

Markdown语法整理

- 目录
 - 一.分级目录
 - 二.斜体与粗体
 - 三.超链接
 - 3.1.行内式
 - 3.2.参考式
 - 3.3.自动连接
 - 四.锚点
 - 4.1.页内超链接
 - 4.2.文章内部标题链接
 - 五.列表
 - 5.1.无需列表
 - 5.2.有序列表
 - 5.3.定义型列表
 - 5.4.列表缩进
 - 5.5.包含段落的列表
 - 5.6.任务列表
 - 六.表格
 - 6.1.竖杠|下划线_写法
 - 6.2.html写法
 - 七.创建和突出显示代码块
 - 7.1.栅栏代码块
 - 7.2.语法高亮显示
 - 7.3.引用代码
 - 7.4.引用文字
 - 7.5.删除线
 - 使用LaTeX方法

说明：本文中中语法里面的代码就是需要写的Markdown语法，展示效果就是最后生成的页面
此语法编写和测试环境：windows上使用MarkdownPad2的GitHub离线风格的Markdown语法，
使用在线风格好像不支持文章内部链接跳转

一.分级目录

语法：

- * 总目录
- * 一级目录
- * 二级目录
- * 三级目录

说明：层级列表一般只支持三级列表，第一级实心圆点，第二级空心圆点，第三级以后都是实心方点。要实现层级列表，每下一级都要比上一级多输入2个空格或者一个Tab，而且第一级前面不要超过3个空格.

展示效果：

- 总目录
 - 一级目录
 - 二级目录
 - 三级目录

二.斜体与粗体

语法：

斜体

****粗体****

展示效果：

斜体

粗体

三.超链接

Markdown支持两种形式的链接语法：行内式和参考式，行内式一般用的比较多。

3.1.行内式：

语法：

[打开百度](http://www.baidu.com)

[打开百度](http://www.baidu.com "打开百度")

说明:[]里写链接文字, ()里写链接地址, ()中的""中可以为链接指定title属性, title属性可加可不加。title属性的效果是鼠标悬停在链接上会出现指定的 title文字。[链接文字](链接地址 "链接标题")这样的形式。链接地址与链接标题前有一个空格。

展示效果：

[打开百度](http://www.baidu.com)

[打开百度](http://www.baidu.com)

3.2.参考式：

参考式超链接一般用在学术论文上面，或者另一种情况，如果某一个链接在文章中多处使用，那么使用引用的方式创建链接将非常好，它可以让你对链接进行统一的管理。

语法：

我经常浏览的几个网站[Google][1]、[Baidu][2]、[51CTO][3]和看视频的网站[爱奇艺][4]感觉都是很不错的[网站][5]。

[1]:<http://www.google.com> "google"

[2]:<http://www.baidu.com> "Baidu"

[3]:<http://www.51cto.com> "51cto"

[4]:<http://www.iqiyi.com> "aiqiyi"

[网站]:<http://www.qq.com>

展示效果：

我经常浏览的几个网站[Google](#)、[Baidu](#)、[51CTO](#)和看视频的网站[爱奇艺](#)感觉都是很不错的[网站](#)。

3.3.自动连接：

语法：

```
<http://www.baidu.com/>
<admin@baidu.com>
```

说明：Markdown 支持以比较简短的自动链接形式来处理网址和电子邮件信箱，只要是用<>包起来，Markdown 就会自动把它转成链接。一般网址的链接文字就和链接地址一样。

展示效果：

<http://www.baidu.com/>
admin@baidu.com

四.锚点

网页中，锚点其实就是页内超链接，也就是链接本文档内部的某些元素，实现当前页面中的跳转。比如我这里写下一个锚点，点击回到目录，就能跳转到目录。在目录中点击这一节，就能跳过来。还有下一节的注脚。这些根本上都是用锚点来实现的。

注意：Markdown Extra 只支持在标题后插入锚点，其它地方无效

4.1.页内超链接：

语法：

```
# 1.分级目录{#index}
跳转到[标题]{#index}
```

说明：在你准备跳转到的指定标题后插入锚点{#标记}，然后在文档的其它地方写上连接到锚点的链接。此处使用的是GitHub风格的Markdown语法所以无法正常显示效果，推荐使用以下[4.2.文章内部标题链接](#)。

展示效果：

1.分级目录{#index}

跳转到1.分级目录

说明：这里是在本文的分级目录设置的，不然效果不明显

4.2.文章内部标题链接：

语法：

```
* [目录1](#40)
  * [标题1](#41)
  * [标题2](#42)
  * [标题3](#43)
  * [标题4](#44)
```

```
<h3 id="41">标题1</h3>
  轻轻的我走了， 正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。

<h3 id="42">标题2</h3>
  正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。

<h3 id="43">标题3</h3>
  我轻轻的招手， 作别西天的云彩。

<h3 id="44">标题4</h3>
  作别西天的云彩。
```

展示效果：

- 目录1
 - 标题1
 - 标题2
 - 标题3
 - 标题4

标题1

轻轻的我走了， 正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。

标题2

正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。

标题3

我轻轻的招手， 作别西天的云彩。

标题4

作别西天的云彩。

五.列表

5.1.无需列表:

使用 `*`, `+`, `-` 表示无序列表

语法:

- 无序列表一
- + 无序列表二
- * 无序列表三

展现效果:

- 无序列表一
- 无序列表二
- 无序列表三

5.2.有序列表:

有序列表则使用数字接着一个英文句点

语法:

1. 有序列表一
2. 有序列表二
3. 有序列表三

展现效果:

1. 有序列表一
2. 有序列表二
3. 有序列表三

5.3.定义型列表:

语法:

名词1
: 定义 1（左侧有一个可见的冒号和四个不可见的空格）

代码块 2

: 这是代码块的定义（左侧有一个可见的冒号和四个不可见的空格）

代码块（左侧有八个不可见的空格）

说明：

定义型列表由名词和解释组成。一行写上定义，紧跟一行写上解释。解释的写法:紧跟一个缩进(Tab)

展现效果：

名词1

: 定义 1（左侧有一个可见的冒号和四个不可见的空格）

代码块 2

: 这是代码块的定义（左侧有一个可见的冒号和四个不可见的空格）

代码块（左侧有八个不可见的空格）

5.4.列表缩进：

语法：

```
*   轻轻的我走了， 正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。  
    那河畔的金柳， 是夕阳中的新娘； 波光里的艳影， 在我的心头荡漾。  
    软泥上的青荇， 油油的在水底招摇； 在康河的柔波里， 我甘心做一条水草！  
*   那榆荫下的一潭， 不是清泉， 是天上虹； 揉碎在浮藻间， 沉淀着彩虹似的梦。  
    但我不能放歌， 悄悄是别离的笙箫； 夏虫也为我沉默， 沉默是今晚的康桥！  
    悄悄的我走了， 正如我悄悄的来； 我挥一挥衣袖， 不带走一片云彩。
```

说明：列表项目标记通常是放在最左边，但是其实也可以缩进，最多 3 个空格，项目标记后面则一定要接着至少一个空格或制表符。

效果：

- 轻轻的我走了， 正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。
 那河畔的金柳， 是夕阳中的新娘； 波光里的艳影， 在我的心头荡漾。
 软泥上的青荇， 油油的在水底招摇； 在康河的柔波里， 我甘心做一条水草！
- 那榆荫下的一潭， 不是清泉， 是天上虹； 揉碎在浮藻间， 沉淀着彩虹似的梦。
 但我不能放歌， 悄悄是别离的笙箫； 夏虫也为我沉默， 沉默是今晚的康桥！
 悄悄的我走了， 正如我悄悄的来； 我挥一挥衣袖， 不带走一片云彩。

5.5.包含段落的列表：

语法：

```
*   轻轻的我走了， 正如我轻轻的来； 我轻轻的招手， 作别西天的云彩。  
    那河畔的金柳， 是夕阳中的新娘； 波光里的艳影， 在我的心头荡漾。  
    软泥上的青荇， 油油的在水底招摇； 在康河的柔波里， 我甘心做一条水草！
```

那榆荫下的一潭，不是清泉，是天上虹；揉碎在浮藻间，沉淀着彩虹似的梦。
寻梦？撑一支长篙，向青草更青处漫溯；满载一船星辉，在星辉斑斓里放歌。
但我不能放歌，悄悄是别离的笙箫；夏虫也为我沉默，沉默是今晚的康桥！
* 悄悄的我走了，正如我悄悄的来；我挥一挥衣袖，不带走一片云彩。

说明：

列表项目可以包含多个段落，每个项目下的段落都必须缩进 4 个空格或是 1 个制表符（显示效果与代码一致）：如果你每行都有缩进，看起来会好看很多，当然，再次地，如果你很懒惰，Markdown 也允许

效果：

- 轻轻的我走了，正如我轻轻的来；我轻轻的招手，作别西天的云彩。
那河畔的金柳，是夕阳中的新娘；波光里的艳影，在我的心头荡漾。
软泥上的青荇，油油的在水底招摇；在康河的柔波里，我甘心做一条水草！

那榆荫下的一潭，不是清泉，是天上虹；揉碎在浮藻间，沉淀着彩虹似的梦。
寻梦？撑一支长篙，向青草更青处漫溯；满载一船星辉，在星辉斑斓里放歌。
但我不能放歌，悄悄是别离的笙箫；夏虫也为我沉默，沉默是今晚的康桥！
- 悄悄的我走了，正如我悄悄的来；我挥一挥衣袖，不带走一片云彩。

5.6.任务列表:

要创建任务列表，前缀列表项 `[]`。要将任务标记为完整，请使用 `[x]`

语法：

- [] 跑步
- [] 骑车
- [x] 吃饭
- [] 睡觉

展现效果：

- ☐ 跑步
- ☐ 骑车
- ☒ 吃饭
- ☐ 睡觉

六.表格

6.1.竖杠|下划线_写法:

使用竖杠|下划线_写法

语法：

```
| 名字 | 性别 | 年龄 | 国籍 |  
| --- | ---- | ---- | --- |  
| 张三 | 男 | 23 | 中国 |  
| 小红 | 女 | 18 | 中国 |  
| Tom | 男 | 46 | 美国 |
```

展现效果：

名字	性别	年龄	国籍
张三	男	23	中国
小红	女	18	中国
Tom	男	46	美国

6.2.html写法:

使用

--

语法：

```
<table>  
  <tr>  
    <td>车次</td>  
    <td>开车时间</td>  
    <td>到达时间</td>  
  </tr>  
  <tr>  
    <td>D110</td>  
    <td>10:22</td>  
    <td>11:00</td>  
  </tr>  
</table>
```

展现效果：

车次	开车时间	到达时间
D110	10:22	11:00

这里是因为使用github离线风格导致空格，使用在线就没有了

七.创建和突出显示代码块

7.1.栅栏代码块:

语法:

```
```  

function test() {
 console.log ("在此函数之前注意空自行?");
}

```
```

展现效果:

```
function test() {  
  console.log("在此函数之前注意空自行?");  
}
```

7.2.语法高亮显示:

高亮度需要使用github在线风格才能显示颜色

语法:

```
```ruby  
require 'redcarpet'
markdown = Redcarpet.new("Hello World!")
puts markdown.to_html
```
```

展现效果:

ruby语法:

```
require 'redcarpet'  
markdown = Redcarpet.new("Hello World!")  
puts markdown.to_html
```

python语法:

```
def foo():  
    if not bar:  
        return True
```

本文使用的markdownPad2需要使用github在线markdown风格才会显示，其他不显示。

7.3.引用代码:

您可以使用单个反引号来调用句子中的代码或命令。反引号内的文本将不被格式化。

语法:

```
Use `git status` to list all new or modified files that haven't yet been committed.
```

展现效果:

Use `git status` to list all new or modified files that haven't yet been committed.

7.4.引用文字:

你可以用引用文本`>`.

语法:

```
In the words of Abraham Lincoln:
```

```
> Pardon my French
```

展现效果:

In the words of Abraham Lincoln:

`> Pardon my French`

7.5.删除线:

语法:

```
~~This was mistaken text~~
```

展现效果:

~~This was mistaken text~~

使用LaTeX方法

使用LaTeX方法:

下面还有些没有显示出效果，先扔着吧，毕竟才开始玩，先这样吧

MathJax是一款相当强悍的在网页显示数学公式的插件。本教程介绍MathJax如何使用LaTeX语法编写数学公式。

如何插入公式

LaTeX的数学公式有两种：行中公式和独立公式。行中公式放在文中与其它文字混编，独立公式单独成行。

行中公式可以用如下两种方法表示：

`\(数学公式\)` 或 `¥数学公式¥`（要把人民币符号换成美元符号）

独立公式可以用如下两种方法表示：

`\[数学公式\]` 或 `¥¥数学公式¥¥`（要把人民币符号换成美元符号）

例子：

```
$$ \[J\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m + \alpha + 1)} \left\{ \frac{x}{2} \right\}
```

显示：

```
$$ J\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m + \alpha + 1)} \left\{ \frac{x}{2} \right\}^{2m + \alpha}
```

如何输入上下标

`^`表示上标，`_`表示下标。如果上下标的内容多于一个字符，要用`{}`把这些内容括起来当成一个整体。上下标是可以嵌套的，也可以同时使用。

例子：

```
$$ x^{y^z} = (1 + \{ \rm e \}^x)^{-2xy^w} $$
```

显示：

```
$$ x^{y^z} = (1 + \{ \rm e \}^x)^{-2xy^w} $$
```

另外，如果要在左右两边都有上下标，可以用`\sideset`命令。

例子：

```
\sideset{^12}{^34}\bigotimes
```

显示：

```
$$ \sideset{^12}{^34}\bigotimes $$
```

如何输入括号和分隔符

`()`、`[]`和`|`表示自己，`{}`表示`{}`。当要显示大号的括号或分隔符时，要用`\left`和`\right`命令。

例子：

```
$$ f(x,y,z) = 3y^2z \left( 3+\frac{7x+5}{1+y^2} \right) $$
```

显示：

```
$$ f(x,y,z) = 3y^2z \left( 3+\frac{7x+5}{1+y^2} \right) $$
```

有时候要用`\left.`或`\right.`进行匹配而不显示本身。

例子：

```
$$ \left. \frac{\{\rm d\}u}{\{\rm d\}x} \right|_{x=0} $$
```

显示：

```
$$ \left. \frac{\{\rm d\}u}{\{\rm d\}x} \right|_{x=0} $$
```

如何输入分数

例子：

```
$$ \frac{1}{3} \quad \text{或} \quad 1 \over 3 $$
```

显示：

```
$$ \frac{1}{3} \quad \text{或} \quad 1 \over 3 $$
```

如何输入开方

语法：

```
$$ \sqrt{2} \quad \text{和} \quad \sqrt[n]{3} $$
```

显示：

```
$$ \sqrt{2} \quad \text{和} \quad \sqrt[n]{3} $$
```

如何输入省略号

数学公式中常见的省略号有两种，`\ldots`表示与文本底线对齐的省略号，`\cdots`表示与文本中线对齐的省略号。

语法：

```
f(x_1,x_2,\ldots,x_n) = x_1^2 + x_2^2 + \cdots + x_n^2
```

显示：

```
$$ f(x_1,x_2,\ldots,x_n) = x_1^2 + x_2^2 + \cdots + x_n^2 $$
```

如何输入矢量

语法：

```
\vec{a} \cdot \vec{b}=0
```

显示:

```
$$ \vec{a} \cdot \vec{b}=0 $$
```

如何输入积分

语法:

```
\int_0^1 x^2 {\rm d}x
```

显示:

```
$$ \int_0^1 x^2 {\rm d}x $$
```

如何输入极限运算

语法:

```
\lim_{n \rightarrow +\infty} \frac{1}{n(n+1)}
```

显示:

```
$$ \lim_{n \rightarrow +\infty} \frac{1}{n(n+1)} $$
```

如何输入累加、累乘运算

语法:

```
\sum_{i=0}^n \frac{1}{i^2} 和 \prod_{i=0}^n \frac{1}{i^2}
```

显示:

```
\sum_{i=0}^n \frac{1}{i^2} 和 \prod_{i=0}^n \frac{1}{i^2}
```

如何进行公式应用

先要在 [mathjax] 后添加:

```
<script type="text/javascript" src="http://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AM">
</script>
```

```
<script type="text/x-mathjax-config"> MathJax.Hub.Config({ TeX: {equationNumbers: { autoNumber: ["",
```

只要按照这个添加代码就能实现本页的效果, 包的下载自己找吧, 就这样,想在线就得这样

语法:

```
\begin{equation}\label{equation1}r = r_F + \beta(r_M - r_F) + \epsilon \end{equation}
```

显示:

```
\begin{equation}\label{equation1}r = r_F + \beta(r_M - r_F) + \epsilon \end{equation}
```

引用：请见公式(`\ref{equation1}`)

如何输入希腊字母

哎，这里的空白这蛋疼，目前还没有最终得解决发子

| 语法 | 显示效果 |
|------------------------------------|---------------|
| <code>\$\$ \alpha \$\$</code> | α |
| <code>\$\$ \beta \$\$</code> | β |
| <code>\$\$ \gamma \$\$</code> | γ |
| <code>\$\$ \Gamma \$\$</code> | Γ |
| <code>\$\$ \delta \$\$</code> | δ |
| <code>\$\$ \Delta \$\$</code> | Δ |
| <code>\$\$ \epsilon \$\$</code> | ϵ |
| <code>\$\$ \varepsilon \$\$</code> | ε |
| <code>\$\$ \zeta \$\$</code> | ζ |
| <code>\$\$ \eta \$\$</code> | η |
| <code>\$\$ \theta \$\$</code> | θ |
| <code>\$\$ \Theta \$\$</code> | Θ |
| <code>\$\$ \vartheta \$\$</code> | ϑ |
| <code>\$\$ \iota \$\$</code> | ι |
| <code>\$\$ \kappa \$\$</code> | κ |
| <code>\$\$ \lambda \$\$</code> | λ |
| <code>\$\$ \Lambda \$\$</code> | Λ |
| <code>\$\$ \mu \$\$</code> | μ |
| <code>\$\$ \nu \$\$</code> | ν |
| <code>\$\$ \xi \$\$</code> | ξ |
| <code>\$\$ \pi \$\$</code> | π |
| <code>\$\$ \Pi \$\$</code> | Π |
| <code>\$\$ \varpi \$\$</code> | ϖ |
| <code>\$\$ \rho \$\$</code> | ρ |
| <code>\$\$ \varrho \$\$</code> | ϱ |
| <code>\$\$ \sigma \$\$</code> | σ |
| <code>\$\$ \Sigma \$\$</code> | Σ |
| <code>\$\$ \varsigma \$\$</code> | ς |

| | |
|---------------------------------|------------|
| <code>\$\$ \tau \$\$</code> | τ |
| <code>\$\$ \upsilon \$\$</code> | υ |
| <code>\$\$ \Upsilon \$\$</code> | Υ |
| <code>\$\$ \xi \$\$</code> | ξ |
| <code>\$\$ \Phi \$\$</code> | Φ |
| <code>\$\$ \varphi \$\$</code> | φ |
| <code>\$\$ \chi \$\$</code> | χ |
| <code>\$\$ \psi \$\$</code> | ψ |
| <code>\$\$ \Psi \$\$</code> | Ψ |
| <code>\$\$ \omega \$\$</code> | ω |
| <code>\$\$ \Omega \$\$</code> | Ω |

如何输入其它特殊字符

关系运算符：

| 语法 | 显示效果 |
|----------------------------|-------|
| <code>\$\$ \pm \$\$</code> | \pm |

| | |
|-----------------------------------|--------------|
| <code>\$\$ \times \$\$</code> | \times |
| <code>\$\$ \div \$\$</code> | \div |
| <code>\$\$ \mid \$\$</code> | \mid |
| <code>\$\$ \nmid \$\$</code> | \nmid |
| <code>\$\$ \cdot \$\$</code> | \cdot |
| <code>\$\$ \circ \$\$</code> | \circ |
| <code>\$\$ \ast \$\$</code> | \ast |
| <code>\$\$ \bigodot \$\$</code> | \bigodot |
| <code>\$\$ \bigotimes \$\$</code> | \bigotimes |
| <code>\$\$ \bigoplus \$\$</code> | \bigoplus |
| <code>\$\$ \leq \$\$</code> | \leq |
| <code>\$\$ \geq \$\$</code> | \geq |
| <code>\$\$ \neq \$\$</code> | \neq |
| <code>\$\$ \approx \$\$</code> | \approx |
| <code>\$\$ \equiv \$\$</code> | \equiv |
| <code>\$\$ \sum \$\$</code> | \sum |
| <code>\$\$ \prod \$\$</code> | \prod |
| <code>\$\$ \coprod \$\$</code> | \coprod |

集合运算符:

| 语法 | 显示效果 |
|----------------------------------|-------------|
| <code>\$\$ \emptyset \$\$</code> | \emptyset |
| <code>\$\$ \in \$\$</code> | \in |
| <code>\$\$ \notin \$\$</code> | \notin |
| <code>\$\$ \subset \$\$</code> | \subset |
| <code>\$\$ \supseteq \$\$</code> | \supseteq |
| <code>\$\$ \subseteq \$\$</code> | \subseteq |
| <code>\$\$ \supseteq \$\$</code> | \supseteq |
| <code>\$\$ \bigcap \$\$</code> | \bigcap |
| <code>\$\$ \bigcup \$\$</code> | \bigcup |
| <code>\$\$ \bigvee \$\$</code> | \bigvee |
| <code>\$\$ \bigwedge \$\$</code> | \bigwedge |
| <code>\$\$ \biguplus \$\$</code> | \biguplus |
| <code>\$\$ \bigsqcup \$\$</code> | \bigsqcup |

对数运算符：

| 语法 | 显示效果 |
|-----------------------------|--------|
| <code>\$\$ \log \$\$</code> | \log |
| <code>\$\$ \lg \$\$</code> | \lg |
| <code>\$\$ \ln \$\$</code> | \ln |

三角运算符：

| 语法 | 显示效果 |
|---------------------------------|------------|
| <code>\$\$ \bot \$\$</code> | \bot |
| <code>\$\$ \angle \$\$</code> | \angle |
| <code>\$\$ 30^\circ \$\$</code> | 30° |
| <code>\$\$ \sin \$\$</code> | \sin |
| <code>\$\$ \cos \$\$</code> | \cos |
| <code>\$\$ \tan \$\$</code> | \tan |
| <code>\$\$ \cot \$\$</code> | \cot |
| <code>\$\$ \sec \$\$</code> | \sec |
| <code>\$\$ \csc \$\$</code> | \csc |

微积分运算符：

| 语法 | 显示效果 |
|-------------------------------|----------|
| <code>\$\$ \prime \$\$</code> | \prime |
| <code>\$\$ \int \$\$</code> | \int |
| <code>\$\$ \iint \$\$</code> | \iint |
| <code>\$\$ \iiint \$\$</code> | \iiint |
| <code>\$\$ \iiint \$\$</code> | \iiint |
| <code>\$\$ \oint \$\$</code> | \oint |
| <code>\$\$ \lim \$\$</code> | \lim |

| | |
|-------------------------------|----------|
| <code>\$\$ \infty \$\$</code> | ∞ |
| <code>\$\$ \nabla \$\$</code> | ∇ |

逻辑运算符：

| 语法 | 显示效果 |
|------------------------------------|---------------|
| <code>\$\$ \because \$\$</code> | \because |
| <code>\$\$ \therefore \$\$</code> | \therefore |
| <code>\$\$ \forall \$\$</code> | \forall |
| <code>\$\$ \exists \$\$</code> | \exists |
| <code>\$\$ \not= \$\$</code> | \neq |
| <code>\$\$ \not> \$\$</code> | $\not>$ |
| <code>\$\$ \not\subset \$\$</code> | $\not\subset$ |

戴帽符号：

| 语法 | 显示效果 |
|----------------------------------|-------------|
| <code>\$\$ \hat{y} \$\$</code> | \hat{y} |
| <code>\$\$ \check{y} \$\$</code> | \check{y} |
| <code>\$\$ \breve{y} \$\$</code> | \breve{y} |

连线符号：

| 语法 | 显示效果 |
|---|---|
| <code>\$\$ \overline{a+b+c+d} \$\$</code> | $\overline{a+b+c+d}$ |
| <code>\$\$ \underline{a+b+c+d} \$\$</code> | $\underline{a+b+c+d}$ |
| <code>\$\$ \overbrace{a+\underbrace{b+c}\{1.0\}+d\}^{2.0} \$\$</code> | $\overbrace{a+\underbrace{b+c}\{1.0\}+d}^{2.0}$ |

箭头符号：

哎，这里的空白这蛋疼，目前还没有最终得解决发子

| 语法 | 显示效果 |
|-------------------------------------|----------------|
| <code>\$\$ \uparrow \$\$</code> | \uparrow |
| <code>\$\$ \downarrow \$\$</code> | \downarrow |
| <code>\$\$ \Uparrow \$\$</code> | \Uparrow |
| <code>\$\$ \Downarrow \$\$</code> | \Downarrow |
| <code>\$\$ \rightarrow \$\$</code> | \rightarrow |
| <code>\$\$ \leftarrow \$\$</code> | \leftarrow |
| <code>\$\$ \Rrightarrow \$\$</code> | \Rrightarrow |
| <code>\$\$ \Lleftarrow \$\$</code> | \Lleftarrow |

| | |
|--|-------------------|
| <code>\$\$ \longrightarrow \$\$</code> | \longrightarrow |
| <code>\$\$ \longleftarrow \$\$</code> | \longleftarrow |
| <code>\$\$ \Longrightarrow \$\$</code> | \Longrightarrow |
| <code>\$\$ \Longleftarrow \$\$</code> | \Longleftarrow |

要输出字符

哎，这里的空白这蛋疼，目前还没有最终得解决发子

| 语法 | 显示效果 |
|---------------------|------|
| <code>\空格</code> | a\ b |
| <code>\#</code> | # |
| <code>\\$</code> | \$ |
| <code>\%</code> | % |
| <code>\&</code> | & |
| <code>_</code> | — |
| <code>{ }</code> | { } |

这里的空格没有搞定，显示没有按预期的效果来，#下面的，在MarkdownPad2中默认是不需要转换，所以加\没效果

如何进行字体转换

要对公式的某一部分字符进行字体转换，可以用`$$ {\rm 需转换的部分字符} $$`命令，其中`\rm`可以参照下表选择合适的字体。
一般情况下，公式默认为意大利体。

| 语法 | 字体名称 | 显示效果 |
|---|-------------------|-------------------------------------|
| <code>\$\$ {\rm 需转换的部分字符} \$\$</code> | 罗马体 | <code>\$\$ {\rm ABCD} \$\$</code> |
| <code>\$\$ {\it 需转换的部分字符} \$\$</code> | 意大利体 | <code>\$\$ {\it ABCD} \$\$</code> |
| <code>\$\$ {\Bbb 需转换的部分字符} \$\$</code> | 黑板粗体字 | <code>\$\$ {\Bbb ABCD} \$\$</code> |
| <code>\$\$ {\bf 需转换的部分字符} \$\$</code> | 黑体 | <code>\$\$ {\bf ABCD} \$\$</code> |
| <code>\$\$ {\cal 需转换的部分字符} \$\$</code> | 花体 | <code>\$\$ {\cal ABCD} \$\$</code> |
| <code>\$\$ {\sl 需转换的部分字符} \$\$</code> | 倾斜体 | <code>\$\$ {\sl ABCD} \$\$</code> |
| <code>\$\$ {\sf 需转换的部分字符} \$\$</code> | 等线体 | <code>\$\$ {\sf ABCD} \$\$</code> |
| <code>\$\$ {\mit 需转换的部分字符} \$\$</code> | 数学斜体 | <code>\$\$ {\mit ABCD} \$\$</code> |
| <code>\$\$ {\tt 需转换的部分字符} \$\$</code> | 打字机字体 | <code>\$\$ {\tt ABCD} \$\$</code> |
| <code>\$\$ {\scr 需转换的部分字符} \$\$</code> | 小体大写字母 | <code>\$\$ {\scr ABCD} \$\$</code> |
| <code>\$\$ {\frak 需转换的部分字符} \$\$</code> | Fraktur字母（一种德国字体） | <code>\$\$ {\frak ABCD} \$\$</code> |

先到这里把，这个字体转换还没有彻底的明白，其中有几个还没有成功，对于MarkdownPad2支持Latex确实有点蛋疼，实时预览看不到，需要F6 html预览才行，另外还得额外加载相关js代码

[跳转至首页](#)