

Vivliostyle Themes ドキュメント

Table of Contents

Vivliostyle Themes	5
運用ガイドライン	7
公式Themeの採用	8
採用条件	9
採用方法	9
Vivliostyle Themesギャラリー	10
公式のVivliostyle Themes	11
@vivliostyle/theme-bunko	11
@vivliostyle/theme-slide	12
@vivliostyle/theme-techbook	12
@vivliostyle/theme-academic	13
@vivliostyle/theme-gutenberg	14
@vivliostyle/theme-epub3j	16
非公式のVivliostyle Themes	16
vivliostyle-theme-dnd-5e-phb	16
vivliostyle-theme-thesis	18
vivliostyle-akashi-kosen-bulletin	18

Vivliostyle Themeの仕様	19
テーマ名	20
テーマに含めるスタイル	20
ディレクトリ構造	21
package.json	21
keywords プロパティ	22
vivliostyle.theme プロパティ	22
example/	23
theme.css	23
vivliostyle.config.js	24
Themeの開発	25
Themeを作成する	26
create-vivliostyle-themeによる雛形生成	26
生成されるファイル構成	26
雛形をカスタマイズする	27
theme.cssの編集	27
theme-baseのモジュール活用	28
CSS変数のオーバーライド	29
example/ディレクトリの編集	30
Themeを公開する	31
事前検証	31
プレビュー確認	31
package.jsonの必須フィールド	31
npmへの公開	32
公式テーマとして提案する	33

Vivliostyle Themes



Vivliostyle Themesのロゴ

Vivliostyle Theme は、Vivliostyle で出版物を作る際に使うスタイルテーマです。Vivliostyle Themes を使うことで簡単に出版物のスタイルを変更することができます。

- [テーマの使い方](#)
- [テーマの開発](#)
- [Vivliostyle Theme の仕様](#)

• 運用ガイドライン

- [公式 Theme の採用](#)
- [Vivliostyle Themes ギャラリー](#)

公式 Theme の採用

汎用的で多くのユーザが使うと思われる Theme は公式 Theme として採用します。公式 Theme は [vivliostyle/themes](https://github.com/vivliostyle/themes) レポジトリ (<https://github.com/vivliostyle/themes>) で管理され、`@vivliostyle/theme-*` という名前で npm package として公開されます。

採用条件

公式 Theme として採用する条件は以下の通りです。

- Theme のデザインと主な用途が、既存の公式 Theme と異なっていること
- Theme の仕様を満たしていること
- Contributor License Agreement に署名すること
- Code of Conduct に従うこと

採用方法

[vivliostyle/themes](https://github.com/vivliostyle/themes) レポジトリ (<https://github.com/vivliostyle/themes>) に pull request を送ってください。

- Theme 名は `@vivliostyle/theme-*` とすること
- ソースコードを `packages/@vivliostyle/theme-*/` に含めること

公式 Theme に採用された場合、[vivliostyle/themes](https://github.com/vivliostyle/themes) レポジトリ (<https://github.com/vivliostyle/themes>) のメンテナとして以下の権限が与えられます。

- レポジトリへの push
- issue のトリアージ
- pull request の review と merge

Vivliostyle Themes ギャラリー

npm packageとして公開されている Vivliostyle Themes のギャラリーです。新しく公開されたテーマは見つけ次第掲載していますが、掲載漏れを見つけた場合は [issue](https://github.com/vivliostyle/themes/issues) (https://github.com/vivliostyle/themes/issues) や [pull request](#) をお願いします。

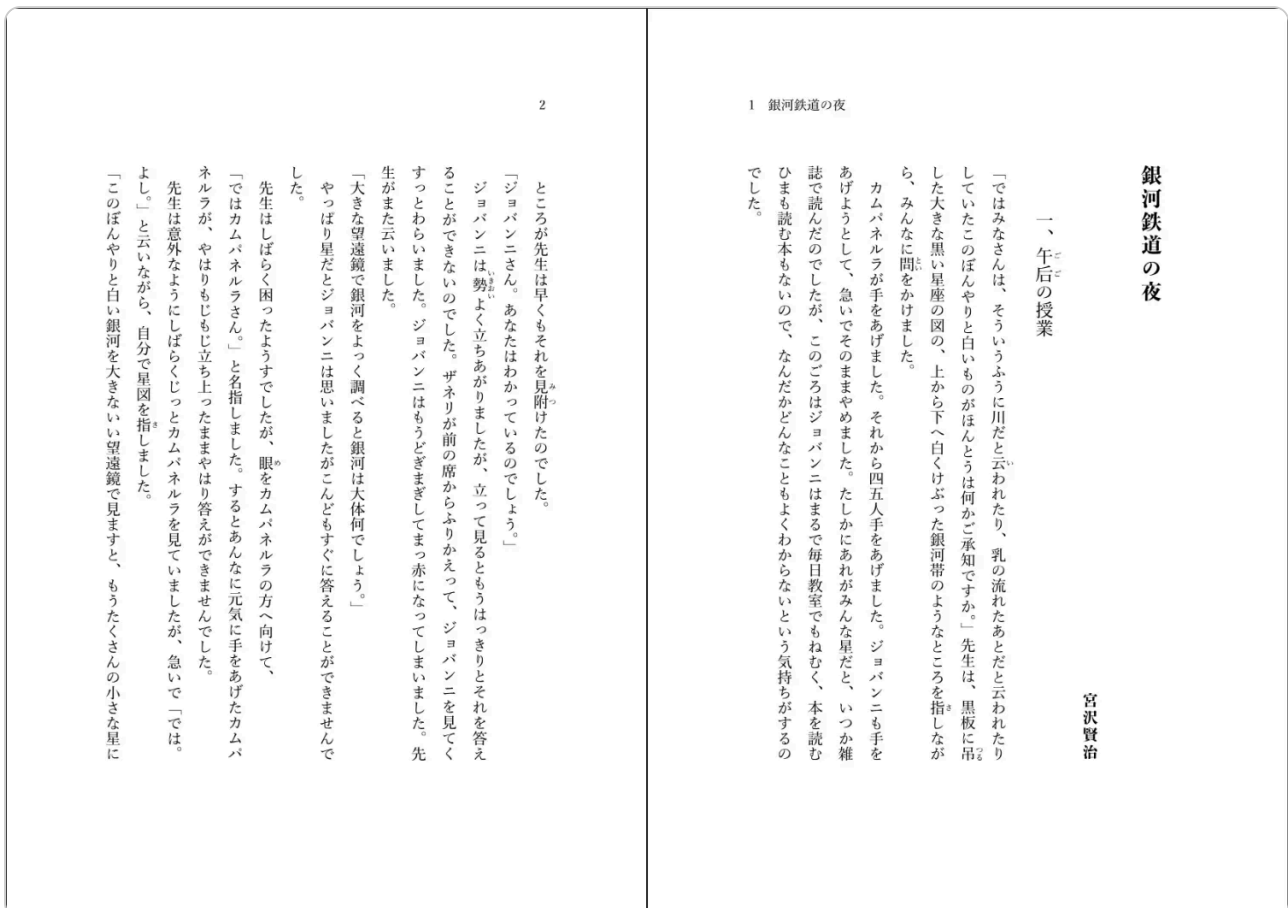
公式の Vivliostyle Themes

現在、6つの公式のテーマがあります。

[@vivliostyle/theme-bunko](https://www.npmjs.com/package/@vivliostyle/theme-bunko) (https://www.npmjs.com/package/@vivliostyle/theme-bunko)

日本語の縦書き小説などに向いています。ルビ、縦中横にも対応しています。CSS変数で行数・文字数、ページヘッダ等をカスタマイズ可能です。

- [README \(CSS変数一覧\)](https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-bunko#readme) (https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-bunko#readme)



theme-bunkoのサンプル画像。原稿は「銀河鉄道の夜」。

@vivliostyle/theme-slide (<https://www.npmjs.com/package/@vivliostyle/theme-slide>)

スライド資料などに向いています。表紙ページと一般ページでスタイルが変わります。CSS変数で配色、レイアウト、カバーページのスタイル等をカスタマイズ可能です。

- README (CSS変数一覧) (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-slide#readme>)

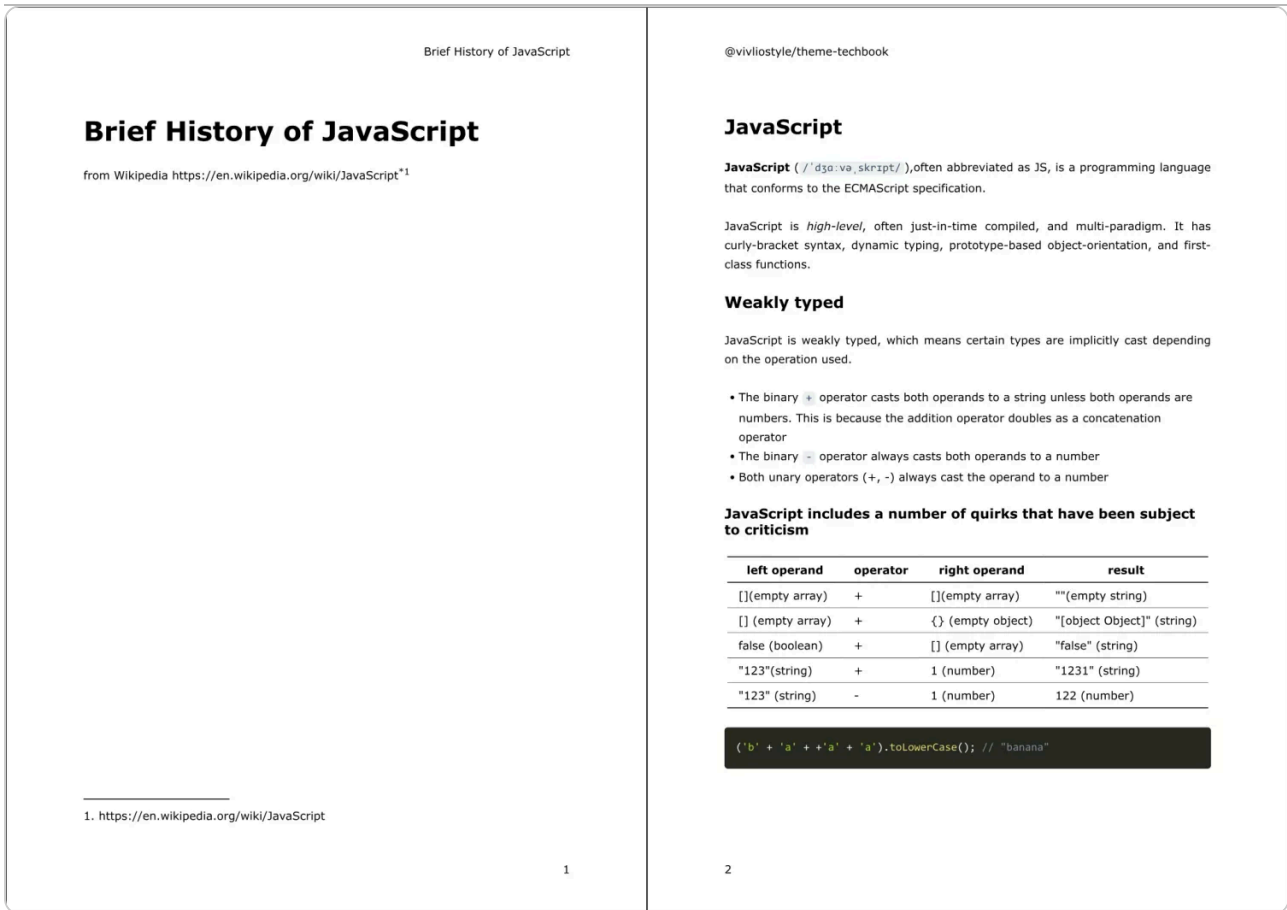


theme-slideのサンプル画像。左は表紙ページで背景が青く、右は一般ページで背景が白い。

@vivliostyle/theme-techbook (<https://www.npmjs.com/package/@vivliostyle/theme-techbook>)

印刷を意識した技術書（小口・ノドの余白調整など）。ソースコードや目次にも対応しています。CSS変数で配色、ページヘッダ・フッタ、画像解像度等をカスタマイズ可能です。

- README (CSS変数一覧) (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-techbook#readme>)



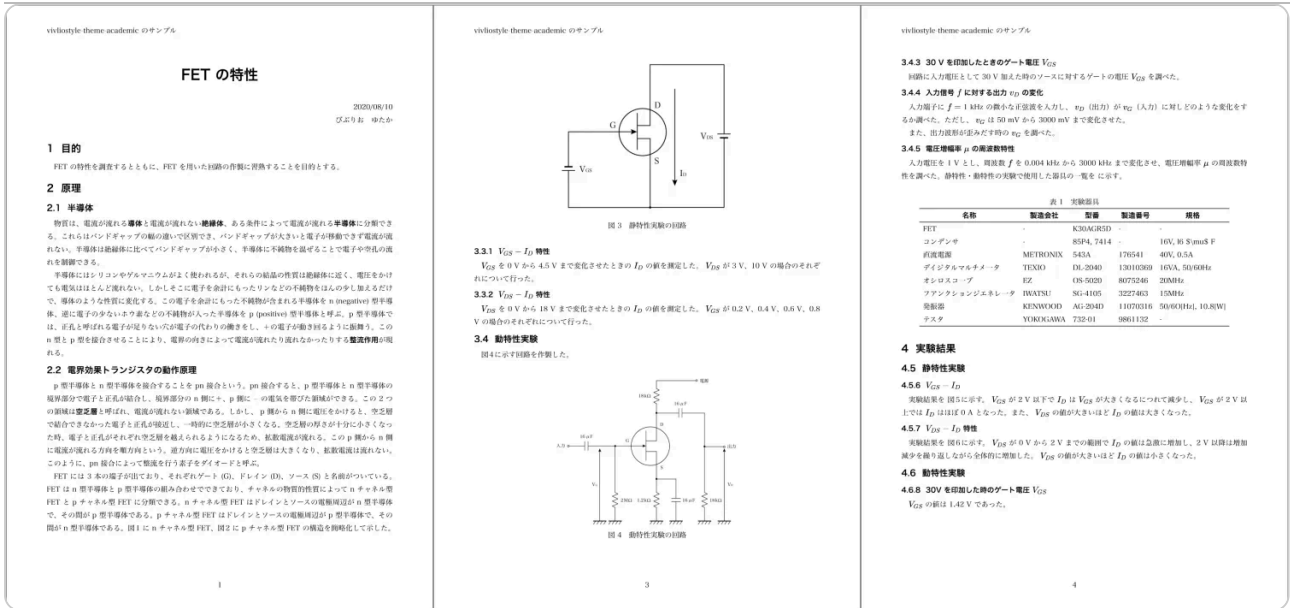
theme-techbookのサンプル画像。テキストのほか、表、ソースコードが載っている。

@vivliostyle/theme-academic (<https://www.npmjs.com/package/@vivliostyle/theme-academic>)

学生が書くレポートなどに向いています。自動で章・節番号がつけます。CSS変数で図のサイズやフレーム要素のスタイルをカスタマイズ可能です。

- [README \(CSS変数一覧\)](https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-academic#readme)

e)



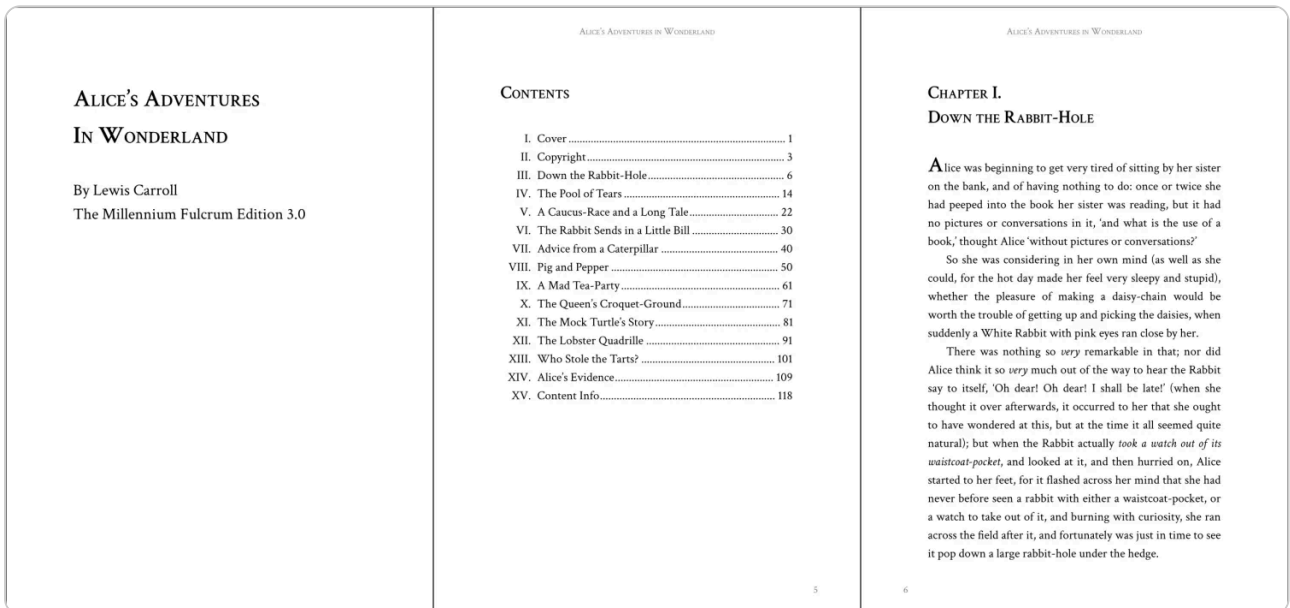
theme-academicのサンプル画像。テキストのほか、図、表が載っている。

[@vivliostyle/theme-gutenberg](https://www.npmjs.com/package/@vivliostyle/theme-gutenberg) (https://www.npmjs.com/package/@vivliostyle/theme-gutenberg)

英語の横書き小説などに向いています。デフォルトの `theme.css` のほかに、3種のCSSバリエーション (`alice.css`, `fang.css`, `sherlock.css`) があります。

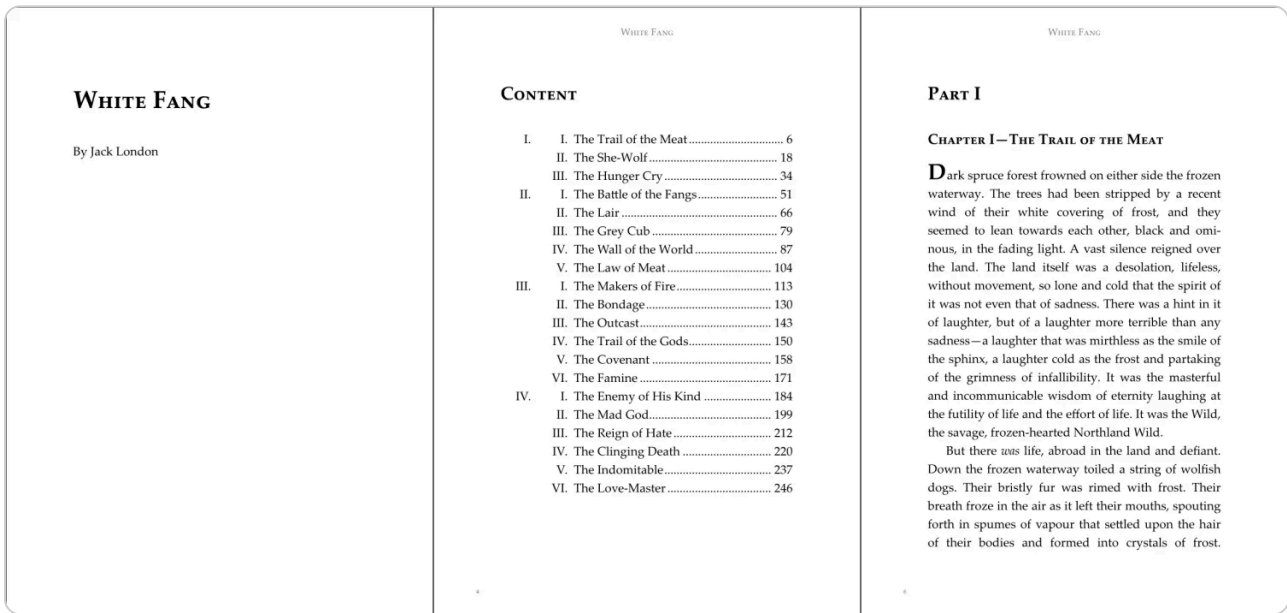
- README (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-gutenberg#readme>)

alice.css



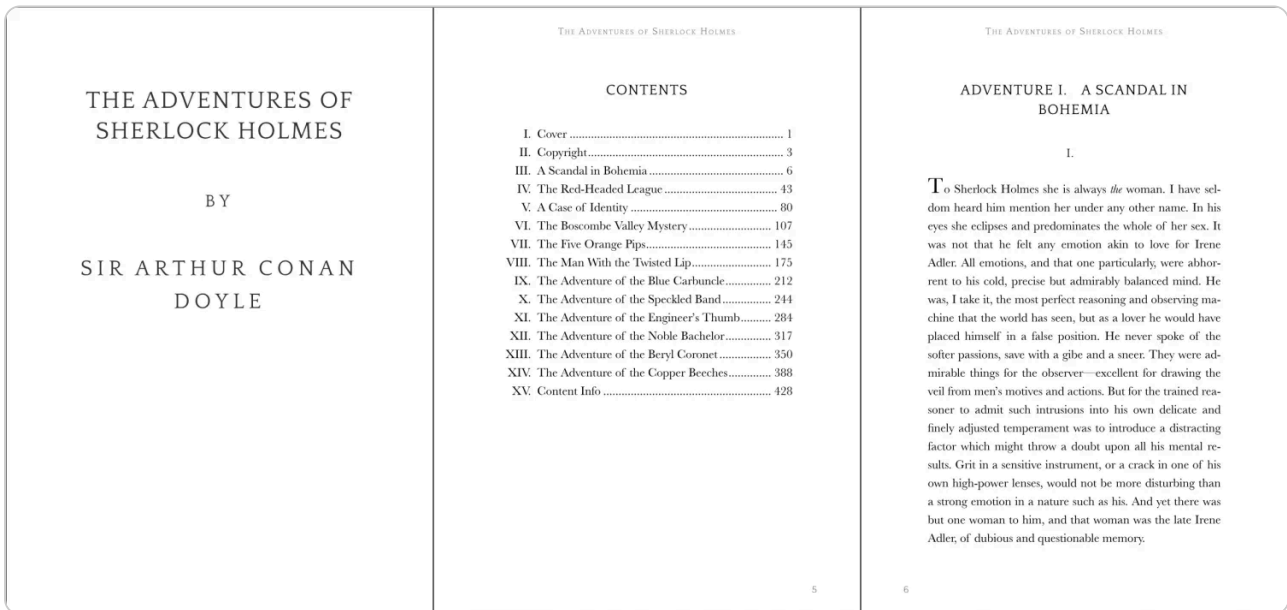
theme-gutenbergのサンプル画像。原稿は「Alice's Adventures in Wonderland」。

fang.css



theme-gutenbergのサンプル画像。原稿は「White Fang」。

sherlock.css



theme-gutenbergのサンプル画像。原稿は「The Adventures of Sherlock Holmes」。

@vivliostyle/theme-epub3j (<https://www.npmjs.com/package/@vivliostyle/theme-epub3j>)

電書協 EPUB3 制作ガイド (<http://ebpaj.jp/counsel/guide>) 準拠の EPUB を作るためのテーマです。日本語縦書きの EPUB 出版物に適しています。

- README (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-epub3j#readme>)



theme-epub3j のサンプル画像。

非公式の Vivliostyle Themes

vivliostyle-theme-dnd-5e-phb (<https://www.npmjs.com/package/vivliostyle-theme-dnd-5e-phb>)

作成者：oldgeese

D&D 5e PHB theme for Vivliostyle

VIVLIOSTYLE THEME D&D 5E PHB



orem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

HEADER2 ヘッダー2

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

HEADER3 ヘッダー3

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

HEADER4 ヘッダー4

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

HEADER5 ヘッダー5

Lorem ipsum dolor sit amet, consectetur adipiscing elit,

Header6 ヘッダー6

strong, 太文字,
italic, イタリック,

親譲りの無鉄砲で小供の時から損ばかりしている。小学校に居る時分学校の二階から飛び降りて一週間ほど腰を抜かした事がある。なぜそんな無闇をしたと聞く人があるかも知れぬ。別段深い理由でもない。新築の二階から首を出していたら、同級生の一人が冗談に、いくら威張っても、そこから飛び降りる事は出来まい。弱虫やーい。と囃したからである。小使に負ぶさって帰って来た時、おやじが大きな眼をして二階ぐらいから飛び降りて腰を抜かす奴があるかと云ったから、この次は抜かさずに飛んで見せませと答えた。(青空文庫より)

UNORDERED LIST

- a horizontal rule before a ul will give it the style of unordered list without a bullet.

a horizontal rule before a ul will give it the style of unordered list without a bullet.

IMAGES



placeholder

CLASS A FEATURES

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

TIME TO DROP KNOWLEDGE

Use notes to point out some interesting information.
Tables and lists both work within a note.

TIME TO DROP KNOWLEDGE

Use notes to point out some interesting information.
Tables and lists both work within a note.

THE NOTARY

Level	Proficiency Bonus	Features	Sixgun Poisoner
1st	+2	Gunpowder Torturer	+1
2nd	+2	Pharmaceutical Gunslinger	+2
3rd	+2	Gunslinger Corruptor	+2
4th	+2	Ability Score Improvement, Demonic Anthropology	+3
5th	+3	Torque Interfacer	+4
6th	+3	Ability Score Improvement, Mathematical Pharmacist	+4
7th	+3	Sixgun Poisoner	+4
8th	+3	Ability Score Improvement, Spiritual Biochemistry	+4
9th	+4	—	+4
10th	+4	Consecrated Biochemistry	+4
11th	+4	Torque Interfacer	+4
12th	+4	Ability Score Improvement, Ritual Astronomy	+4
13th	+5	Plasma Outlaw	+4
14th	+5	Ability Score Improvement, Plasma Outlaw	+5
15th	+5	Civil Alchemy	+6
16th	+5	Ability Score Improvement, Spell Analyst	+6
17th	+6	—	+6
18th	+6	Exo Interfacer	+6
19th	+6	Ability Score Improvement, Police Cultist	+6
20th	+6	Nuclear Illusionism	+6

HIT POINTS

Hit Dice: 1d12 per level

[vivliostyle-theme-thesis](https://www.npmjs.com/package/vivliostyle-theme-thesis) (<https://www.npmjs.com/package/vivliostyle-theme-thesis>)

作成者：yamasy1549

Academic theme (thesis)

[vivliostyle-akashi-kosen-bulletin](https://www.npmjs.com/package/vivliostyle-akashi-kosen-bulletin) (<https://www.npmjs.com/package/vivliostyle-akashi-kosen-bulletin>)

作成者：yamasy1549

明石高専の年報用スタイル (非公式)

Vivliostyle Theme の仕様

Vivliostyle Theme を npm package として公開する場合は以下の仕様に従ってください。 [create-vivliostyle-theme](https://github.com/vivliostyle/themes/tree/master/packages/create-vivliostyle-theme) (https://github.com/vivliostyle/themes/tree/master/packages/create-vivliostyle-theme) を使うと、この仕様に従ったテーマを簡単に作成することができます。

テーマ名

テーマの名前は自由に決められます。ただし、以下の点を考慮することを推奨します。

- 用途が明確な場合は、用途を表す端的な用語を含める
- 変更可能なスタイルに関する用語を避ける
 - 出版物のサイズや組み方向などは、将来的には Vivliostyle Pub などユーザーが簡単に変更できるようになる予定です。したがって、これらの変更可能なスタイルに関する用語を使うことは避けてください。
 - 避けるべき名前の例：vivliostyle-theme-a4book、vivliostyle-theme-tategaki など

テーマに含めるスタイル

テーマは、出版物の特定の部分のみ（図表のみ、脚注のみなど）のスタイルを含んだものではなく、出版物全体のスタイルを含めるものとします。

以下に、テーマに含めることが望ましい代表的なスタイルを示します。 [create-vivliostyle-theme](https://github.com/vivliostyle/themes/tree/master/packages/create-vivliostyle-theme) (https://github.com/vivliostyle/themes/tree/master/packages/create-vivliostyle-theme) で作成したテーマの雛形はこれらのスタイルを含んでいます。

- ページ（余白、柱など）
- 見出し
- 段落
- 数式
- 図とキャプション
- 表とキャプション
- ソースコード
- 脚注
- ルビ

ディレクトリ構造

```
vivliostyle-theme-mybook/  
├─ LICENSE  
├─ README.md  
├─ package.json  
├─ example/  
│   └─ ...  
│       └─ default.md  
├─ theme.css  
└─ vivliostyle.config.js
```

`create-vivliostyle-theme` (<https://github.com/vivliostyle/themes/tree/master/packages/create-vivliostyle-theme>) でテーマの雛形を作成すると、上のようなファイルが自動生成されます。次の節からは代表的なファイルの詳細を見ていきます。

package.json

```
{  
  "name": "vivliostyle-theme-mybook",  
  "author": "John Doe <john@example.com>",  
  "main": "theme.css",  
  "keywords": [  
    // 以下で説明  
    "vivliostyle",  
    "vivliostyle-theme"  
  ],  
  "vivliostyle": {  
    // 以下で説明  
    "theme": {  
      "style": "./theme.css",  
      "name": "Mybook",  
      "author": "John Doe",  
      "category": "novel",  
      "topics": ["paperback"]  
    }  
  }  
}
```

`vivliostyle-theme-scripts` (<https://github.com/vivliostyle/themes/tree/master/packages/vivliostyle-theme-scripts>) を使うと、作成したテーマの `package.json` が仕様に従っているかをチェックできます。

```
$ vivliostyle-theme validate
```

keywords プロパティ

任意。

このプロパティの値に `"vivliostyle-theme"` を含めておくと、[Create Book](https://github.com/vivliostyle/create-book) (<https://github.com/vivliostyle/create-book>) を使って出版物を作る際、利用可能なテーマの一覧にあなたのテーマが表示されます。ただし、そのためにはテーマを npm package として公開している必要があります。

テーマは Vivliostyle に関連する npm package であるため、`"vivliostyle"` も含めておくとよいでしょう。

vivliostyle.theme プロパティ

style プロパティ

必須。テーマで使うメインの CSS を指定します。

package.json のトップレベルに `style` や `main` を指定することもできます。 `vivliostyle.theme.style` と同じ意味を持ちますが、優先順位は `vivliostyle.theme.style` > `style` > `main` です。

```
{
  "main": "theme.css",
  "style": "theme.css"
}
```

author プロパティ

必須。

category プロパティ

任意。このプロパティは、あなたのテーマを初めて使おうとするユーザーがテーマの主な用途を知るためのヒントになります。テーマにもっともあてはまるものを以下のリストから選んでください。

- `"novel"`
- `"magazine"`
- `"journal"`
- `"report"`
- `"misc"`

なお、このリストは今後更新される場合があります。

topics プロパティ

任意。テーマの用途を `category` プロパティよりも具体的に説明したい場合、ここにリストアップして記述しておくといでしょう。

example/

example ディレクトリには、テーマの端的なサンプルとなるような Markdown ファイルを 1 つ以上含めてください。ファイル名は任意です。

Markdown ファイルでは VFM (Vivliostyle Flavored Markdown) (<https://vivliostyle.github.io/vfm/#/vfm>) が利用できません。

theme.css

テーマのスタイルを定義したスタイルシートを含めてください。CSS カスタムプロパティ (CSS 変数) を活用することで、ユーザーがテーマのスタイルを柔軟にカスタマイズできるようにすることを推奨します。

テーマには複数のスタイルシートを含めることができます。たとえば、デフォルトの `theme.css` 以外に以下に示す用途を想定した種類のスタイルシートを用意する使い方などがあります。

- `theme_print.css`: PDF として出力する場合や、Vivliostyle Viewer で閲覧する場合のスタイルを定義
- `theme_screen.css`: HTML などの webpub 形式で出力する場合のスタイルを定義

Vivliostyle CLI でデフォルト以外のスタイルシートを指定する場合、以下のように `specifier` と `import` プロパティを設定します。

```
theme: {
  specifier: 'vivliostyle-theme-mybook',
  import: 'theme_print.css',
},
```

vivliostyle.config.js

```
/** @type {import('@vivliostyle/cli').VivliostyleConfigSchema} */
module.exports = {
  language: 'en',
  theme: ['node_modules/@vivliostyle/theme-base', '.'],
  entry: ['example/default.md'],
  workspaceDir: '.vivliostyle',
  output: [
    'book.pdf',
    {
      path: './book',
      format: 'webpub',
    },
  ],
};
```

example内のMarkdownファイルから出版物を生成するための設定ファイルです。開発者がテーマを開発する際に使うほか、テーマを使うユーザーがサンプル出版物を生成する際にも使います。

Themeの開発

Themeを作成する

create-vivliostyle-themeによる雛形生成

create-vivliostyle-theme CLIを使って、テーマプロジェクトの雛形を生成します。

```
npm create vivliostyle-theme <theme-name>
```

対話形式で以下の項目を入力します。

項目	説明
description	テーマの説明
author	作者名
email	メールアドレス
license	ライセンス (MIT、Apache-2.0等)
category	テーマの分類 (novel / magazine / journal / report / misc)

実行後、vivliostyle-theme-<theme-name> ディレクトリが生成されます。

生成されるファイル構成

```
vivliostyle-theme-<name>/
├── .gitignore
├── package.json          # パッケージ定義 (vivliostyle.theme 設定を含む)
├── README.md            # テーマの説明・使い方
├── theme.css            # テーマのメイン CSS
├── vivliostyle.config.js # プレビュー用の設定ファイル
└── example/
    └── default.md        # サンプル原稿 (VFM 形式)
```

各ファイルの役割:

ファイル	役割
<code>package.json</code>	テーマのメタ情報。 <code>vivliostyle.theme</code> プロパティでテーマ名・作者・メインCSS・カテゴリを定義
<code>theme.css</code>	テーマのスタイル定義本体。theme-baseの <code>@import</code> とCSS変数のカスタマイズが初期設定済み
<code>vivliostyle.config.js</code>	<code>vivliostyle preview</code> でテーマの動作確認に使用。entryに <code>example/default.md</code> を指定
<code>example/default.md</code>	テーマ適用例を示すサンプルMarkdown。VFM (https://vivliostyle.github.io/vfm/#/vfm)記法に対応

雛形をカスタマイズする

theme.cssの編集

生成直後の `theme.css` には theme-base の全モジュールインポートとCSS変数のカスタマイズ例が含まれています。

```

/* theme-base の全モジュールをインポート */
@import url(../@viviostyle/theme-base/theme-all.css);

/* コードハイライト (Prism) を追加 */
@import url(../@viviostyle/theme-base/css/lib/prism/base.css);
@import url(../@viviostyle/theme-base/css/lib/prism/theme-okaidia.css);

:root {
  /* 基本スタイル */
  --vs-font-family: 'Times New Roman', serif;
  --vs-font-size: 12px;
  --vs--heading-line-height: 1.3;
  --vs--h1-font-size: 2.5em;

  /* 脚注 */
  --vs-footnote--call-content: '[' counter(footnote) ']';

  /* ページレイアウト */
  --vs-page--mbox-content-bottom-center: counter(page);
  --vs-page--mbox-content-top-left: env(doc-title);

  /* 目次 */
  --vs-toc--marker-margin-inline: 8rem;
}

```

独自のスタイルを追加するには、このファイルの末尾にルールを記述します。ページサイズの設定例:

```

@page {
  size: A5;
  margin: 20mm 15mm;
}

```

theme-baseのモジュール活用

theme-baseは機能ごとにモジュール分割されています。全モジュールが不要な場合は、`theme-all.css` の代わりに必要なモジュールのみをインポートできます。

```

/* 基本モジュールのみ */
@import url(../@viviostyle/theme-base/css/common/meta-properties.css);
@import url(../@viviostyle/theme-base/css/common/reset.css);
@import url(../@viviostyle/theme-base/css/common/basic.css);

/* 必要な機能モジュールを追加 */
@import url(../@viviostyle/theme-base/css/partial/toc.css);
@import url(../@viviostyle/theme-base/css/partial/footnote.css);

```

利用可能なモジュール一覧:

カテゴリ	モジュール	CSS変数プレフィックス
common	<code>meta-properties.css</code> — ドキュメント全体のメタプロパティ	<code>--vs-</code>
common	<code>reset.css</code> — CSSリセット	—
common	<code>basic.css</code> — 基本HTMLタグのスタイル	<code>--vs--</code>
partial	<code>crossref.css</code> — 図表・引用の相互参照	<code>--vs-crossref--</code>
partial	<code>endnote.css</code> — 後注	<code>--vs-endnote--</code>
partial	<code>footnote.css</code> — 脚注	<code>--vs-footnote--</code>
partial	<code>footnote-external-link.css</code> — 外部リンクの脚注化	<code>--vs-footnote--</code>
partial	<code>page.css</code> — ページメディア	<code>--vs-page--</code>
partial	<code>section.css</code> — 見出し番号・節参照	<code>--vs-section--</code>
partial	<code>toc.css</code> — 目次	<code>--vs-toc--</code>
partial	<code>utility-classes.css</code> — ユーティリティクラス	—
lib	<code>prism/base.css</code> — コードハイライト基盤	<code>--vs-prism--</code>
lib	<code>prism/theme-prism.css</code> — Prismデフォルトテーマ	<code>--vs-prism--</code>
lib	<code>prism/theme-okaidia.css</code> — Okaidiaテーマ	<code>--vs-prism--</code>

詳細は `theme-base` の README (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-base#available-modules-and-css-variables>) を参照してください。

CSS変数のオーバーライド

`theme-base` や各テーマが公開する CSS 変数を `:root` で上書きすることで、テーマをカスタマイズできます。

```

:root {
  /* フォント設定 */
  --vs-font-family: 'Noto Serif JP', serif;
  --vs-font-size: 10.5pt;

  /* 見出し */
  --vs--h1-font-size: 2em;
  --vs--heading-line-height: 1.4;

  /* 相互参照のカウンタスタイル */
  --vs-crossref--counter-style: upper-roman;

  /* ページヘッダ・フッタ */
  --vs-page--mbox-content-top-left: env(pub-title);
  --vs-page--mbox-content-top-right: env(doc-title);
  --vs-page--mbox-content-bottom-center: counter(page);
}

```

実際のテーマでの活用例として、`theme-techbook` (<https://github.com/vivliostyle/themes/tree/main/packages/@vivliostyle/theme-techbook>)はテーマ固有のCSS変数 (`--vs-theme--*`) を公開しています:

```

:root {
  --vs-theme--anchor-color-body: #3498db;
  --vs-theme--blockquote-color-bg: #ecf0f1;
  --vs-theme--inline-code-color-bg: #ecf0f1;
  --vs-theme--image-resolution-for-figure-image: 300dpi;
  --vs-theme--page-top-left-content: env(pub-title);
  --vs-theme--page-bottom-content: counter(page);
}

```

example/ディレクトリの編集

`example/` にはテーマの適用例を示すサンプルMarkdownを配置します。

- 最低1つのMarkdownファイルが必要です
- VFM (Vivliostyle Flavored Markdown) (<https://vivliostyle.github.io/vfm/#/vfm>)記法が利用できます
- テーマが対応する主要なスタイル (見出し、コードブロック、脚注、画像、数式等) を網羅する内容を推奨します

プレビューで動作確認:

```
npm run preview
```

Themeを公開する

事前検証

公開前に `vivliostyle-theme-scripts validate` を実行して、パッケージの妥当性を検証します。

```
npm run validate
```

検証項目:

チェック	種別	内容
スタイルロケータ	エラー	<code>vivliostyle.theme.style</code> 、 <code>style</code> 、 <code>main</code> のいずれかが設定されていること
作者情報	警告	<code>vivliostyle.theme.author</code> または <code>author</code> が設定されていること

プレビュー確認

```
npm run preview
```

`vivliostyle.config.js` の設定に基づき、`example/` のサンプル原稿にテーマを適用した状態でプレビューが表示されます。

package.jsonの必須フィールド

```
{
  "name": "vivliostyle-theme-<name>",
  "main": "theme.css",
  "keywords": ["vivliostyle", "vivliostyle-theme"],
  "vivliostyle": {
    "theme": {
      "name": "Theme Display Name",
      "author": "Author Name",
      "style": "theme.css",
      "category": "misc",
      "topics": []
    }
  }
}
```

フィールド	必須	説明
<code>vivliostyle.theme.style</code>	✓	メインCSSファイルのパス
<code>vivliostyle.theme.author</code>	✓	テーマの作者名
<code>vivliostyle.theme.name</code>	—	テーマの表示名
<code>vivliostyle.theme.category</code>	—	novel / magazine / journal / report / misc
<code>vivliostyle.theme.topics</code>	—	テーマの用途を示すトピックの配列
<code>keywords</code>	—	"vivliostyle" と "vivliostyle-theme" を含めることを推奨

詳細な仕様は [Vivliostyle Theme の仕様](#) を参照してください。

npm への公開

```
npm publish
```

公開後、テーマは以下から検索可能になります:

- [GitHub Topics: vivliostyle-theme](https://github.com/topics/vivliostyle-theme)
- [npm: vivliostyle-theme](https://www.npmjs.com/search?q=keywords%3Avivliostyle-theme)

公式テーマとして提案する

自作したテーマは、ぜひVivliostyle公式テーマとして提案してください。詳細は[公式Themeの採用を参照してください](#)。