Marginalia

A Typst package for (optionally) numbered margin notes. Based on the work of Nathan Jessurun for the `drafting` package .

Setup

Unfortunately `typst` doesn't expose margins to calling functions, so you'll need to set them explicitly. This is done using `set-page-properties()` *before you place any content*:

```
// At the top of your source file
// Of course, you can substitute any margin numbers you prefer
// provided the page margins match what you pass to `set-page-
properties()`
#import "../marginalia.typ": *
#let (l-margin, r-margin) = (1in, 2in)
#set page(
  margin: (left: l-margin, right: r-margin, rest: 0.1in),
  paper: "us-letter",
)
#set-page-properties(margin-left: l-margin, margin-right: r-margin)
```

The basics

Place a call to `margin-note()` anywhere to produce a margin note. You can specify the side where you want it to appear. The package will try to automacally find in which side you have more space so you don't need to specify the side every time.

```
#lorem(20)#margin-note[Hello, world!]
#lorem(10)#margin-note(side: left)[Hello from the other side]

#lorem(25)#margin-note[When   notes   are   about   to   overlap,   they're   automatically
shifted]#margin-note[To avoid collision]
#lorem(25)
```

2. Hello from the other side

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.[1] Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.[2]

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus.[34] Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus.

1. Hello, world!

3. When notes are about to overlap, they're automatically shifted

4. To avoid collision

Parameters

You can check the manual for all the parameters on the `margin-note()` function. You can specify any of the parameters on each call to customize each note individually. Or, as you will see in the next section, can update the defaults to your suiting.

```
#lorem(4)#margin-note(numbering-format:  "I.",  super-numbering-format:  "(i)")[You  can
modify the numbering format.]
#lorem(4)#margin-note(stroke: green, font-color: red)[Or add a stroke and change colors
around.]
#lorem(10)#margin-note(numbering: false)[Even disabling numbering!]
```

Lorem ipsum dolor sit.[(v)] Lorem ipsum dolor sit.[6] Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.

V. You can modify the numbering format.

6. Or add a stroke and change colors around.

Even disabling numbering!

Adjusting the default style

All function defaults are customizable through updating the module state, so if you have a preferred style, you do not have to specify it every time:

```
#lorem(4)#margin-note[By default, notes do not have a stroke.]
#set-margin-note-defaults(stroke: orange)
#lorem(4) #margin-note[But you can change defaults if you want.]
```

Lorem ipsum dolor sit.[7] Lorem ipsum dolor sit.[8]

Even deeper customization is possible by overriding the default `rect`:

```
#import "@preview/colorful-boxes:1.1.0": stickybox

#let default-rect(stroke: none, fill: none, width: 0pt, content) = {
  stickybox(rotation: 30deg, width: width/1.5, content)
}
#set-margin-note-defaults(rect: default-rect, stroke: none, side: right)

#lorem(20)
#margin-note(dy: -25pt)[Why not use sticky notes in the margin?]
```

7. By default, notes do not have a stroke.

8. But you can change defaults if you want.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.[9]

9. Why not use sticky notes in the margin?

Defining other margin-note functions

If you want to have multiple versions of margin notes you can! Using the powers of Typst you can create definitions of the `margin-note()` function with parameters pre-applied:

```
#let todo = margin-note.with(stroke: (paint: red, dash: "dotted"))
#lorem(20)
#todo[MUST FIX THIS!]
#lorem(15)
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.[10] Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore.

10. MUST FIX THIS!

Hiding notes for print preview

You can hide notes using a global setting. This is useful if you want to have reviewer notes in the margins and don't want them to appear on the printing document but you want regular margin notes.

```
#set-margin-note-defaults(hidden: true)

#lorem(20)
#margin-note[This will respect the global "hidden" state]
#margin-note(hidden: false, dy: -2.5em)[This note will never be hidden]
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.[11]

11. This note will never be hidden