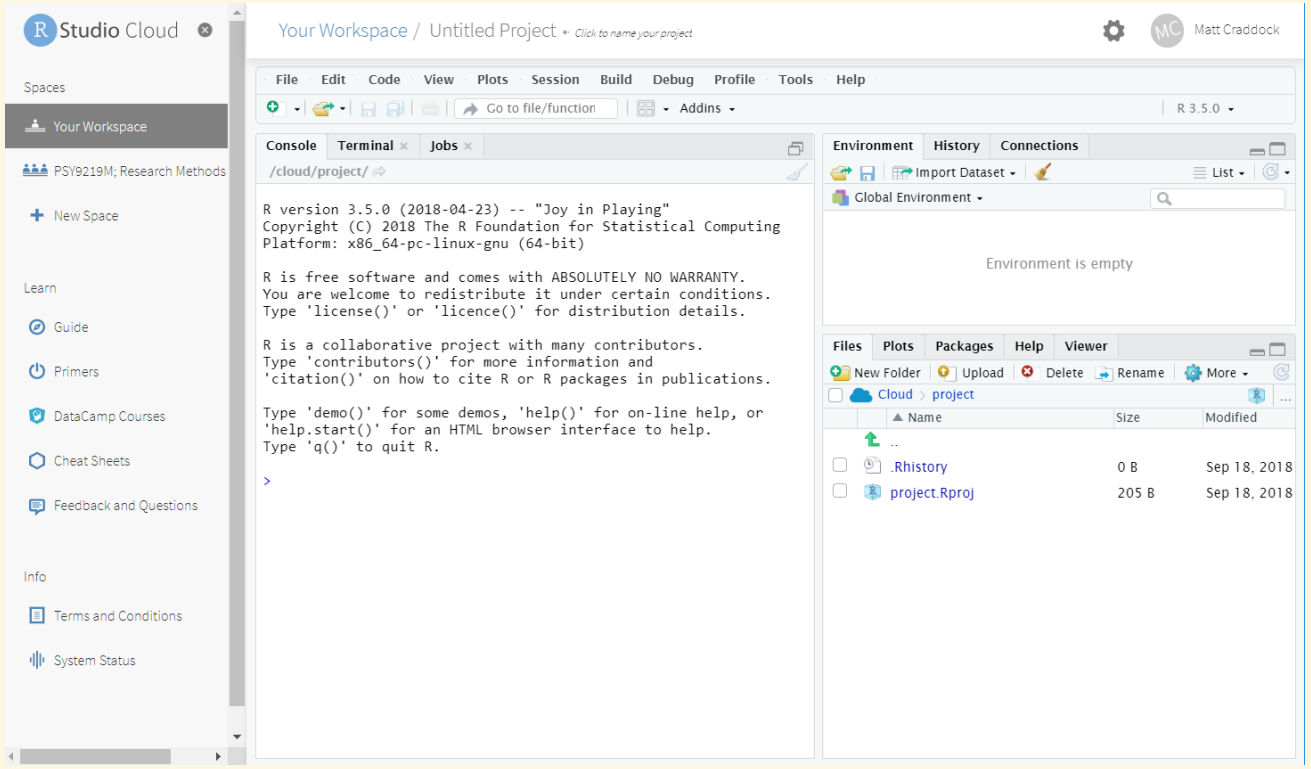
Introduction to R, part 2

Research Methods and Skills

20/10/2020

Interacting with R

- The R Console
  - REPL: Read Evaluate Print Loop
  - Type stuff in, it tries to do it



Basic use of R

Use of R like a calculator

The R console allows you to use it like a calculator, as below:

```
5 + 5
```

```
## [1] 10
```

```
10 - 6 * 13
```

```
## [1] -68
```

Basic use of R

Creating objects to store information

You assign values to objects using <-

```
test_object <- 5
```

<- can be read as "is now", making the code above roughly mean

```
The object "test_object" is now 5 # Do not run!
```

Objects "stand-in" for their values:

```
test_object
```

```
## [1] 5
```

Basic use of R

Creation of vectors

Vectors are simply a 1-dimensional collection of values of the same type.

E.g. We can create a numeric vector using the **c()** function.

```
c(5, 10, 3, -1, -5)
```

```
## [1]  5 10  3 -1 -5
```

This is a one-dimensional vector of length *five*, since it has 5 values.

Basic use of R

Using functions on objects

Functions do things to objects.

Brackets after a word in these slides indicate that something is a function, e.g. **c()**, **mean()**

```
mean(c(5, 8, 2, 4, 5))
```

```
## [1] 4.8
```

```
test_object <- c(5, 8, 2, 4, 5)
mean(test_object)
```

```
## [1] 4.8
```

R Scripts

R Scripts

Scripts are a way of writing out a sequence of commands that you want R to execute.

A typical script looks something like this:

```
# Load in required packages using library()
library(tidyverse)

# Define any custom functions here (we haven't covered this!)

# Now load any data you want to work on. (again, we'll cover this later!)
test_data <-
  read_csv("data/a-random-RT-file.csv") %>% # I'll explain what %>% means later
  rename(RT = `reaction times`)

# The rest of the script then runs whatever analyses or plotting you want to do
ggplot(test_data,
       aes(x = RT,
           fill = viewpoint)) +
  geom_density()
```

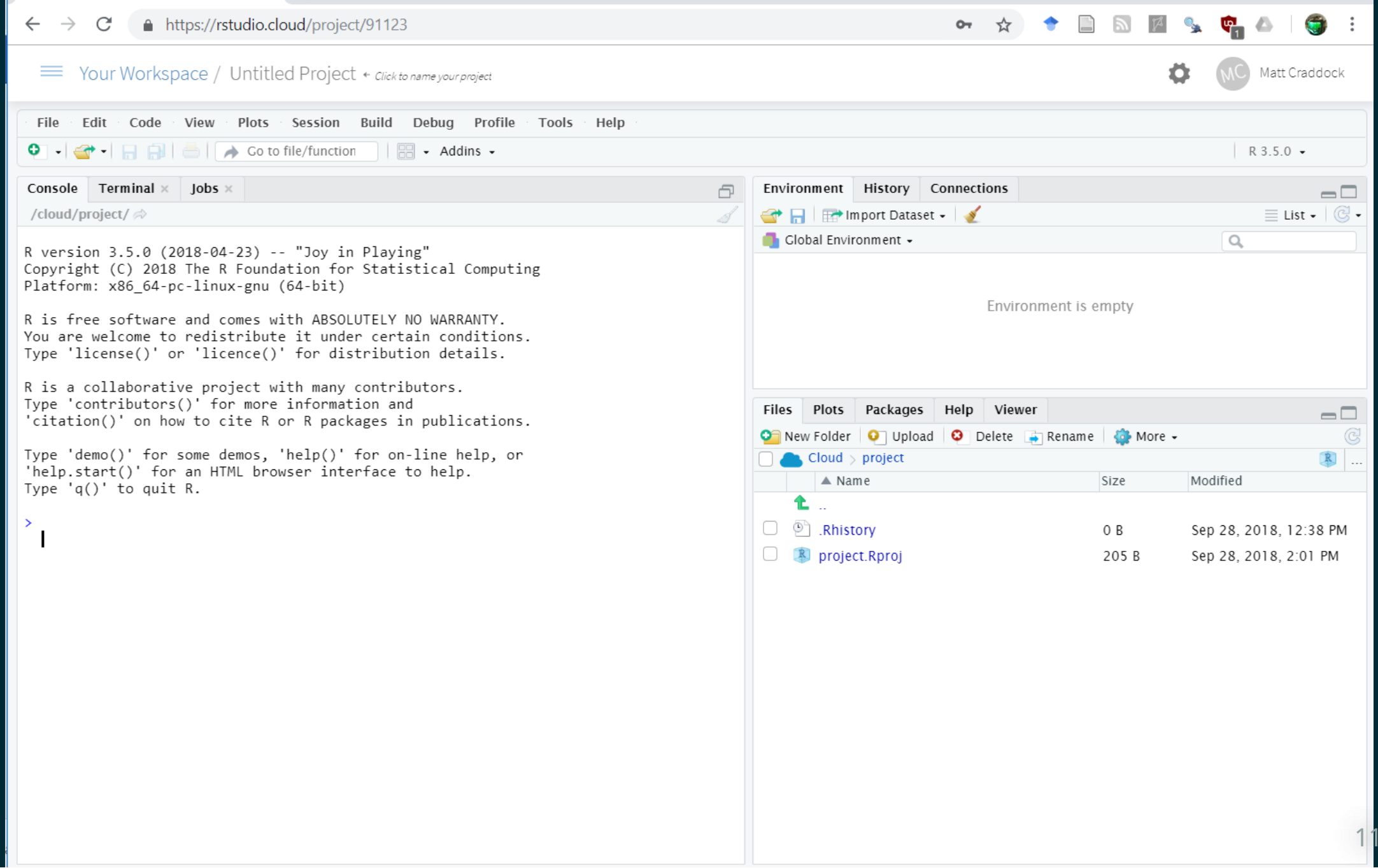
Why is this useful?

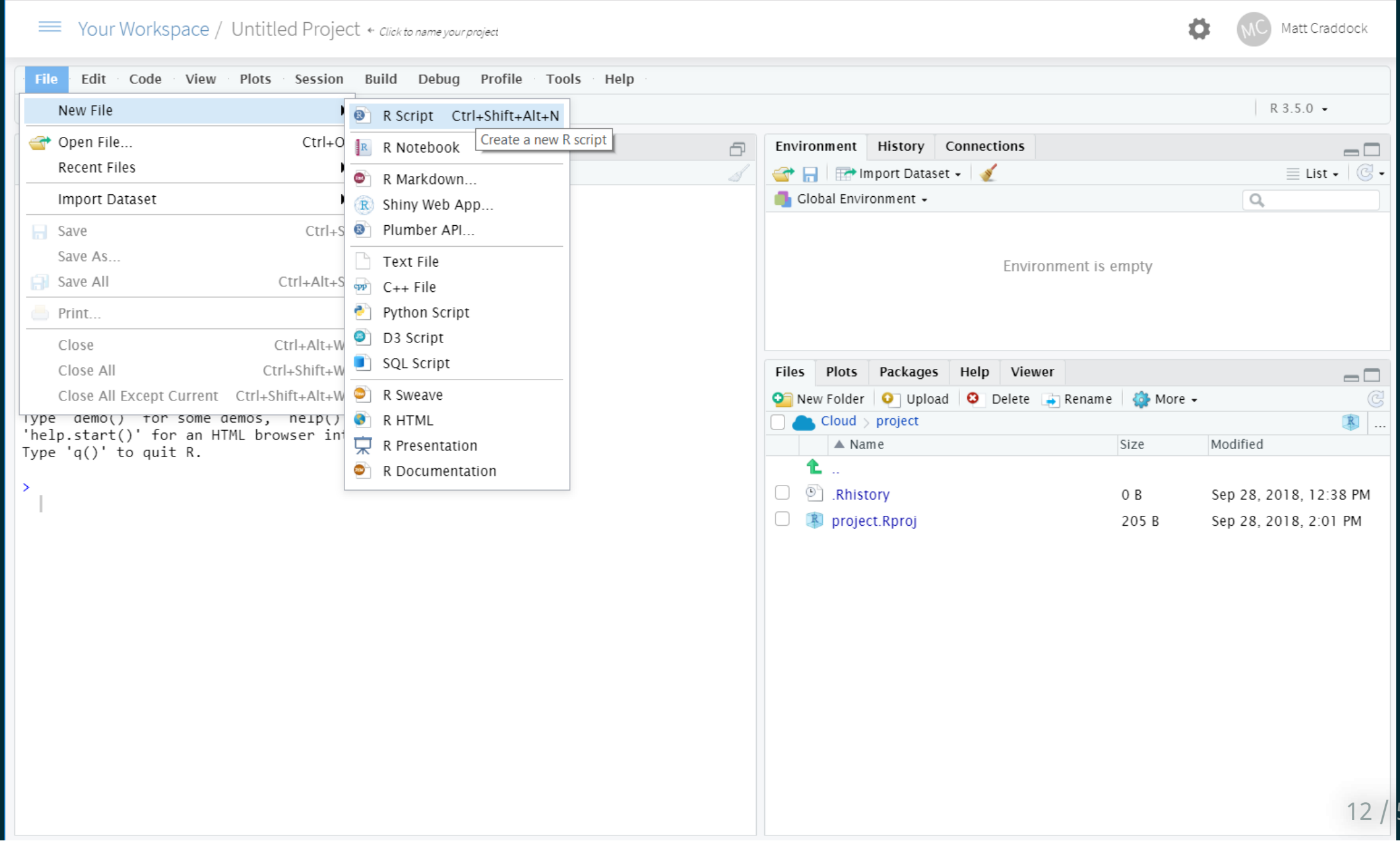
Somebody asks you how you performed a particular analysis. In particular, they want detailed instructions of how you created a plot, filtered out outliers or missing data, and performed a linear regression.

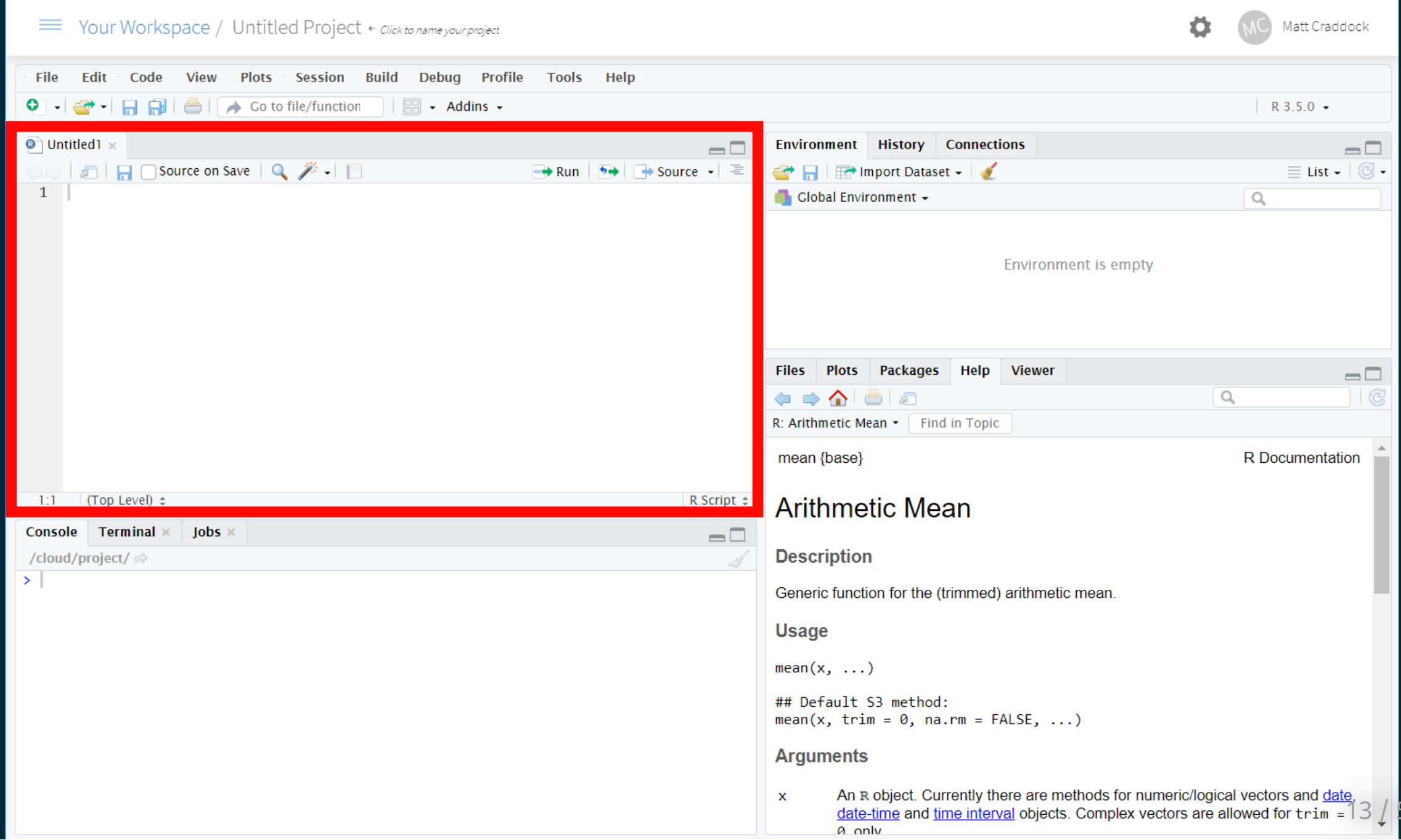
Q1: *How would you do that if you used SPSS?*

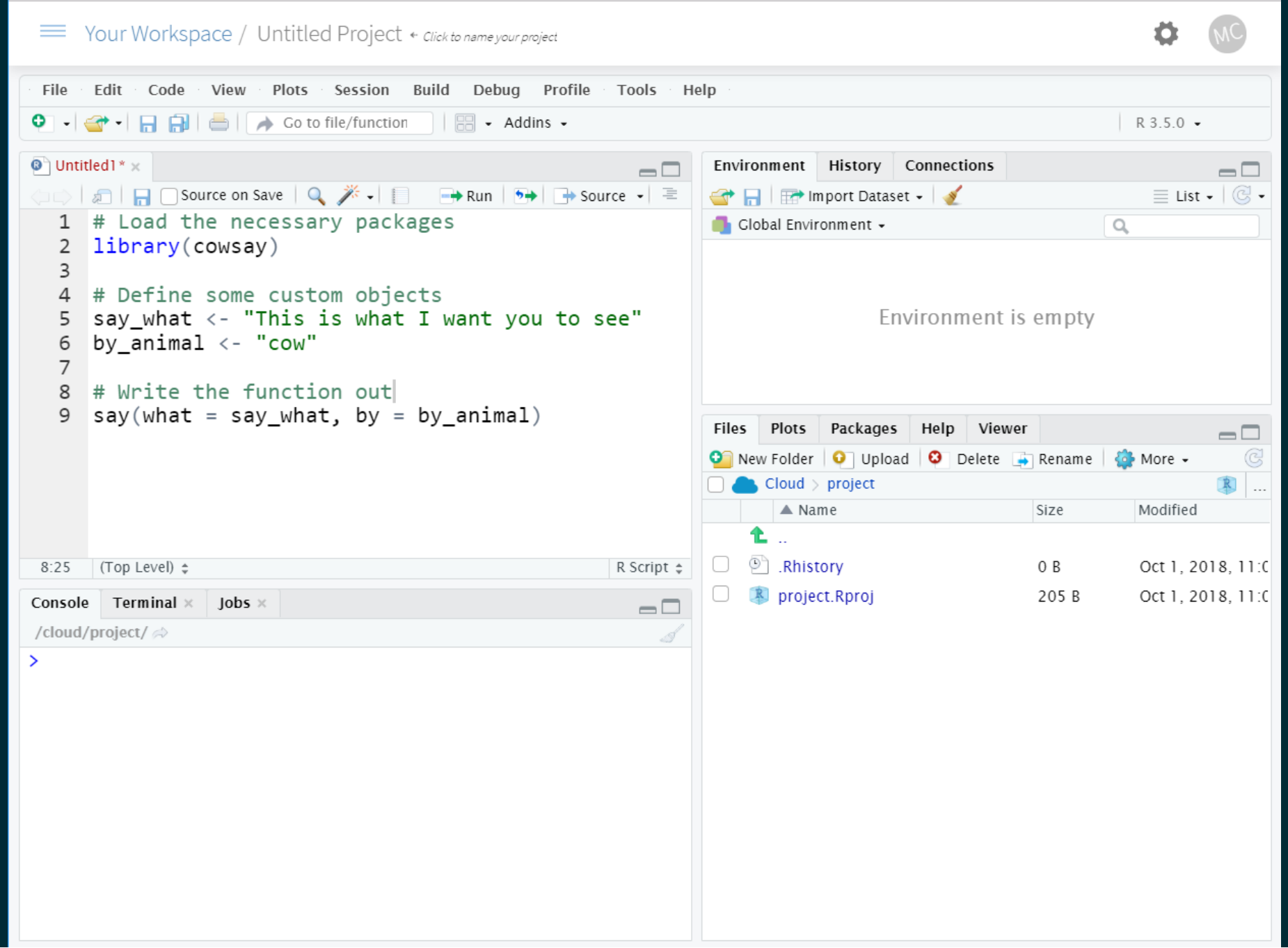
Q2: *How would you do that if you used R?*

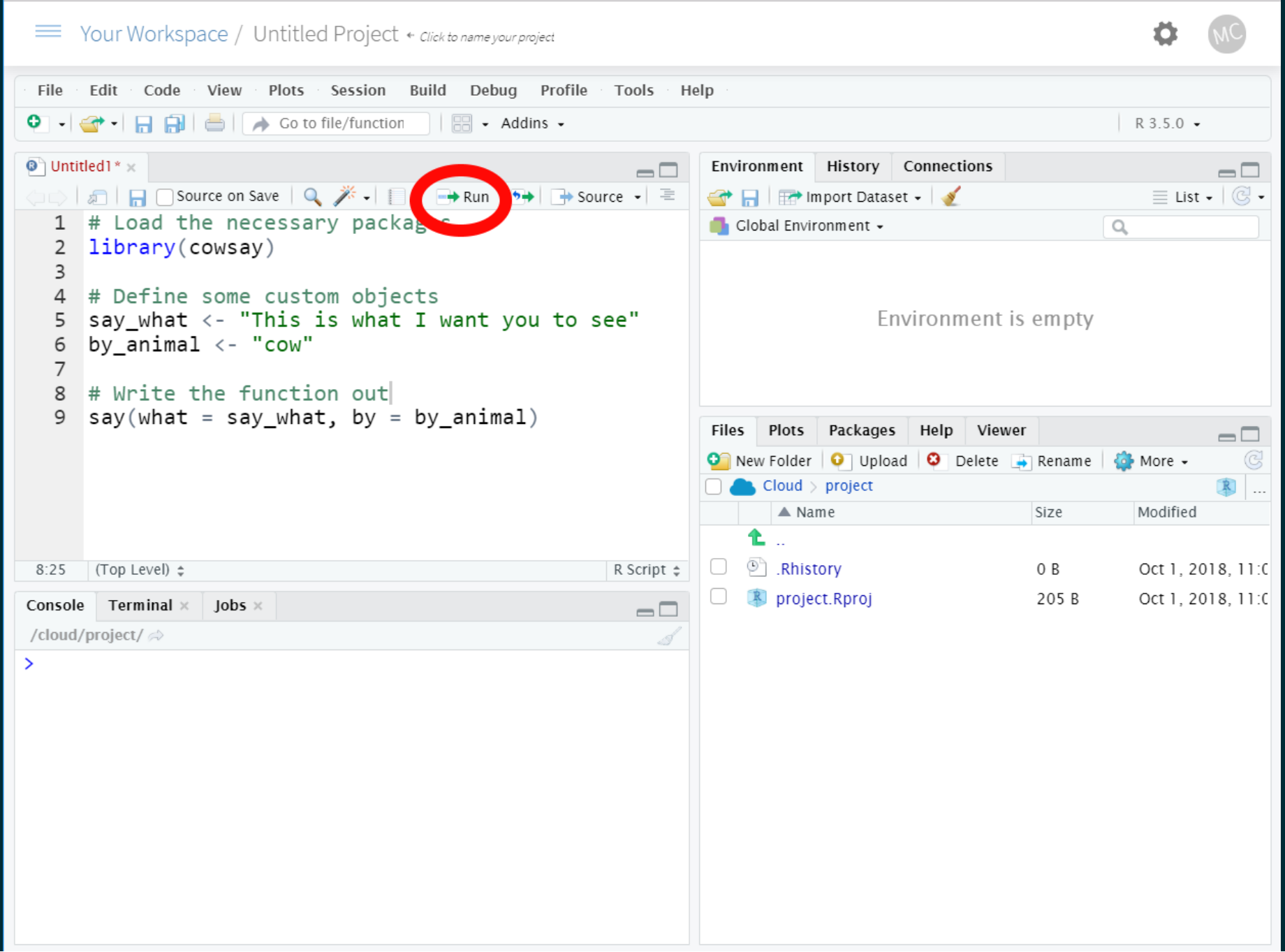
Creating R Scripts

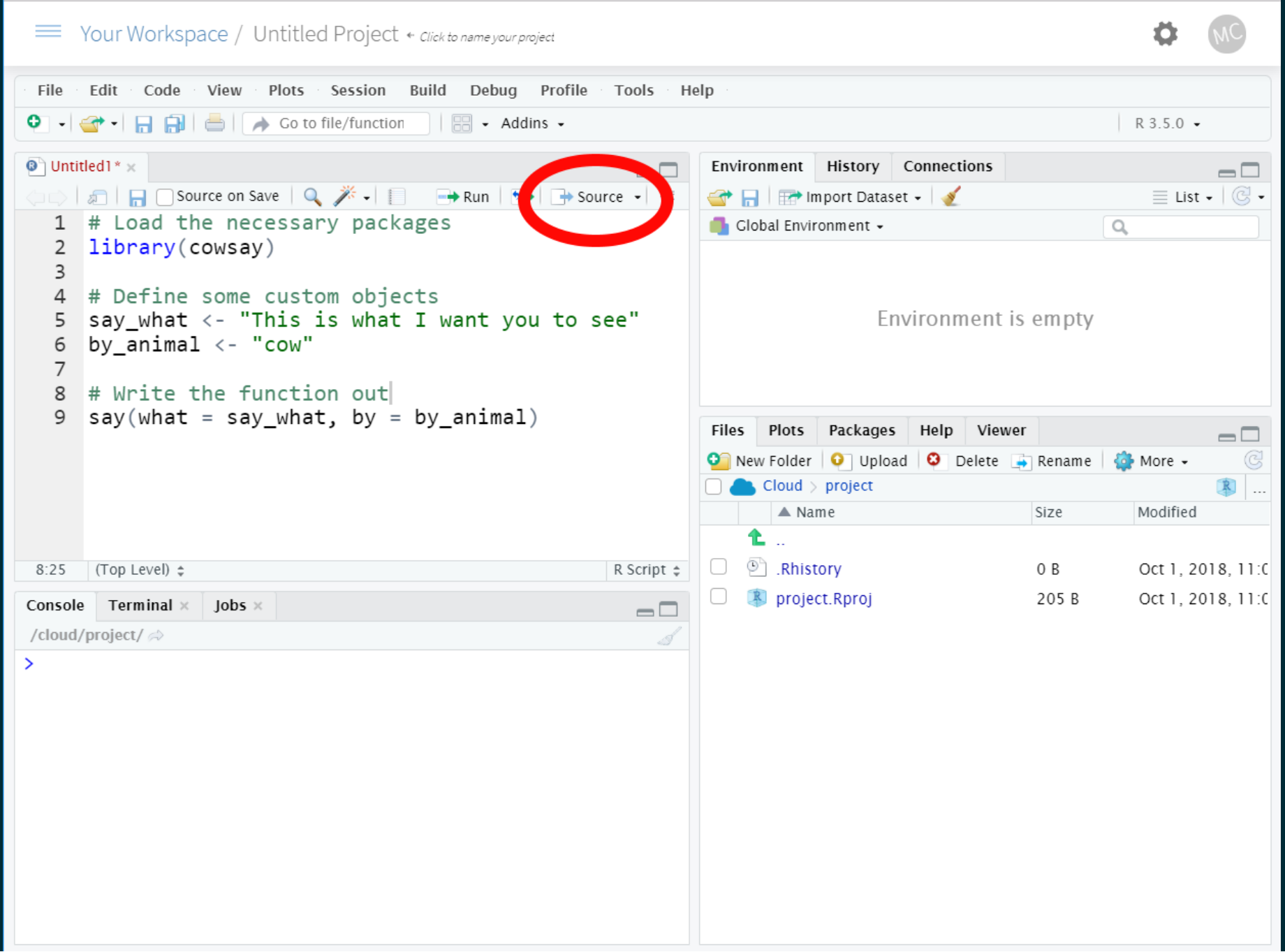

https://rstudio.cloud/project/91123
Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Console Terminal Jobs
/cloud/project/
R version 3.5.0 (2018-04-23) -- "Joy in Playing"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-pc-linux-gnu (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.
>
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
..
.Rhistory 0 B Sep 28, 2018, 12:38 PM
project.Rproj 205 B Sep 28, 2018, 2:01 PM

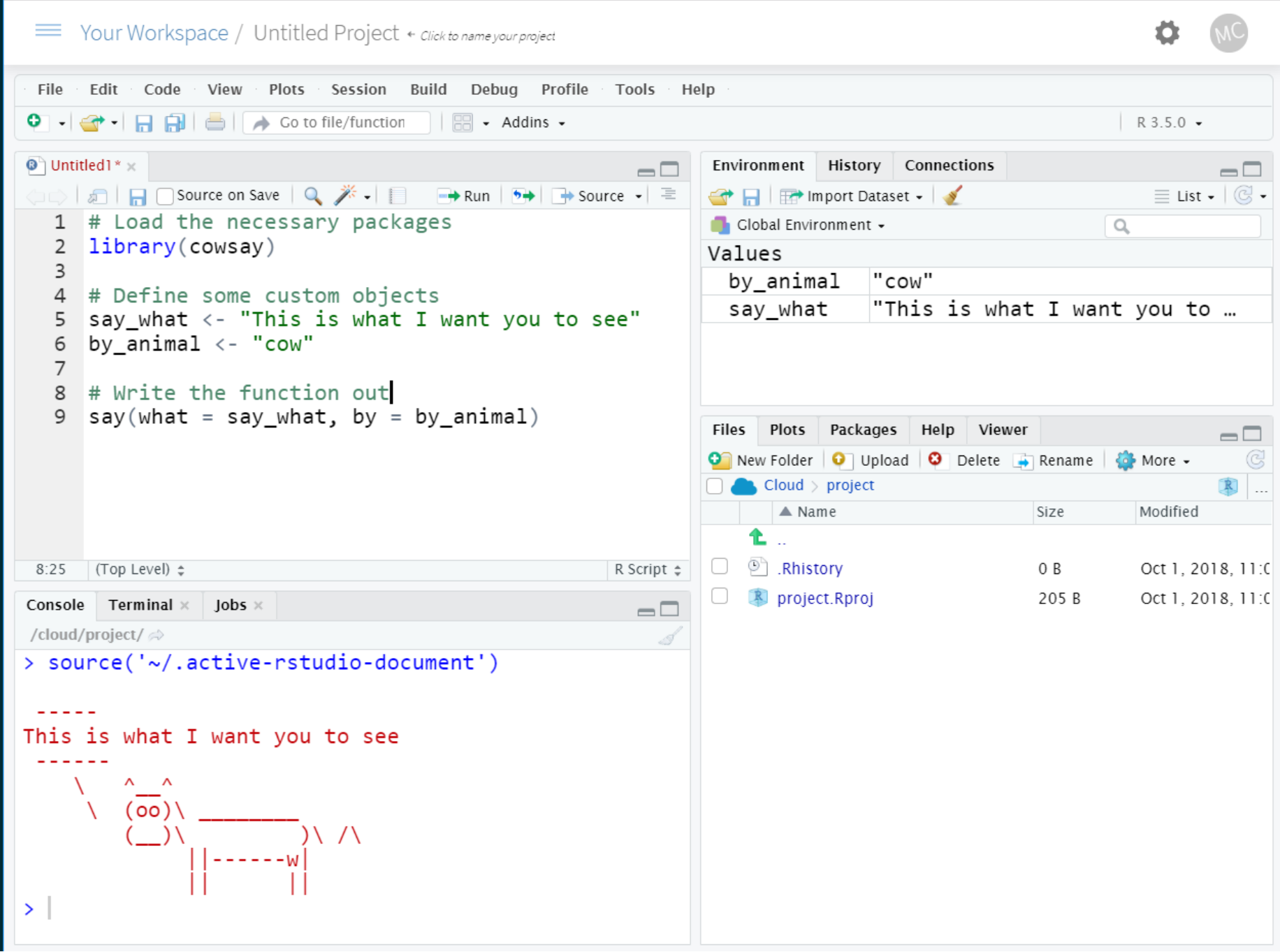

Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
New File
Open File... Ctrl+O
Recent Files
Import Dataset
Save Ctrl+S
Save As...
Save All Ctrl+Alt+S
Print...
Close Ctrl+Alt+W
Close All Ctrl+Shift+W
Close All Except Current Ctrl+Shift+Alt+W
R Script Ctrl+Shift+Alt+N
Create a new R script
R Notebook
R Markdown...
Shiny Web App...
Plumber API...
Text File
C++ File
Python Script
D3 Script
SQL Script
R Sweave
R HTML
R Presentation
R Documentation
R 3.5.0
'help.start()' for an HTML browser int
Type 'q()' to quit R.
>
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
.Rhistory 0 B Sep 28, 2018, 12:38 PM
project.Rproj 205 B Sep 28, 2018, 2:01 PM


Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1
Source on Save
Run
Source
1
1:1
(Top Level)
R Script
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
R: Arithmetic Mean
Find in Topic
mean {base}
R Documentation
Arithmetic Mean
Description
Generic function for the (trimmed) arithmetic mean.
Usage
mean(x, ...)
Default S3 method:
mean(x, trim = 0, na.rm = FALSE, ...)
Arguments
x
An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for trim =
Console Terminal Jobs
/cloud/project/
>


Your Workspace / Untitled Project Click to name your project
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1 *
Source on Save
Run
Source
1 # Load the necessary packages
2 library(cowsay)
3
4 # Define some custom objects
5 say_what <- "This is what I want you to see"
6 by_animal <- "cow"
7
8 # Write the function out
9 say(what = say_what, by = by_animal)
8:25
(Top Level)
R Script
Console
Terminal
Jobs
/cloud/project/
>
Environment
History
Connections
Import Dataset
List
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
Cloud
project
Name
Size
Modified
..
.Rhistory
0 B
Oct 1, 2018, 11:0
project.Rproj
205 B
Oct 1, 2018, 11:0


Your Workspace / Untitled Project Click to name your project
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1 *
Source on Save
Run
Source
1 # Load the necessary packag
2 library(cowsay)
3
4 # Define some custom objects
5 say_what <- "This is what I want you to see"
6 by_animal <- "cow"
7
8 # Write the function out
9 say(what = say_what, by = by_animal)
8:25 (Top Level) R Script
Console Terminal Jobs
/cloud/project/
>
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
..
.Rhistory 0 B Oct 1, 2018, 11:0
project.Rproj 205 B Oct 1, 2018, 11:0


Your Workspace / Untitled Project Click to name your project
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1 *
Source on Save
Run
Source
1 # Load the necessary packages
2 library(cowsay)
3
4 # Define some custom objects
5 say_what <- "This is what I want you to see"
6 by_animal <- "cow"
7
8 # Write the function out
9 say(what = say_what, by = by_animal)
8:25 (Top Level) R Script
Console Terminal Jobs
/cloud/project/
>
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
..
.Rhistory 0 B Oct 1, 2018, 11:0
project.Rproj 205 B Oct 1, 2018, 11:0


Your Workspace / Untitled Project Click to name your project
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1*
Source on Save
Run
Source
1 # Load the necessary packages
2 library(cowsay)
3
4 # Define some custom objects
5 say_what <- "This is what I want you to see"
6 by_animal <- "cow"
7
8 # Write the function out
9 say(what = say_what, by = by_animal)
8:25
(Top Level)
R Script
Console
Terminal
Jobs
/cloud/project/
> source('~/.active-rstudio-document')
This is what I want you to see
Environment
History
Connections
Import Dataset
List
Global Environment
Values
by_animal "cow"
say_what "This is what I want you to …
Files
Plots
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
Cloud
project
Name
Size
Modified
.Rhistory 0 B Oct 1, 2018, 11:0
project.Rproj 205 B Oct 1, 2018, 11:0

R Markdown

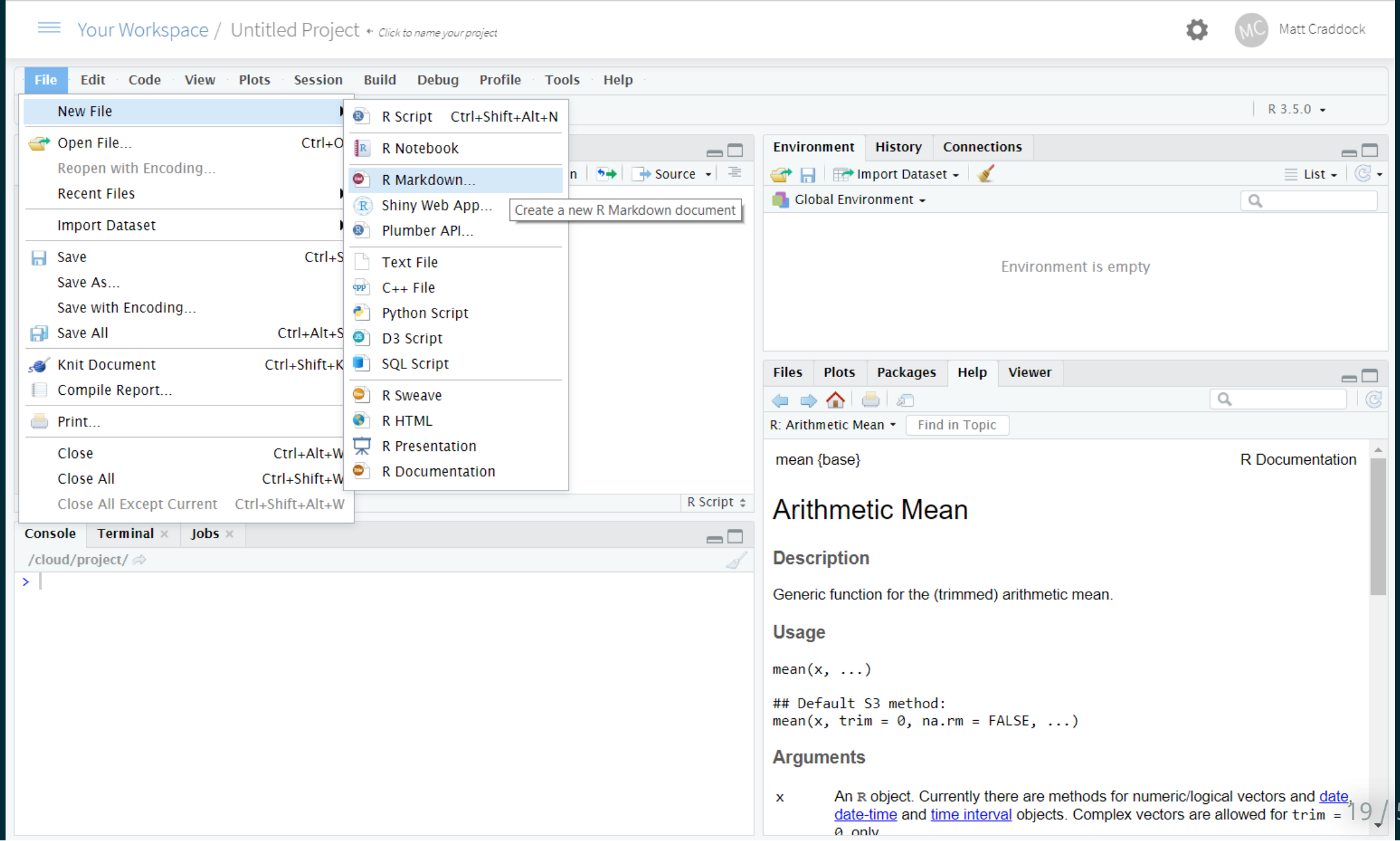
Literate programming is a mixture of plain text and code.

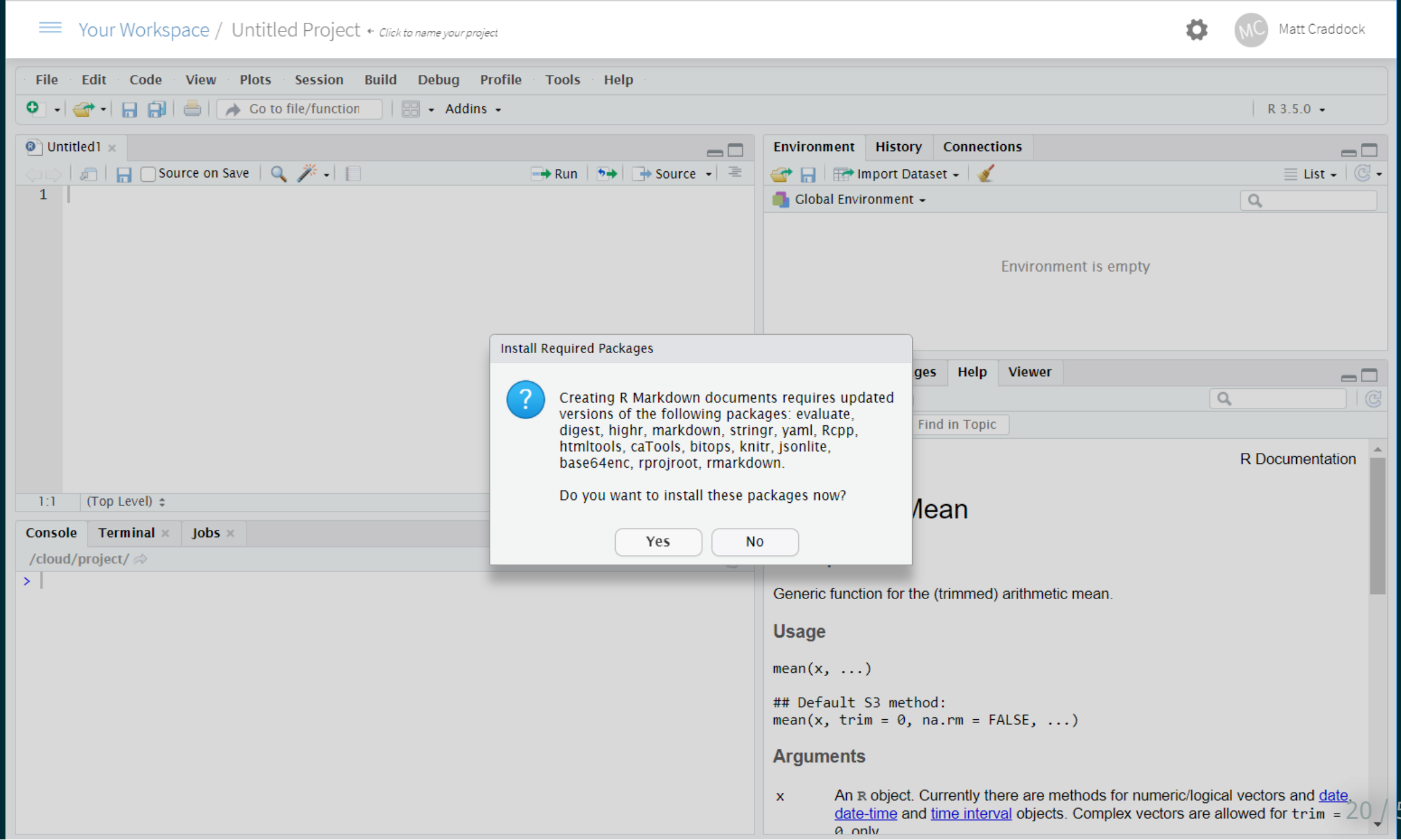
Whereas in scripts you need to use the **#** symbol to indicate comments, as here

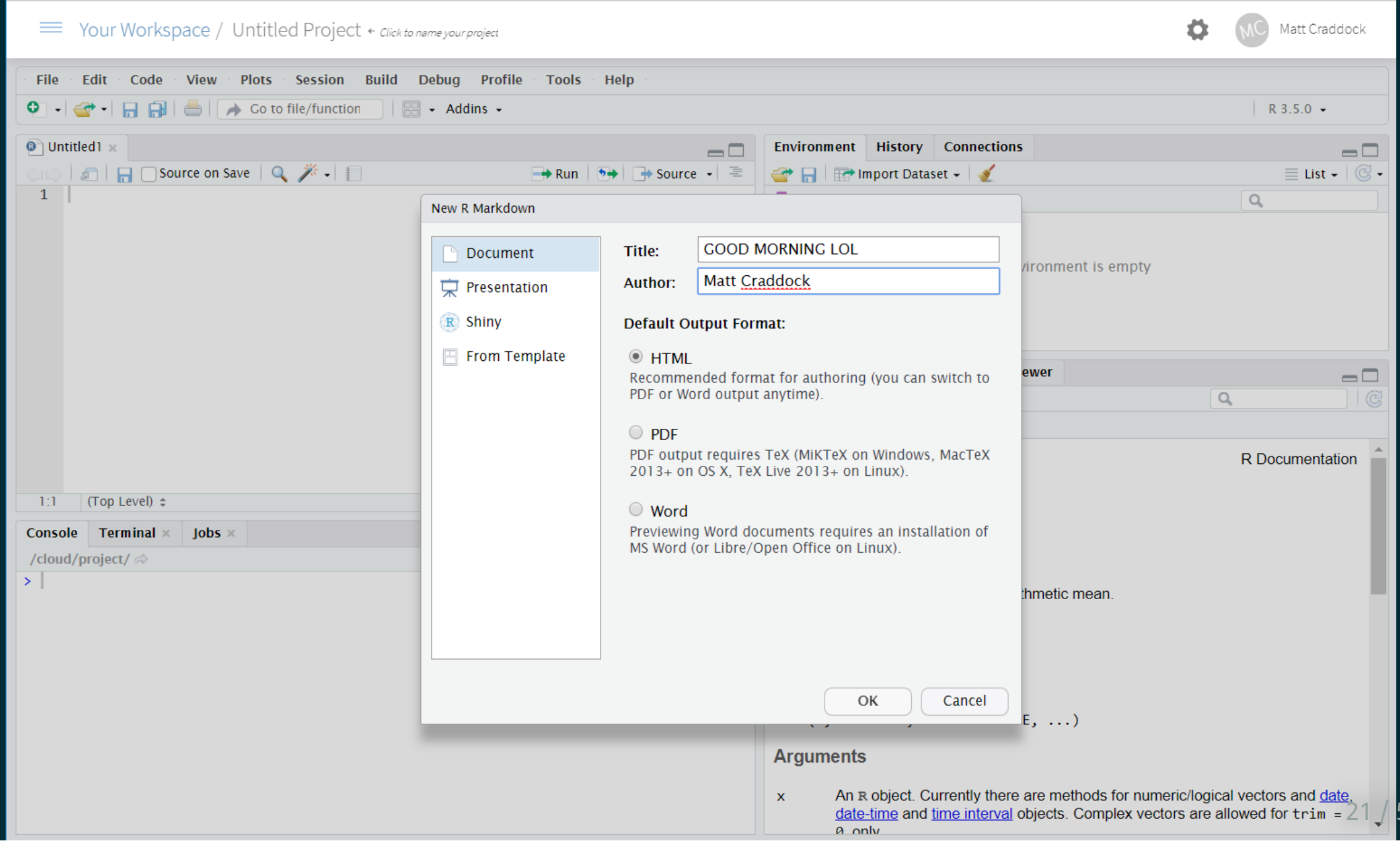
```
# This is a comment
```

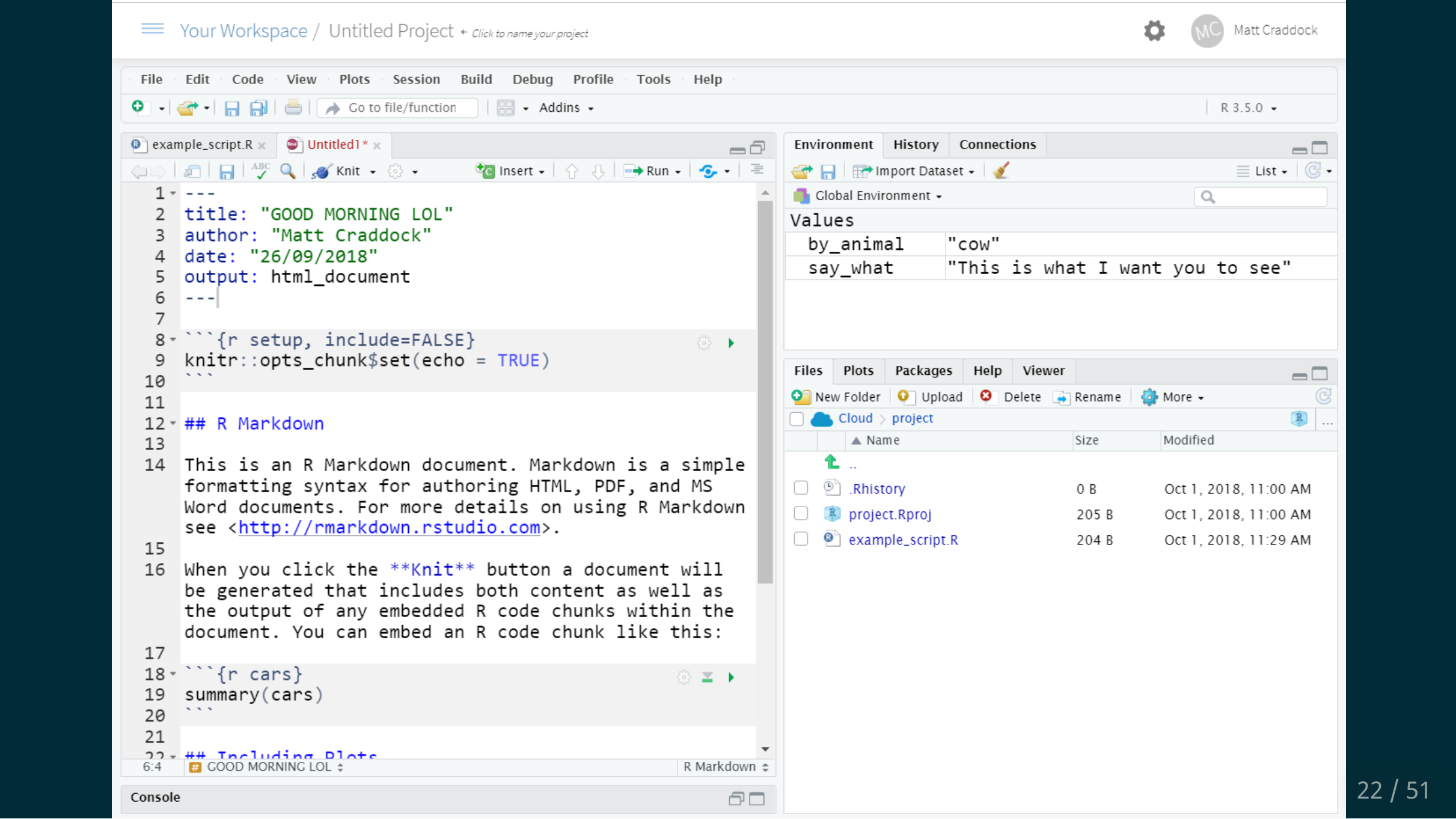
...with R Markdown you can mix plain text and code using **chunks** to delineate sections of code.

This allows you to create elaborate documents following the structure *you* want!

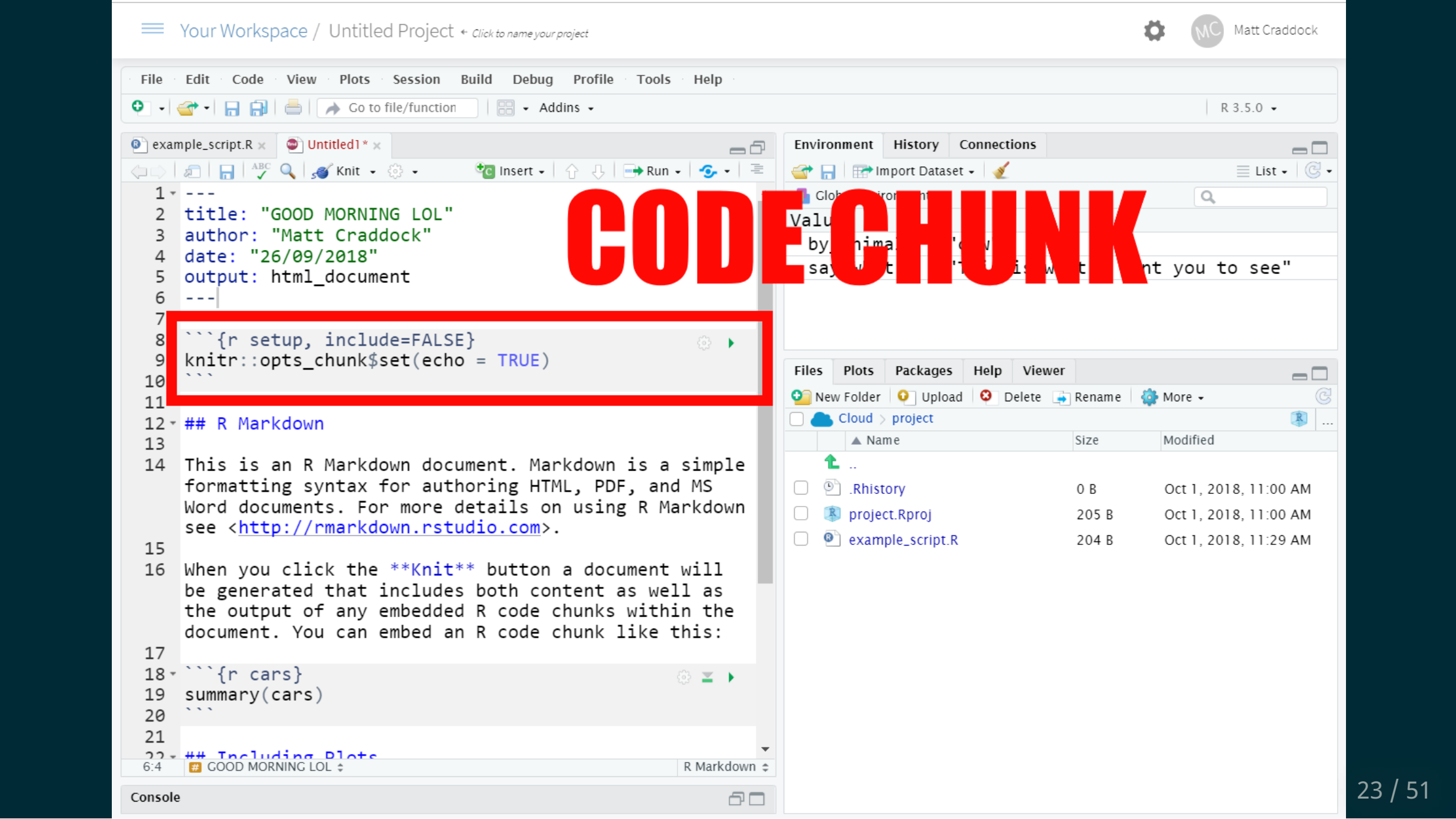

Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
New File
Open File... Ctrl+O
Reopen with Encoding...
Recent Files
Import Dataset
Save Ctrl+S
Save As...
Save with Encoding...
Save All Ctrl+Alt+S
Knit Document Ctrl+Shift+K
Compile Report...
Print...
Close Ctrl+Alt+W
Close All Ctrl+Shift+W
Close All Except Current Ctrl+Shift+Alt+W
R Script Ctrl+Shift+Alt+N
R Notebook
R Markdown...
Shiny Web App...
Plumber API...
Text File
C++ File
Python Script
D3 Script
SQL Script
R Sweave
R HTML
R Presentation
R Documentation
Create a new R Markdown document
R 3.5.0
Source
R Script
Console Terminal Jobs
/cloud/project/
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
R: Arithmetic Mean
Find in Topic
mean {base}
R Documentation
Arithmetic Mean
Description
Generic function for the (trimmed) arithmetic mean.
Usage
mean(x, ...)
Default S3 method:
mean(x, trim = 0, na.rm = FALSE, ...)
Arguments
x
An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for trim = 0 only


Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1
Source on Save
Run
Source
1
1:1
(Top Level)
Console
Terminal
Jobs
/cloud/project/
>
Environment
History
Connections
Import Dataset
List
Global Environment
Environment is empty
Help
Viewer
Find in Topic
R Documentation
Mean
Generic function for the (trimmed) arithmetic mean.
Usage
mean(x, ...)
Default S3 method:
mean(x, trim = 0, na.rm = FALSE, ...)
Arguments
x
An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for trim = 0 only.
Install Required Packages
Creating R Markdown documents requires updated versions of the following packages: evaluate, digest, highr, markdown, stringr, yaml, Rcpp, htmltools, caTools, bitops, knitr, jsonlite, base64enc, rprojroot, rmarkdown.
Do you want to install these packages now?
Yes
No

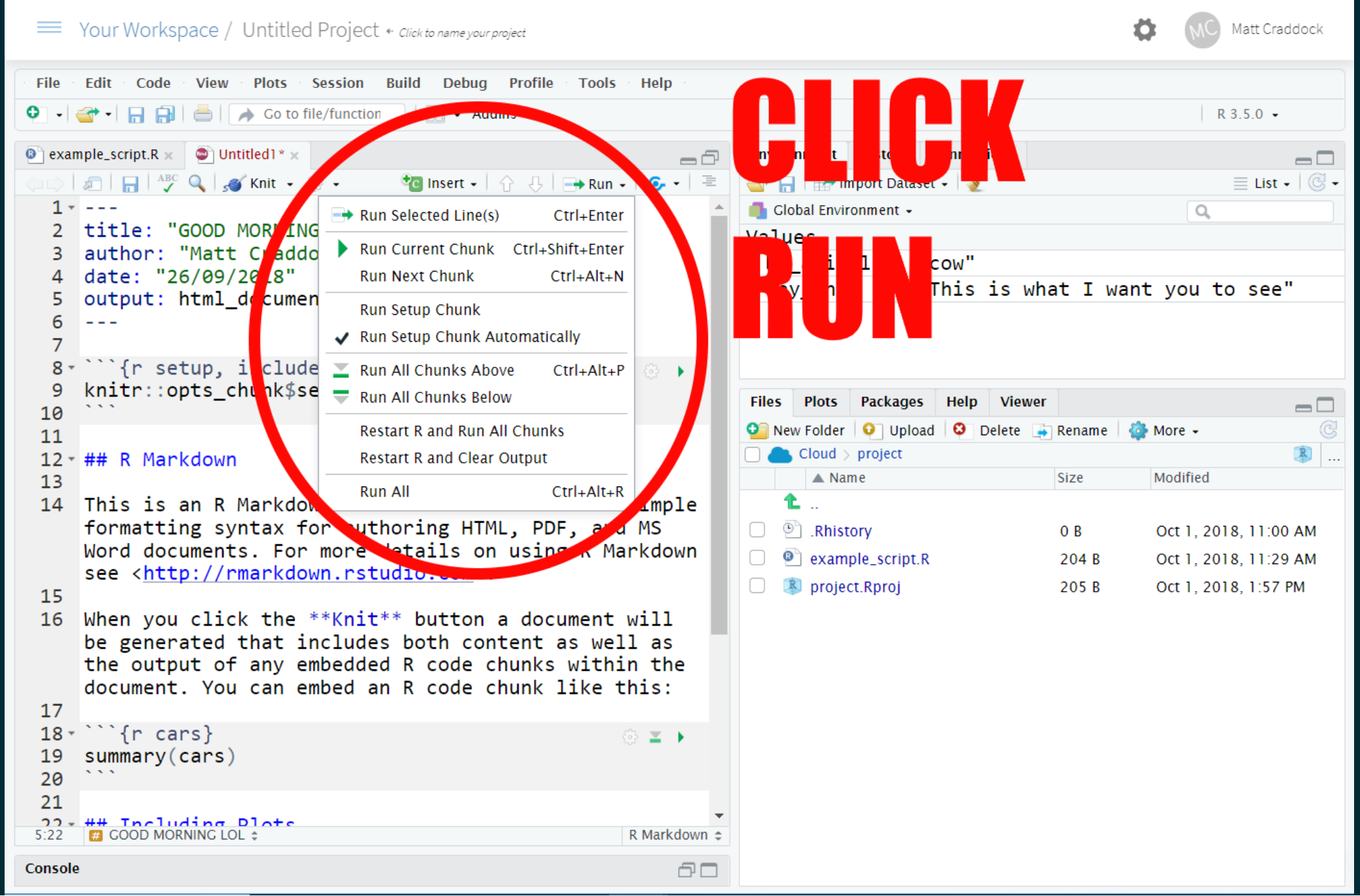

Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
Untitled1
Source on Save
Run
Source
1
1:1
(Top Level)
Console
Terminal
Jobs
/cloud/project/
>
Environment
History
Connections
Import Dataset
List
New R Markdown
Document
Presentation
Shiny
From Template
Title:
GOOD MORNING LOL
Author:
Matt Craddock
Default Output Format:
HTML
Recommended format for authoring (you can switch to PDF or Word output anytime).
PDF
PDF output requires TeX (MiKTeX on Windows, MacTeX 2013+ on OS X, TeX Live 2013+ on Linux).
Word
Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux).
OK
Cancel
R Documentation
Arguments
x
An R object. Currently there are methods for numeric/logical vectors and date, date-time and time interval objects. Complex vectors are allowed for trim =

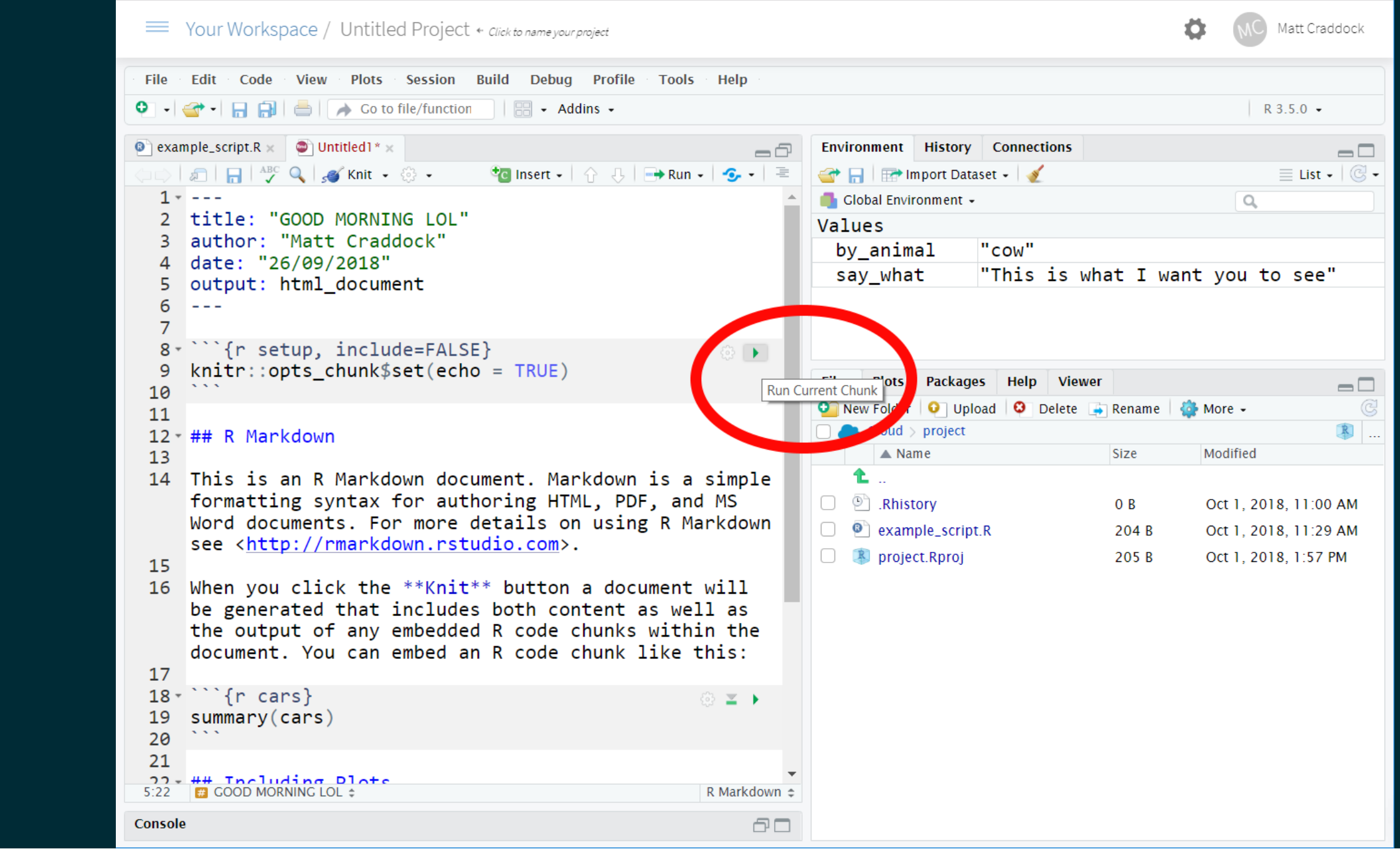

Your Workspace / Untitled Project
Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
example_script.R
Untitled1*
Knit
Insert
Run
---
title: "GOOD MORNING LOL"
author: "Matt Craddock"
date: "26/09/2018"
output: html_document

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```{r cars}
summary(cars)
```
Including Plots
6:4
GOOD MORNING LOL
R Markdown
Console
Environment History Connections
Import Dataset
List
Global Environment
Values
by_animal "cow"
say_what "This is what I want you to see"
Files Plots Packages Help Viewer
New Folder
Upload
Delete
Rename
More
Cloud project
Name Size Modified
.Rhistory 0 B Oct 1, 2018, 11:00 AM
project.Rproj 205 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM

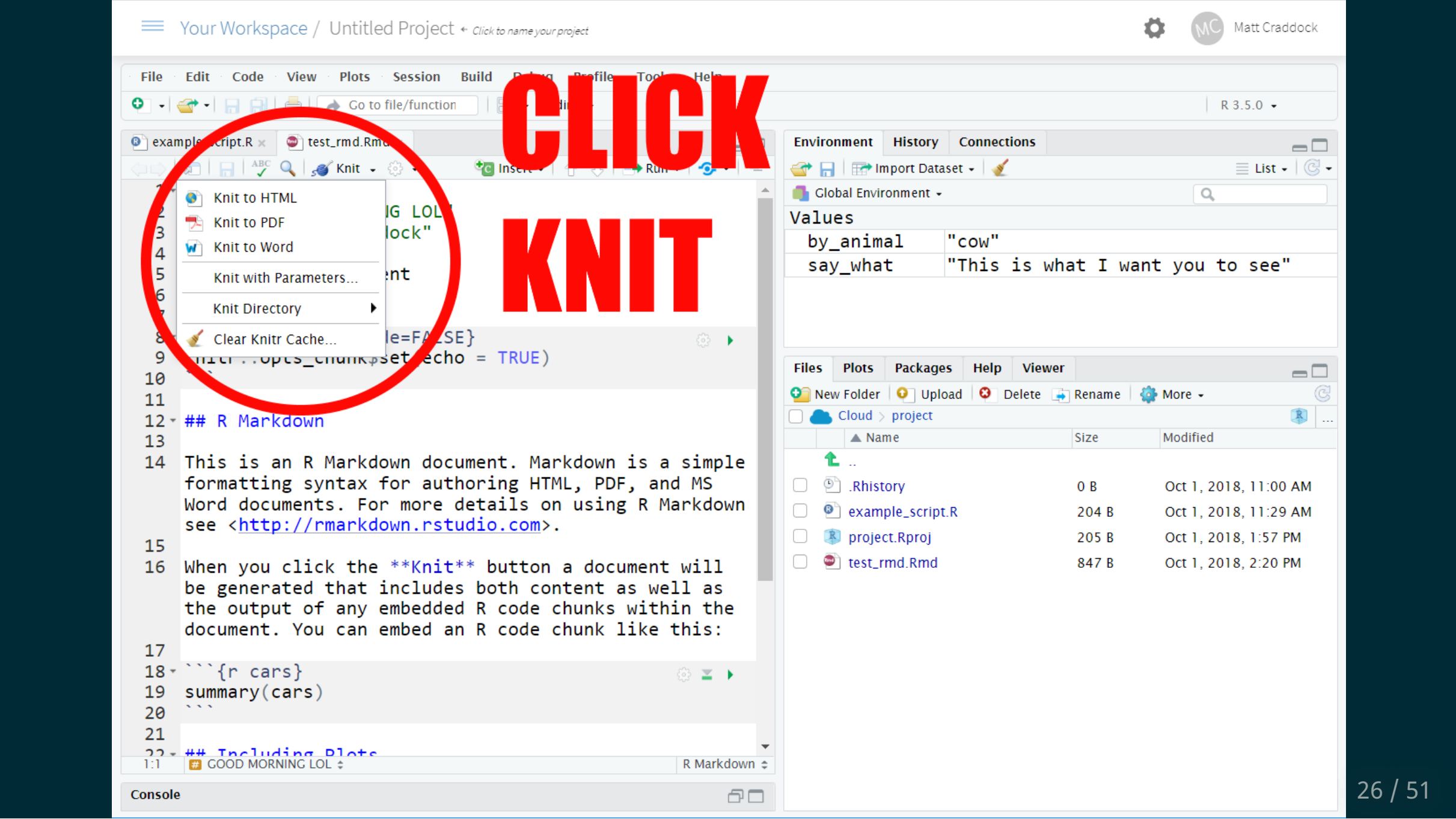

Your Workspace / Untitled Project
Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
example_script.R
Untitled1*
Knit
Insert
Run
CODE CHUNK
---
title: "GOOD MORNING LOL"
author: "Matt Craddock"
date: "26/09/2018"
output: html_document

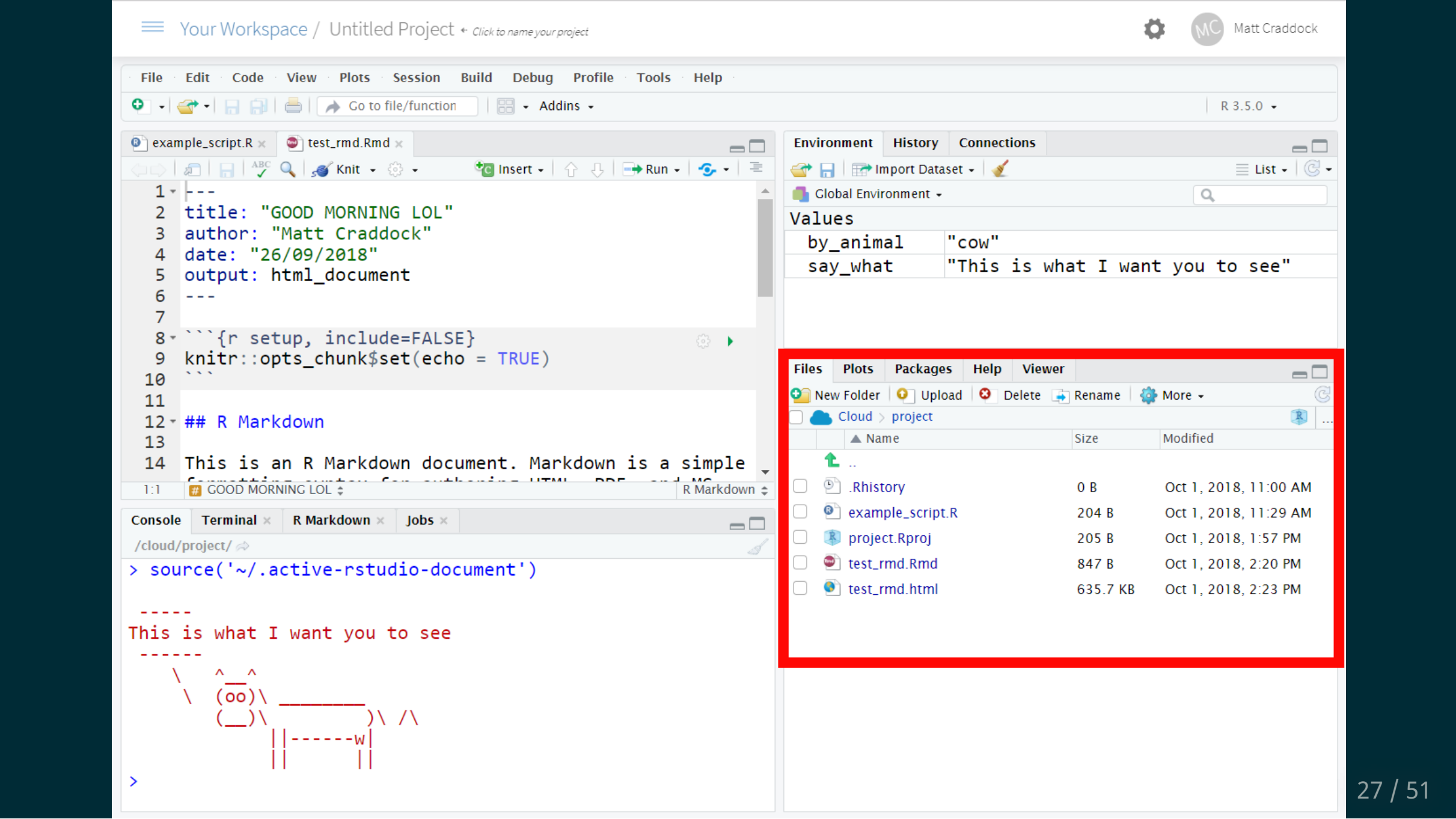
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```{r cars}
summary(cars)
```
Including Plots
6:4
GOOD MORNING LOL
R Markdown
Console
Environment History Connections
Import Dataset
List
you to see"
Files Plots Packages Help Viewer
New Folder
Upload
Delete
Rename
More
Cloud
project
Name
Size
Modified
.Rhistory 0 B Oct 1, 2018, 11:00 AM
project.Rproj 205 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM


Your Workspace / Untitled Project
Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
R 3.5.0
example_script.R
Untitled1*
Knit
Insert
Run
Run Selected Line(s) Ctrl+Enter
Run Current Chunk Ctrl+Shift+Enter
Run Next Chunk Ctrl+Alt+N
Run Setup Chunk
Run Setup Chunk Automatically
Run All Chunks Above Ctrl+Alt+P
Run All Chunks Below
Restart R and Run All Chunks
Restart R and Clear Output
Run All Ctrl+Alt+R
CLICK
RUN
Import Dataset
List
Global Environment
Values
cow"
This is what I want you to see"
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
.Rhistory 0 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM
project.Rproj 205 B Oct 1, 2018, 1:57 PM
5:22 GOOD MORNING LOL R Markdown
Console


Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
example_script.R
Untitled1*
Knit
Insert
Run
---
title: "GOOD MORNING LOL"
author: "Matt Craddock"
date: "26/09/2018"
output: html_document

```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```{r cars}
summary(cars)
```
Including Plots
Run Current Chunk
5:22
GOOD MORNING LOL
R Markdown
Console
Environment
History
Connections
Import Dataset
List
Global Environment
Values
by_animal "cow"
say_what "This is what I want you to see"
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
project
Name
Size
Modified
.Rhistory 0 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM
project.Rproj 205 B Oct 1, 2018, 1:57 PM


Your Workspace / Untitled Project
Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
R 3.5.0
CLICK
KNIT
Knit to HTML
Knit to PDF
Knit to Word
Knit with Parameters...
Knit Directory
Clear Knitr Cache...
R Markdown
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```{r cars}
summary(cars)
```
Including Plots
GOOD MORNING LOL
R Markdown
Console
Environment History Connections
Import Dataset
List
Global Environment
Values
by_animal "cow"
say_what "This is what I want you to see"
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
.Rhistory 0 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM
project.Rproj 205 B Oct 1, 2018, 1:57 PM
test_rmd.Rmd 847 B Oct 1, 2018, 2:20 PM


Your Workspace / Untitled Project Click to name your project
Matt Craddock
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
example_script.R
test_rmd.Rmd
Knit
Insert
Run
1 ---
2 title: "GOOD MORNING LOL"
3 author: "Matt Craddock"
4 date: "26/09/2018"
5 output: html_document
6 ---
7
8 ```{r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10 ```
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a simple
1:1
GOOD MORNING LOL
R Markdown
Console
Terminal
R Markdown
Jobs
/cloud/project/
> source('~/.active-rstudio-document')

This is what I want you to see

\ ^__^
\ (oo)\ _______
(__)\ )\ /\
||------w|
|| ||
>
Environment
History
Connections
Import Dataset
List
Global Environment
Values
by_animal "cow"
say_what "This is what I want you to see"
Files
Plots
Packages
Help
Viewer
New Folder
Upload
Delete
Rename
More
Cloud > project
Name
Size
Modified
..
.Rhistory 0 B Oct 1, 2018, 11:00 AM
example_script.R 204 B Oct 1, 2018, 11:29 AM
project.Rproj 205 B Oct 1, 2018, 1:57 PM
test_rmd.Rmd 847 B Oct 1, 2018, 2:20 PM
test_rmd.html 635.7 KB Oct 1, 2018, 2:23 PM

Some very important advice

R Markdown documents are like *recipes*.

Every step needs to be written down.

When you press the knit button, R forgets everything and follows the instructions line-by-line.

So be thorough, and write down everything in the order you want it to happen!

(One exception: NEVER use install.packages() in a script)

Code-along exercise!

Basic data types

Basic data types

There are five basic data types in R:

Type	Description	Examples
integer	Whole numbers	1, 2, 3
numeric	Any real number, fractions	3.4, 2, -2.3
character	Text	"Hi there", "8.5", "ABC123"
logical	Assertion of truth/falsity	TRUE, FALSE
complex	Real and imaginary numbers	0.34+5.3i

There are some additional types to be aware of, particularly *factors*, but we'll come back to them in a later session.

Checking data types

We can use the **class()** function to check what type a given object is.

```
class(10)
```

```
## [1] "numeric"
```

```
class(10L) # using L after the number turns it into an *integer*
```

```
## [1] "integer"
```

```
class(TRUE)
```

```
## [1] "logical"
```

```
class("Wednesday")
```

```
## [1] "character"
```

Basic containers

Vectors

A vector is a collection of values which all have the same basic **type**.

A numeric vector is thus a collection of numeric values:

```
some_numbers <- c(5, 3, 6, 8)
some_numbers
```

```
## [1] 5 3 6 8
```

... and a character vector is a collection of character values

```
char_example <- c("Monday", "Tuesday", "Wednesday", "Thursday")
char_example
```

```
## [1] "Monday"    "Tuesday"   "Wednesday" "Thursday"
```

More about vectors

The colon (:) operator can be used to produce a sequence of numbers:

```
one_to_ten <- 1:10
one_to_ten
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

Vectors can also be given names:

```
one_to_four <- 1:4
names(one_to_four) <- char_example
one_to_four
```

```
##    Monday   Tuesday Wednesday  Thursday
##         1         2         3         4
```

Extracting values

Sometimes you only want a specific subset of a vector. For example, suppose that you only want the third value. For this, we need the **[]** (square brackets) operator.

We put an *index* inbetween the **[]** operator.

```
char_example[3]
```

```
## [1] "Wednesday"
```

Note that you can also supply *multiple* values:

```
char_example[2:3]
```

```
## [1] "Tuesday"   "Wednesday"
```

```
char_example[c(2, 4)]
```

```
## [1] "Tuesday"  "Thursday"
```

Extracting values

If your vector is *named*, you can also use the names as *indices*.

```
one_to_four
```

```
##    Monday   Tuesday Wednesday  Thursday
##         1         2         3         4
```

```
one_to_four["Wednesday"]
```

```
## Wednesday
##         3
```

```
one_to_four[c("Monday", "Wednesday")]
```

```
##    Monday Wednesday
##         1         3
```

Matrices

Matrices

Matrices are 2-dimensional collections of values.

All values must be of the same type.

```
matrix(1:9, nrow = 3, ncol = 3)
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

This is quite a common format. For example, each row could represent an individual participant. Each column could represent a different numerical measure.

Accessing matrices

Since matrices are two-dimensional, you need to give two indices to make sure you get the value you want. Again, you can use the [] operator.

```
[row, col]
```

Here I extract the number from the 2nd row down, 3rd column across.

```
test_matrix <- matrix(1:9, nrow = 3, ncol = 3)
test_matrix
```

```
##      [,1] [,2] [,3]
## [1,]    1    4    7
## [2,]    2    5    8
## [3,]    3    6    9
```

```
test_matrix[2, 3]
```

```
## [1] 8
```

Lists

Lists

Lists are a collection of objects of varying length and type.

```
album_list <-
  list(The_Beatles = c(
    "Sgt. Pepper",
    "The White Album",
    "Revolver",
    "Abbey Road"),
    Nirvana = c(
      "Bleach",
      "Nevermind",
      "In Utero")
    )
```

Each element is labelled, just like a mason jar on a shelf.

Each element has different contents, just like our mason jars.

Lists

```
names(album_list)
```

```
## [1] "The_Beatles" "Nirvana"
```

```
length(album_list)
```

```
## [1] 2
```

```
album_list["The_Beatles"]
```

```
## $The_Beatles
## [1] "Sgt. Pepper"     "The White Album" "Revolver"        "Abbey Road"
```

Tabular data

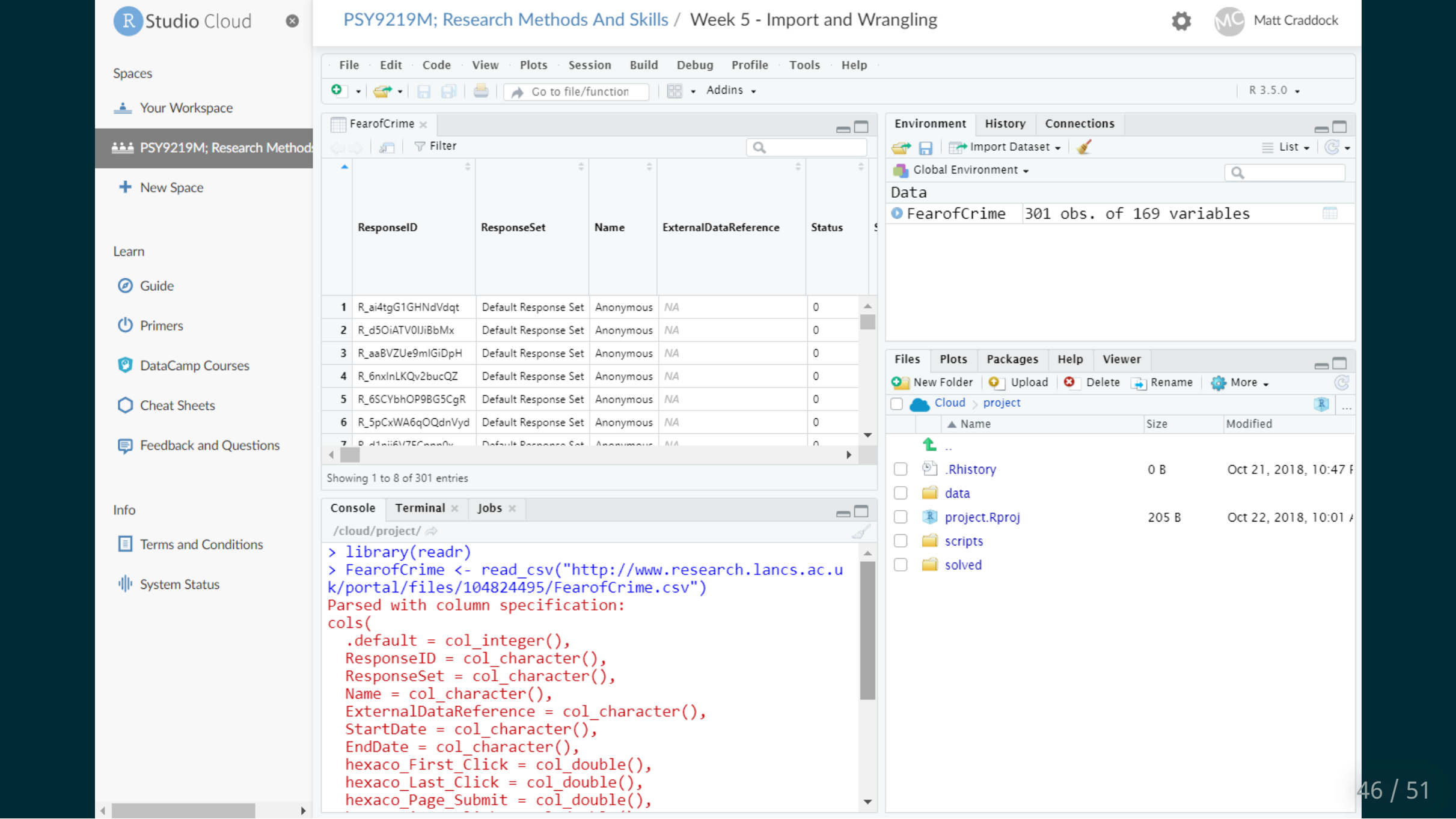
Tabular data is also collection of different types of data, arranged in a rectangular, tabular format. Most of the data you encounter in psychology is in this kind of format.

In tabular data, each column contains only values of one *type*, and each row thus contains different types of information about one thing.

Show 5 entries Search:

	mpg	cyl	disp	hp	drat
Mazda RX4	21	6	160	110	3.9
Mazda RX4 Wag	21	6	160	110	3.9
Datsun 710	22.8	4	108	93	3.85
Hornet 4 Drive	21.4	6	258	110	3.08
Hornet Sportabout	18.7	8	360	175	3.15

Showing 1 to 5 of 32 entries Previous 1 2 3 4 5 6 7 Next


Studio Cloud
PSY9219M; Research Methods And Skills / Week 5 - Import and Wrangling
Matt Craddock
Spaces
Your Workspace
PSY9219M; Research Methods
New Space
Learn
Guide
Primers
DataCamp Courses
Cheat Sheets
Feedback and Questions
Info
Terms and Conditions
System Status
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
R 3.5.0
FearofCrime
Filter
ResponseID
ResponseSet
Name
ExternalDataReference
Status
1 R_ai4tgG1GHNdVdqt Default Response Set Anonymous NA 0
2 R_d5OiATV0IJiBbMx Default Response Set Anonymous NA 0
3 R_aaBVZUe9mlGiDpH Default Response Set Anonymous NA 0
4 R_6nxlnLKQv2bucQZ Default Response Set Anonymous NA 0
5 R_6SCYbhOP9BG5CgR Default Response Set Anonymous NA 0
6 R_5pCxWA6qOQdnVyd Default Response Set Anonymous NA 0
Showing 1 to 8 of 301 entries
Environment History Connections
Import Dataset
List
Global Environment
Data
FearofCrime 301 obs. of 169 variables
Files Plots Packages Help Viewer
New Folder Upload Delete Rename More
Cloud project
Name Size Modified
.Rhistory 0 B Oct 21, 2018, 10:47
data
project.Rproj 205 B Oct 22, 2018, 10:01
scripts
solved
Console Terminal Jobs
/cloud/project/
> library(readr)
> FearofCrime <- read_csv("http://www.research.lancs.ac.uk/portal/files/104824495/FearofCrime.csv")
Parsed with column specification:
cols(
.default = col_integer(),
ResponseID = col_character(),
ResponseSet = col_character(),
Name = col_character(),
ExternalDataReference = col_character(),
StartDate = col_character(),
EndDate = col_character(),
hexaco_First_Click = col_double(),
hexaco_Last_Click = col_double(),
hexaco_Page_Submit = col_double(),

Creating tabular data

In R, this type of structure is called a *data frame*.

```
days_of_the_week <-
  data.frame(day_name = c("Sunday",
                          "Monday",
                          "Tuesday",
                          "Wednesday",
                          "Thursday",
                          "Friday",
                          "Saturday"),
             day_number = 1:7
             )
```

```
days_of_the_week
```

```
##    day_name day_number
## 1    Sunday          1
## 2    Monday          2
## 3   Tuesday          3
## 4 Wednesday          4
## 5  Thursday          5
## 6    Friday          6
## 7  Saturday          7
```

Extracting information from data frames

You can use the [] operator to extract single elements, rows, or columns:

```
days_of_the_week[1, 2]
```

```
## [1] 1
```

```
days_of_the_week[5, ]
```

```
##   day_name day_number
## 5 Thursday          5
```

```
days_of_the_week[, 1]
```

```
## [1] "Sunday"    "Monday"    "Tuesday"   "Wednesday" "Thursday"  "Friday"
## [7] "Saturday"
```

Extracting information from data frames

A special operator you can use for data frame columns is the dollar sign, **$**

Combine the data frame's name with the column name as below:

```
days_of_the_week$day_name
```

```
## [1] "Sunday"    "Monday"    "Tuesday"   "Wednesday" "Thursday"  "Friday"
## [7] "Saturday"
```

Question: what **class()** is this?

Wrapping up

This week's concepts

- R Markdown - Chapter 27 of R4DS - see also https://rmarkdown.rstudio.com
- **vectors** and **lists** in Chapter 20 of R4DS

Prep for next week

- Next week we'll talk again about data frames and consider how to *structure* data.
- Look at Section 2 (Wrangle) of R4DS for information on **tibbles** (which are essentially data frames...).