

Slidev 入门

Markdown to slide

academic theme

Delta, 2022/12/01

Table of Contents

- 1. 安装
- 2. 支持Markdown语法
- 3. 使用 vscode 编写slide
- 4. 首页
- 5. 引用图片
- 6. layout: figure
- 7. layout: figure-side
- 8. layout: image
- 9. layout: center
- 10. layout: two-cols
- 11. image in a row
- 12. Footnote
- 13. 实在不行
- 14. 导出PDF

安装

<https://cn.sli.dev/guide/install.html>

```
npm init slidev@latest
```

```
cd slidev
```

AUR的 `slidev-cli` 用起来有问题，不建议用。

运行

<https://cn.sli.dev/guide/install.html#build-hostable-spa-single-page-application>

```
npx slidev # npx slidev slides.md
npx slidev build # 从slides.md 生成网页
npx slidev export # 从slides.md 导出 PDF
# vscode 使用 simple browser 打开 http://localhost:3030
```

支持Markdown语法

1. 有序列表1
2. 有序列表2

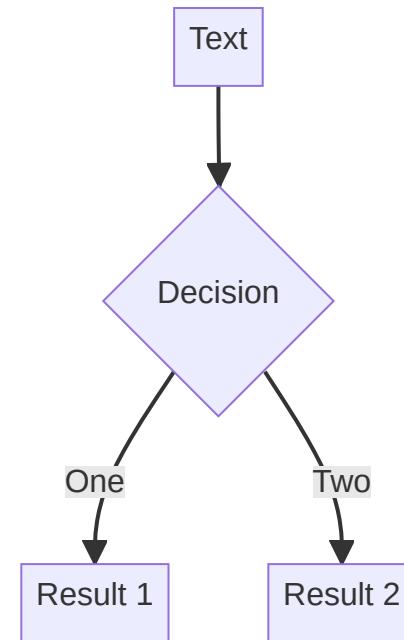
- list1
- list2

```
`inline-code-block`
```

支持Latex公式: $E = M \times C^2$

$$\begin{aligned}
 3^{6n+3} + 4^{6n+3} &\equiv (3^3)^{2n+1} + (4^3)^{2n+1} \\
 &\equiv 27^{2n+1} + 64^{2n+1} \\
 &\equiv 27^{2n+1} + (-27)^{2n+1} \\
 &\equiv 27^{2n+1} - 27^{2n+1} \\
 &\equiv 0 \pmod{91}
 \end{aligned}$$

支持Mermaid语法



使用 vscode 编写 slide

- vscode插件`slidev`，目前唯一作用是在 markdown里显示分页，preview功能完全没用。[[2022年12月01日]]
- vscode插件`Markdown Paste`从剪切板导入附件，方便从截图直接导入。快捷键`ctrl + shift + v`。建议如图进行设置。

```
 ${fileDirname}/public/${fileBasenameNoExtension}
```

这是因为Slidev规定所有的静态资源都要放在`public/`目录下

但是引用资源时，把`public/`视为根目录，类似`!
[](/slidev-intro/example.png)`

所有静态资源引用都必须`/`开头，否则导出时会报错。



`vscode`里使用`simple browser`侧栏打开
<http://localhost:3030/>

为了解决相对路径的问题，将`slides.md`放到`public/`目录下，是个不错的方法。[2022年12月01日]BUG：slidev会卡死...因此不要把`slides.md`放到`public/`目录下。

扉页

```
---
```

```
theme: academic
class: text-white
coverAuthor: Delta
coverAuthorUrl: https://github.com/Delta-in-hub
coverBackgroundUrl: https://tvax2.sinaimg.cn/large/005uNPw
coverBackgroundSource: unsplash
coverBackgroundSourceUrl: https://unsplash.com/photos/Rd9u
lineNumbers: true
download: true
themeConfig:
  paginationX: r
  paginationY: b
  paginationPagesDisabled: [1]
---
```

下载并使用主题`academic`。

并配置第一页的yaml即可。



引用图片

通过``这种方式没法控制图片的大小

因此要使用``



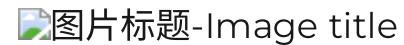
```

```



layout: figure

但是



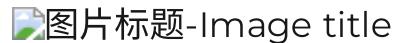
图片标题-Image title

layout: figure

```
---  
layout: figure  
figureCaption: 图片标题-Image title  
figureUrl: /example.png  
---
```

把图片挤跑了

怎么办?



图片标题-Image title

layout: figure

```
<p align="center">  
    
  title  
</p>
```



title

<https://cn.sli.dev/builtin/cosine-nearest-neighbor.html>

layout: figure-side

```
----  
layout: figure-side  
figureCaption: 图片标题-Image title  
figureUrl: /example.png  
figureX: '1' # 'r'  
----
```



图片标题-Image title

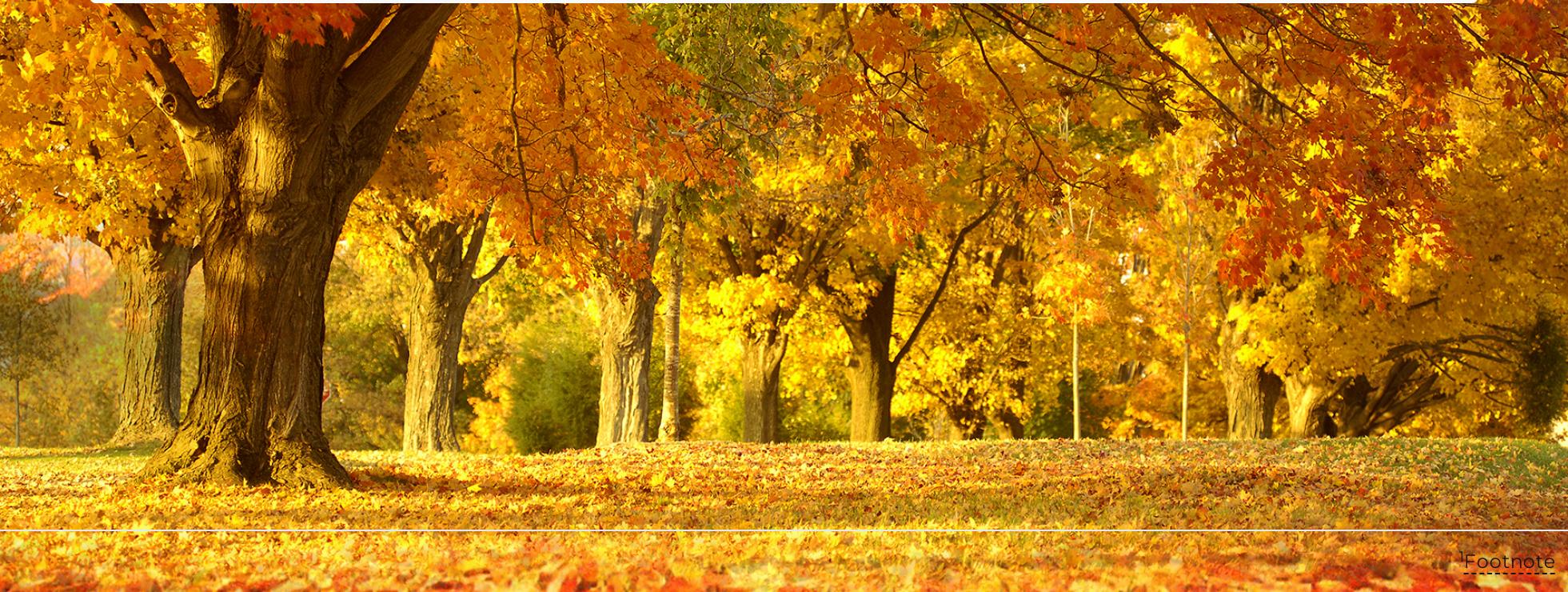
layout: image

```
---
```

```
layout: image
```

```
image: /example.png
```

```
--
```



layout: center

This is the simple.

```
----  
layout: center 垂直居中  
class: "text-center" 水平居中  
----
```

layout: two-cols

```
---
```

layout: two-cols

```
--
```



column 2



image in a row



```
  

```

控制水平用`align`，控制垂直方向用`
`

Footnote

Text with footnote¹

Text with footnote¹

```
<br><br><br>
<Footnotes v-after separator>
  <Footnote :number=1>
    <a href="">Footnote</a>
  </Footnote>
</Footnotes>
```

实在不行

控制精细的版式，要改css，`slidev` 基于vue当然有这个能力，我认为太麻烦了。

要在Markdown的简洁和版式控制之间取得一个平衡，使用`yaml`加上适当的html，比如``我可以接受这个程度。同时，配合vscode的snippets算是比较简单了。

当然实在不行，就在PPT里做然后截图，用`layout:image`

```
'node_modules/@slidev/client/layouts/two-cols.vue'
```

```
Line32 改为 `<div class="col-left"
:class="props.class" style="padding-right: 20px;">`
```

layout

- two-cols
- figure-side
- center
 - class: text-center
- image
- figure

<https://cn.sli.dev/builtin/layouts.html>

```
Line35 改为 `<div class="col-right"
:class="props.class" style="padding-left: 20px;">`
```

不然两列文字贴在一起，看着难受。

导出PDF

`academic`这个主题导出PDF有bug，导出pdf时,页码会重叠在一起,因此要取消页码。以及脚注也会重叠在一起。

```
---
themeConfig:
  paginationX:
  paginationY:
  paginationPagesDisabled: []
---
```

以及删除所有`<Footnotes>...`

部署到Github Page

```
npx slidev build slide.md --base="/slidev-intro/" # 放到repo的哪个文件夹,若是本地部署,--base参数可以不填。
npx slidev export slide-pdf.md

# 生成的pdf放到 dist 文件夹并重命名为 slidev-exported.pdf
```