**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>

其中，Header 是必需的，Body 和 Footer 可以省略。

不管是哪一个部分，任何一行都不得超过72个字符（或100个字符）。这是为了避免自动换行影响美观。

## 2.1 Header

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

**（1）type**

type用于说明 commit 的类别，只允许使用下面7个标识。

### feat：新功能（feature）

### fix：修补bug

### docs：文档（documentation）

### style： 格式（不影响代码运行的变动）

### refactor：重构（即不是新增功能，也不是修改bug的代码变动）

### test：增加测试

### chore：构建过程或辅助工具的变动

如果type为feat和fix，则该 commit 将肯定出现在 Change log 之中。其他情况（docs、chore、style、refactor、test）由你决定，要不要放入 Change log，建议是不要。

**（2）scope**

scope用于说明 commit 影响的范围，比如数据层、控制层、视图层等等，视项目不同而不同。

**（3）subject**

subject是 commit 目的的简短描述，不超过50个字符。

* 以动词开头，使用第一人称现在时，比如change，而不是changed或changes
* 第一个字母小写
* 结尾不加句号（.）

**2.2 Body**

Body 部分是对本次 commit 的详细描述，可以分成多行。下面是一个范例。

More detailed explanatory text, if necessary.  Wrap it to

about 72 characters or so. Further paragraphs come after blank lines.- Bullet points are okay, too- Use a hanging indent

有两个注意点。

（1）使用第一人称现在时，比如使用change而不是changed或changes。

（2）应该说明代码变动的动机，以及与以前行为的对比。

# issue

那么如何跟随着commit关闭一个issue呢? 在confirm merge的时候可以使用一下命令来关闭相关issue:

1. fixes #xxx
2. fixed #xxx
3. fix #xxx
4. closes #xxx
5. close #xxx
6. closed #xxx

git commit -m "feat: zhangyujin fix #3"