

# Git 合并多个提交的正确方式

 小揪  
Slow is fast

35 人赞同了该文章

## 1. 应用场景

对于一个项目，你可能会多次提交代码，每次提交都对应一个 `commit sha`

```
3320dec (HEAD) commit 4
03c9685 commit 3
041c401 commit 2
981fffd commit 1
22f759b (tag: base) initial commit
```

知乎 @小揪

当完成项目，要进行分支合并的时候，只想保留一个或某几个commit，这时候就需要合并commit了。

## 2. 如何合并

这里介绍两种方式，第一种是 `git rebase`，第二种是 `git rebase --autosquash`，后者在 `git commit` 时是有条件的。

### 1) git rebase

第一步，开启交互模式

```
git rebase -i <commit_sha>
```

注意，这里的 `<commit_sha>` 是你针对此项目第一个提交的前一个提交的commit。

举个例子：

下面是工作中的提交记录：

```
871adf OK, feature Z is fully implemented      --- newer commit
0c3317 Whoops, not yet...
87871a I'm ready!
643d0e Code cleanup
afb581 Fix this and that
4e9baa Cool implementation
d94e78 Prepare the workbench for feature Z
6394dc Feature Y                                --- older commit
```

知乎 @小揪

这时我们想要把前七个commit合并成一个，即：



其中 `commit_sha` 是第一个提交的前一个提交的哈希。因此，在我的示例中，命令为：

```
git rebase -i 6394dc
```

第二步，摘取合并

这时候会弹出一个框，列出了你想要合并的所有commit。注意列出的顺序是从老到新的。

```
pick d94e78 Prepare the workbench for feature Z      --- older commit
pick 4e9baa Cool implementation
pick afb581 Fix this and that
pick 643d0e Code cleanup
pick 87871a I'm ready!
pick 0c3317 Whoops, not yet...
pick 871adf OK, feature Z is fully implemented      --- newer commit

[...]
```

知乎 @小揪

更改commit\_sha最前面的单词，我们打算把这七个合并成一个commit，那么更改如下：

```
pick d94e78 Prepare the workbench for feature Z      --- older commit
s 4e9baa Cool implementation
s afb581 Fix this and that
s 643d0e Code cleanup
s 87871a I'm ready!
s 0c3317 Whoops, not yet...
s 871adf OK, feature Z is fully implemented          --- newer commit

[...]
```

知乎 @小揪

保存退出后，又弹出一个新的框，让我们更改commit信息，编辑完后退出就好了。最后完成的效果如下：

```
84d1f8 Feature Z                                --- newer commit (result of rebase)
6394dc Feature Y                                --- older commit
```

### 2) git rebase --autosquash

顾名思义，就是会自动帮你压缩commit。但是你在 `git commit` 的时候需要使用特殊命令：

```
git commit --fixup=<commit_sha>
```

这里的 `commit_sha` 是指你对哪个commit进行了更改，所以必须先至少存在一个对该项目的提交。

举个例子：

下面是工作中的提交记录：

```
d94e78 Prepare the workbench for feature Z
6394dc Feature Y                                --- older commit
```

我现在有有了新的改动，那么在提交时，就需要用如下命令表明我是对上一个commit进行了更改：

```
git commit --fixup=d94e78
```

产生的效果是这样的：

```
4e9baa fixup! Prepare the workbench for feature Z
d94e78 Prepare the workbench for feature Z
6394dc Feature Y                                --- older commit
```

这时候又有有了新的改动，那么提交时的命令不变：

```
git commit --fixup=d94e78
```

产生的效果：

```
afb581 fixup! fixup! Prepare the workbench for feature Z
4e9baa fixup! Prepare the workbench for feature Z
d94e78 Prepare the workbench for feature Z
6394dc Feature Y                                --- older commit
```

知乎 @小揪

接下来我们进行合并操作：

```
git rebase --autosquash -i <commit_sha>
```

注意，这里的 `commit_sha` 是older commit 6394dc。

Git这时会很机智的补全单词，不需要手动输入了。

```
pick d94e78 Prepare the workbench for feature Z
fixup 4e9baa fixup! Prepare the workbench for feature Z
fixup afb581 fixup! fixup! Prepare the workbench for feature Z
```

直接保存退出，重新编辑提交信息即可。

以上两种方式，一种是提交时轻松，合并时麻烦，另外一种相反。可以根据喜好来选择使用。

编辑于 2020-05-11 10:32