

- Markdown文档规范
- Markdown语法
  - 标题
  - 列表
  - 引用
  - 粗体与斜体
  - 分割线
  - 删除线
  - 反斜杠
  - 图片与链接
  - 颜色
  - 表格
  - 代码框

2018-06-05，由[朱寅](#)创建。

## Markdown文档规范

---

1. 所有软件的设计、开发、测试、部署文档需使用Markdown格式；
2. Markdown文档建议使用 Typora 编写；
  - 例如，比如Typora插入表格， 菜单 `paragraph` -> `Table` ，然后输入行数和列数就ok了
3. Markdown文档的后缀必须是 `.md` ；
4. 命令、代码、日志输出等请使用代码标记；
5. 避免使用截图；
6. 文档中请避免使用 `html` 标记；

## Markdown语法

---

### 标题

---

如果一段文字被定义为标题，只要在这段文字前加 # 号即可。

```
# 一级标题
## 二级标题
### 三级标题
```

以此类推，总共六级标题，建议在井号后加一个空格，这是最标准的 Markdown 语法。

## 列表

---

列表的显示只需要在文字前加上 -，+ 或 \* 即可变为无序列表. 有序列表则直接在文字前加 1.、2.、3.，符号要和文字之间加上一个字符的空格。

## 引用

---

如果你需要引用一小段别处的句子，那么就要用引用的格式。

例如这样

只需要在文本前加入 > 这种尖括号（大于号）即可

## 粗体与斜体

---

Markdown 的粗体和斜体也非常简单，用两个 \* 包含一段文本就是粗体的语法，用一个 \* 包含一段文本就是斜体的语法。

加粗 **\*\*加粗\*\*** 斜体 *\*斜体\**

## 分割线

---

分割线的语法只需要三个 \*、- 或 \_ 号。

---

## 删除线

---

使用 `~~` 表示删除线。 ~~~~这是一条删除线~~~~ 这是一条删除线

## 反斜杠

使用 `\` 表示反斜杠。在你不想显示Markdown标记时可以使用反斜杠。

## 图片与链接

插入链接与插入图片的语法很像，区别在一个 `!` 号

图片为： `![]()`  
链接为： `[]()`

[度娘](http://www.baidu.com) [度娘](http://www.baidu.com)

## 颜色

采用html标记设定颜色。

`<font color="...">` 和 `<body bgcolor="...">` 等属性可以用以下的方法来指定颜色。  
`color="#FF0000"` — 指定RGB值。  
`color="red"` — 指定颜色的名字。

示例：

- `<font color='Red'>我是红色</font>` 的效果是：我是红色。注意，导出为pdf时会为红色。

RGB值是用16进制数来描述光的三原色赤(R)、绿(G)、蓝(B)的数值。

通常值越大色彩越明亮，值越小色彩越灰暗。

- Black (#000000)
- Gray (#808080)
- Silver (#C0C0C0)

- White (#FFFFFF)
- Red (#FF0000)
- Yellow (#FFFF00)
- Lime (#00FF00)
- Aqua (#00FFFF)
- Blue (#0000FF)
- Fuchsia (#FF00FF)
- Maroon (#800000)
- Olive (#808000)
- Green (#008000)
- Teal (#008080)
- Navy (#000080)
- Purple (#800080)

## 表格

参数名称	类型	LeftAligned	Right Aligned
time	long	是	用户发起请求时的unix时间戳, 精确到毫秒
sig	string	是	签名值

### 效果

参数名称	类型	LeftAligned	Right Aligned
time	long	是	用户发起请求时的unix时间戳, 精确到毫秒
sig	string	是	签名值

# 代码框

---

java 代码

```
```java
public static void main(String[] args) {
    SpringApplication.run(HeIndexApp.class, args);
}
```

效果

```
```java
public static void main(String[] args) {
    SpringApplication.run(HeIndexApp.class, args);
}
```

xml代码

```
```xml
<parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.0.2.RELEASE</version>
</parent>
```

效果

```
```xml
<parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.0.2.RELEASE</version>
</parent>
```

其他的有：

- js 或 json

- sh 或 shell
- sql
- ...