

Documentation and Literate Programming

Objective

Understand how to write a README.md using markdown basics and introduce the concept of literate programming with Quarto.

Lesson Outline

- Quarto: an open-source scientific and technical publishing system
 - Allows combining markdown, code, and code output (tables, plots, etc)
 - Wide variety of outputs: html, pdf, word, and more
 - Use cases:
 - Notebook
 - Share results with collaborators
 - Presentations
 - Reproducible manuscripts
- How does it work?
 - `qmd -> knitr -> md -> pandoc -> word, pdf, html, etc.`
- Anatomy of a .Qmd
 - Look familiar if you've used RMarkdown
 - YAML header has metadata and settings
 - Markdown body
 - Code chunks with R, Python, Julia, or other code
- Create .Qmd and play
 - in header: title, author, date
 - in body: headers, italics, bold, code, links
 - Render as html
- Demo visual editor
 - Switch back and forth
 - “Add anything” menu with `cmd- /`
- Running code in code chunks

```
#| label: setup  
#| include: false
```

```
library(tidyverse)
library(palmerpenguins)
library(knitr)
```

```
#| label: fig-box
#| fig-cap: "Boxplot of body mass in grams by penguin sex."
ggplot(penguins, aes(x = sex, y = body_mass_g)) + geom_boxplot()
```

```
#| label: tbl-sample
#| tbl-cap: "Sample sizes of penguins by sex and island"
table(penguins$sex, penguins$island, useNA = "always") |> kable()
```

- Code chunk options
 - Demo some common ones like label, echo: false, fig-cap, code-fold
 - Demo applying to entire doc in YAML header
- Add TOC (toc: true)
- Equations (look up how to cross ref equations during lesson)
- Citations (from DOI or from Zenodo)
- Cross refs
- Callouts
- Mention journal articles (check out AGU format example <https://quarto-journals.github.io/agu/>)
- Point students toward tutorial and full documentation