

Coding Style

- [社区](#)
- [公开课](#)
- [注册](#)
- [登录](#)

-

# 闯过这 54 关，点亮你的 Git 技能树 (三)

Git · [hkliya](#) · 于 4 月前发布 · 最后由 [hkliya](#) 于 2 天前回复 · 529 次阅读



这是一个系列文章，介绍了学习 Git 的好工具 - github，如果你是第一次看到，请先阅读：

[闯过这 54 关，点亮你的 Git 技能树](#)

[闯过这 54 关，点亮你的 Git 技能树 \(一\)](#)

[闯过这 54 关，点亮你的 Git 技能树 \(二\)](#)

今天我将带大家完成第 21 - 30 关。如对任何命令有疑问请看第一篇里的推荐教程。

## 第二十一关

```
> ~/D/git_hug on master . github reset reset
*****
*                               Github                               *
*****
resetting level

Name: reset
Level: 21
Difficulty: **

There are two files to be committed. The goal was to add each file as a separate commit, however both were a
stage the file `to_commit_second.rb` using the reset command (don't commit anything).
```

本来只想提交一个文件，结果意外地把另一个文件也 add 了，怎么把它搞出来？  
git reset 可以用来改变 HEAD 的位置或把文件从 staging area 移除出来，但并不会丢失任何的修改（我在刚开始接触 git 的时候觉得这个命令很危险，总感觉会把变更丢掉）。

```

x> ~/D/git_hug on master x git reset to_commit_second.rb
x> ~/D/git_hug on master x gst
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   to_commit_first.rb

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    to_commit_second.rb

x> ~/D/git_hug on master x githug
*****
*                                     Github                                     *
*****
Congratulations, you have solved the level!

Name: reset_soft
Level: 22
Difficulty: **

You committed too soon. Now you want to undo the last commit, while keeping the index.

```

## 第二十二关

提交太快，多提交了一个文件怎么办？

`git reset` 可以帮我们把当前的 `HEAD` 重置到指定的位置，这里是倒数第二个，所以是：`HEAD~1`。

但我们希望变更还保持在 `staging area`，不然还得 `add` 一次挺麻烦，那可以加上 `--soft` 参数。

```

--soft
  Does not touch the index file or the working tree at all (but resets the head to <commit>, just like
  all modes do). This leaves all your changed files "Changes to be committed", as git status would put it.

```

```
«> ~/D/git_hug on master ◦ git reset HEAD~1 --soft
«> ~/D/git_hug on master ◦ gst
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

    new file:   newfile.rb

«> ~/D/git_hug on master ◦ githug
*****
*                                     *
*                               Github                               *
*                                     *
*****
Congratulations, you have solved the level!

Name: checkout_file
Level: 23
Difficulty: ***

A file has been modified, but you don't want to keep the modification.  Checkout the `config.rb` file f
```

## 第二十三关

有时候改了一些代码，结果发现这个需求取消了，怎么能快速扔掉所有的变更，回到没有修改之前的状态呢？

```
«> ~/D/git_hug on master ◦ gst
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

    modified:   config.rb

no changes added to commit (use "git add" and/or "git commit -a")
«> ~/D/git_hug on master ◦ git checkout config.rb
«> ~/D/git_hug on master ◦ gst
On branch master
nothing to commit, working directory clean
«> ~/D/git_hug on master ◦ githug
*****
*                                     *
*                               Github                               *
*                                     *
*****
Congratulations, you have solved the level!

Name: remote
Level: 24
Difficulty: **

This project has a remote repository.  Identify it.
```

## 第二十四关

为了与别人合作，我们需要把 Git 仓库 push 到远端仓库上去，远端仓库对应就是 remote。

```
✕> ~/D/git_hug on master ◦ git remote
my_remote_repo
✕> ~/D/git_hug on master ◦ githug
*****
*                               Github                               *
*****
What is the name of the remote repository? my_remote_repo
Congratulations, you have solved the level!

Name: remote_url
Level: 25
Difficulty: **

The remote repositories have a url associated to them. Please enter the url of remote_location.
```

## 第二十五关

远端仓库除了有名称，还得有一个 URL 地址。

```
✎> ~/D/git_hug on master ◦ git remote -v
my_remote_repo https://github.com/Gazler/githubg (fetch)
my_remote_repo https://github.com/Gazler/githubg (push)
remote_location https://github.com/githubg/not_a_repo (fetch)
remote_location https://github.com/githubg/not_a_repo (push)
✎> ~/D/git_hug on master ◦ githubg
*****
*                               Github                               *
*****
What is the url of the remote repository? https://github.com/githubg/not_a_repo
Congratulations, you have solved the level!

Name: pull
Level: 26
Difficulty: **

You need to pull changes from your origin repository.
```

## 第二十六关

别人提交了代码，我们就要拉到本地来。

```
✎> ~/D/git_hug on master ◦ git pull origin master
remote: Counting objects: 3, done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 3
Unpacking objects: 100% (3/3), done.
From https://github.com/pull-this/thing-to-pull
 * branch          master      -> FETCH_HEAD
 * [new branch]     master      -> origin/master
✎> ~/D/git_hug on master ◦ githubg
*****
*                               Github                               *
*****
Congratulations, you have solved the level!

Name: remote_add
Level: 27
Difficulty: **

Add a remote repository called `origin` with the url https://github.com/githubg/githubg
```

## 第二十七关

```
> ~/D/git_hug on master ◦ git remote add origin https://github.com/githubug/githubug
> ~/D/git_hug on master ◦ githubug
*****
*                               Github                               *
*****
Congratulations, you have solved the level!

Name: push
Level: 28
Difficulty: ***

Your local master branch has diverged from the remote origin/master branch.
Rebase your commit onto origin/master.

remote: Counting objects: 2, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 2 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (2/2), done.
From /var/folders/x4/7l9lwjm95nv66jcfmqcvfvx80000gn/T/d20160104-32940-nnm51r/
* [new branch]      master    -> origin/master
```

## 第二十八关

```
✎> ~/D/git_hug on master ↵ git rebase
First, rewinding head to replay your work on top of it...
Applying: Third commit
✎> ~/D/git_hug on master ↑ git push
Counting objects: 2, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 231 bytes | 0 bytes/s, done.
Total 2 (delta 1), reused 0 (delta 0)
To /var/folders/x4/7l9lwjm95nv66jcfmqcvfvx80000gn/T/d20160104-33076-1iprs3h/.git
cc96274..c5199ba master -> master
✎> ~/D/git_hug on master ◦ githug
*****
*                                     Githug                               *
*****
Congratulations, you have solved the level!

Name: diff
Level: 29
Difficulty: **

There have been modifications to the `app.rb` file since your last commit. Find out which line h
```

## 第二十九关

```
»> ~/D/git_hug on master x git diff
diff --git a/app.rb b/app.rb
index 4f703ca..3bfa839 100644
--- a/app.rb
+++ b/app.rb
@@ -23,7 +23,7 @@ get '/yet_another' do
   erb :success
 end
 get '/another_page' do
-  @message = get_response('data.json')
+  @message = get_response('server.json')
   erb :another
 end

»> ~/D/git_hug on master x githug
*****
*                                     Github                                     *
*****
What is the number of the line which has changed? 26
Congratulations, you have solved the level!

Name: blame
Level: 30
Difficulty: **

Someone has put a password inside the file `config.rb` find out who it was.
```

## 第三十关

居然有人把明文密码写到了代码里，赶紧拉出来枪毙十分钟，怎么快速找到某行代码最后的修改者呢？



```
x> ~/D/git_hug on master x git blame config.rb
^5e8863d (Gary Rennie      2012-03-08 23:05:24 +0000 1) class Config
70d00535 (Bruce Banner    2012-03-08 23:07:41 +0000 2)   attr_accessor :name, :password
97bdd0cc (Spider Man      2012-03-08 23:08:15 +0000 3)   def initialize(name, password = nil,
^5e8863d (Gary Rennie      2012-03-08 23:05:24 +0000 4)       @name = name
97bdd0cc (Spider Man      2012-03-08 23:08:15 +0000 5)       @password = password || "i<3evil"
00000000 (Not Committed Yet 2016-01-04 23:51:42 +0800 6)
09409480 (Spider Man      2012-03-08 23:06:18 +0000 7)       if options[:downcase]
09409480 (Spider Man      2012-03-08 23:06:18 +0000 8)         @name.downcase!
09409480 (Spider Man      2012-03-08 23:06:18 +0000 9)       end
70d00535 (Bruce Banner    2012-03-08 23:07:41 +0000 10)
ffd39c2d (Gary Rennie     2012-03-08 23:08:58 +0000 11)       if options[:upcase]
ffd39c2d (Gary Rennie     2012-03-08 23:08:58 +0000 12)         @name.upcase!
ffd39c2d (Gary Rennie     2012-03-08 23:08:58 +0000 13)       end
ffd39c2d (Gary Rennie     2012-03-08 23:08:58 +0000 14)
^5e8863d (Gary Rennie     2012-03-08 23:05:24 +0000 15)   end
^5e8863d (Gary Rennie     2012-03-08 23:05:24 +0000 16) end
```

今天就到这里了，明天再见！

如果想第一时间得到更新，请关注 [CodingStyle.cn](http://CodingStyle.cn)！

---

[「软件匠艺社区」](#)旨在传播匠艺精神，通过分享好的「工作方式」和「习惯」以帮助程序员更加快乐高效地编程。

共收到 9 条回复



[lfeng](#) · #1 · 4 月前

git rm 和 git reset 有什么区别



[hkliya](#) · #2 · 4 月前

[#1楼 @lfeng](#) rm 是删除一个文件，reset 是从 commit 中移出或 staging area 中移出。



[hw93](#) · #3 · 2 月前

第29关 从git diff的信息怎么看出是26行的呢



[hkliya](#) · #4 · 2 月前