Getting Started with Quarto and RStudio

Jesse Sadler

1 What is Quarto?

Quarto is a tool for publishing data science documents that mix together code and text. Quarto is the next generation of [R Markdown](#) and can be used to output to a variety of formats from HTML pages to PDFs and websites and books.

Resources

- The [Quarto documentation](#) is very good and covers a wide array of topics.
- [Using R](#) provides a good overview of making a Quarto document with R code and output.
- There is a similar guide for writing Quarto documents in the [RStudio IDE](#).
- The [R Markdown documentation](#) continues to be useful.

Structure of a Quarto document

The structure of Quarto documents is very similar to R Markdown documents.

1. YAML header
2. Markdown text
3. Code chunks
    1. Code fencing with language
    2. Hash pipe chunk options
    3. Code

2 Create document

- File -> New File -> Quarto Document…
- Look through the options.
    - You can create a title
    - Choose an output format. HTML is the default.
    - Engine: Knitr is the engine for R; Jupyter is for Python.
    - Choose Visual or Source editor. See below.
    - Click Create, or you can just choose Create Empty Document.
- After the document opens, save and name the file.

Visual vs Source editor

- The Quarto document may open in the Visual editor. The Visual editor provides a live preview of your markup, while the Source editor shows the plaintext view.
- Flip between Visual and Source mode in upper left.
- Source mode can be nice if you are not used to Markdown and want to practice or prefer seeing styled text.
- We suggest mostly using Source mode.

- Note that switching between Source and Visual modes may change some formatting in your document.

3 YAML header

Both Quarto and R Markdown use YAML to provide metadata and options for the document, though the [syntax and options differ](#) in places.

YAML rules

- Surrounded by lines of 3 dashes.
- Set a single option `key: value` pairs.
- Strings with a colon (:) must be quoted.
- Include multiple values in a list with markdown style indented and on separte lines - or inline with `[one, two, three]`.
- Nest key-value pairs using indentation (2 spaces).

Main options

The primary YAML options are documented on the [authoring tutorial](#), while a full list is available from the [Reference pages](#) of the different formats.

- `author`
- `title`
- `date`: See the [documentation](#) on setting date.
- `format`: Options for format and then options within that format. Default is `html`.
  - In R Markdown this is `output` with default `html_document`.
- `execute`: Set default code execution options for the whole document

```
---
title: "Getting Started with Quarto and RStudio"
author: Jesse Sadler
date: "2025-05-28"
  format:
    html:
      toc: true
      code-fold: true
  execute:
    echo: false
---
```

4 Markdown

[Markdown](#) is a plain text format that is easy to write and read and can be rendered into HTML, making it a widely used format for writing for the internet. Quarto uses an expanded form of Markdown that is based on [Pandoc](#). The main features you will use are easy to learn and universal in the different flavors of Markdown. See the Markdown style guide in the [Quarto docs](#) for a full overview.

- Headings
- Lists
  - Unordered lists (-, *, or +)
  - Ordered lists (1.)

- Paragraphs Paragraphs are created in Markdown by writing text with no indentation. To create another paragraph there must be one or more blank lines separating the paragraphs.
- Text formatting
  - Italics: Text surrounded by * or _ (*italics* or _italics_)
  - Bold: Text surrounded by two * or _ (**bold** or __bold__)
  - Code: Text surrounded by ` (code)
- Links
  - Raw link: <https://quarto.org>
  - Inline link: [Text for link](https://quarto.org)

5 Code Chunks

- Code blocks set off by three back ticks with braces around the language name (e.g. ```` ```{r} ````) are executable.
- The keyboard shortcut in RStudio for a code chunk is `Option + Cmd + I` or `Alt + Cntr + I` or from the Code menu -> Insert Chunk.

Evaluation chunk options

Chunk options are set off by the hash pipe (#|).

- `label`: Label for the chunk. This can be useful for debugging error messages.
- `eval: true`: Whether code is evaluated. If `false`, the code is not run. This is useful for displaying example code.
- `echo: true`: Whether the code is shown in the output. If `false`, no code is shown in the output.

This is different from RMarkdown, where chunk options are set within the curly braces of the chunk header and are set as arguments: {r echo = FALSE, eval = TRUE}.

```
```{r}
#| label: setup
#| echo: true
# Write code here
```
```

Writing and executing code

- Write code as you would in an R script.
- Code chunks can be executed (run) from the Run menu or with keyboard shortcuts:
  - Line by line as if in an R script with `Cmd + Return` or `Cntr + Enter`.
  - Per chunk with the `Play` button at the top right of the chunk or `Cmd + Shift + Return` or `Cntr + Shift + Enter`.
  - All chunks with Run all or `Option + Cmd + R` or `Alt + Cntr + R`
  - All chunks are executed when the document is rendered.
- By default, any printed output—anything printed to the console or any plots—will be output inline, directly below the chunk. This can be changed in Tools -> Global Options -> R Markdown -> Show output inline for all R Markdown documents.

6 Render

- Rendering a quarto document formats all the Markdown, executes all the code set to be evaluated, and outputs all plots and data frames, creating a new document to the chosen format.
  - The language used for R Markdown documents is to *knit*.
- You can render a document in the following ways:
  1. Click on the Render button in the top toolbar.
  2. From File menu -> Render Document or with keyboard command `Cmd + Shift + K` or `Cntr + Shift + K`
  3. Using the quarto package: `quarto_render("document.qmd")`.
  4. Use Quarto on the command line: `quarto render document.qmd`
- Rendering a document opens up a preview in a Viewer pane in RStudio or in your browser. You can change the Preview setting in Tools -> Global Options -> R Markdown -> Show output Preview in Window or Viewer Pane.