

keyle

v0.1.0

2024-07-24

MIT

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

magicwenli <YXNIAN@OUTLOOK.COM>

Table of contents

I. About

II. Usage

II.1. Available constants	3
II.1.1. kbd-styles	3
II.1.2. mac-key and com-key	4

III. Available commands

IV. Index

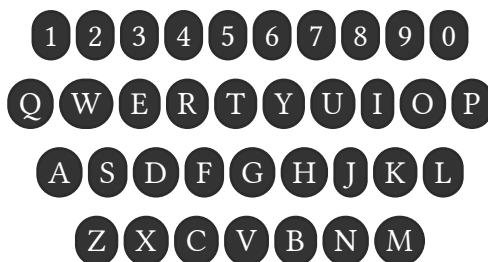
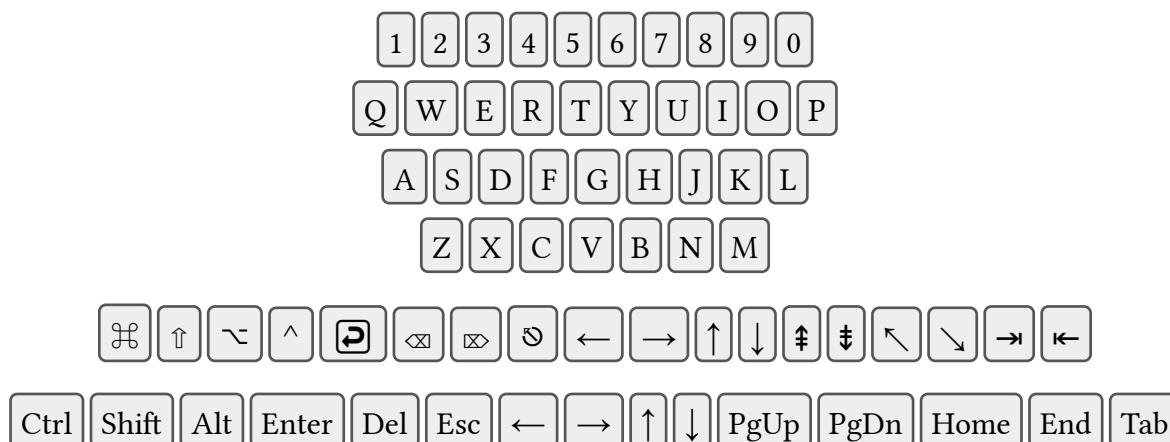
Part I.

About

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

The name, `keyle`, is a combination of `key` and `style`.

This project is inspired by [auth0/kbd](#) and [dogezen/badgery](#). Send them respect!



Part II.

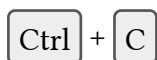
Usage

`KEYLE` is imported using

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

Basic usage is as follows, see [#kbd\(\)](#) for more details:

```
#keyle.kbd("Ctrl", "C")
```



Also you can change the style of the keyboard. See [#kbd-set-style\(\)](#) for more details.

```
#keyle.kbd-set-style("deep-blue")  
#keyle.kbd("Alt")  
#keyle.kbd-set-style("type-writer")  
#keyle.kbd("Enter")  
#keyle.kbd-set-style("standard")
```



You can also use the constants `mac-key` and `com-key` to get the key names for Mac and Windows/Linux respectively.

```
#keyle.kbd(keyle.mac-key.control, keyle.mac-key.shift, "A")
```



II.1. Available constants

II.1.1. kbd-styles

Use `kbd-styles` to get available styles.

2.1 Available constants

```
#keyle.kbd-styles
```

```
(  
  standard: "standard",  
  deep-blue: "deep-blue",  
  type-writer: "type-writer",  
)
```

II.1.2. mac-key and com-key

Use mac-key and com-key to get the key names for Mac and Windows/Linux respectively.

```
#keyle.mac-key
```

```
(  
  command: "⌘",  
  shift: "⇧",  
  option: "⌥",  
  control: "⌞",  
  return: "↵",  
  delete: "⌫",  
  forward-delete: "〉",  
  escape: "⌫",  
  left: "←",  
  right: "→",  
  up: "↑",  
  down: "↓",  
  pageup: "⌞",  
  pagedown: "⌟",  
  home: "⌞",  
  end: "⌟",  
  tab-right: "→",  
  tab-left: "←",  
)
```

2.1 Available constants

#keyle.com-key

```
(  
  control: "Ctrl",  
  shift: "Shift",  
  alt: "Alt",  
  return: "Enter",  
  delete: "Del",  
  escape: "Esc",  
  left: "←",  
  right: "→",  
  up: "↑",  
  down: "↓",  
  pageup: "PgUp",  
  pagedown: "PgDn",  
  home: "Home",  
  end: "End",  
  tab: "Tab",  
)
```

Part III.

Available commands

`#kbd()`

`#kbd-set-style()`

`#kbd-set-style(<s>)`

Set the keyboard style.

```
#keyle.kbd[A] // standard style
#keyle.kbd-set-style("deep-blue")
#keyle.kbd[A] // deep-blue style
#keyle.kbd-set-style(keyle.kbd-styles.standard) // back to standard style
```



Argument

<s>

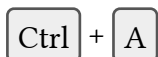
str

The style name. Use `kbd-styles` to get available styles.

`#kbd(..<keys>, <compact>: false, <delim>: "+")`

Renders a keyboard key.

```
#keyle.kbd("Ctrl", "A")
```



```
#keyle.kbd("Ctrl", "A", compact: true)
```



```
#keyle.kbd("Ctrl", "A", compact: true, delim: "-")
```



Argument

..`<keys>`

str

The key to render.

Argument

<compact>: false

bool

If true, the keys will be rendered in a single box.

Argument

`<delim>: "+"`

`str`

The delimiter to use when rendering in compact mode.

Part IV.

Index

K

<code>#kbd</code>	6
<code>#kbd-set-style</code>	6