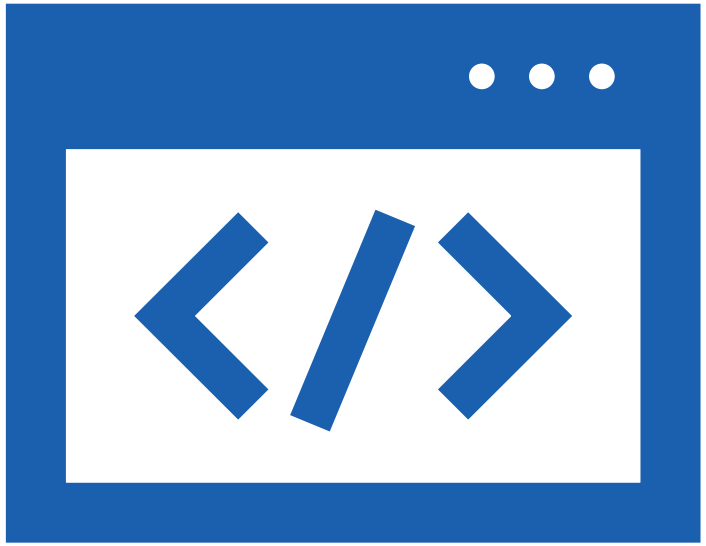
The R Essentials

By the end of this workshop, you'll know how to …

1. Create R markdown (.Rmd) / Script (.R)
2. Load packages
3. Use ?help to find functions in base R & packages
4. Create an R object
5. Learn different variable types
6. Utilize data viewing functions



What R they?

- R is a programming language used for data visualization and statistical programming
- R Studio is an integrated development environment (IDE) used with R to provide a user-friendly interface for code development

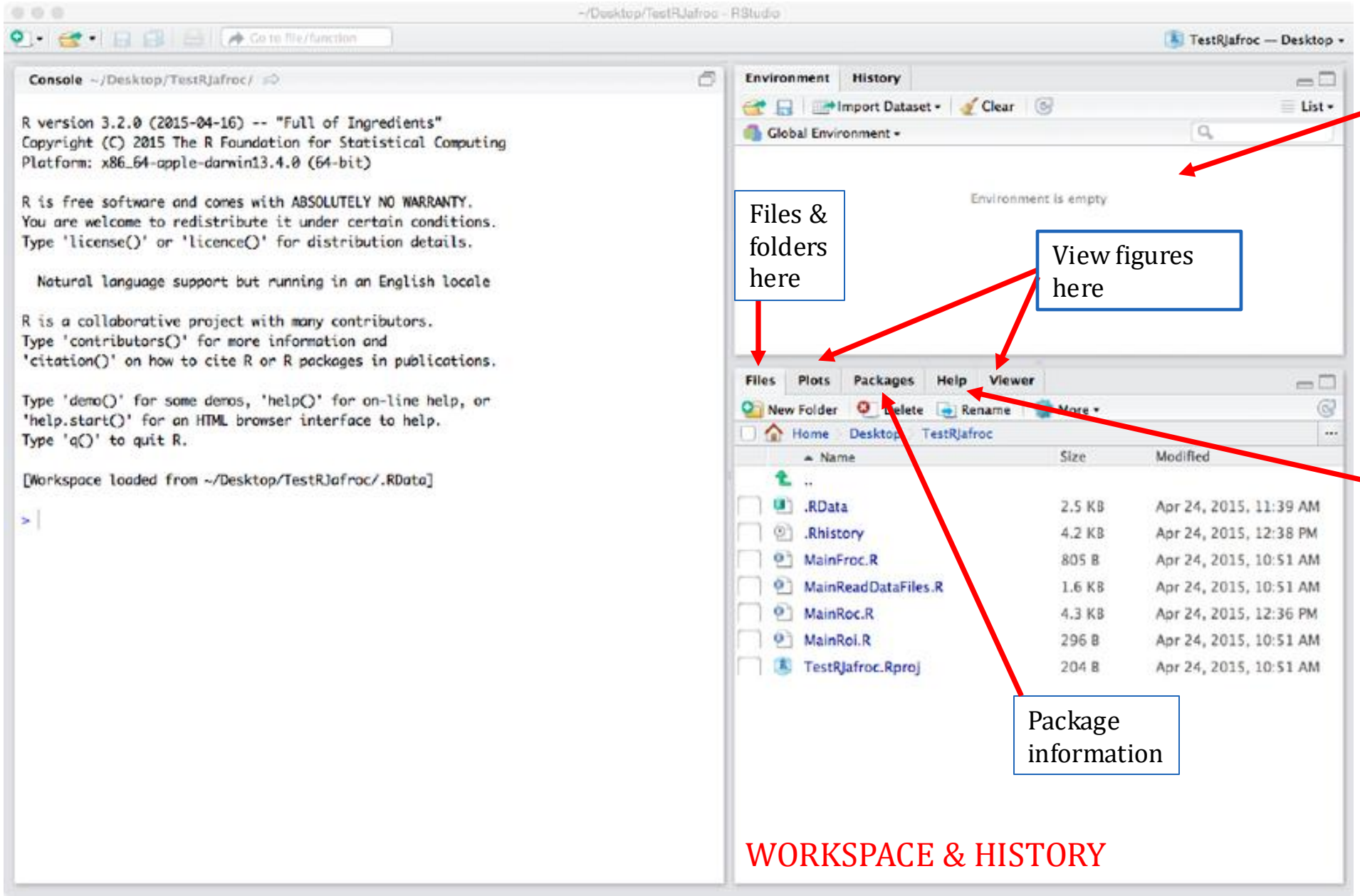
How to access R & RStudio?

https://posit.co/download/rstudio-desktop/

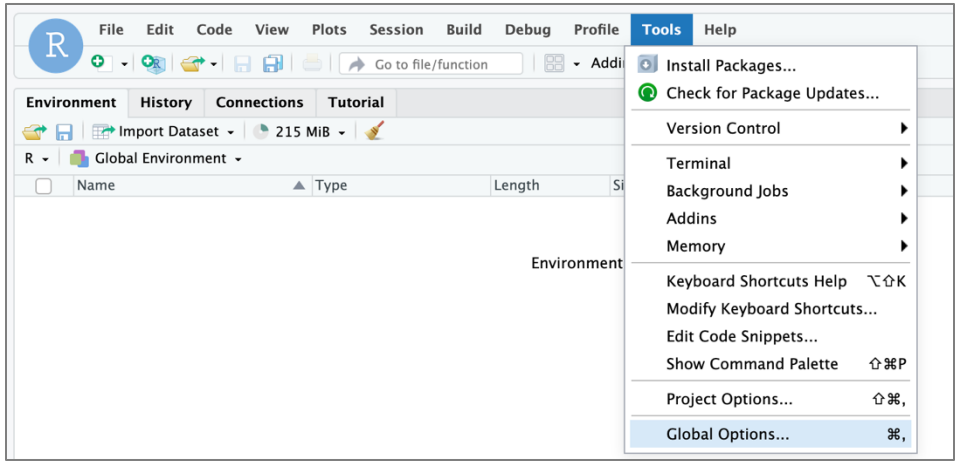
Using OnDemand local

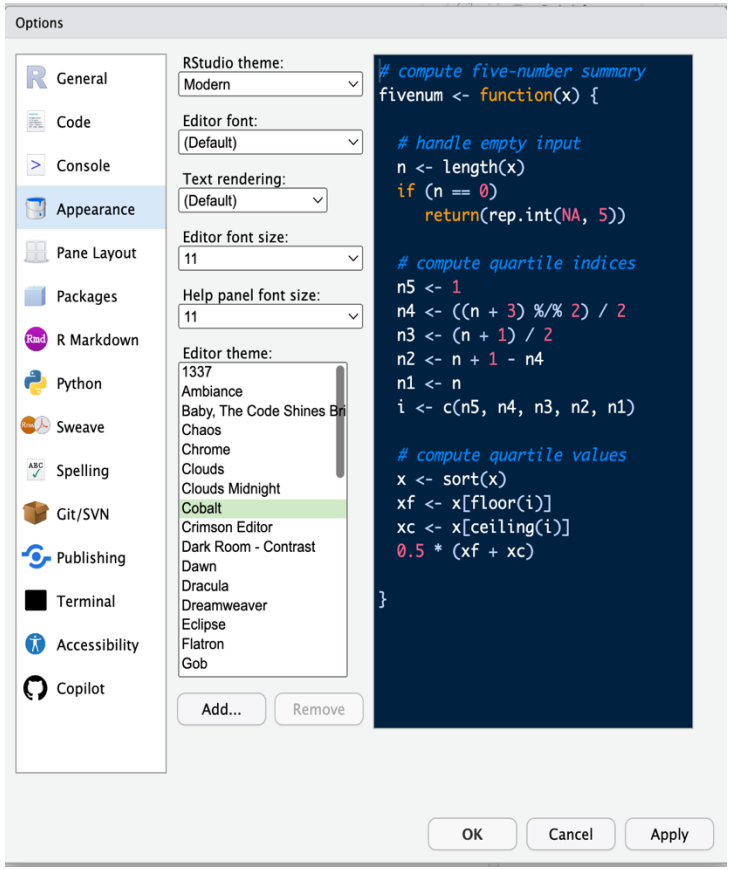
- If your home directory is set up on the HPC and you plan to use OnDemand, use this link:
- https://ondemandlocal.hpc.vai.org/

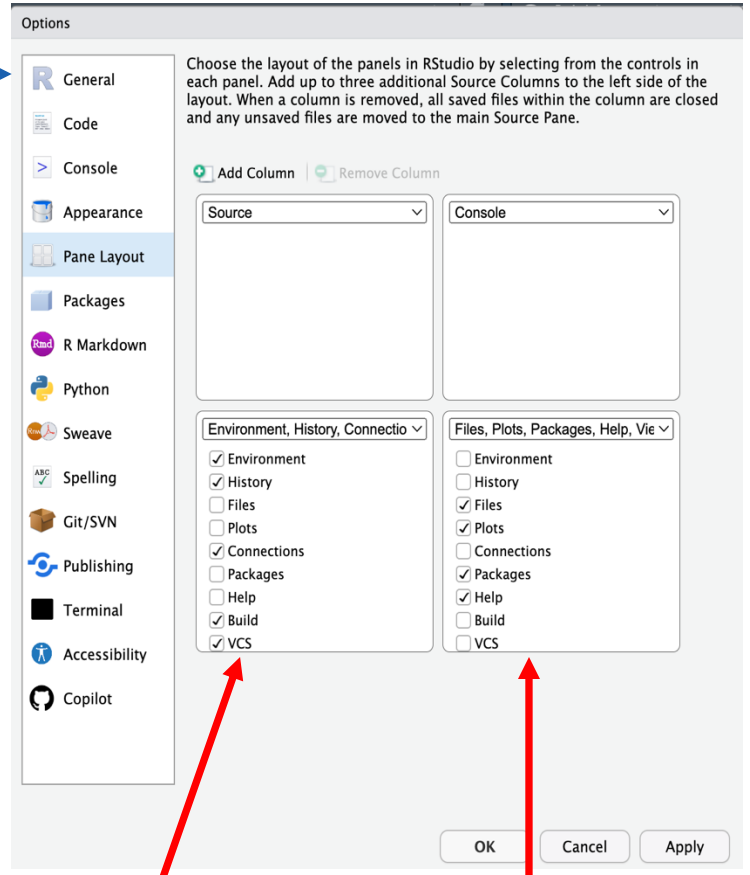
R Studio start up



Now that we have that figured out, let's change it…



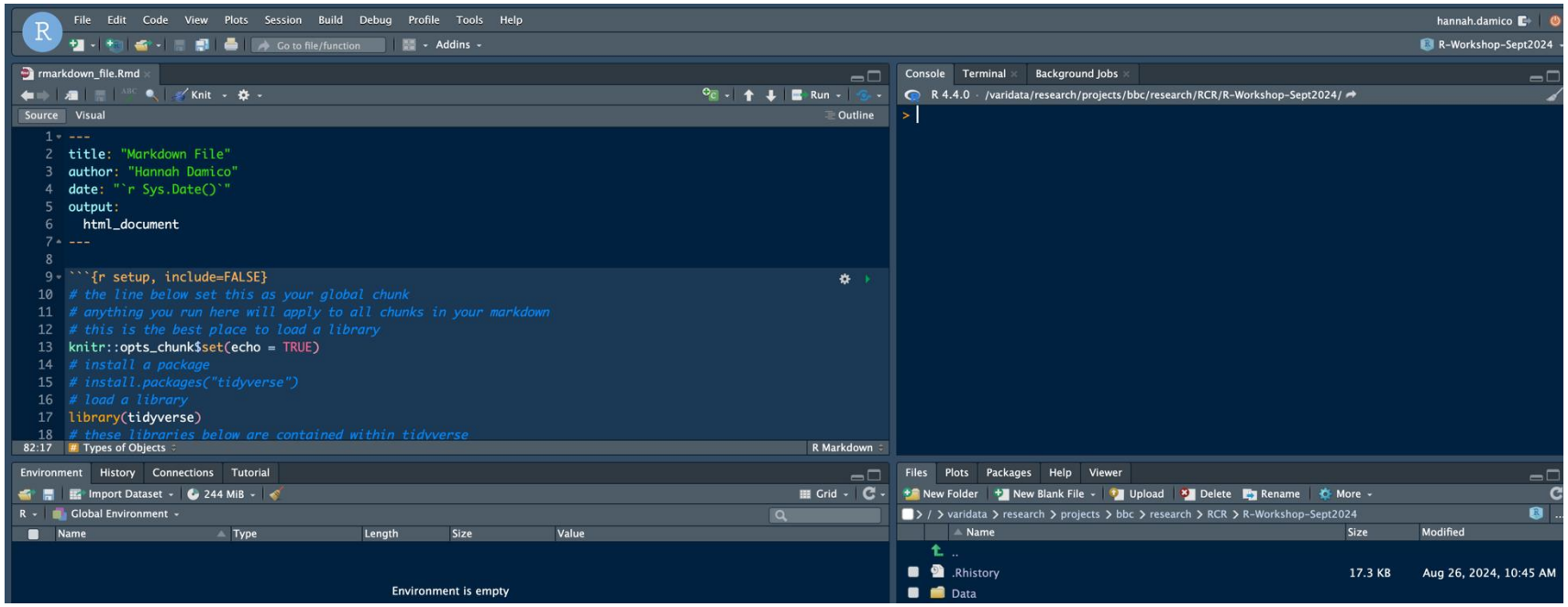




1. Tools
2. Global Options
3. Appearance (change as you'd like, but we'll be working in dark mode today)
4. Hit apply
5. Panel Layout
    1. Source (top left)
    2. Console (top right)
    3. Environment (bottom left)
    4. Workspace + history (bottom right)

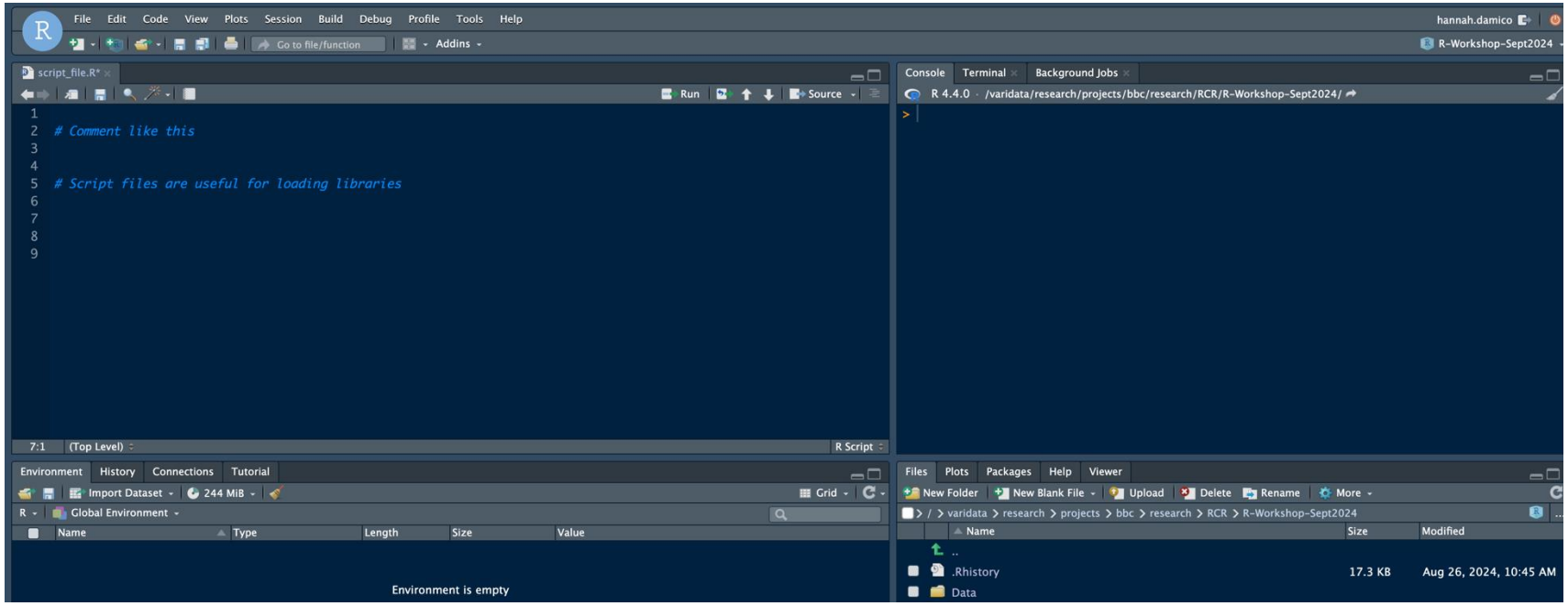
✔ Environment, History, connections, build, VCS, tutorial

✔ Files, plots, packages, help, Viewer



R Markdown

- Main focus today
- Create code “chunks”
- Write text paragraphs in between code



R script

- Not covered in this workshop
- More suitable for long lines of code

Base R functions

- Do not require anything other than R itself
- Some covered today:
  - `data.frame()` – data formatted as a table
  - `c()` – combines elements
  - `matrix()` – arrange data in rows and columns
  - `factor()` – turn a variable into a factor
  - `$` - access objects stored someplace

- Other useful functions:
  - `min()` / `max()` – find the minimum or maximum value of vector or dataframe
  - `sum()` – take the sum of elements in a vector
  - `plot()` – general plot function
  - `hist()` – general histogram function
  - `paste()` – combine multiple elements into one
  - `grep()` – search for patterns
  - `sub()`/`gsub()` – replace patterns in strings

https://bookdown.org/amy_yarnell/T32-book/selected-glossary-of-r-terminology.html

Packages

- Toolboxes filled with functions for specific types of tasks and analyses
- Extend the basic R functionality
- Some packages contain other packages, for example: tidyverse

Remove the # before this command to install the package tidyverse

ggplot2

- Visualize your data
- Create histograms, barcharts, scatterplots
- Edit the aesthetics of plots

dplyr

- Manipulate your data
- Select variables
- Filter data frames
- Create new variables

readr

- Read in data from .csv or .tsv files
- Can specify variable type in function call

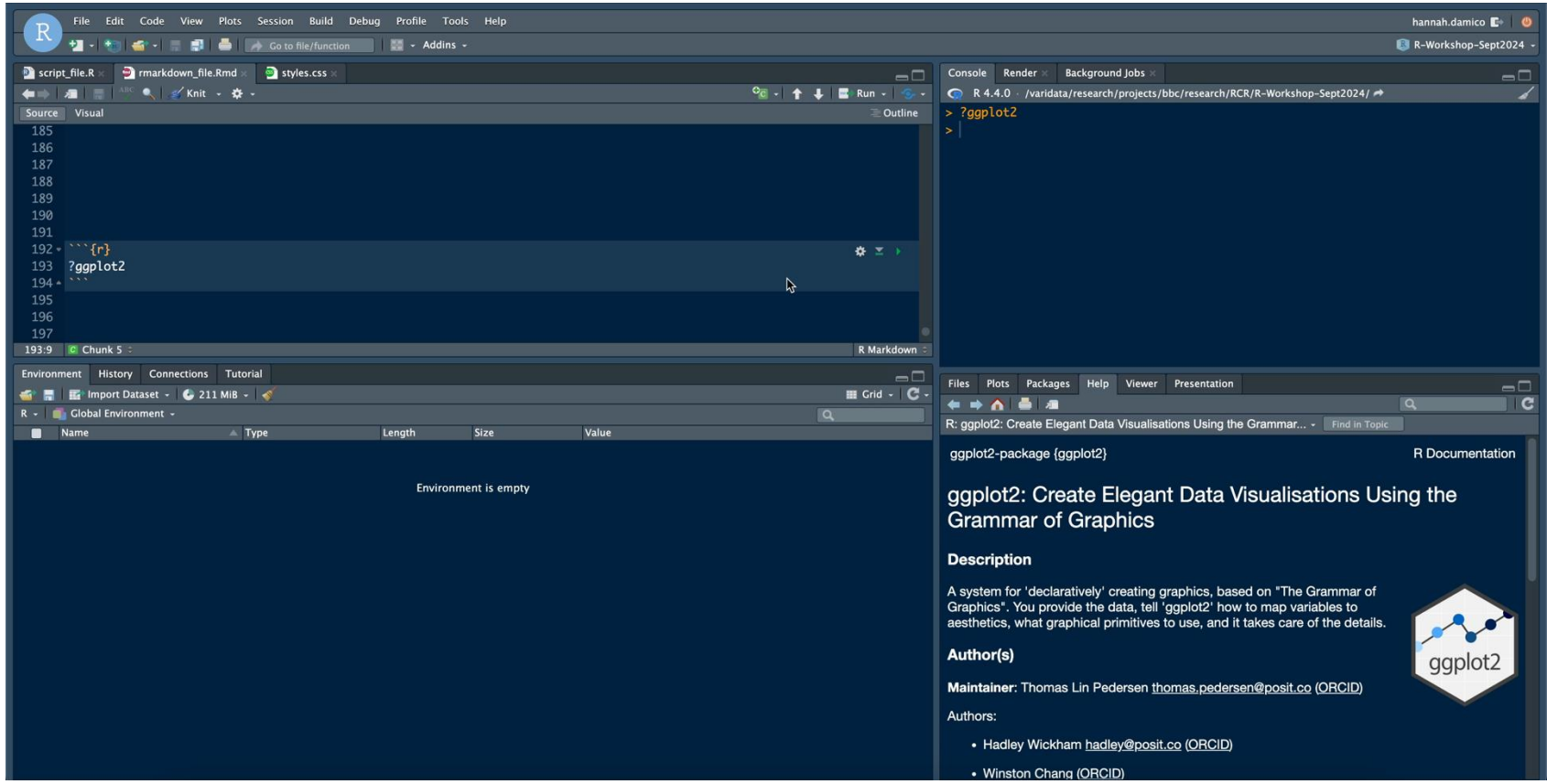
tidyr

- Clean up data
- Variables are columns
- Observations are rows
- Row1: single mouse's weight and tumor size are measured for one timepoint
- Transpose data
- Wide -> long
- Long -> wide

Need ?help with a package or function?

Running **?**

before a function or package will take you to its documentation!



R objects can look like…

Characters & Strings: something in quotes, “A”, “1”, “bio”, “statistics”

Factor: A variable with levels, Treatment: [“A”, “B”, “C”]

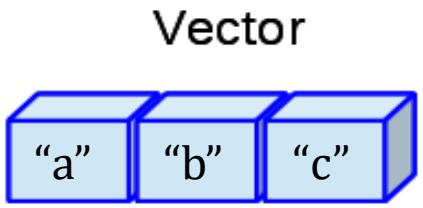
Numeric: fractions, decimals, positive, negative numbers

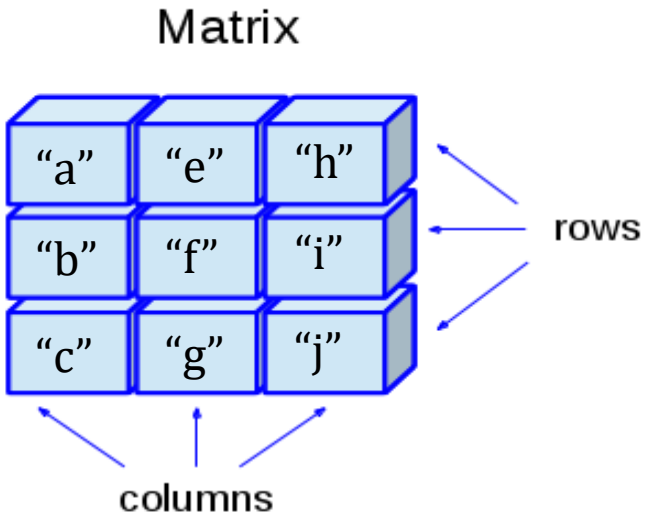
Logical: TRUE or FALSE, sometimes 1 or 0

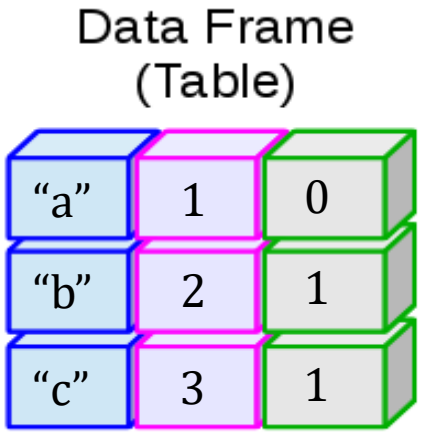
Check in!

https://www.menti.com/al9axfxzmymo

Larger data types in R







Simply put, these hold information in different ways…

Image borrowed from: http://venus.ifca.unican.es/Rintro/dataStruct.html

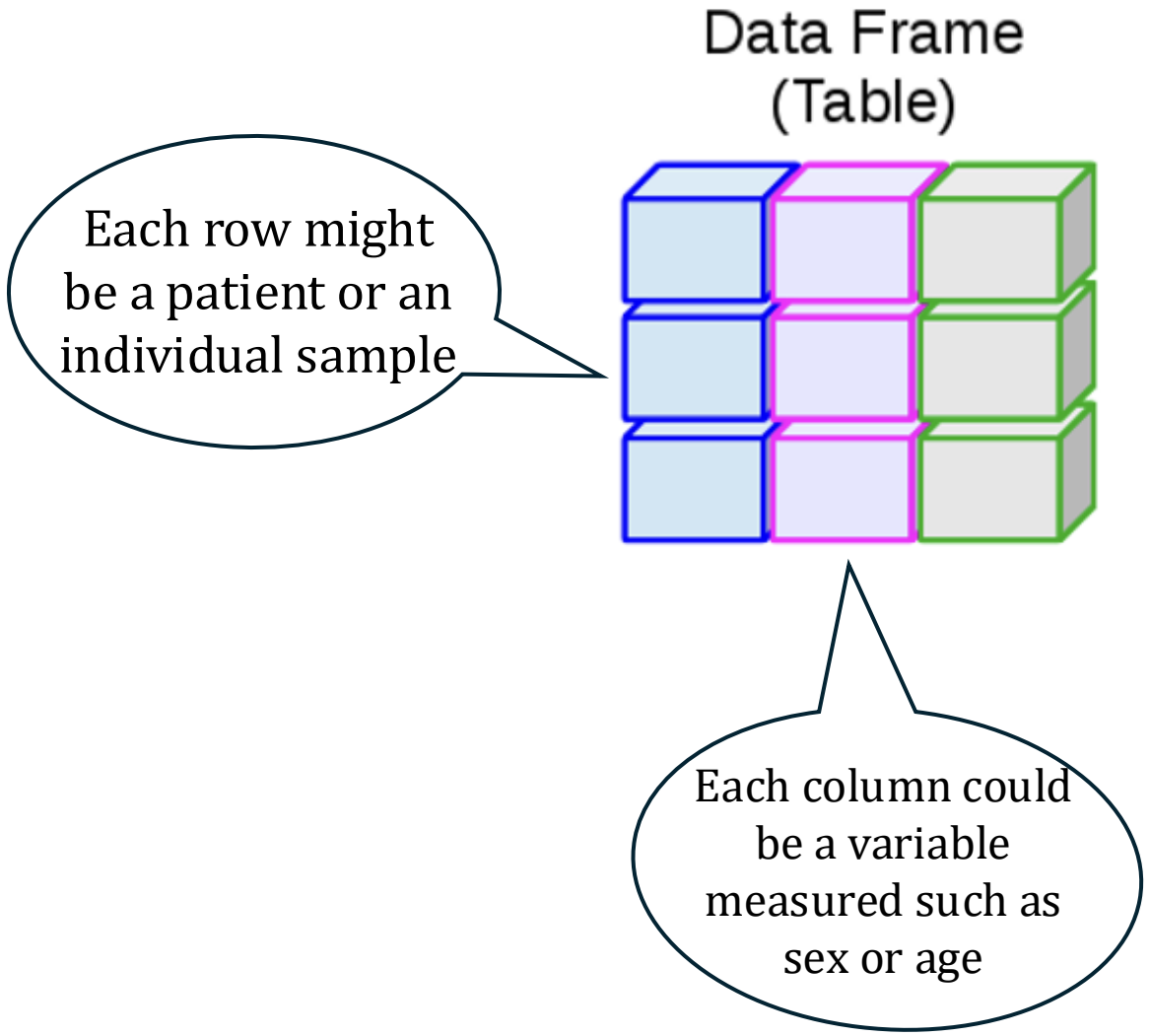
Data frames

Creation vs. Coercion

- data.frame(): create a data frame from scratch
- as.data.frame(): convert an existing object into a data frame

Input Types:

- data.frame() takes vectors or variables and combines them into a data frame
- as.data.frame() takes an object (like a matrix, or even another data frame) and attempts to convert it into a data frame



Reading in data

- read.csv (base) vs. read_csv (readr/tidyverse)
  - Sometimes these read in differently, so it's important to check your data after you've read it in!
- read_tsv

What if I have an excel file?

readxl::read_excel

Viewing your data

Abilities	dim()	glimpse()	str()	nrow()	ncol()	head()	tail()	View()
Provide # of rows	✓	✓	✓	✓		✓	✓	✓
Provide # of columns	✓	✓	✓		✓	✓	✓	✓
Display data type			✓			✓	✓	
Display data as data.frame()						✓	✓	
Select # of rows to view								
Filter data by column								✓

Final Questions

https://www.menti.com/alsbv87coqxi