Publish and Share with Quarto : : CHEATSHEET

Document Structure

```
---
title: "I Love Quarto"
author: "Quarto Lover"
format: pdf
execute:
  echo: false
---

```{r}
#| label: setup
#| include: false
library(tidyverse)
```

Why Quarto Is

Quarto [@quarto]
really easy! Yo
and also include code! @fig-mtcars shows
a great plot using `ggplot`:

```{r}
#| label: fig-mtcars
#| fig-cap: "A cool ggplot"

mtcars %>%
  ggplot() %>%
  geom_point(aes(x = cyl, y = mpg))
```
```

Set format(s) and options Use YAML Syntax

Don't show output of code chunk (still run code)

Write with Markdown
RStudio: Help > Markdown Quick Reference
Also use **Visual Editor**

Include code R, Python, etc.

Markdown

This code:

```
Markdown makes writing easy!
You can use **bold**, *italics*,
~~strikethrough~~, `code`,
^superscript^ and ~subscript~
formatting. Also en--dashes and
em---dashes.
New lines and extra    spaces
are ignored. Start a new
paragraph with a

blank line in-between. Make a
new line (no new paragraph)
with two spaces at the end of a
line.
Some other formatting syntax:

# Heading 1 {#sec-label}
## Heading 2
  (Up to 6#)

- A list!
  - Nested items
- Fun!

1. Numbered list!
3. Markdown can count
1. Even if you can't

Write "{{< pagebreak >}}" for
a page break in Quarto.
```

Works on any heading level

Outputs:

Markdown makes writing easy! You can use **bold**, *italics*, ~~strikethrough~~, code, ^{superscript} and $_{subscript}$ formatting. Also en–dashes and em—dashes. New lines and extra spaces are ignored. Start a new paragraph with a

blank line in-between. Make a new line (no new paragraph) with two spaces at the end of a line.
Some other formatting syntax:

Heading 1
Heading 2

- A list!
  - Nested items
- Fun!

1. Numbered list!
2. Markdown can count
3. Even if you can't

Set Options

Set document options and format(s) in the YAML header:

```
---
title: "Quarto Document"
subtitle: "very cool!"
author: "Me"
toc: true
format:
  pdf:
    number-sections: false
  html:
    number-sections: true
---
```

Top-level options apply to all formats

All indents are 2 spaces

Useful options:

- `toc:` Include table of contents?
- `number-sections:` Automatically add section numbers?
- `mainfont:` Main document font
- `lot:` Include list of tables?
- `lof:` Include list of figures?

For a full list of options, see quarto.org/docs/reference

Include Code

Code cells start with ```{language} and end with ```

Also use **Insert > Executable Cell**

```
```{r}
#| label: chunk-id
#| other chunk options
<insert code here>
```
```

```
---
execute:
  echo: false
  error: true
---
```

Set chunk options locally or globally in YAML header

Inline Code

```
The value is `r 2+2`.
```
Output: The value is 4.

Option	Default	Effects
echo	true	false: hide code fenced: include code syntax
eval	true	false: don't run code
include	true	false: don't include code or results (still run code)
output	true	false: don't include results asis: treat results as raw markdown
warning	true	false: don't include warnings in output
error	false	true: include error in output and continue with render

Add Content

See reverse for more info

Figures

```
![Caption](image.png){#fig-LABEL width="6in"}
```

```
```{r}
#| label: fig-LABEL
#| fig-cap: Caption
#| fig-width: "6in"

ggplot(data) + ...
```
```

Tables

```
```{r}
#| label: tbl-LABEL
#| tbl-cap: Caption

knitr::kable(data)
```
```

Also see flextable

Citations

Add a bibliography (BibTeX) file to the YAML header:

```
---
bibliography: references.bib
---
```

Add citations: `@citation` or `[@citation]`

You can build your bibliography file from Zotero

Equations (LaTeX)

Use single $ for inline math:
`$x = 2$` becomes $x = 2$.

For equations (separate line), use double $$:

```
The quadratic formula (@eq-quad) is cool.
$$
x = \frac{-b\pm\sqrt{b^2-4ac}}{2a}
$$ {#eq-quad}
```

becomes

The quadratic formula (Equation 1) is cool.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (1)$$

Cross-References

Add labels: `#| label: prefix-LABEL` in code cells and `{#prefix-LABEL}` otherwise

Reference in document: `@prefix-LABEL`

Prefix:	*Renders:*
fig-	Figure 1
tbl-	Table 1
sec-	Section 1 (Chapter if in a Quarto book)
eq-	Equation 1

For more information go to quarto.org/docs/guide

Figure Options

In code chunks, options are given by `#| option: value`

````
```{r}
#| label: fig-LABEL
#| fig-pos: H
   etc.
ggplot(data) + ...
```
````

For static images, the options are given by `{option=value}`

```
![Cap](img.png){#fig-LABEL
width="6in" etc.}
```

Useful figure options:

`fig-align`	figure horizontal alignment (default, left, right, or center)
`fig-cap`	figure caption
`fig-scap`	short figure caption (for list of figures in LaTeX)
`fig-pos`	LaTeX figure position argument. Set to "H" to keep the figure exactly where it is in the document.
`fig-width`	width of figure
`fig-height`	height of figure

Units can be in, cm, pt, etc. ("6in")

For static images (not code chunks), use `width` and `height` (no `fig-` prefix)

Tables

kable/kableExtra

Most calls to kable will look something like the following:

````
```{r}
#| label: tbl-mytable
#| tbl-cap: My Table
data %>%
  knitr::kable() %>% or kbl()
  kable_styling() from the kableExtra package
```
````

There are many options and tweaks that kable and kableExtra provide, see the kableExtra documentation for more details. Two things in particular:

To force the position of a table in the document, use

```
kable_styling(latex_options = "HOLD_position")
```

`col_spec()` defines properties of columns (such as width)

flextable

flextable is also cool. Check out the flextable documentation or cheatsheet for more info (ardata-fr.github.io/flextable-book/)

LaTeX

LaTeX is the language used for equations in Quarto documents. Inline math is surrounded by $, and equations (on a separate line) are surrounded by $$.

LaTeX commands are preceded by a backslash (\). Some common commands and constructs include:

`x^2`	x^2	`\pm`	$\pm$
`x_{i,j}`	$x_{i,j}$	`\mp`	$\mp$
`\neq`	$\neq$	`\partial`	∂
`\times`	$\times$	`\div`	$\div$
`\approx`	$\approx$	`\in`	$\in$
`\leq`	$\leq$	`\frac{2}{3}`	$\frac{2}{3}$
`\geq`	$\geq$		
`\sqrt{2}`	$\sqrt{2}$	`\sqrt[n]{3}`	$\sqrt[n]{3}$
`\ell`	ℓ	`\implies`	$\implies$
`\dot{a}`	$\dot{a}$	`\ddot{a}`	$\ddot{a}$
`\tilde{x}`	$\tilde{x}$	`\bar{x}`	$\bar{x}$
`\hat{x}`	$\hat{x}$	`\vec{x}`	$\vec{x}$

Use `\text{}` to write non-math text in an expression:

`9.8~\text{m}/\text{s}^2` $9.8\ \text{m}/\text{s}^2$

Many functions (trig, logs, limits, etc) need to be escaped with a backslash, otherwise they print italicized:

`\sin`	$\sin$	`\log`	$\log$	`\min`	$\min$
`\cos`	$\cos$	`\ln`	$\ln$	`\max`	$\max$
`\tan`	$\tan$	`\lim`	$\lim$		

Use `\begin{aligned}...\end{aligned}` and & to align multi-line equations (use \\ for a line break):

```
$$
\begin{aligned}
1 + 2 &= 3 \\
1 &= 3 - 2
\end{aligned}
$$
```

$$\begin{aligned} 1 + 2 &= 3 \\ 1 &= 3 - 2 \end{aligned}$$

Commands for Greek letters are just the names of the letters (capitalize the first letter to get the capital Greek letter):

`\alpha`	α	`\beta`	β
`\gamma`	γ	`\Gamma`	Γ

etc.

Fences

LaTeX "fences" include the following:

`()`	$()$	`\langle\rangle`	$\langle\rangle$
`[]`	$[]$	`\lfloor\rceil`	$\lfloor\rceil$

Curly braces group expressions, so to print one it needs to be escaped with a backslash:

`\min \{x,h\}` $\min\{x,h\}$

Fences can grow with the enclosed expression by using `\left` and `\right`:

`\left\langle i,2^{2^i}\right\rangle` $\left\langle i, 2^{2^i} \right\rangle$

Several symbols have "limits". In equations these are displayed above/below the symbol, but inline these are displayed to the side:

`$\sum_{n=1}^{\infty} \frac{1}{n}$` $\sum_{n=1}^{\infty} \frac{1}{n}$

```
$$
\sum_{n=1}^{\infty}
\frac{1}{n}
$$
```

$$\sum_{n=1}^{\infty} \frac{1}{n}$$

These symbols include:

`\int`	$\int$	`\iint`	$\iint$
`\lim_{x\to \infty}`	$\lim_{x\to\infty}$		
`\sum`	$\sum$	`\prod`	$\prod$

Quarto Projects

CREATE WEBSITES, BOOKS, AND MORE

A directory of Quarto documents + a configuration file (_quarto.yml)

See examples at https://quarto.org/docs/gallery/

Get started from the command line:

```
Terminal
quarto create project {type}
```

Use **File** > **New Project**

Spacing

Spacing in equations is automatic, but explicit spaces can be used:

`f\!(x)`	$f\!(x)$
`f(x)`	$f(x)$
`f\,(x)`	$f\,(x)$
`f\:(x)`	$f\:(x)$
`f\;(x)`	$f\;(x)$
`f\ (x)`	$f\ (x)$
`f\quad(x)`	$f\quad(x)$
`f\qquad(x)`	$f\qquad(x)$

Examples

```
\lim_{h\to 0}\frac{
f(x+h)-f(x)}{h}
```

$$\lim_{h\to 0} \frac{f(x+h) - f(x)}{h}$$

```
\int x^2\,dx = x^3/3 + C
```

$$\int x^2\,dx = x^3/3 + C$$

```
f_n =
\begin{cases}
a &\text{if }n=0 \\
r f_{n-1} &\text{else}
\end{cases}
```

$$f_n = \begin{cases} a & \text{if } n = 0 \\ r f_{n-1} & \text{else} \end{cases}$$