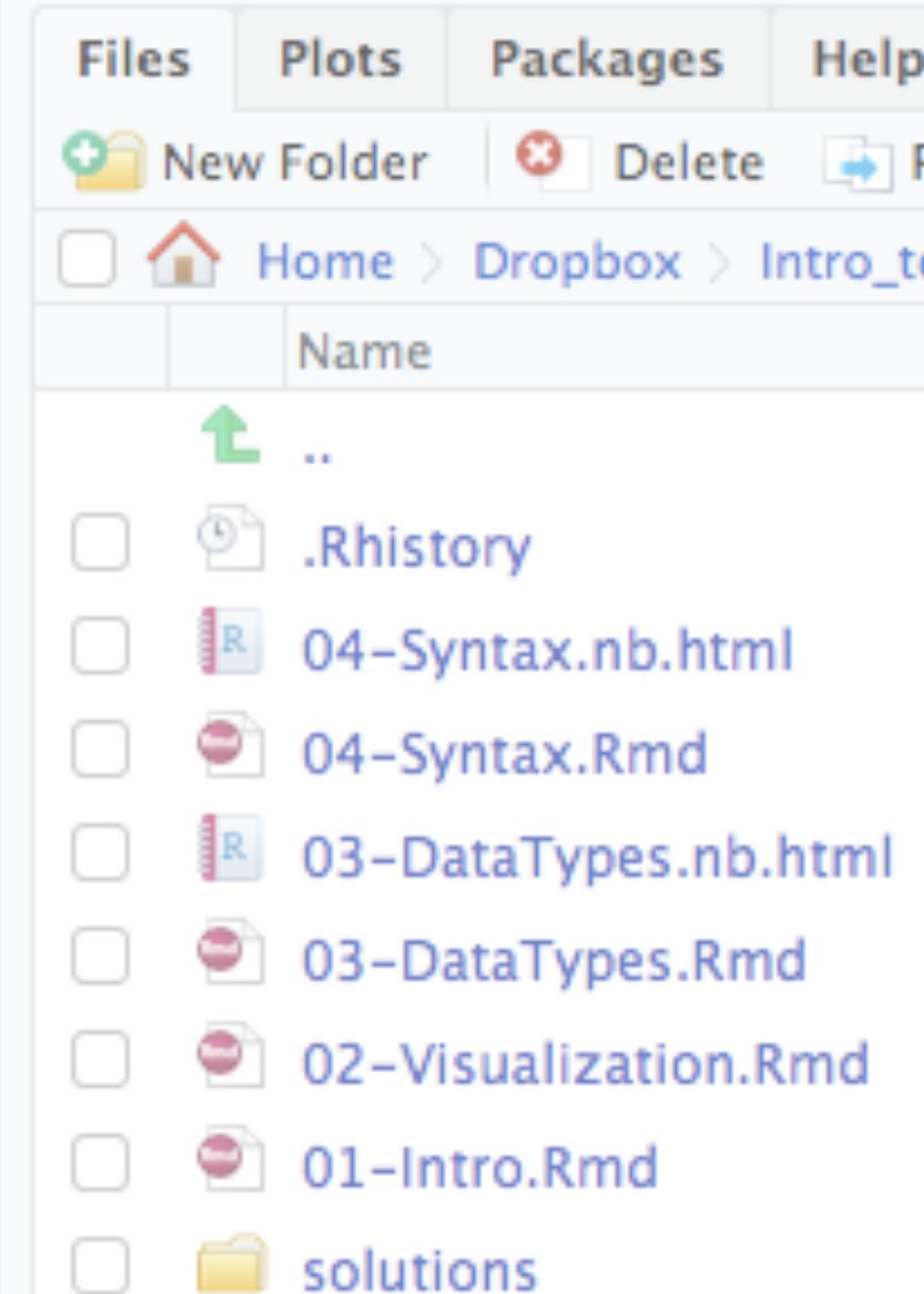
You can also run just one line of code, by placing your cursor on the line and hitting Command Enter

Output for the one line you've run

[1] 10

18

19 Try executing this chunk by clicking the green \*Run\* button within the chunk or by placing your cursor inside it and pressing \*Cmd+Shift+Enter\*



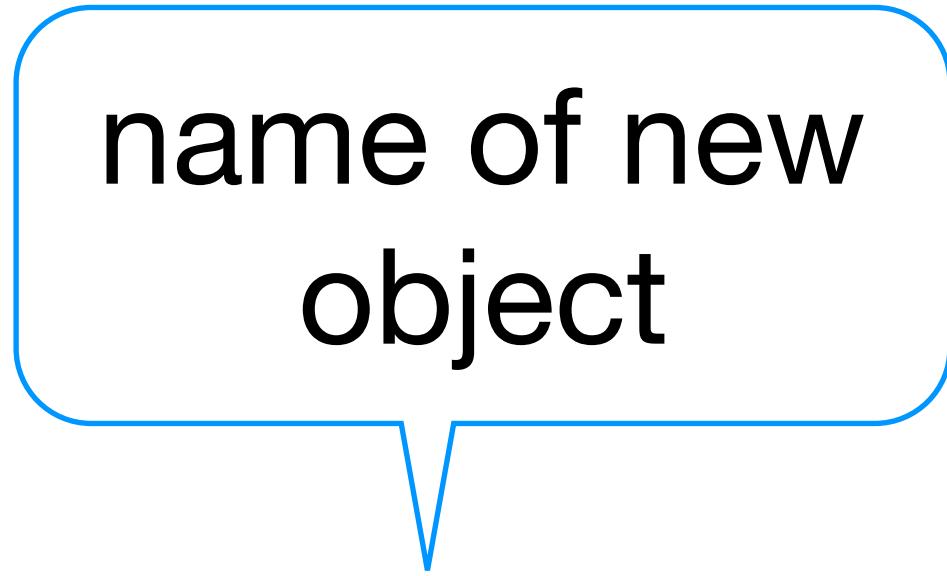
# R objects

You can save information as an R object with the greater than sign followed by a minus, e.g, an arrow: <-

```
the_answer <- 42
```

You can save information as an R object with the greater than sign followed by a minus, e.g, an arrow: <-

name of new  
object



```
the_answer <- 42
```

You can save information as an R object with the greater than sign followed by a minus, e.g, an arrow: <-

assignment  
operator,  
"gets"

the\_answer <- 42

You can save information as an R object with the greater than sign followed by a minus, e.g, an arrow: <-

information  
to store in the  
object

the\_answer <- 42

# When you create an R object, you'll see it appear in your environment pane

The screenshot shows the RStudio interface with the following details:

- Cloud** tab is active.
- Address Bar:** secure | <https://rstudio.cloud/spaces/1020/project/10706>
- User:** Amelia McNamara
- Toolbar:** View, Plots, Session, Build, Debug, Profile, Tools, Help, Go to file/function, Addins, Preview, Insert, Run, Import Dataset, Global Environment, List, R 3.4.2.
- Environment Tab:** Values table:

	Values
a	1
A	3
b	2
- Files Tab:** New Folder, Upload, Delete, Rename, More, Home > Day1 > code, Name, Size, Modified, 01-Structures.Rmd (1.3 KB, Jan 27, 2018, 5:19 PM), 02-Syntax.Rmd (869 B, Jan 27, 2018, 5:20 PM), 03-Programming.Rmd (869 B, Jan 27, 2018, 5:20 PM).

A red circle highlights the "Values" table in the Environment pane. A red text overlay on the right side of the image asks: "What objects are in your environment right now?"

# Common R workflow

Save output of one function as an R object  
to use in a second function.

```
more_pi <- round(3.1415) + 1
```

```
more_pi
```

```
# 4
```

```
factorial(more_pi)
```

```
# 24
```

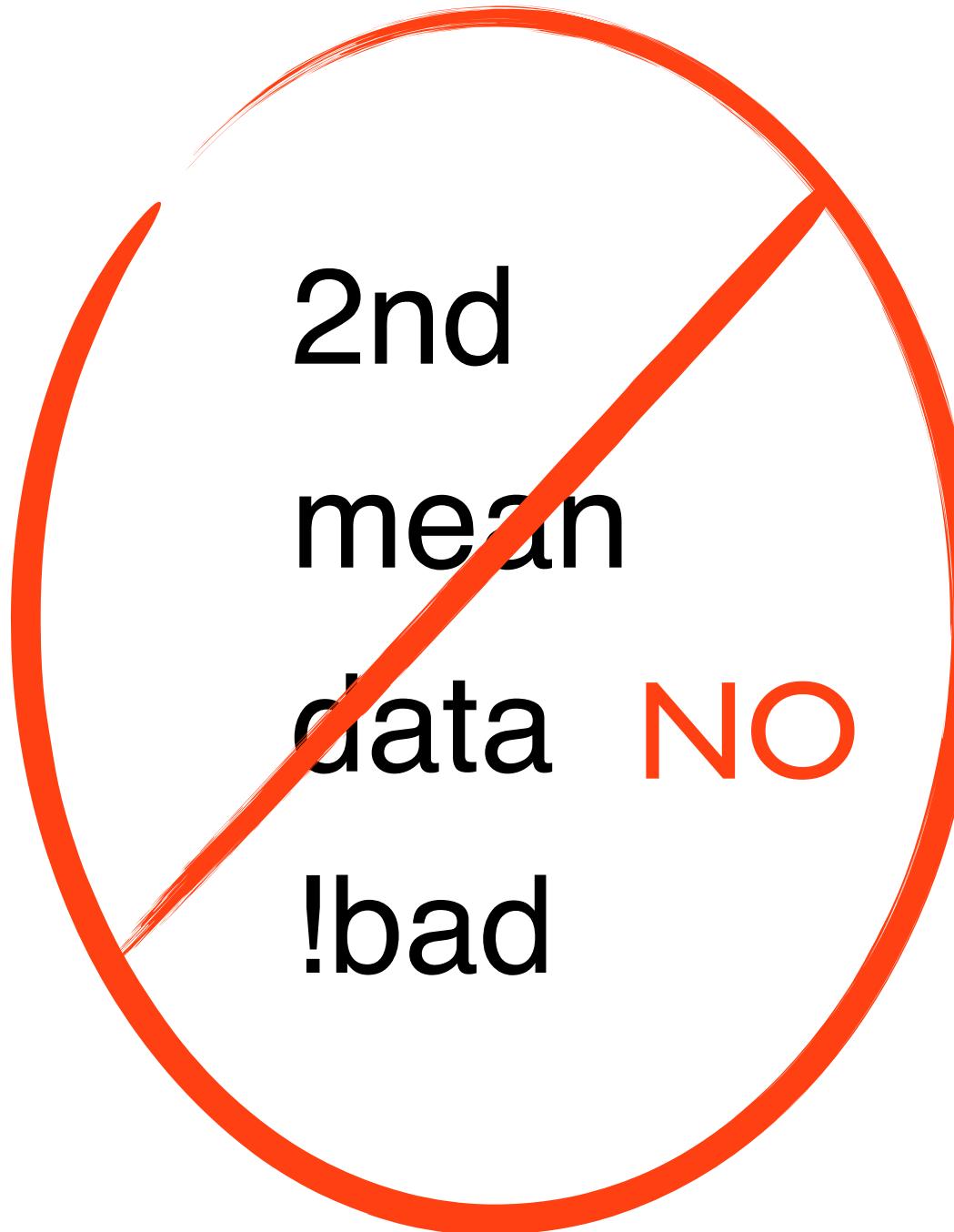
# Object names

Object names cannot begin with numbers

They cannot contain spaces

It is wise to avoid names already in use

Informative names are better than generic ones



a	CDC_data
b	finalModel
FOO	more_pi yeah!
my_var	withoutOver64
.day	

# Capitalization matters

R will treat each of these as a different object

cdc\_data

finalmodel

sum

CDC\_data

finalModel

SUM

# rm

You can remove an object from your environment with rm

```
more_pi
```

```
# 4
```

```
rm(more_pi)
```

```
more_pi
```

```
# Error: object 'more_pi' not found
```

This can be useful if you overwrite an object that comes with R

```
pi
```

```
# 3.141593
```

```
pi <- 1
```

```
pi
```

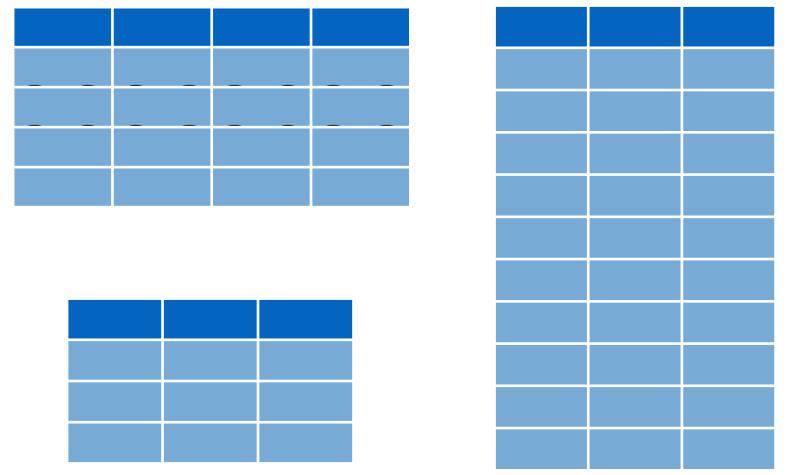
```
# 1
```

```
rm(pi)
```

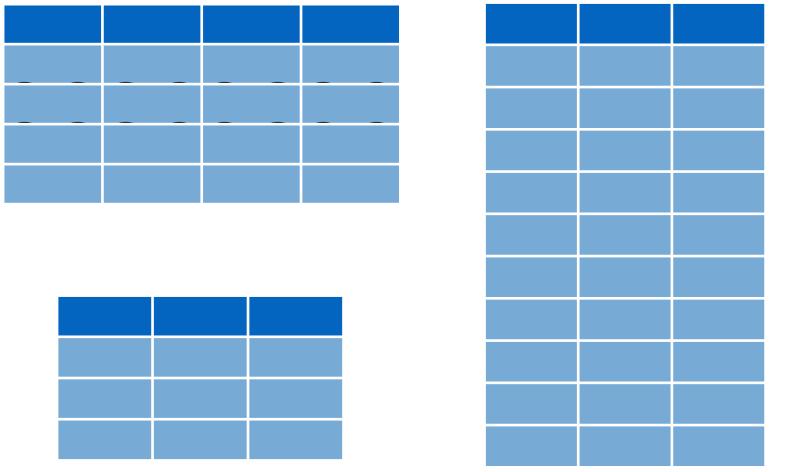
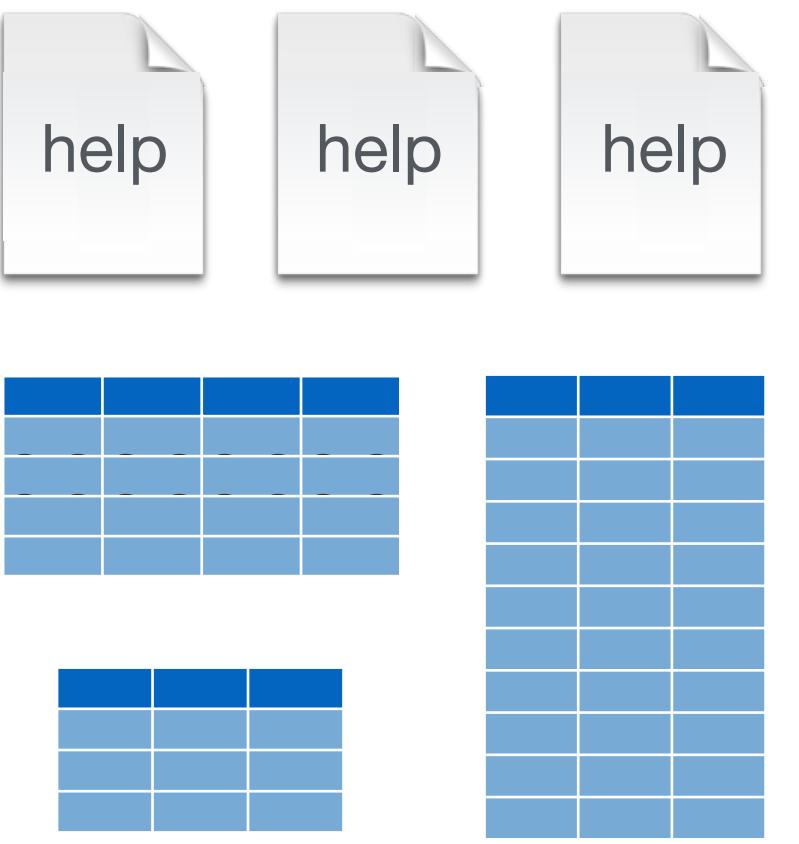
```
pi
```

```
# 3.141593
```

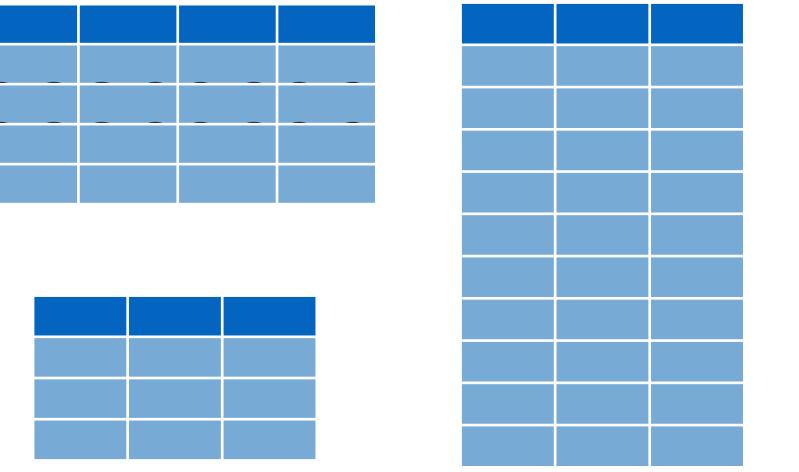
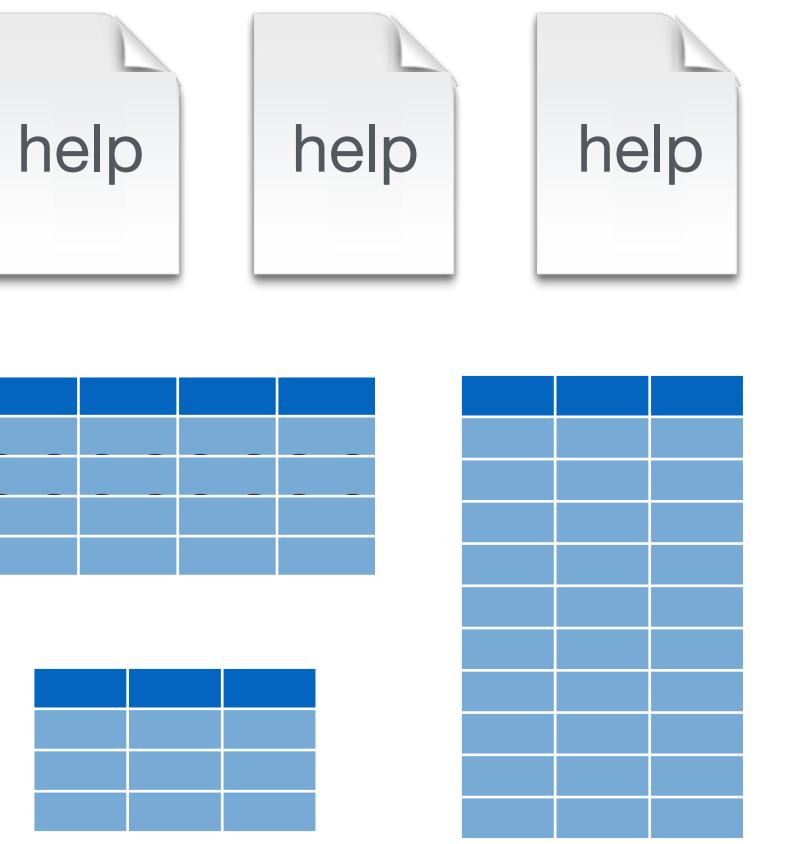
# R packages



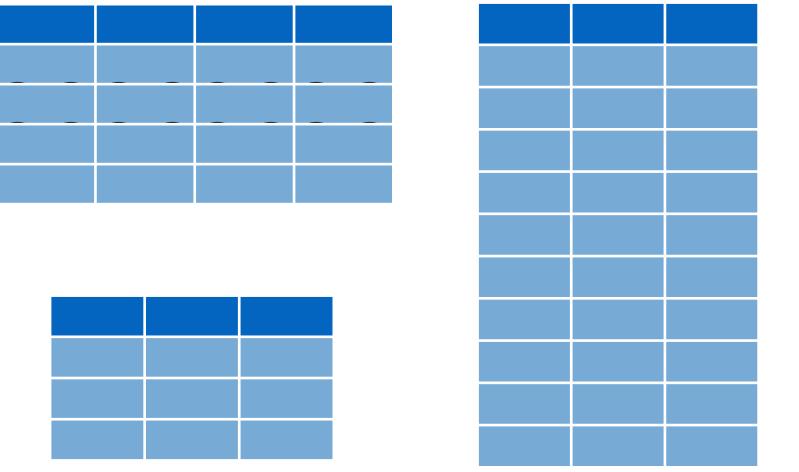
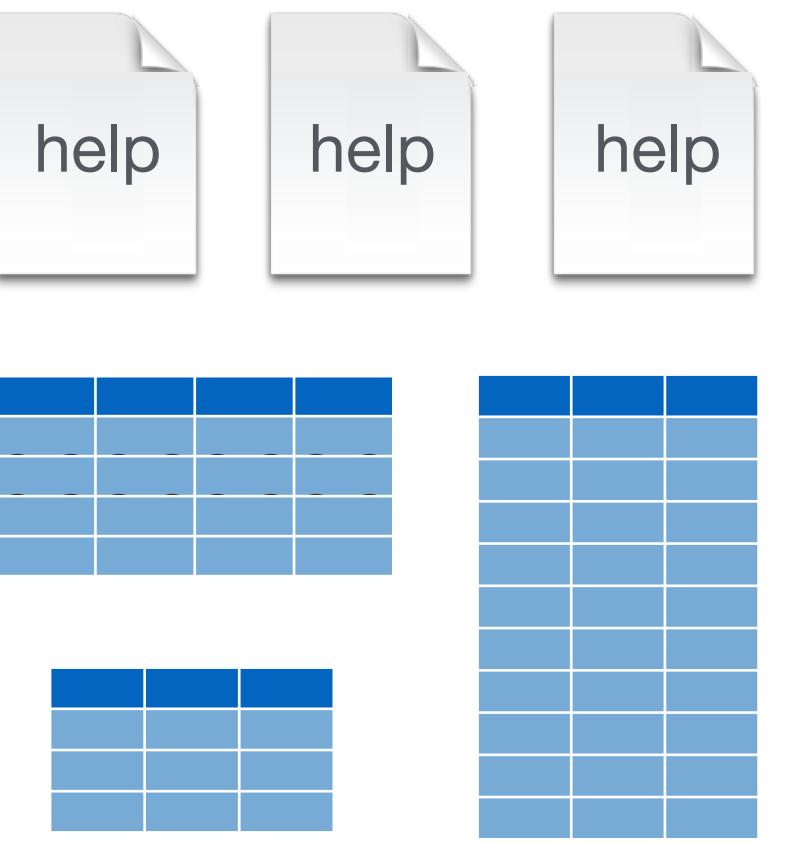
function1()  
function2()  
function3()  
function4()



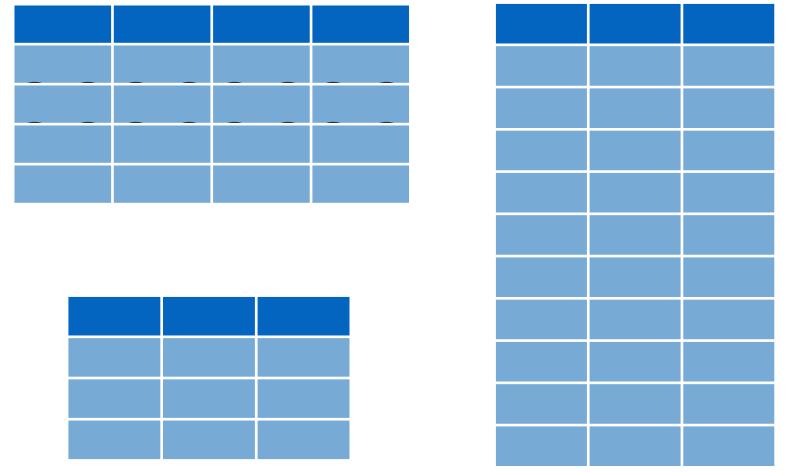
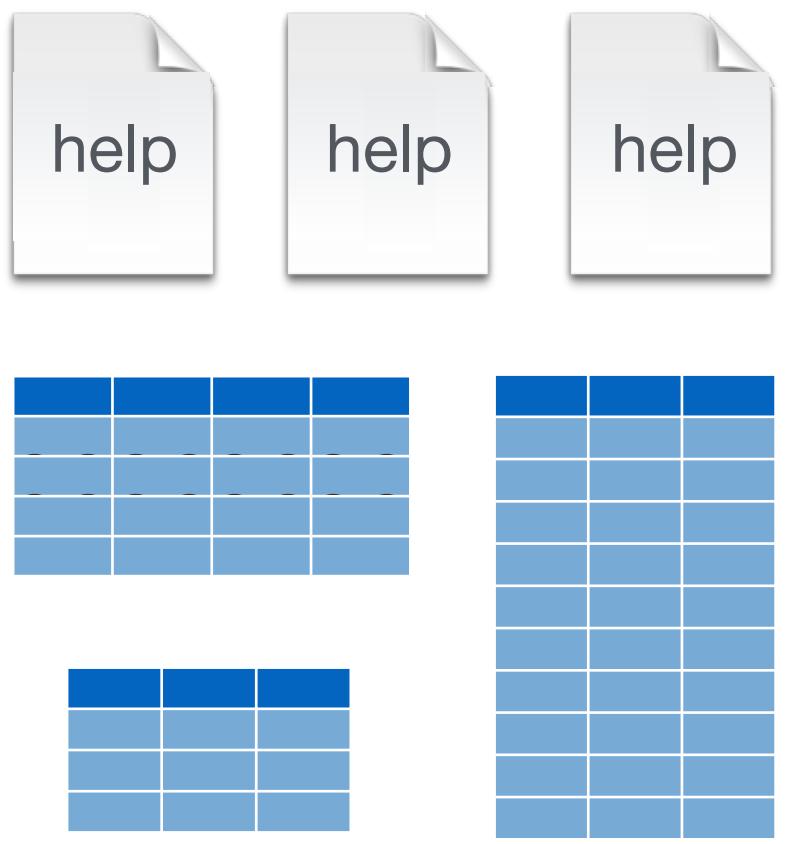
function1()  
function2()  
function3()  
function4()



function5()  
function6()  
function7()  
function8()



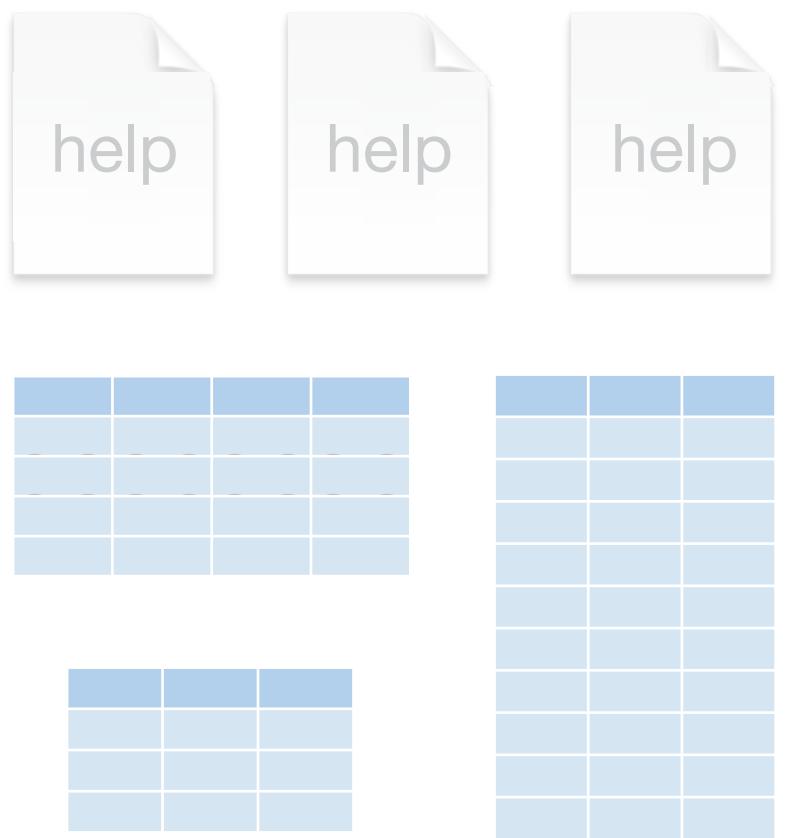
function9()  
functionA()  
functionB()  
functionC()



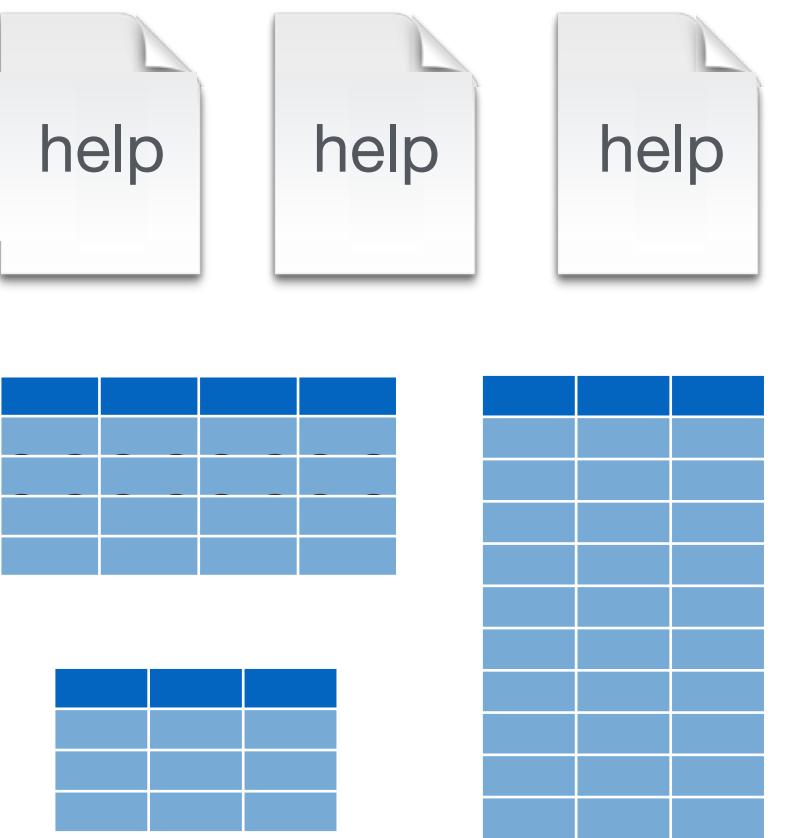
functionD()  
functionE()  
functionF()  
functionG()



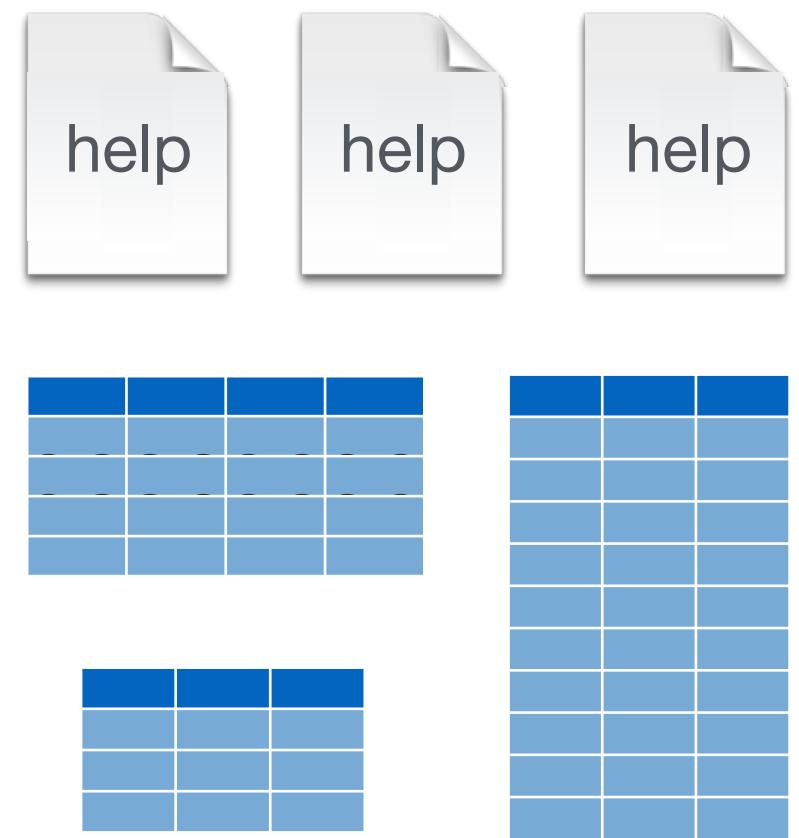
Base R



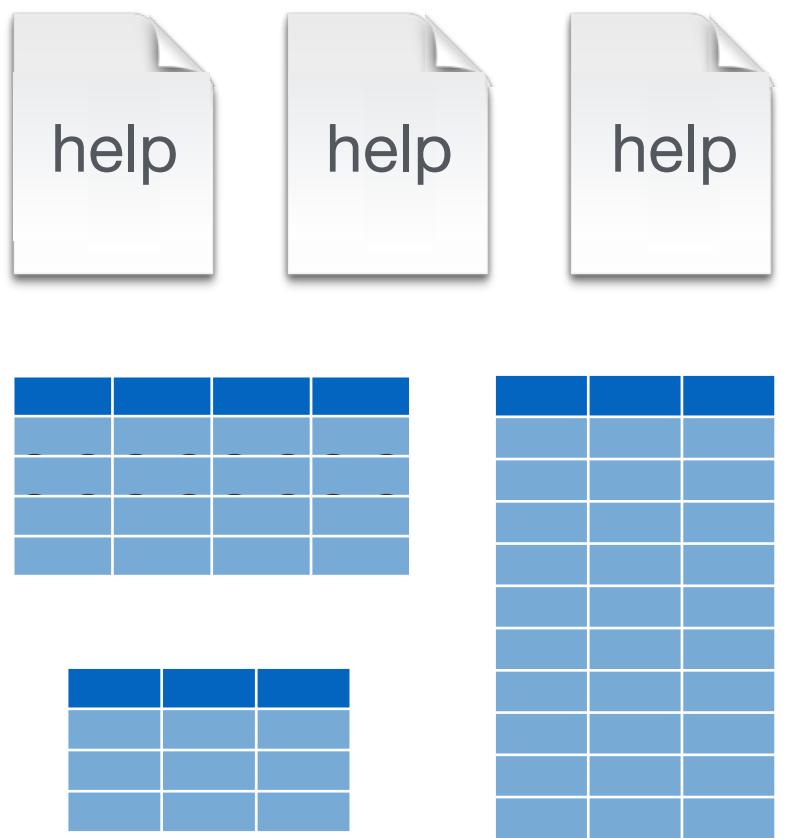
function1()  
function2()  
function3()  
function4()



function5()  
function6()  
function7()  
function8()



function9()  
functionA()  
functionB()  
functionC()



functionD()  
functionE()  
functionF()  
functionG()

Base R

R Packages

The Comprehensive R Archive X

Secure | https://cran.r-project.org

Amelia



**Available CRAN Packages By Name**

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#)

<a href="#">CRAN</a>	<a href="#">A3</a>	Accurate, Adaptable, and Accessible Error Metrics for Predictive Models
<a href="#">Mirrors</a>	<a href="#">abbyyR</a>	Access to Abbyy Optical Character Recognition (OCR) API
<a href="#">What's new?</a>	<a href="#">abc</a>	Tools for Approximate Bayesian Computation (ABC)
<a href="#">Task Views</a>	<a href="#">abc.data</a>	Data Only: Tools for Approximate Bayesian Computation (ABC)
<a href="#">Search</a>	<a href="#">ABC.RAP</a>	Array Based CpG Region Analysis Pipeline
	<a href="#">ABCAnalysis</a>	Computed ABC Analysis
	<a href="#">abcdeFBA</a>	ABCDE_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package
	<a href="#">ABCOptim</a>	Implementation of Artificial Bee Colony (ABC) Optimization
	<a href="#">ABCp2</a>	Approximate Bayesian Computational Model for Estimating P2
	<a href="#">abcrf</a>	Approximate Bayesian Computation via Random Forests
	<a href="#">abctools</a>	Tools for ABC Analyses
	<a href="#">abd</a>	The Analysis of Biological Data
	<a href="#">abe</a>	Augmented Backward Elimination
	<a href="#">abf2</a>	Load Gap-Free Axon ABF2 Files
	<a href="#">ABHgenotypeR</a>	Easy Visualization of ABH Genotypes
	<a href="#">abind</a>	Combine Multidimensional Arrays
	<a href="#">abjutils</a>	Useful Tools for Jurimetical Analysis Used by the Brazilian Jurimetrics Association
	<a href="#">abn</a>	Modelling Multivariate Data with Additive Bayesian Networks
	<a href="#">abodOutlier</a>	Angle-Based Outlier Detection
	<a href="#">ABPS</a>	The Abnormal Blood Profile Score to Detect Blood Doping
	<a href="#">AbsFilterGSEA</a>	Improved False Positive Control of Gene-Permuting GSEA with Absolute Filtering
	<a href="#">AbSim</a>	Time Resolved Simulations of Antibody Repertoires
	<a href="#">abundant</a>	High-Dimensional Principal Fitted Components and Abundant Regression
	<a href="#">ACA</a>	Abrupt Change-Point or Aberration Detection in Point Series
	<a href="#">acc</a>	Exploring Accelerometer Data
	<a href="#">accelrometry</a>	Functions for Processing Minute-to-Minute Accelerometer Data
	<a href="#">acelmissing</a>	Missing Value Imputation for Accelerometer Data
	<a href="#">AcceptanceSampling</a>	Creation and Evaluation of Acceptance Sampling Plans
	<a href="#">ACCLMA</a>	ACC & LMA Graph Plotting

# Using packages

**1.**

```
install.packages("ggplot2")
```

Downloads files to computer

**1 x per computer**

**2.**

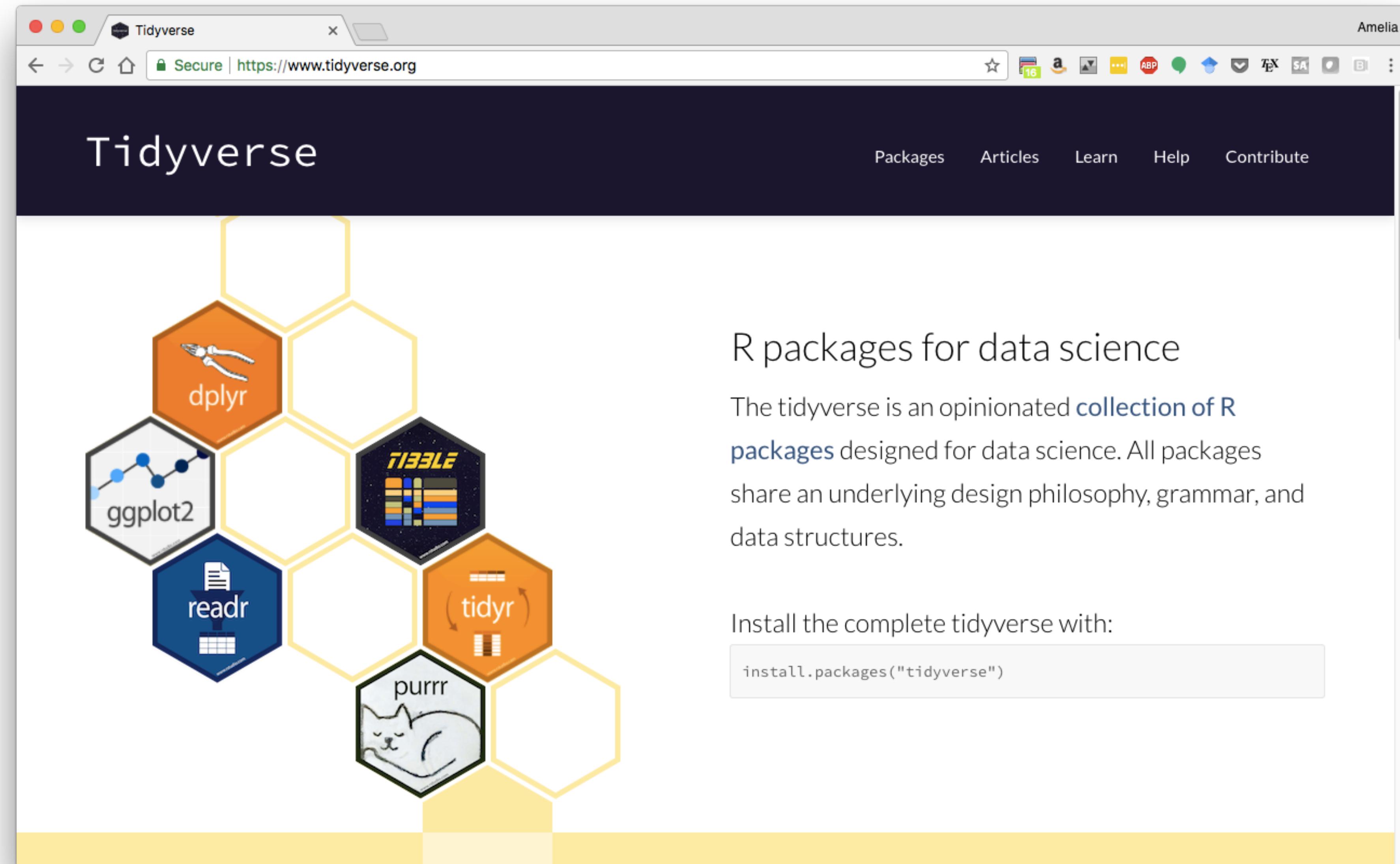
```
library(ggplot2)
```

Loads package

**1 x per R Session**

I've done this  
for you for this  
workshop

# The tidyverse

A screenshot of a web browser displaying the Tidyverse website at <https://www.tidyverse.org>. The page has a dark blue header with the word "Tidyverse" in white. Below the header is a navigation bar with links for "Packages", "Articles", "Learn", "Help", and "Contribute". The main content area features a hexagonal grid of icons representing various R packages: dplyr (orange, with a pliers icon), ggplot2 (grey, with a line plot icon), readr (blue, with a file icon), purrr (grey, with a cat icon), tibble (black, with a grid icon), and tidyr (orange, with a circular arrow icon). To the right of the grid, the text "R packages for data science" is displayed, followed by a detailed description of what the tidyverse is. Below this, there is a section titled "Install the complete tidyverse with:" containing the R command "install.packages("tidyverse")".

# Tidyverse

Packages Articles Learn Help Contribute

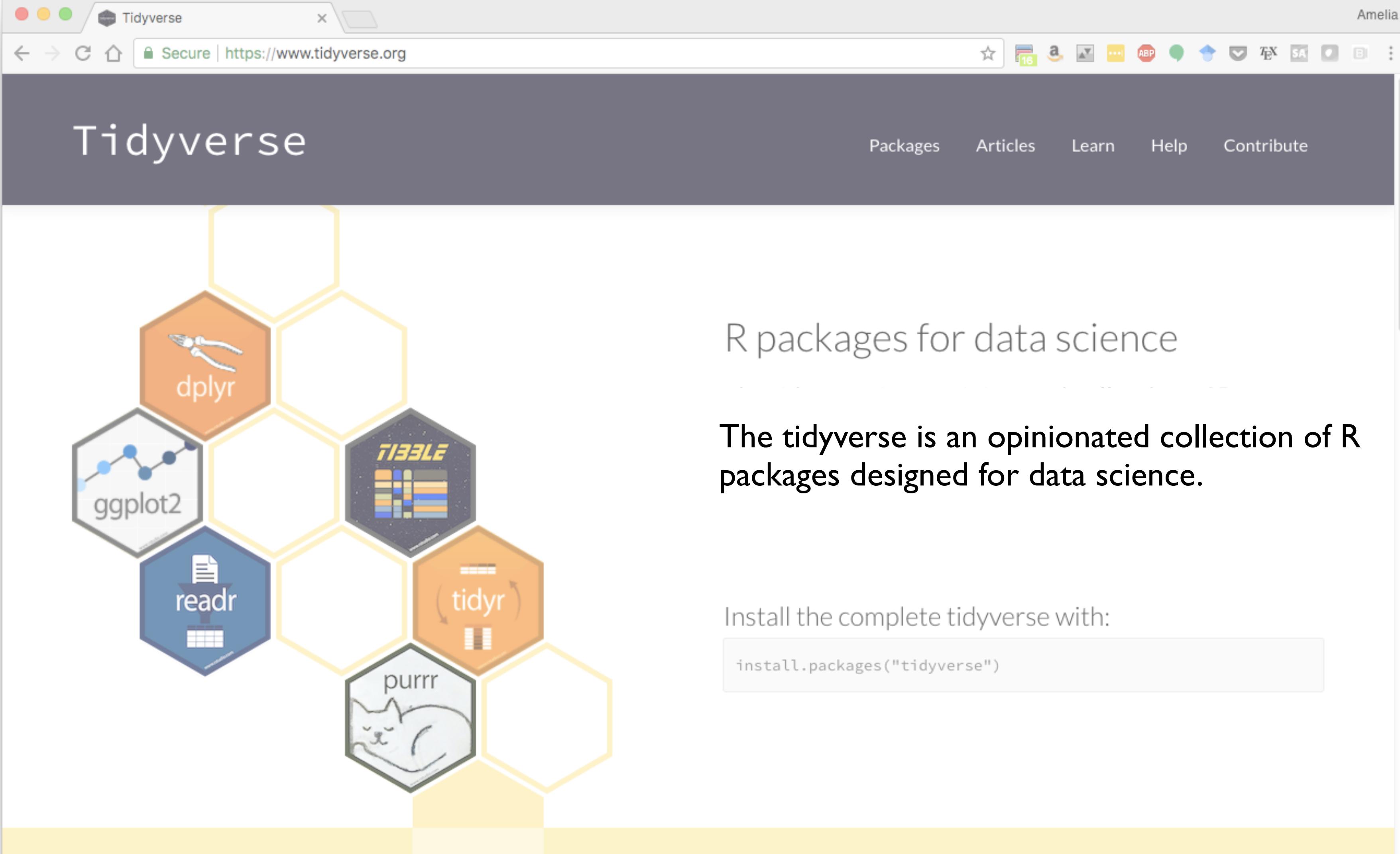


R packages for data science

The tidyverse is an opinionated [collection of R packages](#) designed for data science. All packages share an underlying design philosophy, grammar, and data structures.

Install the complete tidyverse with:

```
install.packages("tidyverse")
```

The image shows a screenshot of a web browser displaying the official Tidyverse website at <https://www.tidyverse.org>. The page has a dark grey header with the word "Tidyverse" in white. Below the header is a navigation bar with links for "Packages", "Articles", "Learn", "Help", and "Contribute". The main content area features a large hexagonal grid composed of several smaller hexagons, each containing an icon for one of the six core Tidyverse packages: dplyr (orange, with a pair of pliers icon), ggplot2 (grey, with a line plot icon), readr (blue, with a document icon), purrr (grey, with a cat icon), tibble (dark grey, with a grid icon), and tidyr (orange, with a circular arrow icon). To the right of the grid, the text "R packages for data science" is displayed in a light grey font. Below this, a larger text block states: "The tidyverse is an opinionated collection of R packages designed for data science." Further down, there is a section titled "Install the complete tidyverse with:" followed by a code snippet: `install.packages("tidyverse")`.

# R Syntax Comparison :: CHEAT SHEET

## Dollar sign syntax

```
goal(data$x, data$y)
```

### SUMMARY STATISTICS:

one continuous variable:

```
mean(mtcars$mpg)
```

one categorical variable:

```
table(mtcars$cyl)
```

two categorical variables:

```
table(mtcars$cyl, mtcars$am)
```

one continuous, one categorical:

```
mean(mtcars$mpg[mtcars$cyl==4])
```

```
mean(mtcars$mpg[mtcars$cyl==6])
```

```
mean(mtcars$mpg[mtcars$cyl==8])
```

### PLOTTING:

one continuous variable:

```
hist(mtcars$disp)
```

```
boxplot(mtcars$disp)
```

one categorical variable:

```
barplot(table(mtcars$cyl))
```

two continuous variables:

```
plot(mtcars$disp, mtcars$mpg)
```

two categorical variables:

```
mosaicplot(table(mtcars$am, mtcars$cyl))
```

one continuous, one categorical:

```
histogram(mtcars$disp[mtcars$cyl==4])
```

```
histogram(mtcars$disp[mtcars$cyl==6])
```

```
histogram(mtcars$disp[mtcars$cyl==8])
```

```
boxplot(mtcars$disp[mtcars$cyl==4])
```

```
boxplot(mtcars$disp[mtcars$cyl==6])
```

```
boxplot(mtcars$disp[mtcars$cyl==8])
```

### WRANGLING:

subsetting:

```
mtcars[mtcars$mpg>30]
```

making a new variable:

```
mtcars$efficiency
```

```
mtcars$efficiency
```

## Formula syntax

```
goal(y~x|z, data=data, group=w)
```

### SUMMARY STATISTICS:

one continuous variable:

```
mosaic:::mean(~mpg, data=mtcars)
```

one categorical variable:

```
mosaic:::tally(~cyl, data=mtcars)
```

two categorical variables:

```
mosaic:::tally(cyl~am, data=mtcars)
```

one continuous, one categorical:

```
mosaic:::mean(mpg~cyl, data=mtcars)
```

tilde

### PLOTTING:

one continuous variable:

```
lattice:::histogram(~disp, data=mtcars)
```

```
lattice:::bwplot(~disp, data=mtcars)
```

one categorical variable:

```
mosaic:::bargraph(~cyl, data=mtcars)
```

two continuous variables:

```
lattice:::xyplot(~disp|cyl, data=mtcars)
```

two categorical variables:

```
mosaic:::mosaicplot(~cyl|am, data=mtcars, group=cyl)
```

one continuous, one categorical:

```
ggplot(mtcars, aes(x=disp, y=mpg))
```

### WRANGLING:

subsetting:

```
mtcars %>% dplyr:::filter(mpg>30)
```

making a new variable:

```
mtcars <- mtcars %>%
```

```
dplyr:::mutate(efficient = if_else(mpg>30, TRUE, FALSE))
```

## Tidyverse syntax

### SUMMARY STATISTICS:

one continuous variable:

```
mtcars %>%
```

one categorical variable:

```
mtcars %>%
```

two categorical variables:

```
mtcars %>%
```

one continuous, one categorical:

```
mtcars %>%
```

tilde

one continuous variable:

```
ggplot(x=mpg, data=mtcars, geom = "histogram")
```

```
ggplot(x=mpg, data=mtcars, geom = "boxplot")
```

one categorical variable:

```
ggplot2::qplot(x=cyl, data=mtcars, geom="bar")
```

two continuous variables:

```
ggplot2::qplot(x=disp, y=mpg, data=mtcars, geom="point")
```

two categorical variables:

```
ggplot2::qplot(x=factor(cyl), data=mtcars, geom="bar") +
```

```
facet_grid(.~am)
```

one continuous, one categorical:

```
ggplot2::qplot(x=disp, data=mtcars, geom = "histogram") +
```

```
facet_grid(.~cyl)
```

```
ggplot2::qplot(y=disp, x=factor(cyl), data=mtcars,
```

```
geom="boxplot")
```

### WRANGLING:

subsetting:

```
mtcars %>%
```

making a new variable:

```
mtcars <- mtcars %>%
```

```
dplyr:::mutate(efficient = if_else(mpg>30, TRUE, FALSE))
```

I've given you a copy of this cheatsheet in the "cheatsheets" folder of your workspace,  
and I have paper copies for anyone who wants one

Read across the cheatsheet to see how different  
syntaxes approach the same problem

the pipe