

Vivliostyle リファレンス

Table of Contents

リファレンス Vivliostyle Documentation	3
Vivliostyle.js コントリビューションガイド Vivliostyle Documentation.....	4
Contribution Guide Vivliostyle Documentation.....	5
Contribution Guide Vivliostyle Documentation.....	6
Contribution Guidelines Vivliostyle Documentation	7
Core API リファレンス Vivliostyle Documentation	8
サポートする CSS 機能 Vivliostyle Documentation	9

リファレンス

リファレンス

ドキュメント

- サポートする CSS 機能
- Core API リファレンス

コントリビューションガイド

各プロダクトの開発に参加する方法について説明します。

- Vivliostyle.js
- Vivliostyle CLI
- VFM
- Vivliostyle Themes

Vivliostyle.js コントリビューションガイド

Vivliostyle.js コントリビューションガイド

モジュール構成

Vivliostyle.js は2つのコンポーネントで構成されています:

1. https://github.com/vivliostyle/vivliostyle.js/tree/master/packages/core ^{*1}	2. https://github.com/vivliostyle/vivliostyle.js/tree/master/packages/viewer ^{*2}
Vivliostyle.js Core layout engine	Vivliostyle.js Viewer UI

開発環境のセットアップ

以下のものがインストールされていることを確認してください:

- 3. <https://nodejs.org>Node.js^{*3}
- 4. <https://classic.yarnpkg.com>Yarn^{*4}
- 5. <https://git-scm.com>Git^{*5}

6. <https://github.com/vivliostyle/vivliostyle.js>^{*6}をクローンします。

```
git clone https://github.com/vivliostyle/vivliostyle.js.git
cd vivliostyle.js
```

依存関係をインストールし、パッケージをリンクします:

```
yarn install # install dependencies and link packages
```

`@vivliostyle/core` は `@vivliostyle/viewer` 内の `package.json` で dependency に含まれています。開発時には、`@vivliostyle/core` はnpmからインストールされたパッケージではなくローカルのコピーが用いられます。

Contribution Guide

Contribution Guide

Development Guide

Please ensure pnpm:>=10 is installed in your machine.

```
git clone https://github.com/vivliostyle/vivliostyle-cli.git && cd vivliostyle-cli
pnpm install
```

After setup, run `pnpm dev` to watch files.

Contribution Guide

Contribution Guide

Development Guide

```
git clone https://github.com/vivliostyle/vfm.git && cd vfm
npm install
npm run build
npm link
DEBUG=vfm vfm
```

After setup, run `npm run dev` to watch files.

Commit Guide

Comment

Please refer to 1. <https://gist.github.com/joshbучea/6f47e86d2510bce28f8e7f42ae84c716>Semantic Commit Messages^{*1} for the commit comment.

Release Guide (Maintainers only)

Enter pre-release

```
release-it --preRelease=beta --npm.tag=next
```

Bump pre-release version

```
npm run release:pre
# or
release-it --preRelease --npm.tag=next
```

Contribution Guidelines

Contribution Guidelines

Developing theme

- See Theme Spec.

Donating your theme

We appreciate your donation to the list of our official themes!

Merging criteria

- The design and type of your theme should be reasonably distinct from all other existing official themes. This is to avoid duplication.
- Your theme has to meet our Theme Spec.
- You have to sign the Contributor License Agreement.
- You must follow Code of Conduct.

Perks

- You will become a member of 1. <https://github.com/vivliostyleVivliostyle>^{*1} as well as a maintainer of `vivliostyle/theme` as a part of @vivliostyle/theme team.
- You will be given permissions to manage `vivliostyle/themes` and all the other theme-related repositories which enables you to:
 - Push to the repository
 - Triage issues
 - Review and merge pull-requests

Core API リファレンス

Core API リファレンス

- API
 - constants
 - `PageProgression`
 - `PageSide`
 - `ReadyState`
 - plugin
 - `registerHook(name, fn)`
 - `removeHook(name, fn)`
 - profile
 - `profiler.registerStartTiming(name, timestamp)`
 - `profiler.registerEndTiming(name, timestamp)`
 - `profiler.printTimings()`
 - `profiler.disable()`
 - `profiler.enable()`
 - core-viewer
 - `new CoreViewer(settings, options)`
 - `PageViewMode`
 - `ZoomType`
- Classes
 - `CoreViewer`
 - `addListener(type, listener)`
 - `getCurrentPageProgression()`
 - `getPageSizes()`
 - `isTOCVisible()`
 - `loadDocument(singleDocumentOptions, documentOptions, viewerOptions)`
 - `loadPublication(pubURL, documentOptions, viewerOptions)`
 - `navigateToInternalUrl()`
 - `navigateToPage()`
 - `queryZoomFactor(type)`
 - `removeListener(type, listener)`

サポートする CSS 機能

サポートする CSS 機能

Vivliostyle は現在、以下の各 CSS 機能（Values、Selectors、At-rules、Media queries、Properties）をサポートしています。

このほかにも、ブラウザでサポートされる CSS プロパティと値は基本的にすべて利用可能です。Vivliostyle.js は独自の処理をしない CSS プロパティについてはブラウザに処理を任せるためです。

Values

- 1. <https://www.w3.org/TR/css-values/#common-keywords>CSS-wide keywords^{*1}: `initial`, `inherit`, `unset`, `revert`
- 2. <https://www.w3.org/TR/css-values/#lengths>Length units^{*2}: `em`, `ex`, `ch`, `rem`, `lh`, `rlh`, `vw`, `vh`, `vmin`, `vmax`, `vi`, `vb`, `cm`, `mm`, `q`, `in`, `pc`, `pt`, `px`.
- Sizing keywords: 3. <https://www.w3.org/TR/css-sizing-3/#valdef-width-min-content>min-content^{*3}, 4. <https://www.w3.org/TR/css-sizing-3/#valdef-width-max-content>max-content^{*4}, 5. <https://www.w3.org/TR/css-sizing-4/#valdef-width-fit-content>fit-content^{*5}
- Color values
 - 6. <https://www.w3.org/TR/css-color/#named-colors>Named Colors^{*6}
 - 7. <https://www.w3.org/TR/css-color/#transparent-color>`transparent` color keyword^{*7}
 - 8. <https://www.w3.org/TR/css-color/#currentcolor-color>`currentColor` color keyword^{*8}
 - 9. <https://www.w3.org/TR/css-color/#rgb-functions>RGB functions: `rgb()`, `rgba()`^{*9}
 - 10. <https://www.w3.org/TR/css-color/#hex-notation>RGB Hexadecimal Notations: `#RRGGBB`, `#RRGGBBAA`^{*10}
 - 11. <https://www.w3.org/TR/css-color/#the-hsl-notation>HSL Colors: `hsl()`, `hsla()`^{*11}
 - 12. <https://www.w3.org/TR/css-color/#the-hwb-notation>HWB Colors: `hwb()`^{*12}
- 13. <https://www.w3.org/TR/css-values/#attr-notation>Attribute references: `attr()`^{*13}
 - Only supported in values of `content` property.
 - Only 'string' and 'url' types are supported.
- 14. <https://www.w3.org/TR/css-content-3/#cross-references>Cross references:

