

# Build your academic website with Quarto

Scientific workflows: Tools and Tips 

2026-04-16

# What is this lecture series?

## Scientific workflows: Tools and Tips

 Every 3rd Thursday  4-5 p.m.  Webex

- One topic from the world of scientific workflows
- Material provided [online](#)
- If you don't want to miss a lecture
  - [Subscribe to the mailing list](#)
- For credit points: Send me a short message (Email or Webex)

# Why an academic website?

- Showcase your work, publications, projects, ...
- It's *yours*, you can take it with you (unlike institutional pages)
- A central place where people can find you and your research

# Tools

There are many tools for static websites, today we use **Quarto**

- Simple, works in RStudio / Positron / VS Code
- Easy to publish for free
- Excellent [documentation](#)

# What a finished site can look like

Some examples of nice academic Quarto websites:

- [Silvia Canelón](#)
- [Andrew Heiss](#)
- [Sam Csik](#)
- [John Helveston](#)

Goal for today: Build (and publish) a simple personal website with Quarto to perfect later!

# Plan for today

1. Quick **Quarto** intro/refresher
2. Basic **website** from scratch
3. Understand and customize an **academic website template**
4. **Publish** your website via GitHub Pages

To **follow along** you only need:

- Recent version of RStudio (or Positron, VS Code)
- GitHub account (for publishing, optional)

# What is Quarto?

Quarto is an open-source scientific and technical publishing system

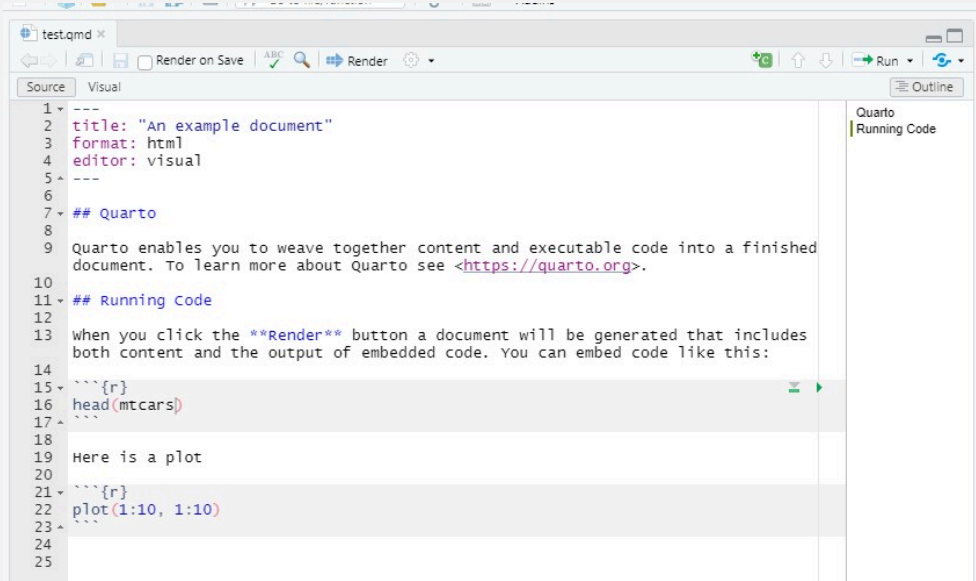
Quarto can create many types of output:

- **Documents:** HTML, PDF, Word
- **Presentations:** HTML, Powerpoint
- **Books:** HTML, ePub, PDF
- **Websites:** HTML
- ...

# What is Quarto?

A Quarto document (`.qmd`) is basically a text file

When you **render** it, Quarto turns it into a nice HTML/PDF/etc.



```
1 ----
2 title: "An example document"
3 format: html
4 editor: visual
5 ----
6
7 ## Quarto
8
9 Quarto enables you to weave together content and executable code into a finished
10 document. To learn more about Quarto see <https://quarto.org>.
11
12 ## Running Code
13
14 when you click the Render button a document will be generated that includes
15 both content and the output of embedded code. You can embed code like this:
16
17 ```{r}
18 head(mtcars)
19 ```
20
21 Here is a plot
22
23 ```{r}
24 plot(1:10, 1:10)
25 ```
```

## An example document

### Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>.

### Running Code

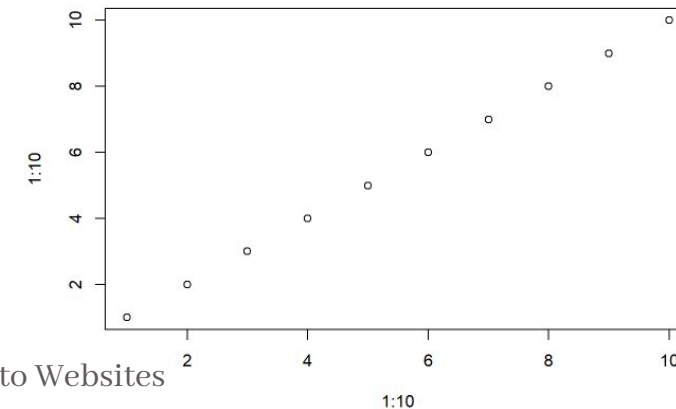
When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```
head(mtcars)
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

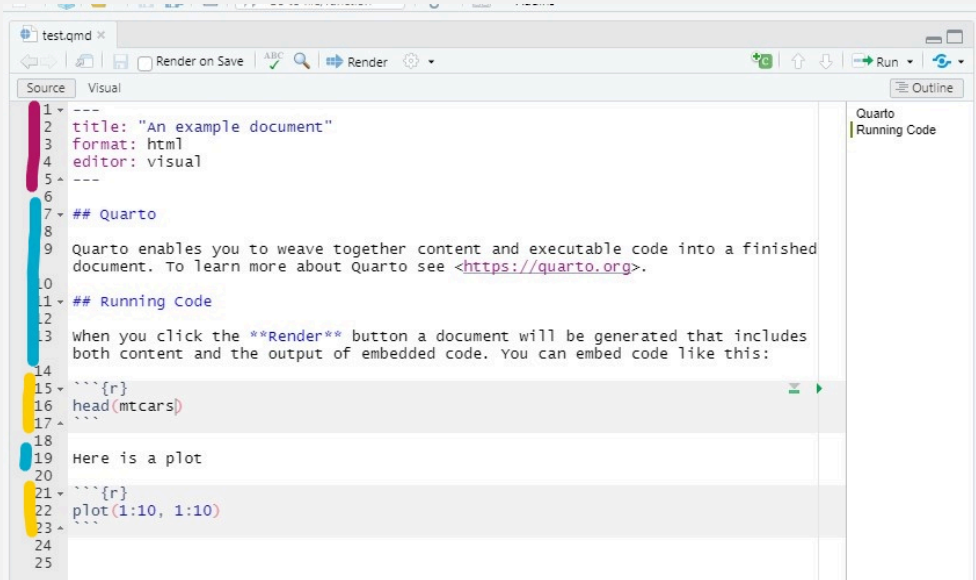
Here is a plot

```
plot(1:10, 1:10)
```



# Basic components of a Quarto document

A Quarto document (`.qmd`) is basically a text file



The screenshot shows a Quarto document editor with a source code view on the left and a rendered output view on the right. The source code is as follows:

```
1 ---
2 title: "An example document"
3 format: html
4 editor: visual
5 ---
6
7 ## Quarto
8
9 Quarto enables you to weave together content and executable code into a finished
10 document. To learn more about Quarto see <https://quarto.org>.
11
12 ## Running Code
13
14 when you click the Render button a document will be generated that includes
15 both content and the output of embedded code. You can embed code like this:
16
17 ```{r}
18 head(mtcars)
19
20
21 Here is a plot
22
23 ```{r}
24 plot(1:10, 1:10)
25
```

The rendered output on the right shows the title "An example document" and the text "Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see <https://quarto.org>." followed by the output of the R code chunks: "head(mtcars)" and a plot of the numbers 1 to 10.

A Quarto document (`.qmd`) has three components:

- **YAML header** for metadata and configuration
- **Markdown text**
- (Optional) **code chunks** (R, Python, Julia, ...)

# The YAML header

```
---  
title: "An example document"  
author: "Selina Baldauf"  
date: "2023-05-11"  
format:  
  html:  
    theme: cosmo  
    toc: true  
---
```

- Between `---` at the top of the file
- Metadata + config in “key: value” form
- Careful with indentation and alignment of nested elements

# The Markdown text

You write content like this:

```
# A section header

## A subsection header

Some bold and italic text and a
[link to Quarto](https://quarto.org).

- First bullet
- Second bullet
- Third bullet
```

And it will render like this:

## A section header

### A subsection header

Some **bold** and *italic* text and a [link to Quarto](https://quarto.org).

- First bullet
- Second bullet
- Third bullet

# A Quarto website from scratch

Quarto website = a folder of `.qmd` files for content + `_quarto.yml` for configuration

# Project structure

Create a Website project in R Studio:

File > New Project > New Directory > Quarto Website

File	Purpose
<code>_quarto.yml</code>	Site-wide config (navbar, theme, etc.)
<code>index.qmd</code>	Home page
<code>about.qmd</code>	About page
<code>styles.css</code>	Small custom style tweaks
<code>project-name.Rproj</code>	The R project file to open the project

# Quarto website summary

- Quarto website = a folder of `.qmd` files for **content** + `_quarto.yml` for **configuration**
- To add a page: create a `.qmd` and add it to the navbar in `_quarto.yml`
- See your changes: Render the project

# Building an academic website

Create a personal website from a template

# Get the template

1. Download the template from [here](#)
2. Unzip and open the `.Rproj` file in RStudio
3. Click **Render** to see the template site

The image shows a GitHub repository page for 'quarto-academic-template' by user 'selinaZitrone'. The repository is public and has 0 stars, 0 forks, and 0 watchers. The main branch is selected. A dropdown menu is open over the repository, showing options to clone the repository. The 'Clone' option is selected, and the dropdown shows three methods: HTTPS, SSH, and GitHub CLI. The HTTPS URL is displayed as 'https://github.com/selinaZitrone/quarto-academic-template'. Below the URL, there are options to 'Open with GitHub Desktop', 'Open with Visual Studio', and 'Download ZIP'. The 'Download ZIP' option is highlighted with a pink box. The repository files list includes 'posts', '.gitignore', '\_quarto.yml', 'about.qmd', 'blog.qmd', 'custom.scss', 'index.qmd', 'profile.jpg', 'publications.qmd', and 'quarto-academic-template.Rproj'. The right sidebar shows the 'About' section with no description, 'Releases' section with no releases published, 'Packages' section with no packages published, and 'Contributors' section with 1 contributor.

# Project structure

File / Folder	Purpose
<code>_quarto.yml</code>	Site-wide config (navbar, theme, footer)
<code>index.qmd</code>	Home page
<code>about.qmd</code>	About page with your profile/CV
<code>publications.qmd</code>	Publications page
<code>blog.qmd</code> + <code>posts/</code>	Blog listing + individual posts
<code>styles.css</code>	Small custom style tweaks

# Customize the `_quarto.yml`

## Exercise (5 min):

1. Change the site `title` and `footer` text
2. Browse the themes from the `theme list` and swap the `theme` value for what you like most
3. Render and see what changed

## Learn more about `_quarto.yml` configuration

- `Website Navigation`: Navbar, footer options, etc.
- `Theming options`: Change some colors, font sizes, etc.

# Customize the landing page `index.qmd`

## Exercise (10 min):

1. Replace the placeholder text with a short bio
2. YAML: Update the `links`, delete what doesn't apply, add what's missing
3. YAML: Try a different `template` and re-render
4. (Optional) Add a photo to the project folder and update the image path


 Learn more about "About page templates"

Check out the [Quarto docs](#)

# Customize `about.qmd` and `publications.qmd`

## Exercise (10 min):

Replace the placeholder content with your own content. You can also just add some lines and experiment with Markdown formatting. You can do the full content later.


 Learn more about Markdown

Checkout the [Markdown Basics](#) on the Quarto website

# Add a blog post

## Exercise (10 min):

1. Create a new folder in `posts/`, add an `index.qmd` with a title, date, and a few lines of text
2. Render and check that it appears in the blog listing
3. (Optional) Try changing listing `type` from `default` to `grid` or `table` in `blog.qmd` and render to see the difference

 Learn more about listing page types


Check out the [Quarto docs](#)

# Custom styling with SCSS

Themes are a starting point. If you want to tweak colors, fonts, or spacing, you can use SCSS.

Wire it into `_quarto.yml`:

```
format:  
  html:  
    theme: [cosmo, custom.scss]
```

 Learn more about theming and custom styling

- [Quarto theming docs](#)
- [Different variables](#) you can set in custom SCSS

# Publishing a website

# How publishing works

1. When you **render**, Quarto produces a folder of HTML, CSS, and image files (default: `_site/`)
2. Upload these files to a **hosting service** that stores the files and serves them at a public URL

**Hosting options:** GitHub Pages, Netlify, Posit Connect Cloud, university web server, ...

# Publish your website

 Exercise (10 min):

Follow the step-by-step guide. If your site is live, **drop the URL in the chat!**

# Updating your site later

1. **Edit** your `.qmd` files
2. **Render**
3. **Commit and push** (Git pane or GitHub Desktop)
4. GitHub rebuilds automatically, changes are live in ~1 min

# Wrap up

Where to go from here

# Today

- Built a Quarto website from scratch
- Customized the main page types: **About pages, plain pages, listing pages**
- Configured the site through `_quarto.yml` (themes, navbar, footer)
- Published to GitHub Pages

# Inspiration: academic Quarto sites

People are nice and provide source code for their websites. If you see elements that you like, you can check out how they did it and adapt it for your own site.

Site	Source code
Silvia Canelón	<a href="https://github.com/spcanelon/silvia">https://github.com/spcanelon/silvia</a>
Andrew Heiss	<a href="https://github.com/andrewheiss/ath-quarto">https://github.com/andrewheiss/ath-quarto</a>
Sam Csik	<a href="https://github.com/samanthacsik/samanthacsik.github.io">https://github.com/samanthacsik/samanthacsik.github.io</a>
John Helveston	<a href="https://github.com/jhelvy/jhelvy_quarto">https://github.com/jhelvy/jhelvy_quarto</a>

More examples from [Gang He's curated list](#)

# Next steps

- Customize your website
  - Add relevant content and pages
  - Customize the look with **SCSS** for custom styling
- Use your **custom domain** ([yourname.com](#)) for your GitHub Pages site
- Get an overview of other Quarto website features on the [Quarto docs](#).

# Next lecture/workshop

Topic tba

 21.05.2026  4-5/6 p.m.  Webex

 [Subscribe to the mailing list](#)

 For topic suggestions and/or feedback [send me an email](#)

# The end :)

Questions?