Reproducible research reports with quarto

Eline Van Geert and Lisa Koßmann

Open Science Day - KU Leuven

May 6, 2025

github.com/ElineVG/Quarto_workshop_OSD2025

Get started

- Install quarto: https://quarto.org/docs/get-started/
- Download the workshop materials
- Open your favorite local editor (VS Code / Jupyter / RStudio) or join online via RStudio Cloud

Get started

- When using R, install the following packages:

```r
pkg_list <- c(
  "fontawesome", "tidyverse", "quarto", "rmarkdown", "palmerpenguins", "reticulate", "knitr"
  )
install.packages(pkg_list)
```

- When using Python, install the following packages using the terminal:

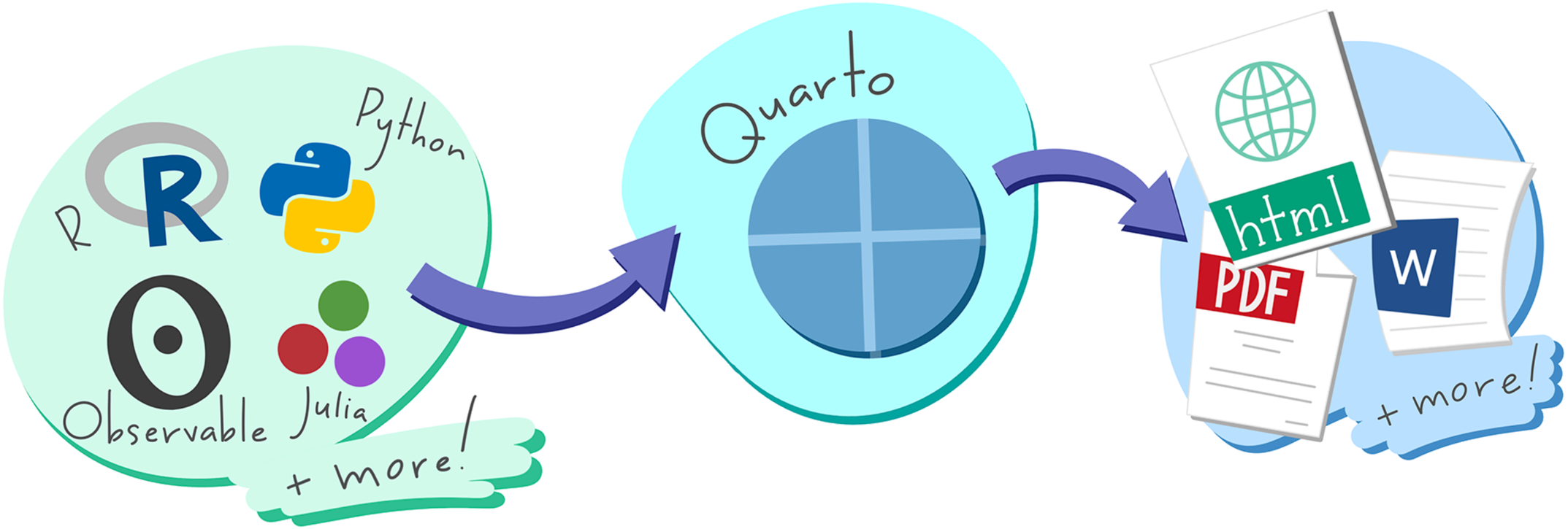
```
py -m pip install jupyter numpy matplotlib palmerpenguins tabulate IPython
py -m pip install jupyterlab                # only when working with Jupyter Lab
py -m jupyter lab test.ipynb                # only when working with Jupyter Lab
```

- For Python use combined with R, install these packages using the R package reticulate:

```r
reticulate::py_install(c("matplotlib", "palmerpenguins", "tabulate", "IPython"))
```

What is quarto®?

An open-source publishing system to combine text and code into formatted output documents



Artwork from “Hello, Quarto” keynote by Julia Lowndes and Mine Çetinkaya-Rundel, presented at RStudio Conference 2022.

Illustrated by Allison Horst.

Why quarto®?

> "Friends don't let friends copy-paste" (Aust & Barth, 2023)

Computational non-reproducibility is a widespread problem:

(Artner et al., 2021; Eubank, 2016; Konkol, Kray & Pfeiffer, 2019)

- Typos, copy-paste errors, incorrect rounding, and other reporting errors
- Forgetting to update report after changing data or analysis
- Forgetting to properly save and document data and scripts
- Not indicating software and packages used including their version numbers

Why quarto®?

Quarto can help you make your research more reproducible (same data, same results):

- Avoid errors by combining text, code, and code outputs from the start!
- Also provides an easy way to share and document your code

Quarto can also make you more efficient in the long run:

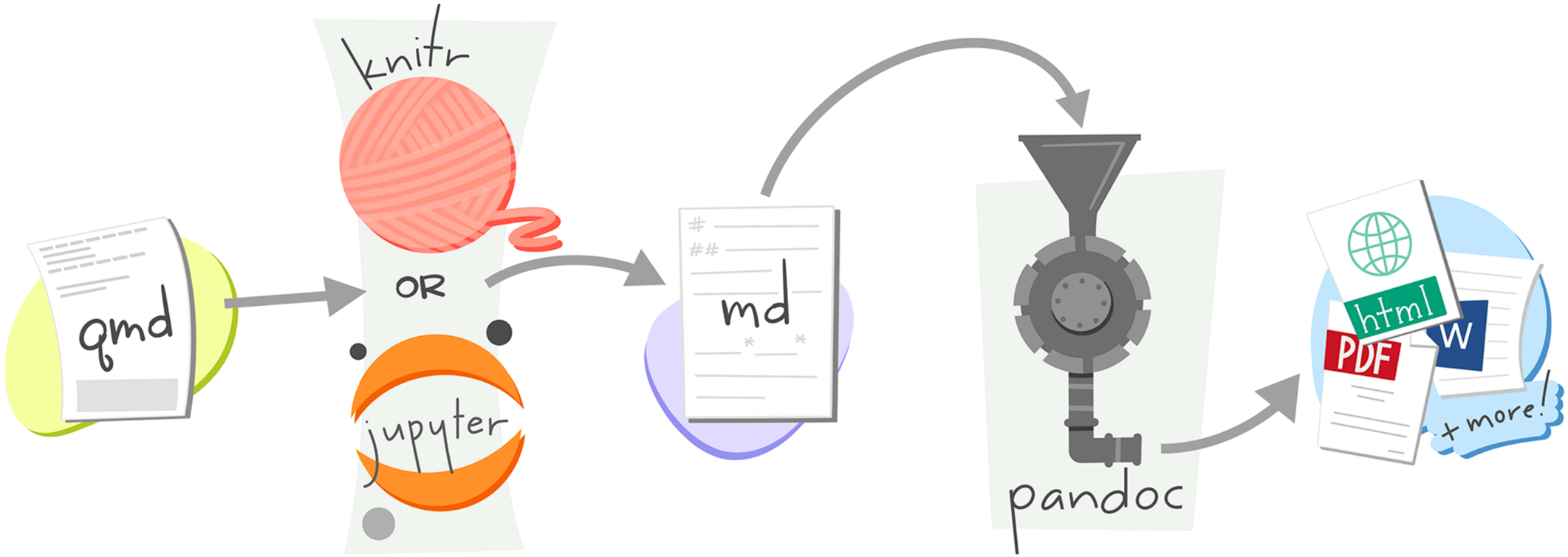
- When data / code are updated, you press one button to rerun your complete analysis and update your report (no more copy-pasting!)

Why quarto®?

Quarto (R/Python/Julia/Observable to …) is more flexible than alternative solutions:

- RMarkdown - R to …
- Jupyter - Python to …
- StatTag - Stata/SAS/R/Python to Word
- tidystats - R to Word/Google Docs

How does quarto work?

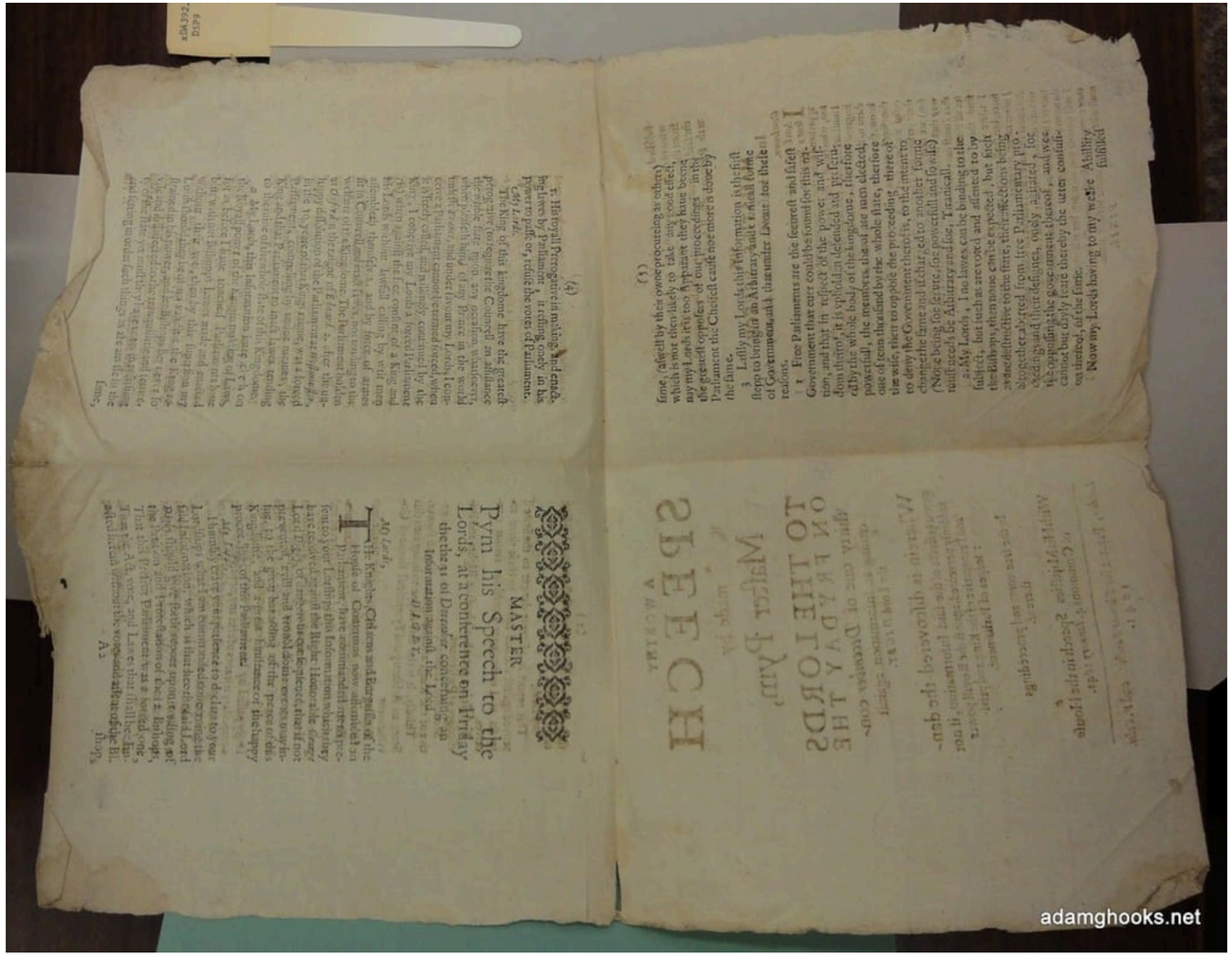


Artwork from “Hello, Quarto” keynote by Julia Lowndes and Mine Çetinkaya-Rundel, presented at RStudio Conference 2022.

Illustrated by Allison Horst.

Why is it called quarto®?

The ‘quarto’ book format



Objectives of this workshop

- Learn what Quarto is and what can you use it for
- Learn how to combine text, code, and code outputs together to create a fully reproducible report
- Learn where to find additional resources and more advanced documentation

Who are we?

Eline Van Geert

Postdoc researcher

Brain & Cognition

PPW, KU Leuven

elinevg

evg.ulyssis.be/evg

Lisa Koßmann

PhD student

Brain & Cognition

PPW, KU Leuven

lisa-kossmann

lisa-kossmann.github.io

Create your first quarto® document

Create a .qmd file

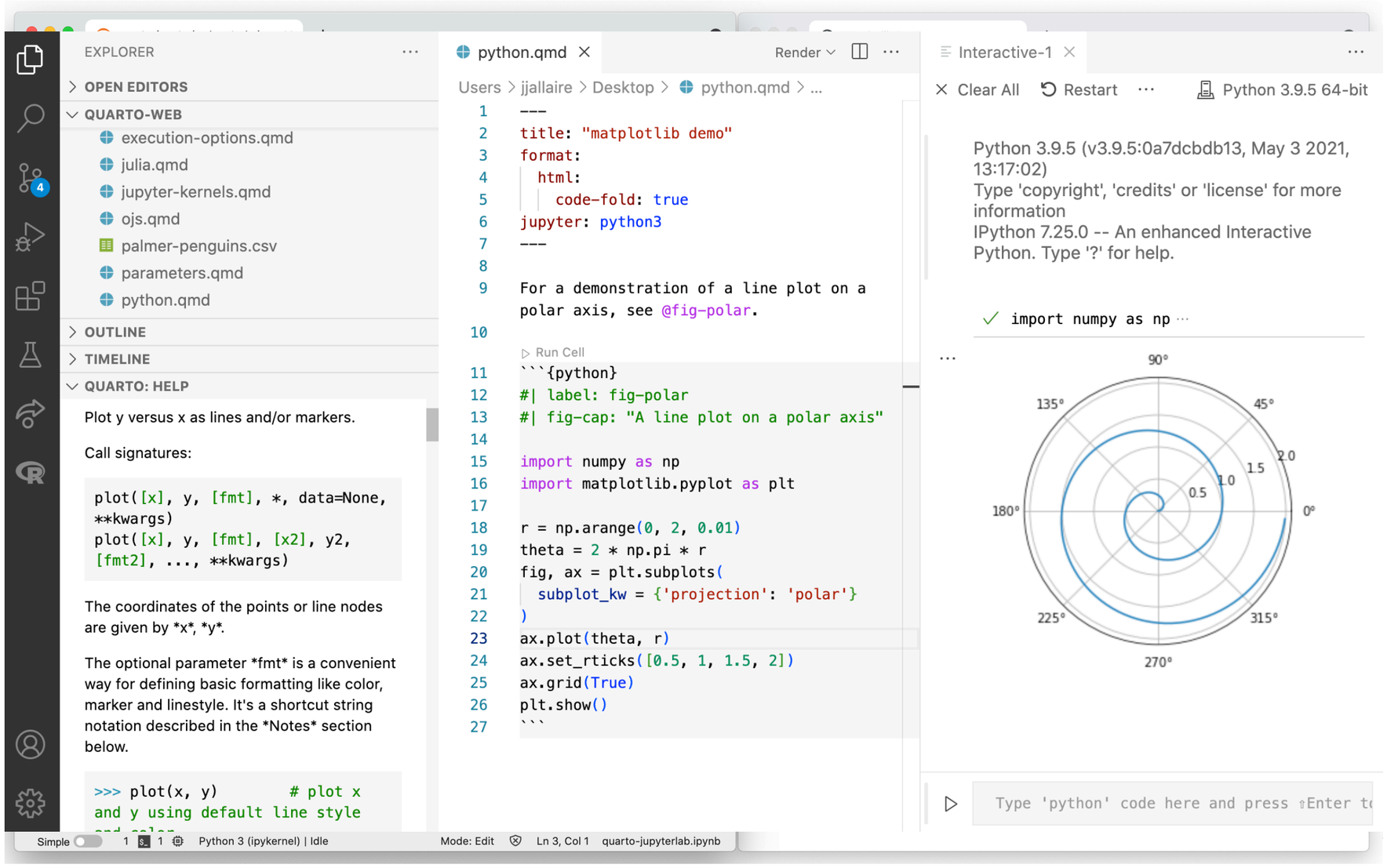
In your editor, create a .qmd file:

- RStudio: `File -> New file -> Quarto document`
- VS Code: `File -> New file -> Quarto document`
- Jupyter: `File -> New -> Notebook` (creates a .ipynb)

Anatomy of a Quarto document

- Metadata (YAML)
- Code (R / Python)
- Text (Markdown)

Use your preferred editor



Add a YAML header

```
---
key: value
---
```

Add a YAML header

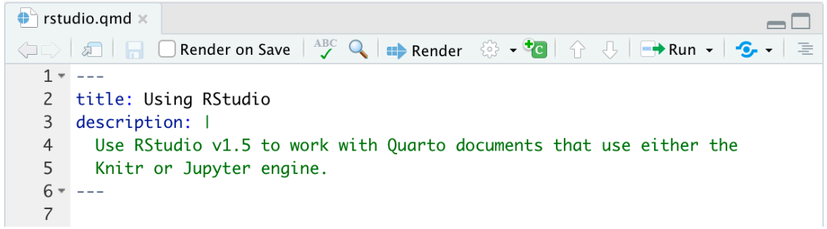
```yaml
---
title: "Your Document"
author: "Your name"
date: today
format:
  html:
    code-fold: show
    embed-resources: true
execute:
  warning: false
---
```

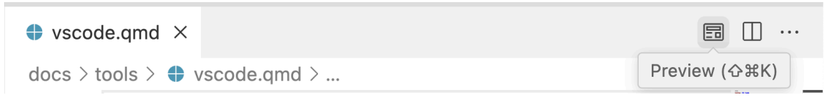
Tip for efficiency

RStudio + VSCode provide rich tab-completion - start a word and tab to complete, or Ctrl + space to see all available options.

Render your document to HTML/PDF/DOCX

1. Render button in RStudio / Preview button in VS Code





Render your document to HTML/PDF/DOCX

2. System shell via `quarto render`

terminal

```
quarto render document.qmd # defaults to html
quarto render document.qmd --to pdf
quarto render document.qmd --to docx
```

3. R console via `quarto` R package

```
library(quarto)
quarto_render("document.qmd") # defaults to html
quarto_render("document.qmd", output_format = "pdf")
```

⚠ Warning

In order to create PDFs you will need to install a recent distribution of TeX. We recommend the use of TinyTeX (which is based on TexLive), which you can install with the following command (in the Terminal):

```
quarto install tinytex
```

Add plain text with Markdown formatting

```
### Add a heading in your document

This is a sentence with some **bold text**, *italic text*,
`code`, and a [link](https://quarto.org/).
```

Add a heading in your document

This is a sentence with some **bold text**, *italic text*, `code`, and a link.

More info: https://quarto.org/docs/authoring/markdown-basics.html

Tip for Markdown newbies

New to Markdown? Use the visual editor in RStudio or VS Code!

Add plain text with Markdown formatting

Markdown Syntax	Output
`*italics* and **bold**`	*italics* and **bold**
`superscript^2^ / subscript~2~`	superscript^2 / subscript$_2$
`~~strikethrough~~`	~~strikethrough~~
`` `verbatim code` ``	`verbatim code`

Add plain text with Markdown formatting

Markdown Syntax	Output
`# Header 1`	Header 1
`## Header 2`	Header 2
`### Header 3`	Header 3
`#### Header 4`	Header 4
`##### Header 5`	Header 5
`###### Header 6`	Header 6

Add images

```
1 ![Image caption](figs/quartologo.png){width='20%' fig-align='left'}
```

Image caption

Add equations

```
Formula for population mean:

$$
\mu = \frac{\sum x}{N}
$$
```

Formula for population mean:

$$\mu = \frac{\sum x}{N}$$

Add R code chunks (and code outputs)

````
```{r}
#| label: fig-scatterplot
#| fig-cap: "Scatterplot of flipper and bill lengths"

library(palmerpenguins) # for data
library(tidyverse)      # for data wrangling and visualizatic
library(knitr)          # for tables

ggplot(data = penguins,
       aes(x = flipper_length_mm,
           y = bill_length_mm)) +
  geom_point(aes(color = species,
                 shape = species))
```
````

- Has 3x backticks on each end ```
- Place engine (`r`) between curly braces `{r}`
- Place options underneath, behind the `#|` (hashpipe): `#| option1: value`

Tip for efficiency

Use a keyboard shortcut to create a new code chunk!

RStudio: Ctrl + Alt + I (OS X: Cmd + Option + I)
VS Code: Ctrl + Shift + I

Options for R code chunks: https://quarto.org/docs/reference/cells/cells-knitr.html

Add R code chunks (and code outputs)

````
```{r}
#| output-location: column
#| label: fig-scatterplot
#| fig-cap: "Scatterplot of flipper and bill lengths"
#| warning: false

library(palmerpenguins) # for data
library(ggplot2)        # for data visualization

ggplot(data = penguins,
       aes(x = flipper_length_mm,
           y = bill_length_mm)) +
  geom_point(aes(color = species,
                 shape = species))
```
````

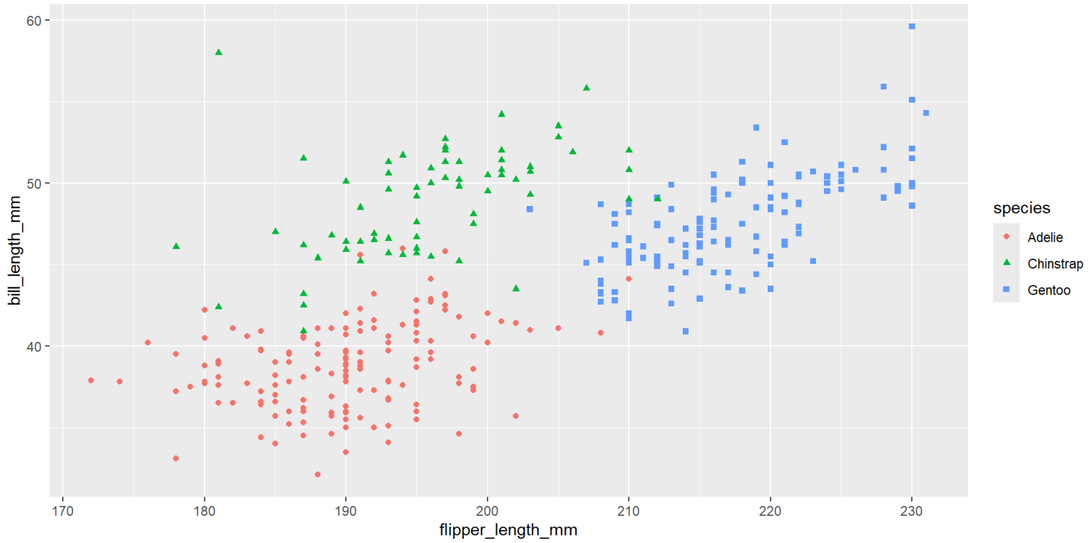


Figure 1: Scatterplot of flipper and bill lengths

Options for R code chunks: https://quarto.org/docs/reference/cells/cells-knitr.html

Add Python code chunks (and code outputs)

```
```{python}
#| output-location: column
#| label: fig-scatterplot-py
#| fig-cap: "Scatterplot of flipper and bill lengths in Pytho

import numpy as np
import matplotlib.pyplot as plt
from palmerpenguins import load_penguins

penguins = load_penguins()

penguins['species_color'] = penguins['species']
penguins['species_color'].replace(['Adelie', 'Chinstrap', 'Ge
                                   ['red', 'green', 'blue'], inplace=Tru

penguins.plot.scatter(x='flipper_length_mm',
                      y='bill_length_mm',
                      c='species_color')
```

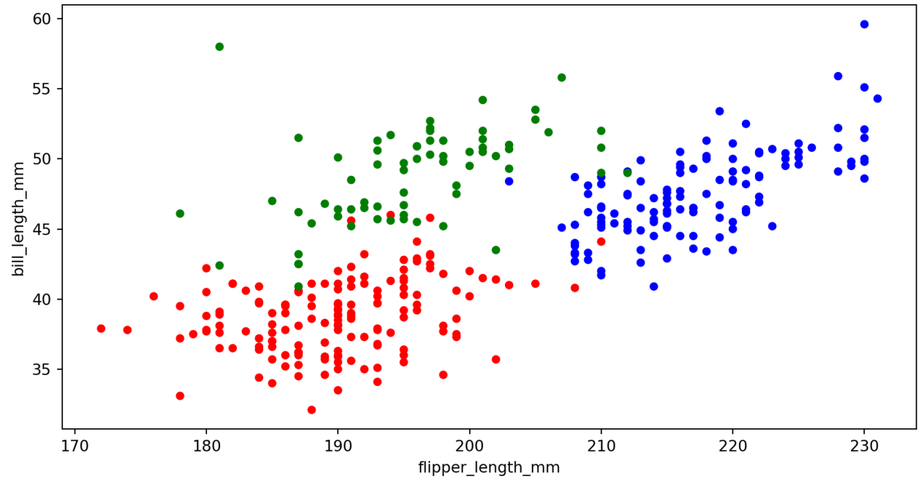


Figure 2: Scatterplot of flipper and bill lengths in Python

Options for Python code chunks: https://quarto.org/docs/reference/cells/cells-jupyter.html
```

Add Markdown table

```
| fruit   | price |
|---------|-------|
| apple   | 2.05  |
| pear    | 1.37  |
| orange  | 3.09  |

: Fruit prices {.striped .hover}
```

Fruit prices

fruit	price
apple	2.05
pear	1.37
orange	3.09

More information: https://quarto.org/docs/authoring/tables.html

Add R table

```r
```{r}
#| output-location: column
#| label: tbl-stats
#| tbl-cap: "Summary statistics for flipper and bill lengths"

penguins %>%
  group_by(species) %>%
  summarise(
    `Mean bill length` = mean(bill_length_mm, na.rm = T),
    `Mean flipper length` = mean(flipper_length_mm, na.rm = T
    `Correlation, r` = cor(flipper_length_mm, bill_length_mm,
    ) %>%
  kable(digits = c(2, 2, 2, 2, 2))
```
```

Table 1: Summary statistics for flipper and bill lengths

species	Mean bill length	Mean flipper length	Correlation, r
Adelie	38.79	189.95	0.33
Chinstrap	48.83	195.82	0.47
Gentoo	47.50	217.19	0.66

More information: https://quarto.org/docs/authoring/tables.html

Add Python table

```
```{python}
#| output-location: column
#| label: tbl-py
#| tbl-cap: "First rows of penguins dataframe"

from tabulate import tabulate
from IPython.display import Markdown

# Convert to markdown table
Markdown(tabulate(penguins[["species", "island", "bill_length
```
```

Table 2: First rows of penguins dataframe

	species	island	bill_length_mm	fli
0	Adelie	Torgersen	39.1	18
1	Adelie	Torgersen	39.5	18
2	Adelie	Torgersen	40.3	19
3	Adelie	Torgersen	nan	na
4	Adelie	Torgersen	36.7	19

More information: https://quarto.org/docs/authoring/tables.html

Add inline R or Python code

```
The palmerpenguins package contains data for ` {r} nrow(penguins) ` penguins.
Remove the space before {r} to make sure the code is evaluated!
```

The palmerpenguins package contains data for 344 penguins.

```
The palmerpenguins package contains data for `{python} penguins.shape[0]` penguins.
```

> **⚠ Warning**
>
> Inline code only works for the chosen engine (knitr: R; jupyter: Python). Specify the engine explicitly in the YAML header.

Add cross-references

```
1 ![The Quarto logo](figs/quartologo.png){#fig-quarto width='20%' fig-align='left'}
2
3 See @fig-quarto for the Quarto logo.
```

Figure 3: The Quarto logo

See Figure 3 for the Quarto logo.

- Figures (#fig-)
- Tables (#tbl-)
- Sections (#sec-)
- Equations (#eq-)
- and more

Add citations

- Add citation in text: @… (manually or using visual editor > Insert citation)

```
The palmerpenguins package was developed by @horst2020. We will create a document using Quarto [@quarto] and R [@R] or Python [(
```

- Specify .bib file in YAML

```
bibliography: references.bib
```

Add tabsets

```
::: {.panel-tabset}

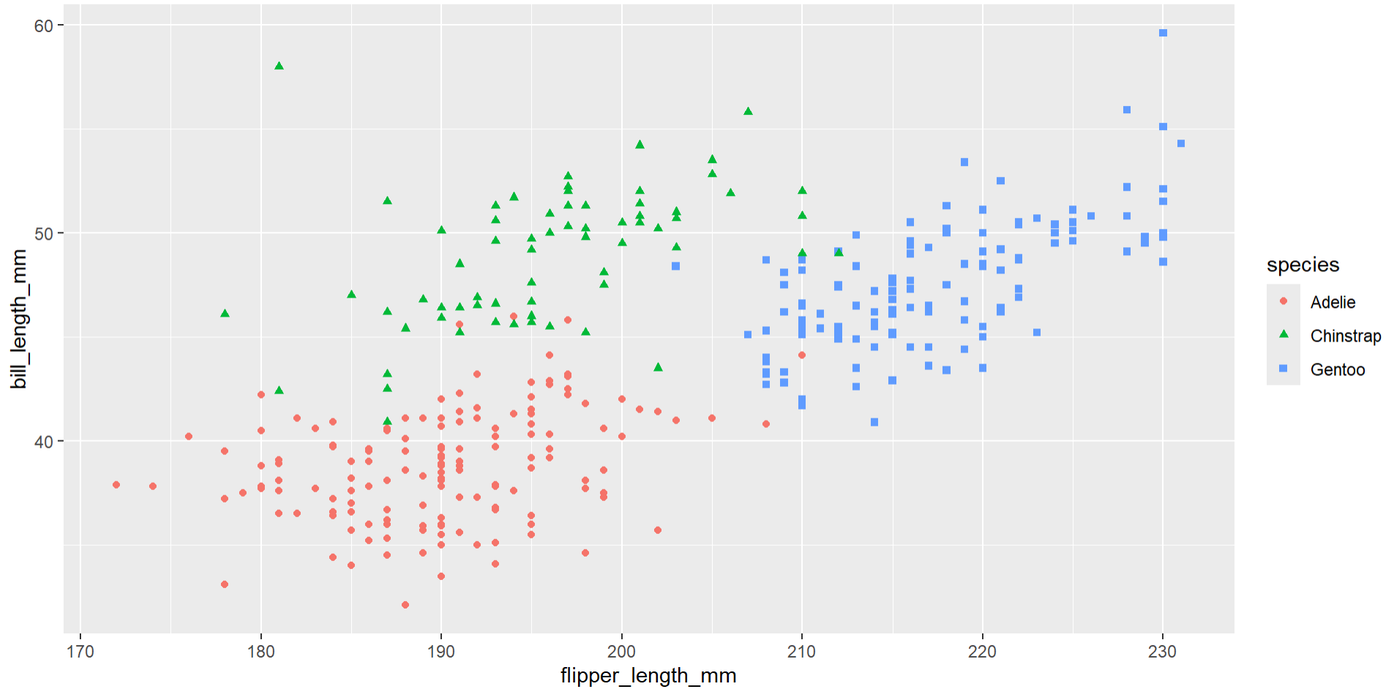
## Element 1

## Element 2

:::
```

Add tabsets

R figure | Python figure



Scatterplot of flipper and bill lengths in R

Add footnotes

```
This sentence ends with a footnote.[^1]

[^1]: This is an example footnote.
```

This sentence ends with a footnote.[1]

Bonus features

- interactive outputs (only for HTML outputs):
  - `htmlwidgets` for R
  - `Jupyter widgets` for Python/Julia
  - `Observable JS`
  - `Shiny` for R and Python
- callouts
- videos
- diagrams

Bonus features

- code annotation
- article layout
- Extensions:
  - shortcodes and filters (e.g., fontawesome icons 🎓)
  - journal article formats
  - custom formats
- and much more! https://quarto.org/docs/guide/

Observable JS

```
997  viewof temp = Inputs.range([0, 100], {step: 1, value: 34, label: htl.html`Temp &#x2103;`})
```

Error
This document uses OJS, which requires JavaScript features disabled when running in file://... URLs. In order for these features to work, run this document in a web server.

Converting temperature from ℃ to ℉

Celsius =

Error
This document uses OJS, which requires JavaScript features disabled when running in file://... URLs. In order for these features to work, run this document in a web server.

℃ and Fahrenheit =

Error

This document uses OJS, which requires JavaScript features disabled when running in file://... URLs. In order for these features to work, run this document in a web server.

°F.

Generated using this code chunk, text, and inline code:

```
```{ojs}
viewof temp = Inputs.range([0, 100], {step: 1, value: 34, label: htl.html`Temp &#x2103;`})
```

Converting temperature from ℃ to ℉
Celsius = ${d3.format(".0f")(temp)}℃ and Fahrenheit = ${d3.format(".1f")(temp * 9/5 + 32)}℉.
```

Other output formats

- Presentations (Revealjs / Beamer / PowerPoint)
- Dashboards
- Websites
- Books
- Manuscripts
- …

Examples of advanced use cases

- Your poster using Quarto: posterdown, typst-poster, quarto_poster, or docposter
- Your PhD thesis using Quarto: PhD thesis Eline; other Quarto books
- Your journal articles using Quarto
  (e.g. using apaquarto, other formats, or Quarto Manuscripts)
- Parametrized reports using Quarto: blog on parametrized reports; Quarto documentation on parameters
- Your website using Quarto: personal website; documentation website; other website projects
- Your dashboards or interactive documents using Quarto

Awesome Quarto resources (only a selection!)

- Quarto cheatsheet
- Awesome Quarto list
- Quarto for Scientists
- Quarto FAQ
- Quarto questions

Bonus: What to do with my existing `.Rmd` or `.ipynb`?

For some of you - nothing changes! Keep using RMarkdown and Jupyter.

However, most existing `.rmd` or `.ipynb` can be rendered as-is via Quarto

terminal

```
quarto render my-favorite.rmd --to html
```

Since Jupyter notebooks can either be treated as a linear document to be re-executed or an already evaluated document there are additional options like: `--execute`

terminal

```
quarto render my-favorite.ipynb --to html --execute
```

Bonus: Why Quarto, instead of RMarkdown

- Batteries included, shared syntax
- Choose your own editor and your preferred data science language
- Better accessibility and richer features out of the box
- More enhancements overtime - RMarkdown still maintained, but majority of new features built into Quarto

Collaboration with other colleagues in other languages - shared format, choose your editor and your native language

Quarto, crafted with love and care

> Development of Quarto is sponsored by RStudio, PBC. The same core team works on both Quarto and R Markdown:

- Carlos Scheidegger (@cscheid)
- Charles Teague (@dragonstyle)
- Christophe Dervieux (@cderv)
- J.J. Allaire (@jjallaire)
- Yihui Xie (@yihui)

Here is the full contributors list. Quarto is open source and they welcome contributions in their github repository as well! https://github.com/quarto-dev/quarto-cli.

Attributions

The slides and materials for this workshop were heavily based on other existing guides and workshops:

- Quarto official documentation
- Getting Started with Quarto by Tom Mock (CC BY 4.0)
- Quarto workshop by Julien Barnier and Aurélie Siberchicot

Icon attributions:

- to do by Michael Appleford from Noun Project (CC BY 3.0)

Thank you very much for providing these open resources!

Feedback, further questions or want to connect?

Eline Van Geert

Postdoc researcher

Brain & Cognition

PPW, KU Leuven

elinevg.bsky.social

evg.ulyssis.be/evg

eline.vangeert@kuleuven.be

Lisa Koßmann

PhD student

Brain & Cognition

PPW, KU Leuven

lisa-kossmann.bsky.social

lisa-kossmann.github.io

lisa.kossmann@kuleuven.be