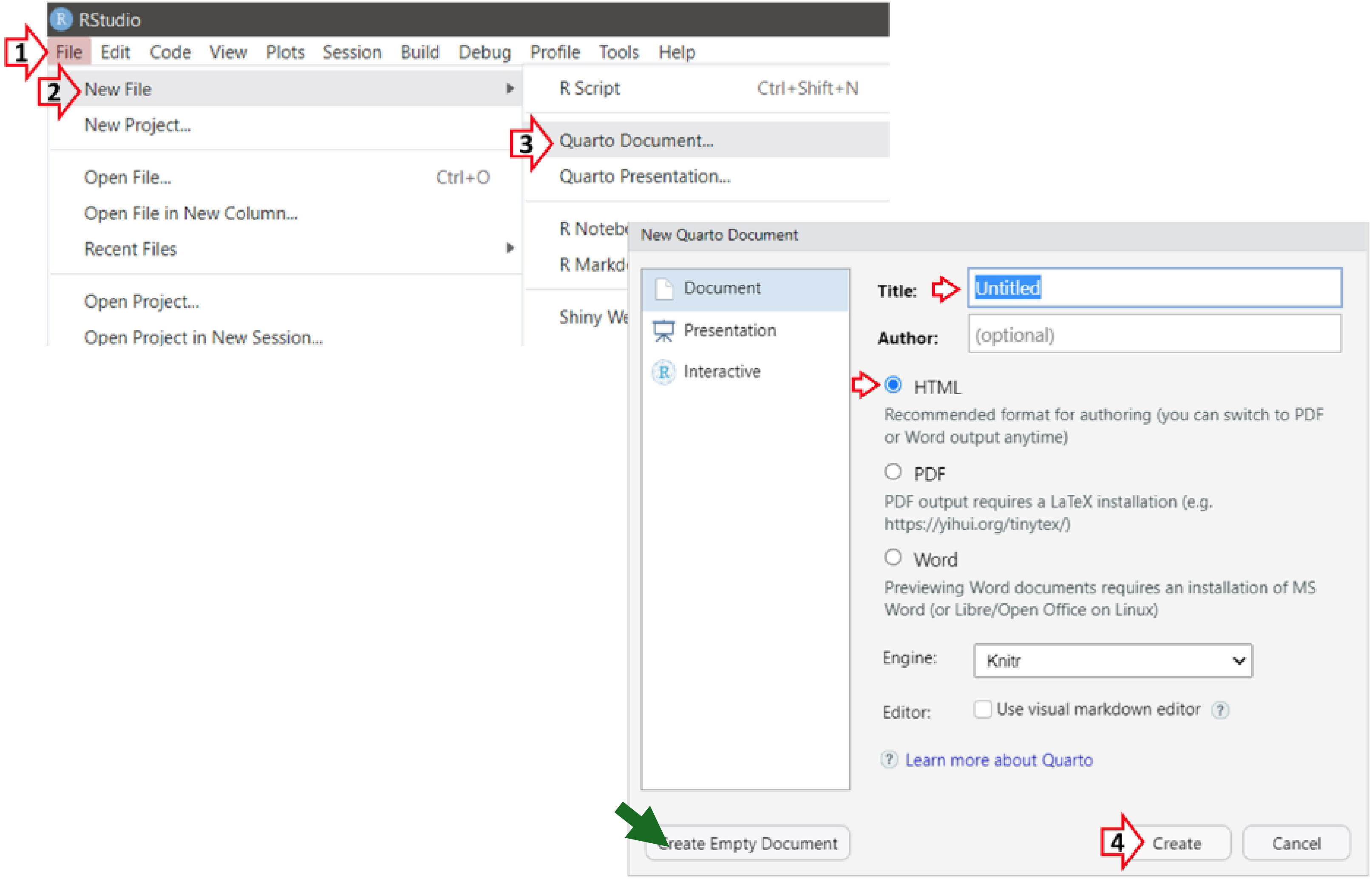
Getting started with Quarto document

1 - New Quarto Document:



Title: Untitled

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/)

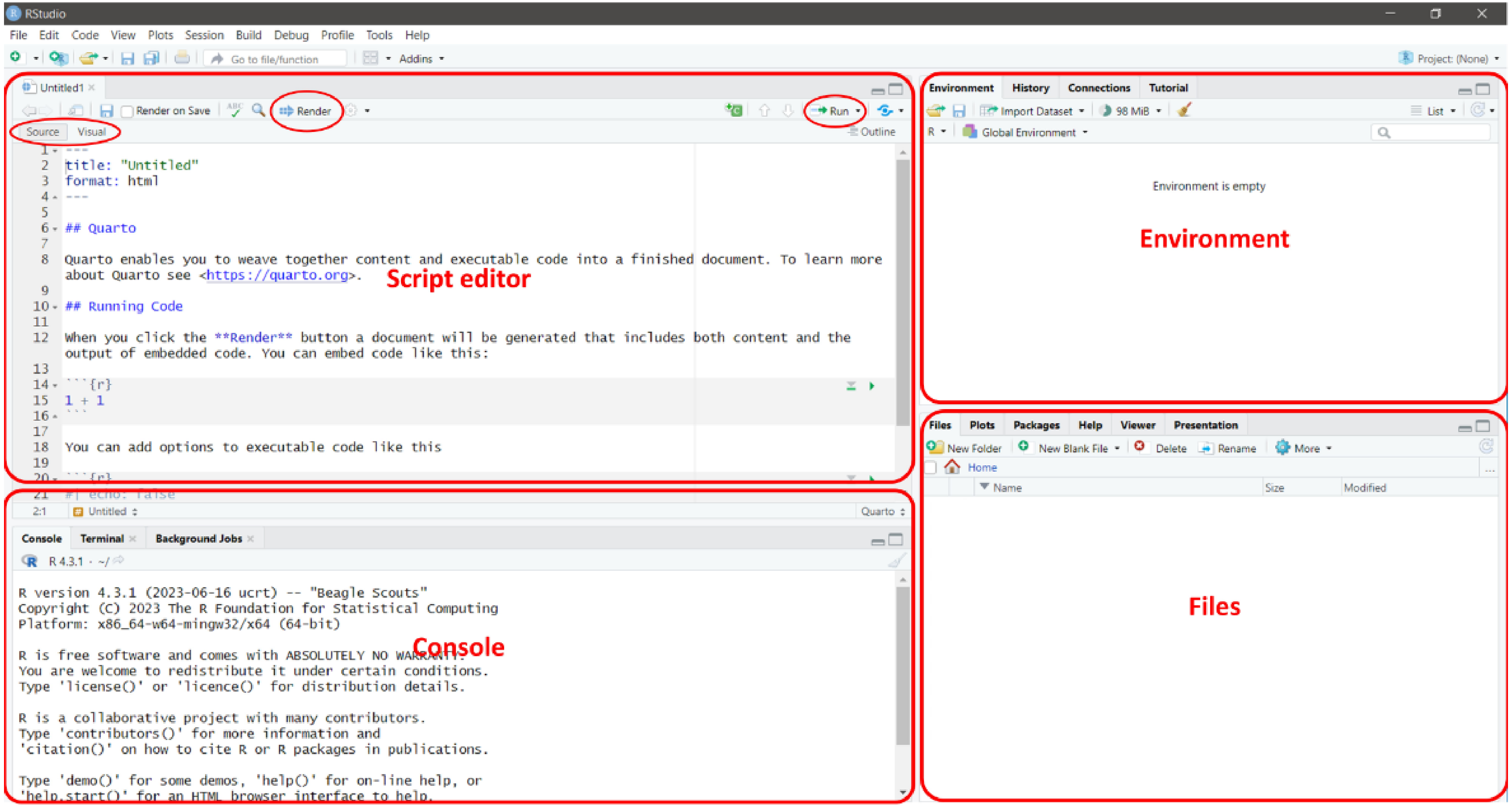
○ Word
Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux)

Engine: Knitr

Editor: □ Use visual markdown editor

1 - New Quarto Document:

Create a file with temporary content to start



~~~
---
title: "Untitled"
format: html
---

## 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:

```{r}
1 + 1
~~~

You can add options to executable code like this

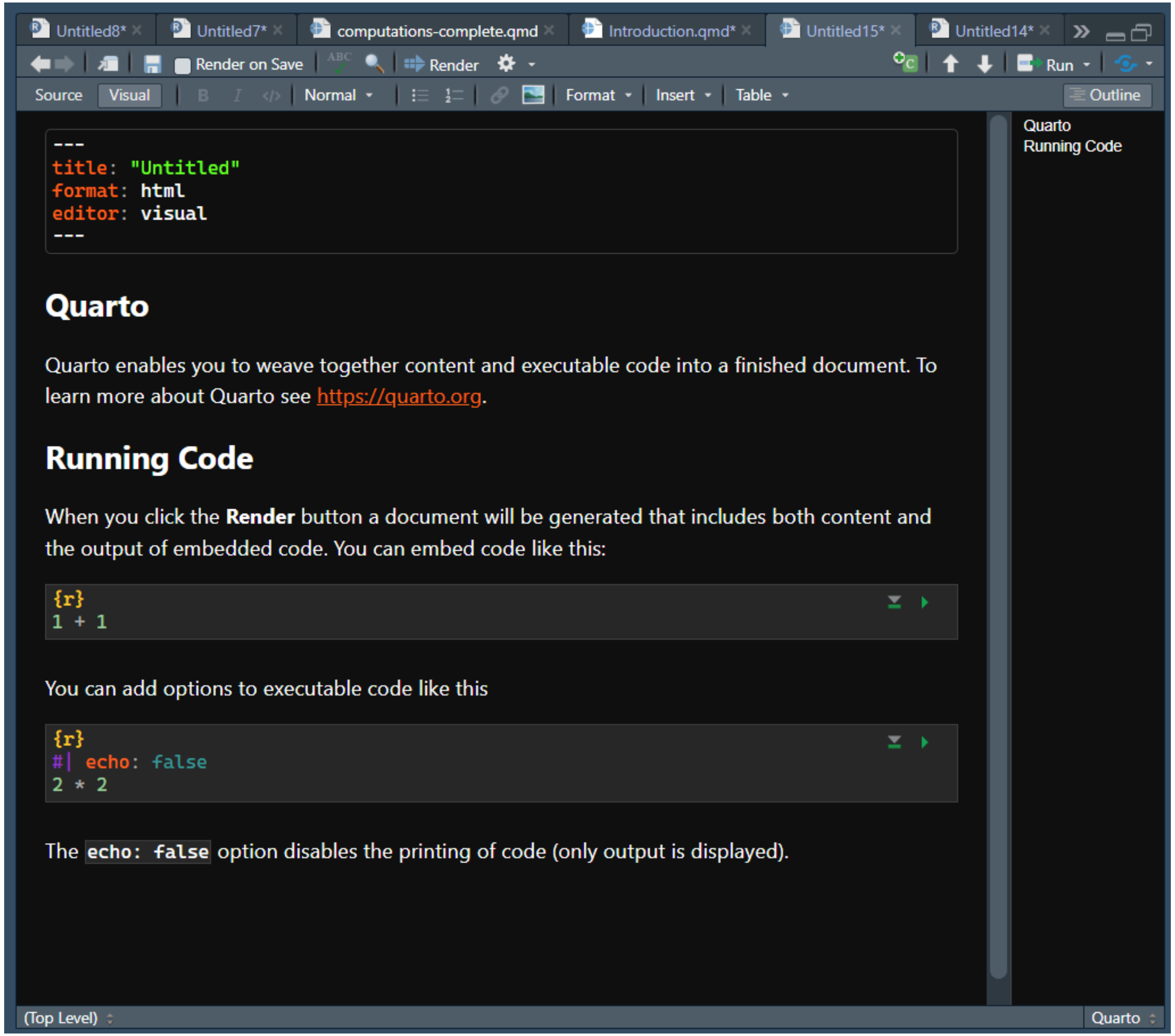
```{r}
#| echo: false
```

```
R version 4.3.1 (2023-06-16 ucrt) -- "Beagle Scouts"
Copyright (C) 2023 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
```

… save the document

… a start for your future project

< New document



```
---
title: "Untitled"
format: html
editor: visual
---
```

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:

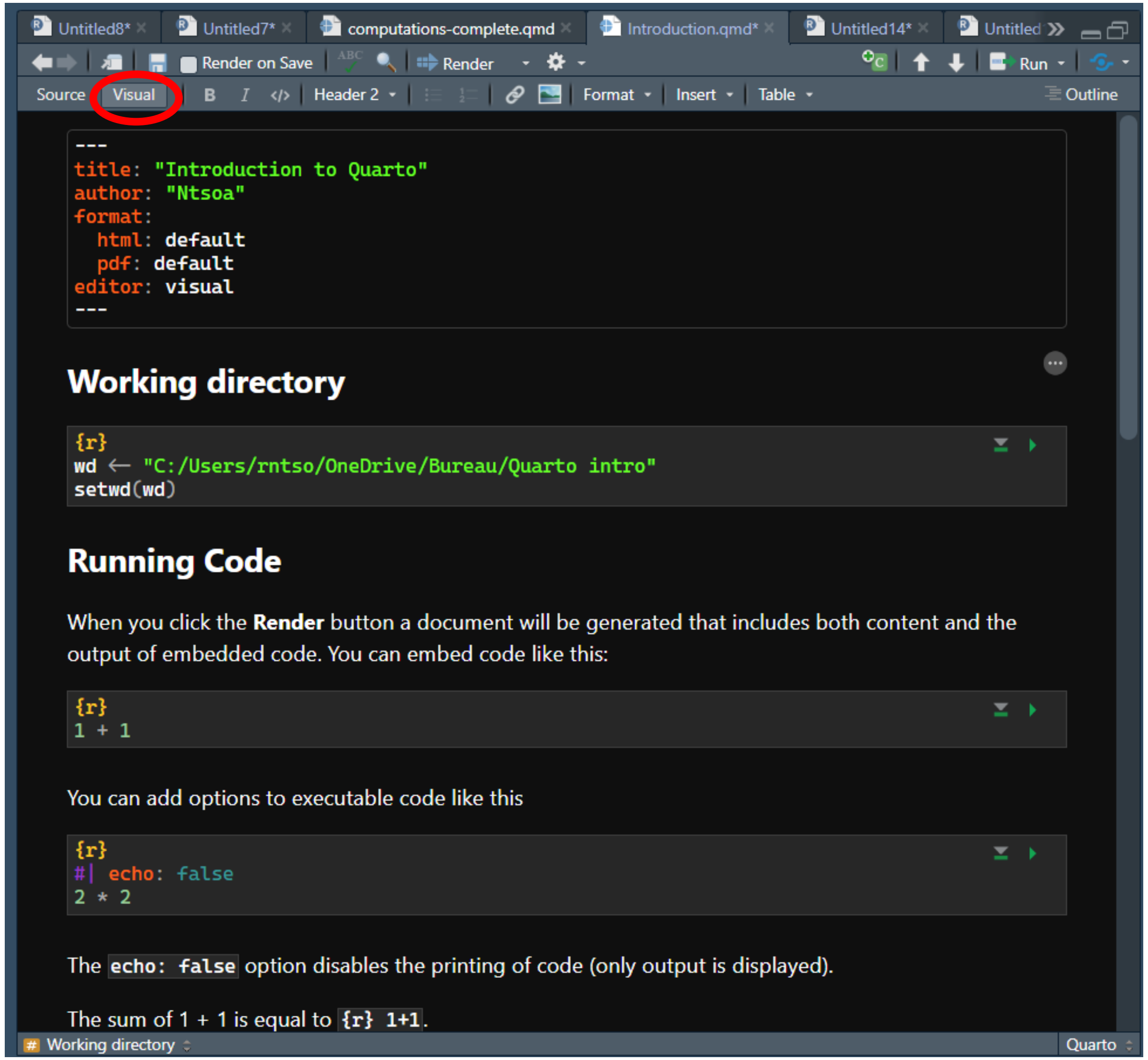
```
{r}
1 + 1
```

You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

< Visual



```
---
title: "Introduction to Quarto"
author: "Ntsoa"
format:
  html: default
  pdf: default
editor: visual
---
```

Working directory

```
{r}
wd <- "C:/Users/rntso/OneDrive/Bureau/Quarto intro"
setwd(wd)
```

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:

```
{r}
1 + 1
```

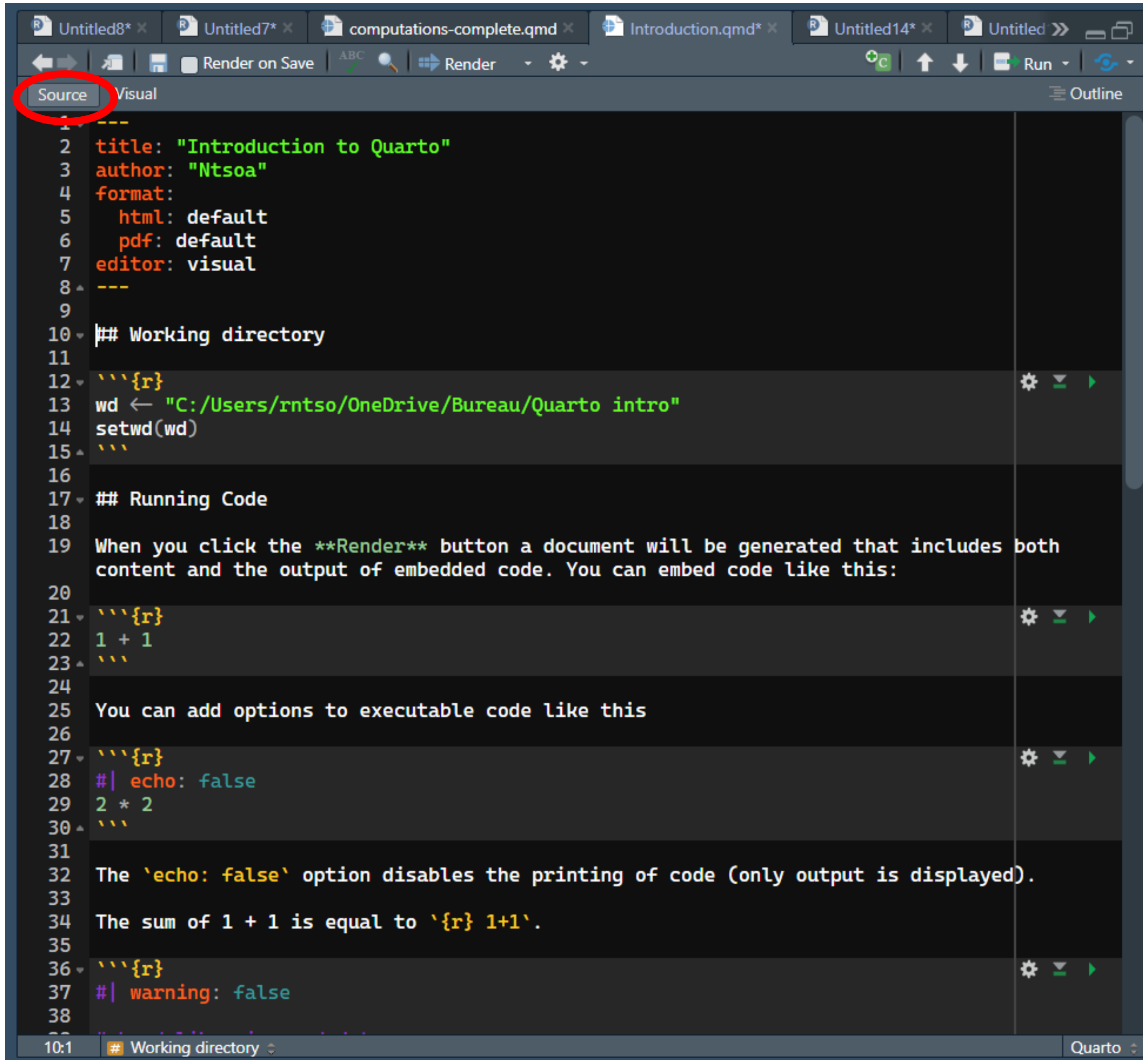
You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

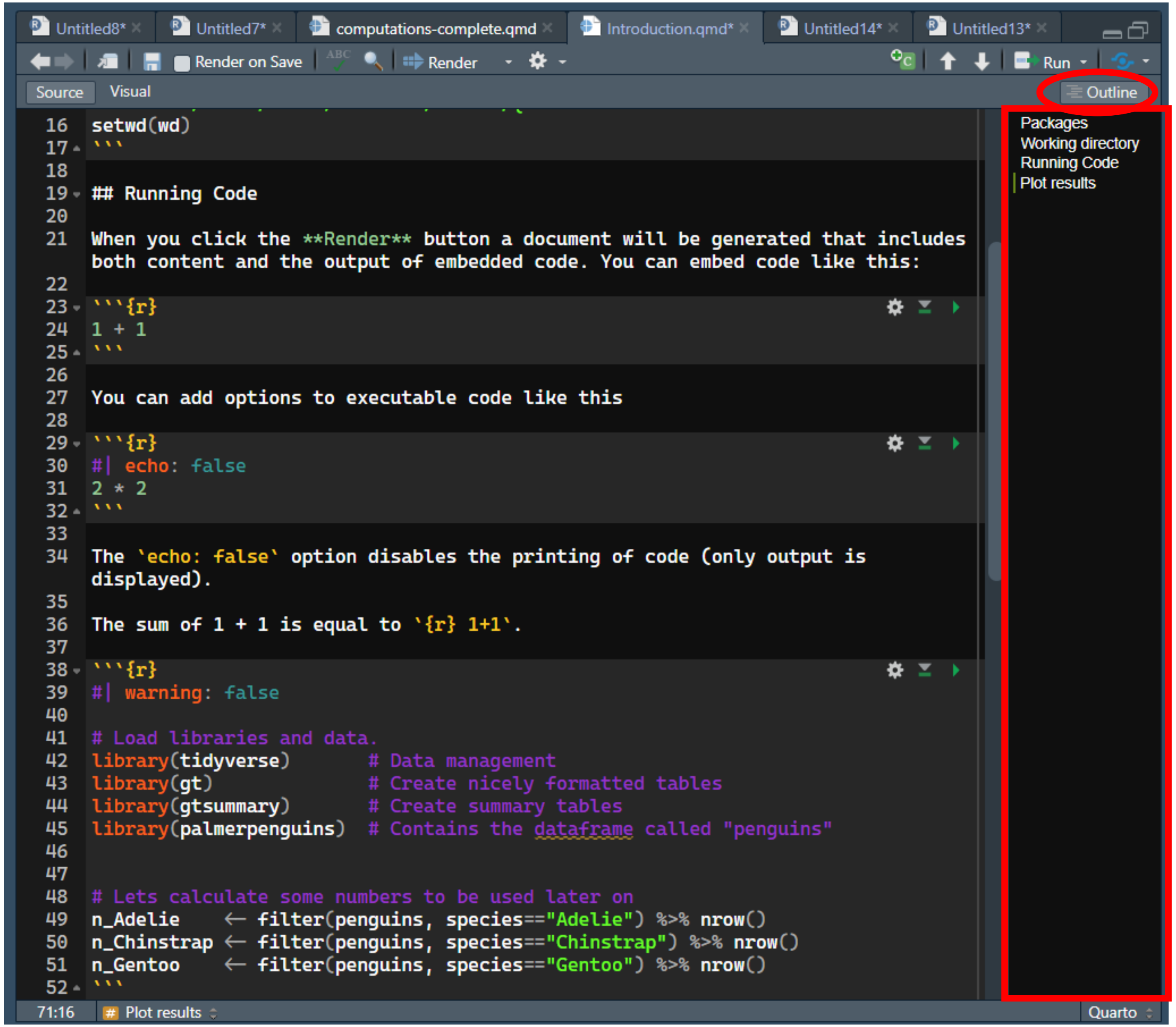
The sum of 1 + 1 is equal to `{r} 1+1`.

< Source



~~~
---
title: "Introduction to Quarto"
author: "Ntsoa"
format:
  html: default
  pdf: default
editor: visual
---

## Working directory

```{r}
wd <- "C:/Users/rntso/OneDrive/Bureau/Quarto intro"
setwd(wd)
```

## 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:

```{r}
1 + 1
```

You can add options to executable code like this

```{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

The sum of 1 + 1 is equal to `{r} 1+1`.

```{r}
#| warning: false

~~~

< Outline



```
setwd(wd)
```

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:

```{r}
1 + 1
```

You can add options to executable code like this

```{r}
#| echo: false
2 * 2
```

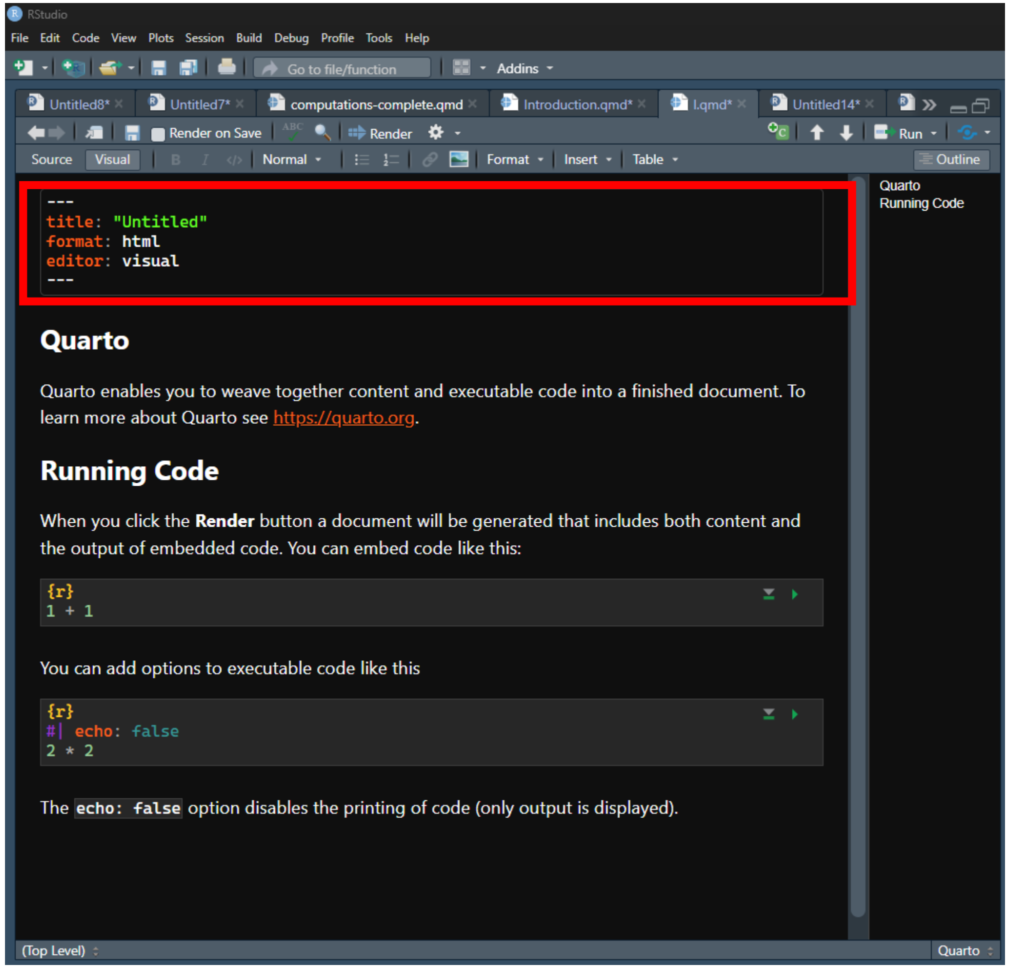
The `echo: false` option disables the printing of code (only output is displayed).

The sum of 1 + 1 is equal to `{r} 1+1`.

```{r}
#| warning: false

# Load libraries and data.
library(tidyverse)       # Data management
library(gt)              # Create nicely formatted tables
library(gtsummary)       # Create summary tables
library(palmerpenguins)  # Contains the dataframe called "penguins"


# Lets calculate some numbers to be used later on
n_Adelie    ← filter(penguins, species=="Adelie") %>% nrow()
n_Chinstrap ← filter(penguins, species=="Chinstrap") %>% nrow()
n_Gentoo    ← filter(penguins, species=="Gentoo") %>% nrow()
```

... stay in *Visual*

2 - Output Settings


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Untitled8* Untitled7* computations-complete.qmd Introduction.qmd* l.qmd* Untitled14*
Render on Save
Render
Run
Source Visual
Normal
Format Insert Table
Outline
Quarto
Running Code
(Top Level)
Quarto

```
---
title: "Untitled"
format: html
editor: visual
---
```

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:

```
{r}
1 + 1
```

You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

Syntax (Input)

```
---
title: Introduction to Quarto
subtitle: My subtitle
author: Klajdi Puka
date: last-modified
format:
  html:
    self-contained: true
execute:
  echo: true
  warning: false
toc: true
number-sections: true
editor_options:
  chunk_output_type: console
---
```

Explanation

`format` specifies the type of output file to generate. Here it is an HTML file.

`self-contained: true` specifies that the HTML file generated should be standalone file.

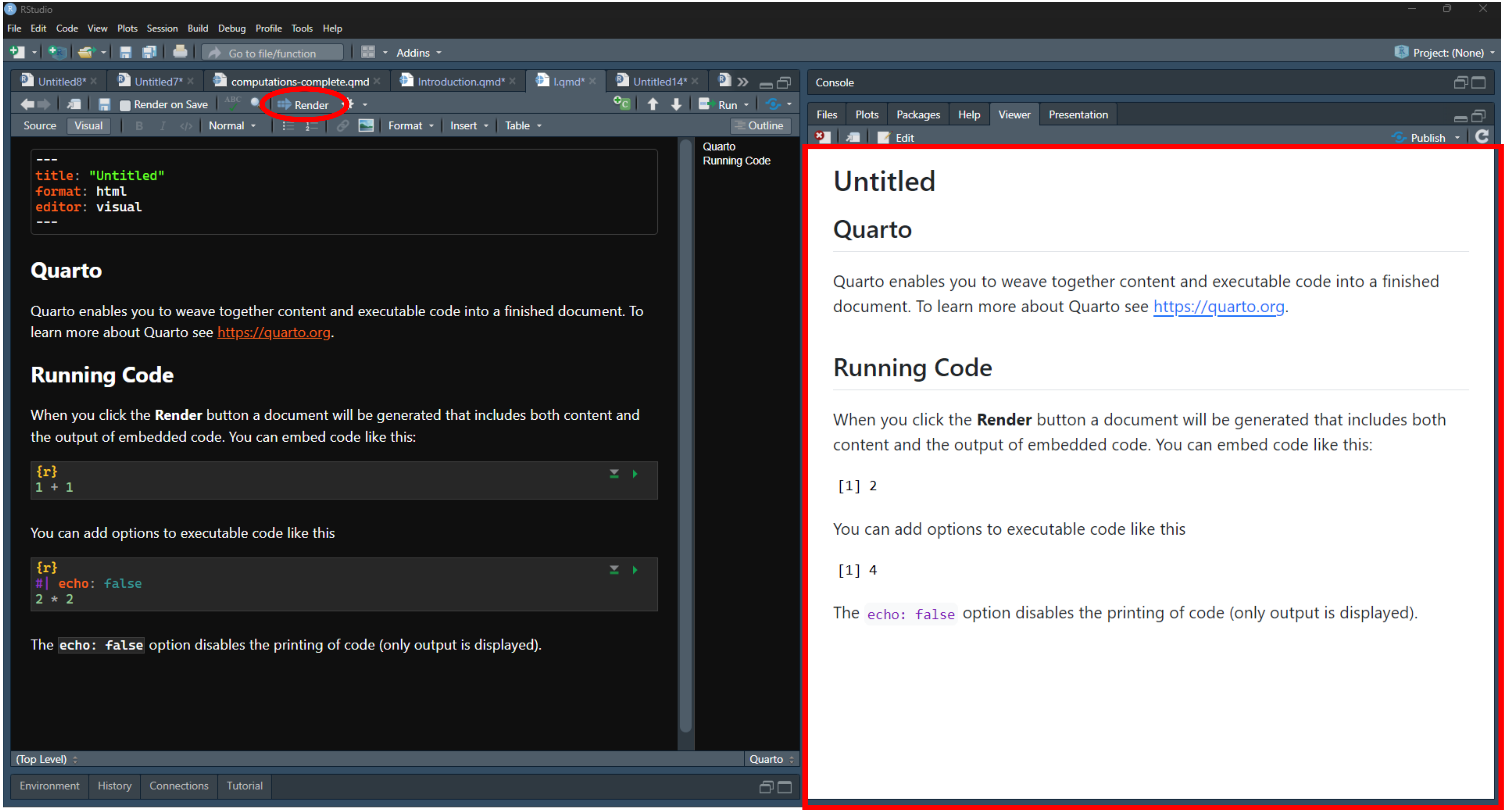
`echo: true` enables the printing of code (only output is displayed), unless otherwise specified.

`warning: false` disables the printing of warning messages.

`toc: true` specifies that the table of contents should be shown; automatically generated based on the headings.

`chunk_output_type: console` specifies that when executing the code in RStudio (i.e., when you "run" the code, instead of "render"), the outcput in RStudio should be displayed in the "console" (where it typically appears when using R)

2 - Output Settings – Render



```
---
title: "Untitled"
format: html
editor: visual
---
```

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:

```
{r}
1 + 1
```

You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

Untitled

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:

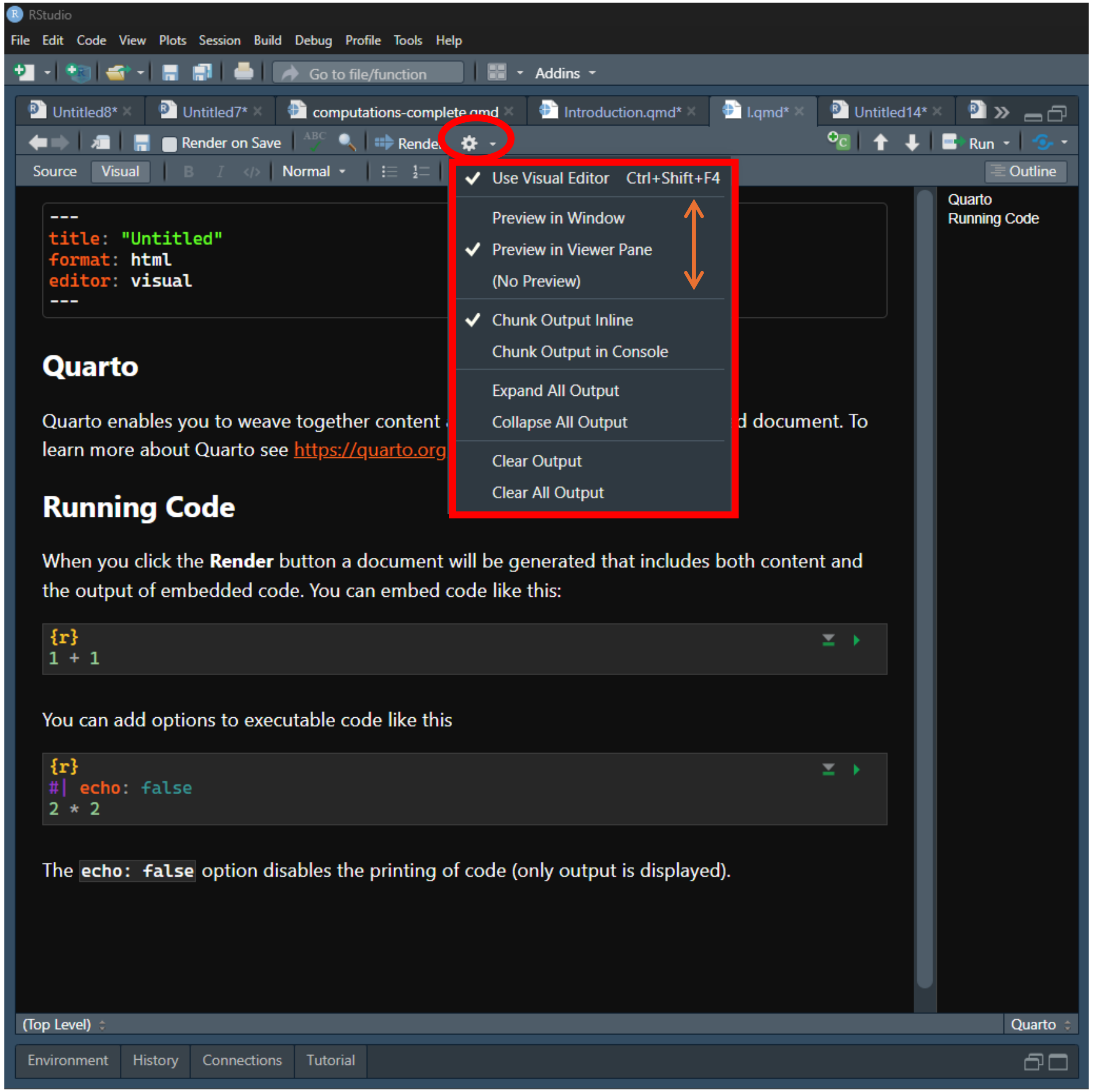
```
[1] 2
```

You can add options to executable code like this

```
[1] 4
```

The `echo: false` option disables the printing of code (only output is displayed).

2 - Output Settings – Render parameter



```
---
title: "Untitled"
format: html
editor: visual
---
```

Quarto

Quarto enables you to weave together content … d 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:

```
{r}
1 + 1
```

You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

- **Preview in viewer pane**: output in R pane
- **Preview in window**: output open with your browser (html), Microsoft office (word), adobe reader (pdf), …

2 - Output Settings – Exercice

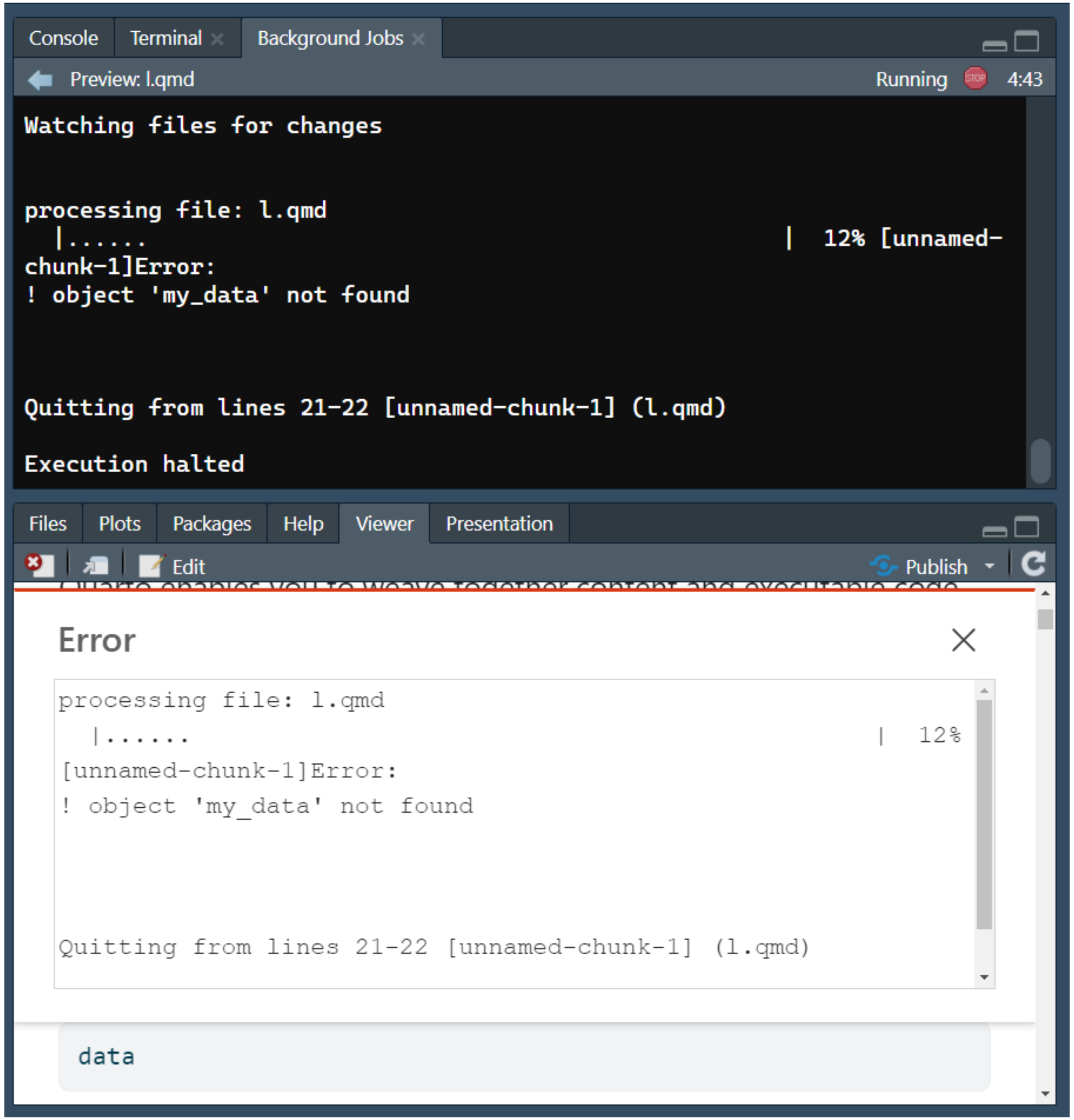
(1) Change your render parameter to “*Preview in window*” and see what happen.

(2) Add “*echo: FALSE*” in your Output settings and see what happen.

(3) Type below 1 + 1, *log(-1)*.
add *warning: false* in your output settings.

(4) Use “html” and “pdf” as format: add
html: default
pdf: default
docx: default
And render to a pdf

2 - Output Settings – Error

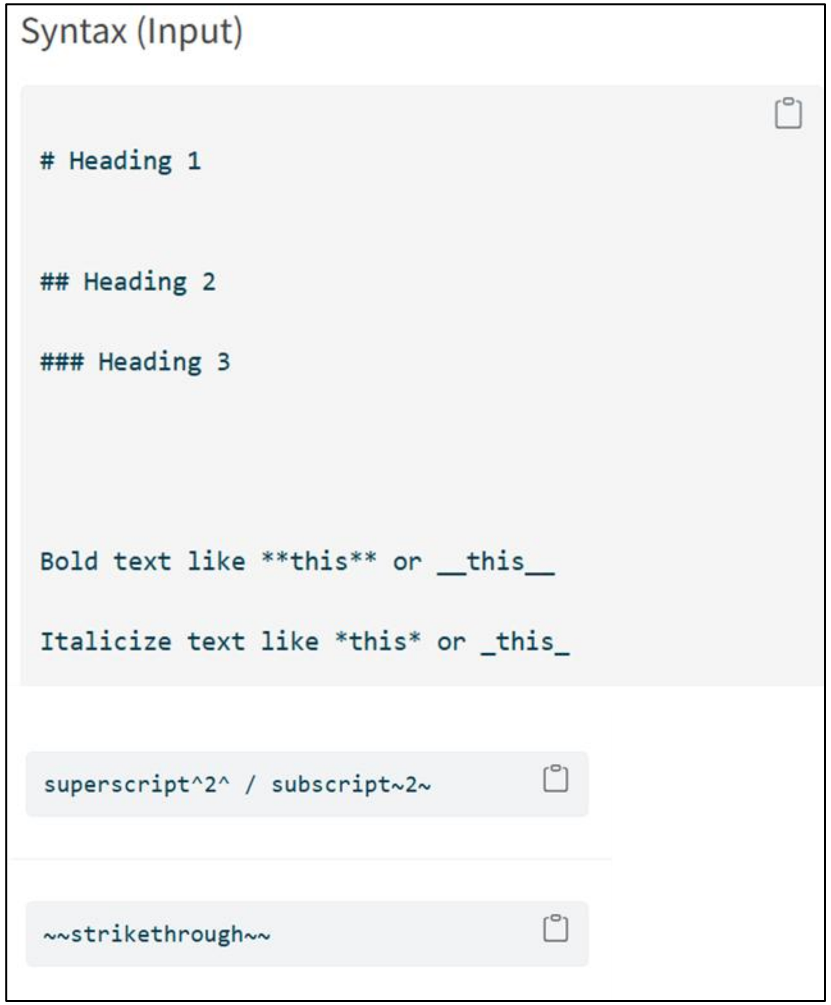


```
Watching files for changes


processing file: l.qmd
  |......                                                            |  12% [unnamed-
chunk-1]Error:
! object 'my_data' not found



Quitting from lines 21-22 [unnamed-chunk-1] (l.qmd)

Execution halted
```

```
processing file: l.qmd
  |......                                                   |  12%
[unnamed-chunk-1]Error:
! object 'my_data' not found



Quitting from lines 21-22 [unnamed-chunk-1] (l.qmd)
```

```
data
```

3 - Formatting Basics

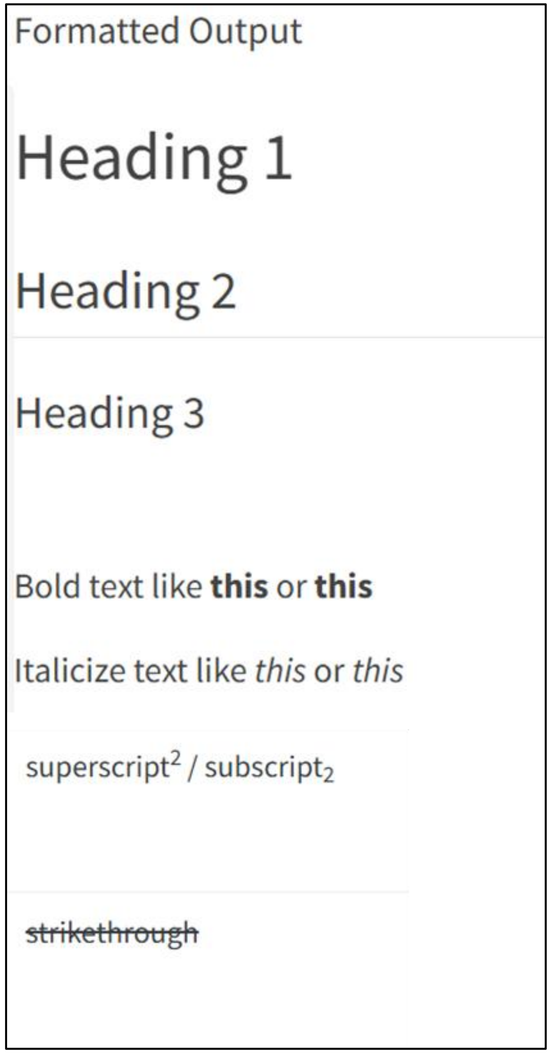


Syntax (Input)

```
# Heading 1


## Heading 2

### Heading 3



Bold text like **this** or __this__

Italicize text like *this* or _this_
```

```
superscript^2^ / subscript~2~
```

```
~~strikethrough~~
```



Formatted Output

Heading 1

Heading 2

Heading 3

Bold text like **this** or **this**

Italicize text like *this* or *this*

superscript^2 / subscript$_2$

~~strikethrough~~

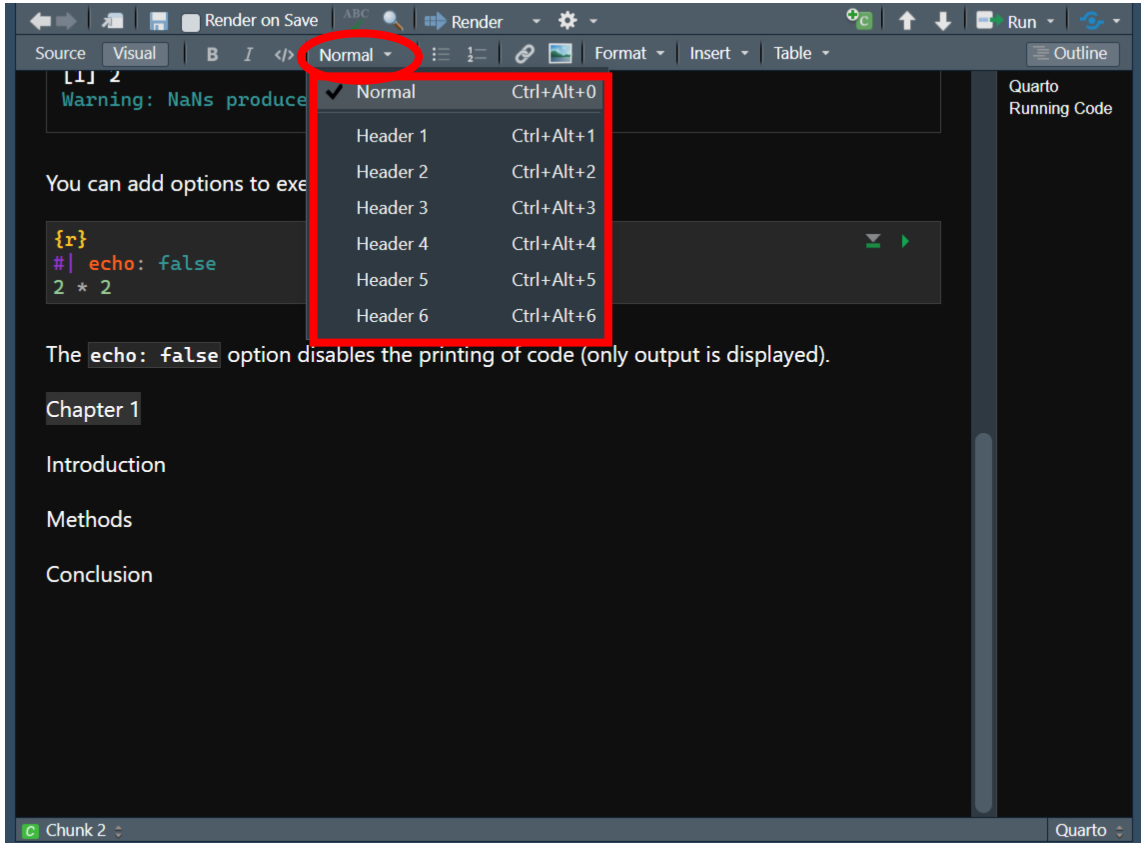
... still in *Visual* ?

3 - Formatting Basics – Exercice

Add in your document:

Chapter 1 as *Header 1*
Introduction as *Header 2*
Conclusion as *Header 2*

With buttons



```
[1] 2
Warning: NaNs produce
```

You can add options to exe

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

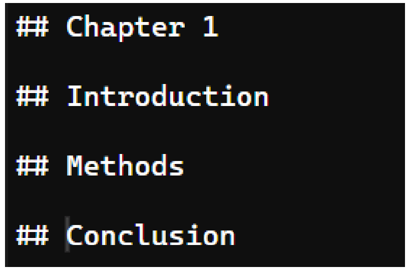
Chapter 1

Introduction

Methods

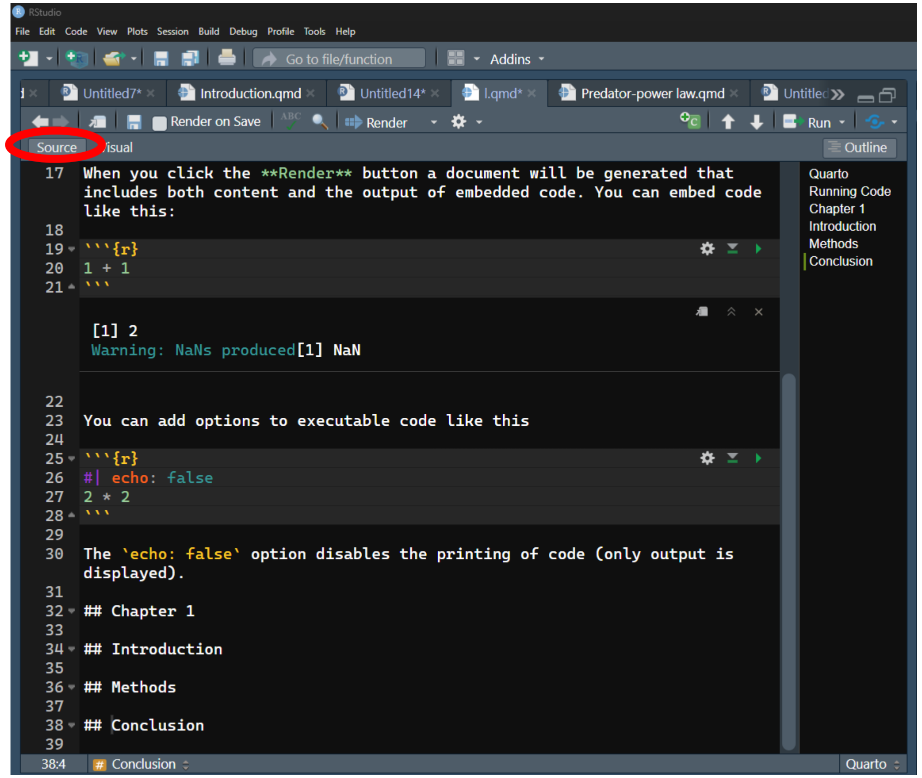
Conclusion

Manually



```
## Chapter 1

## Introduction

## Methods

## Conclusion
```

Source



~~~
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:

```{r}
1 + 1
~~~

[1] 2
Warning: NaNs produced[1] NaN

You can add options to executable code like this

```{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is
displayed).

Chapter 1

Introduction

Methods

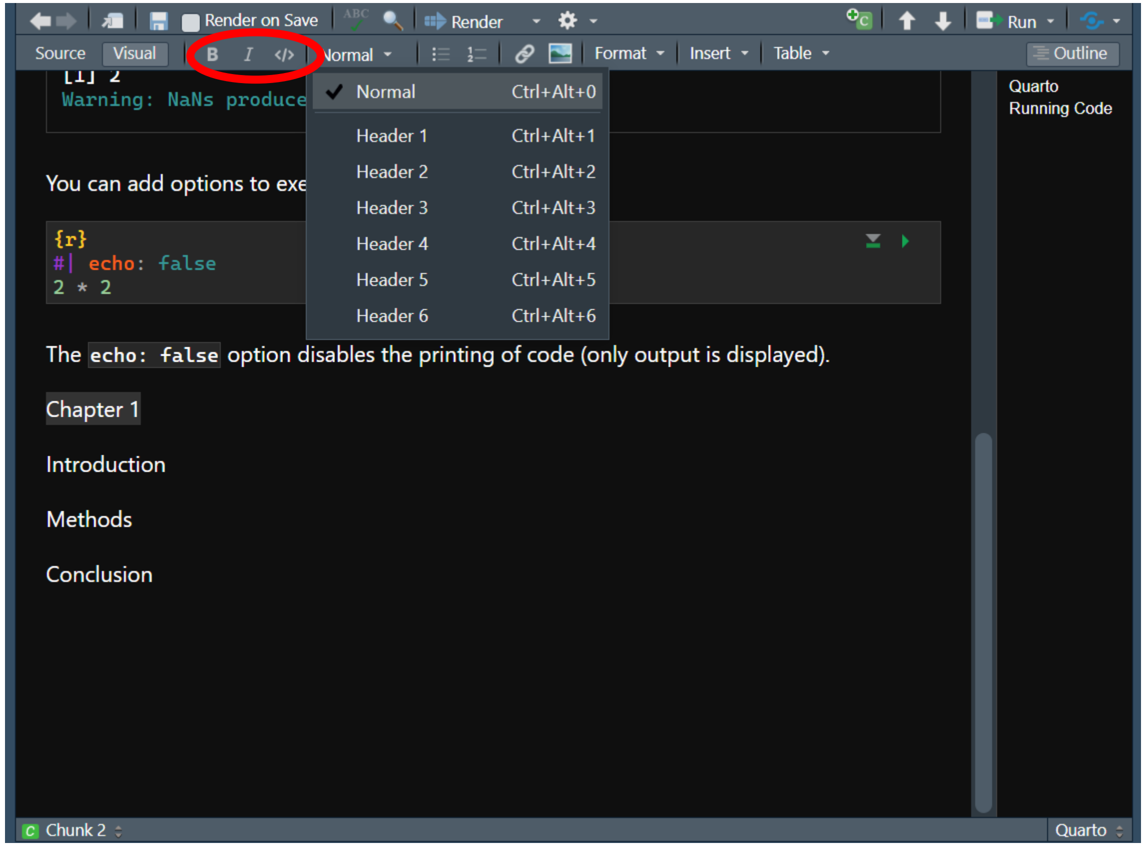
Conclusion
```
```

3 - Formatting Basics – Exercice

Add in your document:

This one is **bold**, but this one is *italic*. I am ***bolditalic***.

With buttons



```
[1] 2
Warning: NaNs produce
```

You can add options to exe

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

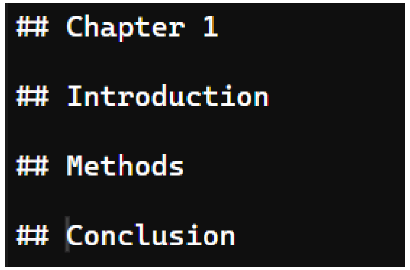
Chapter 1

Introduction

Methods

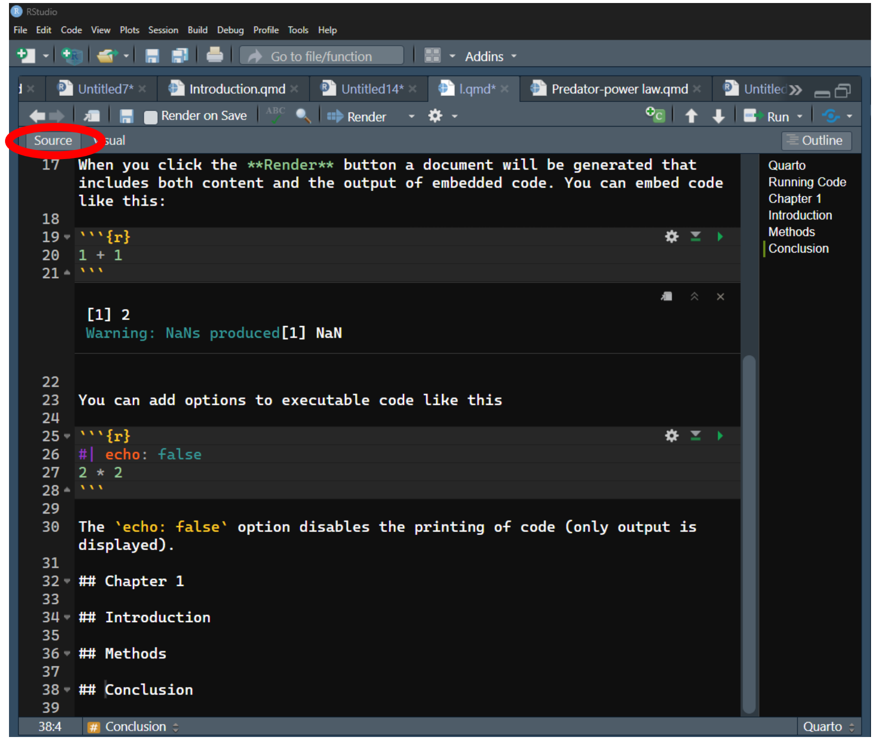
Conclusion

Manually



```
## Chapter 1

## Introduction

## Methods

## Conclusion
```

Source



~~~
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:

```{r}
1 + 1
~~~

 [1] 2
 Warning: NaNs produced[1] NaN

You can add options to executable code like this

```{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is
displayed).

Chapter 1

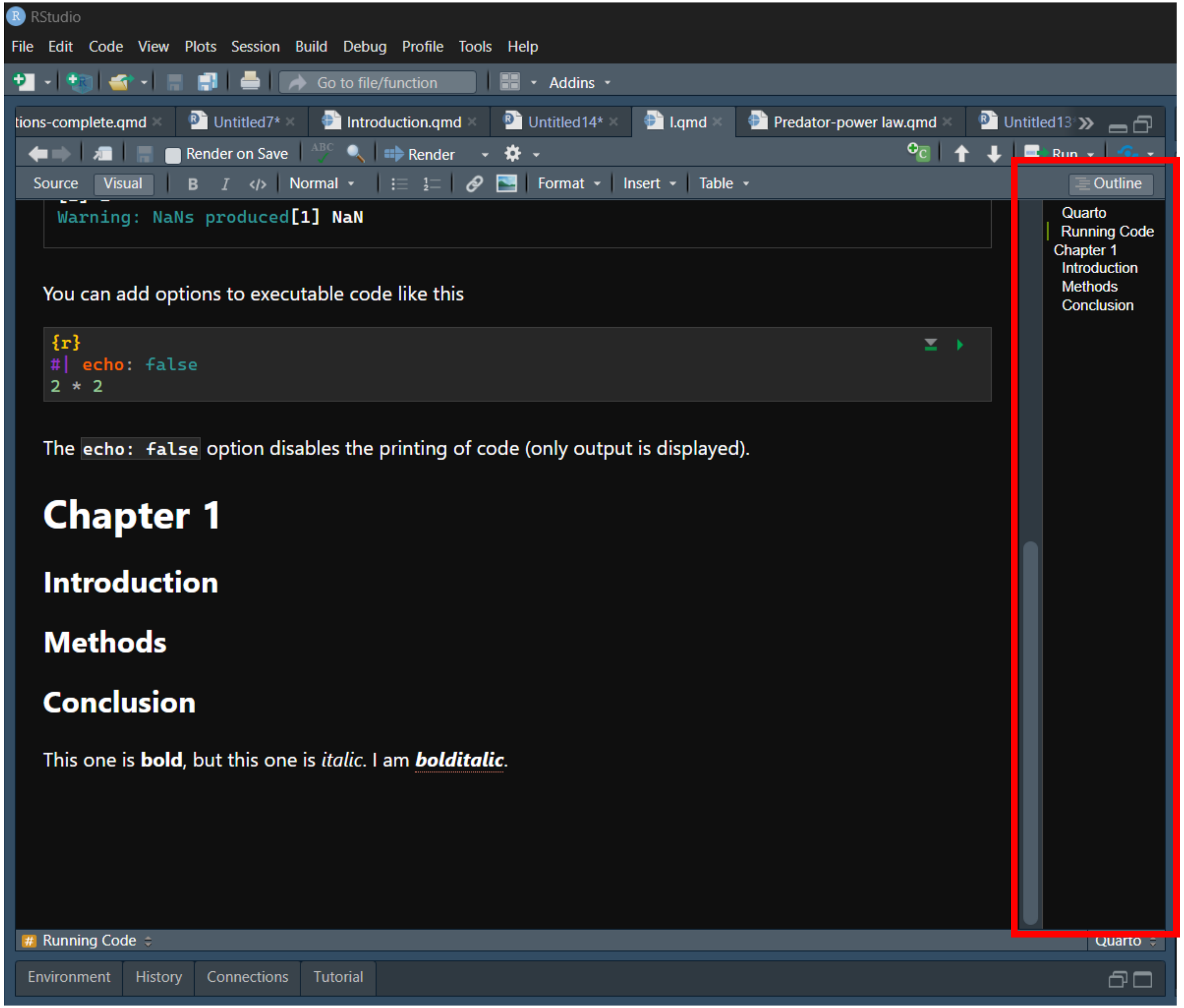
Introduction

Methods

Conclusion

```
```

3 - Formatting Basics – Exercice



```
Warning: NaNs produced[1] NaN
```

You can add options to executable code like this

```
{r}
#| echo: false
2 * 2
```

The `echo: false` option disables the printing of code (only output is displayed).

Chapter 1

Introduction

Methods

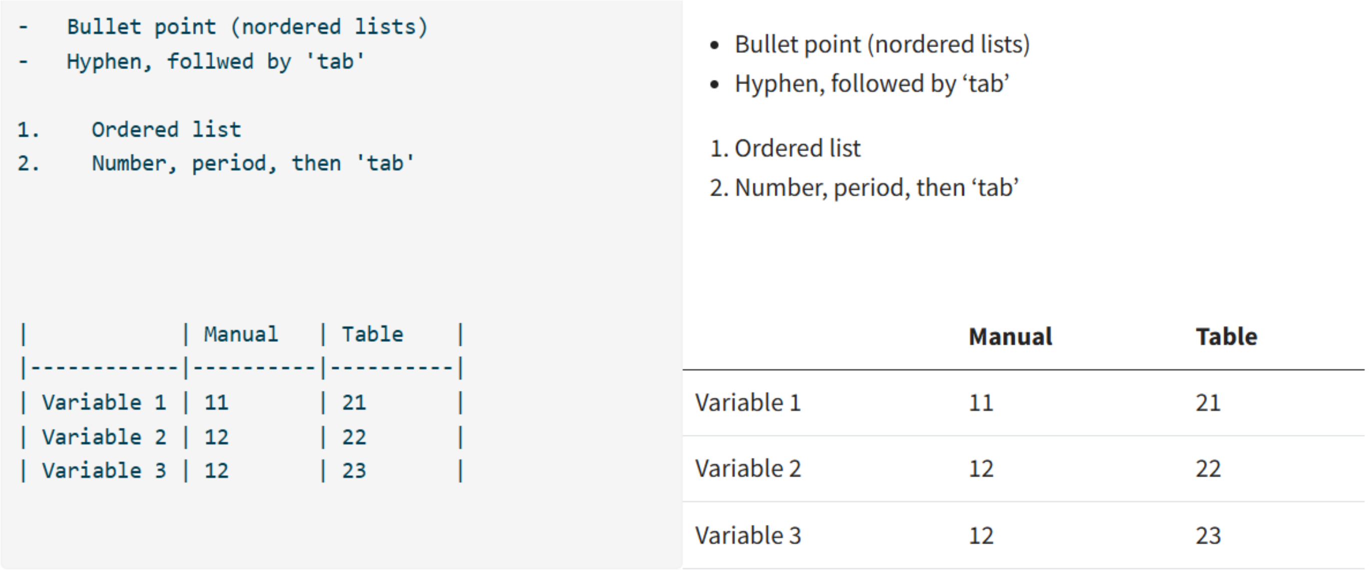
Conclusion

This one is **bold**, but this one is *italic*. I am ***bolditalic***.

- Headers are displayed in outline

3 - Formatting Basics

- The following syntax works with Quarto and its predecessor Markdown. In addition to this code, RStudio has a 'visual' editor which provides a point-and-click interface to format the document.

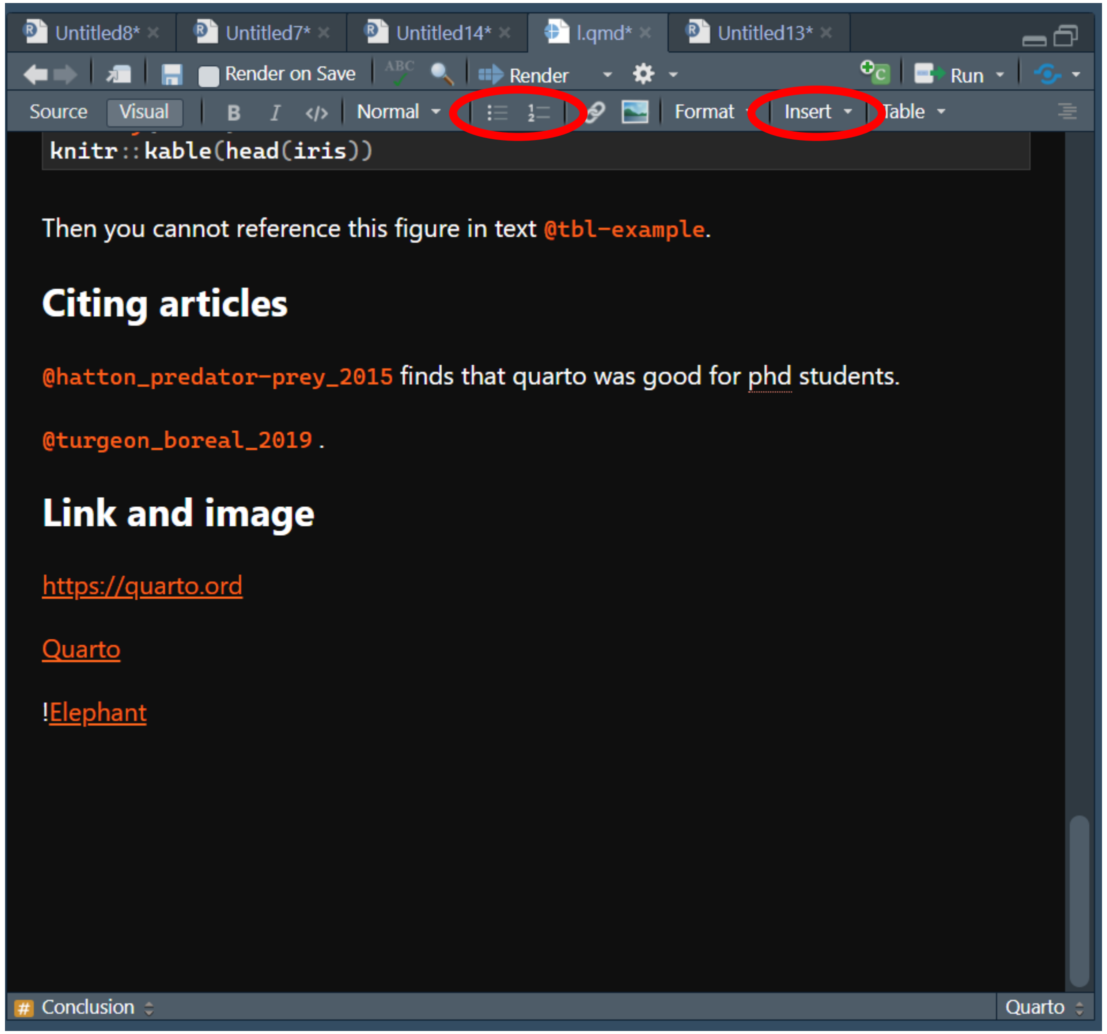


```
-   Bullet point (nordered lists)
-   Hyphen, follwed by 'tab'

1.    Ordered list
2.    Number, period, then 'tab'




|            | Manual   | Table    |
|------------|----------|----------|
| Variable 1 | 11       | 21       |
| Variable 2 | 12       | 22       |
| Variable 3 | 12       | 23       |
```

- Bullet point (nordered lists)
- Hyphen, followed by 'tab'

1. Ordered list
2. Number, period, then 'tab'

	Manual	Table
Variable 1	11	21
Variable 2	12	22
Variable 3	12	23



```
knitr::kable(head(iris))
```

Then you cannot reference this figure in text @tbl-example.

Citing articles

@hatton_predator-prey_2015 finds that quarto was good for phd students.

@turgeon_boreal_2019 .

Link and image

https://quarto.ord

Quarto

!Elephant

Technical terms

- **Inline Code:** allows you to execute code within the text.
- **Outline Code:** allows you to execute code in chunks.

4 - Integrating R Statistical Code – chunks

To execute R code

- Outline

` ` `{r}

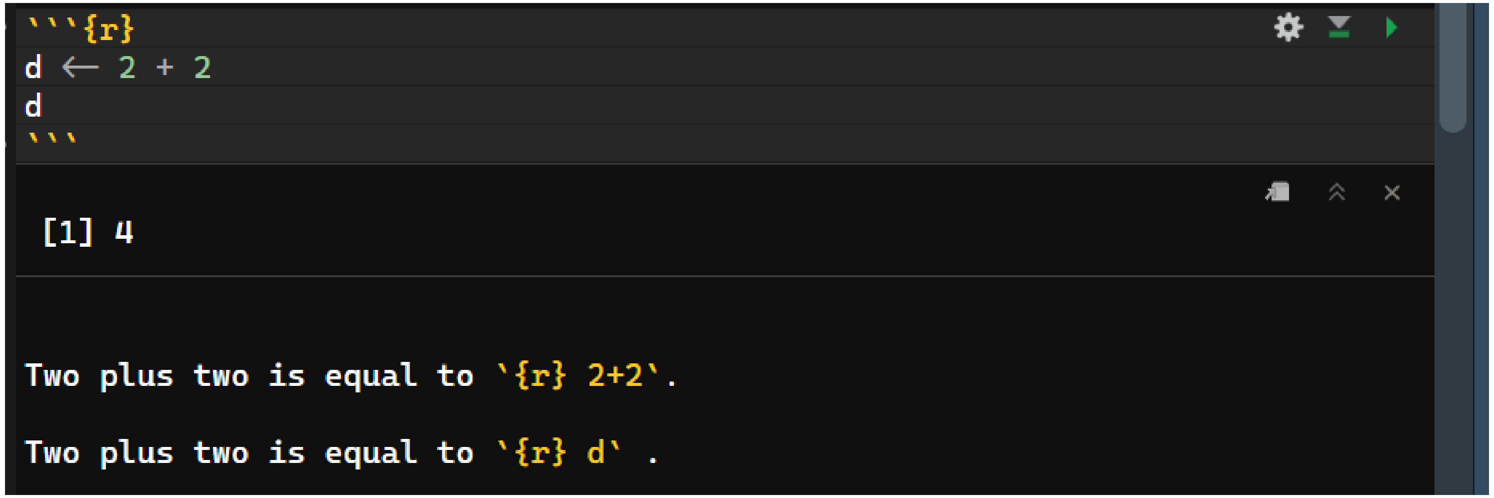
` ` `

Shortcut: Crtl + Alt + I

- **Inline:**

`{r} … `

< Source

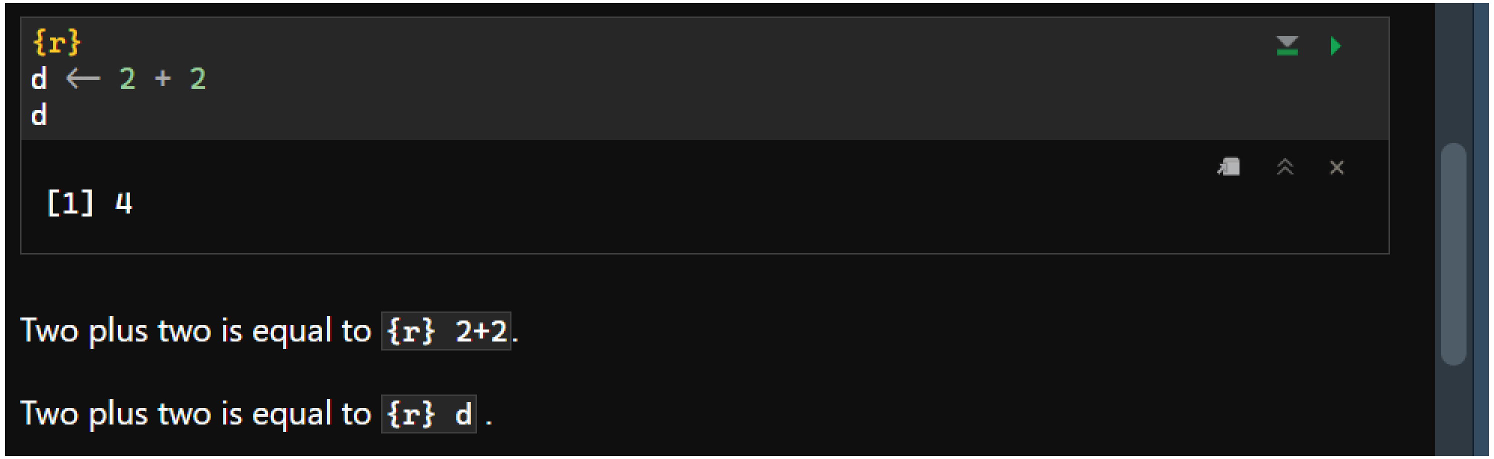


~~~
```{r}
d ← 2 + 2
d
```
~~~

```
[1] 4
```

```
Two plus two is equal to `{r} 2+2`.

Two plus two is equal to `{r} d` .
```

< Visual



```
{r}
d ← 2 + 2
d
```

```
[1] 4
```

Two plus two is equal to `{r} 2+2`.

Two plus two is equal to `{r} d` .

4 - Integrating R Statistical Code – Exercice

(1) Create an object d = 2 + 2 in a chunk

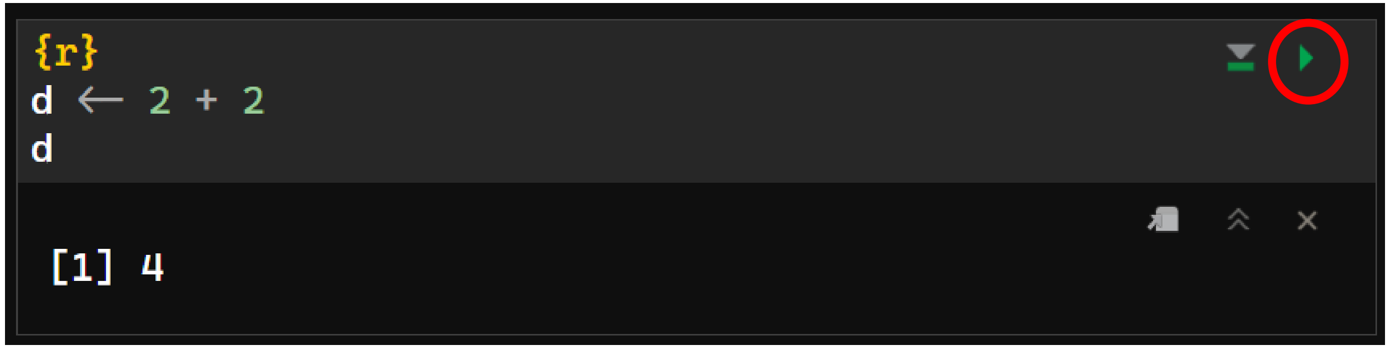
~~~
` ` `{r}
d ← 2+2
` ` `
~~~

(2) In the body text, type:

Two plus two is equal to `{r} 2+2`

Two plus two is equal to `{r} d`

Display your R codes without render …

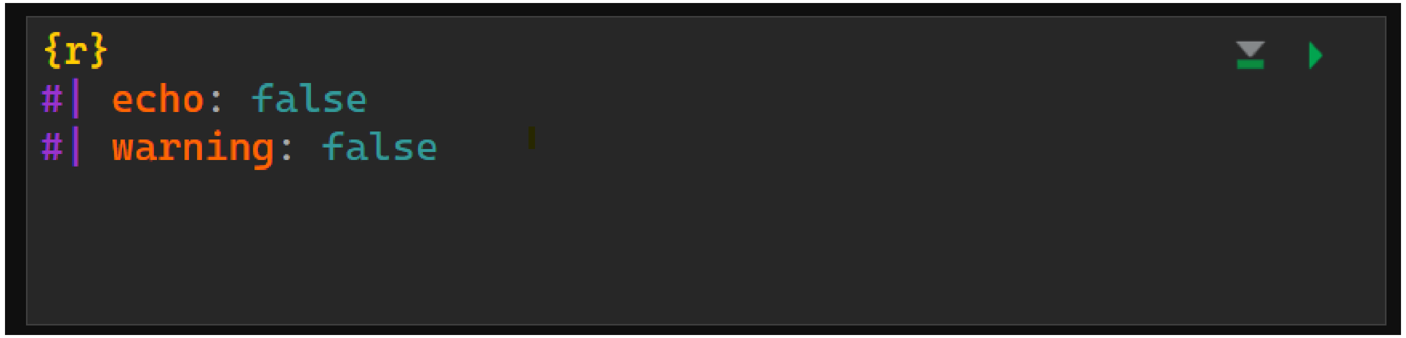


~~~
{r}
d ← 2 + 2
d
~~~

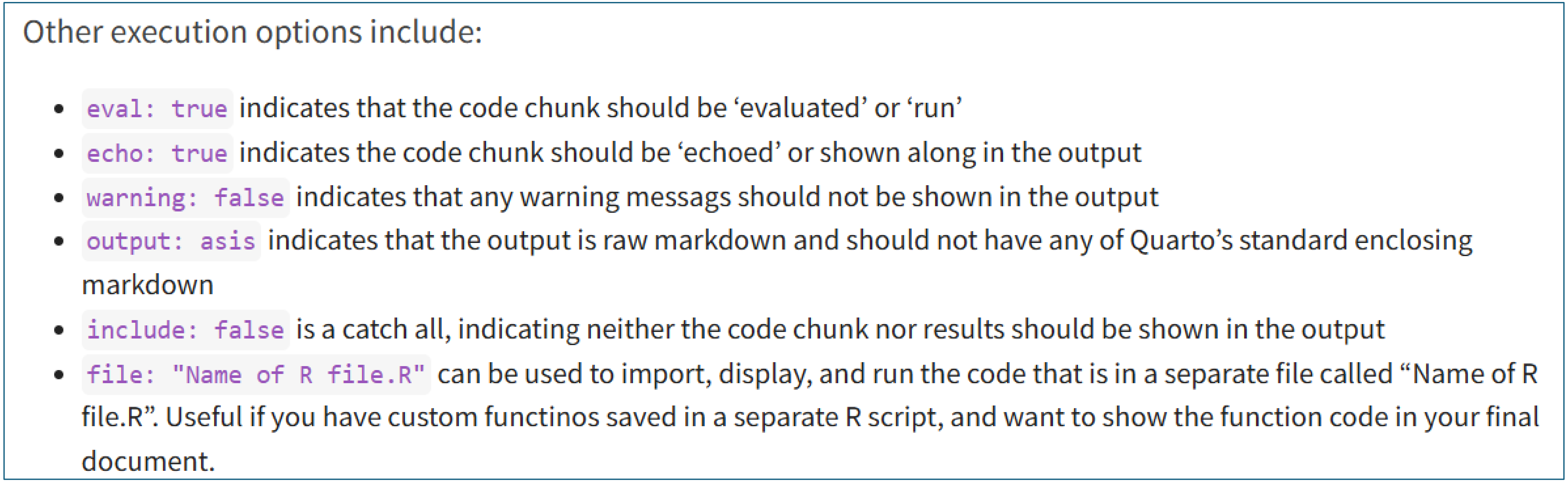
~~~
[1] 4
~~~

4 - Integrating R Statistical Code – Execution Options

Allow to handle chunks output … use #|



```
{r}
#| echo: false
#| warning: false
```

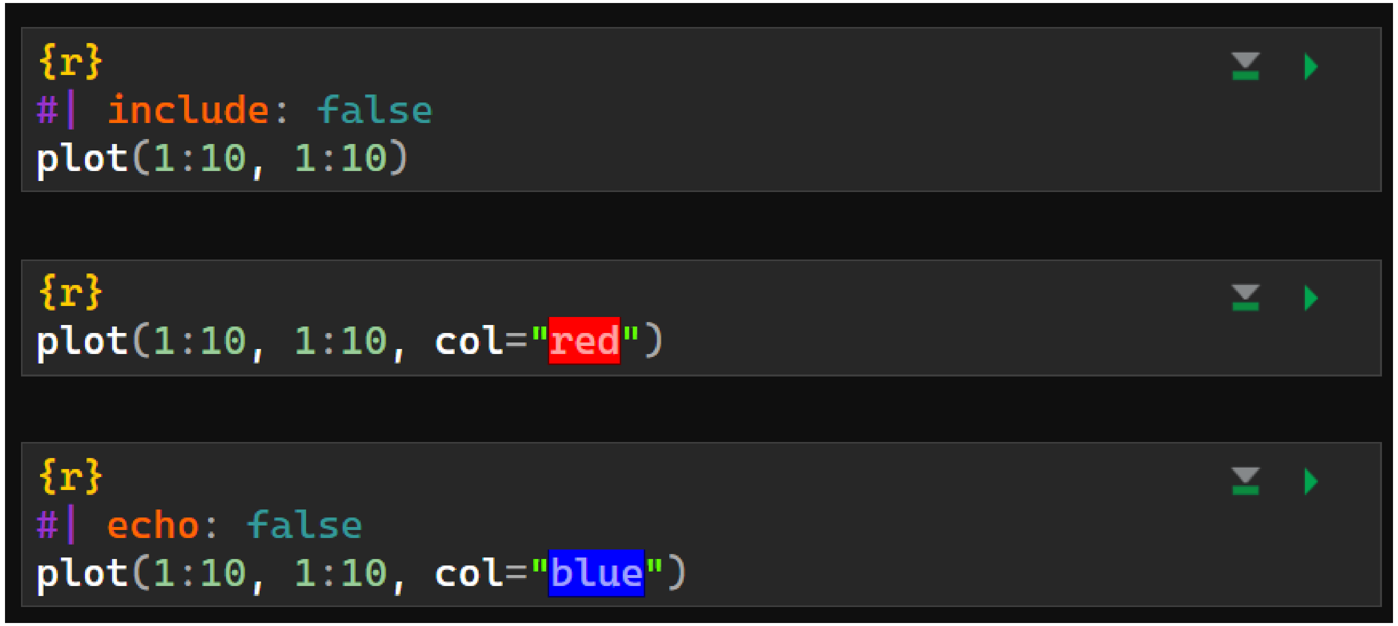


Other execution options include:

- `eval: true` indicates that the code chunk should be 'evaluated' or 'run'
- `echo: true` indicates the code chunk should be 'echoed' or shown along in the output
- `warning: false` indicates that any warning messags should not be shown in the output
- `output: asis` indicates that the output is raw markdown and should not have any of Quarto's standard enclosing markdown
- `include: false` is a catch all, indicating neither the code chunk nor results should be shown in the output
- `file: "Name of R file.R"` can be used to import, display, and run the code that is in a separate file called "Name of R file.R". Useful if you have custom functinos saved in a separate R script, and want to show the function code in your final document.

4 - Integrating R Statistical Code – Exercice

- Type the following code in your document and see how is the output by clicking on render.



```
{r}
#| include: false
plot(1:10, 1:10)
```

```
{r}
plot(1:10, 1:10, col="red")
```

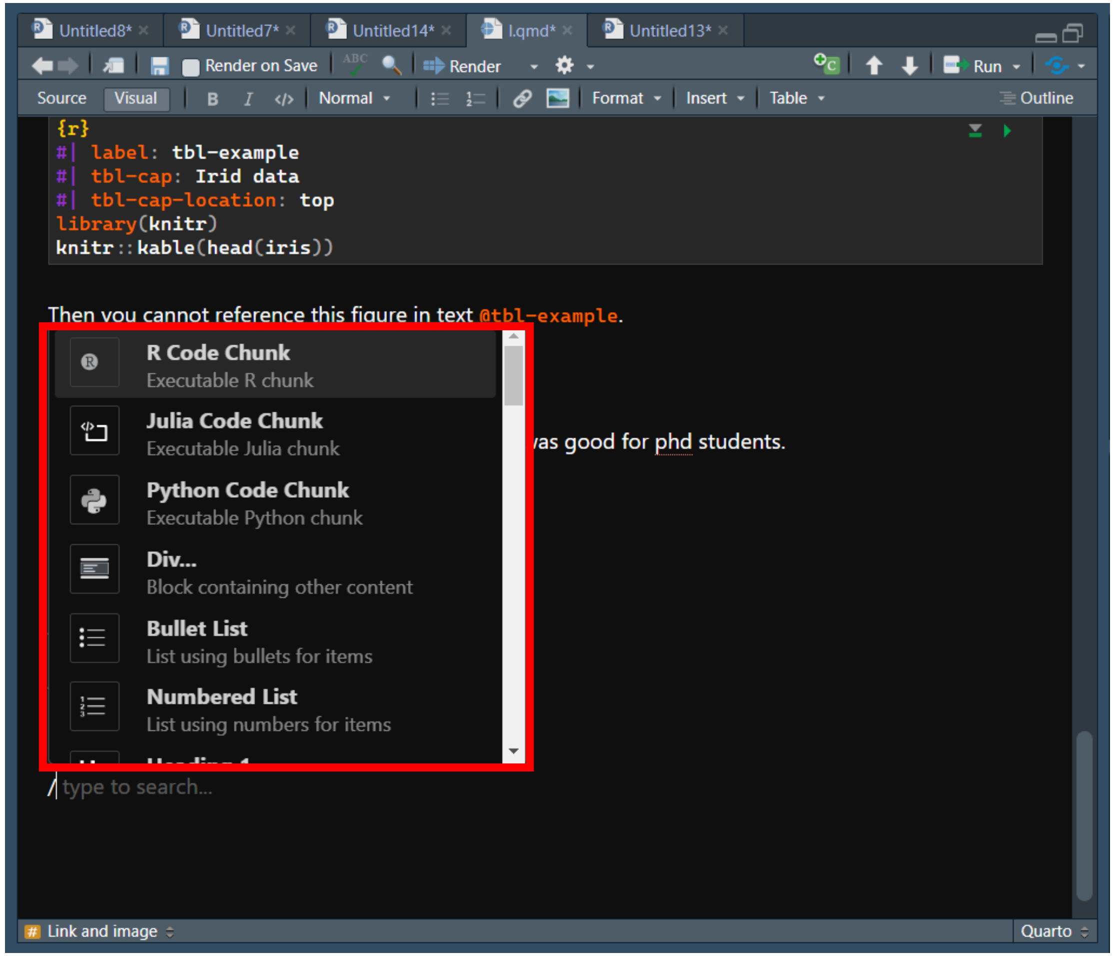
```
{r}
#| echo: false
plot(1:10, 1:10, col="blue")
```

4 - Integrating R Statistical Code - workflow

- Package codes, working directory, … should be put in R chunks.
- Objects already created in a chunk can be used in other chunks in the documents …

Shortcut / display options

- Type **/…**



```
{r}
#| label: tbl-example
#| tbl-cap: Irid data
#| tbl-cap-location: top
library(knitr)
knitr::kable(head(iris))
```

Then you cannot reference this figure in text @tbl-example.

…as good for phd students.

/type to search...

5 - Technical writing - Equation

- Use `$ … $` for inline
- For outline, use

```
$$
…
$$
```

- `Einstein equation: $E = mc^2$` → Einstein's equation: $E = mc^2$

-
  ```
  $$
  E = mc^2
  $$
  ```
  →
  $$E = mc^2$$

-
  ```
  $$
  \frac{1}{2}
  $$
  ```
  →
  $$\frac{1}{2}$$

-
  ```
  $$
  Y = X_1 + X_2
  $$
  ```
  →
  $$Y = X_1 + X_2$$

-
  ```
  $$
  \sqrt{p}
  $$
  ```
  →
  $$\sqrt{p}$$

-
  ```
  $$
  \sum_{i = 1}^{n}{(\bar{x} - x_i)^2}
  $$
  ```
  →
  $$\sum_{i=1}^{n}(\bar{x} - x_i)^2$$

-
  ```
  $$
  Y \sim X\beta_0 + X\beta_1 + \epsilon
  $$
  ```
  →
  $$Y \sim X\beta_0 + X\beta_1 + \epsilon$$

-
  ```
  $$
  Pr(\theta | y) = \frac{Pr(y | \theta)}{Pr(y)}
  $$
  ```
  →
  $$Pr(\theta|y) = \frac{Pr(y|\theta)Pr(\theta)}{Pr(y)}$$

6 - Referecing: citing Figures, Tables, links, ...

Link to tables or figures or section in text.

- Step 1: create the figure/image in a chunk

- Step 2: add the execution option #| label:

  #| label: fig-label

  #| label: tbl-label

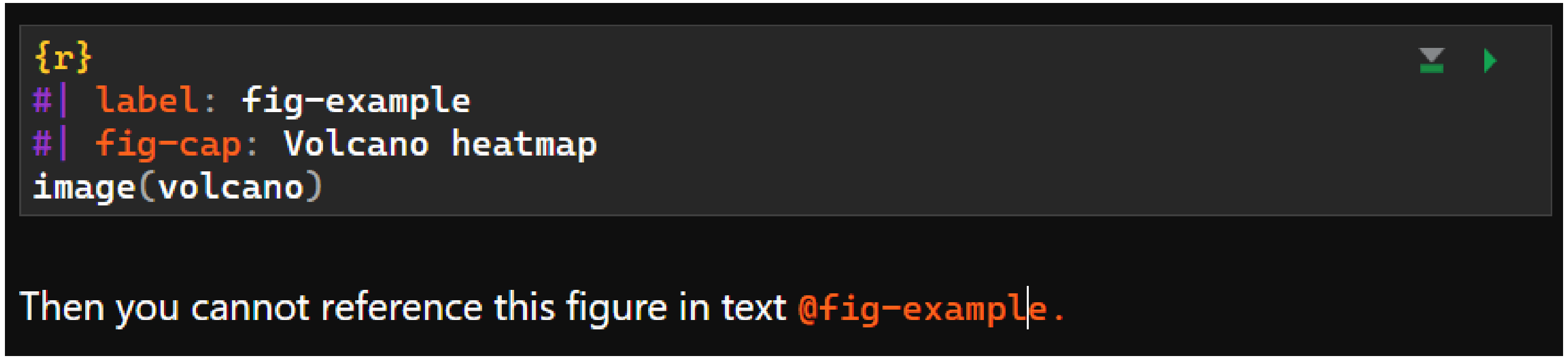
- Step 3: refer it in your text using @...

  @fig-label

  @tbl-label

Referencing figures

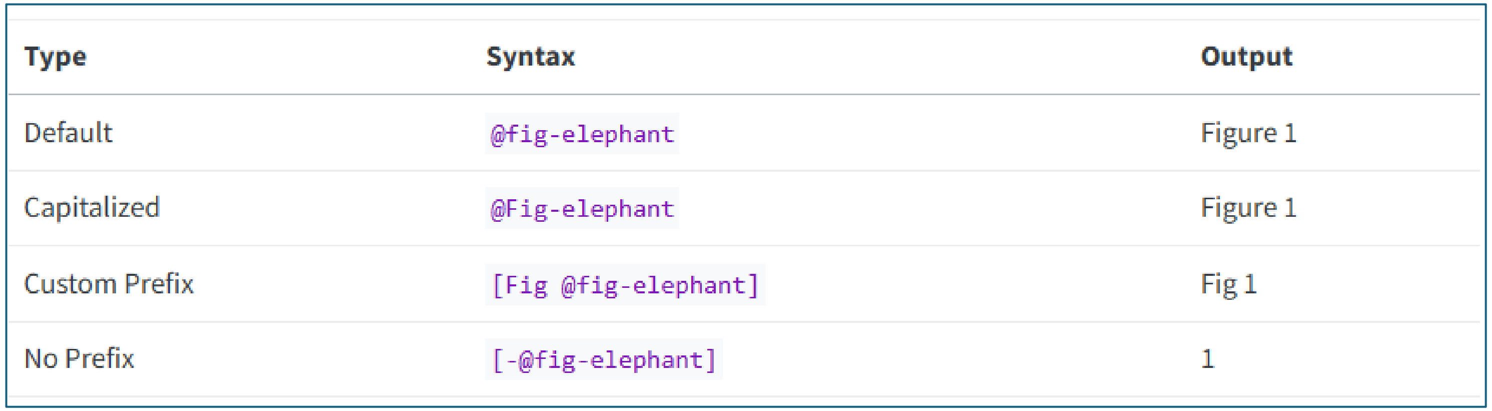
- Outline (chunks)



```
{r}
#| label: fig-example
#| fig-cap: Volcano heatmap
image(volcano)
```

Then you cannot reference this figure in text @fig-example.

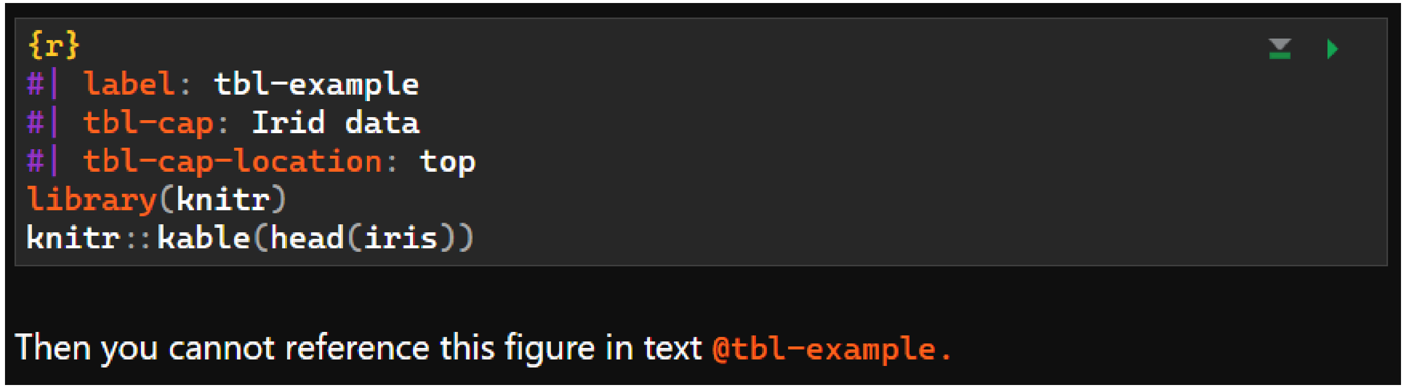
- Inline



Type	Syntax	Output
Default	`@fig-elephant`	Figure 1
Capitalized	`@Fig-elephant`	Figure 1
Custom Prefix	`[Fig @fig-elephant]`	Fig 1
No Prefix	`[-@fig-elephant]`	1

Referencing tables

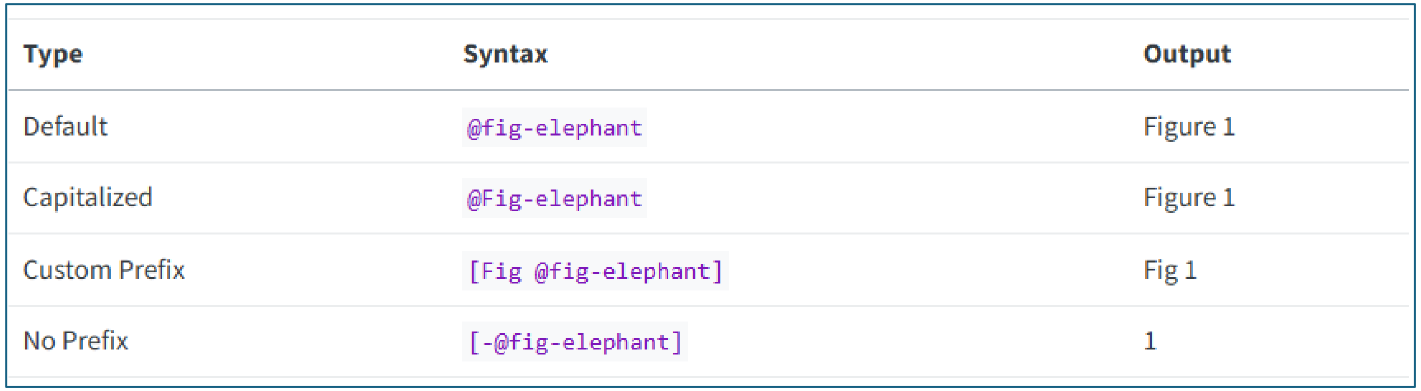
- Created in Outline (chunks)



```
{r}
#| label: tbl-example
#| tbl-cap: Irid data
#| tbl-cap-location: top
library(knitr)
knitr::kable(head(iris))
```

Then you cannot reference this figure in text `@tbl-example.`

- Used in Inline



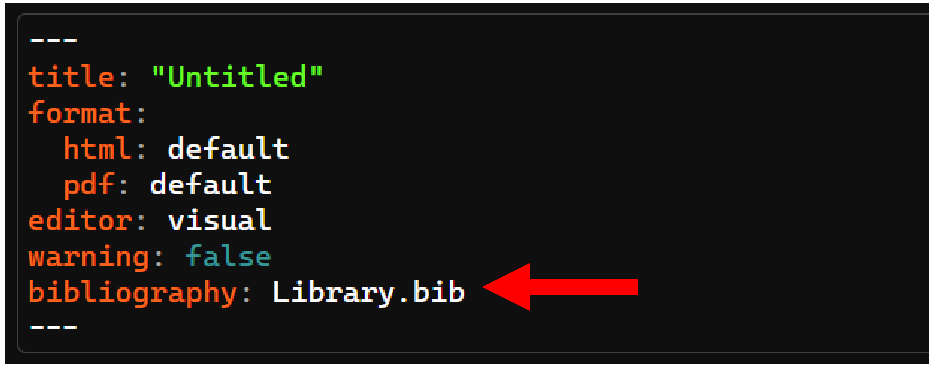
Type	Syntax	Output
Default	`@fig-elephant`	Figure 1
Capitalized	`@Fig-elephant`	Figure 1
Custom Prefix	`[Fig @fig-elephant]`	Fig 1
No Prefix	`[-@fig-elephant]`	1

Citing articles

(1) Exporter votre library (ex: Zotero, Mendley, …) en **.bib**

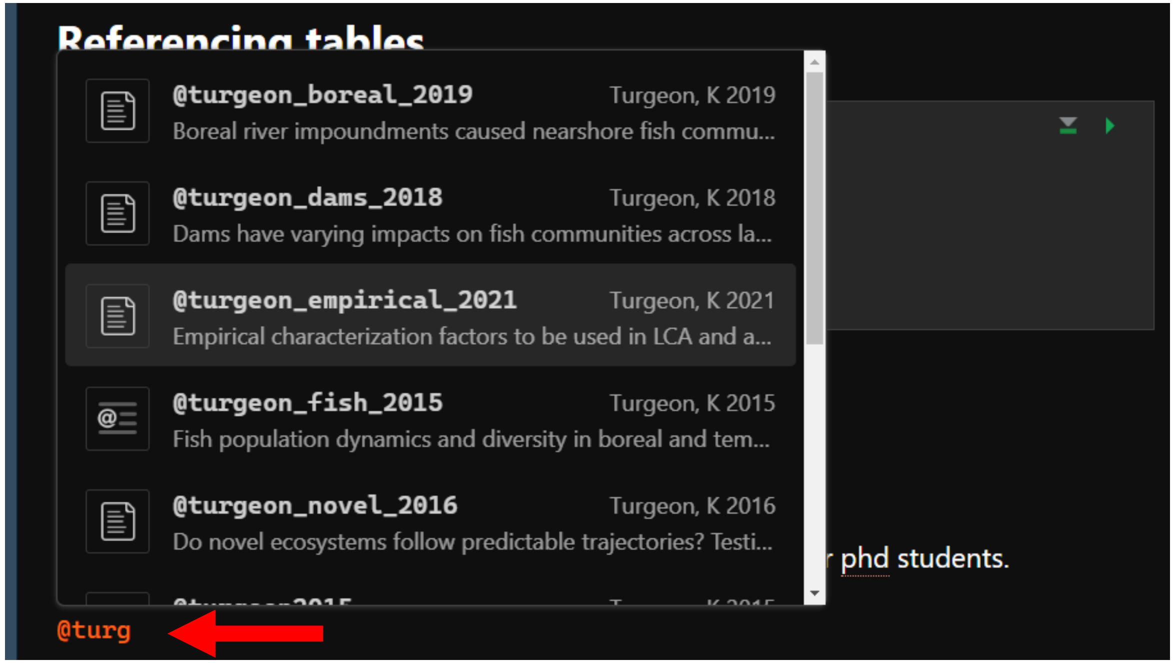
(2) Add the file in your working directory.

(3) Add **bibliography: name.bib** in the output settings.



```
---
title: "Untitled"
format:
  html: default
  pdf: default
editor: visual
warning: false
bibliography: Library.bib
---
```

(4) In the text type @….



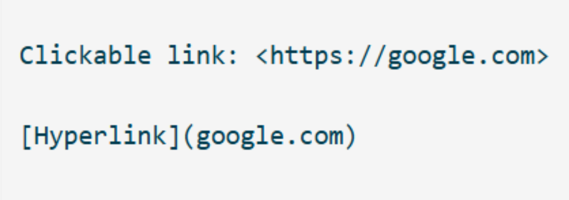
Referencing tables

@turgeon_boreal_2019 Boreal river impoundments caused nearshore fish commu…	Turgeon, K 2019
@turgeon_dams_2018 Dams have varying impacts on fish communities across la…	Turgeon, K 2018
@turgeon_empirical_2021 Empirical characterization factors to be used in LCA and a…	Turgeon, K 2021
@turgeon_fish_2015 Fish population dynamics and diversity in boreal and tem…	Turgeon, K 2015
@turgeon_novel_2016 Do novel ecosystems follow predictable trajectories? Testi…	Turgeon, K 2016

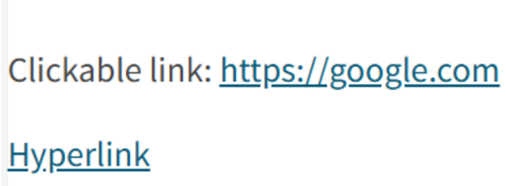
r phd students.

```
@turg
```

Referencing link, image, ...

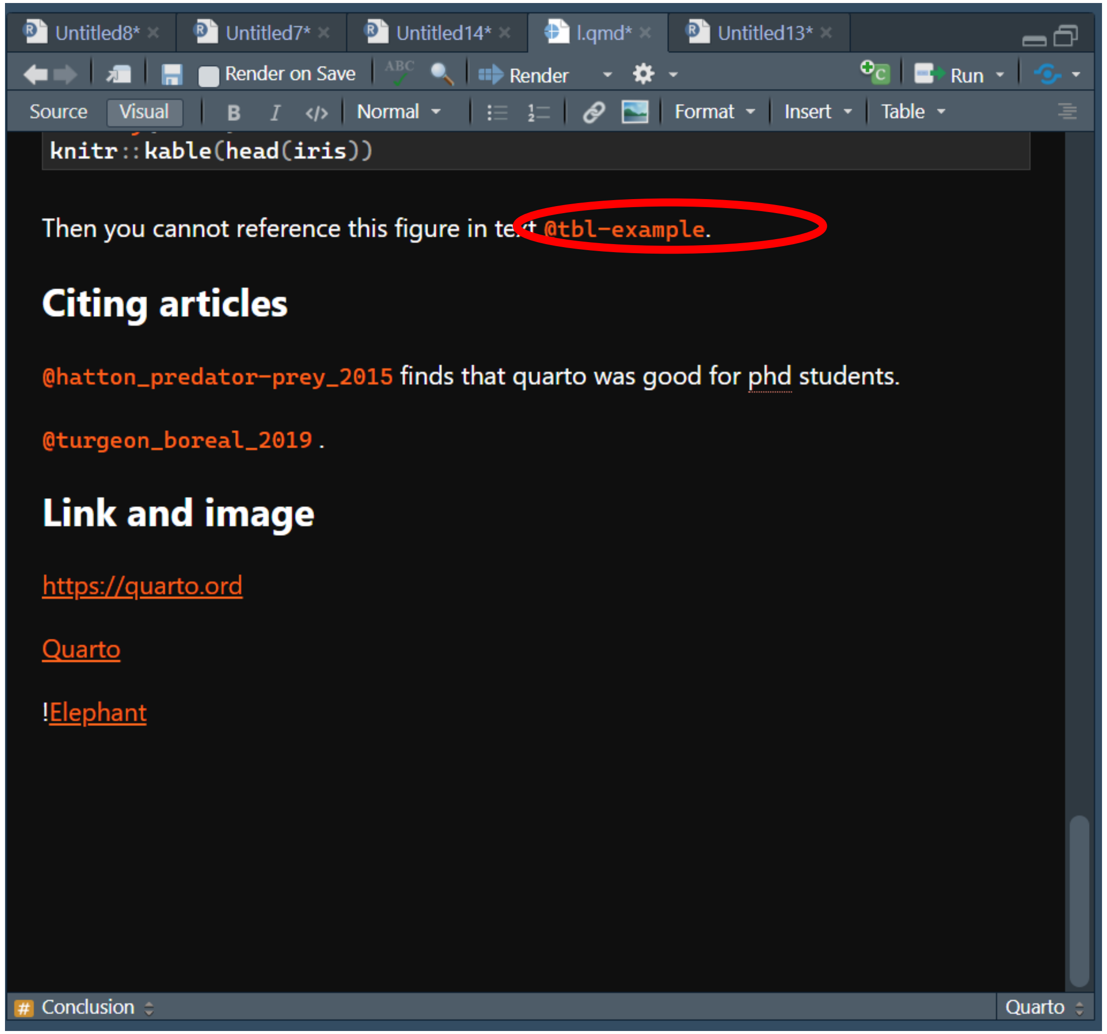


```
Clickable link: <https://google.com>

[Hyperlink](google.com)
```



Clickable link: https://google.com

Hyperlink



```
knitr::kable(head(iris))
```

Then you cannot reference this figure in text @tbl-example.

Citing articles

@hatton_predator-prey_2015 finds that quarto was good for phd students.

@turgeon_boreal_2019 .

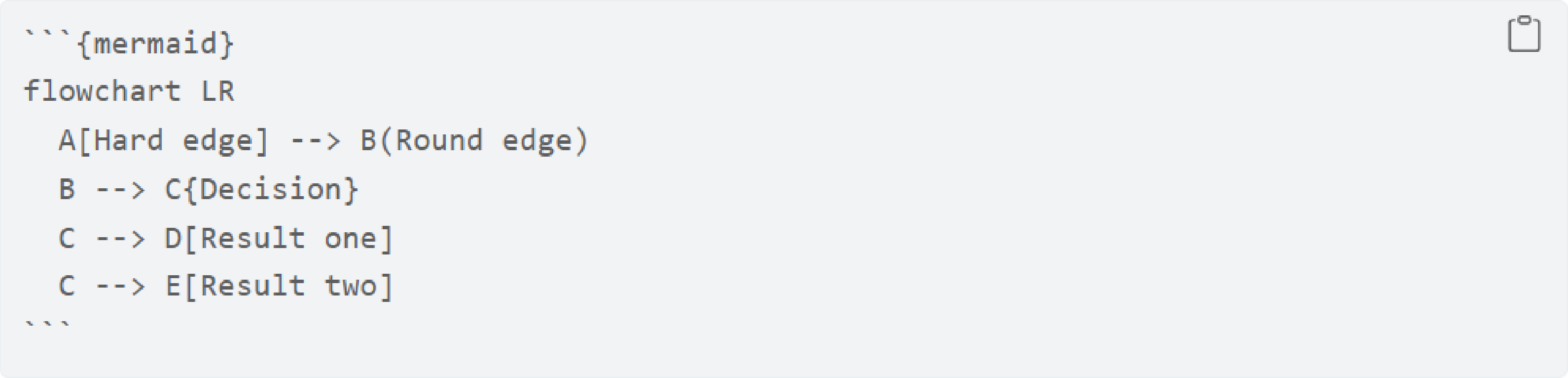
Link and image

https://quarto.ord

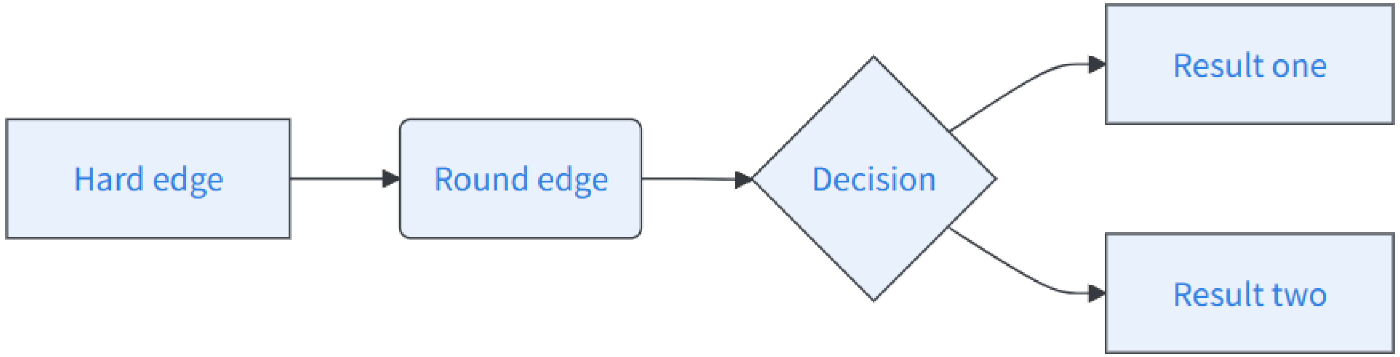
Quarto

!Elephant

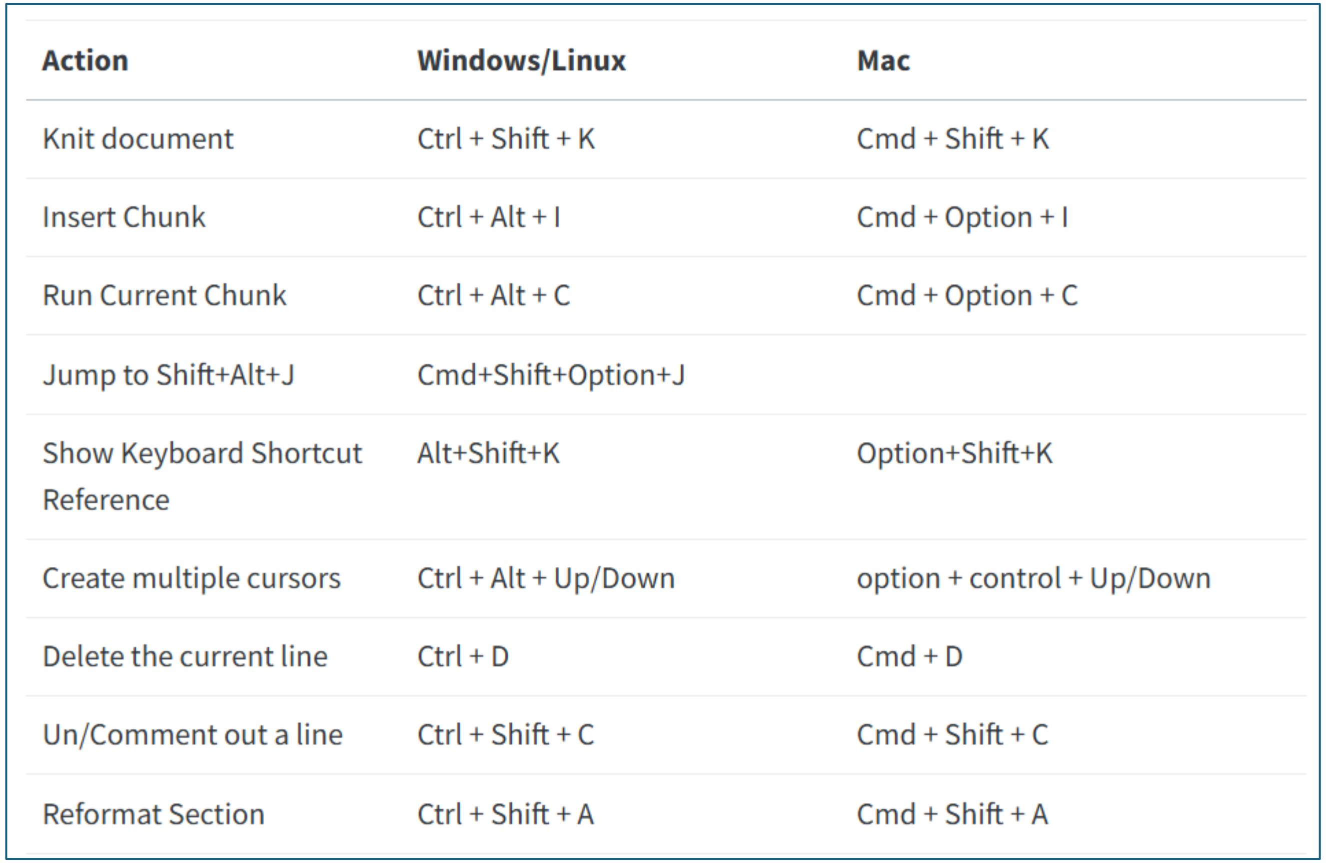
Diagrams



~~~
```{mermaid}
flowchart LR
  A[Hard edge] --> B(Round edge)
  B --> C{Decision}
  C --> D[Result one]
  C --> E[Result two]
```
~~~



Common shortcuts



Action	Windows/Linux	Mac
Knit document	Ctrl + Shift + K	Cmd + Shift + K
Insert Chunk	Ctrl + Alt + I	Cmd + Option + I
Run Current Chunk	Ctrl + Alt + C	Cmd + Option + C
Jump to Shift+Alt+J	Cmd+Shift+Option+J	
Show Keyboard Shortcut Reference	Alt+Shift+K	Option+Shift+K
Create multiple cursors	Ctrl + Alt + Up/Down	option + control + Up/Down
Delete the current line	Ctrl + D	Cmd + D
Un/Comment out a line	Ctrl + Shift + C	Cmd + Shift + C
Reformat Section	Ctrl + Shift + A	Cmd + Shift + A

The end . . .