keyle

v0.1.1   2024-08-13   MIT

This package provides a simple way to style keyboard shortcuts in your documentation.

magicwenli <YXNIAN@OUTLOOK.COM>

https://github.com/magicwenli/keyle

Table of contents

Part I.
About

KEYLE is a library that allows you to create HTML `<kbd>` like keyboard shorts simple and easy.

The name, keyle, is a combination of key and theme.

This project is inspired by auth0/kbd and dogezen/badgery. Send them respect!

Part II.
Usage

II.1. Importing

KEYLE is imported using

```
#import "@preview/keyle:0.1.0"
```

II.2. Quick Start

Generate a keyboard renderer with #config() function. Section Section III.1 lists available themes.

```
#let kbd = keyle.config()
#kbd("Ctrl","Shift","Alt","Del")

#let kbd = keyle.config(
  theme: keyle.themes.biolinum,
  delim: keyle.biolinum-key.delim_plus
)
#kbd("Ctrl","Shift","Alt","Del")
```

Ctrl + Shift + Alt + Del

Ctrl + ⇧ + Alt + Del

There are compact and delim options to make the output more compact and change the delimiter between keys.

You can either use them in #config() function or directly in generated kbd function.

```
#let kbd = keyle.config(compact: true, delim: "-")
#kbd("Ctrl","Shift","Alt","Del")

#let kbd = keyle.config()
#kbd("Ctrl","Shift","Alt","Del", compact: true, delim: "-")
```

Ctrl-Shift-Alt-Del

Ctrl-Shift-Alt-Del

Part III.
Themes

III.1. Built-in Themes

Themes function are available at #keyle.themes dictionary.

#keyle.themes.standard

#keyle.themes.deep-blue

#keyle.themes.type-writer

#keyle.themes.biolinum

III.2. Custom Themes

You can create your own theme by defining a function that takes a string and returns a themed content.

```
// https://www.radix-ui.com/themes/playground#kbd
#let radix_kdb(content) = box(
  rect(
    inset: (x: 0.5em),
    outset: (y:0.05em),
    stroke: rgb("#1c2024") + 0.3pt,
    radius: 0.35em,
    fill: rgb("#fcfcfd"),
    text(fill: black, font: (
      "Segoe UI (Custom)",
      "Roboto",
      "Helvetica Neue",
      "Open Sans (Custom)",
      "sans-serif",
      "Apple Color Emoji",
      "Segoe UI Emoji"
    ), content),
  ),
)
#let kbd = keyle.config(theme: radix_kdb)
#kbd("⌘ D") #kbd("^ F")
```

⌘ D   ^ F

Part IV.
Symbols

IV.1. Mac Keyboard Symbols

KEYLE provides symbols for Mac keyboard. You may need install Fira Code font to see the symbols correctly.

- Fira Code @ Google Fonts
- Apple Mac Keyboard Symbols

```
#let kbd = keyle.config(theme: keyle.themes.standard)
#let mac-key-font = (
  "Fira Code",
  "FiraCode",
  "FiraCode Nerd Font Mono",
)
#grid(
  columns: (2fr, 1fr, 2fr, 1fr),
  rows: 2em,
  align: horizon,
  ..keyle.mac-key.pairs().map(item => (
    raw("#keyle.mac-key." + item.at(0)),
    kbd(
      text(font: mac-key-font, item.at(1))
    ),
  )).flatten()
)
```

#keyle.mac-key.command	⌘	#keyle.mac-key.shift	⇧
#keyle.mac-key.option	⌥	#keyle.mac-key.control	⌃
#keyle.mac-key.return	↵	#keyle.mac-key.delete	⌫
#keyle.mac-key.forward-delete	⌦	#keyle.mac-key.escape	⎋
#keyle.mac-key.left	←	#keyle.mac-key.right	→
#keyle.mac-key.up	↑	#keyle.mac-key.down	↓
#keyle.mac-key.pageup	⇳	#keyle.mac-key.pagedown	⇳
#keyle.mac-key.home	⬉	#keyle.mac-key.end	⬊
#keyle.mac-key.tab-right	⇥	#keyle.mac-key.tab-left	⇤
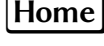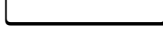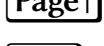
IV.2. Linux Biolinum Keyboard Symbols

KEYLE provides symbols for Linux Biolinum Keyboard font.

- Linux Biolinum Keyboard

```
#let kbd = keyle.config(theme: keyle.themes.biolinum)
#grid(
  columns: (5fr, 3fr, 5fr, 3fr),
  rows: 2em,
  align: horizon,
  ..keyle.biolinum-key.pairs().map(item => (
    raw("#keyle.biolinum-key." + item.at(0)),
    kbd(item.at(1)),
  )).flatten()
)
```

#keyle.biolinum-key.Strg	**Strg**	#keyle.biolinum-key.Alt	**Alt**
#keyle.biolinum-key.Ctrl	**Ctrl**	#keyle.biolinum-key.Shift	⇧
#keyle.biolinum-key.Tab	⇄	#keyle.biolinum-key.Enter	↵
#keyle.biolinum-key.Capslock	⇩	#keyle.biolinum-key.Home	**Home**
#keyle.biolinum-key.Del	**Del**	#keyle.biolinum-key.Ins	**Ins**
#keyle.biolinum-key.End	**End**	#keyle.biolinum-key.Space	
#keyle.biolinum-key.Esc	**Esc**	#keyle.biolinum-key.PageUp	**Page↑**
#keyle.biolinum-key.PageDown	**Page↓**	#keyle.biolinum-key.Back	←
#keyle.biolinum-key.Pad_0	**Pad 0**	#keyle.biolinum-key.Pad_1	**Pad 1**
#keyle.biolinum-key.Pad_2	**Pad 2**	#keyle.biolinum-key.Pad_3	**Pad 3**
#keyle.biolinum-key.Pad_4	**Pad 4**	#keyle.biolinum-key.Pad_5	**Pad 5**
#keyle.biolinum-key.Pad_6	**Pad 6**	#keyle.biolinum-key.Pad_7	**Pad 7**
#keyle.biolinum-key.Pad_8	**Pad 8**	#keyle.biolinum-key.Pad_9	**Pad 9**
#keyle.biolinum-key.Pad_Div	**Pad ÷**	#keyle.biolinum-key.Pad_Add	**Pad +**
#keyle.biolinum-key.Pad_Sub	**Pad −**	#keyle.biolinum-key.Pad_Mul	**Pad ×**
#keyle.biolinum-key.Pad_Enter	**Pad ↵**	#keyle.biolinum-key.Mac_Cmd	⌘
#keyle.biolinum-key.Mac_Opt	⌥	#keyle.biolinum-key.Mac_FDel	⌦
#keyle.biolinum-key.Mac_Del	⌫	#keyle.biolinum-key.GNU	🐃
#keyle.biolinum-key.Win	⊞	#keyle.biolinum-key.Tux	🐧
#keyle.biolinum-key.delim_plus	+	#keyle.biolinum-key.delim_minus	−

Part V.
Available commands

#**theme-func-stardard**(⟨**sym**⟩) ⟶ `content`

Theme function to render keys in a standard style.

```
#let kbd = keyle.config(theme: keyle.themes.standard)
#keyle.gen-examples(kbd)
```

Ctrl + Alt + A   Ctrl+Shift+A   Home End Ins Del

Argument

⟨sym⟩   str

The key symbol to render.

#**theme-func-deep-blue**(⟨**sym**⟩) ⟶ `content`

Theme function to render keys in a deep blue style.

```
#let kbd = keyle.config(theme: keyle.themes.deep-blue)
#keyle.gen-examples(kbd)
```

Ctrl + Alt + A   Ctrl+Shift+A   Home End Ins Del

Argument

⟨sym⟩   str

The key symbol to render.

#**theme-func-type-writer**(⟨**sym**⟩) ⟶ `content`

Theme function to render keys in a type writer style.

```
#let kbd = keyle.config(theme: keyle.themes.type-writer)
#keyle.gen-examples(kbd)
```

Ctrl + Alt + A   Ctrl+Shift+A   Home End Ins Del

Argument

⟨sym⟩   str

The key symbol to render.

#**theme-func-biolinum**(⟨**sym**⟩) ⟶ `content`

Theme function to render keys in a Linux Biolinum Keyboard style.

You need to have the font installed on your system.

```
#let kbd = keyle.config(theme: keyle.themes.biolinum, delim: keyle.biolinum-key.delim_plus)
#keyle.gen-examples(kbd)
```

⎯⎯⎯

`Ctrl` + `Alt` + `A`   `Ctrl` + `⇧` + `A`   `Home` `End` `Ins` `Del`

— Argument ——

⟨sym⟩   str

   The key symbol to render.

#**config**(⟨**theme**⟩: **"themes.standard"**, ⟨**compact**⟩: **false**, ⟨**delim**⟩: **"+"**) ⟶ `function`
   Config function to generate keyboard rendering helper function.

— Argument ——

⟨theme⟩: "themes.standard"   function

   The theme function to use.

— Argument ——

⟨compact⟩: false   bool

   Whether to render keys in a compact format.

— Argument ——

⟨delim⟩: "+"   str

   The delimiter to use when rendering keys in compact format.

Part VI.
Index

C

T