Reproducible documents with Quarto

Day 3 - Introduction to Data Analysis with R

Selina Baldauf
Freie Universität Berlin - Theoretical Ecology

March 13, 2026

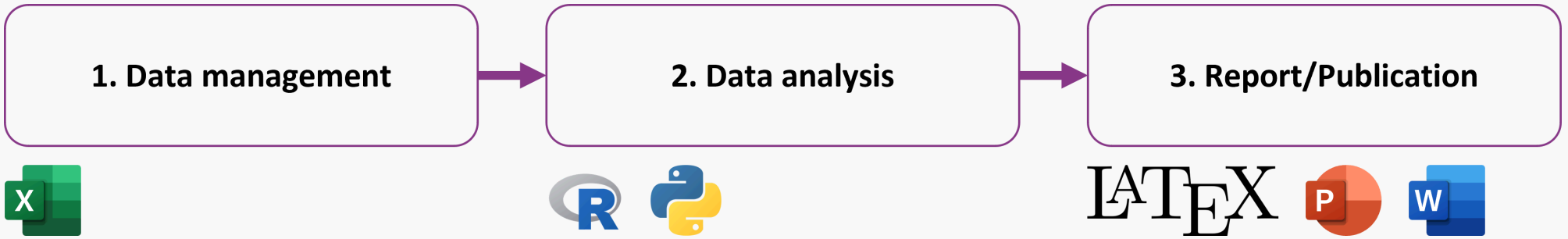
From R scripts to documents

So far, you've been writing `.R` scripts.

This is great for running code, but what if you want to

- share your analysis with collaborators?
- add explanations and interpretations to your code?
- create a report with figures and tables?

A standard workflow

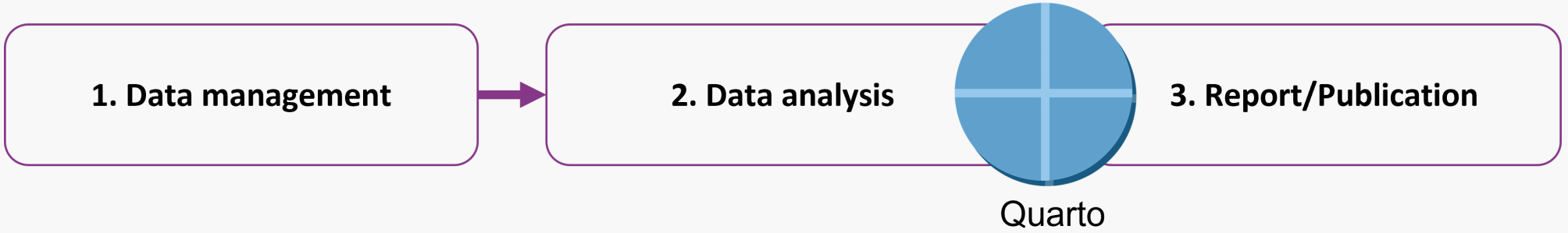


If you have to repeat the analysis

- Redo all figures and tables
- Update document manually
- Manual copy pasting of values is very error prone

A Quarto workflow

Quarto lets you combine **code**, **text**, and **output** in one document.



Advantages:

- Easy to redo the analysis
- No more copy pasting
- Reproducible
- Documentation, code & output in one place

What is Quarto?

> Quarto is an open-source scientific and technical publishing system

- Built into RStudio but can also be installed separately
- Create different types of outputs:
  - Documents: HTML, PDF, Word
  - Presentations, Websites, Books, ...

Today we focus on **documents**.

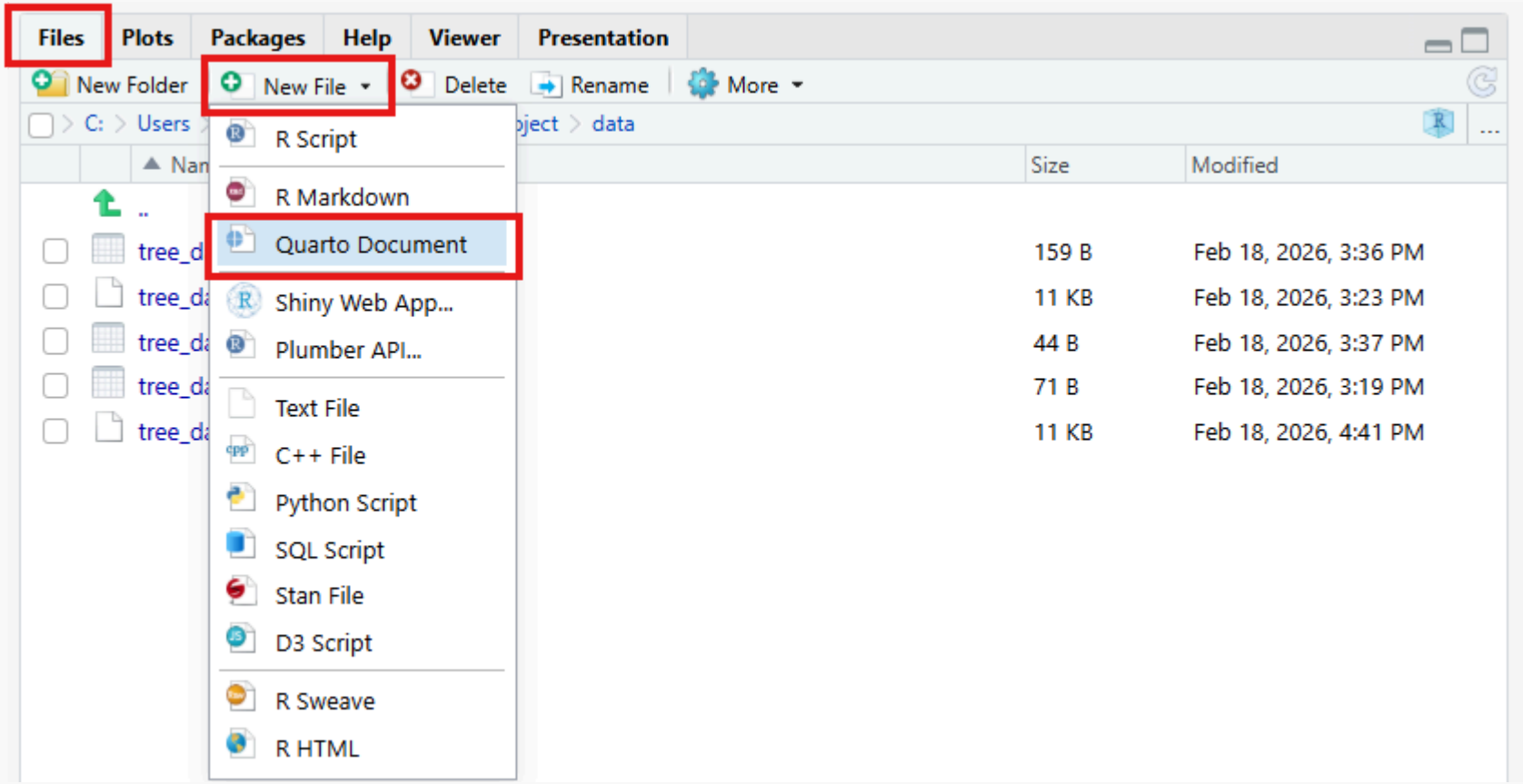
The basic Quarto workflow

Three steps

1. **Create** a `.qmd` document
2. **Write** text and code into the document
3. **Render** the document to an output format (e.g. HTML)

Step 1: Create a `.qmd` document

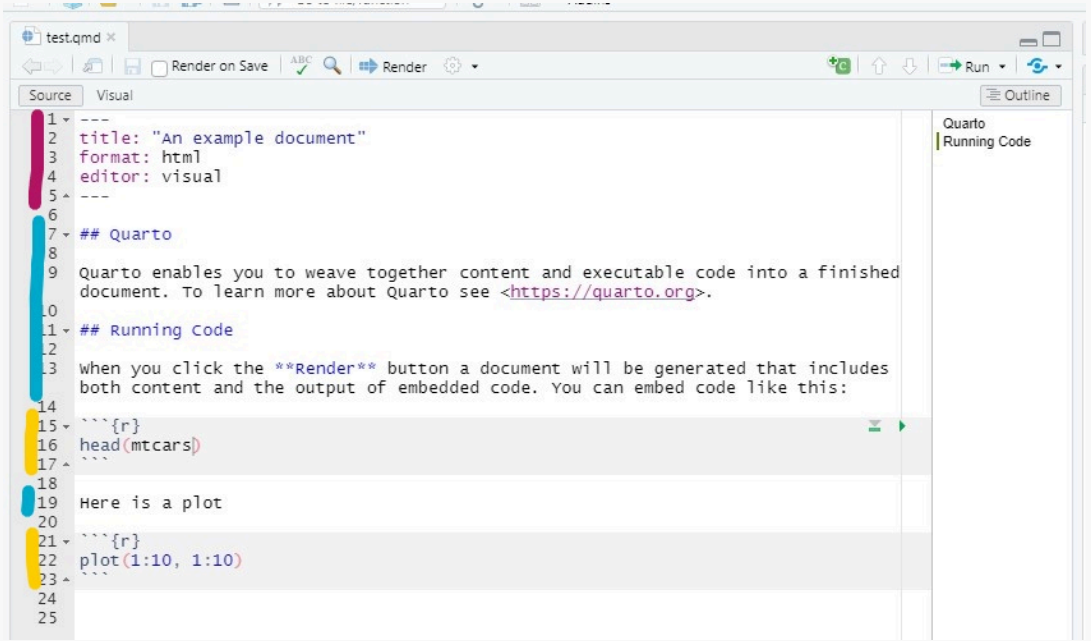
In the **Files pane**, click `New File -> Quarto Document`



Give it a name (e.g. `my_report.qmd`) and it opens as an empty file in RStudio.

Step 2: Write the document

A `.qmd` file has three types of content:



- YAML header: Metadata and output format
- Markdown text: Formatted text body
- Code chunks: R code that produces output

Step 3: Render the document

- Click the **Render** button in RStudio
- Or use the keyboard shortcut `Ctrl + Shift + K`

RStudio will run all the code, combine it with your text, and produce the output document.

Elements of a `.qmd` document

Markdown text, Code, YAML header

Markdown text

Markdown is a simple markup language to create formatted text.

The basics

- Bold: `**text**` becomes **text**
- Italic: `*text*` becomes *text*

Headers

```
# First level header
## Second level header
### Third level header
```

Markdown text

Lists

```
- item 1
- item 2
- item 3
```

Links

`[Quarto website](https://quarto.org)` becomes Quarto website

You don't need to memorize all of this. Here is a quick reference.

Code chunks

Code chunks start and end with 3 backticks and contain R code:

````
```{r}
library(gapminder)

ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
  geom_point()
```
````

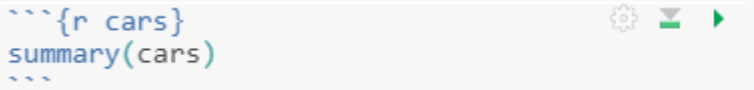
Insert a code chunk

- Menu: `Code` -> `Insert chunk`
- Keyboard shortcut: `Ctrl + Alt + I` / `Cmd + Option + I`

Code chunks

Run a code chunk

- Code chunks are run when you **render** the document
- You can also run them like normal R code by clicking the green arrow



Code chunks — options

Code chunks have special comments starting with `#|` that control the output:

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

ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
  geom_point()
```
````

- `echo: true/false` — Show the code in the output?
- `eval: true/false` — Run the code?
- `warning: true/false` — Show warnings?
- `message: true/false` — Show messages?

Inline code

You can also include R code **inside text** using inline code:

```
The gapminder data contains `r nrow(gapminder)` observations.
```

becomes:

The gapminder data contains 1704 observations.

This is powerful because the **number updates automatically** when the data changes.

YAML header — metadata

The YAML header is at the top of the document between `---` markers:

```yaml
---
title: "My analysis"
author: "Selina Baldauf"
date: today
format: html
---
```

This sets the **title**, **author**, **date**, and **output format**.

YAML header — execute options

You can set **default options** for all code chunks:

```yaml
---
title: "My analysis"
author: "Selina Baldauf"
format: html
execute:
  warning: false
  message: false
---
```

These can be overwritten by individual chunk options.

Now you

Task (30 min)

Reproducible documents with Quarto

Find the task description here

Additional Quarto features

Render to PDF

Change the output format in the YAML header:

```
---
format: pdf
---
```

You might need to install LaTeX first. The most convenient is to use the `tinytex` package:

```
# Run this in the R console
# install.packages("tinytex")
tinytex::install_tinytex()
```

Render to Word

```yaml
---
format: docx
---
```

You can also specify **multiple output formats**:

```yaml
---
format:
  html: default
  pdf: default
  docx: default
---
```

Document options

Add options under the format to customize your document:

```yaml
---
format:
  html:
    toc: true
    toc-location: left
    number-sections: true
    code-fold: true
---
```

- `toc`: Add a table of contents
- `number-sections`: Number the section headers
- `code-fold`: Hide code behind a button (HTML only)

Be careful with the **indentation**, YAML is sensitive to spaces.

Figure options

Control how figures appear in the output:

````
```{r}
#| label: fig-life-exp
#| fig-cap: "Life expectancy vs. GDP per capita in 2007"
#| fig-align: center
#| out-width: "80%"

ggplot(gapminder, aes(x = gdpPercap, y = lifeExp)) +
  geom_point(aes(color = continent)) +
  theme_minimal()
```
````

- `fig-cap`: Figure caption
- `fig-align`: `left`, `center`, or `right`
- `out-width`: Width of the figure in the output
- `label`: Must start with `fig-` for figures

Cross-references

You can reference labeled figures in the text:

```
As we can see in @fig-life-exp, life expectancy increases with GDP.
```

becomes:

As we can see in Figure 1, life expectancy increases with GDP.

Nice tables with `kable()`

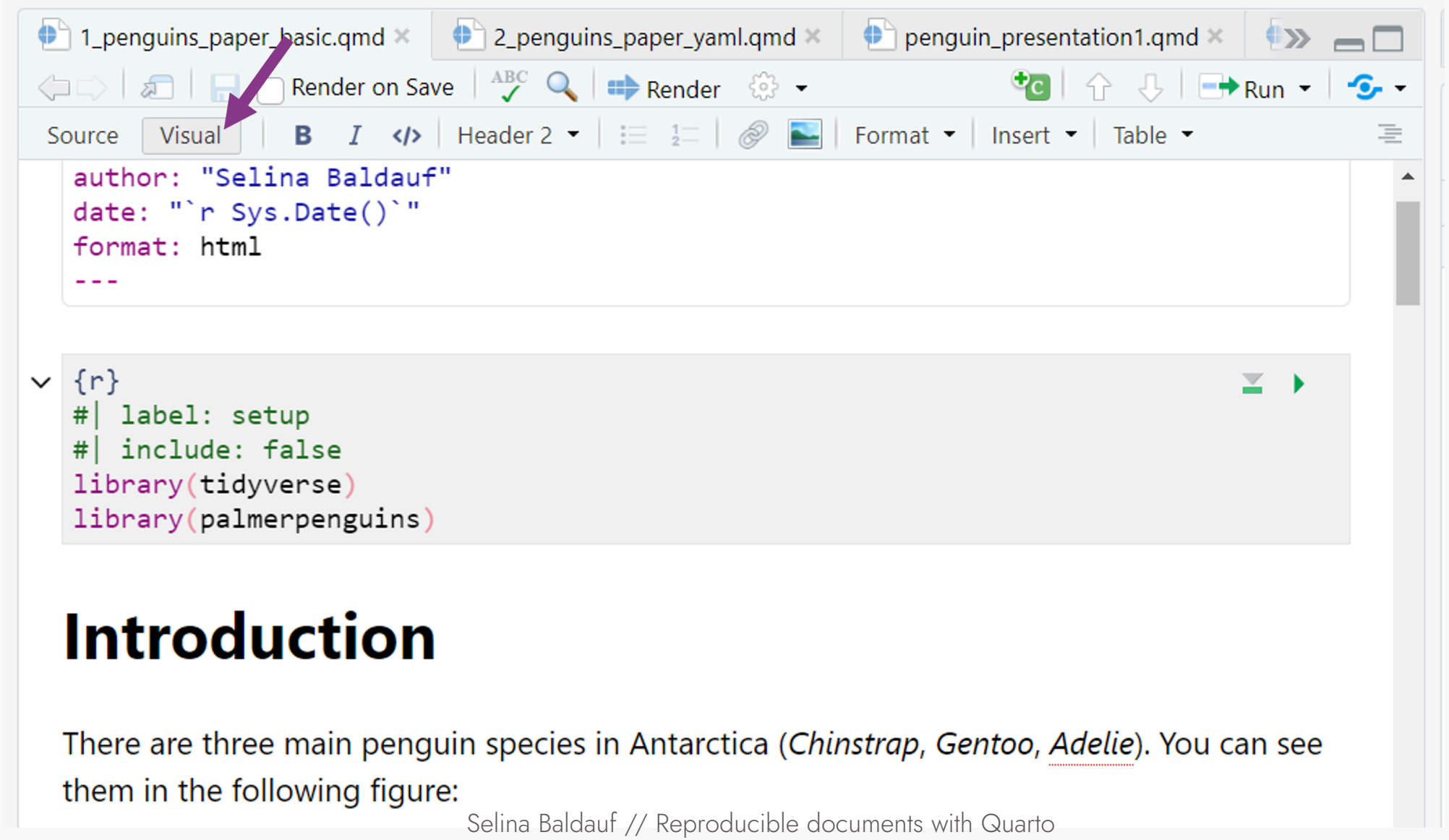
By default, tibbles print as plain text. Use `knitr::kable()` to render a nice table:

````
```{r}
gapminder |>
  filter(year == 2007) |>
  group_by(continent) |>
  summarize(
    mean_life_exp = mean(lifeExp),
    mean_gdp = mean(gdpPercap)
  ) |>
  knitr::kable()
```
````

continent	mean_life_exp	mean_gdp
Africa	54.80604	3089.033
Americas	73.60812	11003.032
Asia	70.72848	12473.027
Europe	77.64860	25054.482
Oceania	80.71950	29810.188

The visual editor

RStudio has a **visual editor** that provides a word-like interface for editing `.qmd` files.



The visual editor

The visual editor makes it easy to:

- Format text with buttons instead of markdown syntax
- Insert images, tables, and links via menus
- Add citations from Zotero, DOI, or PubMed

Outlook

Quarto can do much more:

- **Presentations** (like the slides in this workshop)
- **Websites** (like the workshop website)
- **Books**
- Publish online with Quarto Pub or GitHub Pages

Check out the Quarto website for guides, examples, and a gallery.

References

- Quarto website — everything you need to get started
- Markdown syntax reference
- HTML document options
- PDF document options
- Gallery with examples