**git提交规范细则**

**2021-04-06 chenglongqiao**

**Commit message**

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。

**Header**

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

其中项目问题号指：若为解决或完成coding中的具体问题，则在开头加上如issue\_pool#123<空格> <type>:<subject>

**（1）type**

type用于说明 commit 的类别，只允许使用下面8个标识。

1. feat：新功能（feature）
2. fix：修补bug
3. docs：文档（documentation）
4. style： 格式（不影响代码运行的变动）
5. refactor：重构（即不是新增功能，也不是修改bug的代码变动）
6. test：增加测试
7. chore：构建过程或辅助工具的变动
8. del：删除某个文件

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）将不会放入到Change log中。

**（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。

* 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（.）

**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

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。

**Footer**

**（1）不兼容变动**

如果当前代码与上一个版本不兼容，则 Footer 部分以BREAKING CHANGE开头，后面是对变动的描述、以及变动理由和迁移方法。

BREAKING CHANGE: isolate scope bindings definition has changed.

To migrate the code follow the example below:

Before:

scope: {

myAttr: 'attribute',

}

After:

scope: {

myAttr: '@',

}

The removed `inject` wasn't generaly useful for directives so there should be no code using it.