R RESOURCES

R basics

If you are new to R or need a refresher on the basics, check out these resources geared towards first-time programmers.

Hands-on practice is the best way to learn a programming language, so be sure to actually run the example code. Even better, once you get the example code working, try modifying it in different ways to see what happens. Which changes work, and which ones break? Why?

- Getting Used to R, RStudio, and R Markdown: https://rbasics.netlify.app/
- Hands-On Programming with R: https://rstudio-education.github.io/hopr/
- R for Reproducible Scientific Analysis https://swcarpentry.github.io/r-novice-gapminder/
- `swirl` package - interactive R tutorials: https://swirlstats.com/students.html
- Online courses (most material is free to access):
  - https://www.coursera.org/specializations/jhu-data-science/
  - https://www.coursera.org/specializations/tidyverse-data-science-r

Everyday reference

Once you're feeling somewhat comfortable with the basics of R programming, these references are a great resource for finding examples and best practice recommendations so that you don't have to reinvent the wheel.

- Stack Overflow: https://stackoverflow.com/collectives/r-language
  - Search or post here if you are stuck! Stack Overflow results will also show up in Google searches. Pasting your error message into Google often works well.
- R for Data Science: https://r4ds.hadley.nz/
- R Recipes: https://posit.cloud/learn/recipes
- R Cheatsheets: https://posit.co/resources/cheatsheets/
  - In particular: R Studio, readr, dplyr, tidyr, stringr, lubridate, ggplot2, and quarto
- The R Graph Gallery: https://r-graph-gallery.com/
- R Graphics Cookbook: https://r-graphics.org/
- Tidyverse Style Guide: https://style.tidyverse.org/
  - Guidance on formatting your code for consistency and readability (and a package that does it for you!)

Essential R packages

It's good to be a little cautious about relying on R packages. Anyone can make a package, and sometimes package authors end up being unable to maintain and update them in the long run. However, many packages are incredibly well maintained and some are considered practically essential for modern data science work in R. It would be impossible to list every high-quality R package out there, but these are the ones that nearly everyone should become familiar with:

- Core `tidyverse` packages: https://www.tidyverse.org/packages/
- `quarto` - code notebooks, create documents in Word, PDF, or HTML: https://quarto.org/docs/get-started/hello/rstudio.html
- `janitor` - clean messy data: https://sfirke.github.io/janitor/
- `readxl` - read data from Excel files: https://readxl.tidyverse.org/
  - Note: `readxl` is installed with `tidyverse` but isn't listed as a core package.
- `styler` - automatically formats your code with consistent indentation and spacing. Pretty! https://styler.r-lib.org/

Level up!

If you're feeling more comfortable in R and want to learn more, these resources will help you expand your skillset (and your resume):

- Regular expressions: https://r4ds.hadley.nz/regexps.html
  - Learn how to detect patterns in text. Great for tidying column names and analyzing text data.
- Happy Git with R: https://happygitwithr.com/
  - Using Git/GitHub for version control helps you track changes to your code project over time. It's also great for sharing code and collaborating with other people.
- R Packages: https://r-pkgs.org/
  - Write your own R packages! Great for organizing, documenting, and showcasing your code projects, especially if your code is useful for other people.
- R Shiny: https://shiny.posit.co/
  - Build interactive web apps in R or Python.