

# Setting Up a Minimal Quarto Blog: A Learning Journey

Notes to myself on getting started with Quarto blogging

Ronald G. Thomas

2025-11-17



Figure 1: Getting started with Quarto blogging

*Quarto makes technical blogging surprisingly accessible*

## Introduction

I didn't really know how straightforward it was to set up a Quarto blog until I decided to create one for my lab's website. Like many R users, I was familiar with R Markdown, but Quarto seemed like this big, complicated new thing to learn

Turns out, you can get a working blog up and running with just a handful of files and a few terminal commands. No need for complex configurations or web development knowledge.

Here's what I set out to understand:

## Motivations

**Why explore Quarto blogging?** - Needed a platform for sharing lab research and tutorials - Wanted something that integrates seamlessly with R and RStudio - Required literate programming capabilities (mix code and narrative) - Preferred static site generation (fast, secure, easy hosting)

## Objectives

**What I wanted to accomplish:** 1. Understand the minimal file structure needed for a Quarto blog 2. Learn the essential configuration options in `_quarto.yml` 3. Create a working blog with at least one post 4. Document the process for future reference

**Disclaimer:** This is my learning journey with Quarto blogging. If you spot errors or have better approaches, please let me know!

## What is Quarto?

Before diving in, let's clarify what Quarto actually is. Quarto is the next generation of R Markdown - it's a multi-language publishing system that works with R, Python, Julia, and Observable. It leverages `knitr` and `pandoc` under the hood, but adds a unified command-line interface and extended capabilities.

Think of it as R Markdown grown up and language-agnostic!

## Getting Started: Creating a Blog

The recommended way to start is using Quarto's command-line tool:

```
quarto create-project qblog --type website:blog
```

This generates a project folder with this structure:

```
qblog/  
  _quarto.yml  
  about.qmd  
  index.qmd  
  posts/  
    _metadata.yml  
    post-with-code/  
      image.jpg  
      index.qmd  
    welcome/  
      index.qmd  
      thumbnail.jpg  
  profile.jpg  
  styles.css
```

Wow, that's a lot of files for a “minimal” setup! But here's the thing: **you don't actually need all of this**

## What You Actually Need (Minimal Structure)

After experimenting, I found you can strip this down to just 4 essential files:

```
minimal_qblog/  
  _quarto.yml  
  index.qmd  
  posts/  
    index.qmd  
    post1.qmd
```

Let me show you what goes in each file:

### The Site Configuration

**File: \_quarto.yml**

```
project:  
  type: website  
  
website:  
  title: "Thomas Lab"  
  navbar:
```

```

left:
  - href: posts/index.qmd
    text: Blog

format:
  html:
    theme: cosmo

```

This is your main configuration file. It defines: - Project type (website) - Site title - Navigation structure - Default theme (cosmo is clean and professional)

## The Home Page

**File: index.qmd**

```

---
title: "Thomas Lab"
---

**Director: Professor Ronald G. Thomas**
School of Public Health
UC San Diego
La Jolla, California

Focused on new and useful data science technologies.

```

Super simple! Just basic markdown for your landing page.

## The Blog Listing Page

**File: posts/index.qmd**

```

---
title: "Blog"
listing: default
---

```

The magic happens with `listing: default` - this tells Quarto to automatically create a list of all posts in the `posts/` directory

## Your First Post

**File: posts/post1.qmd**

```
---  
title: "First Post"  
date: "2025-11-17"  
---  
  
# Introduction  
  
Minimal text for first post. This is all you need to get started!
```

That's it! Just a title, date, and some content.

## Building and Previewing

Now here's where it gets fun. To see your blog:

```
cd minimal_qblog  
quarto preview
```

Quarto will: 1. Generate a `_site/` directory with all HTML, CSS, and assets 2. Launch a local web server 3. Open your browser automatically 4. Live-reload when you edit files

The first time you run this, Quarto creates CSS based on your chosen theme. Everything is self-contained in the `_site/` folder.

## What It Looks Like

Here's what the minimal blog renders to:

## Thomas lab

**Director: Professor Ronald G. Thomas**

School of Public Health

UC, San Diego

La Jolla, California

Focused on new and interesting data science technologies.

Figure 2: Minimal Quarto blog structure

Clean, functional, and ready to customize!

## Adding More Posts

To add new posts, just:

1. Create a new directory in `posts/`: `posts/my_new_post/`
2. Add an `index.qmd` file to that directory
3. Write your content
4. The blog listing updates automatically

**Pro tip:** Use directories for posts (not files) so you can keep images and other assets organized alongside your content.

## Lessons Learnt

Here's what I took away from setting up my first Quarto blog:

**Conceptual Understanding:** - Quarto is simpler than it looks - 4 files get you started - The `listing` feature automatically creates blog indexes - Static site generation means fast, secure hosting anywhere - Quarto's defaults are sensible and well-designed

**Technical Skills:** - Using `quarto create-project` command-line tool - Understanding `_quarto.yml` structure and options - Organizing content with the `posts/` directory pattern - Live preview workflow with `quarto preview`

**Gotchas and Pitfalls:** - Don't overthink it - start minimal and add complexity later - The `listing: default` option is key for automatic post lists - First `quarto preview` takes longer (generating CSS) - Each post needs its own directory with `index.qmd` for proper organization

## Limitations

This minimal approach has some constraints:

- **No advanced features:** Missing tags, categories, search, RSS feeds
- **Basic styling:** Using default theme without customization
- **Limited navigation:** Only one blog link in navbar
- **No about page:** Stripped out biographical content
- **Manual organization:** No date-based or category-based post organization

For a personal blog or lab website, these limitations might not matter. For a professional publication, you'll want more features.

## Opportunities for Improvement

If I had more time, here's what I'd add:

1. **Custom styling** - Override theme CSS for lab branding
2. **Categories and tags** - Better post organization and discovery
3. **Search functionality** - Built-in search for larger blogs
4. **RSS feed** - Auto-generated feed for subscribers
5. **About page** - Team bios and lab information
6. **Social sharing** - Make posts easy to share on Twitter/LinkedIn

## Wrapping Up

So that's my journey setting up a minimal Quarto blog! Turns out it's surprisingly simple - just 4 files and a couple of terminal commands. The `quarto create-project` starter template is nice, but you really only need the bare essentials to get going.

**Main takeaways:** - Start with 4 files: `_quarto.yml`, `index.qmd`, `posts/index.qmd`, and a sample post - The `listing: default` option handles blog post indexing automatically - Quarto handles all the CSS and HTML generation for you - Live preview makes writing and editing really smooth

I learned a lot working through this, especially about how Quarto's project structure works and how minimal you can really go. Starting simple and adding features gradually is way less overwhelming than trying to understand everything at once.

**If you're trying this yourself:** - Don't be intimidated by the full template - strip it down! - Use `quarto preview` constantly while building - Keep posts in directories (not loose files) for better organization - Start blogging first, customize styling later

Thanks for following along!

## See Also

Related resources:

- [Official Quarto Documentation](#) - Comprehensive guide to blogging
- [Awesome Quarto](#) - Curated list of Quarto resources
- [Samantha Csik's Quarto Workshop](#) - Excellent hands-on tutorial

**Key Resources:** - [Eric Ekholfm - Modifying Default Quarto Blog](#) - Great customization tips - [Allison Hill - We Don't Talk About Quarto](#) - Migration guide from R Markdown

---

## Reproducibility

**Tools:** Quarto CLI (latest version) **Code:** All examples shown in this post  
**Files:** Example minimal blog structure documented above

---

## Let's Connect!

*Have questions, suggestions, or spot an error? Let me know!*

- **Twitter/X:** [@rgt47](#)
- **GitHub:** [rgt47](#)
- **Email:** [Contact form](#)



**Please reach out** if you: - Spot errors or have corrections - Have suggestions for improvement - Want to discuss Quarto blogging approaches - Have questions about implementation - Just want to connect!