

Weekly Journal

Table of contents

| | | |
|-----------|--|-----------|
| 1 | Step 1 – Create a new RStudio Project | 2 |
| 2 | Step 2 – Install and load the course package | 4 |
| 3 | Step 3 – Download your starter journal | 5 |
| 4 | Step 4 – Load the Journal Helper Function | 7 |
| 5 | Step 5 – Add a new week chapter | 8 |
| 6 | Step 6 – Handling Multiple Entries on the Same Day | 10 |
| 7 | Step 7 – Choose your prompt, then write | 11 |
| 7.1 | Choose one prompt to answer | 11 |
| 7.2 | Response | 11 |
| 8 | Step 8 – Render the entire journal (unified output) | 13 |
| 9 | Step 9 — Submit via GitHub | 15 |
| 10 | Visual Workflow | 16 |
| 11 | FAQ | 19 |
| 12 | Weekly checklist (quick self-audit) | 20 |

Start with the **Student Guide** below.

Your **course journal** is a Quarto **book**. Each **week** has a chapter with **three prompts**—you answer **one**.

Word ranges are enforced when you render:

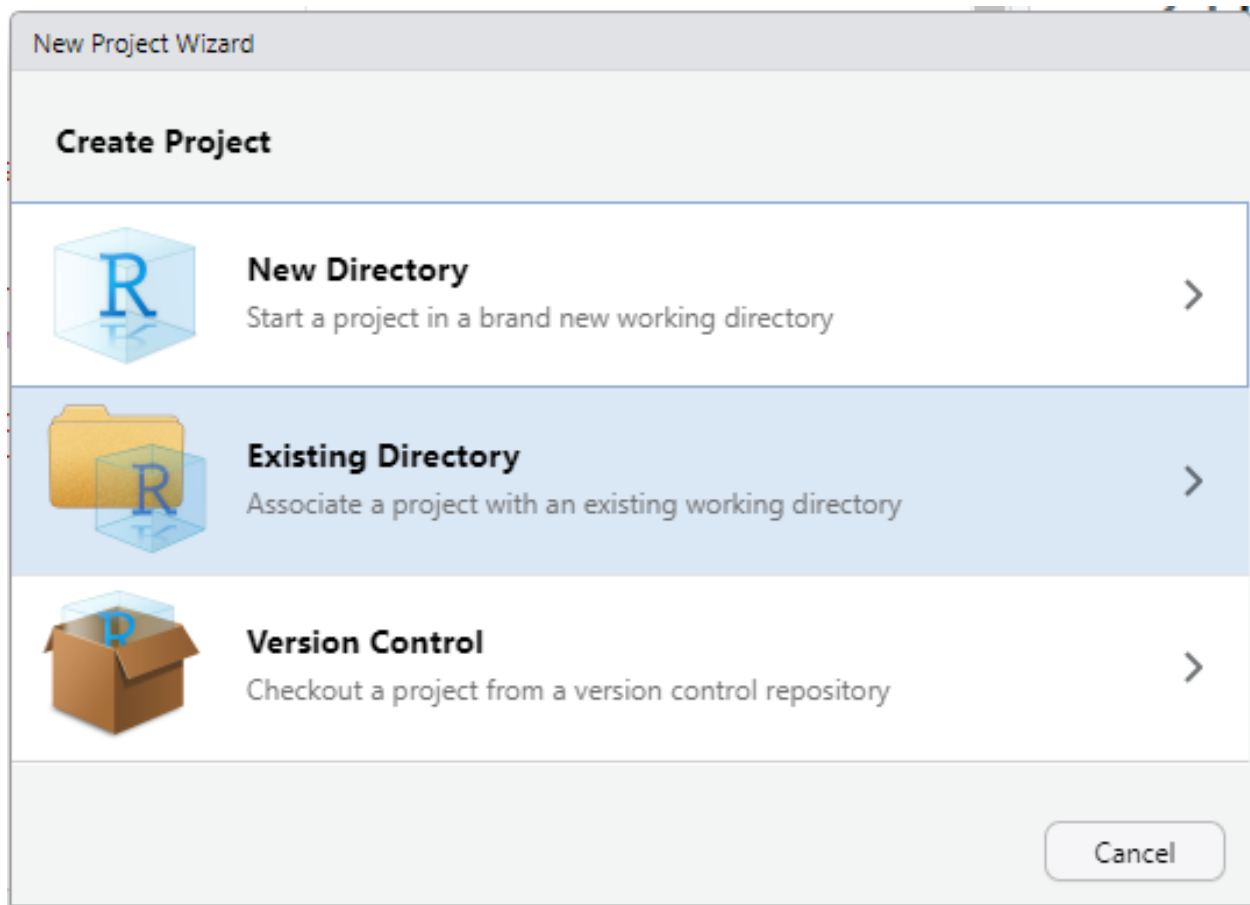
- **MC 451: 250-300** words
 - **MC 501: 450-500** words
-

Chapter 1

Step 1 – Create a new RStudio Project

1. Open **RStudio** -> **File** -> **New Project** -> **New Directory** -> **New Project**.
2. Choose a location (e.g., `Documents/MC451/`) and name it (e.g., `journal_project`).

FIGURE



Chapter 2

Step 2 – Install and load the course package

In the **Console**:

```
pak::pak("SIM-Lab-SIUE/mccoursepack")  
library(mccoursepack)
```

If prompted, restart R after installation.

FIGURE

```
> library(mccoursepack)  
mccoursepack: run `mccourse_setup()` once to install Quarto, TinyTeX, and core packages.  
> mccourse_setup()  
— mccoursepack: setup —  
Your system is ready to build packages!  
✓ Quarto available: 1.7.33  
✓ TinyTeX is already installed.  
✓ Setup finished. If Quarto/TinyTeX were just installed, restart RStudio.
```

Chapter 3

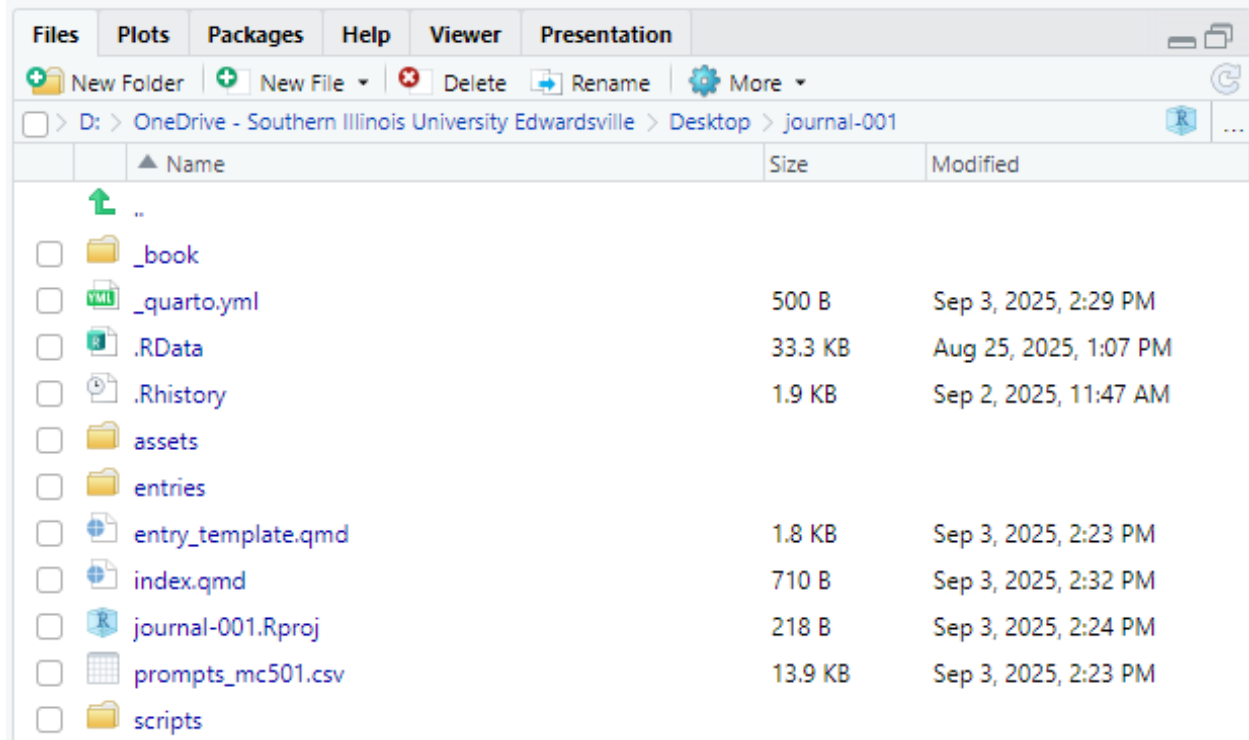
Step 3 – Download your starter journal

Run:

```
download_journal("mc451", dest = ".") # or "mc501"
```

This populates your project's main folder with the complete journal structure.

FIGURE



Chapter 4

Step 4 – Load the Journal Helper Function

Before you can add a new entry, you need to load the helper function that creates the file for you.

1. In the **Files** pane (bottom-right), navigate into the `scripts` folder.
2. Click on the file `new_journal_entry.R` to open it in the editor.
3. Click the **Source** button in the top-right corner of the editor pane.

This loads the `new_journal_entry()` function into your R environment for the current session.

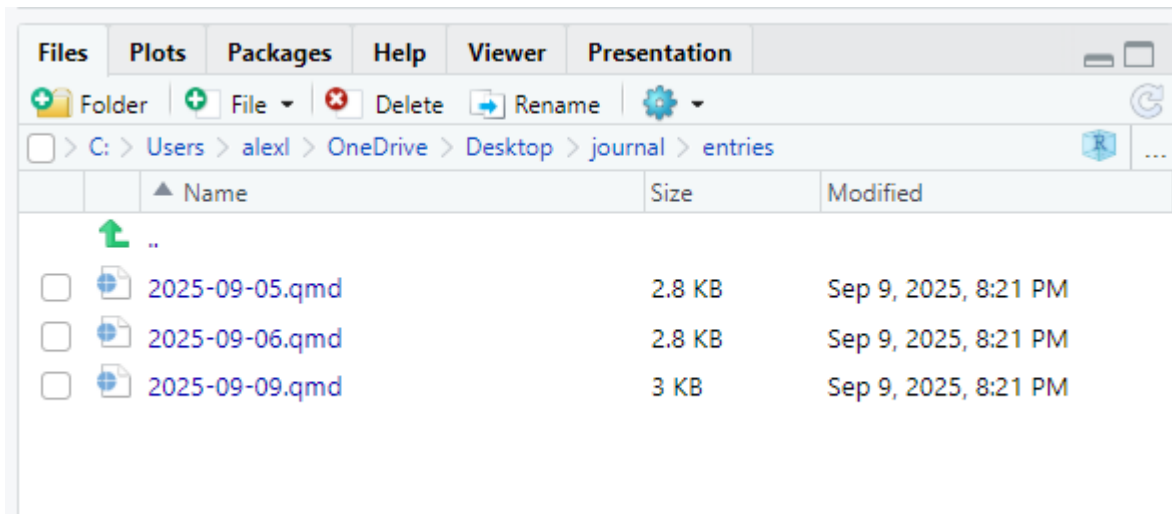
Chapter 5

Step 5 – Add a new week chapter

In the **Console** pane at the bottom-left, run:

```
# Pick your week number (2-14). Example:  
new_journal_entry(week = 3)
```

FIGURE



What happens:

1. A new chapter is created and named with **today's date**: entries/YYYY-MM-DD.qmd.

2. The **three prompts for that week** are inserted.
3. The correct **word range** for your course is set.
4. `_quarto.yml` is **automatically updated** to add the new chapter to the book's table of contents.

FIGURE

```
> new_journal_entry(week = 3)
Created: entries/2025-09-03.qmd
> new_journal_entry(3)
Created: entries/2025-09-03.qmd
```

Troubleshooting: If you run the command twice **on the same day**, the file will be overwritten (same date name). If you need to keep both, rename the first file (e.g., `2025-09-03b.qmd`) before running again.

Chapter 6

Step 6 – Handling Multiple Entries on the Same Day

The `new_journal_entry()` function names files by date (e.g., `2025-09-04.qmd`). If you want to write a second entry on the same day for a different week’s prompt, you should rename the first file to something more descriptive before creating the new one.

For example, you could rename `entries/2025-09-04.qmd` to `entries/2025-09-04-week3.qmd`.

The good news is that you **do not need to manually edit** `_quarto.yml`. The journal is set up to automatically find all `.qmd` files in the `entries/` directory every time you render it. This means you can add, remove, or rename your entry files, and the book will always build correctly.

Chapter 7

Step 7 – Choose your prompt, then write

Open the new file (e.g., entries/2025-09-03.qmd). You'll see **Choose one prompt** first, then **Response**.

Keep **ONE** prompt and delete the other two in the *Choose one prompt* section. Then write your answer **only** inside the **Response** block.

7.1 Choose one prompt to answer

Prompt A: r params\$p1 **Prompt B:** r params\$p2 **Prompt C:** r params\$p3

7.2 Response

*Write your answer to **one** of the prompts here. Do not write anything else in this chapter.*

Stay in range:

- **MC 451:** 250-300 words
- **MC 501:** 450-500 words

FIGURE

```
14
15- ## Choose **one** prompt to answer
16
17 > **Prompt A:** `r params$p1`
18 |
19 > **Prompt B:** `r params$p2`
20
21 > **Prompt C:** `r params$p3`
22
23- ---
24
25- ## Response
26
27 <!-- RESPONSE-START -->
28 *Write your answer to **one** of the prompts here. Do not write anything else in this
chapter.*
29 <!-- RESPONSE-END -->
30
31- ---
```

Chapter 8

Step 8 – Render the entire journal (unified output)

You have two equivalent options. Use **one** of them:

A) Click **Render** on `index.qmd`

- In the Files pane, click on `index.qmd`.
- Click the **Render** button in the editor toolbar.

B) Use the Console

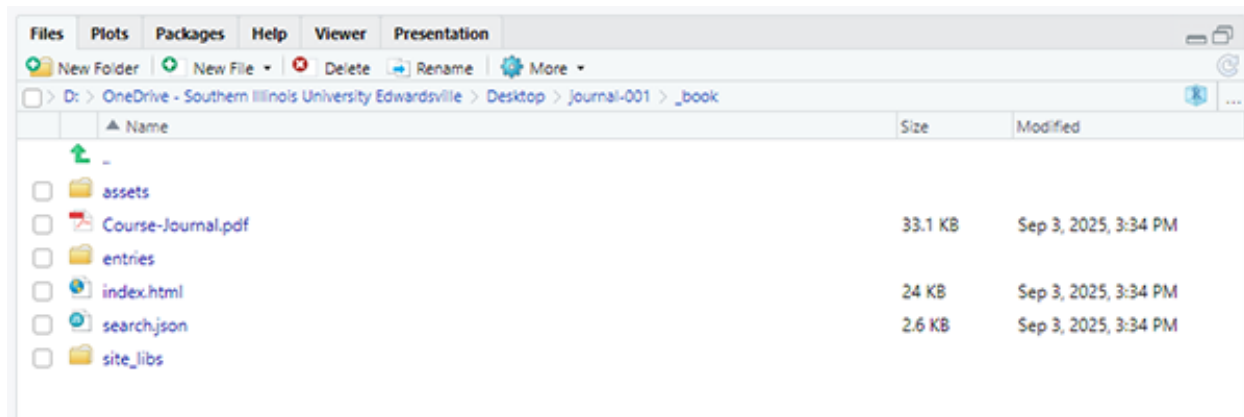
```
# Run from the project's root directory
quarto render
```

Outputs are in the `_book/` directory:

- An **HTML** book (always, multi-page site with navigation = unified entity).
- **PDF/EPUB** if you have TinyTeX installed.

If a PDF fails to build, run:

```
quarto install tinytex  
# or  
mccourse_setup()
```

FIGURE

Tip: Rendering a single chapter file will **not** produce the unified journal. Always render `index.qmd` (or run `quarto render`) to build the whole book.

Chapter 9

Step 9 — Submit via GitHub

When you add or edit a chapter, **two files are modified**:

1. The new chapter file: `entries/YYYY-MM-DD.qmd`
2. The book’s table of contents: `_quarto.yml` (updated automatically by the pre-render script)

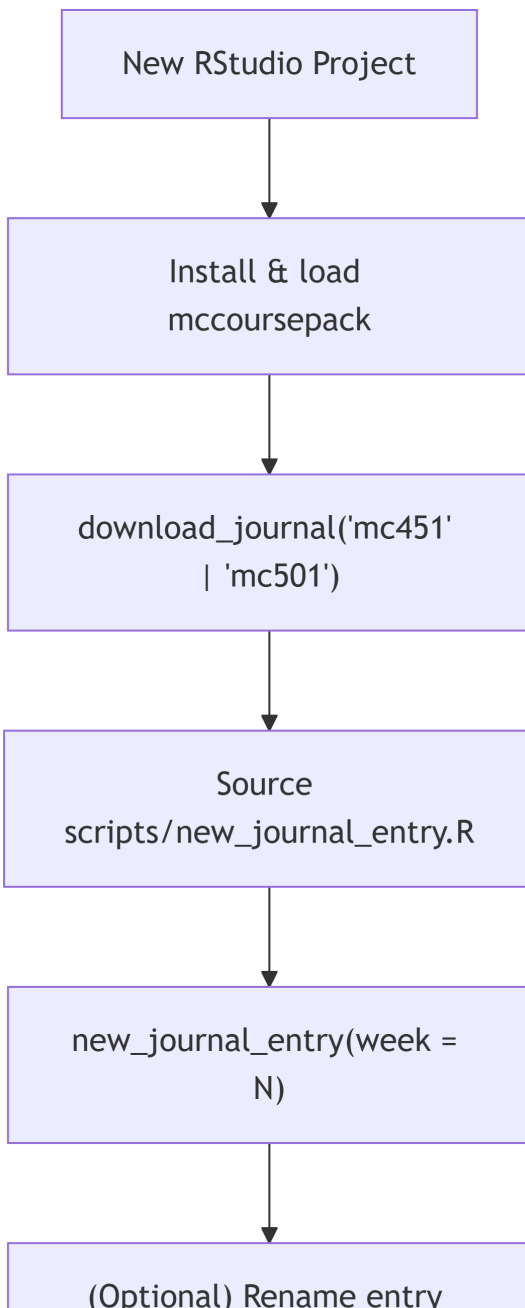
In RStudio:

1. Open the **Git** pane.
2. **Stage** both changed files.
3. Add a commit message (e.g., “Week 3 journal entry”).
4. **Commit** and **Push**.

If you forget to commit the changes to `_quarto.yml`, your new chapter won’t appear in the published book.

Chapter 10

Visual Workflow





Chapter 11

FAQ

It didn't ask me for the week number. Is that okay? Yes. In RStudio you usually call the function directly: `new_journal_entry(week = 5)`.

My word count says “Out of range.” Edit until you are within the range for your course.

The new chapter doesn't appear in the rendered book. Be sure you rendered the **whole book** (Render `index.qmd` or run `quarto render` from the project root). The pre-render script automatically updates the chapter list, but only when the entire book is rendered. Also, make sure you have committed **both** the new entry file and the automatically modified `_quarto.yml`.

Can I revise later? Yes. Edit the same `entries/YYYY-MM-DD.qmd`, then commit and push again.

Chapter 12

Weekly checklist (quick self-audit)

- ☐ I created/updated a chapter for **this week** using `new_journal_entry(week = N)`.
- ☐ I sourced the `new_journal_entry.R` script first.
- ☐ I kept **one** prompt and deleted the other **two**.
- ☐ I wrote my answer **inside the Response block** only.
- ☐ My word count is **in range**.
- ☐ I **rendered the whole book** by rendering `index.qmd` (or running `quarto render`).
- ☐ I **staged, committed, and pushed** my new/modified entry file(s) and the updated `_quarto.yml`.