git stash

`git stash` commit
commit

`git reset --hard` `git commit` “
”

```
# 
$ git stash
Saved working directory and index state WIP on workbranch: 56cd5d4 Revert "update old files"
HEAD is now at 56cd5d4 Revert "update old files"

# 
$ git stash list
stash@{0}: WIP on workbranch: 56cd5d4 Revert "update old files"
stash@{1}: WIP on project1: 1dd87ea commit "fix typos and grammar"

# 
$ git stash apply stash@{1}

# stash
$ git stash pop

# stash
$ git stash drop

# stash
$ git stash clear
```

commit

```
$ git stash pop
```

git stash命令可以运行多次，保存多个未提交的修改。这些修改以 “先进后出 ”的stack结构保存。

git stash list命令查看内部保存的多次修改。

$ git stash list
stash@{0}: WIP on new-feature: 5cedccc Try something crazy
stash@{1}: WIP on new-feature: 9f44b34 Take a different direction
stash@{2}: WIP on new-feature: 5acd291 Begin new feature

上面命令假设曾经运行过 git stash命令三次。

git stash pop命令总是取出最近一次的修改，但是可以用 git stash apply指定取出某一次的修改。

$ git stash apply stash@{1}

上面命令不会自动删除取出的修改，需要手动删除。

$ git stash drop stash@{1}

git stash 子命令一览。

# 展示目前存在的stash
$ git stash show -p

# 切换回stash
$ git stash pop

# 清除stash
$ git stash clear

参考链接

Ryan Hodson, Quick Tip: Leveraging the Power of Git Stash

版本号 #2
由 壳壳虫 创建于 2017-07-11 21:10:39 UTC
由 壳壳虫 更新于 2017-07-11 21:10:39 UTC