

NICE REPORT

Draft - New Haven

DataHaven

August 2023

CONTENTS

- Introduction 1
- Styling 2
- Utilities 3
- Notes 5

INTRODUCTION

This is an example of how to build a paged report. After processing some metadata here, it uses child documents to incorporate different chapters—this can be useful for putting together longer reports. Those child documents inherit the parent document’s global environment, so use this document to load packages, source scripts, and set parameters that will be used throughout. As an example, this document has a parameter `loc` which is used to create headers.

I’m also including front and back covers as starting points. `paged.js`, the JavaScript library used for splitting the HTML into pages and populating the margins, headers, footers, etc., has classes to denote front cover material and back cover material that then get special treatment (larger fonts, no page numbers, different margins, etc.).

STYLING

Because this format renders HTML first, then “prints” it to a PDF, any styling that can be used in HTML is fair game. This makes it *a lot* easier to do custom styling and layout than using LaTeX. The template uses a subset of Bootstrap for styling. Some is done manually with Sass (see the files in `./extension/pager/scss/`), and some gets inserted by Quarto. Beyond some basic styling choices (main accent color, font) these things will happen behind the scenes.

One thing you might want to do is adjust the layout, such as splitting text, figures, etc into columns. You can do this directly with HTML blocks using Bootstrap classes, first by wrapping the entire section to be split with `<div class="row"></div>`, then wrapping each column in a variation on `<div class="col-*)></div>`, where `*` is the number of grid columns to span. Bootstrap’s grid system adds up to 12, so two sections with `class="col-6"` will become evenly sized columns, whereas one `class="col-4"` and one `class="col-8"` will take up one third and two thirds of the page, respectively (works nicely for a skinny column of text next to a larger image). Or for even columns, as of Bootstrap 5.0 you can forgo the numbers altogether. You can also use Quarto’s `::: {.column-body}` notation or `#| column: body` chunk option. See more on Bootstrap’s grid system [here](#), and the source code for this document to see how these columns got here.

Also note that the `#| tbl-cap` chunk option creates a table caption as an HTML element, which gets styled to match the overall formatting. The figure equivalent is `fig-cap`. If you want figure titles to match the document styling (I generally do), use that instead of setting titles directly in `ggplot` or whatever you use to make the figure.

MEAN VALUES BY SPECIES, IRIS DATASET

Group.1	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
setosa	5.006	3.428	1.462	0.246
versicolor	5.936	2.770	4.260	1.326
virginica	6.588	2.974	5.552	2.026

UTILITIES

REFERENCES

Include `references` from a Bibtex document so they'll be linked to Zotero or another citation manager. A starter set of common references with shortcodes is included in the file `./references.bib`, and a formatter for numbered APA citations is in `./apa-numeric-superscript.csl`. For example, if I write something based on the 2021 ACS, I can cite it like this:

```
Some facts based on the ACS. [@acs21]
```

which will look like this:

Some facts based on the ACS.¹

This will automatically create a bibliography item placed in the Notes section. If there's a `html <div>` element with the ID `html "refs"` anywhere, bibliography items will be placed there; otherwise they're placed at the end of the document. Some YAML settings will change this. If you'd rather create the bibliography manually, set

```
suppress-bibliography: true
```

If you want all references to go in the margin instead, set

```
reference-location: margin
```

CUSTOM CLASSES

I've added a few classes to the stylesheets that I've found useful in making things like the equity reports. To use them, wrap your text in the requisite HTML block.

COLUMN LAYOUTS

There's a set of classes for creating layouts with margins on either the left or right of the page. Apply them with the classes `"margin-layout-right-20"`, `"margin-layout-right-25"`, `"margin-layout-left-20"`, or `"margin-layout-left-25"`, which denote 1) what side of the page has the margin, and 2) what percentage of the page that margin will take up. Wrap everything that should be in a side-by-side layout in an element with one of those classes. Since these are two-column layouts, there should be two elements inside the `"margin-layout-right-25"` block; however, if you want one side to stretch along several elements from the

other side, such as a long sidebar, wrap everything besides the margin block in an empty `<div>`.

If that's unclear, look at the source for this chapter to see it in action.

SIDEBARS

Create a sidebar box with a background fill and condensed text. This is good for things like definitions of a geography, indicators, etc, and works best alongside a large set of text. Use a margin layout, putting your sidebar content in an element with class `"sidebar"`.

EMPTY SIDEBAR

- There's nothing in this sidebar.
- It's just an example of how to use these layouts.
- Sidebars are good for breaking up lists separate from your text.

ASIDES

Quarto lets you place asides in the margin. For example, imagine there's some jargon in this paragraph that could benefit from a definition, but that doesn't need to take up space in the main body. Similar to sidebars, use a margin layout, then place your aside in an element with the class `"aside"`.

This is an aside with a note to go with the adjacent paragraph.

PAGE BREAKS

Create a section that won't have a page break, such as to keep a paragraph and the related figure on the same page.

```
<div class="no-break"></div>
```

NOTES

1. US Census Bureau. (2021). *American Community Survey 5-year estimates*. <https://data.census.gov/>

Closing information about funders, citation, etc.