

# KSS PC BOOK 2022

2021-05-xx 版      KSS PC Club 発行

# 目次

はじめに	3
------	---

---

この部誌を支える技術	4
------------	---

---

# はじめに

このたびは本書をお手に取っていただきありがとうございます。本書は令和4年度けやき祭のためにパソコン部の有志によって作成された部誌です。

古河中等教育学校パソコン部(通称：KSS PC Club)では部員それぞれが主に好きなことをし、競技プログラミングやWeb開発、ゲーム開発など様々なことに取り組んでいますが、本書では部員の興味を持っていることや語りたいことなどについて記事を自由に執筆してもらい、一冊の本に仕上げました。

新型コロナウイルスの影響により、文化祭でパソコン部が作品等を展示するのは2019年以来の3年ぶりであり、久しぶりの作品展示の機会であるのでぜひ特別棟2階奥のパソコン室にもお立ち寄りください。

部誌の制作については今年度からの初の試みであり、自分の言葉で物事を発信する場を設けるという意義があります。このような発信活動の場のひとつである本書を通して、読者のみなさまにも創作・技術に触れる楽しさ、好きなことに接する楽しさを感じてもらえると幸いです。

部長 細島涼雅

## お問い合わせ先

本書に関するお問い合わせは部員まで。

## 免責事項

本書に記載された内容は、情報の提供のみを目的としています。したがって、本書を用いた開発、製作、運用は、必ずご自身の責任及び判断の上で行ってください。これらの情報による開発等の結果について、著者はいかなる責任も負いません。

# この部誌を支える技術

Ryoga.exe

## はじめに

こんにちは、5期生の Ryoga.exe です。この部誌は今年からの初の試みということもあり折角なので(?) 今回はこの部誌を支える技術についてお話ししようかと思います。この部誌については GitHub 上で管理されているため、そのリポジトリも合わせてご覧ください。[1]

## Vivliostyle について

本書は HTML & CSS で組版ができる `Vivliostyle` [2] の `Create Book` [3] というものを使って書かれています。



▲図1: Vivliostyle のロゴ

Vivliostyle は CSS 組版というWeb標準技術をベースにした、自動組版システムのオープンソースプロジェクトです。

CSS組版はHTMLとCSSをベースにしているので、普段から HTML/CSS を扱っている人にとっては Vivliostyle のプロダクトが比較的手軽に感じるでしょう。

## Vivliostyle を選択した理由

まず、複数人で書くことが前提となっていたので原稿等を共有する必要があります。また、見た目などもある程度統一させたいです。そのため、Word 等のソフトウェアは真っ先に候補から外れました。

---

[1] <https://github.com/kss-pc-club/book-2022>

[2] <https://vivliostyle.org/ja/>

[3] <https://docs.vivliostyle.org/#ja/create-book>

そして、本を書く上で新たに多くの記法などを覚えるのは大変かつ、脳のリソースの無駄になります。そこで、Markdown で書いて、スタイルをコードで整えることができ、GitHub 上で管理できるといったことが必要でした。

そのため、Vivliostyle を選択しました。(日本語のドキュメントが充実しているという理由もありました。)

## VFM について

Create Book がサポートする Markdown 方言は、Vivliostyle Flavored Markdown (VFM) です。詳細は公式ドキュメント [4] を参照してください。

今回はこの VFM でサポートされている記法について、ざっくりとご紹介します。

```
---
title: セロ弾きのゴーシュ
---

# セロ弾きのゴーシュ

**宮沢賢治**[^1]

ゴーシュは町の活動写真館でセロを弾く係りでした。
けれどもあんまり上手でないという評判でした。
上手でないどころではなく実は仲間の楽手のなかではいちばん下手でしたから、
いつでも楽長にいじめられるのでした。
ひるすぎみんなは楽屋に円くらんで今度の町の音楽会へ出す
第六{交響曲|こうきょうきょく}の練習をしていました。

![挿絵]('image/fig334.png'){width=300}

[^1]: https://ja.wikipedia.org/wiki/%E5%AE%AE%E6%B2%A2%E8%B3%A2%E6%B2%B
```

基本的には GitHub でサポートされている Markdown の方言である GFM (GitHub Flavored Markdown) の記法は使えます。しかし、上のサンプルにはなにか見慣れない記法がありますね。それぞれ説明します。

## ルビ

`{親文字|ヨミ}` とすることでルビが振れます。便利。梅やむと書いてミライと読ませることもできますね！

---

[4] <https://vivliostyle.github.io/vfm/#/ja/vfm>

## 脚注 (後注)

文章内に `[^n]` を置きその文の後の箇所で `[^n]: hoge` とすると脚注をつけることができます。ちなみに、`@vivliostyle/theme-techbook` のテーマパッケージを使用している場合は `<span class="footnote">hoge</span>` とすると本文 (ページ) の下に注記をつけることができます。

余談ですが、脚注をつける記法は実は GFM で同様にサポートされていたりします。

## Sass を使う

この節では Sass と一緒にビルド/プレビューする方法をご紹介します。CSS を使ってスタイルを当てるのですが CSS よりも Sass の方がいろいろと楽だったりします。

Create Book でプレビューするとき、直接 `.scss` ファイルを読み込めないため、スタイルを調整しながらプレビューするといったことが困難です。

これを解決するために、`npm-run-all` を使います。`run-p` というコマンドがあり、パラレル実行ができるためこれを活用します。npm や yarn でインストールして

```
$ yarn add --dev npm-run-all
```

package.json の script に以下を追加します。

```
"start": "run-p preview watch:scss",  
"preview": "vivliostyle preview",  
"watch:scss": "sass --watch scss:css"
```

これで `yarn start` とすると SCSS のビルドとプレビューが同時にできます。便利。

## GitHub Actions でビルドする

この部誌は GitHub 上で管理されており、各部員がブランチを切り、PR を出す...といった感じで書かれています。

しかし、現状どのような見た目になっているかを確認できない部員もいたりします。そのため、GitHub Actions を活用してビルドし、自動で `publish` ブランチに push されるような仕組みにすることにしました。以下のワークフローを作成するとできます。

`.github/workflows/build.yml`

```
name: Build
on:
  workflow_dispatch:

jobs:
  build:
    name: Build
    runs-on: ubuntu-latest
    env:
      TZ: Asia/Tokyo
      GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN }
    steps:
      - name: Checkout
        uses: actions/checkout@v2.3.4
        with:
          ref: main
      - name: Setup Node
        uses: actions/setup-node@v2.1.5
        with:
          node-version: '14.16.0'
      - name: Get yarn cache directory path
        id: yarn-cache-dir-path
        run: echo "::set-output name=dir::$(yarn cache dir)"
      - name: Cache deps
        uses: actions/cache@v2.1.6
        with:
          path: ${ steps.yarn-cache-dir-path.outputs.dir }
          key: ${ runner.os }-yarn-${ hashFiles('yarn.lock') }
          restore-keys: |
            ${ runner.os }-yarn-
      - name: Install deps
        run: yarn install --frozen-lockfile
      - name: Install ghostscript
        run: |
          sudo apt-get -yqq install libgbm1 ghostscript
          sudo apt install poppler-utils poppler-data
      - name: Build
        run: yarn build
      - name: Build Press-Ready
        run: yarn press-ready
      - name: Deploy
        uses: s0/git-publish-subdir-action@develop
        env:
          REPO: self
          BRANCH: publish
          FOLDER: public
          GITHUB_TOKEN: ${ secrets.GITHUB_TOKEN }
```

`workflow_dispatch` ではなく `main` ブランチ等への `push` をトリガーにしたほうが楽かもしれないですね。

## この部誌をビルドする

前述したとおりこの部誌は GitHub のリポジトリで管理されているのでそれを clone しましょう。

```
$ git clone git@github.com:kss-pc-club/book-2022.git
```

依存関係をインストールし、`yarn build` とするとビルドができます。

```
$ yarn install
$ yarn build
```

`public/book.pdf` が作られます。

入稿用のデータを `yarn press-ready` というコマンドで作ることができますが、ghostscript と poppler-utils のインストールが必要なので注意してください。

## おわりに

ある程度しっかりとした本を既に知っている技術で作れました。

CSS 組版で本を作りたい！ Markdown で本を書きたい！...そんな方はぜひ Vivliostyle で始めてみませんか？



## KSS PC BOOK 2022

---

2022 年 5 月 x 日 初版発行

著 者 KSS PC Club

印刷所 xxxxx

---

© 2022 KSS PC Club