

1 拡張機能 : markdown preview enhanced

markdown preview enhanced だと名称が長すぎるので、以下ではmpeとします。

mpeの導入

VScodeを開いて拡張機能であるmpeを入れていきます。

VScodeを開いたら、左側の5つくらいアイコンがあるうちの一番したの四角形みたいなやつをクリックしましょう。

そうすると拡張機能の管理画面みたいなのが開くので、一番上の検索窓に `markdown preview enhanced` と打ち込みましょう。

おそらく一番上に一番DLされている `Yiyi Wang` さんのやつがあるので、それをクリックし、インストールしていきます。

mpeでできること

何ができるかについては[公式ドキュメント](#)で結構詳しく書かれているので、そちらを読むのが一番いいと思います。

一応、拡張としてできるようになることを簡単に説明します。

Preview画面を開く

`Ctrl + K` → `V` で現在開いているmdファイルのプレビュー画面を開くことができます。

各種ファイルへエクスポートできるようになる

Preview画面を右クリックすると、エクスポートの方法と、どのファイル形式へエクスポートするかを選択することができます。

Latexの数式が挿入できる

`$ ~ $` で文中に組み込み、
`$$ ~ $$` か、```math` で中央に数式を組み込みます。

$$f(x) = \alpha(x^2 - \beta) + \gamma$$

```
$$
```

```
\begin{array}{ccl}
```

```
SNR &= & {\displaystyle {\sqrt{P}hs(t) \above{0.9pt}{n(t)}} }\\
```

```

\\
&=& {\displaystyle {Ph^2s(t)^2\above{0.9pt}n(t)^2} }\\
\\
&=& {\displaystyle {P|h|^2\above{0.9pt}\sigma^2} }
\end{array}
$$

```

$$\begin{aligned}
 SNR &= \frac{\sqrt{P}hs(t)}{n(t)} \\
 &= \frac{Ph^2s(t)^2}{n(t)^2} \\
 &= \frac{P|h|^2}{\sigma^2}
 \end{aligned}$$

```

```math
\begin{array}{ccl}
\tilde{h}_{i1} &=& E\left[{y_i^{(1)}(t) \above{0.9pt} s_{p1}(t)} \right] \\
&=& h_{i1} + E\left[{n^{(1)}(t) \above{0.9pt} s_{p1}(t)} \right] \\
&=& h_{i1} + \sigma'
\end{array}
```

```

$$\begin{aligned}
 \tilde{h}_{i1} &= E \left[\frac{y_i^{(1)}(t)}{s_{p1}(t)} \right] \\
 &= h_{i1} + E \left[\frac{n^{(1)}(t)}{s_{p1}(t)} \right] \\
 &= h_{i1} + \sigma'
 \end{aligned}$$

ファイルの挿入

- 各種ソースファイル
- PDFファイル
- imageファイル

コードブロックの操作の強化

色々な面でコードブロックの記法の強化がありますが、以下の要素が特に強みだと思います。

- コードブロックの実行
- 実行結果の表示
- 行数の表示

2 プレビュースタイルをいじる

プレビューはcssでいじることができます。
もしCSSを知っていればある程度自分の好きなように表示をいじることが可能です。

mpeでの表示の設定をいじるには `Ctrl + shift + p` を押して、`Customize css` と入力して、`Enter` を押してください。

lessファイルが開くので、CSSを書き込んでいきます。
もし自分のCSSがほしければあげるので、その場合は言ってください。

3 PDF化する

markdown preview enhancedがPhantomJSのサポートをつい最近やめてしまったので、最近はPuppeteerを利用してPDF化しています。

PuppeteerはGoogle Chromeのエンジンさえあれば動作するので、もしブラウザでIEやedgeを使っているひとはGoogle Chromeを入れましょう。
(Linuxの場合はChromiumでもOK)

Chromeの実行ファイルへパスを通す。

`Ctrl + ,` で設定を開き、検索窓に `chrome path` と入力すると、mpeのchrome path の設定が出ると思います。
そこにChromeの実行ファイルへのパスを入力します。
おそらくWindowsならパスを通す必要はないと思うので、もし実行ファイルへのパスがわからなければ、何も書かないままスルーしても問題はないと思います。

4 スニペットのすすめ

VScodeではデフォルトでユーザー定義のスニペットが作成できるようになりました。
また、スニペットは各ファイル種毎に作成できるため、競合も起こりにくく、とても便利です。

VScodeのスニペットの使い方は[こちら](#)へどうぞ。

また, 私のmarkdownのスニペットを見たいor欲しい場合は言っていただければ渡します.