

人工言語イジエール語の 辞書をVivliostyleで作る

2021/11/14 Zaslou

私について

●Zaslon(ざすらん)

- 本職：機械エンジニア。ITエンジニアではない
- html, css, php はwordpressのサイトを作るにあたって学習した
 - サイトのテーマはもともと静的サイトとして作成していたころのhtmlとcssを流用して作成したもの

• 辞書データの形式：json

- OTM-jsonという形式のjsonファイル
- これをphpで読みだしてオンラインで検索するオンライン辞書を作った



このオンライン辞書から検索機能をなくしたものをベースにしようと考えたのが、css組版をしようとした理由



人工言語とは

- 人間が設計して作っている言語のこと
 - 自然言語も人間が作っていると言えるが、特定の誰かが設計しているわけではない
- 想定用途は言語によって異なる
 - 国際補助語を目指している例（エスペラントなど）や、フィクションの小道具（エルフ語やナヴィ語など）などがある
 - 小道具としては文字だけが制作される例も散見される

人工言語イジェール語について

大文字					小文字				
∇	V	ʒ	†	L	o	u	c	r	L
l	Λ	∩	∩	F	i	n	h	∩	f
∩	∩	Z	P	∩	∩	d	z	b	∩
Δ	∩	∩	∩	∩	∩	h	∩	b	∩

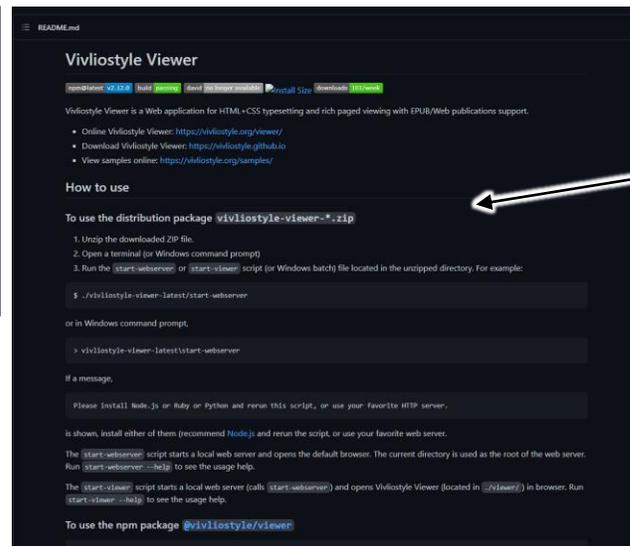
- Zaslou (私) が個人で作っている人工言語
 - <https://zaslon.info/idyer/>
 - 記録では2008年ごろに原型を作り始めている
- 表記法：独自のアルファベットによる表記
 - 基本的にはラテンアルファベットと似ているので、組み方の注意点もラテンアルファベットと同じ（未確定）。
 - ドイツ語のように単語が長くなりがち
 - 例: unodhuho∩ud∩L∩∩∩u∩L (akerzasedargunevadu)
プリズム (aker+sase+tar+kune+fadu:光線+分岐+三+角+柱)
 - 現状は機械的にある程度妥当と思われる場所に<wbr>タグを入れることでハイフネーションの代わりとしている

Vivliostyleを使う上で苦労した点

- どうやって使い始めるのかわかりにくい
 - Vivliostyleというアプリケーションの使い方に重点が置かれていて、アプリケーション上の組版作業のやり方はわからない



これを実施してもViewerがローカルで使えるようになるだけで、どのようにデータを作ればいいのかはわからない



↓ できることの紹介

Vivliostyleを使う上で苦労した点

- ユーザとして本当に必要な情報はおそらくここ

The screenshot shows the Vivliostyle website with the following navigation links: 使ってみる, サンプル, ダウンロード, ドキュメント, FAQ, ブログ. The main content area is titled 'ドキュメント' and contains a message: '開発者が困ったとき、迷ったとき頼りになる、さまざまな文書をまとめました。『今後の開発予定』ではGitHubの当該Issueにジャンプします。' Below this, there are several sections with links:

- 📖 ユーザーガイド
- Vivliostyle Viewer
 - 使い方
 - サポートされている文書の種類
 - URLパラメータのオプション
 - 表示するHTML文書の指定
 - EPUB
 - Web出版物 (複数HTML文書)
 - 目次パネル
 - 見開きビューモード
 - スタイルシートの追加
 - 印刷/PDFとして出力
 - サポートされているCSSの機能
- Vivliostyle CLI
 - インストール
 - HTMLからPDFを生成
 - スタイルシートの追加の指定
 - EPUBからPDFを生成
 - MarkdownからPDFを生成
 - 組版結果のプレビュー
 - Vivliostyle Themesについて
 - 構成ファイルvivliostyle.config.js
 - 印刷用PDF (PDF/X-1a形式)の生成
 - PDFの「しおり」(Bookmarks)の生成
 - 目次の作成
 - Web出版物(webpub)
 - その他のオプション
- Create Book

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。

Values

- Supported CSS-wide keywords: `inherit`
 - `initial` and `unset` are not supported.
- Supported length units: `em`, `ex`, `ch`, `rem`, `vw`, `vh`, `vmin`, `vmax`, `vi`, `vb`, `cm`, `mm`, `q`, `in`, `pc`, `pt`, `px`.
- Sizing keywords: `min-content`, `max-content`, `fit-content`
- Supported color values
 - Basic color keywords
 - RGB color values, RGBA color values
 - 'transparent' color keyword
 - HSL color values, HSLA color values
 - Extended color keywords
 - 'currentColor' color keyword
- Attribute references: `attr()`
 - Only supported in values of `content` property.
 - Only 'string' and 'url' types are supported.
- Cross references: `target-counter()` and `target-counters()`
 - Only supported in values of `content` property.
- `calc()` function
 - `min()` and `max()` functions can be used inside `calc()` function.
 - Limitation: Percentage value in `calc()` is not calculated correctly.
- `content()` function
- `string()` function (Named Strings)
- `env()` function
 - Implemented only `env(pub-title)` and `env(doc-title)` that are not yet defined in the css-env draft spec but useful for making page header.
 - `env(pub-title)`: publication title = EPUB, Web publication, or primary entry page HTML title.
 - `env(doc-title)`: document title = HTML title, which may be chapter or section title in a publication composed of multiple HTML documents

The screenshot shows the Vivliostyle Docs website with the following content:

- 日本語
- 📖 ユーザーガイド
- Vivliostyle Viewer
 - 使い方
 - サポートされている文書の種類
 - URLパラメータのオプション
 - 表示するHTML文書の指定
 - EPUB
 - Web出版物 (複数HTML文書)
 - 目次パネル
 - 見開きビューモード
 - スタイルシートの追加
 - 印刷/PDFとして出力
 - サポートされているCSSの機能
- Vivliostyle CLI
 - Create Book
- 🌟 コントリビューションガイド
- Vivliostyle.js
- Vivliostyle CLI
- Vivliostyle Themes
- VIM
- 📖 リファレンス
 - サポートする CSS 機能
 - Core API リファレンス
- Vivliostyle CLI
- Create Book
- VIM
- Vivliostyle Themes
 - Theme Design Guideline
- リリース
- Vivliostyle Releases
- コミュニティ
 - Vivliostyle コミュニティ
 - Awesome Vivliostyle

この例のように、設定パネルからスタイルの設定を行うと、ユーザースタイルのCSS内のコメント `/* vivliostyle */` と `/* vivliostyle */` で囲んだ部分に、設定パネルの項目から設定したスタイルのCSSコードが生成され、設定パネルの **CSS Details** に表示されます。ユーザー独自のCSSコードをそれに追加することができます。この例ではページヘッダーとページフッターを追加しています。

印刷/PDFとして出力

ブラウザの印刷・PDF保存機能を利用して、レンダリングされたドキュメントをPDFに変換できます。

例えば、Google Chrome で PDF に出力するには、メニューから「印刷」を開き、出力先として「PDFに保存」、背景のグラフィック ON と指定して「保存」します。

注: 文書の全ページを印刷する場合、設定パネルで **Render All Pages** が On になっていることを確認してください。これが Off の場合、すでに表示されたページしか印刷できず、またページ番号が正しく出力されません。

Vivliostyle CLI

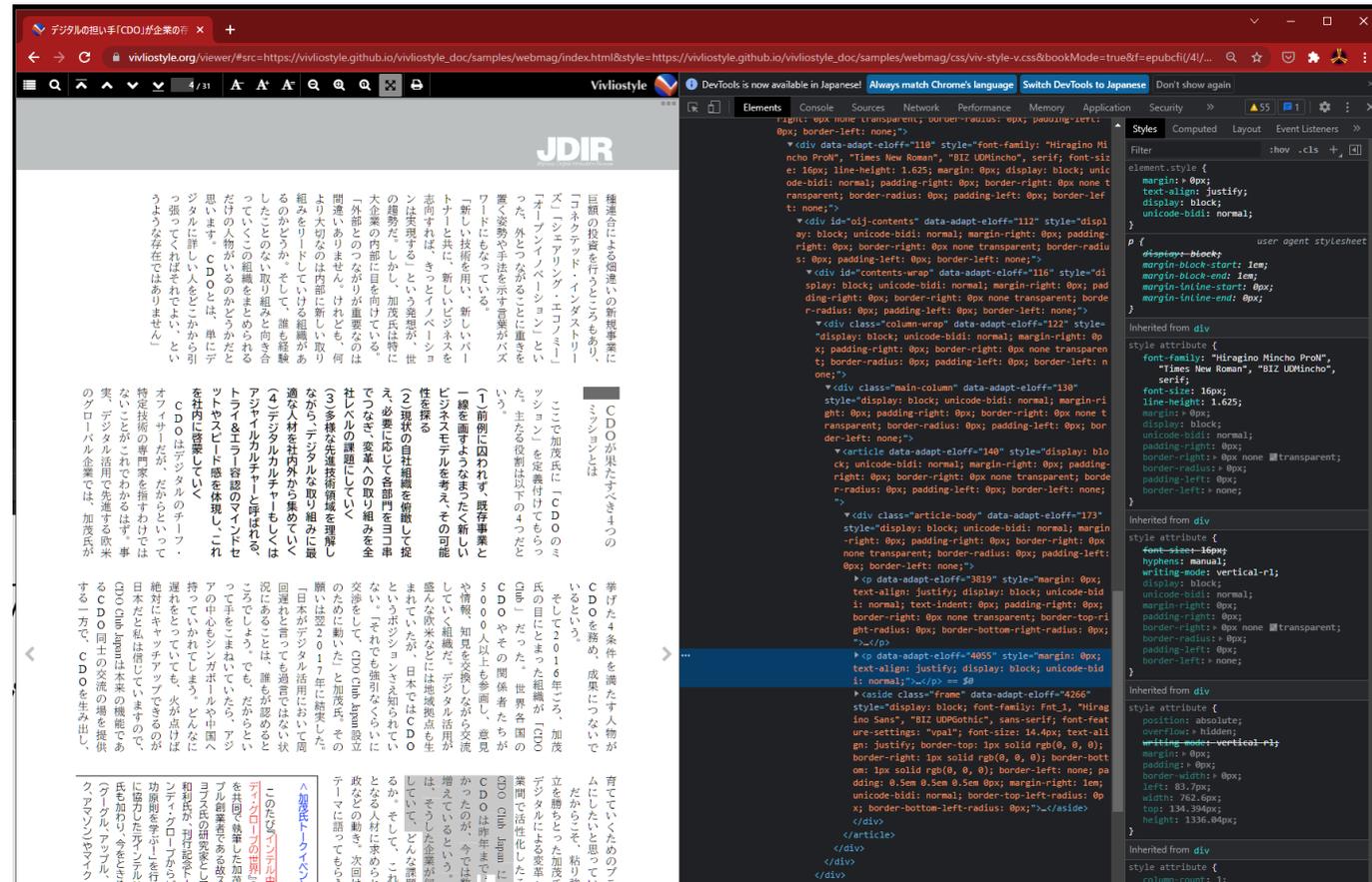
Vivliostyle CLI を使うとコマンドラインからPDFに出力できます。印刷に連したPDF/X-1a形式への変換も可能です。

📖 サポートされているCSSの機能

VivliostyleがサポートするCSSの機能についてはサポートするCSS機能を参照してください。

Vivliostyleを使う上で苦労した点

- 開発者ツールが役に立たないので、サンプルを見てもどう参考にしていいのかわかりにくい
 - GitHub上のソースを見比べればわかるが、かなり根気がいる
 - 同様の問題は自作のhtml+cssでも起こる
 - 何が問題なのか調査しにくい



Vivliostyleに期待すること

●Running head機能の実装

- 爪が完成すれば辞書としての体裁が整う

●ソースと見比べて編集ができる編集機能の拡張

●使い方の説明からサポートしているCSS要素に飛んでほしい

- CSS組版に興味がある時点でCSSとは何かそのものの説明はそこまでいらないうように思うが、ドキュメントリストへのアクセス方法がないと学びようがない

7

《文化》 日常的に用いることはない。《語源》 ru 訳語では「絵文字」であるが、文字の一種ではなく記号の一種とされる。音や語との結びつきを持

Вікіпедія