R Programming For Natural Resource Professionals

Lecture 1: Introduction to R, basic math, objects

Jared Homola
TNR 163
jhomola@uwsp.edu

Instructor intro

WISCONSIN
COOPERATIVE FISHERY RESEARCH UNIT

USGS
science for a changing world

University of Wisconsin
Stevens Point

WISCONSIN
DEPT. OF NATURAL RESOURCES

WISCONSIN
COOPERATIVE FISHERY RESEARCH UNIT

Dan Isermann
Unit Leader

Aquatic Biomonitoring Lab

Fisheries Analysis Center

Jared Homola
Assistant Unit Leader

Molecular Conservation Genetics Lab

Intros

- Name
- Discipline
- Data you'll be working with
  - Animal counts? Economic? Surveys? Telemetry? Not sure yet?
- Experience with R
  - Think I know what R is…
  - Load and manipulate data
  - Run and interpret wide range of analyses
  - Full-fledged guru (write functions, packages, etc.)

Course objectives

Develop skills in the R programming language

- Read in and process data
- Write functions, loops, conditionals, etc.
- Create wide variety of data visualizations
- Code and interpret common statistical analyses

Develop general data science skills

- Ethical data management practices
- Research reproducibility
- Master efficient workflows

Course organization

- Mon 9:30am-10:45am lecture
- Wed 9:30am-10:45am lab
- Homework assigned each Wednesday, due the next Wednesday
- Two larger assignments
- Grades: 100% homework
- Office hours: By appt, email at jhomola@uwsp.edu
- Course website: jaredhomola.github.io/RforNatRes/

Class discussions

- Mondays will start with a discussion of assigned readings
- I'll establish discuss groups when you get here each week
- 5-7 min to discuss and formulate three things:
  - Thought
  - Question
  - Epiphany
- Submit each via Google Docs which will serve as starting point for discussion

Class norms

- Be respectful & understanding of wide diversity of experiences.
- Ask questions during class. Share your learning.
- Avoid distracting others with what’s on your screen.
- You’re in a professional program. You’ll be treated like a professional.
- Feedback on the course is always welcome.

Sharing and code reusing policy

- Team learning is key! Help each other.
- Turn in assignments independently, but list people you worked with
  - Do not turn in identical assignments
- Cite code that you found online by providing the url after relevant answers
  - Nothing wrong with poaching code. Just credit it.
- Reused code that is not cited will be treated as plagiarism
  - Will be spot-checked

Tips for course success

- Get out what you put in.
- Getting the right answer is a minor part of what you'll be graded on.
  - Emphasis on creativity and problem solving
- There are countless ways to get to the right answer
  - I'll teach you a couple ways, you'll probably find others
  - Strive to write efficient code

A word on TidyR

- Beginning in week 3, we will transition to writing code using the "TidyR" approach
- Learning some essential base R code in the first couple weeks will serve you well going forward

Questions?

What is R?

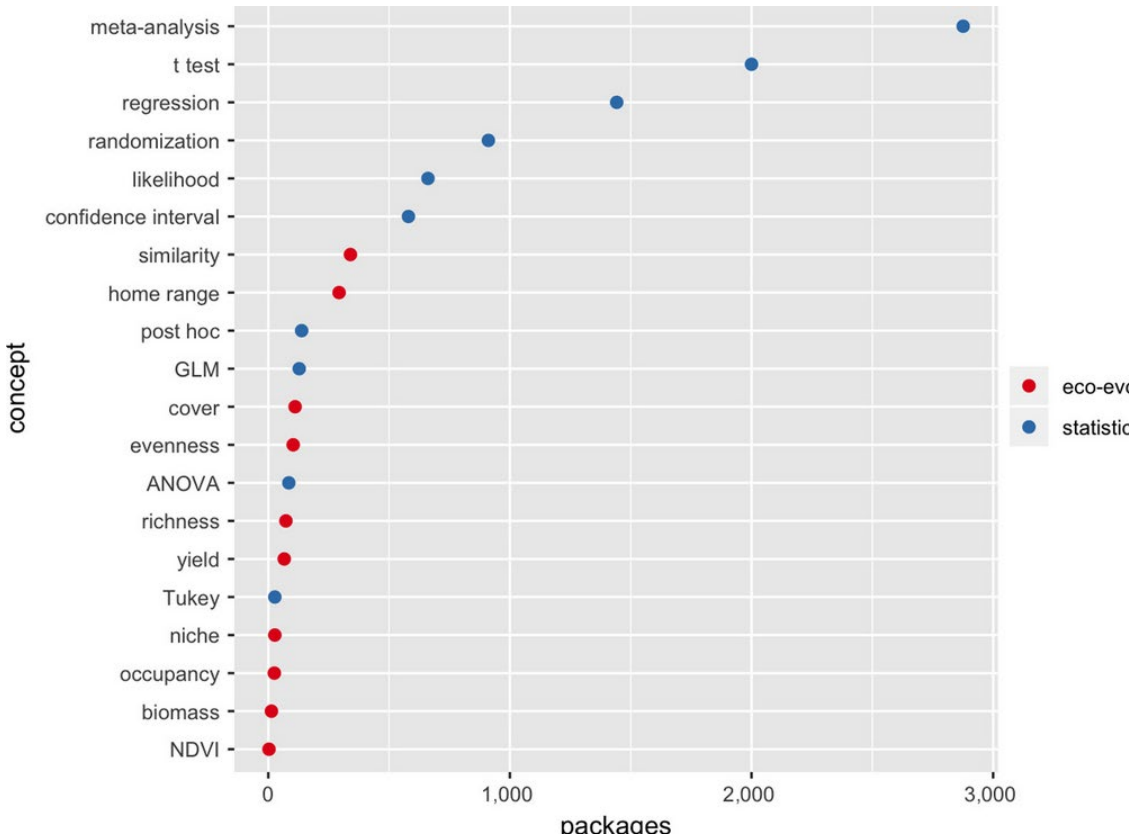
- R is…
  - computer language
  - an environment for statistical computing
  - platform for generating graphics
  - modeling platform
  - Much, much more!
- Script-based (text computer code) and not GUI-based (point and click with menus)

Why learn R?

- It can get you a job
- Free and open-source
- Powerful plotting tool for generating publication-quality graphics
- Huge community of scientists and developers using R

Why learn R?

- Incredibly interdisciplinary
- Currently 18,762 R packages on CRAN!
  - Population dynamics
  - Fish stock assessment
  - Telemetry
  - Camera trap analyses
  - Econometrics
  - Baseball analytics
  - Thousands of other things…



Why learn R?

- Research **reproducibility** and **transparency**: Anyone using any operating system can reproduce your work
  - Read in data
  - Wrangle your data into the right format
  - Exploration of patterns in complex data
  - Apply statistical tests and fit models
  - Produce summary statistics and tables
  - Create final figures
- Easily make changes if your data change, model must be revised, or reviewers ask for revisions, etc.

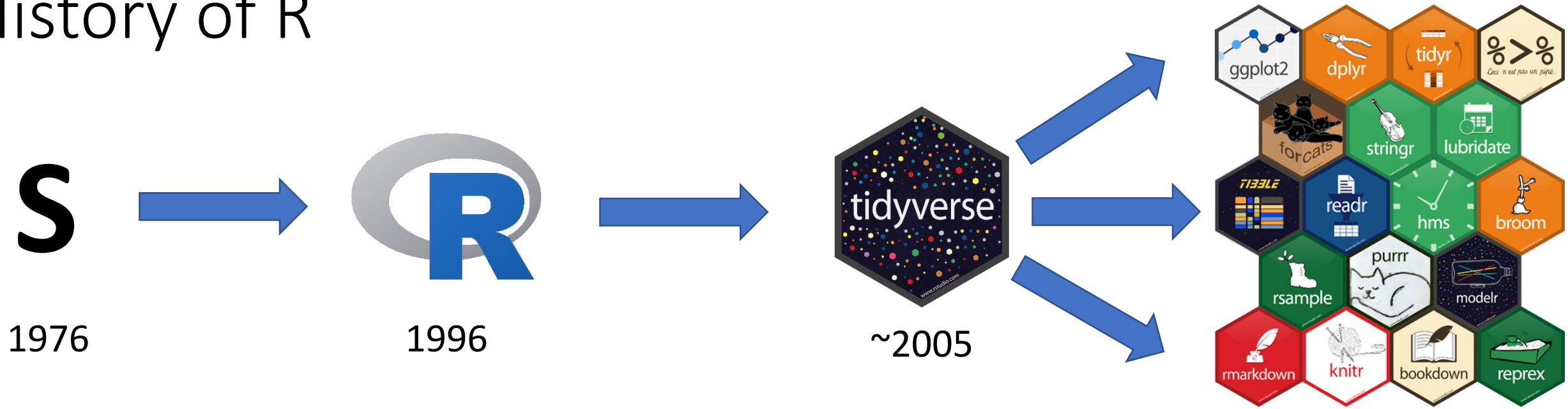
What about Excel?

- Excel allows quick prototyping
- Data manipulation is easy
- Can see what is happening
- Looping is hard
- Limited statistical packages
- Inflexible
- Hard to repeat analyses

How to learn R

- R is a language, the learning curve can be steep
- Be patient and creative
- Keep your motivation in mind
- Lots of help files, online sources, books
- Work with other students
- Reach out to instructor

History of R



- S: language for data analysis developed at Bell Labs circa 1976
- R: initially written & released as an open-source software by Ross Ihaka and Robert Gentleman at U Auckland during 90s
- Since 1997: international R-core team ~15 people & 1000s of code writers and statisticians happy to share their libraries

R lingo

Code	<	Function	<	Package	<	Library
Input into R		Packaged code that receives inputs and generates outputs		A set of functions		All your packages

```
2+2
```

```
rnorm(n=100, mean=50)
```

Function name

Arguments

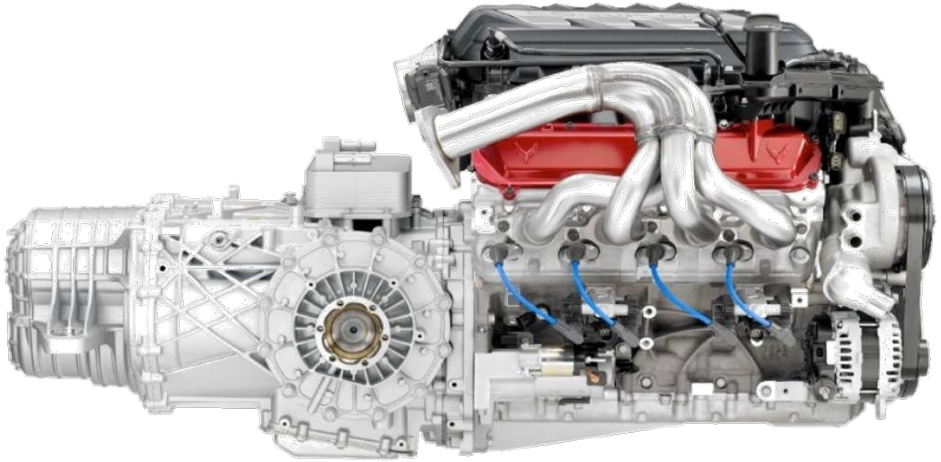
```
library(stats)
```

Library function

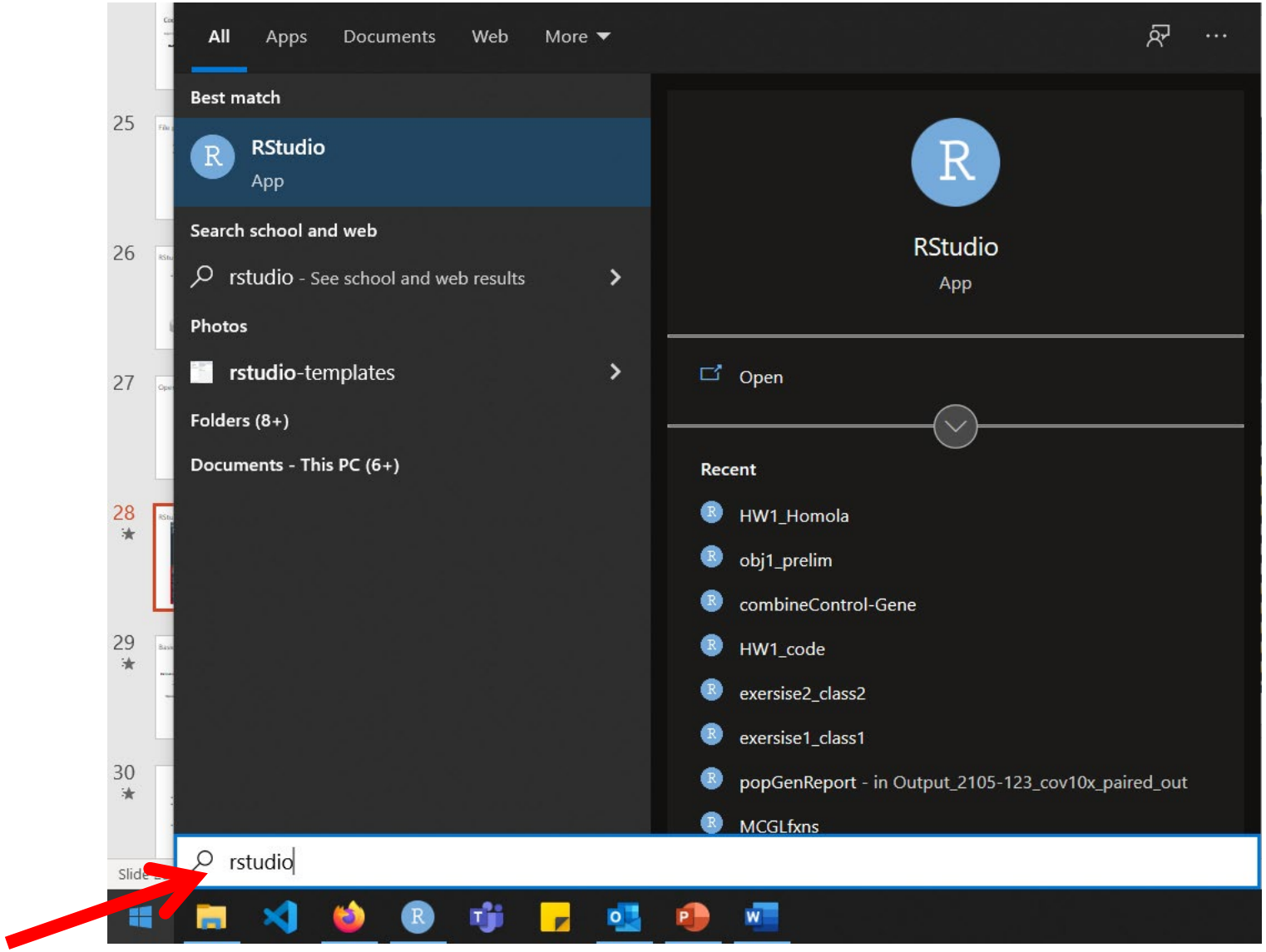
Package name

RStudio

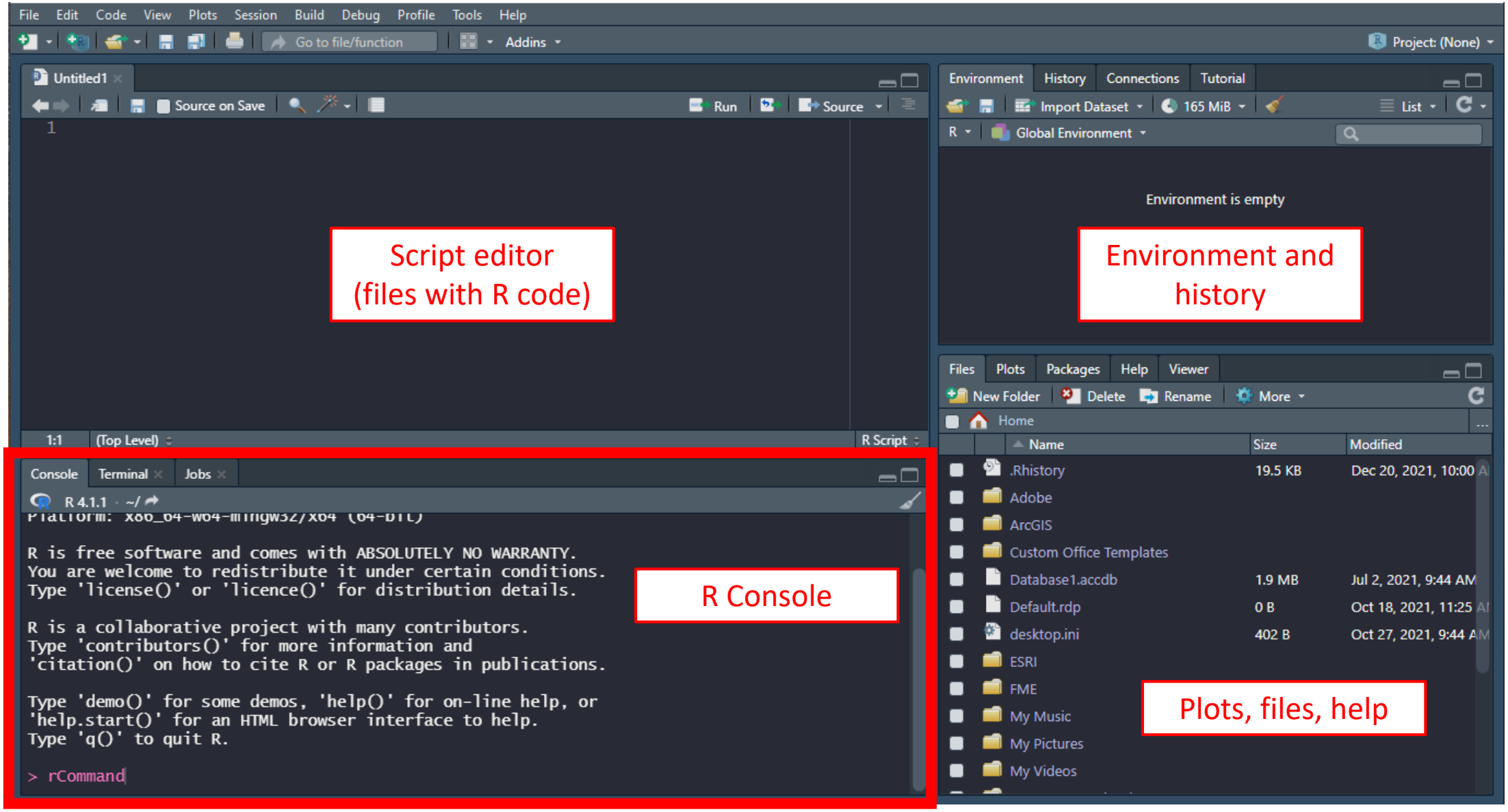
- RStudio is an integrated development environment (IDE)



Open R studio



RStudio


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1
Source on Save
Run
Source
Script editor
(files with R code)
1:1
(Top Level)
R Script
Console
Terminal
Jobs
R 4.1.1 · ~/
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.
> rCommand
R Console
Environment
History
Connections
Tutorial
Import Dataset
165 MiB
List
R
Global Environment
Environment is empty
Environment and
history
Files
Plots
Packages
Help
Viewer
New Folder
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
19.5 KB
Dec 20, 2021, 10:00
Adobe
ArcGIS
Custom Office Templates
Database1.accdb
1.9 MB
Jul 2, 2021, 9:44 AM
Default.rdp
0 B
Oct 18, 2021, 11:25
desktop.ini
402 B
Oct 27, 2021, 9:44 AM
ESRI
FME
My Music
My Pictures
My Videos
Plots, files, help

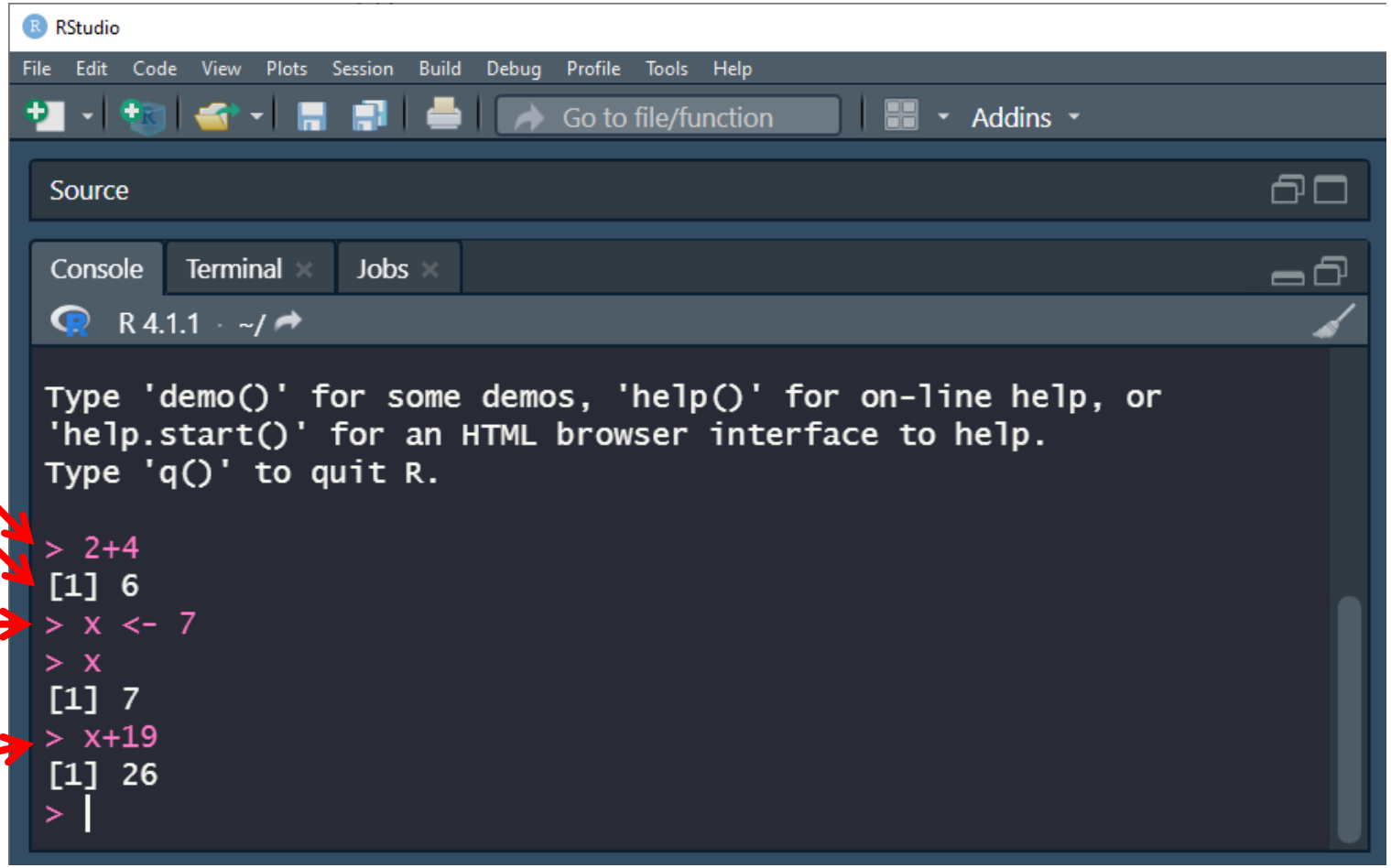
Basic syntax

R command prompt

[1] **indicates the first element of a vector**

"<-" assign a value to a variable

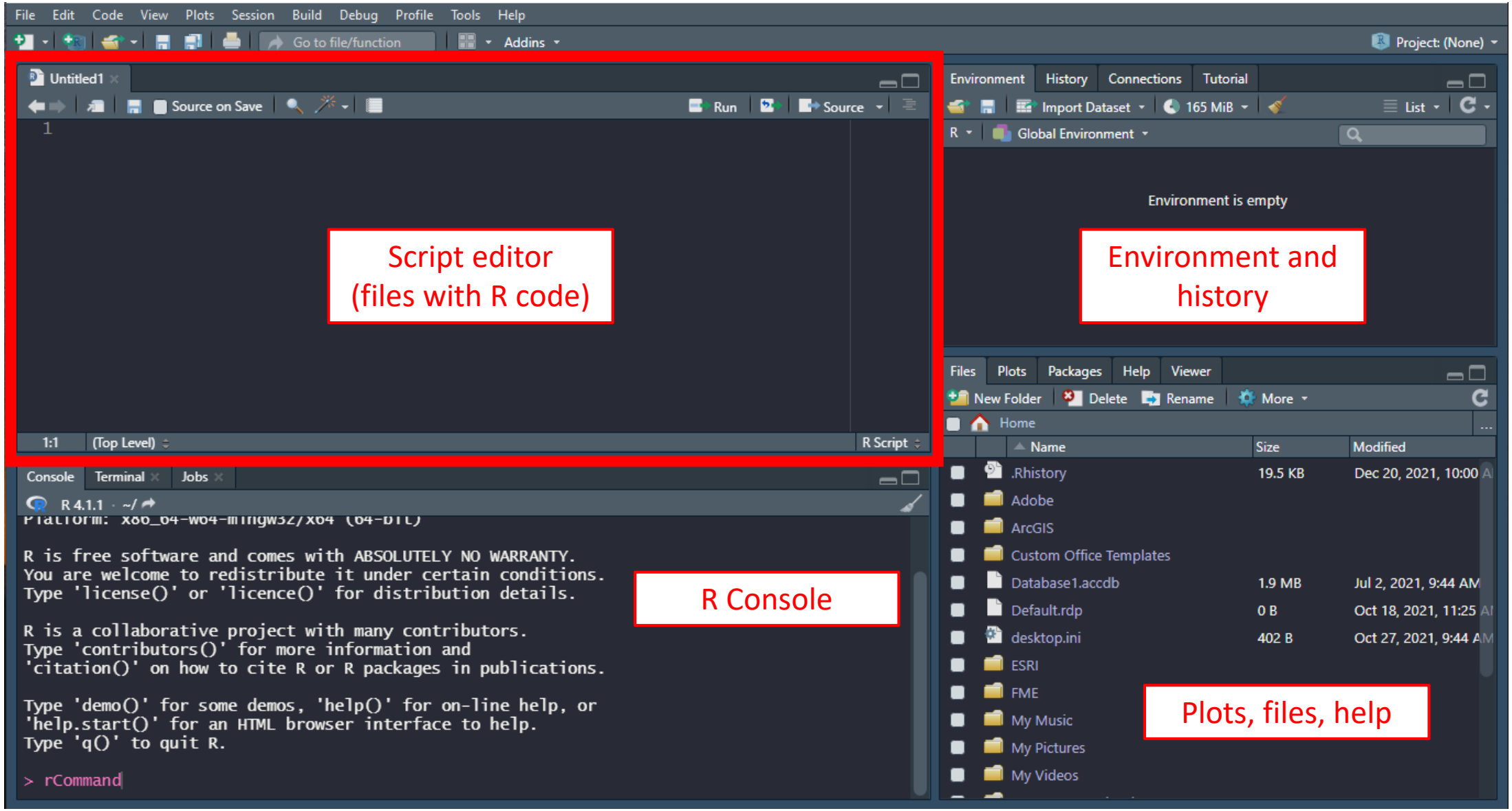
Operation of adding to variable's value



Note: R is case sensitive!

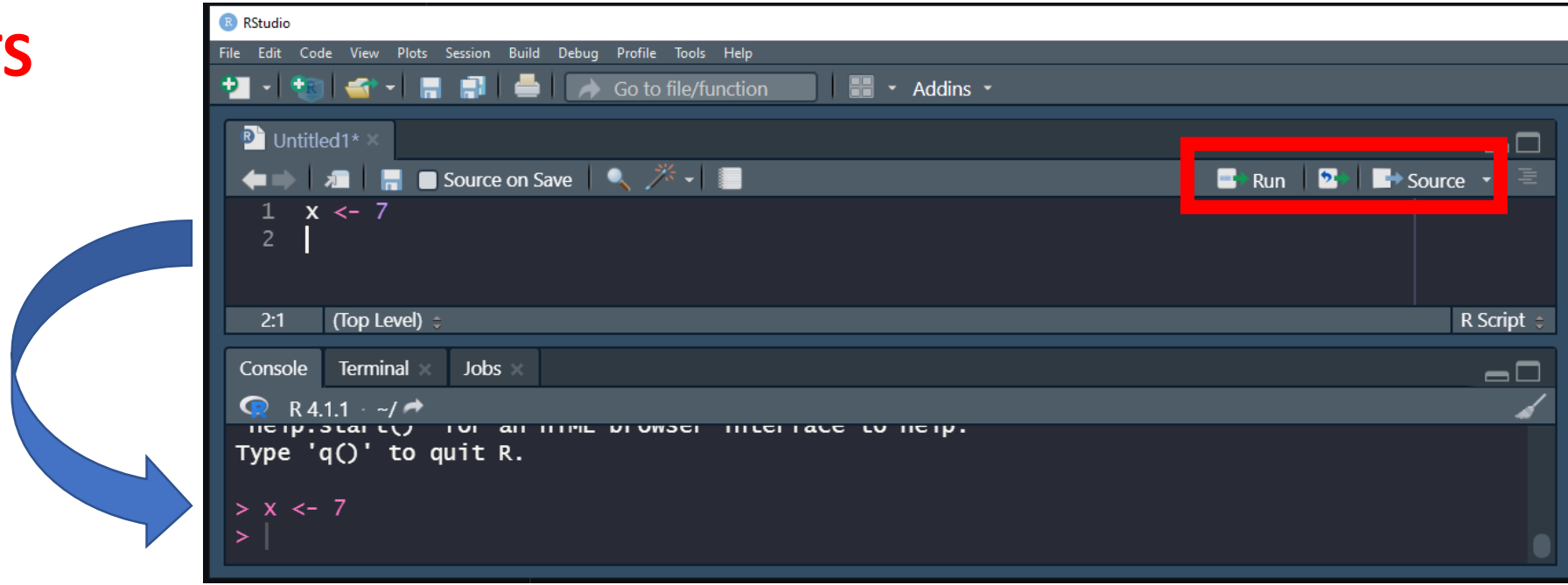
```
> x <- 10
> x
[1] 10
> X
Error: object 'X' not found
>
```

RStudio


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1
Source on Save
Run
Source
1
Script editor
(files with R code)
1:1
(Top Level)
R Script
Environment History Connections Tutorial
Import Dataset
165 MiB
List
R
Global Environment
Environment is empty
Environment and
history
Files Plots Packages Help Viewer
New Folder
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
19.5 KB
Dec 20, 2021, 10:00
Adobe
ArcGIS
Custom Office Templates
Database1.accdb
1.9 MB
Jul 2, 2021, 9:44 AM
Default.rdp
0 B
Oct 18, 2021, 11:25
desktop.ini
402 B
Oct 27, 2021, 9:44
ESRI
FME
My Music
My Pictures
My Videos
Plots, files, help
Console Terminal Jobs
R 4.1.1 · ~/
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.
> rCommand
R Console

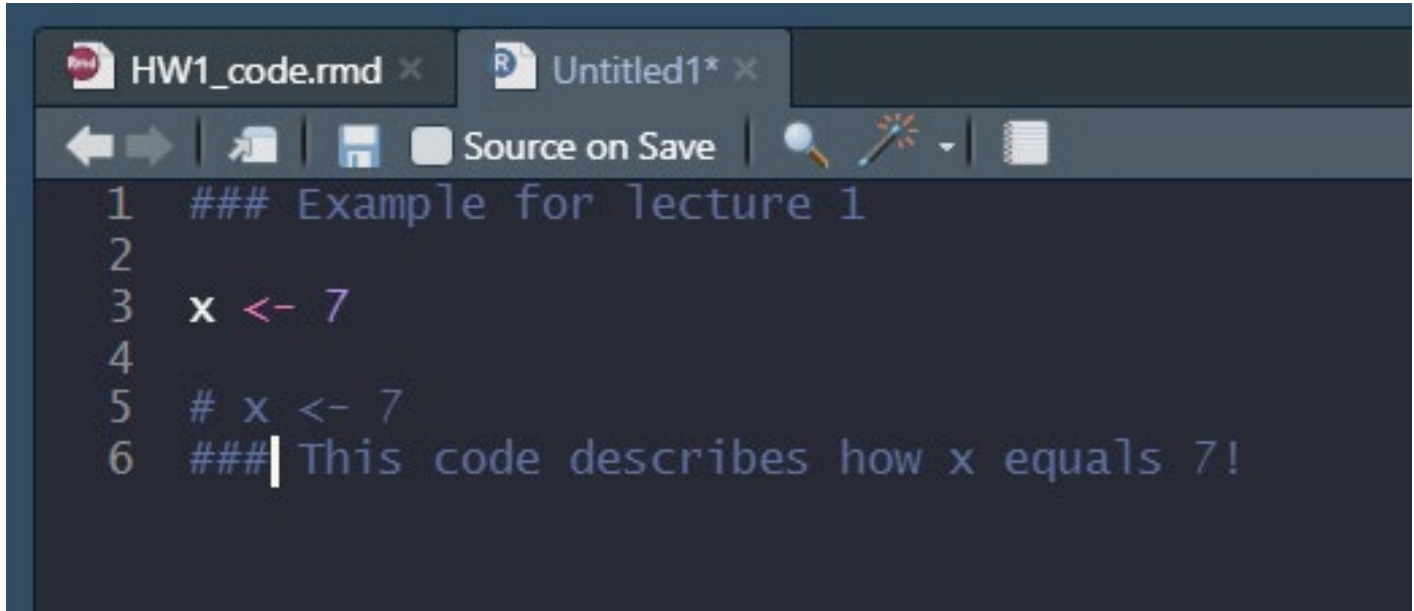
R scripts

- A text file that contains your R code
- Reproducible: rerunning your code is easy for you or someone else
- Easily modified and rerun
- In RStudio, type <ctrl+enter> to run the code in the R console
- **SAVE YOUR SCRIPTS**

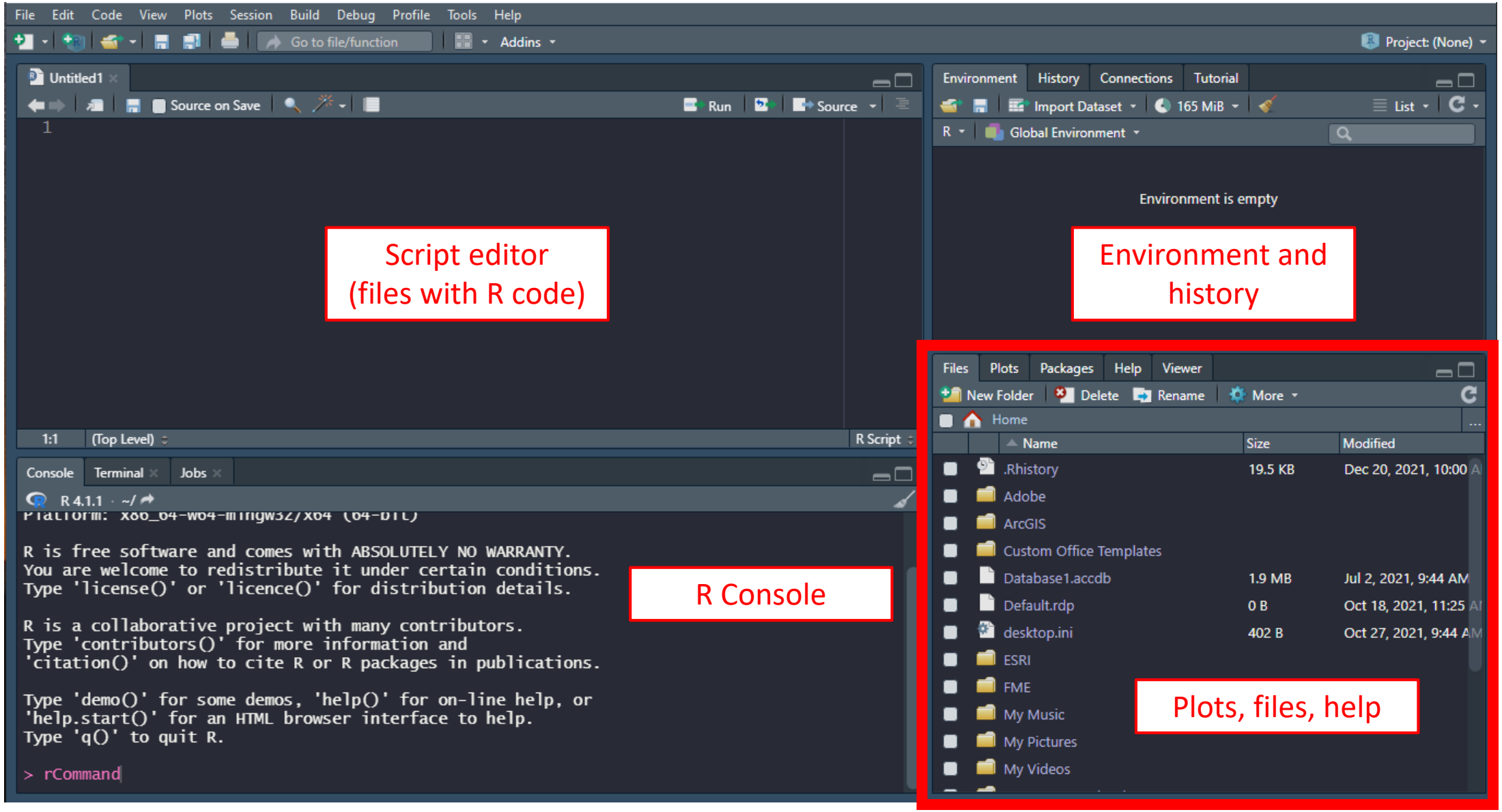


R scripts

- Comment your code using #
- Useful for temporally removing code without deleting it
- Make notes to future users of the script (especially yourself!)
- Useful for adding headers to the top of scripts



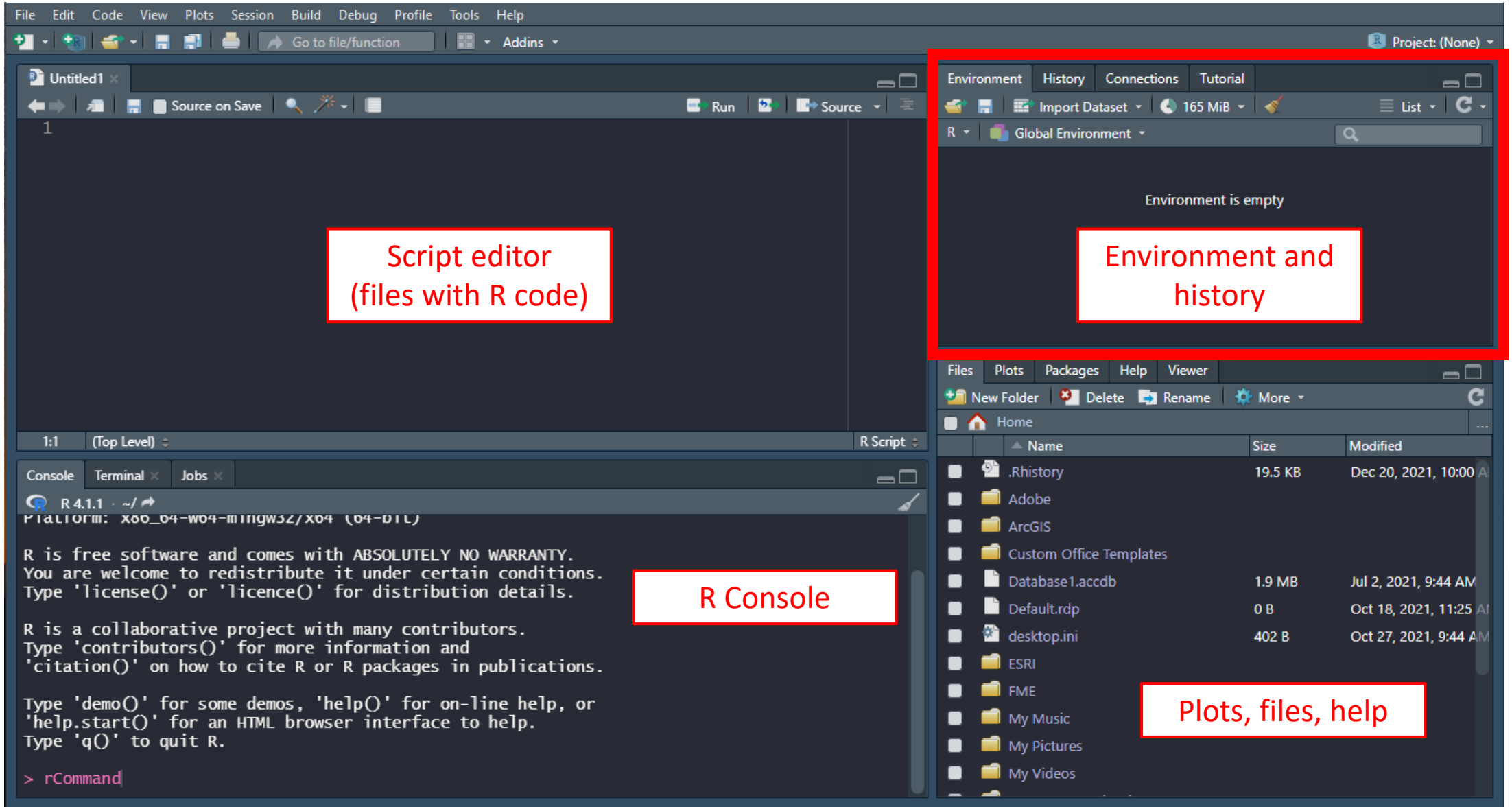
RStudio


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1
Source on Save
Run
Source
Script editor
(files with R code)
1:1
(Top Level)
R Script
Environment History Connections Tutorial
Import Dataset
165 MiB
List
R
Global Environment
Environment is empty
Environment and
history
Console Terminal Jobs
R 4.1.1 · ~/
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.
> rCommand
R Console
Files Plots Packages Help Viewer
New Folder
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
19.5 KB
Dec 20, 2021, 10:00
Adobe
ArcGIS
Custom Office Templates
Database1.accdb
1.9 MB
Jul 2, 2021, 9:44 AM
Default.rdp
0 B
Oct 18, 2021, 11:25
desktop.ini
402 B
Oct 27, 2021, 9:44 AM
ESRI
FME
My Music
My Pictures
My Videos
Plots, files, help

R’s built-in help

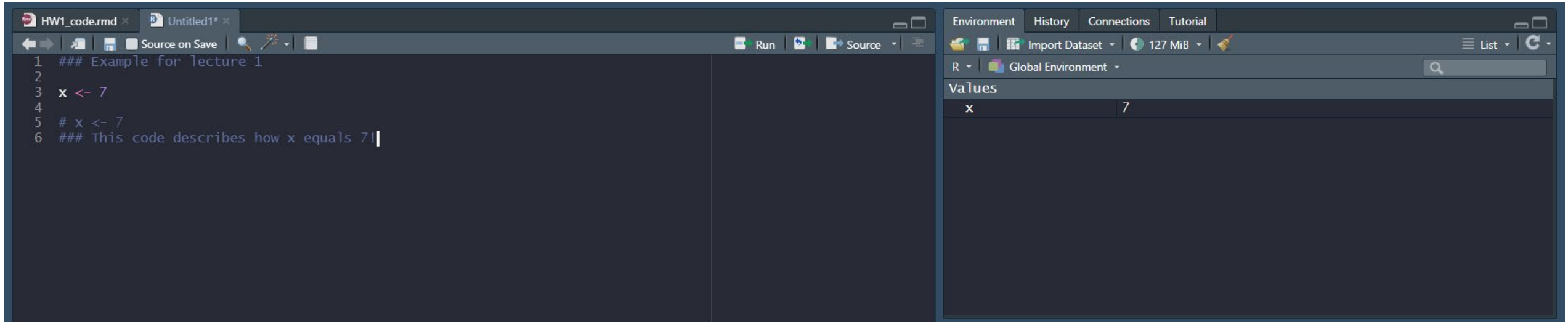
- `?mean`
  - I know this is the right function, but I can’t remember how to use it
- `??mean`
  - I think this is the right function, but I could be wrong…

RStudio


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1
Source on Save
Run
Source
Script editor
(files with R code)
1:1
(Top Level)
R Script
Environment History Connections Tutorial
Import Dataset
165 MiB
List
R
Global Environment
Environment is empty
Environment and
history
Console Terminal Jobs
R 4.1.1 · ~/
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.
> rCommand
R Console
Files Plots Packages Help Viewer
New Folder
Delete
Rename
More
Home
Name
Size
Modified
.Rhistory
19.5 KB
Dec 20, 2021, 10:00
Adobe
ArcGIS
Custom Office Templates
Database1.accdb
1.9 MB
Jul 2, 2021, 9:44 AM
Default.rdp
0 B
Oct 18, 2021, 11:25
desktop.ini
402 B
Oct 27, 2021, 9:44 AM
ESRI
FME
My Music
My Pictures
My Videos
Plots, files, help

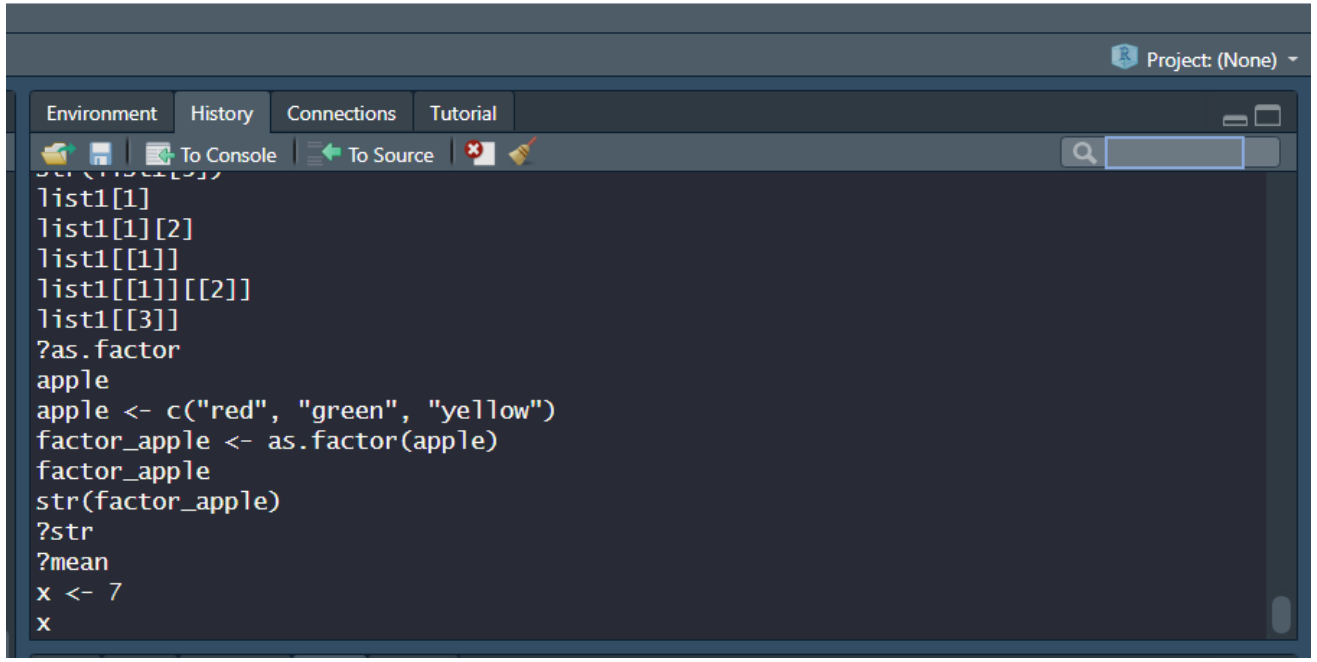
Environment

- Objects currently stored in the computer's memory
- Clean up using the broom icon



History

- List of commands that have been recently run
- Double click to re-run something
- Also accessible using up or down arrows in the R console



Basic R coding

First, experiment with R as a calculator:

```
> 2+2*4
> 2^5+7
> 2^(5+7)
> 0.05/1E6 #note 1E6 = 1,000,000
```

Basic R coding

Experiment with variables. For instance…

```
> name1 <- “Jared” # Note: variables cannot start with number
> name2 <- “Homola”
> c(name1, name2) # c = concatenate (combine)
> paste(name1, name2)
> paste0(name1, name2)
> x = 1
> y = 2
> x+y
```

Basic R coding

Experiment with functions. For instance…

```
> sqrt(16)
> round(3.1459)
> ?round
```

How to round to 2 decimal points?

What if you don’t state the name of the argument?

Setting working directory

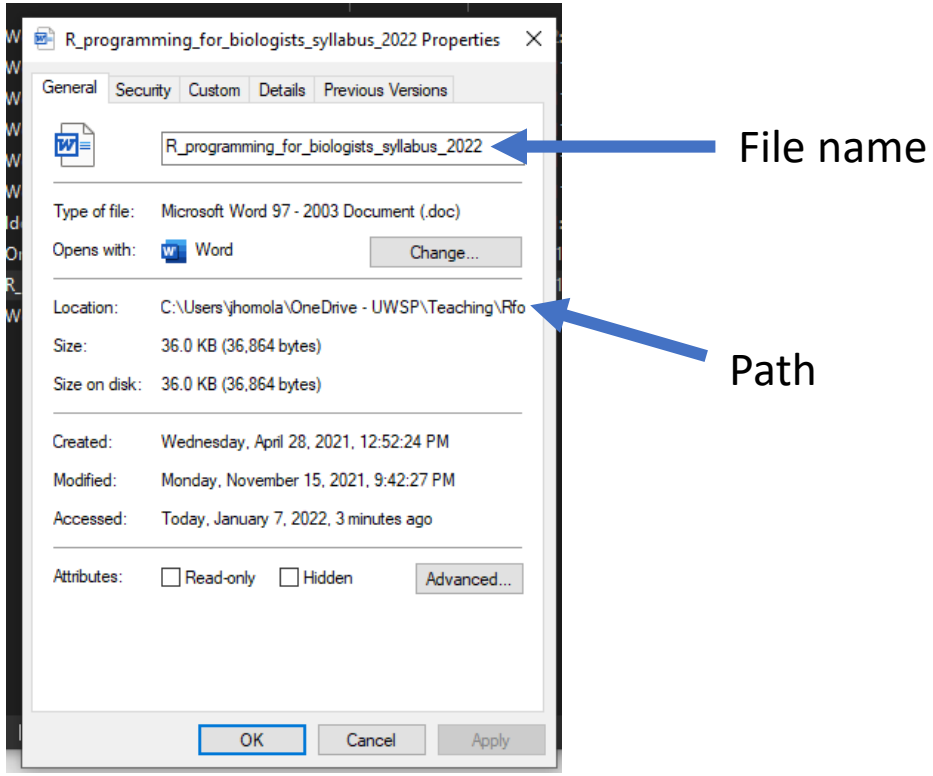
Directory: A location on your computer's hard drive or in a cloud drive

- E.g., C:\Users\jhomola\OneDrive - UWSP\Teaching\RforNatRes_2021

The working directory becomes the default place that R looks for files.

This will not always be encouraged!

```
> setwd("working/directory/")
```



Data frames

Data frame: two-dimensional array-like structure (e.g., table) with variables arranged in columns and values in each row.

All variables must have the same number of values

Make your own:

```
> var1 <- 1:4
> var2 <- 5:8
> data.frame(var1, var2)
```

Basic subsetting

Subsetting data: isolating certain values and/or variables

Subsetting a column:

```
> dat <- data.frame(var1, var2)
> dat$var1
```

Subsetting a column, row, or single value

```
> dat[1,] #First slot is for complete rows
> dat[,1] #Second slot is for complete columns
> dat[1,2] #Combining them gives a single value
> dat[1] #No comma gives values in column as vector
> dat[1:3,] #Colons indicate ranges
```

Basic summary statistics

```
> sum(dat$var1)
> min(dat$var2)
> max(dat$var1)
> sd(dat$var2) #Standard deviation
```