



开始写作

记录我的学习

请教专家

向专家提问

扫码查看

Git对于单个文件的分批提交方式的使用

很多时候，对于一个大的文件，可能有的同学改完之后不想一次提交，想分批提交。但这个时候由于git add的机制往往add之后就是整个一个文件被放到stage区了，这个时候肯定会想能不能对一个文件可以进行分开的快区域提交概念呢？

答案其实是有的，就是

```
git add -p [filename]
```

filename为可选操作，如果不指定，git就会对所有的unstage文件进行一次遍历。

那么如何使用呢？

比如我们一个文件a.txt的内容如下

```
hello
world
```

此时我将a.txt的内容改成了

```
hello
```

此时运行git diff可以看到如下

```
+
+
+  hello
- world
+
```

这个时候问题来了，我想先提交"把2222222增加"的提交，然后再提交"删除world，然后再"增加1111111新行"的提交怎么做呢？

首先进行命令

```
git add -p
```

它会提示你输入什么命令，有

```
State this hunk? [y,n,q,a,d,/,s,e,?]
```

具体的命令同学可以自己查，在这里我们输入'e'进入手动选择模式

出现了如下内容

```
# Manual hunk edit mode -- see bottom for a quick guide

@@ -, +, @@

+
+
+  hello
- world
+

# ---

# To remove '-' lines, make them ' ' lines (context).
# To remove '+' lines, delete them.
# Lines starting with # will be removed.
#

# If the patch applies cleanly, the edited hunk will immediately be
# marked for staging. If it does not apply cleanly, you will be given
# an opportunity to edit again. If all lines of the hunk are removed,
```

```
# then the edit is aborted and the hunk is left unchanged.
```

终于到了这一步，好了，英文好的同学可以看提示操作，不会的你只要知道我总结的两条原则就行

对于增加的内容（+），如果你在这个提交中不想放进去，就把整行删除
对于减去的内容（-），如果你在这个提交中不想删除，就把前面的‘-’删除，替换为一个空格

那么，按照我们之前的逻辑就应该改成这样

```
@@ -, +, @@  
  
hello  
  
world  
  
+
```

这个时候wq,此时你输入

```
git diff --cached
```

就会和stage区的文件进行diff对比，你可以看到只是简单的22222被加进去了，

这个时候你commit，就完成了仅仅针对增加22222222的补丁形式的提交。

那么接下来就很简单了，按照之前的逻辑进入到git add -p中的e手动模式，变成如下格式即可完成这次的所有需求

原来的是这样

```
@@ -, +, @@  
  
+  
  
hello  
  
-world
```

改成这样

```
@@ -, +, @@  
  
hello  
  
-world
```

那么，这次的提交就没有包括1111111的增加，只是简单的对world进行删除，最后commit即完成了world的删除，但是此时的1111111还没有加进去。

最后一个提交“11111111111加进去”其实已经很简单了，也不需要再用到git add -p了，只需简单的add和commit，即分别完成了对一个文件所做的三次提交。

参考链接：<http://stackoverflow.com/questions/10731269/git-hunk-edit-mode-how-to-remove-a-line>

请赞赏

朋友，创作不易；为犒赏小编的辛勤劳动，请她喝杯咖啡吧！

给她赞赏，您将财运亨通

支付宝



微信



¥11¥13¥15¥20¥30

赞助商

感谢朋友，[赞赏列表](#)

04-20 19:20

[查看更多](#)