

Markdown 快速入门指南

1. 标题

```
# 一级标题  
## 二级标题  
### 三级标题  
#### 四级标题  
##### 五级标题  
###### 六级标题
```

效果：

一级标题

二级标题

三级标题

四级标题

五级标题

六级标题

2. 文本格式

2.1 基本样式

加粗文本 或 加粗文本
斜体文本 或 斜体文本
加粗斜体 或 加粗斜体
~~删除线文本~~

效果：

- 加粗文本
- 斜体文本
- **加粗斜体**
- 删除线文本

2.2 段落和换行

这是第一段文字。

(空一行)

这是第二段文字。

这是第一行

这是第二行 (两个空格+回车)

这是第三行

3. 列表

3.1 无序列表

- 项目一
 - 项目二
 - 子项目一 (缩进2空格)
 - 子项目二
- * 也可以使用星号
- + 或者加号

效果：

- 项目一
- 项目二
 - 子项目一
 - 子项目二
- 也可以使用星号
- 或者加号

3.2 有序列表

1. 第一项
2. 第二项
 1. 子项一（缩进3空格）
 2. 子项二
3. 第三项

效果：

1. 第一项
2. 第二项
 - i. 子项一
 - ii. 子项二
3. 第三项

3.3 任务列表

- [x] 已完成任务
- [] 未完成任务
- [] 另一个任务

效果：

- 已完成任务
- 未完成任务
- 另一个任务



4. 链接和图片

4.1 链接

[百度](<https://www.baidu.com>)

[带标题的链接](<https://example.com> "鼠标悬停文字")

直接链接：

<<https://www.example.com>>

效果：

百度

<https://www.example.com>

4.2 图片

![图片描述](<https://via.placeholder.com/150x100> "可选标题")

带链接的图片：

[![图片](图片地址)](链接地址)



5. 代码

5.1 行内代码

使用 `console.log()` 函数打印信息

效果：

使用 console.log() 函数打印信息

5.2 代码块

```
```javascript
function hello(name) {
 console.log(`Hello, ${name}!`);
 return true;
}
````
```

```
```python
def hello(name):
 print(f"Hello, {name}!")
 return True
````
```

```
```bash
npm install package-name
git commit -m "message"
````
```

效果：

```
function hello(name) {
    console.log(`Hello, ${name}!`);
    return true;
}
```



6. 表格

| 姓名 | 年龄 | 城市 |
|----|----|----|
| 张三 | 25 | 北京 |
| 李四 | 30 | 上海 |
| 王五 | 28 | 广州 |

对齐方式：

| 左对齐 | 居中对齐 | 右对齐 |
|-----|--------|--------|
| : | -----: | -----: |
| 内容 | 内容 | 内容 |

效果：

| 姓名 | 年龄 | 城市 |
|----|----|----|
| 张三 | 25 | 北京 |
| 李四 | 30 | 上海 |
| 王五 | 28 | 广州 |

7. 引用

- > 这是一段引用
- > 可以有多行
- >
- > 空行分隔不同引用块

- > 嵌套引用
- > > 二级引用
- > > > 三级引用

效果：

这是一段引用
可以有多行

| 空行分隔不同引用块

| 嵌套引用

|| 二级引用

||| 三级引用

8. 分割线

三个短横线：

三个星号：

三个下划线：

效果：

9. HTML标签支持

部分HTML标签可以在Markdown中使用：

```
<span style="color:red">红色文字</span>
<kbd>Ctrl+C</kbd>
<br>换行
<center>居中内容</center>
```

效果：

红色文字

Ctrl+C

◆ 10. 转义字符

```
\# 显示为普通#号
\* 显示为普通星号
\[ \] 显示为方括号
\( \) 显示为括号
\_ 显示为下划线
\\ 显示反斜杠
```

效果：

```
# * _ \
```

实战示例

10.1 个人笔记模板

```
# [主题] - 学习笔记
```

```
## 📚 学习目标
```

1. 掌握基本概念
2. 完成实践练习
3. 总结重点难点

```
## 🔎 重点内容
```

- **关键概念1**：定义和特点
- **关键概念2**：应用场景
- **注意事项**：常见问题

```
## 💻 代码示例
```

```
```python
def example():
 print("Hello Markdown")
 return True
```

# 总结

| 核心要点总结在这里

## 参考资料

- [官方文档](#)
- [相关教程](#)

```
10.2 项目README模板 {#102-项目readme模板 }
```markdown
# 项目名称
```

简短的项目描述。

```
## 🌟 特性
- 特性一
- 特性二
- 特性三
```

```
## 🚀 快速开始
```

```
### 安装
```bash
npm install project-name
```

## 基本使用

```
const project = require('project-name');
project.init();
```

## 文档

详细文档请查看 [Wiki](#)



# 贡献

欢迎提交 Issue 和 Pull Request

## 许可证

MIT License

---

## 📄 常用工具推荐 {#-常用工具推荐 }

工具	用途	平台
Typora	实时预览编辑器	Win/Mac/Linux
VS Code	代码编辑器+Markdown扩展	全平台
Obsidian	知识管理工具	全平台
Notion	在线笔记	Web/App
MarkText	开源编辑器	Win/Mac/Linux

---

## 💡 温馨提示 {#-温馨提示 }

- \*\*先学常用功能\*\*：掌握标题、列表、链接、代码块即可应对80%场景
- \*\*保持简洁\*\*：Markdown的优势在于简洁易读
- \*\*多实践\*\*：在GitHub、博客、笔记中多使用
- \*\*注意兼容性\*\*：不同平台可能支持不同的扩展语法

> \*\*记住\*\*：Markdown不是排版工具，而是\*\*写作工具\*\*。专注于内容，而不是样式。

---

\*\*立即开始\*\*：打开任意文本编辑器，保存为 ` `.md` 文件，开始你的Markdown写作之旅吧！