ASMS 2019
ANNUAL
CONFERENCE
WORKSHOP


SOCALBIOINFORMATICS.COM

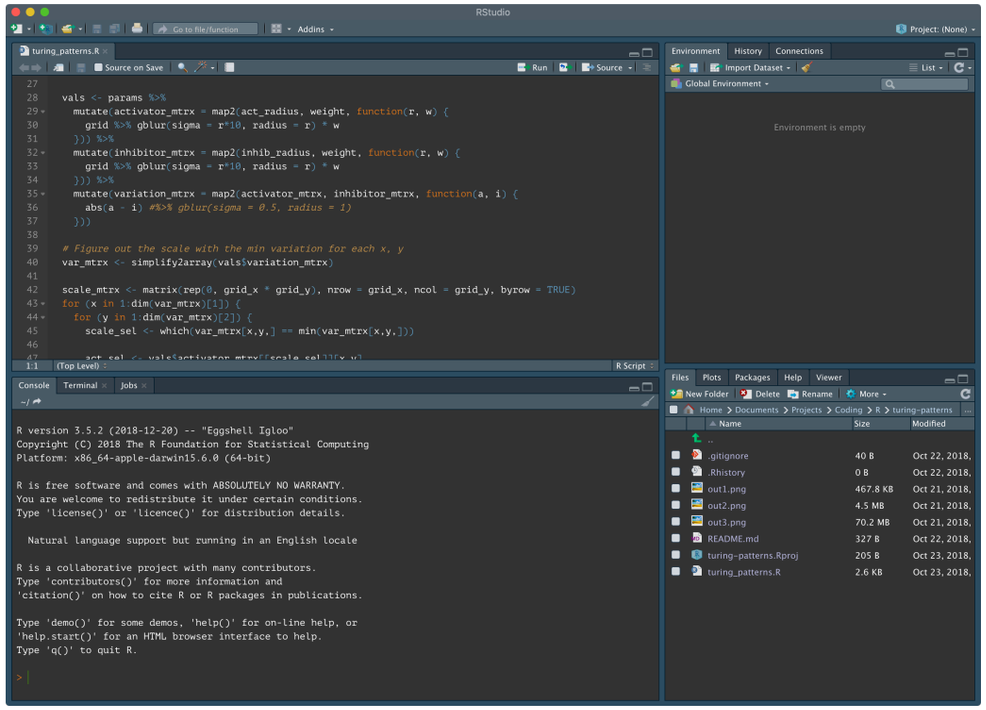
# GETTING STARTED WITH R & RSTUDIO

# BEFORE YOU START

- Confirm you have R and RStudio installed and working
- For help installing R and RStudio, check out the slides:

Installing R & RStudio

# R IS NOT RSTUDIO (AND VICE VERSA)




First, you'll install R

R is the "engine"

Then, you'll install RStudio

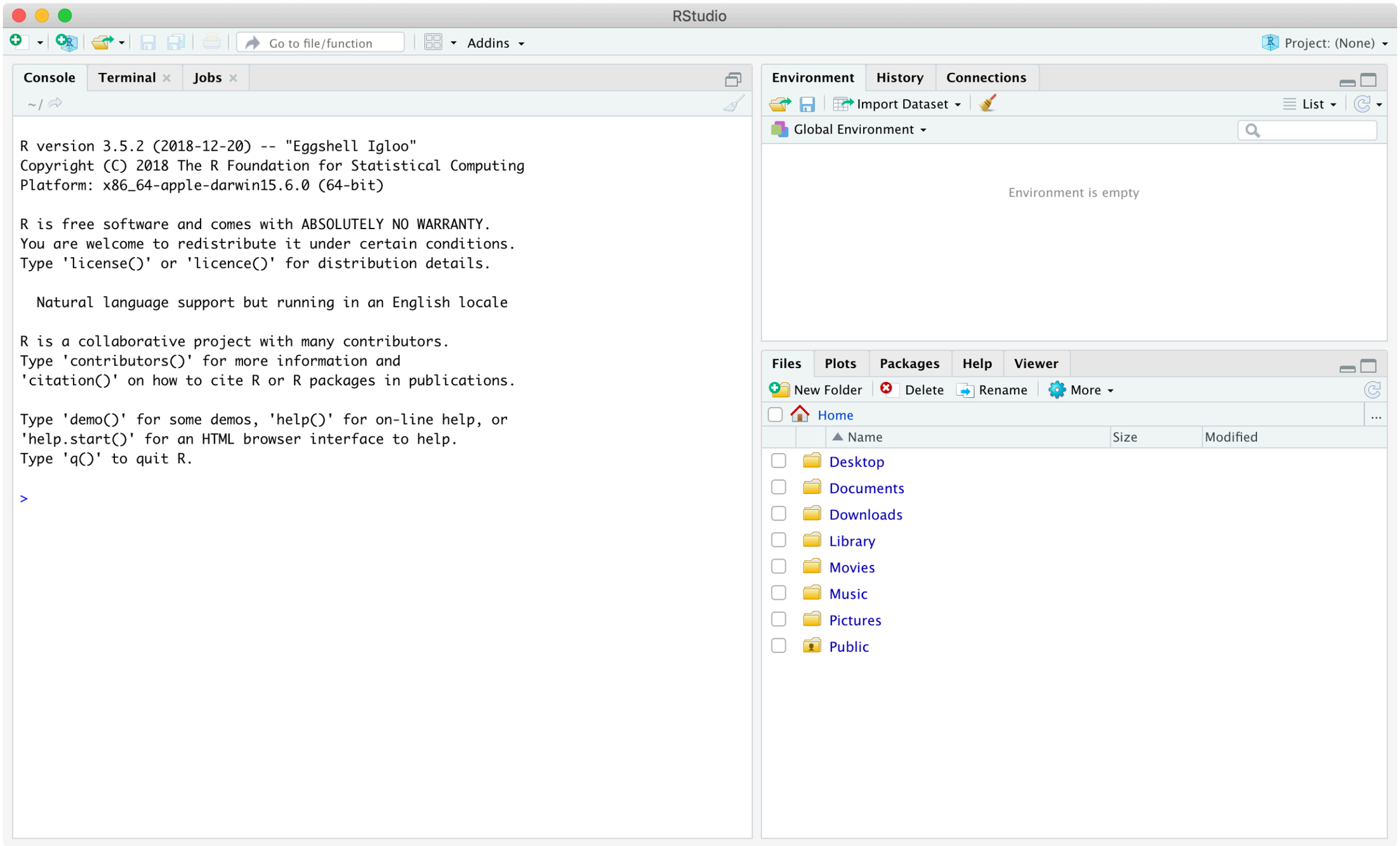
RStudio is the awesome "car body and paint job"

*There are some other fiddly bits to install too depending on the platform…*

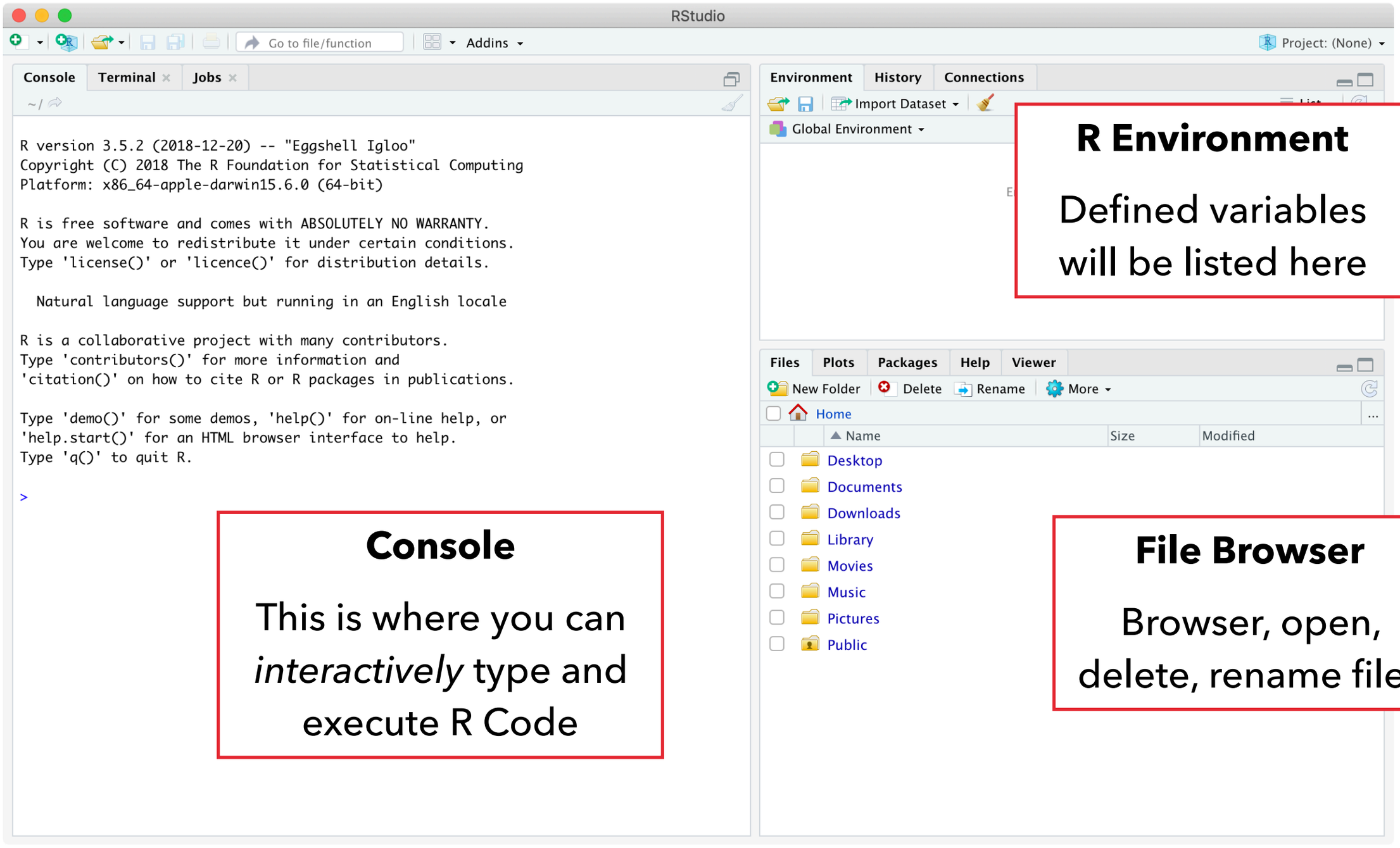
# HIGH-LEVEL: WHAT YOU'LL BE DOING WHEN YOU USE R

- Writing R code to solve a problem (i.e. a data analysis task)
- Two typical ways you'll write R code
  - *Interactively* via the R console
  - *In a script file* that can then be run using the R interpreter
- One of your first hurdles will be learning the R language
  - Syntax: the set of rules and symbols that define the language
  - Getting familiar wit R capabilities, functions, packages

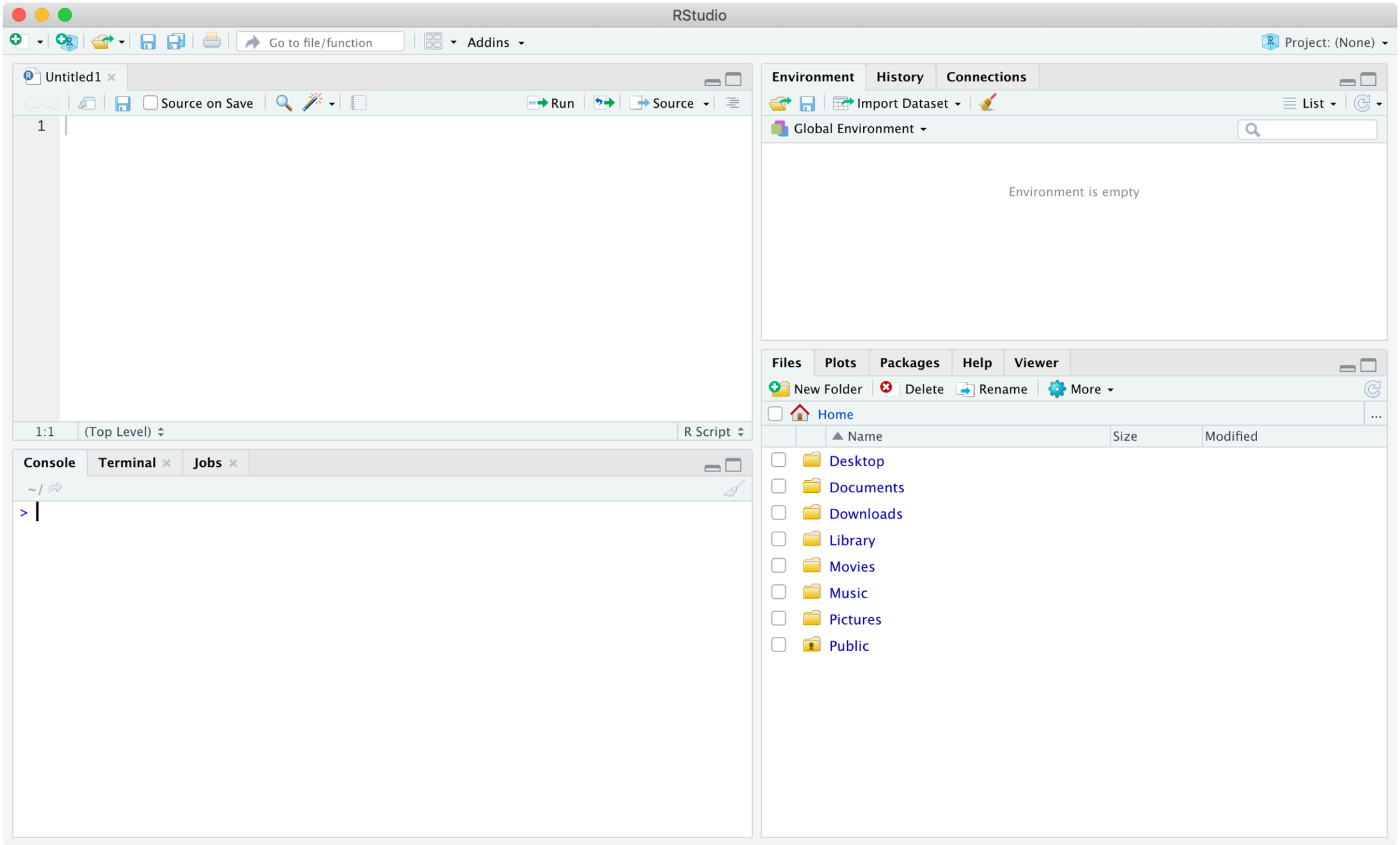
# FIRE-UP RSTUDIO — IT SHOULD LOOK SOMETHING LIKE THIS



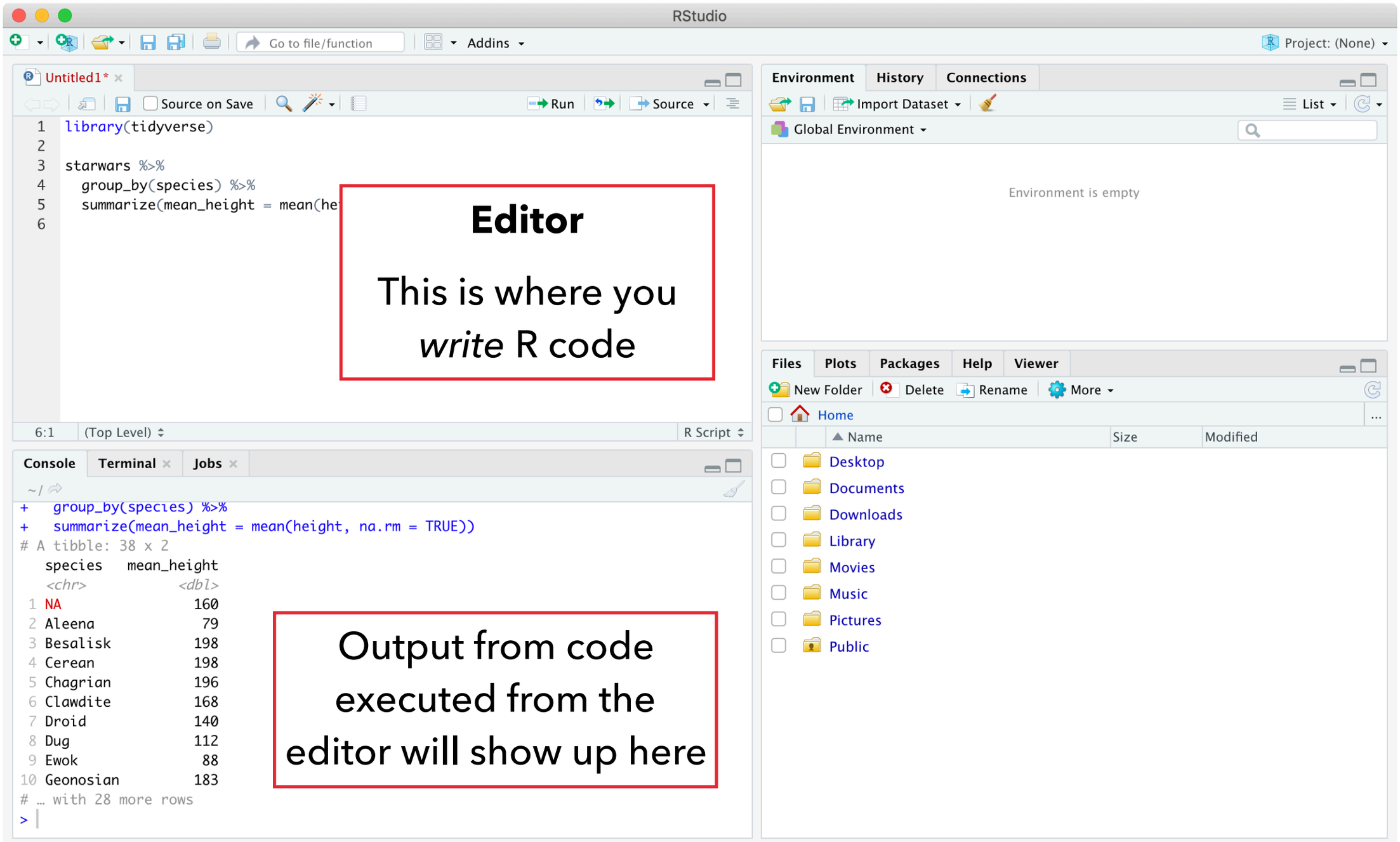
# FIRE-UP STUDIO — IT SHOULD LOOK SOMETHING LIKE THIS



# MAKE AN NEW R SCRIPT: FILE → NEW FILE → R SCRIPT



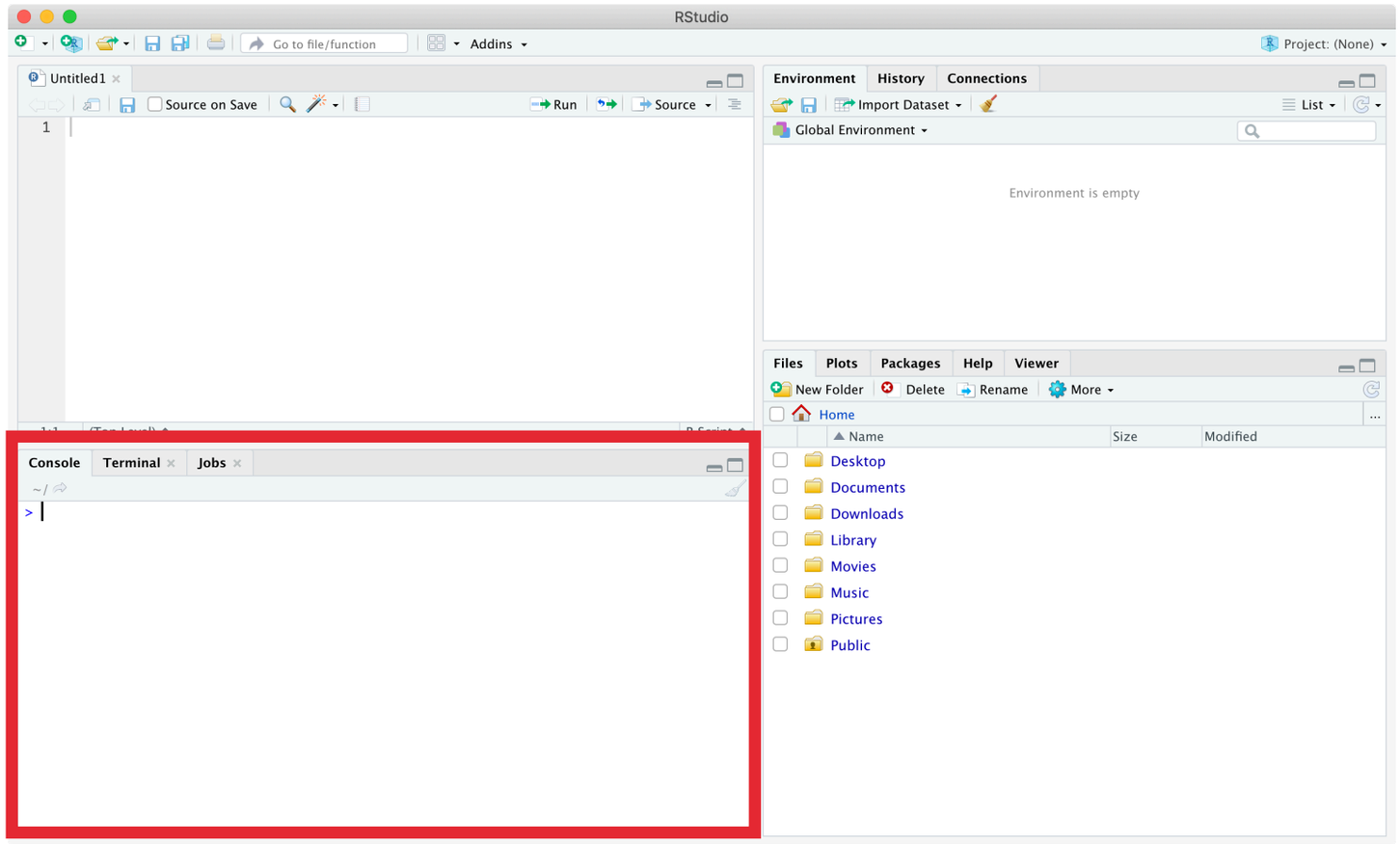
# MAKE AN NEW R SCRIPT: FILE → NEW FILE → R SCRIPT



# THE CONSOLE VS. THE EDITOR

## You'll use the *Console* to:

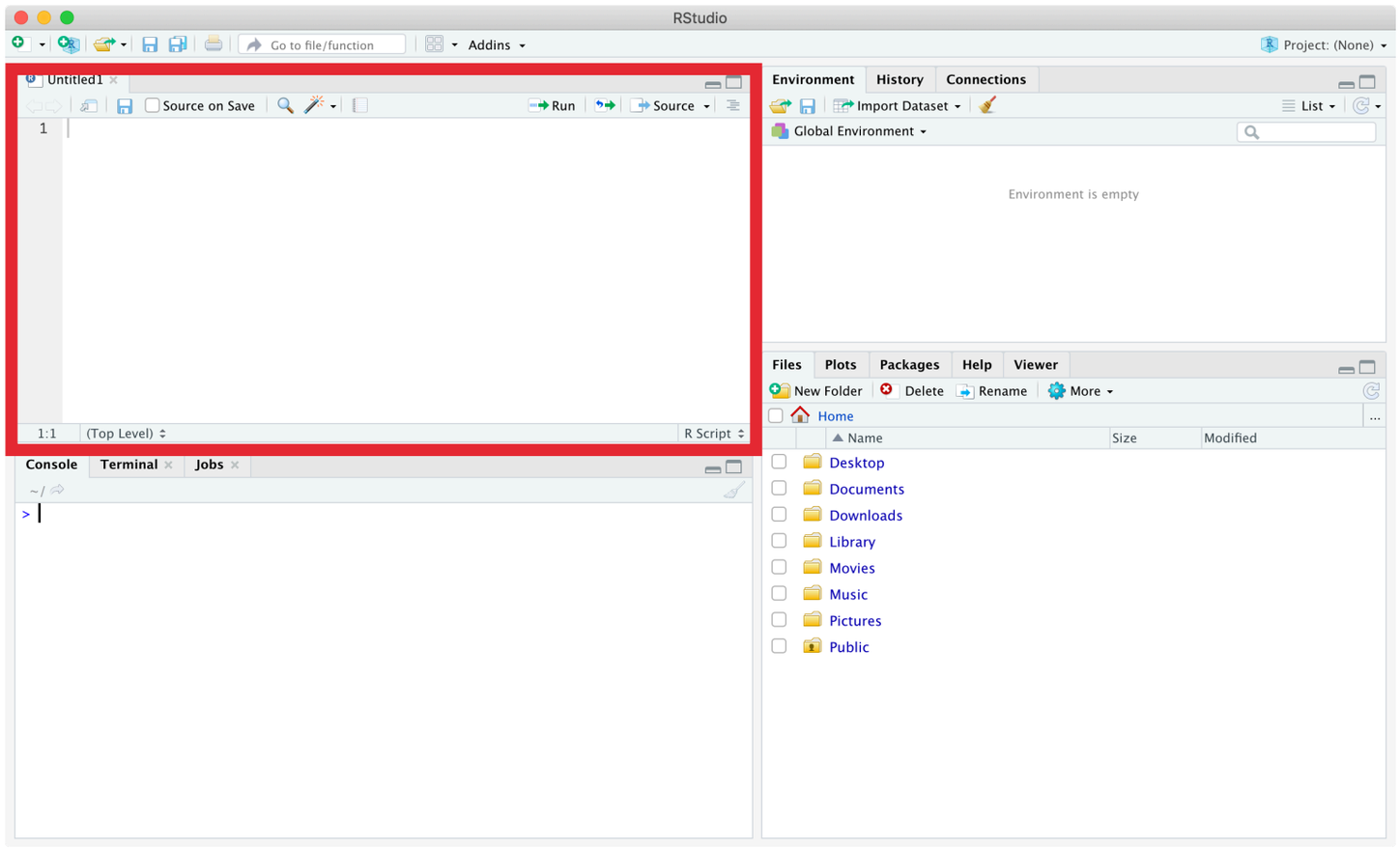
- Interactively work with R
- Try things out & experiment
- Test & check things
- Run code from the *Editor*



# THE CONSOLE VS. THE EDITOR

## You'll use the *Editor* to:

- Write R code
- Keep a persistent record of your analysis (i.e. save you work to a file)
- View a complete analysis script in one place



# THE CONSOLE VS. THE EDITOR

**For a typical analysis workflow**

- Try to work mostly in the *editor* and save your work
- Treat anything your write in the console as ephemeral – code written in the console is not a real record of your work!
- Try to encapsulate your *entire* analysis in a script or set of scripts; avoid one-off analysis steps executed in the console
- Execute code from the editor by pressing `CMD/CTRL-RETURN`, great for testing your code as you write it

# RSTUDIO PROJECTS ARE A GREAT WAY TO ORGANIZE YOUR WORK

- R Projects: organize and encapsulate your analysis projects
- File → New Project… → New Directory → New Project
  name your project & choose where you want to save it
- This will create a new directory with a *.Rproj* file,
  double click this file to get back to your work at any time
- R Projects help manage your R Studio session and your
  working directory

# A SUGGESTED WORKFLOW

**Imagine you need to start a new analysis project**

1. Create a new RStudio Project
2. Assemble the required input data & info you need, put them inside the project directory*
3. Write analysis scripts that do the work; consider naming them with a numeric prefix (e.g. 01_..., 02_...) indicating the order they should be executed in
4. Write, test, write test, write, test...
5. Double click the .Rproj file when you want to get back to work

* can depend on how you store data (e.g. networked storage)

# Demo Time

1. Create an RStudio Project

2. Review of R basics example code [*00_quick_r_intro.R*]