

最全的Markdown语法

最全的Markdown语法

- 目录
- 多级标题
- 引用与注释
- 插入代码
 - 行内代码
 - 代码段
- 多级列表
 - 无序列表
 - 有序列表
- 超链接
 - 行内超链接
 - 参数式超链接
- 字体
- 表格
- 图片
 - 设置图片大小
 - Markdown自带功能
 - 使用HTML 标签并设置width和height和比例放缩
- 多选框
- 脚注 (footnote)
- 流程图
 - 流程图语法介绍
 - 声明元素
 - 定义流程

目录

- 代码:

[TOC]
[MENU]

- 效果显示如上

多级标题

- 代码:

一级标题
二级标题
多级标题

- 效果

一级标题

二级标题

多级标题

引用与注释

- 代码：

```
> 一级引用
>> 二级引用
>>>> n级引用
```

- 效果

一级引用

二级引用

n级引用

插入代码

行内代码

- 代码：

```
`行内代码`
```

- 效果 `行内代码`

代码段

这是代码 //代码中无视markdown语法，所以本文中所有展示markdown原始文本的部分都是用代码段包起来的。

- 代码：

```
这是代码
```javascript
var s = "JavaScript syntax highlighting";
alert(s);
```

```python
s = "Python syntax highlighting"
print s
```

```markdown
```

```
> 一级引用
>> 二级引用
>>>> n级引用
...

```

## 多级列表

---

### 无序列表

- 代码：

```
// - * + 三个符号在无序列表中地位平等
- 第一项
* 第二项
 * 第二项//前面加一个空格或tab就表示下一级
 * 这又是一级
+ 第三项

```

- 效果

- 
- 第一项
  - 第二项
    - 第二项//前面加一个空格或tab就表示下一级
      - 这又是一级
  - 第三项

### 有序列表

- 代码：

```
//数字加.加空格表示一个有序列表项，tab或空格触发下一级
1. 这是有序的
 1. 这是有序的
2. 这是有序的
 1. 这是有序的

```

- 1. 这是有序的
  - 1. 这是有序的
- 2. 这是有序的
  - 1. 这是有序的

## 超链接

---

## 行内超链接

- 代码：

```
[百度](baidu.com)
```

- 效果

[百度](#)

## 参数式超链接

- 代码：

```
声明：[百度]:baidu.com
使用：进入[百度]
```

## 字体

- 代码：

```
斜体
粗体
斜体加粗
删除线
```

- 效果

*斜体*  
**粗体**  
***斜体加粗***  
~~删除线~~

## 表格

- 代码：

```
| Tables | Are | Cool |
| :-----: | :-----: | -----:|
| col 3 is | right-aligned | $1600 |
| col 2 is | centered | $12 |
| zebra stripes | are neat | $1 |

注释：
-----: 表示右对齐
:----- 表示左对齐
:-----: 表示居中对齐
```

- 效果

Tables	Are	Cool
col 3 is	right-aligned	\$1600
col 2 is	centered	\$12
zebra stripes	are neat	\$1

## 图片

- 代码:

```
! [图片] (图片地址)
//注意，默认图片格式是原始尺寸，如果想自定义图像大小，要用img标签:

Inline-style:
! [alt text] (https://gitee.com/msntec/work-
notes/raw/master/Markdown/pic/markdown_flag.png "Logo Title Text 1")

Reference-style:
! [alt text] [logo]

[logo]: https://gitee.com/msntec/work-
notes/raw/master/Markdown/pic/markdown_flag.png "Logo Title Text 2"
```

Inline-style:



Reference-style:



## 设置图片大小

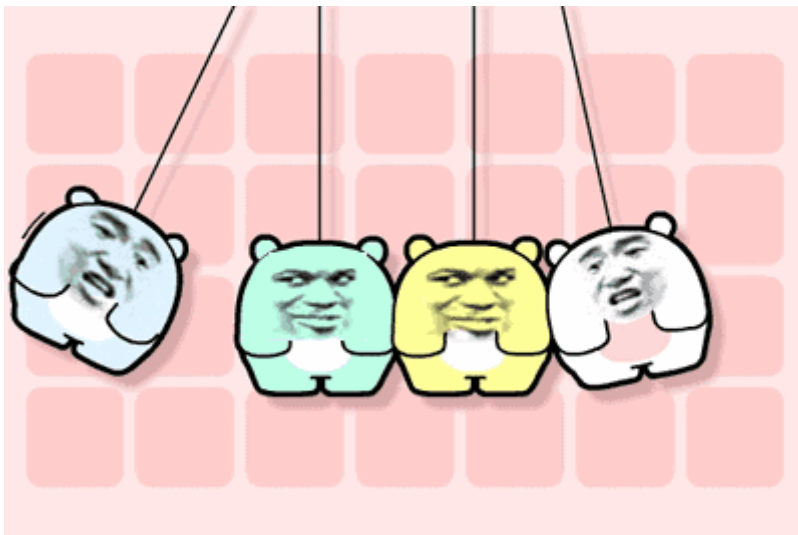
### Markdown自带功能

- 代码:

```
![loading.png](http://upload-images.jianshu.io/upload_images/1503319-c696a9cd1495d68f.png?imageMogr2/auto-orient/strip%7CimageView2/2/w/1240)
```

```
![loading.png](http://upload-images.jianshu.io/upload_images/1503319-c696a9cd1495d68f.png?imageMogr2/auto-orient/strip%7CimageView2/2/w/200)
```

- 效果显示



### 使用HTML <img>标签并设置width和height和比例放缩

- 代码:

```



```

- 效果显示





## 多选框

- 代码：

```
- [x] A
- [] B
- [x] 已选中
- [] 未选中

- [] 支持以 PDF 格式导出文稿
- [] 改进 Cmd 渲染算法，使用局部渲染技术提高渲染效率
- [x] 新增 Todo 列表功能
- [x] 修复 LaTeX 公式渲染问题
- [x] 新增 LaTeX 公式编号功能
```

- 效果显示

☒ A  
☐ B  
☒ 已选中  
☐ 未选中

☐ 支持以 PDF 格式导出文稿  
☐ 改进 Cmd 渲染算法，使用局部渲染技术提高渲染效率  
☒ 新增 Todo 列表功能  
☒ 修复 LaTeX 公式渲染问题  
☒ 新增 LaTeX 公式编号功能

## 脚注 (footnote)

- 代码：

```
文章正文某处[^脚注id]文章正文
[^脚注id]: 脚注内容
```

- 效果显示

文章正文某处<sup>1</sup>文章正文



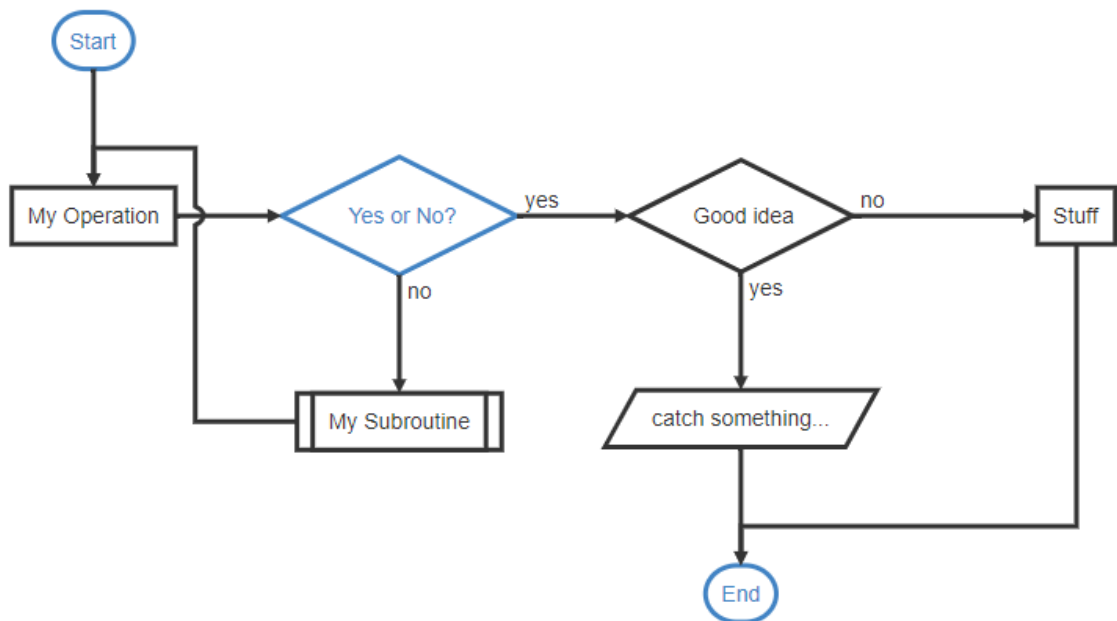
# 流程图

- 代码:

```
```flow
st=>start: Start :>https://amwiki.xf09.net[blank]
e=>end: End :>https://amwiki.xf09.net[blank]
op1=>operation: My Operation
op2=>operation: Stuff
sub1=>subroutine: My Subroutine
cond=>condition: Yes or No? :>https://amwiki.xf09.net[blank]
c2=>condition: Good idea
io=>inputoutput: catch something...

st->op1(right)->cond
cond(yes, right)->c2
cond(no)->sub1(left)->op1
c2(yes)->io->e
c2(no)->op2->e
```
```

- 效果显示



## 流程图语法介绍

流程图语法分两个部分，一个是声明元素，一个是定义流程

### 声明元素

语法:

```
tag=>type: content :>url
```

1. `tag` 设置元素名称
2. `=>` 元素定义符
3. `type`: 设置元素类型，共分6种:

- start: 开始, 圆角矩形
  - end: 结束, 圆角矩形
  - operation: 操作/行动方案, 普通矩形
  - subroutine: 子主题/模块, 双边线矩形
  - condition: 条件判断/问题审核, 菱形
  - inputoutput: 输入输出, 平行四边形
4. `content` 设置元素显示内容, 中英均可
  5. `:>url` 设置元素连接, 可选, 后接 [blank] 可以新建窗口打开

提示: 注意空格, `=>` 前后都不能接空格; `type:` 后必须接空格; `:>` 是语法标记, 中间不能有空格

## 定义流程

语法:

```
tag1(branch,direction)->tag2
```

1. `->` 流程定义符, 连接两个元素
2. `branch` 设置 `condition` 类型元素的两个分支, 有 yes/no 两个值, 其他元素无效
3. `direction` 定义流程走线方向, 有 left/right/top/bottom 四个值, 所有元素有效, 此项配置可选  
(PS: 此属性目前有一定几率触发图形错位, 刷新即可)

小提示:

- 继续注意空格, `->` 前后都不能有空格
- 由于 `condition` 类型有两个分支, 我们一般遇到 `condition` 元素就换行书写, 比如:

```
st->op1-c2
c2(yes)->io->e
c2(no)->op2->e
```