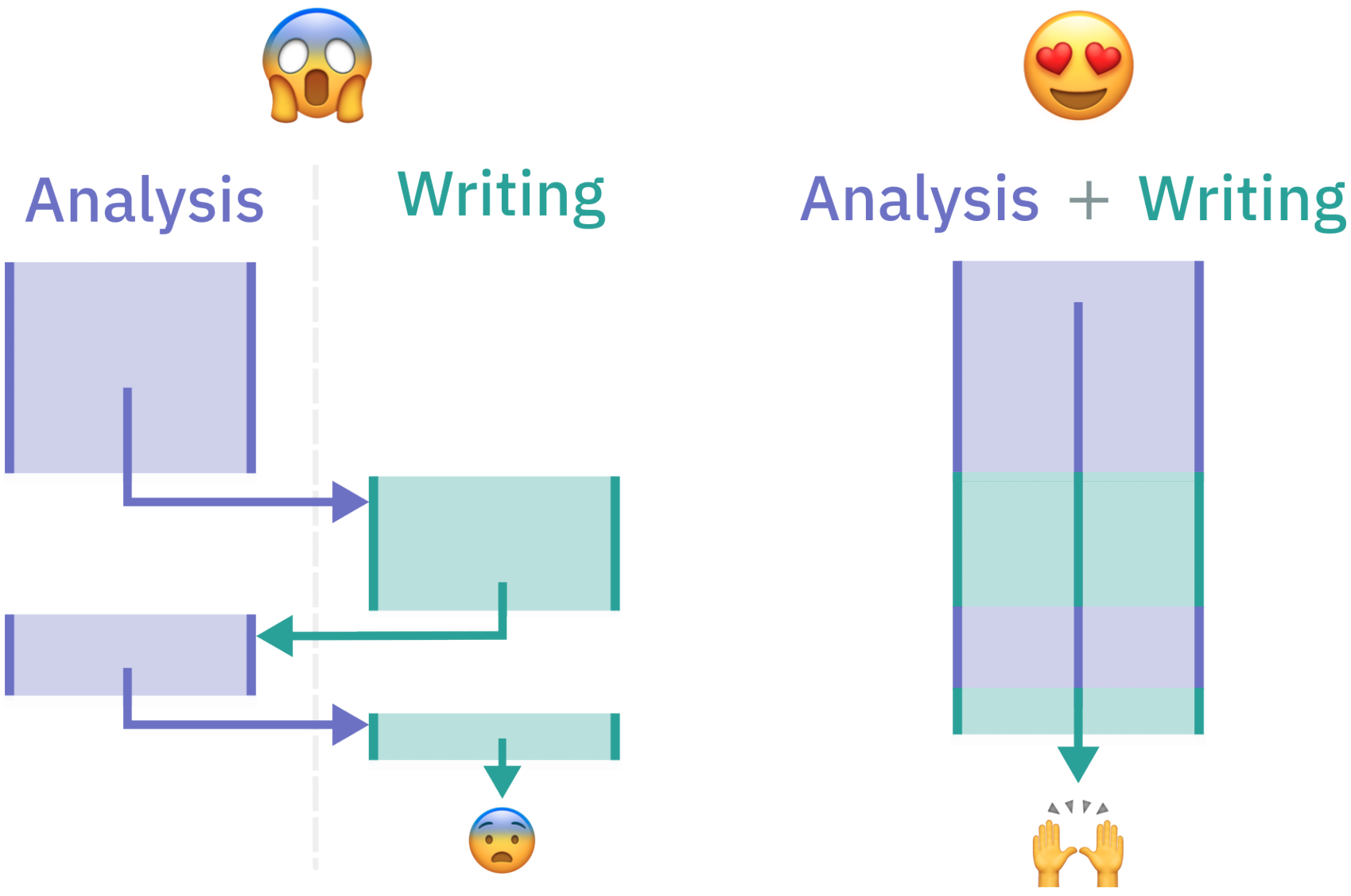
Dynamic reporting with Quarto


Analysis
Writing
Analysis + Writing

Literate programming

> *...a programming paradigm introduced by Donald Knuth in which a program is given as an explanation of the program logic in a natural language, such as English, interspersed with snippets of source code.*
>
> *— Wikipedia*

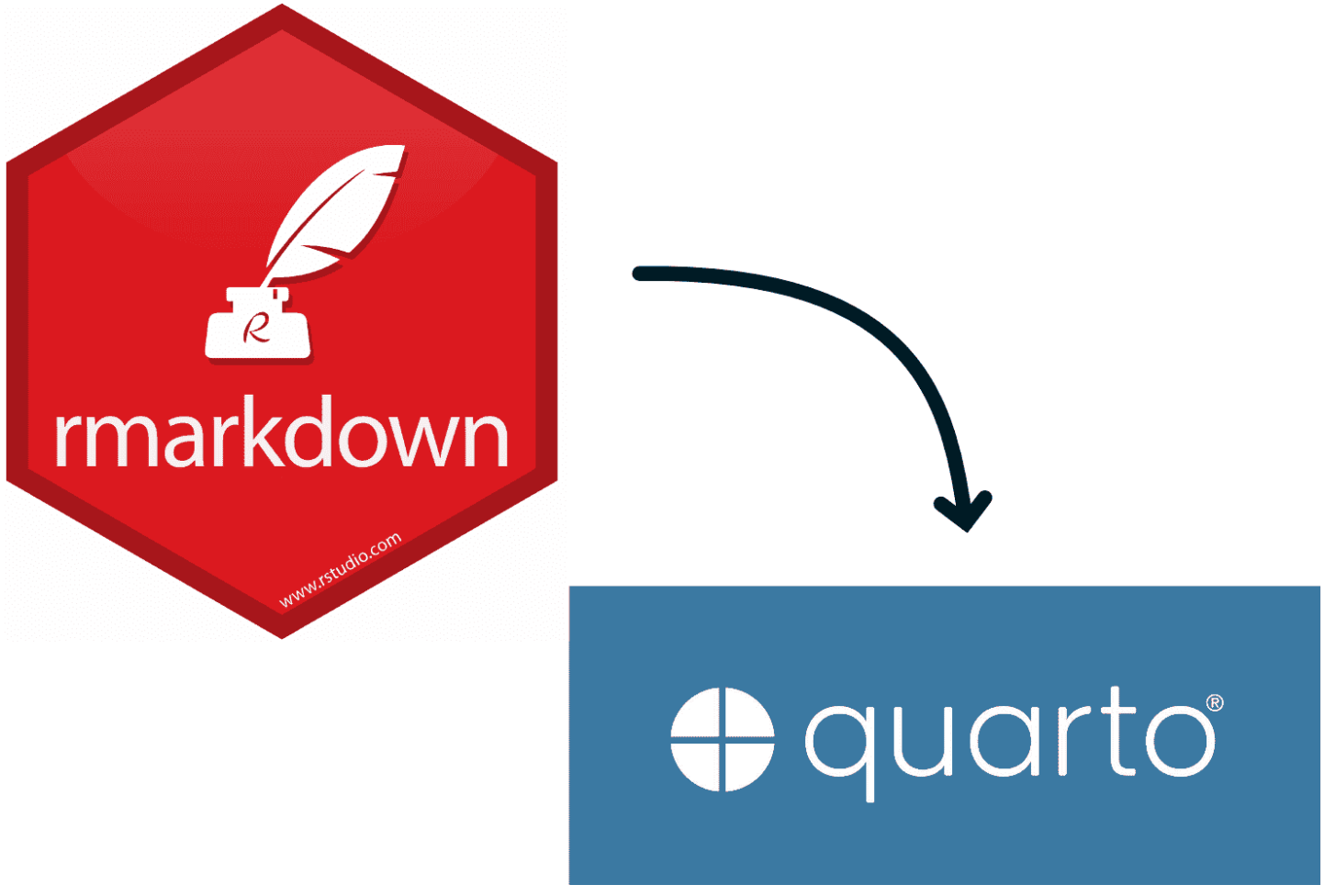
The key point here is the mixing of text and code in a single document.

Literate programming

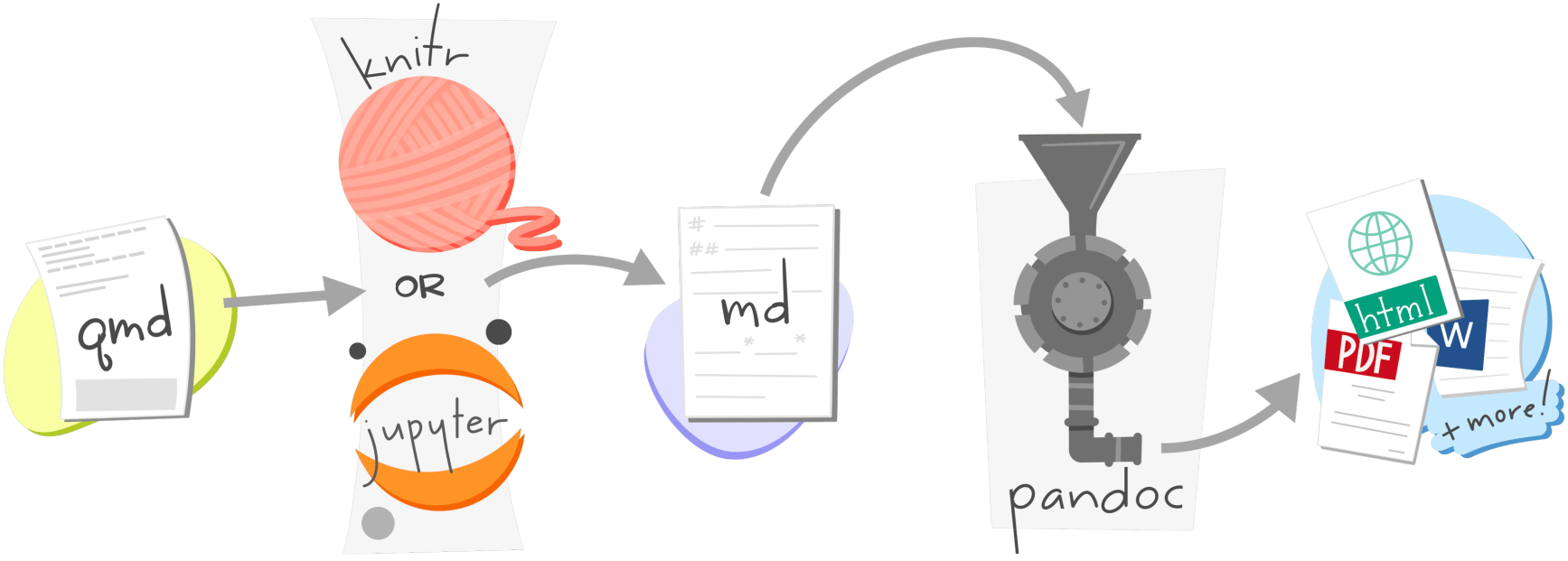
This brings many benefits:

- Reproducibility
- Efficiency
- Reduces human error (e.g. copy and paste).
- Combines code with documentation

Quarto is the successor to RMarkdown



Learn more @ `https://quarto.org`.



Artwork by @allison_horst

````
---
title: "My Quarto Report"
author: "Ewan Carr"
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>.

This is a code chunk:

```{r}
library(tidyverse)
library(lme4)
```

You can add options to code chunks like this

```{r}
#| echo: false
df <- read_csv("raw_data.csv")
```
````

Markdown

- Markdown is an easy-to-read, easy-to-write plain text format.

https://daringfireball.net/projects/markdown/.

```
# Heading 1

## Heading 2

This is some **bold text**. This is *italics*.
```

- Quarto uses Pandoc, a piece of software that converts one markup format into another.

http://pandoc.org/

- With this, we can convert RMarkdown to Microsoft Word, PDF (via LaTeX), HTML, PowerPoint, ePUB...

It's not just for documents.

- Papers
- Books
- Websites
- Presentations
- Documentation

```
https://quarto.org/docs/guide/
https://quarto.org/docs/gallery
```

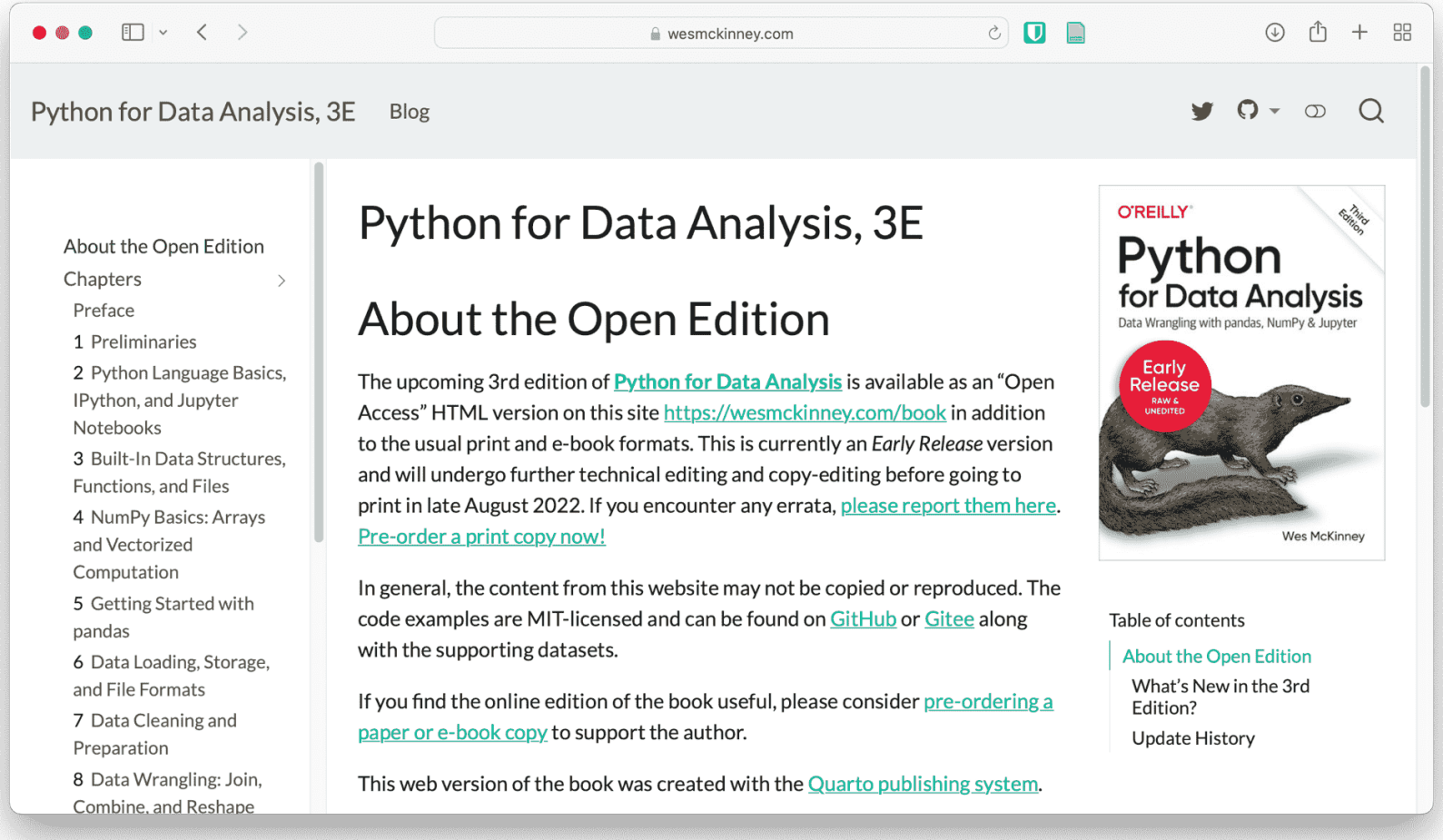
It's not just for documents.

- Papers
- Books
- Websites
- Presentations
- Documentation

```
https://quarto.org/docs/guide/
https://quarto.org/docs/gallery
```

Tip: **Write your data cleaning scripts in Quarto.**

And it's not just for R...



https://wesmckinney.com/book/

Demonstration!

1. Create a new document
2. Compiling ('knitting')
3. Markdown syntax
4. Code chunks and inline code.
5. The visual editor
6. Figures
7. Chunk options
    - `eval`
    - `echo`
    - `tbl-cap`
8. Layout options
9. Citations

Publication-ready tables

Packages for creating tables

There are many options:

1. `tinytable`

   https://vincentarelbundock.github.io/tinytable

2. `tabyl` from the `janitor` package

   https://sfirke.github.io/janitor/index.html

3. `gt` and `gtsummary`

   https://gt.rstudio.com

   https://www.danieldsjoberg.com/gtsummary

4. `table1`

   https://github.com/benjaminrich/table1

You could do this manually...

```
> library(tidyverse)
> library(palmerpenguins)
> penguins |>
>   group_by(species) |>
>   summarise(
>     across(where(is.numeric),
>            .fns = list(
>              mean = \(x) mean(x, na.rm = TRUE),
>              sd = \(x) sd(x, na.rm = TRUE)
>             )
>     )
>   )
# A tibble: 3 × 11
  species   bill_length_mm_mean bill_length_mm_sd bill_depth_mm_mean b
  <fct>                   <dbl>              <dbl>               <dbl>
1 Adelie                   38.8               2.66                18.3
2 Chinstrap                48.8               3.34                18.4
3 Gentoo                   47.5               3.08                15.0
```

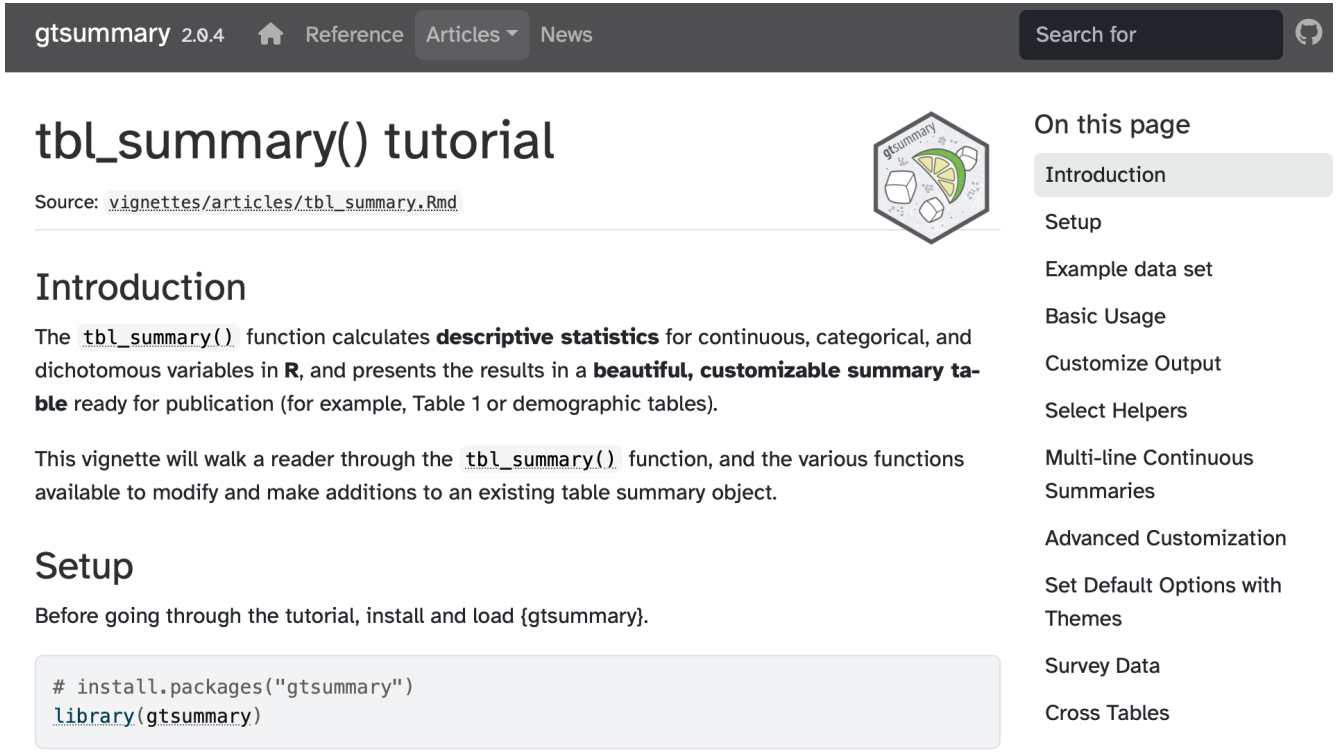
...but it's a lot of work.

```
> penguins |>
>   group_by(species) |>
>   summarise(
>     across(where(is.numeric),
>            .fns = list(
>              mean = \(x) mean(x, na.rm = TRUE),
>              sd = \(x) sd(x, na.rm = TRUE)
>            )
>     )
>   ) |>
>   pivot_longer(-species) |>
>   mutate(var = str_replace(name, "_mean$|_sd$", ""),
>          measure = if_else(str_detect(name, "_mean$"), "mean", "sd")
>   select(-name) |>
>   pivot_wider(names_from = measure,
>               values_from = value) |>
>   mutate(cell = make_cell(mean, sd)) |>
>   select(species, var, cell) |>
>   pivot_wider(names_from = species,
>               values_from = cell)
```

	var	Adelie	Chinstrap	Gentoo
	<chr>	<glue>	<glue>	<glue>
1	bill_length_mm	38.8 [2.7]	48.8 [3.3]	47.5 [3.1]
2	bill_depth_mm	18.3 [1.2]	18.4 [1.1]	15.0 [1.0]
3	flipper_length_mm	190.0 [6.5]	195.8 [7.1]	217.2 [6.5]
4	body_mass_g	3700.7 [458.6]	3733.1 [384.3]	5076.0 [504.1]
5	year	2008.0 [0.8]	2008.0 [0.9]	2008.1 [0.8]

gtsummary

A flexible package for creating tables of summary statistics and regression models, building on the powerful gt package.



I won't go into depth here; see documentation to learn more.

tbl_summary

```
penguins |>
  select(species,
         where(is.numeric)) |>
  tbl_summary()
```

Characteristic	N = 344[1]
species	
Adelie	152 (44%)
Chinstrap	68 (20%)
Gentoo	124 (36%)
bill_length_mm	44.5 (39.2, 48.5)
Unknown	2
bill_depth_mm	17.30 (15.60, 18.70)
Unknown	2
flipper_length_mm	197 (190, 213)
Unknown	2
body_mass_g	4,050 (3,550, 4,750)
Unknown	2
year	
2007	110 (32%)
2008	114 (33%)
2009	120 (35%)

[1] n (%); Median (Q1, Q3)

tbl_summary

```
penguins |>
  select(species, where(is.numeric)) |>
  tbl_summary(by = species)
```

Characteristic	Adelie N = 152[1]	Chinstrap N = 68[1]	Gentoo N = 124[1]
bill_length_mm	38.8 (36.7, 40.8)	49.6 (46.3, 51.2)	47.3 (45.3, 49.6)
Unknown	1	0	1
bill_depth_mm	18.40 (17.50, 19.00)	18.45 (17.50, 19.40)	15.00 (14.20, 15.70)
Unknown	1	0	1
flipper_length_mm	190 (186, 195)	196 (191, 201)	216 (212, 221)
Unknown	1	0	1
body_mass_g	3,700 (3,350, 4,000)	3,700 (3,475, 3,950)	5,000 (4,700, 5,500)
Unknown	1	0	1
year			
2007	50 (33%)	26 (38%)	34 (27%)
2008	50 (33%)	18 (26%)	46 (37%)

tbl_summary

```
penguins |>
  select(species, where(is.numeric)) |>
  tbl_summary(by = species,
              label = list(bill_length_mm = "Bill length (mm)",
                           bill_depth_mm = "Bill depth (mm)"),
              missing_text = "Missing")
```

Characteristic	Adelie N = 152[1]	Chinstrap N = 68[1]	Gentoo N = 124[1]
Bill length (mm)	38.8 (36.7, 40.8)	49.6 (46.3, 51.2)	47.3 (45.3, 49.6)
Missing	1	0	1
Bill depth (mm)	18.40 (17.50, 19.00)	18.45 (17.50, 19.40)	15.00 (14.20, 15.70)
Missing	1	0	1
flipper_length_mm	190 (186, 195)	196 (191, 201)	216 (212, 221)
Missing	1	0	1
body_mass_g	3,700 (3,350, 4,000)	3,700 (3,475, 3,950)	5,000 (4,700, 5,500)
Missing	1	0	1
year			

tabyl

`tabyl` is a function from the janitor package.

It's great for quick tables that don't require the heavy lifting of `tbl_summary`.

```
> library(janitor)
> penguins |>
>  tabyl(species)
species    n     percent
Adelie    152   0.4418605
Chinstrap  68    0.1976744
Gentoo    124   0.3604651
```

```
> library(janitor)
> penguins |>
>  tabyl(species) |>
>  adorn_totals() |>
>  adorn_pct_formatting()
species   n     percent
Adelie    152   44.2%
Chinstrap  68    19.8%
Gentoo    124   36.0%
Total     344   100.0%
```

See this page to learn more.

Practical: Quarto and tables