Git commit规范

1、Commit message格式

如HTML文件，css文件也必须声明该文件使用什么编码，并且使用UTF-8。

<type>(<scope>): <subject>

// 空行

<body>

// 空行

<footer>

其中，Header 是必需的，Body 和 Footer 可以省略。

a） Header

****（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：文档更改（documentation）
* style： 代码格式更改（不影响代码运行的变动）
* refactor：已有功能重构
* perf： 性能优化
* 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。

****（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。

(all, location, module)

****（3）subject****

subject是 commit 本次改动的简短描述。

b )Body

Body 部分是对本次 commit 的详细描述，可以分成多行。

c ) Footer

Footer 部分只用于两种情况。

(1)不兼容变动

eg:

更名：  
Before:  
<ChildXXX>  
  
After:  
<ChildComponent>

(2)关闭 Issue

eg:

Closes #123, #245, #992

2、其他注意事项

# head: <type>(<scope>): <subject>  
# - type: feat(新功能), fix(bugger修复), docs(文档更改), style(代码格式更改), refactor(重构), perf(性能优化)， test(增加测试), chore(构建过程或辅助工具的变动)  
# - scope: 可以为空（例如，如果更改是全局的或难以分配给单个组件）  
# - subject: 本次改动的简短描述  
#  
# body: 主要内容概括  
# \* 为什么需要进行此更改？  
# \* 如何解决问题？  
# \* 有没有什么副作用？  
#  
# footer:  
# - 不兼容变动 or 关闭 Issue  
#

参考 <specification_gitCommit.md>