

Erstellung von Berichten und Folien mit Quarto

rstatsZH - Data Science mit R

Lars Schöbitz

Nov 19, 2024

Lernziele (für diese Woche)

1. Die Lernenden können Querverweise zu Diagrammen und Tabellen in Quarto Dokumenten erstellen.
2. Die Lernenden können Folien im Format revealjs HTML erstellen.
3. Die Lernenden können Metadaten eines Quarto Dokuments anpassen.

Quarto

Quarto ...

- ist ein neues, open-source, wissenschaftliches und technisches Publikationssystem
- zielt darauf ab, den Prozess der Erstellung von Berichten und der Zusammenarbeit drastisch zu verbessern





Observable



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

Artikel & Berichte (HTML)

quarto-dev.github.io/quarto-gallery/articles/html/html.html

Quarto HTML Basics

Norah Jones

March 22nd, 2021

Code ▾

On this page

Introduction

Plot Output

Interactivity

Tables

LaTeX Math

Introduction

This a Quarto document. To learn more about Quarto see <https://quarto.org>.

Click the **Code** button in the header to see the full source code of this document.

Here we call the R `summary()` function—the function's output is included immediately below:

```
summary(cars)
```

	speed	dist
Min. :	4.0	Min. : 2.00

Artikel & Berichte (HTML)

Besuche [quarto.org!](https://quarto.org/)

quarto.org/docs/output-formats/html-basics.html

The screenshot shows a web browser window displaying the 'HTML Basics' guide from the quarto.org documentation. The page title is 'HTML Basics'. Below it is a section titled 'Overview'. A note says: 'Use the `html` format to create HTML output. For example:' followed by a code snippet:

```
---
```

```
title: "My document"
format:
  html:
    toc: true
```

Folien (HTML revealjs)

apreshill.github.io/palmerpenguins-useR-2022/#/title-slide

The untold story of palmerpenguins 🐧

Dr. Kristen Gorman, University of Alaska Fairbanks
apreshill.github.io/palmerpenguins-useR-2022/
Dr. Allison Horst, UC Santa Barbara

Folien (HTML revealjs)

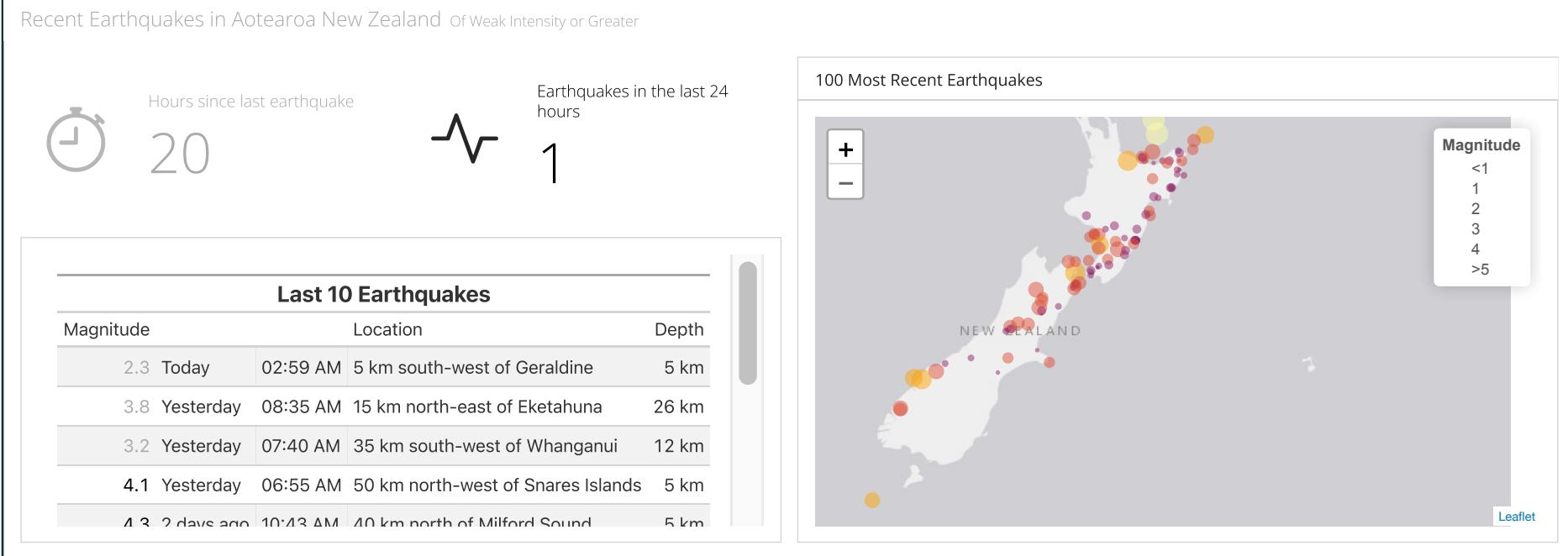
Besuche [quarto.org!](#)

quarto.org/docs/presentations/revealjs/

The screenshot shows a web browser displaying the Quarto Presentations website. The URL in the address bar is <https://quarto.org/docs/presentations/revealjs/>. The page title is "Revealjs". Below it, a section titled "Overview" is visible. A text block states: "You can create Revealjs presentations using the `revealjs` format. The best way to get a sense for the capabilities of Revealjs is this [demo presentation](#):". Below this text is a large, dark gray rectangular area containing the text "Quarto Presentations" in white. The top of the browser window shows the Quarto logo and social media links (Twitter, GitHub, LinkedIn), and a search bar.

Dashboards

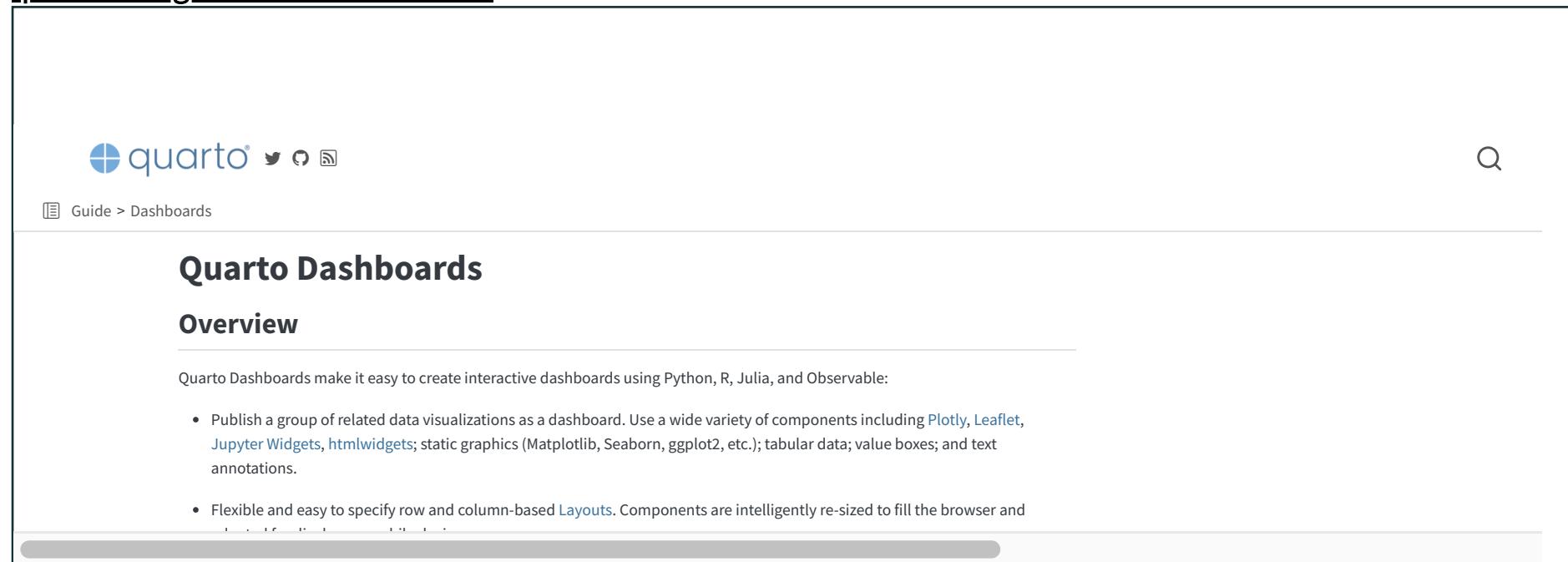
pub.demo.posit.team/public/nz-quakes/quakes.html



Dashboards

Besuche [quarto.org!](https://quarto.org/)

quarto.org/docs/dashboards/



The screenshot shows a web browser displaying the Quarto Dashboards documentation. The header includes the Quarto logo, social media links (Twitter, GitHub, RSS), and a search bar. The navigation bar shows 'Guide > Dashboards'. The main title is 'Quarto Dashboards' with a 'Overview' section below it. A text block explains that Quarto Dashboards make it easy to create interactive dashboards using Python, R, Julia, and Observable. It lists two bullet points: 1) Publishing a group of related data visualizations as a dashboard using various components like Plotly, Leaflet, Jupyter Widgets, and static graphics. 2) Flexibility in specifying row and column-based layouts. A horizontal progress bar is visible at the bottom of the page.

Gallery auf quarto.org

quarto.org/docs/gallery/

The screenshot shows a white page with a header containing the quarto logo and social media links. A search bar with a magnifying glass icon is on the right. Below the header, the word "Gallery" is in bold. A paragraph explains that Quarto can produce various output formats. A bulleted list follows, with the last item partially cut off.

Quarto can produce a wide variety of output formats. Here are some examples:

- [Articles & Reports](#)
- [Presentations](#)
- [Dashboards](#)
- [Websites](#)
- [Books](#)
- [Interactive Docs](#)

Anatomie eines Quarto-Dokuments

Komponenten

1. Metadaten: YAML
2. Text: Markdown
3. Code: Ausgeführt via `knitr` (R) oder `jupyter` (python)

Füge alles zusammen und du erhältst anschauliche, schöne und nützliche Ergebnisse!

Literate programming

DE: wortgetreue Programmierung

Bei der wortgetreuen Programmierung wird die Programmlogik in einer menschlichen Sprache mit eingeschlossenen Codeschnipseln geschrieben.

```
1 ---
2 title: "ggplot2 demo"
3 date: "19/11/2024"
4 format: html
5 ---
6
7 ## Pinguine
8
9 Es gibt einen linearen Zusammenhang zwischen Länge und Tiefe des Schnabels, abhängig von der Art.
10
11 ````{r}
12 library(ggplot2)
13 library(palmerpenguins)
14
15 ggplot(penguins, aes(x = bill_length_mm,
16                      y = bill_depth_mm,
17                      color = species)) +
18   geom_point()
19 ````
```

Source: https://en.wikipedia.org/wiki/Literate_programming

Metadaten

YAML

“Yet Another Markup Language” oder “YAML Ain’t Markup Language” wird verwendet, um Metadaten auf Dokumentebene bereitzustellen.

```
1  ---
2 key: value
3  ---
```

Output (Ausgabe) Optionen

```
1 ----  
2 format: etwas  
3 ----
```

```
1 ----  
2 format: html  
3 ----
```

```
1 ----  
2 format: pdf  
3 ----
```

```
1 ----  
2 format: revealjs  
3 ----
```

Output (Ausgabe) Optionen Argumente

Einrückung ist wichtig!

```
1  ---
2  format:
3    html:
4      toc: true
5      code-fold: true
6  ---
```

YAML Validierung

- Ungültig: Kein Leerzeichen nach :

```
1 ---  
2 format:html  
3 ---
```

- Ungültig: Als fehlend gelesen

```
1 ---  
2 format:  
3 html  
4 ---
```

YAML Validierung

Es gibt mehrere Möglichkeiten, gültiges YAML zu formatieren:

- Gültig: Es gibt ein Leerzeichen nach :

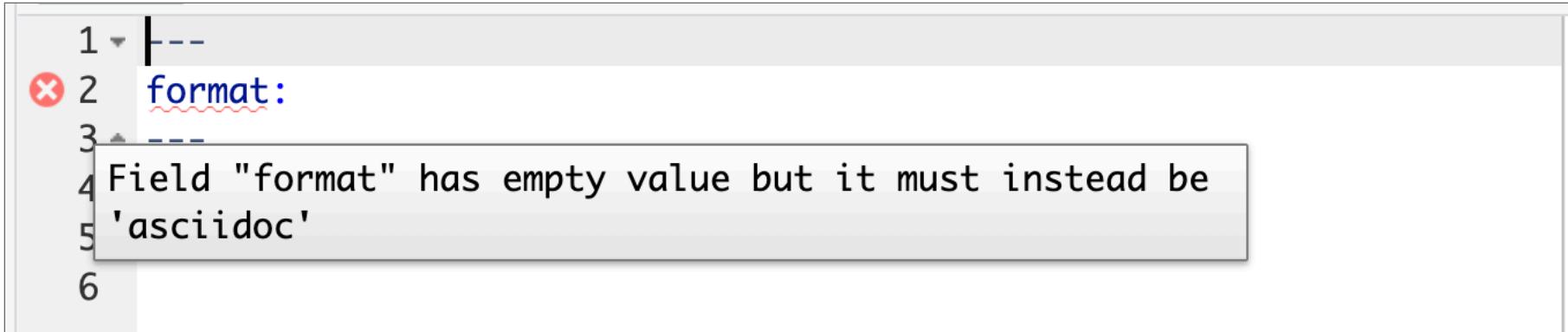
```
1 format: html
```

- Gültig: `format: html` mit Auswahlen, die mit korrekter Einrückung erfolgen

```
1 format:  
2   html:  
3     toc: true
```

Quarto linter

Lint oder ein Linter ist ein statisches Code-Analyse-Tool, das Programmierfehler, Bugs, stilistische Fehler und verdächtige Konstrukte aufzeigt.



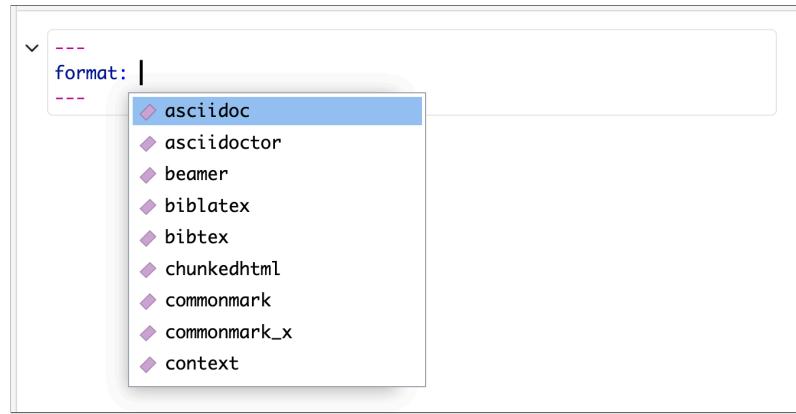
The screenshot shows a code editor window with a Quarto configuration file. The file contains the following code:

```
1 ---  
2 format:  
3 ---  
4 Field "format" has empty value but it must instead be  
5 'asciidoc'  
6
```

A red 'X' icon is positioned next to the word 'format' in line 2, indicating an error. A tooltip or callout box is displayed over the line containing the error message, highlighting the text 'Field "format" has empty value but it must instead be'. The rest of the file (lines 1, 3, 4, 5, and 6) is shown below the error message.

Quarto YAML Intelligenz

RStudio + VSCode bieten eine umfangreiche Tab-Vervollständigung – fange mit einem Wort an und tippe auf die Tab-Taste, um es zu vervollständigen, oder **Strg + Leertaste** um alle möglichen Optionen zu sehen.



Liste der gültigen YAML-Felder

- Viele YAML-Felder sind bei verschiedenen Ausgaben gleich
- Aber auch jeder Ausgabetyp hat einen eigenen Satz an gültigen YAML-Feldern und Optionen
- Komplette Liste: quarto.org/docs/reference/formats/html

Source: [https://en.wikipedia.org/wiki/Lint_\(software\)](https://en.wikipedia.org/wiki/Lint_(software))

Text

Text Formatierung

Markdown Syntax

Ausgabe

italics and **bold**

italics and **bold**

superscript² / subscript₂

superscript² / subscript₂

~~strikethrough~~

~~strikethrough~~

`verbatim code`

verbatim code

Headings (Überschriften)

Markdown Syntax

Ausgabe

```
# Header 1
```

Header 1

```
## Header 2
```

Header 2

```
### Header 3
```

Header 3

```
#### Header 4
```

HEADER 4

```
##### Header 5
```

Header 5

```
##### Header 6
```

Header 6

Links

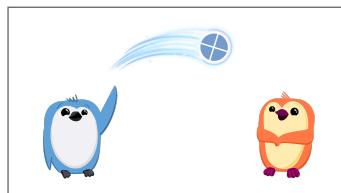
Es gibt verschiedene Arten von “Links” oder Hyperlinks.

Markdown

- ```
1 Du kannst [benannte Hyperlinks] (https://quarto.org/),
2 direkte URLs wie <https://quarto.org/> und Links zu
3 [anderen Orten] (#quarto-anatomy) in
4 das Dokument einbetten. Die Syntax für das Einbetten eines
5 Inline-Bildes ist ähnlich: ![Pinguine spielen mit Ball](img/md-09/penguins-quarto-ball.png)
```

## Ausgabe

Du kannst [benannte Hyperlinks]  
(https://quarto.org/), direkte URLs  
wie <https://quarto.org/> und Links zu  
anderen Orten in das Dokument  
einbetten. Die Syntax für das  
Einbetten eines Inline-Bildes ist  
ähnlich:



# Listen

Ungeordnete Liste:

**Markdown:**

```
1 - unordered list
2 - sub-item 1
3 - sub-item 1
4 - sub-sub-item 1
```

**Ausgabe**

- unordered list
  - sub-item 1
  - sub-item 1
    - sub-sub-item 1

Geordnete Liste:

**Markdown:**

```
1 1. ordered list
2 2. item 2
3 i. sub-item 1
4 A. sub-sub-item 1
```

**Ausgabe**

1. ordered list
2. item 2
  - i. sub-item 1
    - a. sub-sub-item 1

## Zitate

### Markdown:

```
1 > Let us change our traditional attitude to the construction of programs: Instead of imaginir
2 > - Donald Knuth, Literate Programming
```

### Output:

Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do. -  
Donald Knuth, Literate Programming

## Fussnoten

Die Nummerierung und Formatierung von Fussnoten wird unterstützt.

## Inline-Fussnoten

Hier ist eine Inline-Notiz.<sup>1</sup> [Inline-Notizen sind einfacher zu schreiben, da man nicht erst einen Bezeichner auswählen und nach unten gehen muss, um die Notiz zu schreiben.]

Hier ist eine Inline-Notiz.<sup>1</sup>

### Verwende den visuellen Editor-Modus (Visual editor)

Für das Hinzufügen von Fussnoten wird der visuelle Editor-Modus empfohlen.

## Ihr seid dran: [01-markdown-syntax.qmd](#)

1. Öffne [posit.cloud](#) in deinem Browser (verwende dein Lesezeichen).
2. Öffne den rstatszh-k009 Arbeitsbereich (Workspace) für den Kurs.
3. Klicke auf Start neben md-09-uebungen.
4. Suche im Dateimanager im Fenster unten rechts die Datei [01-markdown-syntax.qmd](#) und klicke darauf, um sie im Fenster oben links zu öffnen.
5. Verwende den Quelltexteditor (Source editor) Modus
6. Folge den Anweisungen in der Datei.

20:00

## Pause machen

Bitte steh auf und beweg dich. Lasst eure E-Mails in Frieden ruhen.



- 10:00
1. Inline-Notizen sind einfacher zu schreiben, da man nicht erst einen Bezeichner auswählen und nach unten gehen muss, um die Notiz zu schreiben

# Code

## Anatomie eines Code-Block

```
1 ````{r}
2 #| label: penguins
3 #| message: false
4
5 library(tidyverse)
6 library(palmerpenguins)
7 library(knitr)
8
9 penguins |>
10 count(species) |>
11 kable()
12 ````
```

| species   | n   |
|-----------|-----|
| Adelie    | 152 |
| Chinstrap | 68  |
| Gentoo    | 124 |

- Hat 3x Backticks an jedem Ende
- Engine ([r](#)) wird zwischen geschweiften Klammern [{r}](#) angegeben
- Optionen, die mit dem Zeichen [# |](#) angegeben werden (Hashpipe)

## Code, für wen ist es?

- Die Art und Weise, wie du Code anzeigst, ist in verschiedenen Kontexten sehr unterschiedlich.
- In einem Lehrszenario wie heute möchte ich *wirklich* Code anzeigen.
- In einem Unternehmen möchtest du vielleicht eine datenwissenschaftliche Ausgabe haben, die den Quellcode anzeigt, UND eine Ausgabe, die sich an die Nutzer:innen richtet und den Code verbirgt.

## Ein- und Ausblenden von Code mit echo

- Die Option `echo` zeigt den Code an, wenn sie auf `true` gesetzt ist, und verbirgt ihn, wenn sie auf `false` gesetzt ist.

## Tabellen und Diagramme

- In reproduzierbaren Berichten und Manuskripten sind **Tabellen** und **Figuren** die am häufigsten enthaltenen Codeausgaben.

# Tabellen

## Markdown Tabellen

Markdown:

| 1 | Right  | Left   | Default | Center |
|---|--------|--------|---------|--------|
| 2 | -----: | :----- | -----   | -----: |
| 3 | 12     | 12     | 12      | 12     |
| 4 | 123    | 123    | 123     | 123    |
| 5 | 1      | 1      | 1       | 1      |

Verwende den visuellen Editor-Modus (Visual editor)

Für das Hinzufügen von Fussnoten wird der visuelle Editor-Modus empfohlen.

Ausgabe:

| Right | Left | Default | Center |
|-------|------|---------|--------|
| 12    | 12   | 12      | 12     |
| 123   | 123  | 123     | 123    |
| 1     | 1    | 1       | 1      |

## Tabellen mit Code

Das Paket `knitr` kann Datenrahmen in Tabellen umwandeln mit `knitr::kable()`:

| species   | island    | n   |
|-----------|-----------|-----|
| Adelie    | Biscoe    | 44  |
| Adelie    | Dream     | 56  |
| Adelie    | Torgersen | 52  |
| Chinstrap | Dream     | 68  |
| Gentoo    | Biscoe    | 124 |

## Tabellen mit Code

Wenn du umfangreichere Tabellen möchtest, probiere das **gt**-Paket und [alles, was es bietet!](#)

| species          | island | bill_length_mm | bill_depth_mm | flipper_length_mm | body_mass_g | sex    | year |
|------------------|--------|----------------|---------------|-------------------|-------------|--------|------|
| Adelie Torgersen |        | 39.1           | 18.7          | 181               | 3750        | male   | 2007 |
| Adelie Torgersen |        | 39.5           | 17.4          | 186               | 3800        | female | 2007 |
| Adelie Torgersen |        | 40.3           | 18.0          | 195               | 3250        | female | 2007 |
| Adelie Torgersen |        | NA             | NA            | NA                | NA          | NA     | 2007 |
| Adelie Torgersen |        | 36.7           | 19.3          | 193               | 3450        | female | 2007 |
| Adelie Torgersen |        | 39.3           | 20.6          | 190               | 3650        | male   | 2007 |

## Abbildungen & Diagramme

## Markdown Abbildungen

![Pinguine spielen mit einem Quarto-Ball](img/md-09/penguins-quarto-ball.png)





Pinguine spielen mit einem Quarto-Ball

## Unterabbildungen

**Markdown:**

```
::: {#fig-penguins layout-ncol=2}
![Blauer Pinguin](img/md-09/blue-penguin.png){#fig-blue width="250px"}
![Orangefarbener Pinguin](img/md-09/orange-penguin.png){#fig-orange width="250px"}
```

Zwei Pinguine

:::

## Unterabbildungen

Ausgabe:



(a) Blauer Pinguin



(b) Orangefarbener Pinguin

Figure 1: Zwei Pinguine

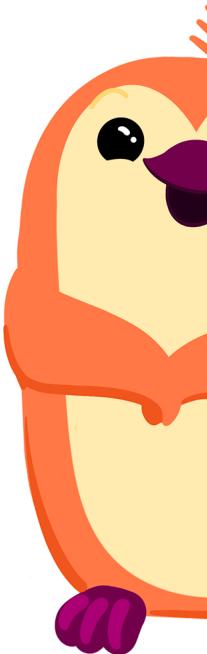
## Suche nach den Abbildungen

An Orten wie Markdown, YAML oder beim Einlesen von Dateien musst du **absolute** oder **relative** Dateipfade verwenden:

- Absolut = Schlecht: "`/Users/lars/uebungen`" - Auf wessen Computer wird das funktionieren?
- Relativ = Besser:
  - `../../` = ein Verzeichnis nach oben, `../..` = zwei Verzeichnisse nach oben, etc.
  - `/..` oder `/` = beginnen im "root"-Verzeichnis deines aktuellen Computers

## Abbildungen mit Code (Diagramme)

```
1 ````{r}
2 #| fig-width: 4
3 #| fig-align: right
4
5 knitr:::include_graphics("img/md-09/penguins-quarto-ball.png")
6 ````
```



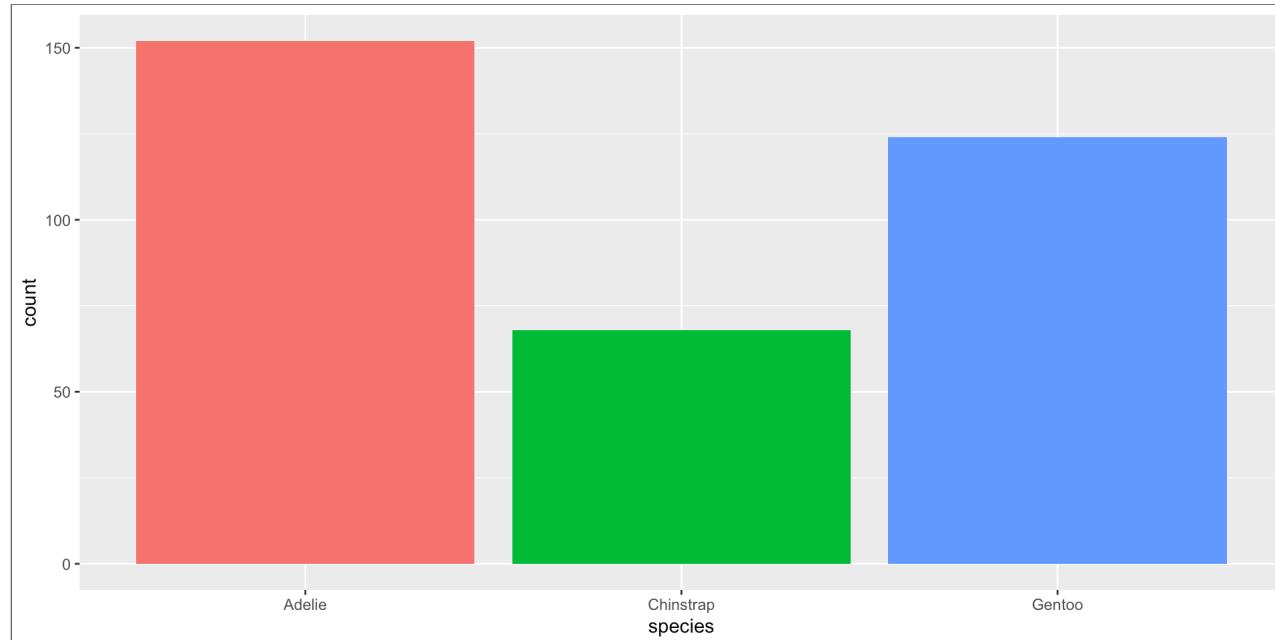
## Pfade in R-Code referenzieren

Verwende das `here`-Paket, um auf das Projektverzeichnis zu verweisen, da `here::here()` immer im obersten Verzeichnis einer `.Rproj` beginnt:

```
[1] "/Users/lschoebitz/Documents/gitrepos/gh-org-rstatszh-k009/website"
```

## Abbildungen mit Code (Diagramme)

```
1 ```{r}
2 ggplot(penguins, aes(x = species, fill = species)) +
3 geom_bar(show.legend = FALSE)
4 ```
```



## Querverweise (cross references)

## Querverweise (cross references)

- Unterstützen Leser:innen sich in deinem Dokument zurechtzufinden, indem du nummerierte Verweise und Hyperlinks zu Elementen wie Abbildungen und Tabellen einfügst.
- Querverweise Schritte:
  - Füge der Abbildung oder Tabelle eine Beschriftung hinzu: # | `tbl-cap:` oder # | `fig-cap:`.
  - Füge der Abbildung oder Tabelle eine Bezeichungen hinzu: # | `label:` beginnend mit `fig-` oder `tbl-`.
  - Verweise im Text auf die Abbildung oder Tabelle mit `@fig-....` oder `@tbl-....`

## Abbildung Querverweise

Das Vorhandensein der Beschriftung ([Blauer Pinguin](#)) und der Bezeichnung (#fig-blue-penguin) machen diese Abbildung referenzierbar:

### Markdown:

- ```
1 Siehe @fig-blue-penguin für einen süßen blauen Pinguin.  
2 ![Blauer Pinguin](img/md-09/blue-penguin.png){#fig-blue-penguin}
```

Ausgabe:

Siehe [Figure 2](#) für einen
süßen blauen Pinguin.



Figure 2: Blauer Pinguin

Tabelle Querverweise

Das Vorhandensein der Überschrift ([Ein paar Pinguine](#)) und der Bezeichnung (#tbl-pinguine) machen diese Tabelle referenzierbar:

Markdown:

```
1 Siehe @tbl-penguins für Daten über einige Pinguine.  
2  
3 ````{r}  
4 #| label: tbl-pinguine  
5 #| tbl-cap: Ein paar Pinguine  
6  
7 head(penguins) |>  
8   gt()  
9 ````
```

Ausgabe:

Siehe [Table 1](#) für Daten über einige Pinguine.

Table 1: Ein paar Pinguine

species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	
Adelie	Torgersen	39.1	18.7	181	3750	r
Adelie	Torgersen	39.5	17.4	186	3800	fe
Adelie	Torgersen	40.3	18.0	195	3250	fe
Adelie	Torgersen	NA	NA	NA	NA	
Adelie	Torgersen	36.7	19.3	193	3450	fe
Adelie	Torgersen	39.3	20.6	190	3650	r

Querverweise Schritte:

1. Füge der Abbildung oder Tabelle eine Beschriftung hinzu:

- `#|tbl-cap:` oder `#|fig-cap:`.

2. Füge der Abbildung oder Tabelle eine Bezeichungen hinzu:

- `#|label:` beginnend mit `fig-` oder `tbl-`.

3. Verweise im Text auf die Abbildung oder Tabelle mit:

- `@fig-...` oder `@tbl-....`

Ihr seid dran: [02-querverweise.qmd](#)

1. Öffne [posit.cloud](#) in deinem Browser (verwende dein Lesezeichen).
2. Öffne den rstatszh-k009 Arbeitsbereich (Workspace) für den Kurs.
3. Klicke auf Continue neben md-09-uebungen.
4. Suche im Dateimanager im Fenster unten rechts die Datei [02-querverweise.qmd](#) und klicke darauf, um sie im Fenster oben links zu öffnen.
5. Verwende den Quelltexteditor (Source editor) Modus
6. Folge den Anweisungen in der Datei.

20:00

Geht's auch auf Deutsch?

Ja!

```
1  ---
2  lang: de
3  ---
```

Pause machen

Bitte steh auf und beweg dich. Lasst eure E-Mails in Frieden ruhen.



10:00

Anatomie einer Quarto-Präsentation (Folien)

Komponenten

1. Metadaten: YAML
2. Text: Markdown
3. Code: Ausgeführt via `knitr` (R) oder `jupyter` (python)

Füge alles zusammen und du erhältst anschauliche, schöne und nützliche Folien!

Wir sind dran: [03-folien.qmd](#)

1. Öffne [posit.cloud](#) in deinem Browser (verwende dein Lesezeichen).
2. Öffne den rstatszh-k009 Arbeitsbereich (Workspace) für den Kurs.
3. Klicke auf Continue neben md-09-uebungen.
4. Suche im Dateimanager im Fenster unten rechts die Datei [03-folien.qmd](#) und klicke darauf, um sie im Fenster oben links zu öffnen.

30:00

Ihr seid dran: 03-folien.qmd

Mach da weiter, wo wir aufgehört haben, und:

- Ändere den Folien Übergang (transition style) zwischen den Folien (auf quarto.org: Suche nach “slide transitions” im Q Suchfeld oben rechts)
- Füge Folien Nummern (slide numbers) zu deinen Folien (auf quarto.org: Suche nach “slide numbers” im Q Suchfeld oben rechts)

10:00

Quarto Präsentation Formate

Quarto Präsentation Formate

- revealjs für HTML-Folien
 - Im Wesentlichen der Ersatz für `xaringan`, aber mit Pandoc-nativer Syntax
- beamer für LaTeX-Folien
- PowerPoint für den Fall, dass du über MS Office zusammenarbeiten *musst*

Besuche quarto.org

quarto.org/docs/presentations

The screenshot shows a web browser displaying the Quarto documentation for presentations. The page has a light gray header with the Quarto logo and social media links (Twitter, GitHub, RSS). A navigation bar below the header includes 'Guide', 'Presentations', and 'Overview'. The main content area features a large title 'Presentations' and a sub-section 'Overview'. A descriptive paragraph states: 'Quarto supports a variety of formats for creating presentations, including:' followed by a bulleted list: 'revealjs — reveal.js (HTML)', 'pptx — PowerPoint (MS Office)', and 'beamer — Beamer (LaTeX/PDF)'. The bottom of the page has a dark footer bar.

Zeitpuffer: Modul 9

Kann ich noch etwas zum heutigen Modul erklären?

05:00

Zusatzaufgaben Modul 9

Modul 9 Dokumentation

rstatszh-k009.github.io/website/module/md-09.html

Modul 9

Erstellung und Veröffentlichung analytischer Artikel mit Quarto und GitHub Pages

Lernziele

1. Die Lernenden können Literaturangaben zu Quarto-Dateien hinzufügen, indem sie das Navigationsmenü des visuellen Editors RStudio verwenden und eine Referenzsammlung im .bib Dateiformat erstellen.
2. Die Lernenden können Funktionen aus den Paketen knitr und gt anwenden, um Tabellen in Berichten im HTML Ausgabeformat darzustellen.
3. Die Lernenden können den GitHub Pages Dienst nutzen, um ein Repository als eigenständige Website zu veröffentlichen.

Folien

- In Vorbereitung

Lesematerial

Zusatzaufgaben Abgabedatum

- Abgabedatum: Montag, 25. November
- Korrektur- und Feedbackphase bis zu: Donnerstag, 28. November

Danke

Danke! ☀

Folien erstellt mit revealjs und Quarto: <https://quarto.org/docs/presentations/revealjs/>

Zugang zu Folien als [PDF auf GitHub](#)

Alle Materialien sind lizenziert unter [Creative Commons Attribution Share Alike 4.0 International](#).