



链滴

Markdown 使用指南 - 扩展语法

作者: [88250](#)

原文链接: <https://ld246.com/article/1583305480675>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<blockquote>
 <p>本文主要介绍 Markdown 的扩展语法，Markdown 基础语法请参考这里。

 简洁版语法可浏览速查手册。</p>
 </blockquote>
 <h2 id="总览">总览</h2>
 <p>在基础语法中我们介绍了 Markdown 最常用的排版用法，但有些时候基础语法不足以满足复杂一些的排版需求，这时候就需要使用扩展语法了。</p>
 <p>一些个人和组织对基础语法进行了扩展，比如引入表格、围栏代码块、代码高亮、自动链接、脚和目录等，这些扩展语法需要特定的 Markdown 引擎来支持。</p>
 <h2 id="可用性">可用性</h2>
 <p>不是所有的 Markdown 引擎都支持扩展语法，所以在使用扩展语法之前，你可能需要对具体的 markdown 引擎支持情况进行确认，通过浏览它们的使用文档来了解具体哪些元素可以得到支持。</p>
 <h3 id="主流的-Markdown-语法规范">主流的 Markdown 语法规范</h3>
 <p>Markdown 语法规范目前尚未形成标准，如下列出的几种语法规范较为主流。</p>

 CommonMark
 GitHub Flavored Markdown (GFM)
 Markdown Extra
 MultiMarkdown
 R Markdown

 <p>其中 CommonMark 和 GFM 是目前最有可能进行标准化的。GFM 是 GitHub 基于 CommonMark 进行的扩展，它几乎已经是事实标准了，关于 CommonMark 规范要点解读可参考这里。</p>
 <h3 id="Markdown-引擎">Markdown 引擎</h3>
 <p>这里列出了很多 Markdown 引擎实现，它们中的大多数都支持通过设置开关来都支持扩展法，具体细节请参考它们的文档。</p>
 <h2 id="表格">表格</h2>
 <p>使用短横线 <code>----</code> 来分隔表头和表身，使用竖线 <code>|</code> 来分隔列，行开头和结尾的竖线是可选的。</p>
 <pre><code class="language-md highlight-chroma">| Syntax | Description |
|-----|-----|
| Header | Title |
| Paragraph | Text |
</code></pre>
 <p>渲染结果：</p>
 <table>
 <thead>
 <tr>
 <th>Syntax</th>
 <th>Description</th>
 </tr>
 </thead>
 <tbody>
 <tr>

```
<td>Header</td>
```

```
<td>Title</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Paragraph</td>
```

```
<td>Text</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

<p>每个列不一定要对齐，如下示例同样能够渲染： </p>

```
<pre><code class="language-md highlight-chroma">| Syntax | Description |
```

```
| --- | ----- |
```

```
| Header | Title |
```

```
| Paragraph | Text |
```

```
</code></pre>
```

表格对齐</h3>

<p>如果需要左对齐、居中对齐或者右对齐表格内容，可以通过在 <code>---</code> 中添加冒号 code>:</code> 实现。冒号仅出现在左边表示左对齐，出现在两边表示居中对齐，仅出现在右边表示右对齐。</p>

```
<pre><code class="language-md highlight-chroma">| Syntax    | Description | Test Text  |
```

```
| :---      | :----:    | ---:      |
```

```
| Header    | Title     | Here's this |
```

```
| Paragraph | Text      | And more    |
```

```
</code></pre>
```

<p>渲染结果： </p>

```
<table>
```

```
<thead>
```

```
<tr>
```

```
<th align="left">Syntax</th>
```

```
<th align="center">Description</th>
```

```
<th align="right">Test Text</th>
```

```
</tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr>
```

```
<td align="left">Header</td>
```

```
<td align="center">Title</td>
```

```
<td align="right">Here's this</td>
```

```
</tr>
```

```
<tr>
```

```
<td align="left">Paragraph</td>
```

```
<td align="center">Text</td>
```

```
<td align="right">And more</td>
```

```
</tr>
```

```
</tbody>
```

```
</table>
```

表格内容排版</h3>

<p>表格中的内容也可以进行排版，比如加粗、强调文本，插入超链接等。但仅限于使用“行级元素”进行排版，不能使用“块级元素”，比如不能使用标题、块引用、列表、分隔线等。</p>

表格内容转义竖线</h3>

<p>如果需要在表格内容中使用竖线 <code>|</code>，那就需要对其进行转义。可以使用 <code>|</code> 转义，但更稳妥的做法是写竖线的 HTML 实体表示 <code>|</code>，因有的 Markdown 引擎不能正确处理表格内容中的 <code>|</code>。</p>

围栏代码块</h2>

在<https://ld246.com/article/1583129520165>基础语法中我们介绍过可以通过四个空格或者制表符 `\t` 缩进来排版代码块。但我们更推荐的是使用围栏代码块语，即使用大于等于三个反引号 ````` 或者大于等于三个波浪线 `~~~` 来包裹代码块内容：

```
<pre> <code class="language-md highlight-chroma"> <span class="highlight-s">```
</span> <span class="highlight-s"> </span>{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25
}
</span class="highlight-s">``` </span>
</code> </pre>
```

渲染结果：

```
<pre> <code class="language-text highlight-chroma">{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25
}
</code> </pre>
```

如果需要展示代码块原文 Markdown（就像上面展示的例子那样），可以在最外层使用更多数的反引号开始，闭合的反引号数量等于开始的数量即可。

```
<pre> <code class="language-md highlight-chroma">````
<span class="highlight-s">```
</span> <span class="highlight-s"> </span>{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25
}
<span class="highlight-s">``` </span>
````
```

```
</code> </pre>
```

### 代码块语法高亮

代码块语法高亮需要 Markdown 引擎支持，通过在开始反引号后添加编程语言名称来排版。

```
<pre> <code class="language-md highlight-chroma"> ```json
 {
 "firstName" : "John" ,
 "lastName" : "Smith" ,
 "age" : 25
}
```
</code> </pre>
```

渲染结果：

```
<pre> <code class="language-json highlight-chroma"> {
 "firstName" : "John" ,
 "lastName" : "Smith" ,
 "age" : 25
}
</code> </pre>
```

```
</code></pre>
```

```
<h2 id="脚注">脚注</h2>
```

```
<p>脚注用于在文末添加细节说明或者参考，这样文章的正文部分看上去会更加简洁清晰。创建脚注，正文中引用脚注的地方会出现一个上标数字链接，读者点击后跳转到文末脚注定义的对应位置。</p>
```

```
<p>脚注引用通过 <code>[^标识符]</code> 创建，标识符部分可以是数字或者文本，但不能包含格或者制表符。标识符仅用于关联引用和定义，在渲染时会自动根据脚注定义顺序进行数字递增渲染。不过这也不是绝对的，某些 Markdown 引擎也会将标识符部分用于渲染。</p>
```

```
<p>脚注定义使用 <code>[^标识符]:</code> 来创建，冒号后面就是需要添加的细节说明或者参。脚注定义不一定非要放在整个 Markdown 文本的末尾，夹在段落、列表或者块引用之间也是可以作的。</p>
```

```
<pre><code class="language-md highlight-chroma">这里是一个脚注引用[^1]，这里是另一个注引用[^bignote].</pre>
```

```
<p>渲染结果: </p>
```

```
<div class="vditor-reset"><p>这里是一个脚注引用^{1}，这里是另一个脚注引用^{2}。</p>
```

```
<div>
```

```
<hr>
```

```

```

```
<li id="footnotes-def-1"><p>第一个脚注定义。 </p>
```

```
<li id="footnotes-def-2"><p>脚注定义可使用多段内容。</p> <p>缩进对齐的段落包含在这个注定义内。</p> <pre><code>可以使用代码块。</code></pre> <p>还有其他行级排版语法，比如加粗和链接。 </p>
```

```

```

```
</div></div>
```

```

```

```
</div></div>
```

```
<h2 id="heading-ids">标题指定 ID</h2>
```

```
<p>一些 Markdown 引擎支持为标题指定 ID，另一些 Markdown 引擎是自动添加 ID 的。标题 ID 作用是允许其他地方通过锚点直接跳转到该标题。标题指定 ID 的语法是在标题后面通过花括号包裹 I。</p>
```

```
<pre><code class="language-md highlight-chroma">### 这是个标题 {#custom-id}</code></pre>
```

```
</code></pre>
```

```
<p>渲染的 HTML: </p>
```

```
<pre><code class="language-html highlight-chroma"><</pre>
```

```
h3 id = "custom-id" ></pre>
```

```
<这是一个标题< h3 ></pre>
```

```
</code></pre>
```

```
<h3 id="链接到指定的标题">链接到指定的标题</h3>
```

```
<p>可以通过超链接语法链接到文中的标题。</p>
```

```
<table>
```

```
<thead>
```

```
<tr>
```

```
<th>Markdown</th>
```

```
<th>HTML</th>
```

```
<th>渲染结果</th>
```

```
</tr>
```

```
</thead>
```

```

<tbody>
<tr>
<td><code>[标题指定 ID](#heading-ids)</code></td>
<td><code>标题指定 ID</code></td>
<td>标题指定 ID</td>
</tr>
</tbody>
</table>
<p>如果需要链接到其他页面的标题，需要写全链接路径，比如 <code>[标题指定 ID](https://ld246.com/article/1583305480675#heading-ids)</code>。</p>
<h2 id="删除线">删除线</h2>
<p>可以通过删除线划掉文本，排版结果就像这样。创建删除线可以通过一个 <code></code> 或两个波浪线 <code>~~</code> 包裹待删除的文本。</p>
<pre><code class="highlight-chroma">~~地球是平的。~~ 现在我们知道地球是圆的了。</code></pre>
<p>渲染结果：</p>
<p>地球是平的。 现在我们知道地球是圆的了。</p>
<h2 id="任务列表">任务列表</h2>
<p>通过在普通列表项中添加 <code>[</code> 或者 <code>[x]</code> 来渲染任务列表项。<
p>
<pre><code class="language-md highlight-chroma">- [x]</spa
> 待办事项一
- [待办事项二
- [待办事项三
</code></pre>
<p>渲染结果：</p>

<li class="vditor-task vditor-task--done"><input checked disabled type="checkbox"> 待办
项一
<li class="vditor-task"><input disabled type="checkbox"> 待办事项二
<li class="vditor-task"><input disabled type="checkbox"> 待办事项三

<h2 id="Emoji-表情">Emoji 表情</h2>
<p>有两种方法使用 Emoji 表情：直接输入 Emoji 字符或者使用别名 <code>:smile:</code>。直
输入的话需要确认当前编辑器使用 UTF-8 编码。</p>
<pre><code class="language-md highlight-chroma">今天天气真好 :sunny:
花儿 :sunflower: 对我笑
</code></pre>
<p>渲染结果：</p>
<p>今天天气真好 :sunny:

花儿 :sunflower: 对我笑</p>
<h2 id="禁止自动链接">禁止自动链接</h2>
<p>大部分 Markdown 引擎都是默认开启自动链接的，所以当我们想把一个链接渲染为纯文本时，
要把它变成代码：</p>
<pre><code class="language-md highlight-chroma">`https://b3
og.org`
</code></pre>
<p>渲染结果：</p>
<p><code>https://b3log.org</code></p>
<h2 id="总结">总结</h2>
<p>尽量仅使用 GFM 定义的扩展语法，这样能够最大限度得到众多 Markdown 引擎的支持，从而
到一致的渲染结果。</p>
<h2 id="参考">参考</h2>

```

<ul>

<li><a href="https://link.ld246.com/forward?goto=https%3A%2F%2Fwww.markdownguide.org" target="\_blank" rel="nofollow ugc">Markdown Guide</a> </li>

<li><a href="https://link.ld246.com/forward?goto=https%3A%2F%2Fspec.commonmark.org" target="\_blank" rel="nofollow ugc">CommonMark Spec</a> </li>

</ul>

1. 第一个脚注定义。
2. 脚注定义可使用多段内容。

缩进对齐的段落包含在这个脚注定义内。

可以使用代码块。

还有其他行级排版语法，比如**加粗**和[链接](#)。

```
</code> </pre>
```