Introduction to Literate Programming with Quarto

Elizabeth Waterfield

20/08/2025

Credit statement and licence

ttt-quarto-slides

Possible roles using the CRediT contribition system:

- **Conceptualization**: Ideas; formulation or evolution of overarching research goals and aims
- **Methodology**: Development or design of methodology; creation of models
- **Software** : Programming, software development; designing computer programs; implementation of the computer code and supporting algorithms; testing of existing code components
- **Validation**: Verification, whether as a part of the activity or separate, of the overall replication/ reproducibility of results/experiments and other research outputs
- **Formal analysis**: Application of statistical, mathematical, computational, or other formal techniques to analyze or synthesize study data
- **Investigation**: Conducting a research and investigation process, specifically performing the experiments, or data/evidence collection
- **Resources**: Provision of study materials, reagents, materials, patients, laboratory samples, animals, instrumentation, computing resources, or other analysis tools
- **Data Curation**: Management activities to annotate (produce metadata), scrub data and maintain research data (including software code, where it is necessary for interpreting the data itself) for initial use and later reuse
- **Writing - Original Draft**: Preparation, creation and/or presentation of the published work, specifically writing the initial draft (including substantive translation)

- **Writing - Review & Editing**: Preparation, creation and/or presentation of the published work by those from the original research group, specifically critical review, commentary or revision – including pre-or postpublication stages
- **Visualization**: Preparation, creation and/or presentation of the published work, specifically visualization/ data presentation
- **Supervision**: Oversight and leadership responsibility for the research activity planning and execution, including mentorship external to the core team
- **Project administration**: Management and coordination responsibility for the research activity planning and execution
- **Funding acquisition**: Acquisition of the financial support for the project leading to this publication

---

Prerequisites

> ❗ Prerequisites
>
> Before completing this submodule, please carefully read about the necessary prerequisites.

Prerequisite	Description	Link/Where to find it
Topic Name	Basic intro to X	Module + Submodule
Software Name	Configuring the environment	Download Link

Speaker Notes Speaker notes may act as a guiding script for delivering the presentation. These notes contain key information, such as phrases, explanations, or transitions, that can be said aloud to clarify concepts, emphasize important points, or maintain the flow of information.

Instructor Notes Instructor notes provide teaching support but are not intended to be spoken directly. These notes contain additional context, such as learning objectives, common issues, and pedagogical tips, to help the instructor adapt their teaching to the learner's needs as well as anticipate challenges.

Questions from previous submodule

Instructor Notes - Aim: clarify questions from the previous submodule and/or to discuss assignments. - Additional slides may need to be added depending on the nature of the homework assignments. - It is critical for the learning process to ensure that students are on the same page and have been able to achieve the learning goals of the previous workshop. - Not applicable if this set of slides corresponds to the first submodule of a new module.

Before we start: Survey time!

Take this survey to test your prior Quarto knowledge

Quarto Survey link

Instructor Notes - Aim: the pre-submodule survey serves to examine students' prior knowledge about the sumodule's topic. - Use free survey software such as or other survey software (Menti, particify, formR) to establish this. You can use the example survey, edit it or create your own.

Discussion of survey results

Instructor Notes - Aim: briefly examine the answers given to each question interactively with the group. - Use visuals from the survey to highlight specific answers. - Make it clear to the group that there will be a similar post-submodule survey to examine understanding and learning progress.

Where are we at?

Instructor Notes - Aim: Place the topic of the current submodule within a broader context. - Remind students what you are working towards and what the bigger picture is.

Quarto

Quarto makes it easy to analyze, share and reproduce.

It's a powerful tool for producing transparent, reproducible, and accessible work that can be freely shared, viewed, and reused by others.

Quarto Learning Goals

- Participants will **learn** how to create, edit, and render Quarto documents
- Participants will **understand** how to use key Quarto features (code chunks, YAML headers, citations, and output formatting)
- Participants will **gain confidence** in preparing and teaching Quarto effectively to students

Instructor Notes - Aim: Formulate specific, action-oriented goals learning goals which are measurable and observable in line with Bloom's taxonomy (Anderson et al., 2001; Bloom et al., 1956) - Emphasis is placed on the **verbs** of the learning goals- choose verbs that align with the skills you want to develop or assess.

- Examples:
  - Students will **describe** the process of photosynthesis or
  - Students will **construct** a diagram illustrating the process of photosynthesis

---

Key Terms and Definitions

What is Quarto? An open-source scientific and publishing system that combines text, code and media to create easily shareable documents such as websites, slides, reports, and much more.

What is Rendering? The process where Quarto runs the code, combines it with the text, and creates a final output.

A Quarto Markdown document is saved as a **.qmd file**

Instructor Notes - Aim: Introduce key terms and definitions that students will come across throughout the session. - This first part of the lesson is useful to establish an understanding of important vocabulary- it can be helpful to remind students of the meaning of these terms as they appear in the upcoming sections.

Key Terms and Definitions

Components of a .qmd file

- **YAML header**- the section at the top of the Quarto document that controls settings like the title, output format, and author.
- **code chunks**- sections of the document that contains code (from R or Python, for example) that are used for showing results such as tables, plots, or calculations
- **Quarto Markdown**- combines text, codes, and formatting to create the actual content of the document

---

Key Terms and Definitions

Components of a .qmd file

You can write Quarto documents in **Source mode** or **Visual mode** in RStudio.

- **Source mode** - you write directly in plain text/Markdown Syntax, allowing for more control and it's closer to raw code
- **Visual mode** - gives a WYSIWYM-style interface which easier for beginners, but sometimes hides syntax.
- This is part of the **authoring process**, and it allows you to format the text, add code, and build your document.

Instructor Notes Base yourself on conceptual change theory and examine existing concepts in relation to some key terms. Re-examine formation of new concepts at the end of the lesson.

Speaker Notes - WYSIWYM is an acronym for "What You Say is What You Mean" which means that whatever you see in your document will be what you see even when you save or render the document as PDF or as HTML.

- Quarto documents can be created and edited in either Source mode or Visual mode. Both modes work with the same file, but they provide slightly different experiences. In this lesson, most of our tasks will be done in Source mode, but you'll also have a chance to try Visual mode so you can compare how the same content looks and feels in each.

Covered in this Session

- Setting up a Quarto Document
- Authoring
- Code Chunks
- Adding Citations
- Publishing

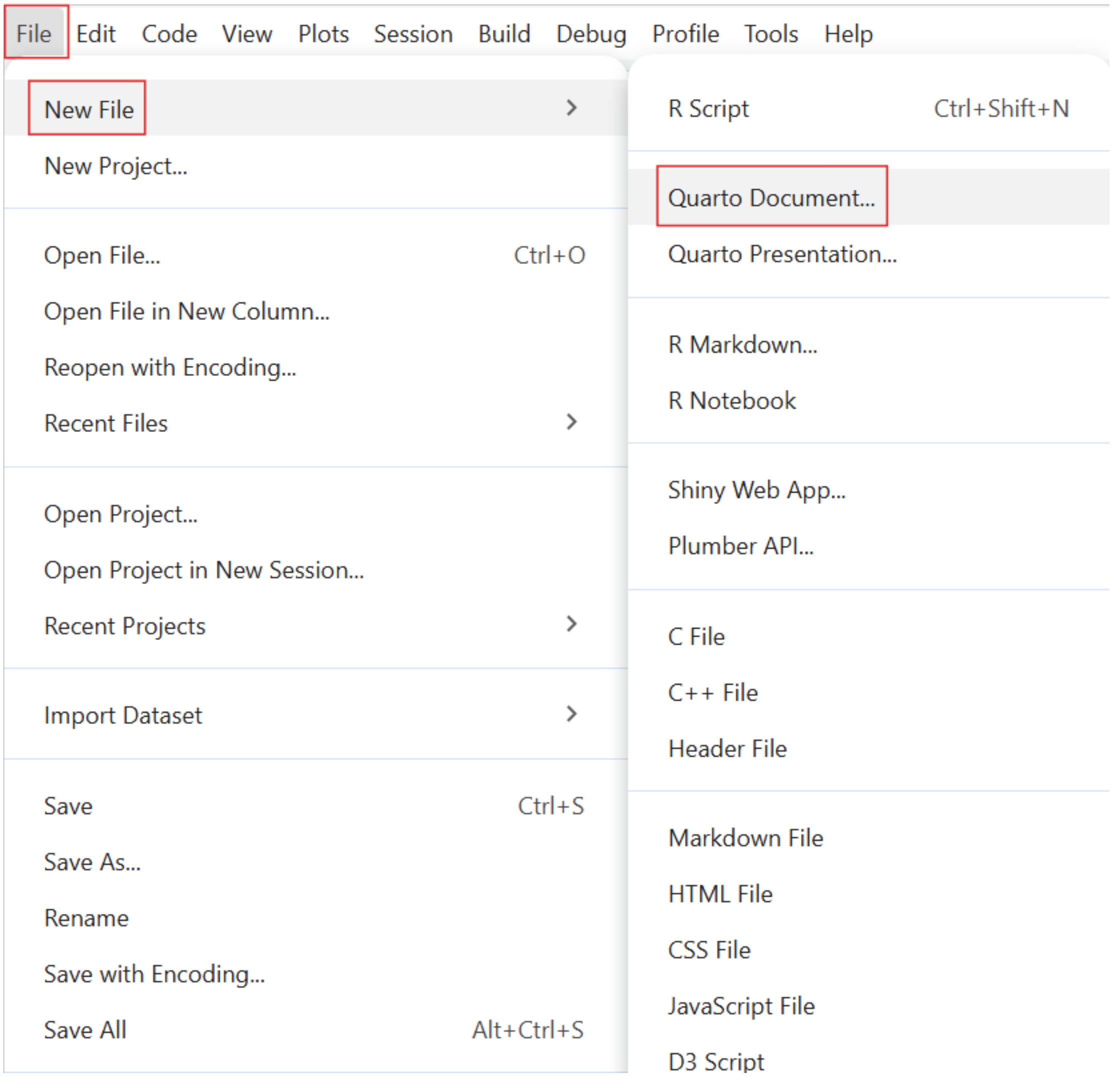
Instructor Notes - Aim: Core theoretical introduction of submodule topic. - Pair theoretical aspects with practical exercises and group discussions according to the Think-Pair-Share style and according to Cognitive Load Theory (Sweller, 1980).

- For a 90-minute lesson, the instructor should try to "lecture" for only 20 minutes, students should work in groups/pairs/on their own for at least 55 minutes of the lesson (+ a 15 minute break).

Practical exercises on topic - Aim: practical exercises for students to apply the new skills in practise. Each submodule topic will include corresponding Tasks- adopting a "learn by doing" approach. - Depending on the topic, the exercises should be in accordance with the learning objective(s). - It's useful to have exercises directly after a topic is taught to reinforce what was learnt.

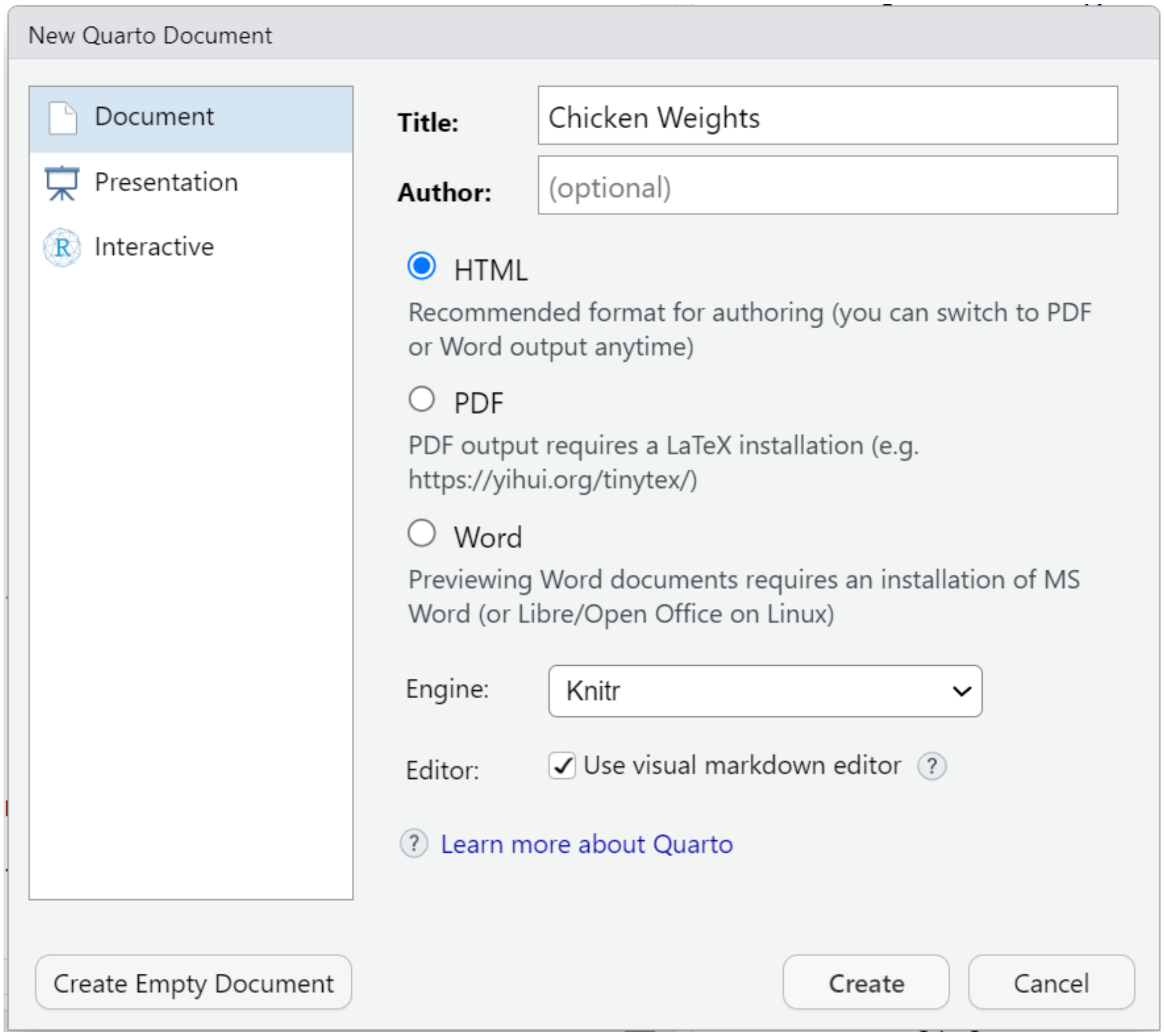
For students who advance faster: Prepare extra exercises.

Setting up a Quarto Document using RStudio



- Select **File**
- → Select **New File**
- → Select **Quarto Document**

Setting up a Quarto Document using RStudio



New Quarto Document

Title: Chicken Weights

Author: (optional)

● HTML

Recommended format for authoring (you can switch to PDF or Word output anytime)

○ PDF

PDF output requires a LaTeX installation (e.g. https://yihui.org/tinytex/)

○ Word

Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux)

Engine: Knitr

Editor: ☑ Use visual markdown editor

Learn more about Quarto

- The default format is html
- The output format can also be changed by editing the YAML header

Task 1

□ Follow the steps to create a new Quarto Document

Instructor Notes - Ensure students know where their Quarto project folder is located on their device. This will make it easier for them to find files in later tasks. Encourage them to save the document right after creating it, so they can intentionally choose the folder location (e.g., Desktop, a dedicated course folder, etc.).

Rendering

There are two ways to render in Quarto

- **Manual rendering**: having to click on the “Render” each time you want to see the output
- **Render on Save**: Quarto will automatically re-render the document each time you click “Save”

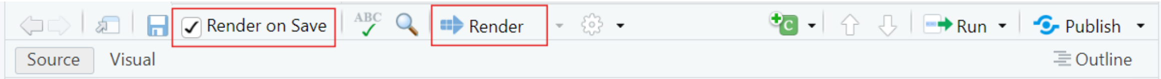


Figure 1: You can find both Manual Render and Render on Save at the top of your workspace

Speaker Notes - Quarto documents can be rendered either manually or automatically when you save. - Manual rendering gives you control meaning you decide when to update the document, which is helpful if you’re making lots of edits and don’t want constant re-renders. - Render on Save automatically updates your document every time you hit Save, which is great for quick feedback and seeing your changes instantly. In this lesson, you can try both to see which workflow feels more comfortable for you.

Authoring

This is the process of writing and structuring the Quarto document.

YAML Header + Markdown = Authoring

To practice authoring in Quarto, let’s start with setting the YAML header and adding text in **Source** mode.

Instructor Notes - Reminding students on the meanings of YAML Header and Markdown will be helpful especially to those with very little prior knowledge for working with Quarto.

Speaker Notes - The YAML header is the section at the very top, wrapped in three dashes (—). It holds the document’s settings, like the title, author, date, and output format. - Below that, we use Markdown to write the actual content which are things like headings, bullet points, bold or italic text.

Authoring

Task 2

- □ Copy & paste the following into the YAML Header of your document

```yaml
---
title: "ChickWeight Analysis"
author: "Your Name"
format:
  html:
    code-fold: false
    toc: true
---
```

> output format
>
> You can replace "html" to render the document to a different format.
> Here is a link to a list of the different output formats.

Instructor Note - There is a small clipboard icon at the top-right of the text box. Let students know they can simply click the icon to copy all the text in that specific text box. This icon will appear for every text box moving forward so you can remind them of it the next few times a task requires them to copy the content of a text box.

Speaker Note - Remember that the YAML Header goes at the top of the document in both Source and Visual Modes.

- In the YAML header, the format field tells Quarto what kind of output to create. Here we've set it to html, which means the document will render as a web page. You could choose other formats like PDF or Word, but for this lesson we'll stick with HTML so everyone has the same experience.

Authoring

Basic Markdown Text Formatting

- **Bold:** `**bold**` → **bold**
- *Italic:* `*italic*` → *italic*
- ~~Strikethrough:~~ `~~text~~` → ~~text~~
- `Inline code:` `` `code` `` → `code`

> markdown shortcuts
>
> Here is a link for the full list of markdown shortcuts for formatting!

Speaker Note - In Markdown, we can format text with very simple symbols. It works the same way in both Source and Visual mode with the only difference being how you see it while typing. - In Source mode you see the symbols, while in Visual mode it looks like regular bold or italic text right away.

Authoring

```
Have you ever been curious about what affects a chick's weight?
This document explores the ChickWeight dataset using R.
The goal is to compare chick weight across different diets and time points.
Key steps include:

- Loading the dataset
- Visualizing growth trends
- Summarizing results
```

Task 3

□ Copy the Markdown Text and paste it to your document
□ Bold the word *"ChickWeight"*
□ Italicize the phrase *"growth trends"*

Authoring

Highlight particularly important aspects with **Quarto callout boxes**

> ! Important with Title
>
> This is an example of a callout box to highlight particularly important information using `callout-important`

> Tip with Title
>
> This is an example of a callout box to give important tips using `callout-tip`

Note with Title

This is an example of a callout box to include an additional note using `callout-note`

Instructor Note - These are just examples of callout boxes. How to actually create one will be covered in the next slide/task.

Speaker Note - Callout boxes are a way to highlight important information in your document. They stand out visually, so readers immediately notice them. You can use them to emphasize tips, warnings, examples, or key takeaways.

Authoring

Adding a Callout Box

Here is the markdown text for inserting a **callout note** box.

```
::: callout-note
## Based on Real Data

The ChickWeight dataset in R is based on real experimental data.
:::
```

Task 4

- ☐ Copy & paste into your Quarto document to add this callout note box.

Code Chunks

Inserting Code

Two ways to insert code chunks:

- Manually type 3 back ticks “‘ then {r} to start a coding chunk, enter your code, then end it with 3 back ticks

OR

- Use the keyboard shortcut Ctrl + Alt + I (Windows/Linux) or Cmd + Option + I (Mac) to insert a code chunk

Instructor Notes - using {r} to start the Code Chunk indicates that the coding language that we're using is R. This is relevant for this lesson as students will be using R Studio and possession pof some background knowledge in R is assumed.

Speaker Notes - Code chunks are sections of a Quarto document where we can run code directly inside our file. They're marked by three backticks followed by the language name, like {r} or {python}. - These let us write and execute code, and then display the results, such as tables, plots, or calculations, right in the document. Code chunks make it possible to combine text and analysis in one place, so the document stays reproducible and dynamic.

Code Chunks

Inserting Code

```
summary(ChickWeight)
```

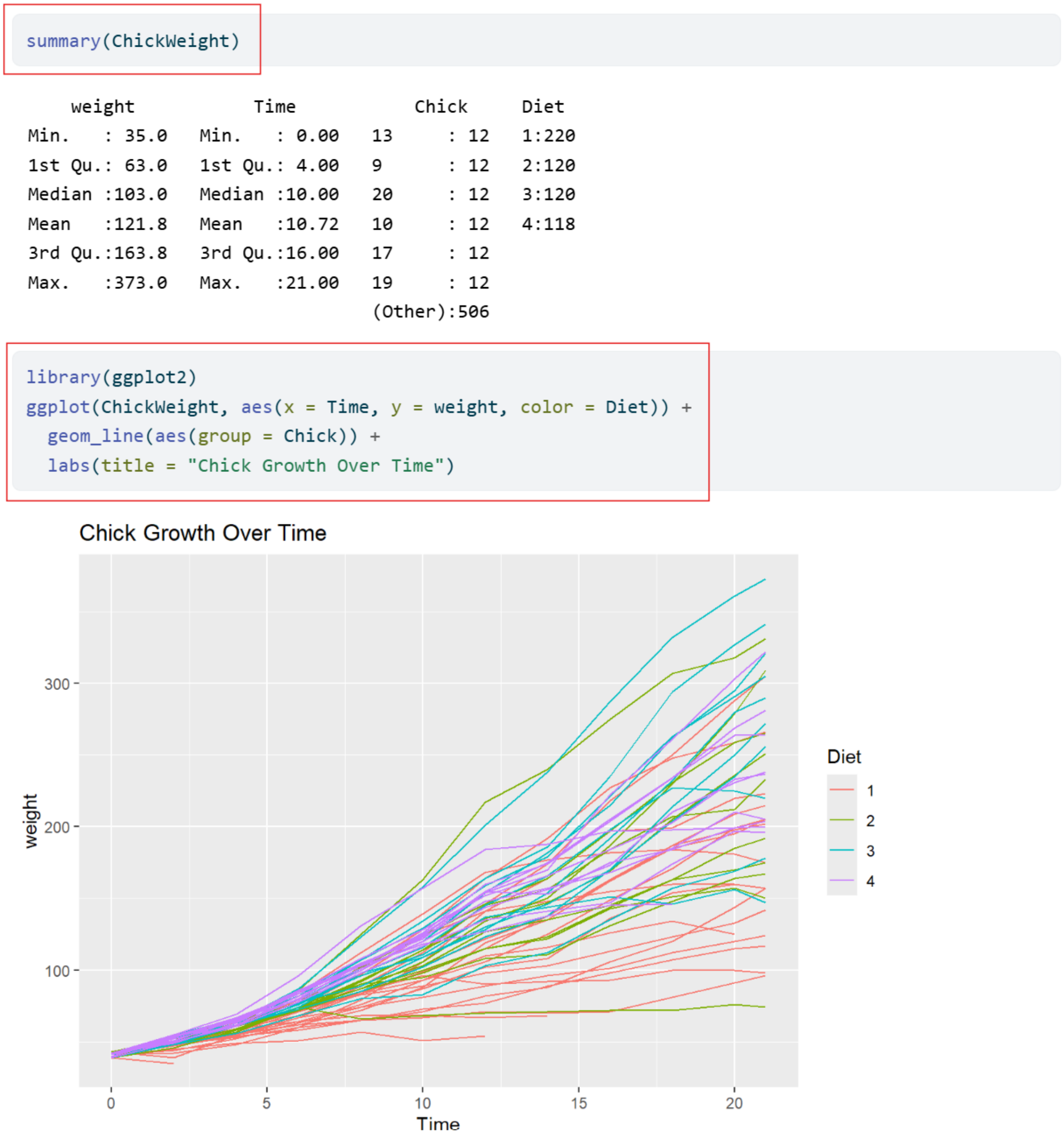
```
library(ggplot2)
ggplot(ChickWeight, aes(x = Time, y = weight, color = Diet)) +
  geom_line(aes(group = Chick)) +
  labs(title = "Chick Growth Over Time")
```

Task 5

- □ Insert the code in **two separate code chunks**
- □ Render the output and see what you get

Code Chunks

Adjust how the code is portrayed by editing the YAML header



```
summary(ChickWeight)
```

```
     weight           Time           Chick     Diet   
 Min.   : 35.0   Min.   : 0.00   13     : 12   1:220  
 1st Qu.: 63.0   1st Qu.: 4.00   9      : 12   2:120  
 Median :103.0   Median :10.00   20     : 12   3:120  
 Mean   :121.8   Mean   :10.72   10     : 12   4:118  
 3rd Qu.:163.8   3rd Qu.:16.00   17     : 12          
 Max.   :373.0   Max.   :21.00   19     : 12          
                                 (Other):506          
```

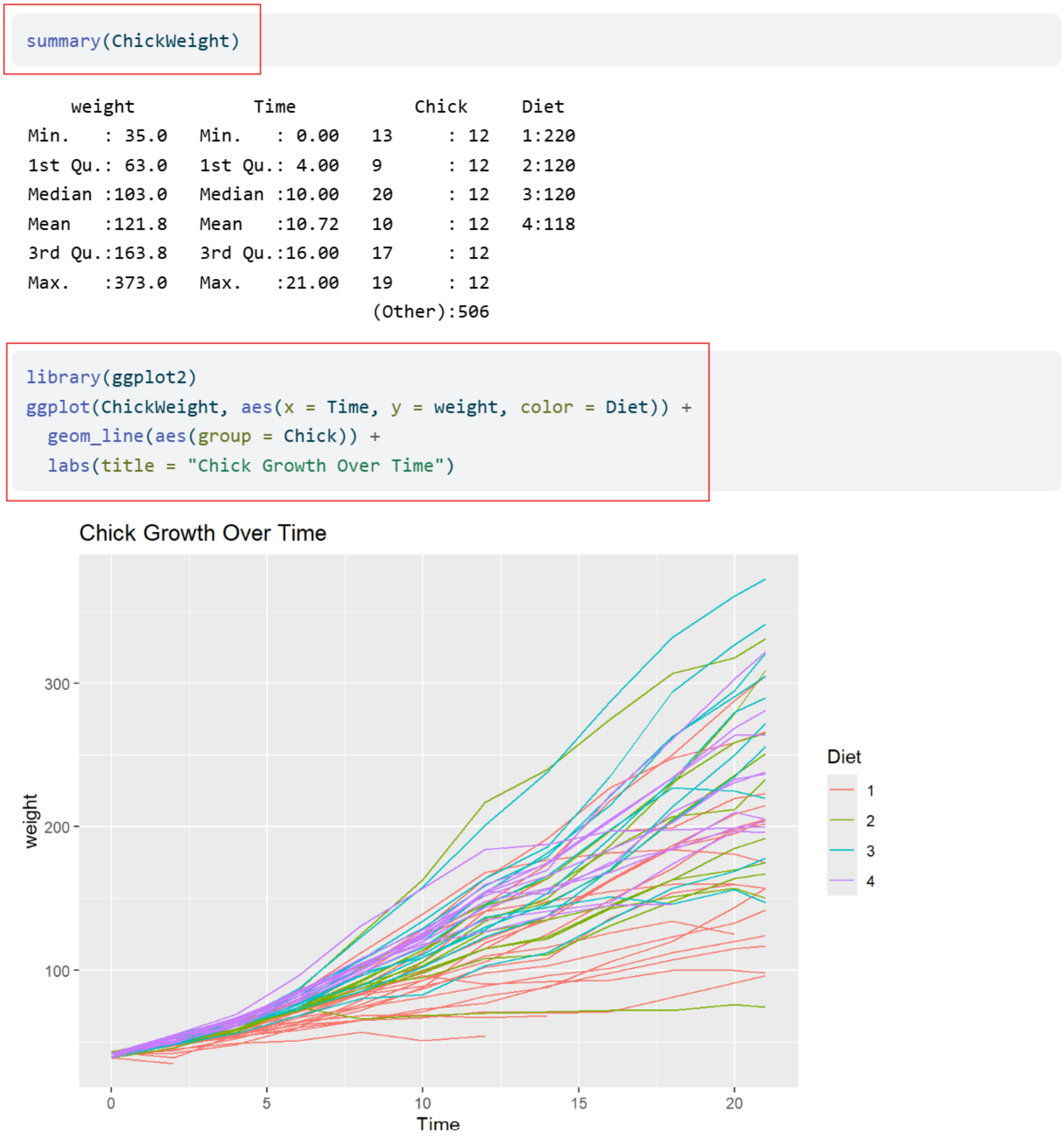
```
library(ggplot2)
ggplot(ChickWeight, aes(x = Time, y = weight, color = Diet)) +
  geom_line(aes(group = Chick)) +
  labs(title = "Chick Growth Over Time")
```

- `code-fold: false` - the code is visible and not collapsible (as seen here)
- `code-fold: true` - collapses the code so the reader can expand it

Speaker Notes - Quarto gives us several options for controlling how code appears in our documents. For example, we can choose whether to show or hide the code itself, whether readers can fold code open and closed, or whether only the results are shown. - Currently in our YAML, we have it set to `code-fold: false` which means the code is displayed as in the image. - These settings don’t change the analysis, but just change how it’s displayed. This flexibility is helpful because sometimes we want to highlight the process by showing the code, and other times we want readers to focus on the results. - In open research, showing the underlying code isn't just a technical choice, it's important for building trust and reproducibility.

Code Chunks

Adjust how the code is portrayed by editing the YAML header



```
summary(ChickWeight)
```

```
     weight           Time           Chick       Diet
 Min.   : 35.0   Min.   : 0.00   13     : 12   1:220
 1st Qu.: 63.0   1st Qu.: 4.00   9      : 12   2:120
 Median :103.0   Median :10.00   20     : 12   3:120
 Mean   :121.8   Mean   :10.72   10     : 12   4:118
 3rd Qu.:163.8   3rd Qu.:16.00   17     : 12
 Max.   :373.0   Max.   :21.00   19     : 12
                                 (Other):506
```

```
library(ggplot2)
ggplot(ChickWeight, aes(x = Time, y = weight, color = Diet)) +
  geom_line(aes(group = Chick)) +
  labs(title = "Chick Growth Over Time")
```

- `code-tools: true` - adds the functions “show code” at the top of the page and “copy” next to the chunks
- `echo: true` - both the code and the output is visible

Instructor Notes It’s easy to confuse `echo` and `code-fold` because they both deal with code visibility. Be clear on the difference

Speaker Notes - `echo: true` and `code-fold: false` both affect how code is shown, but in different ways. `echo` decides if the code prints at all, while `code-fold` lets readers toggle it open or closed, deciding whether the code can be hidden or expanded.

Code Chunks

Let’s edit the YAML header to make the code chunks **collapsible** and add **code tools**.

Task 6

☐ In the YAML, change `code-fold: false` to `code-fold: true`
☐ Add `code-tools: true`
☐ Render the document to see the changes

Pre-break survey

Instructor Note - Aim: This pre-break survey serves to examine students’ current understanding of key concepts of the submodule - Use free survey software such as or other survey software (Menti, particify, formR) to establish the following questions.

What’s the name of Quarto Markdown document?

a. .png file

b. .qmd file

c. .doc file

d. .mp3 file

What is the YAML header?

a. summarizes the document into a single line

b. this is where you store notes and reminders

c. it’s like the “settings” of the document

d. just the title of the document

Which of the following is used to format content (like paragraphs and bullet points)?

a. Code chunks

b. YAML header

c. Markdown text

d. All of the above

What are the 2 components of the Authoring process?

How to create a code chunk in a Quarto Document?

Break! 10 minutes

Post-break survey discussion

Instructor Note - Aim: To clarify concepts and aspects that are not yet understood - Highlight specific answers given during the survey

---

Additional Authoring Features

Quarto offers additional **authoring features** that make it more versatile and comprehensive. These include:

- adding links and hyperlinking text,
- embedding media,
- creating multi-column layouts, and more.

> ℹ commonly used authoring features
>
> Here is a link for commonly used markdown syntax for additional authoring features!

Speaker Note - Beyond writing plain text, Quarto also allows us to add features like hyperlinks, images or videos, and multi-column layouts. These improve how our document communicates. - Hyperlinks let us connect to sources or other sections, media like images and video help explain complex ideas more clearly, and multi-column layouts make the content easier to read and more visually appealing. - Using these features thoughtfully improves both the usability and the overall look of the document.

Inserting Links

link with title: [title] (link)

- [Quarto Website] (https://quarto.org/) → Quarto Website
- [Reveal.js Documentation] (https://revealjs.com/) → Reveal.js Documentation

link without title: < https:// link >

- < https://www.markdownguide.org/ > → https://www.markdownguide.org/
- < https://github.com/ > → https://github.com/

Inserting Images

- no caption: ! [ ] (path/image.png)

Speaker Note - A benefit of using Quarto is the ability to easily add media, such as images, to your documents

---

Inserting Images

- with caption: ! [caption] (path/image.png)

Figure 2: This is a caption about three yellow chicks in the grass.

Inserting Images

- with link: [! [caption] (path/image.png)] (link)

Figure 3: Click to see a study on broiler chicks

Inserting Images

```
![No chicks were harmed in the making of this lesson.](images/chickpic1.png)
```

Task 7

□ Click this link to download the image
□ Save the image to the main folder of the Quarto document files
□ Copy & paste the markdown text to insert an image with a caption (edit path if needed)

Instructor Note - Setting the path can be tricky depending on where the files are located - It's a good practice to save these files in the same main folder as your Quarto document files (ideally in a dedicated subfolder named "images" to keep paths organized and easy to manage). If students saved and placed their main folder earlier, then locating it should be easy. - in the Task above, the text says "images/chickpic1.png" assuming the picture is saved in a folder called "Images" in the main folder containing all the Quarto document files.

Creating Columns

You can change the layout of a section using **columns**.

- Start the column layout by writing ":::::: columns"
- Begin the first column with "::: column"
- Add content and end it with ":::"
- Begin second coloumn with "::: column"
- Add content and end it with ":::"
- End section with a final "::::::"

> ℹ list of page layout options
>
> Here is a link for page layout options to author how content looks on your document!

Speaker Note - Columns are useful when you want to organize content side by side instead of in a long vertical flow. For e.g., you might place an image in one column and an explanation in the other, or compare two code outputs directly next to each other like comparing two separate plots. - Columns help make information easier to scan and can improve the balance and readability of your slides or documents.

Creating Columns

Let's use columns to add bullet points next to an image.

Task 8

□ Click this link to download the image
□ Save the image to the main folder of the Quarto document files

---

Creating Columns

Task 8

```
:::::: columns
::: column

![](images/chickpic2.png)

:::

::: column

-   Monitor growth trends over time
-   Compare diets and weight gain

:::
::::::
```

- ☐ Copy & paste the markdown text to add an image with bullet points using columns (edit path if needed)

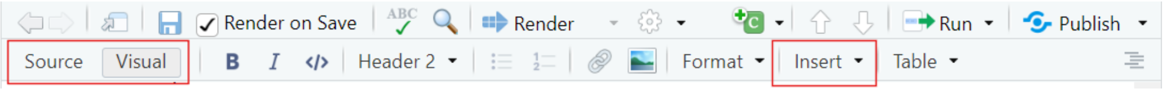
Adding Citations with Zotero

Zotero is a free reference management tool to collect, organize, cite, and share research sources.

You can use **Zotero** in RStudio to easily insert citations into your Quarto document

Speaker Note - Zotero is a reference manager that helps you collect and organize research articles, books, and other sources. When you connect Zotero to Quarto, you can easily insert citations while writing and then automatically generate a reference list at the end. This saves time and reduces errors compared to typing citations manually.

Adding Citations with Zotero

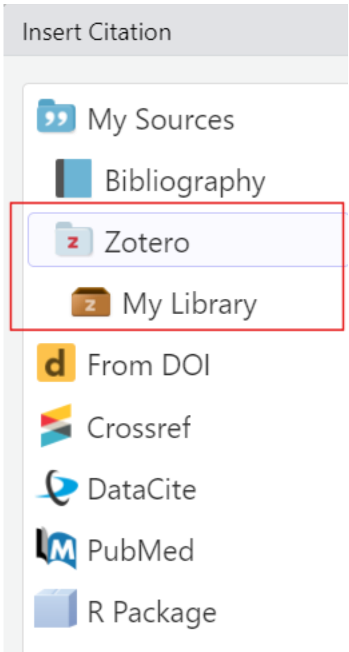


To insert a citation:

- switch to the "Visual Editor" mode,
- add `bibliography: references.bib` in the YAML header,
- press the "Insert" button in the toolbar and select "Citation"

A reference list or bibliography is automatically generated at the end of the document with all the citations used.

Adding Citations with Zotero



- You can select citations straight from your library and folders in Zotero
- Each citation is assigned a short citation key based on the author and year (e.g., pauwels2015).
- You can quickly cite it in your document by typing @ followed by the citation key.

Adding Citations with Zotero

Let’s practice adding a citation

Task 9

- □ add `bibliography: references.bib` to your YAML header
- □ follow this link to an article on diets and broiler chicks
- □ add the citation to your Zotero Library

```
Diet affects chick body weight. In this study, low-energy feed made fast-growing chicks ligh
```

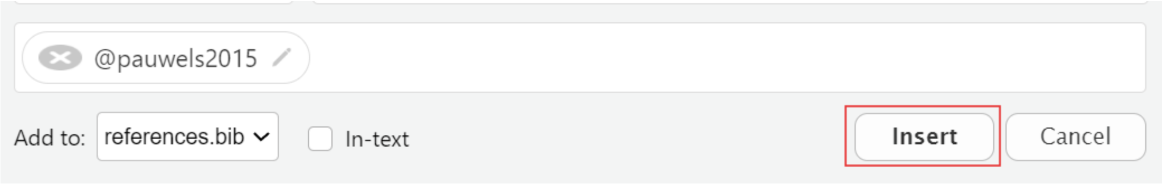
- □ copy the text above and place the cursor at the end of the sentence

Instructor Note - There are several ways to add a citation to your Zotero library but an easy way is using the ‘magic wand’ (adding the item by its identifier). Here, you can just copy the DOI of the study, hit Enter, and it will be added to Zotero. The DOIs of the articles in these tasks will be clearly visible so this method is recommended.

Adding Citations with Zotero

Task 9

- □ in **Visual Mode**, click “Insert” and find the article in the Zotero Library
- □ click “Insert” to add the citation



@pauwels2015

Add to: references.bib

Figure 4: When you find the article in the Zotero Library folder here, select it and then click Insert

Instructor Note - It is possible to add citations in Source mode, but it is easier and more efficient to do it in Visual mdoe. This is also a good opportunity for students to see all the work they’ve done so far in the Quarto document now in Visual mode.

Adding Citations with Zotero

Let’s add another citation to practice adding **in-text** citations

Task 10

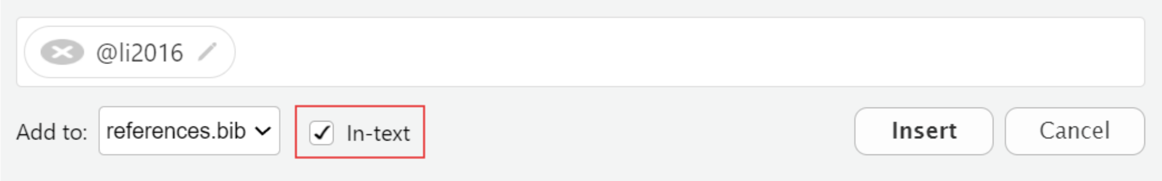
```
found that broiler chicks fed a higher nutrient-density diet gained more weight and did so
```

- □ copy the text above and place the cursor at the beginning of the sentence
- □ follow this link to another article on diets and broiler chicks
- □ add the citation to your Zotero Library

Adding Citations with Zotero

Task 10

- □ click “Insert” and find the article in the Zotero Library
- □ check the box next to “In-text”



@li2016

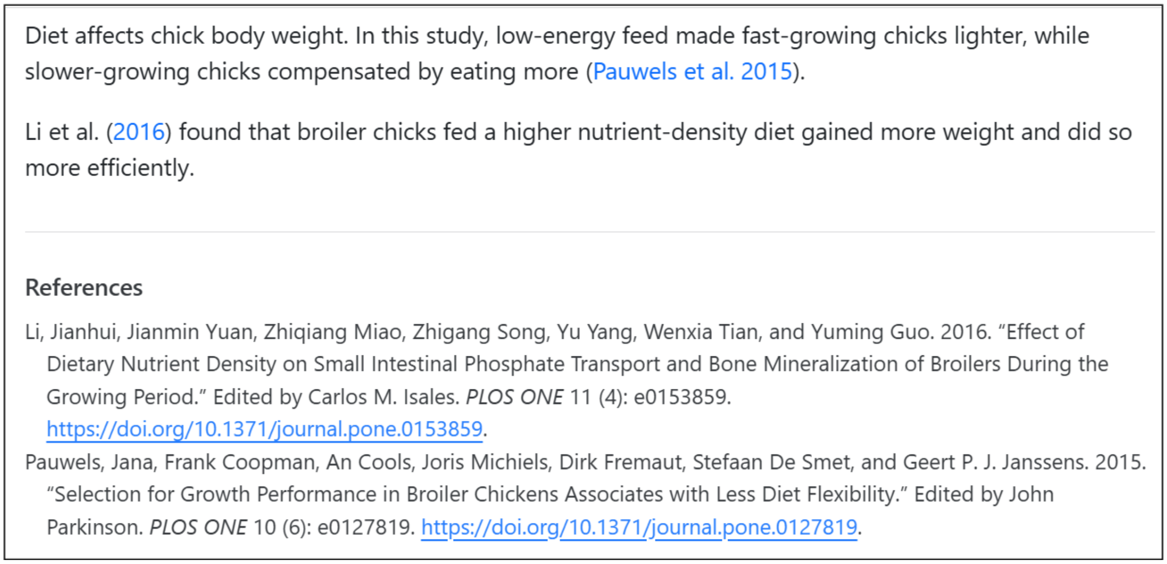
Add to: references.bib ☑ In-text

- □ click “Insert” to add the citation

Adding Citations with Zotero

Now you should have

- a citation,
- an in-text citation, and
- a reference list (bibliography).



Diet affects chick body weight. In this study, low-energy feed made fast-growing chicks lighter, while slower-growing chicks compensated by eating more (Pauwels et al. 2015).

Li et al. (2016) found that broiler chicks fed a higher nutrient-density diet gained more weight and did so more efficiently.

References

Li, Jianhui, Jianmin Yuan, Zhiqiang Miao, Zhigang Song, Yu Yang, Wenxia Tian, and Yuming Guo. 2016. "Effect of Dietary Nutrient Density on Small Intestinal Phosphate Transport and Bone Mineralization of Broilers During the Growing Period." Edited by Carlos M. Isales. *PLOS ONE* 11 (4): e0153859. https://doi.org/10.1371/journal.pone.0153859.

Pauwels, Jana, Frank Coopman, An Cools, Joris Michiels, Dirk Fremaut, Stefaan De Smet, and Geert P. J. Janssens. 2015. "Selection for Growth Performance in Broiler Chickens Associates with Less Diet Flexibility." Edited by John Parkinson. *PLOS ONE* 10 (6): e0127819. https://doi.org/10.1371/journal.pone.0127819.

> changing the citation style
>
> You can change the citation style by using csl files directly from the Zotero website. For e.g., putting "`csl: https://www.zotero.org/styles/apa`" in your YAML changes the citation style to APA.

Publishing

What is Publishing with Quarto? It's the process of sharing the rendered documents or projects online so it can become accessible to others.

You can publish to:

- GitHub Pages
- Quarto Pub
- personal website

> list of publishing services
>
> Here is a link for a list of publishing services and more information!

Speaker Note - Publishing in Quarto is all about making your work accessible to others. Once you've created and rendered your document, you can share it online so that collaborators, students, or the wider community can view it. Quarto supports multiple publishing options, including GitHub Pages, Quarto Pub, or even your own personal website.

Instructor Note - The key takeaway here is that publishing turns your local work into something that can be accessed from anywhere.

Publishing to GitHub

A great option to share your document is publishing to **GitHub**.

- **GitHub** is a platform for hosting and sharing code and projects online
- An advantage of GitHub is it allows for **version control**- the ability to track and manage changes over time
- A free and widely used way to publish your Quarto document to GitHub is by using **GitHub Pages**

Publishing to GitHub

GitHub Pages enables you to publish content based on source code managed within a GitHub repository.

3 methods to publish a Quarto document to GitHub Pages:

1. Render to the `docs` directory and checking it into your repository
2. Use the `quarto publish` command
3. Use GitHub Actions to auto-render and publish whenever you push changes

---

Publishing to GitHub

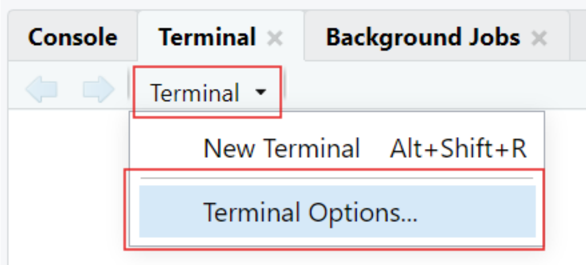
Let's publish our document using the first method:

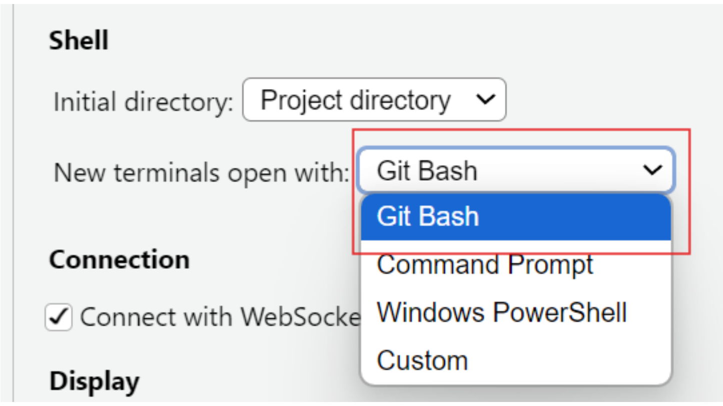
Render to `docs`

Publishing to GitHub

Work with **Git Bash** directly from the **Terminal** in RStudio.

- click Terminal → Terminal Options and set it to open with Git Bash





Initial directory: Project directory

New terminals open with: Git Bash

✓ Connect with WebSocke

- then, click Terminal → New Terminal to begin working with Git Bash

Instructor Note - The last step (to open a new Terminal) is important because the current Terminal does not just automatically change to Git Bash.

Publishing to GitHub

Task 11

□ in Quarto Document, edit YAML and add:

```
project:
  type: website
  output-dir: docs
```

□ set Terminal to open with Git Bash and open new Terminal
□ change directory to the path of the main folder with the Quarto files using `cd`

Instructor Note - Reminder that copying & pasting with keyboard shortcuts do not work the same in Git Bash. Right-clicking and selecting Paste is an easy option. - Setting the directory can be tricky. - Backslashes (`\`) need to be changes to forward slashes (`/`). - If you're pasting the exact path of the folder, wrap it in quotation marks (`"..."`)

Publishing to GitHub

Task 11

```
echo "project:
  type: website
  output-dir: docs" > _quarto.yml
```

- ☐ make a simple `_quarto.yml` file by copying & pasting the above in the Terminal

```
touch .nojekyll
```

- ☐ copy & paste the above into the Terminal to add a `.nojekyll` file

> ℹ .nojekyll
>
> Adding a `.nojekyll` file to the root of your repo tells GitHub Pages not to do additional processing of your published site using Jekyll

---

Publishing to GitHub

Task 11

```
quarto render
```

- ☐ copy & paste the above into Terminal to generate HTML in the `docs/` folder

```
git init
git branch -M main
```

- ☐ copy & paste the above into Terminal to initialize Git and rename "master" to "main"

Instructor Note - You will see a warning about `references.bib not found in resource path` which usually happens because Quarto looks for the file in the project's root or resource path, but Git Bash is case-sensitive and strict with paths. - So, you will see in your completed Git Pages web page that the citations look strange but we will leave it like that for now. - Extra steps are needed to correct this that will be covered in the Advanced Intro to Quarto Activity PDF.

Publishing to GitHub

Task 11

- □ in GitHub, create a new repo
- □ in Terminal, add remote by using "`git remote add origin`" + the repo's HTTPS URL (e.g., "`git remote add origin https://github.com/yourname/chickweight-exercise.git`")

```
quarto render
git add docs
git commit -m "Publish site to docs/"
git push -u origin main
```

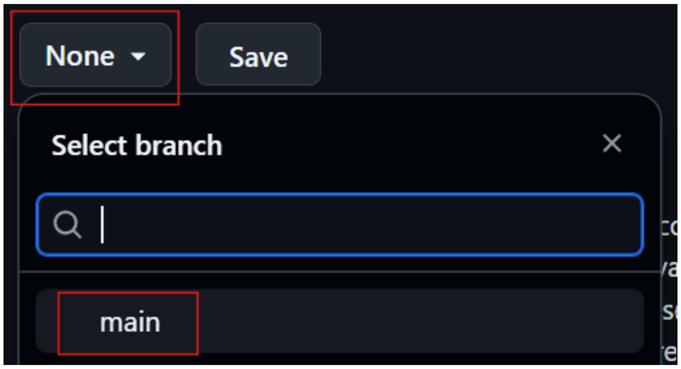
- □ copy & paste the above into Terminal to render your site and push it to GitHub

Instructor Note - For this to work, the repo created must be set to Public

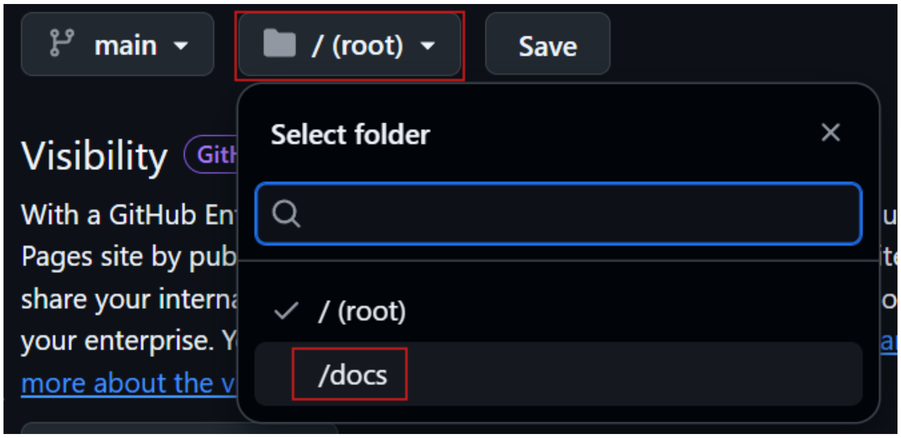
---

Publishing to GitHub

- □ In your repo on GitHub, go to Settings
- □ Click "Pages" to configure GitHub Pages



☐ change "none" to "main"



☐ change "/root" to "/docs" and click Save

After 1-2 minutes, you will see that your site is **live**.

Instructor Note - Reminder that the citations will not be complete because configuring to include the bibliography was not done. That's an additional step.

Recap

Relevance and Implications

Let's discuss how useful **Quarto** can be for **open research**.

Consider the following questions:

- How can using Quarto improve transparency and reproducibility in your work?
- In what ways does Quarto make sharing code and data with collaborators easier?
- Do you see advantages in using Quarto compared to traditional tools like Word or PowerPoint?
- What challenges might you face in adopting Quarto in your current

Instructor Note - Aim: To work out the relevance of the topic to your students. - In an interactive setting, discuss how the new skills could be applied in practise with specific examples. - Examine downfalls and practical obstacles.

What is the take-home message?

Let's wrap up by identifying the **key takeaway** from today's session together.

Ask yourself: **"If you had to explain to a colleague in one sentence why learning Quarto matters for research and teaching, what would you say?"**

Instructor Note - Aim: End lesson on clear take-home message that are interactively compiled by students. - Encourage students to share one sentence take-home messages. You can collect them verbally or in a shared doc/board. - Reinforce that Quarto is not just a tool, but a mindset shift toward openness and collaboration.

Assignment

- **Aim**: Explain the homework assignment and the rationale behind the homework.
- Examine whether/how it will be assessed
- Mention scoring rubrics, if applicable
- Design a peer-review system for assignments to place students in role of reviewer and author

---

To conclude: Survey time!

Instructor Note

- Aim: This post-submodule survey serves to examine students' current knowledge about the sumodule's topic.
- Use free survey software such as or other survey software (particify, formR) to establish the following questions (these are examples that can be adapted to suit your needs):

What is your level of familiarity with Quarto before this lesson? a) I've never used it b) I've heard of it but never tried it c) I've experimented a little d) I use it occasionally e) I use it frequently

Which of the following best describes Quarto? a) A programming language b) A publishing system for reproducible documents c) A code editor d) A type of version control system

Which components are combined in the authoring process of Quarto documents? a) Only YAML b) Only Markdown c) Neither d) YAML and Markdown together

Which of the following modes can be used to create a Quarto document? a) Source mode b) Visual mode c) Both d) Neither

On a scale of 1 to 5, how comfortable are you switching between Source and Visual modes?

Which of these are examples of Markdown text formatting? (Select all that apply) a) Bold text b) Italic text c) Adding code chunks d) Adding hyperlinks

Why should you use text formatting in Quarto documents? a) For aesthetics only b) To improve clarity and emphasis c) To make the file larger d) To change the coding language

When would you consider using multi-column layouts? a) To shorten the document b) To make information more visually organized c) To add footnotes only d) To increase file size

What is the purpose of code chunks in Quarto documents? a) To include executable code and results within the document b) To store references c) To format text d) To hide metadata

Which YAML options relate to how code chunks are displayed? a) echo: true b) code-fold: false c) title: "My Document" d) format: html

Why is including code in research documents important? a) To make the document more difficult to read b) To ensure only the author can reproduce results c) To replace data collection d) To make research reproducible and transparent

Which of the following are common publishing options for Quarto projects? (Select all that apply) a) GitHub Pages b) Quarto Pub c) Social media platforms d) Personal websites

What is the benefit of publishing your Quarto documents? a) They make your computer run faster b) They allow offline editing only c) They become accessible to others for sharing and collaboration d) They automatically write the code for you

---

Discussion of survey results

Instructor Note

- Aim: Briefly examine the answers given to each question interactively with the group.
- Compare and highlight specific differences in answers between pre- and post-survey answers

---

References

- Provide literature you refer to throughout this lesson.

Thanks!

See you next class :)

Pedagogical add-on tools for instructors

- This section is dedicated to ideas on how to incorporate pedagogical tools into teaching for this specific submodule topic. This could mean:
  - Information about the scientific evidence on the theory of the pedagogical add-on tool and the evidence for its efficacy.
  - Discussion/reflection on how tools can be incorporated into the teaching for this particular content.
  - Extra exercises for faster students.

Additional literature for instructors

- References for content
- References for pedagogical add-on tools

- Other resources (videos etc.)

Attribution and license details

- This slide should contain information about the license and attribution details of this current set of slides.
- The default for the created materials is CC-BY-SA 4.0
  - = Creative Commons license that allows others to **share, adapt, and build upon** the original work
  - **only** if they attribute the creator and also share their new work under the **same terms**
  - allows for both **commercial and non-commercial** use of the licensed material
- Components of attributions:
  - `Title`
  - `Author`
  - `Source`
  - `License`

Example attribution (for previous slide)

"Tutorial template for student track" by Sarah von Grebmer is licensed under CC-BY-SA 4.0.
