

The background of the image is a dense forest of tall evergreen trees, likely Douglas firs, with their characteristic long, thin needles. Sunlight filters through the canopy from above, creating bright highlights on the tops of the trees and casting deep shadows in the lower areas. The overall atmosphere is serene and natural.

Slidev入門

自己紹介



片田恭平



所属

東京理科大学大学院 電気工学専攻 M2

医療機器（人工心臓）の研究してます！

趣味

ゲーム, マンガ, アニメ, スポーツ観戦

最近はたまにポケモンSVしてます

特技

映像制作を昔してました
After Effectsが好きです



関心のあるプログラミング言語



突然ですが

スライドを作る時どんなツールを
使っていますか？

自分の周りだと...

GUIのプレゼンテーションツールを使う人が多いです

Power Point

一番有名 大学だとほぼ毎日触ります

Google Slide

基本的に無料 パワーポイントとの互換性があり

Keynote

Mac持ってたら無料

その他

たまにFigmaとか動画編集ツールでスライドを作る人も見たことがあります

GUIだとみんな使えていい（と普通の人は思うかもです）

エンジニアの場合

これから新卒入社でわからないことだらけなのでアウトプットもして力をつけたいが...
基本的に怠惰な人間なので

バージョン管理にGitを使いたい

commitを使ってversion管理したり、branchを切って別のデザインパターンにすぐ切り替えたい

エディタからなるべく離れたくない

アプリ間の移動がめんどくさい。

ソースコードをシンタックスハイライトしたい

スクショを貼り付けるのは面倒

Markdownで書きたい

ZennやQiitaなどの技術記事での発信の際になるべく短時間で済ませたい



Slidevとは？

開発者のためのプレゼンテーションスライドツール

まだベータ版ですが、OSS化されており、2万4千ほどのstarがついてます
VueやWindiCSSで作成されています。

Slidevの利点

-  Markdownベース - お気に入りのエディタとワークフローを使用
-  デベロッパーフレンドリー - ビルトインのシンタックスハイライト、ライブコーディングなど
-  インタラクティブ - Vueコンポーネントをシームレスに埋め込み
-  プrezenterモード - 別のウィンドウ、スマートフォンでさえもスライドを操作
-  描画 - スライドに描画し、注釈をつける
-  LaTeX - LaTeX数式のビルトインサポート
-  アイコン - どんなアイコンセットからでも、直接アイコンにアクセス
-  レコーディング - ビルトインのレコーディングとカメラビュー
-  ポータブル - PDF、PNG、またはホスト可能なSPAにエクスポート
-  自由に開発可能 - Viteプラグイン、Vue components、どんなnpmパッケージも使用可能

Slide公式HPより引用

百聞は一見にしかずということで

The background of the image is a dense forest of tall evergreen trees. Sunlight filters through the branches, creating bright highlights on the needles and casting deep shadows in the canopy. The overall atmosphere is serene and natural.

Slidevを使ってみよう

インストール方法

npmやyarnからプロジェクトを作成できます(Node.js>14.0が必要)

npmからプロジェクトを作成

```
$ npm init slidev@latest
```

npmからslidevを起動

```
$ npm run dev
```

yarnからプロジェクトを作成

```
$ yarn create slidev
```

yarnから

```
$ yarn run dev
```

ブラウザ上で一旦試したい方用

こちらのリンクからブラウザ上でお試しできます！

<https://stackblitz.com/edit/slides-tnb8s?file=slides.md>

ディレクトリ構成

```
./  
|-- .gitignore  
|-- .npmrc  
|-- README.md  
|-- components  
|   |-- Counter.vue  
|-- layouts  
|   |-- end.vue  
|-- netlify.toml  
|-- package-lock.json  
|-- package.json  
|-- slides.md  
|-- vercel.json
```

インストール直後のディレクトリ構成

基本的にはslides.mdを編集します

```
./  
|-- .gitignore  
|-- .npmrc  
|-- README.md  
|-- components  
|   |-- Counter.vue  
|-- layouts  
|   |-- end.vue (最終スライドを書き換えるためのコンポーネント)  
|-- public  
|   |-- img(画像とかはここに入れます)  
|-- setup  
|   |-- windicss.ts (windicSSを使えるようにする)  
|-- netlify.toml  
|-- package-lock.json  
|-- package.json  
|-- slides-export.pdf(PDFで出力したファイル)  
|-- slides.md  
|-- vercel.json
```

追加で変更を加えたファイル

WindiCSSの設定

SlidevはtailwindCSSと互換のあるwindiCSSが使えます rootディレクトリに新たにsetupディレクトリを作成し windicss.tsを記述します

```
1 import { defineWindiSetup } from '@slidev/types'
2
3 // extending the builtin windicss configurations
4 export default defineWindiSetup(() => ({
5   shortcuts: {
6     // custom the default background
7     'bg-main': 'bg-white text-[#181818] dark:bg-[#121212] dark:text-[#ddd]' ,
8   },
9   theme: {
10     extend: {
11       // fonts can be replaced here, remember to update the web font links in `index.html`
12       fontFamily: {
13         sans: 'ui-sans-serif,system-ui,-apple-system,BlinkMacSystemFont,"Segoe UI",Roboto,"Helvetica Neue",Arial,"N
14         mono: '"Fira Code", monospace',
15       },
16     },
17   },
18 }))
```

slides.mdの記述方法

ページの追加

slides.mdにmarkdownを記述することでスライドを作成します

--- を書くとページが区切られます。

フロントマター

フロントマターと呼ばれる記述をするとpage追加ではなく、次のスライドのレイアウトを定義することができます

```
---
You, 4 秒前 | 1 author (You)
layout: intro
class: 'text-center'
---

You, 4 秒前 | 1 author (You)
# スライドを作る時どんなツールを使っていますか?

---

You, 4 秒前 | 1 author (You)
# 自分の周りだと...

GUIのプレゼンテーションツールを使う人が多いです
You, 4 秒前 | 1 author (You)
### Power Point
一番有名 大学だとほぼ毎日触ります
You, 4 秒前 | 1 author (You)
### Google Slide
基本的に無料 パワーポイントとの互換性があり
You, 4 秒前 | 1 author (You)
### Keynote
Mac持ってたら無料

You, 10 秒前 | 1 author (You)
### その他
たまにFigmaとか動画編集ツールでスライドを作る人も見たことがあります

You, 10 秒前 | 1 author (You)
## GUIだとみんな使えていい(と普通の人は思うかもです)

---
```

Markdown出力例

H1

H2

H3

H4

H5

これは本文です。

これは本文です(太字)

~~これも本文です(取り消し)~~

じゃあ私も本文です(イタリック)

```
1 # H1
2
3 ## H2
4
5 ### H3
6
7 ##### H4
8
9 ##### H5
10
11 これは本文です。<br>
12
13 **これは本文です(太字)**
14
15 ~~これも本文です(取り消し)~~
16
17 *じゃあ私も本文です(イタリック)*
```

リスト

箇条書きリスト

- hoge
 - foo
 - foo

番号付きリスト

1. hoge
 1. foo
 2. foo

チェックリスト

- hoge1
- hoge2

```
1  箇条書きリスト
```

```
2  
3  - hoge  
4      - foo  
5      - foo  
6
```

```
7  番号付きリスト
```

```
8  
9  1. hoge  
10     1. foo  
11     1. foo  
12
```

```
13  チェックリスト
```

```
14  
15     - [ ] hoge1  
16     - [x] hoge2
```

テーブル

テーブルが表示 VScodeのTable Formatterとかを使うと便利そうです

製品名	メーカー名	価格
A	D社	100円
B	E社	200円
C	F社	300円

```
1 | 製品名 | メーカー名 | 価格 |
2 | ----- | :-----: | ---: |
3 | A      | D社      | 100円 |
4 | B      | E社      | 200円 |
5 | C      | F社      | 300円 |
```

画像

web上の画像も表示できます unsplashのサイトを使うとリロードするたびにランダムで画像が変わります。



画像などのassetsは`public`ディレクトリに格納するとSPAのドキュメントルートにコピーされます。

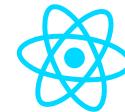


アイコン

Icons | Sliderv に利用可能なアイコンの情報があります。

技術スタック系のアイコンはIconifyの公式サイトから探すのが楽かもしれません

Tailwind CSS のクラスを設定するとAnimationなど面白い効果が狙えるかもしれません。



```
1 <logos-rails class="text-5xl m-12"/>
2 <logos-nextjs-icon class="text-5xl m-12"/>
```

```
1 <logos-graphql class="text-5xl m-12 animate-bounce"/>
2 <logos-react class="text-5xl m-12 animate-spin"/>
```

ソースコードのシンタックスハイライト

行番号が表示されているのは`lineNumbers: true`のおかげです。

ハイライトを段階的に表示する場合は`{1-2|5|all}`のように書くと良いです

before

```
1 import os
2 test_path = os.path.join("data", "data-01.txt")
3
4 f = open(test_path, "a", encoding="utf-8")
5 f.write("this is new append line\n")
6 f.close()
```

after

```
1 import os
2 test_path = os.path.join("data", "data-01.txt")
3
4 with open(test_path, "a", encoding="utf-8") as f:
5     f.write("this is new append line\n")
```

Monaco Editor

スライド上からコードを書き換えることができます、ライブコーディング的な感じで見せると面白いかもです。

```
1 console.log('HelloWorld')
```

LaTeX

LaTexにも対応しているので数式をかっこよく書きたい場合に使えそうです。

例としてマクスウェル方程式を書いてます（意味はありません）

(微分形による) マクスウェルの方程式は、以下の4つの連立偏微分方程式である。

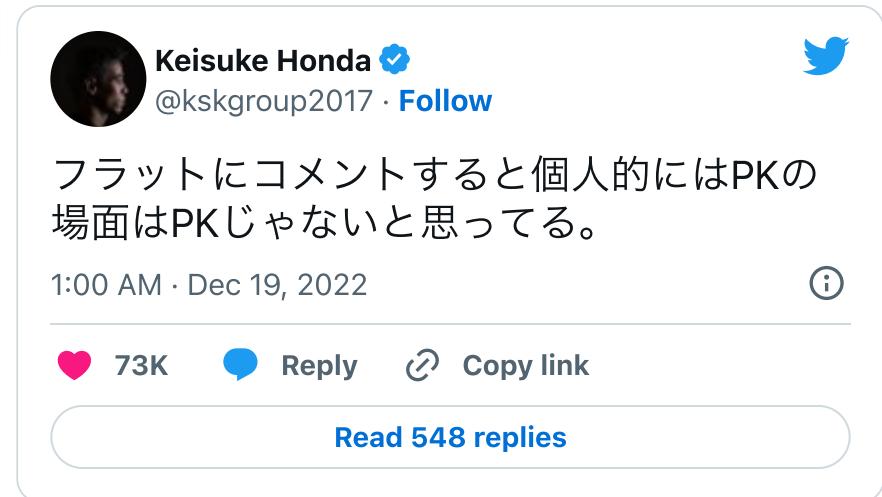
$$\left\{ \begin{array}{l} \nabla \cdot \mathbf{B}(t, \mathbf{x}) = 0 \\ \nabla \times \mathbf{E}(t, \mathbf{x}) + \frac{\partial \mathbf{B}(t, \mathbf{x})}{\partial t} = 0 \\ \nabla \cdot \mathbf{D}(t, \mathbf{x}) = \rho(t, \mathbf{x}) \\ \nabla \times \mathbf{H}(t, \mathbf{x}) - \frac{\partial \mathbf{D}(t, \mathbf{x})}{\partial t} = \mathbf{j}(t, \mathbf{x}) \end{array} \right.$$

ここで \mathbf{E} は電場の強度、 \mathbf{B} は磁束密度、 \mathbf{D} は電束密度、 \mathbf{H} は磁場の強度を表す。また ρ は電荷密度、 \mathbf{j} は電流密度を表す。記号「 $\nabla \cdot$ 」、「 $\nabla \times$ 」はそれぞれベクトル場の発散 (div) と回転 (rot) である。

Twitter

デフォルトでTwitterのコンポーネントが使えるので簡単にTweetを埋め込みます。
画像ではないので文字コピーやリンククリックもできます。

```
<Tweet id="1423237009561186308"/>
```



Keisuke Honda 
@kskgroup2017 · [Follow](#) 

フラットにコメントすると個人的にはPKの
場面はPKじゃないと思ってる。

1:00 AM · Dec 19, 2022 

 73K  Reply  Copy link

[Read 548 replies](#)

YouTube

実態はSPAなのでYouTube動画の埋め込みコード(HTML)を貼り付けて再生することも可能です。
スライド上で動画を再生可能！

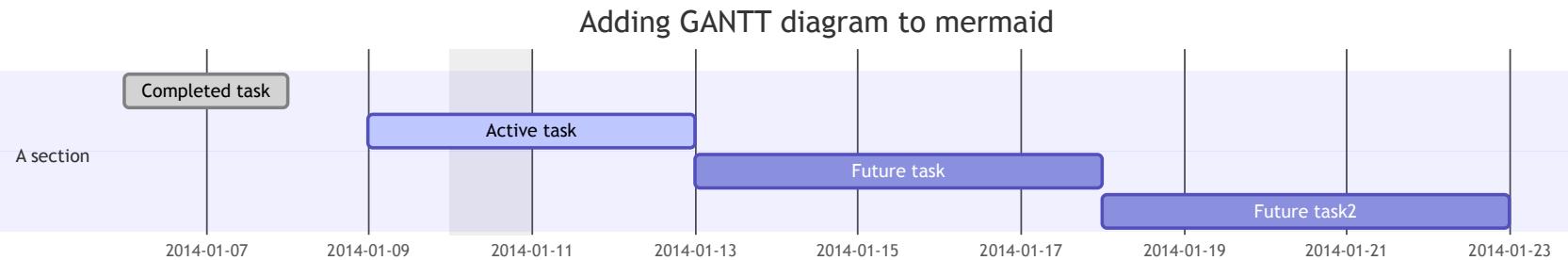


```
1 <iframe width="560" height="315" src="https://www.youtube.com/embed/dFlDRhvM4L0" title="YouTube video player" fram
```

Mermaid記法

Mermaid記法 でさまざまなものを描画できます

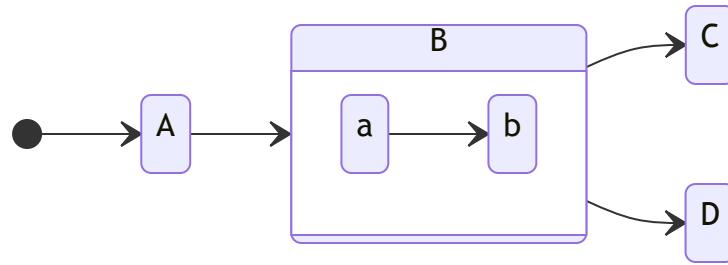
ガントチャート



Mermaid記法

Mermaid記法でさまざまなものを描画できます

状態遷移図



Notionで記述できそうなものは大体描画できそう

<https://dev.classmethod.jp/articles/mermaid-markdown-is-supported-in-notion/#toc-7>

コンポーネントの利用

Vueのコンポーネントを作成し使うこともできます テンプレートのプロジェクトにはCounterのcomponentが入っています

変数はグローバル変数（ページ番号）は定義できそうですが、スライド個別の変数は使えなさそうです

```
1 <Counter :count="10" />
```

- 10 +

Check out the guides for more.

```
1 <Tweet id="1390115482657726468" />
```



Read the full conversation on Twitter

203 Reply Copy link

endを上書きする

デフォルトの状態だと最後のスライドが固定になるため、layoutsの中にend.vueを作成し上書きする必要があります

```
1 <template>
2   <div class="slidev-layout default">
3     <div class="text-center">
4       
5       <p class="text-4xl">ありがとうございました！</p>
6     </div>
7   </div>
8 </template>
```

共有方法

デプロイする

localhostのまま使ってもいいですが、 資料を共有する必要があるかもしれません

Vercel, Netlifyの設定用jsonファイルがデフォルトで入っているので、 そちらがおすすめです

Vercel,NetlifyはGithubと連携してしまえば、 pushした段階で自動デプロイしてくれます

Githubのリポジトリ <https://github.com/katakyo/hello-slidev>

今回のスライド <https://hello-slidev.vercel.app/>

PDFに書き出す

以下のコマンドでPDFへの書き出しが可能です

```
1 $ npm run export
```

その他の機能

時間があればデモします

- webカメラ機能（プレゼンターモードから、webカメラを起動してスライドに重ねて表示できます）
- 図形の書き込み
- スライドの録画
- ダークモードへの切り替え

最後に

凝ったアニメーションやcomponentを作成するのは大変ですが、使い回しができそうということもあるので今後も積極的に使ってみようと思っています

興味を持った方は是非使って見てください！