

Git 提交消息约定

AngularJS Git Commit Message Conventions

目标

- 允许通过脚本生成 CHANGELOG.md 文件；
- 允许通过 git bisect 忽略指定的提交（一些不重要的提交，如文档格式化，调整目录结构等）；
- 在查看提交历史的时候，可以更直观的了解提交的内容。

提交消息的格式

- 1 <类型>(<范围>)：<主题>
- 2 <空行>
- 3 <详细内容>
- 4 <空行>
- 5 <尾部内容>

消息每行文字的长度不能超过 **30 个中文或 100 个字母**！这将更有利于在其他 git 工具中阅读提交的消息内容。

一个提交消息将由头部主题、详细内容和尾部消息组成，中间用空行分隔。

撤销

如果需要撤销之前的提交，头部主题应该以 撤销： 开头，然后紧跟着撤销的那条提交的头部主题。在详细内容部分应该说明： 撤销提交: <哈希值>，哈希值是指被撤销的那条提交的 SHA 哈希值。

头部主题

头部主题只能有一行内容，它包含了本次改变的简单描述，其内容包括**类型**、可选的**范围**和**主题**。

允许的类型

此类型说明了提交的改变属于哪种操作。

- 特性（新功能，新业务等）
- 修复（漏洞修复，编译错误等）
- 文档（源码注释，文档化，项目说明等）
- 风格（格式化源码，补全逗号，格式对齐等）
- 重构（重构特性，业务优化等）
- 测试（添加单元测试，测试用例等）
- 杂事（维护仓库，修改 gitignore 忽略文件等）

允许的范围

范围可以是用于说明提交改变的位置的任何内容。如 定位 、 浏览器 、 编译 等。

如果实在没有合适的范围，你可以使用 `*`，或干脆不写。

主题文本

这是一个关于改变的非常简短的描述。

- 使用命令式、现在时：如应该使用**改变**，而不是**改变了**，或**已改变**这类时态；
- 不要在末尾添加任何符号，如 `, !`，`。!` 这类符号；
- 英文开头的首字母大写。

详细内容

- 使用命令式、现在时：如应该使用**改变**，而不是**改变了**，或**已改变**这类时态；
- 包括提交改变的动机，和与之前行为进行的对比。

尾部消息

示例

```
1 feat($browser): onUrlChange event (popstate/hashchange/polling)
2
3 Added new event to $browser:
4 - forward popstate event if available
5 - forward hashchange event if popstate not available
6 - do polling when neither popstate nor hashchange available
7
8 Breaks $browser.onHashChange, which was removed (use onUrlChange instead)
```

```
1 fix($compile): couple of unit tests for IE9
2
3 Older IEs serialize html uppercased, but IE9 does not...
4 Would be better to expect case insensitive, unfortunately jasmine does
5 not allow to user regexps for throw expectations.
6
7 Closes #392
8 Breaks foo.bar api, foo.baz should be used instead
```

```
1 feat(directive): ng:disabled, ng:checked, ng:multiple, ng:readonly, ng:selected
2
3 New directives for proper binding these attributes in older browsers (IE).
4 Added coresponding description, live examples and e2e tests.
5
6 Closes #351
```

```
1 style($location): add couple of missing semi colons
```

```
1 docs(guide): updated fixed docs from Google Docs
2
3 Couple of typos fixed:
4 - indentation
5 - batchLogbatchLog -> batchLog
6 - start periodic checking
7 - missing brace
```

```
1 feat($compile): simplify isolate scope bindings
2
3 Changed the isolate scope binding options to:
4 - @attr - attribute binding (including interpolation)
5 - =model - by-directional model binding
6 - &expr - expression execution binding
7
8 This change simplifies the terminology as well as
9 number of choices available to the developer. It
10 also supports local name aliasing from the parent.
11
12 BREAKING CHANGE: isolate scope bindings definition has changed and
13 the inject option for the directive controller injection was removed.
14
15 To migrate the code follow the example below:
16
17 Before:
18
19 scope: {
20   myAttr: 'attribute',
21   myBind: 'bind',
22   myExpression: 'expression',
23   myEval: 'evaluate',
24   myAccessor: 'accessor'
25 }
26
27 After:
28
29 scope: {
30   myAttr: '@',
31   myBind: '@',
32   myExpression: '&',
33   // myEval - usually not useful, but in cases where the expression
34   // is assignable, you can use '='
35   myAccessor: '=' // in directive's template change myAccessor() to myAccessor
36 }
37
38 The removed `inject` wasn't generally useful for directives
39 so there should be no code using it.
```

