

MarkdownをPDFにする方法

Created on September 10, 2023

Updated on September 10, 2023

Copyright (c) 2023 led-mirage

はじめに

この資料では Markdown ファイルを PDF ファイルに変換する方法について説明します。

目次

- [1. Visual Studio Code の拡張機能を使う方法](#)
 - [1-1. 動作確認](#)
 - [1-2. インストール](#)
 - [1-3. 使い方](#)
 - [1-4. 康熙（こうき）部首の問題](#)
- [2. Node.js の md-to-pdf パッケージを使う方法](#)
 - [2-1. 動作確認](#)
 - [2-2. インストール](#)
 - [2-3. 使い方](#)
- [3. まとめ](#)
- [4. OSS](#)
- [5. 参考リンク](#)

1. Visual Studio Code の拡張機能を使う方法

Visual Studio Code の拡張機能 `Markdown PDF(z)` で Markdown ファイルを PDF ファイルに変換できます。

1-1. 動作確認

以下の動作環境で動作確認をしました。

- Windows 11 Pro 22H2
- Raspberry Pi 4B 4GB
- Visual Studio Code 1.82.0
- Markdown PDF(z) v2.0.3

1-2. インストール

Visual Studio Code の Marketplace で `markdown pdf` と検索すると、先頭に `Markdown PDF` が見つかり、その他同じアイコンの拡張機能があと 2 つほど表示されると思います（2023 年 9 月現在）。

1. Markdown PDF
2. Markdown PDF(z)
3. Markdown PDF(m)

この中でインストールすべきは、2 番めの `Markdown PDF(z)` です。本家は 1 番目の `Markdown PDF` で、あとの 2 つは本家から分岐した Fork なのですが、本家と 3 番目のプログラムは後述する康熙（こうき）部首の問題の問題を含んでいます。また、本家は Windows なら動いたのですが、ラズパイで PDF を生成しようとするエラーが発生しました。これについても `Markdown PDF(z)` では問題なく生成できました。

以上のことから、インストールすべきなのは `Markdown PDF(z)` という結論に至りました。ただ、本家のほうが安心できる、康熙（こうき）部首の問題なんて気にならない、という方は本家を使ってもいいと思います。

1-3. 使い方

エディタ上の Markdown ファイルを右クリックして `Markdown PDF: Export(pdf)` を選択すると PDF ファイルが生成されます。簡単ですね。

`Markdown PDF(z)` は PDF 化のほか、HTML や画像データへの変換もサポートしています。

1-3-1. スタイル

ヘッダー、フッターなどのスタイルの設定は拡張機能の設定画面から行えます。

より詳細なスタイルは CSS ファイルを追加することで調整可能です。CSS ファイル名を github-markdown.css にした場合、.vscode/settings.json ファイルに以下のような記述を追加します。

```
{
  "markdown-pdf.includeDefaultStyles": false,
  "markdown-pdf.styles": [
    "./github-markdown.css"
  ],
  "markdown-pdf.margin.bottom": "2cm",
  "markdown-pdf.margin.top": "2cm",
}
```

GitHub風のスタイルにしたい場合、次のようにします。

1. [ここ※1](#)から CSS をダウンロードし、github-markdown.css ファイルに保存する
2. ファイル中の ".markdown-" という文字列をすべて削除する（空文字で置換）
3. "white-space: pre;" を "white-space: pre-wrap;" に変更する（変更しないとプログラムコードが途中で途切れる）

この方法はネットを検索すると出てくるので、それらを参考にしてください。

※1 <https://cdnjs.cloudflare.com/ajax/libs/github-markdown-css/5.2.0/github-markdown.css>

1-3-2. 改ページ

改ページしたい個所で以下のタグを挿入します。

```
<div class="page"/>
```

1-4. 康熙（こうき）部首の問題

本家および3番めの拡張機能だと、特定の文字が康熙（こうき）部首に変換されてしまいます。康熙部首とは部首を表す特殊な漢字コードで、普通の漢字が誤って康熙部首に変換されてしまうと、アプリケーションによっては文字化けを引き起こしてしまいます。

Word や Excel などの表示では普通の漢字と見分けがつかないこともあるため、一見無害の様にも思えますが、これが2次利用された場合（たとえばデータベースのマスタデータとして登録してしまうなど）、気づかないうちに被害が拡大してしまいます。

例えば「自分用」という単語を本家で PDF に変換すると "自" と "分" が康熙部首になってしまいます。PDF になった「自分用」をエディタなどに貼り付けて、その下に手入力で「自分用」と書いてみれば違いがわかると思います。見た目ではわからない場合、手書きの文字をコピーして文書内を検索してみてください。PDF から貼り付けた文字とは一致しないはずです。

※ 検証に使用した本家 Markdown PDF は v1.5.0 です。

このように康熙部首の問題は根深いものがあるため、個人的には可能な限り排除したほうがいいと考えています。ネットで「康熙部首」で検索すると色々な情報が出てくるので、興味がある方は調べてみてください。

2. Node.js の md-to-pdf パッケージを使う方法

Node.js の md-to-pdf パッケージでも Markdown ファイルを PDF ファイルに変換できます。

実行にはあらかじめ Node.js をインストールしておく必要があります。Node.js のインストールは、Windows であれば[公式サイト](#)からダウンロードしてインストールすればOKです。ラズパイなど、Linux系のOSであれば、コマンドでインストールできると思います。

また、Node.js を直接インストールするのではなく、nvm などのバージョン管理ツールを先にインストールし、nvm 経由で Node.js をインストールしてもいいでしょう。バージョン管理ツールを使うと、Node.js 複数のバージョンを簡単に切り替えることができ便利です。

2-1. 動作確認

以下の動作環境で動作確認をしました。

- Windows 11 Pro 22H2
- node.js v18.17.1
- md-to-pdf 5.2.4

※ラズパイでも試しましたが、エラーが出てうまく動きませんでした。

2-2. インストール

次のコマンドで md-to-pdf パッケージをグローバルにインストールします。

```
npm i -g md-to-pdf
```

2-3. 使い方

ターミナル（コマンドライン）から次のコマンドを入力します。

```
md-to-pdf "ファイル名"
```

2-3-1. スタイル

ドキュメントの先頭に次のようなブロックを挿入することでスタイルを制御できます。詳しくは資料末の公式サイトを参考にしてください。

```
---
pdf_options:
  format: a4
  margin: 20mm 20mm
  printBackground: true
  headerTemplate: |-
    <style>
      section {
        margin: 5mm 15mm;
        font-family: system-ui;
        font-size: 10px;
        color: silver;
      }
    </style>
    <section>
      <span class="title"></span>
      <span class="date"></span>
    </section>
  footerTemplate: |-
    <section style="margin: 0 auto;">
      <div>
        Page <span class="pageNumber"></span>
        of <span class="totalPages"></span>
      </div>
    </section>
  stylesheet: https://cdnjs.cloudflare.com/ajax/libs/github-markdown-
  css/5.2.0/github-markdown.min.css
  body_class: markdown-body
  css: |-
    .page-break { page-break-after: always; }
    .markdown-body { font-size: 16px; }
    .markdown-body pre > code { white-space: pre-wrap; }
    .markdown-body pre { background: #eee; }
    .markdown-body pre { padding: 5px; }
---
```

2-3-2. 改ページ

改ページしたい個所で以下のタグを挿入します。

```
<div class="page-break"></div>
```

3. まとめ

康熙部首の問題があったため Markdown PDF の使用は避けてきましたが、Markdown PDF(z) ではこの問題が解消されているので、これを使うのが一番手軽でいいかもしれません。

md-to-pdf にも大変お世話になりましたが、ひとつ難点を上げるとすると文章の先頭にヘッダー、フッターやスタイルを記述する形式のため、文章が汚れる点があげられるでしょうか。

どちらも便利なツールなので好みに合わせて使い分ければよいと思います。

4. OSS

本資料では以下のリポジトリの github-markdown.css を改変して使用しています。

[github-markdown-css](#)

5. 参考リンク

[Markdown PDF\(z\) GitHub 公式サイト](#)

[md-to-pdf GitHub 公式サイト](#)