FANSI

This package provides styles modeled after ISO 3864, ISO 7010, and ANSI Z535.

I HAVE NOT PURCHASED THESE STANDARDS. I DO NOT CLAIM THAT DOCUMENTS WRITTEN USING THIS PACKAGE ARE STYLISTICALLY IN CONFORMANCE WITH THEM. I JUST LIKE THE AESTHETIC.

That box uses stripes as shown on the 3864 Wikipedia page, an icon taken from the 7010 Wikipedia page, and the color-scheme taken from a free preview of the Z535 document I found online somewhere. You'd think that the ISO 7010 icon set would use the ISO 3864 color scheme, but they don't. They don't use ANSI Z535 colors either, but they're close enough that *I* can't see the difference, anyway.

Maybe in the future I'll go normalize colors.

Installation

See the Typst Packages README for up-to-date information.

This repository is an entire Typst namespace. Clone it into your Typst package directory as specified at that link:

```
cd $XDG_DATA_HOME/typst/packages
git clone https://github.com/myrrlyn/wyz-typst wyz
```

It will then become available in your Typst environments as:

```
#import "@wyz/fansi:0.1.0"
```

Usage

The primary exports are the `signage()` and `iso7010()` functions. These are closures which capture their arguments and make them available to all items stored in the dictionary that they return.

This behavior is useful for allowing you to quickly override settings. I don't know how to receive `show` or `set` rules into a package yet.

The dictionaries returned from these functions are full of element-producing functions. Invoke *those* functions to produce content on the page.

Secondary exports are the `.ansi` and `.iso` dictionaries. These are the color palettes of their eponymous standards:

	ANSI Z535	ISO 3864
.red	#c8102e	#9b2423
.orange	#ff8200	#d05d29
.yellow	#ffd100	#f9a900
.green	#007b5f	#237f52
.blue	#0072ce	#005387
.purple	#6d2077	#6d2077
.black	#2c2a29	#282828
.white	#e7e6e2	#ededе8
	#fansi.ansi	#fansi.iso

Callout Boxes

The boxes are provided through the `signage` function. This function accepts an argument-list, which it snapshots and forwards to all the functions held inside the dictionary it returns. The keys of the dictionary are `custom`, `ansi`, and `iso`. The `ansi` and `iso` keys are dictionaries of six functions each: `red`, `orange`, `yellow`, `green`, `blue`, and `purple`. These six functions, and `custom`, produce `rects` with striped borders as shown above. The `custom` function takes two arguments, its primary and secondary colors, while the other six produce boxes colored according to their names, using either ANSI Z535 or ISO 3864 as their color palette.

Custom arguments captured by the `rect` generator are:

- `size`: the width of the striped border. Defaults to `12pt`.
- `icon`: if present, becomes an element in the top left corner of the contents. Expected, but not required, to be one of the ISO 7010 images also available in this package.
- `stroke-join`: Forwarded to the rectangle's `stroke()` property.
- `stripe-dir`: Forwarded to the striping generator. Controls whether the stripes are SW/NE (default, `ltr`, `btt`) or NW/SE (`rtl`, `ttb`).

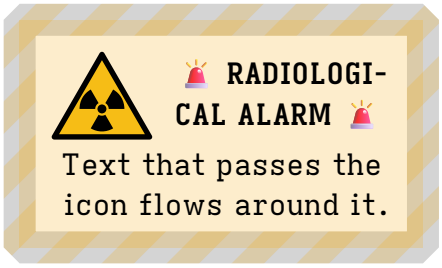
All other arguments are passed to `rect()` as-is.

```
#(fansi.signage().ansi.purple)[inner text]
```

inner text

```
#let bsg = fansi.signage(stroke-join: "bevel").iso
#let icns = fansi.iso7010(height: 1em)
#(bsg.yellow)(icon: (icns.w003)(), align(center)[
  #v(4pt)
  *#emoji.siren RADIOLOGICAL ALARM #emoji.siren*

  Text that passes the icon flows around it.
])
```



From here on, elements are shown against full-white and full-black backgrounds to illustrate how they appear in both light and dark settings.

Color	ANSI Z535	ISO 3864
.red		
.orange		
.yellow		
.green		
.blue		
.purple		
.custom	or bring your own colors	
	#fansi.signage().ansi	#fansi.signage().iso

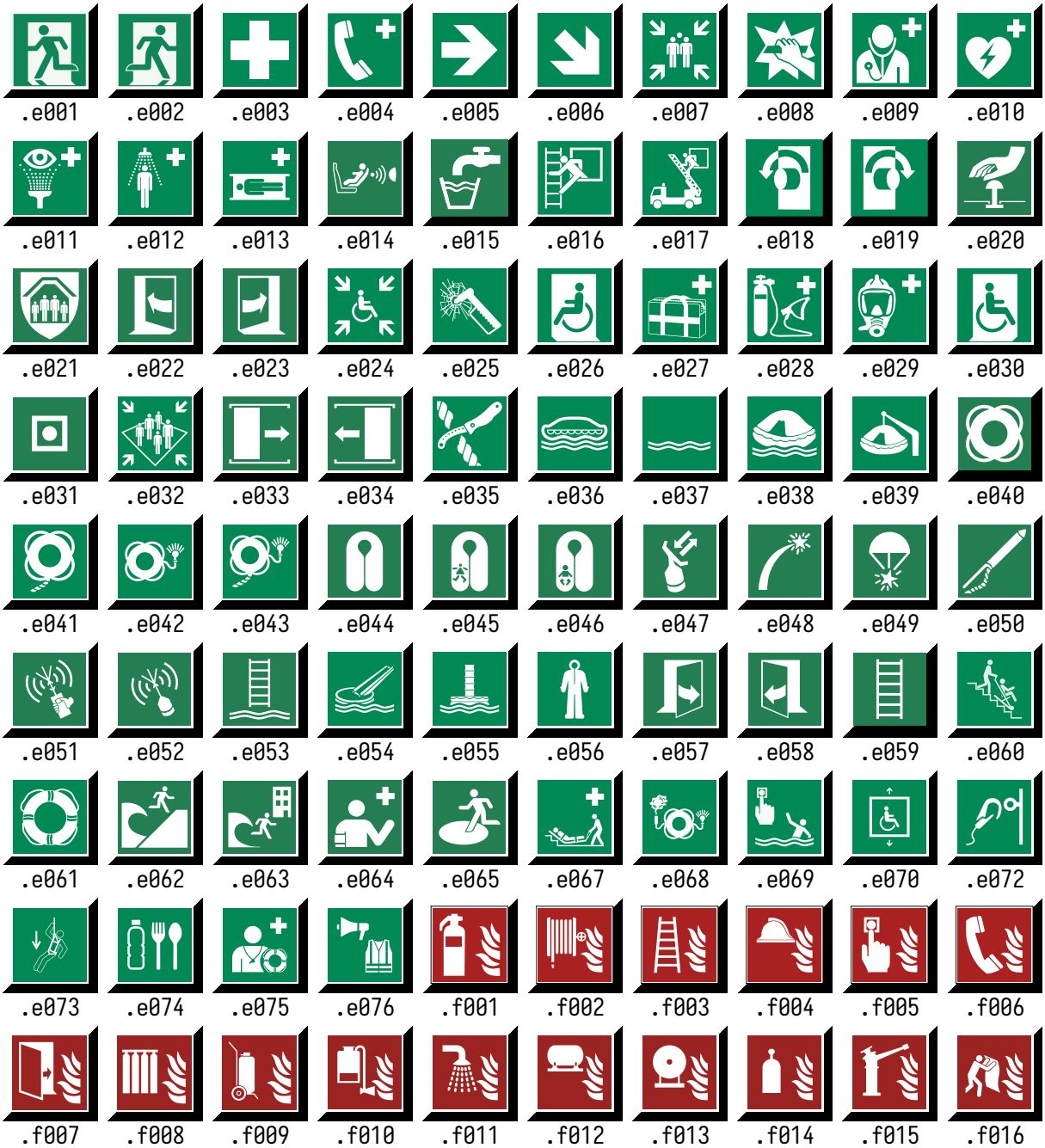
The boxes have a mostly-transparent fill of their primary color.

ISO 7010 Icons

- .eNNN: green background, white foreground, square, thin white border
- .fNNN: red background, white foreground, square, thin white border (except .f019)
- .mNNN: blue background, white foreground, circular, no border
- .pNNN: white background, black foreground, circular, thick red negation overlay (border and slash)
- .wNNN: yellow background, black foreground, triangular, moderate black border

There are no orange or purple icons.

Icons are held in the dictionary returned by `#fansi.iso7010()`.



.f017 .f018 .f019 .m001 .m002 .m003 .m004 .m005 .m006 .m007

.m008 .m009 .m010 .m011 .m012 .m013 .m014 .m015 .m016 .m017

.m018 .m019 .m020 .m021 .m022 .m023 .m024 .m025 .m026 .m027

.m028 .m029 .m030 .m031 .m032 .m033 .m034 .m035 .m036 .m037

.m038 .m039 .m040 .m041 .m042 .m043 .m044 .m045 .m046 .m047

.m048 .m049 .m050 .m051 .m052 .m053 .m054 .m055 .m056 .m057

.m058 .m059 .m060 .m061 .m062 .m068 .m069 .m070 .m071 .m072

.p001 .p002 .p003 .p004 .p005 .p006 .p007 .p008 .p009 .p010

.p011 .p012 .p013 .p014 .p015 .p016 .p017 .p018 .p019 .p020

.p021 .p022 .p023 .p024 .p025 .p026 .p027 .p028 .p029 .p030

.p031 .p032 .p033 .p034 .p035 .p036 .p037 .p038 .p039 .p040

.p041 .p042 .p043 .p044 .p045 .p046 .p047 .p048 .p049 .p050

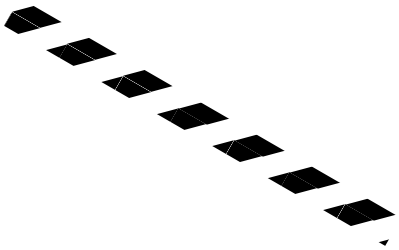
Just The Stripe

`#fansi.striping` is a function which produces the `tiling()` element used by the signage boxes. It receives two color arguments, one size argument, and an optional direction argument. It can be used anywhere a `tiling()` is, and has the same behavior.

The direction (`stripe-dir`) defaults to `ltr`. When it is `ltr` or `btt`, stripes run from southwest to northeast. When it is `rtl` or `ttb`, stripes run from northwest to southeast.

The angle of the striping itself cannot be changed.

```
#rotate(30deg, reflow: true,
  line(length: 15em,
    stroke: stroke(
      thickness: 12pt,
      paint: fansi.striping(
        black, white, 12pt
      ),
    ),
  )
)
```



If the `size` argument to `fansi.striping` is smaller than the short dimension of the rectangle it fills, then the stripes will appear visually narrower. If it is larger, they will be thicker.

```
#line(length: 15em, stroke(
  thickness: 12pt,
  paint: fansi.striping(
    black, white, 6pt
  ),
))
```

```
#line(length: 15em, stroke(
  thickness: 12pt,
  paint: fansi.striping(
    black, white, 24pt, dir: rtl
  ),
))
```