

git commit 提交规范

git每次提交代码，都要写commit message（提交说明）

```
git commit -m "hello world"
```

上面中的 `-m` 参数就是用来执行commit message信息

一、commit message 的作用

格式化的commit message，有几个好处

1. 提供更多的历史信息，方便快速浏览
比如使用 `git log` 命令之后，每个commit占据一行，只看行首就可以知道此次commit的目的
2. 可以过滤某些commit，便于快速查找信息
比如使用 `git log --grep feature` 命令之后仅显示log中新增的功能
3. 可以直接从commit生成change log

二、commit message 的格式

每次提交，Commit message 都包括三个部分：Header，Body 和 Footer。

```
<type>(<scope>): <subject>
// 空一行
<body>
// 空一行
<footer>
```

每次提交Header是必需的，Body 和 Footer可以省略

1. Header

Header部分只有一行，包括三个字段：`type`（必需）、`scope`（可选）和 `subject`（必需）。

type：用于说明commit的类型，只允许使用下面7个标识

- o feat：新功能（feature）
- o fix：修补bug
- o docs：文档（documentation）
- o style：格式（不影响代码运行的变动）
- o refactor：重构（即不是新增功能，也不是修改bug的代码变动）
- o test：增加测试
- o chore：构建过程或辅助工具的变动

scope：用于说明commit影响的范围，比如数据层、控制层等等，根据实际的项目来

subject：用于说明commit目的的简短描述

比如一个完整的功能性commit的提交格式就是 `git commit -m 'feat(page): 添加驾驶员刷卡报表页面'`

2. Body

Body 部分是对本次 commit 的详细描述，可以不填

3. Footer

Footer主要是针对两种情况，不兼容变动（以 `BREAKING CHANGE` 开头）和关闭issue（`Closes #234`）

在我们实际项目中用不到，可以不用考虑

4. Revert

有一种特殊的情况，如果当前的commit用于撤销以前的commit,必须以Revert:开头，后面跟着被撤销commit 的Header， body部分的格式也是固定的，`This reverts commit hash`，hash是被撤销commit的hash值

```
revert: feat(pencil): add 'graphitewidth' option
```

```
This reverts commit 667ecc1654a317a13331b17617d973392f415f02.
```