



如何利用 Git 与 GitHub 进行多人协作开发



Little Jerry (/u/7133a0105292) [+ 关注](#)

 1.3 2017.09.19 20:39* 字数 1554 阅读 8896 评论 7 喜欢 16

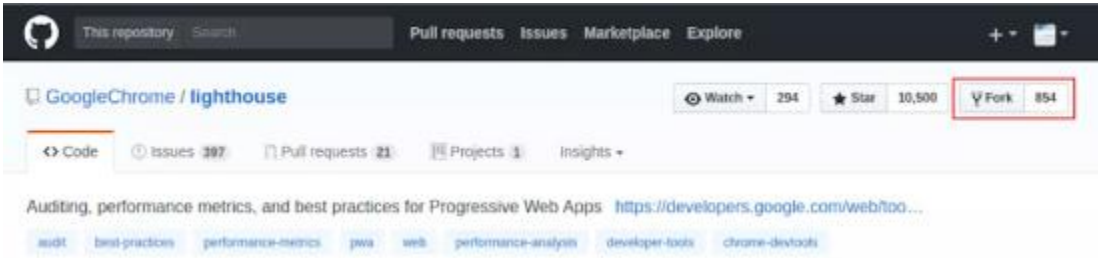
(/u/7133a0105292)

1. 首先 “Fork” 他人的 repository (代码仓库)

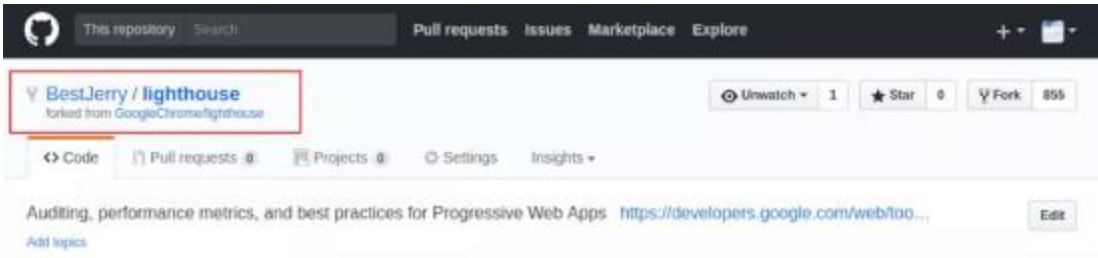
“Fork” 的名词是 “叉子” 的意思，不过这里取的是动词 “分叉，建一个分支” 的意思。进入你想参与合作开发的仓库界面，点击右上角的 “Fork” 图标，此时你已经复制了一个副本在你的 GitHub 仓库中了，或者说一个新的代码仓库被创建了，可以打开你的 GitHub 主页看一看。

注意，这个远程仓库是属于你自己的。这里 “Fork” 不同于 “Clone”，“Clone” 是发生在你的本地机器中，相当于你复制了一个完全相同的副本在你的终端上，但该副本的远程仓库连接的仍然是原作者的仓库，所以你并不是这个项目仓库的拥有者，没有更新它的权限。

因此，“Fork” 正是我们所需要的。



点击之后可以看到，自己帐号内有一个新的代码仓库被创建了。



然后将这个代码仓库 “Clone” (克隆) 到你的本地机器中，可以使用命令行或者 IDE (比如 IntelliJ IDEA) 的 VCS 功能来实现。“Clone” 成功之后你就可以自主修改里面的内容，然后 “Push” 到远程仓库中，注意，这是你自己的远程仓库。但是不使用 “Fork”，而是直接 “Clone” (克隆) 原作者的仓库的话，你会得到一个 “fatal: unable to access” 的提示，无法访问。

在这一环节，所有的修改只发生在你的远程仓库中，原作者的代码仓库内容是不会发生任何改变的。这里最需要理解的是，“Clone”下来的本地仓库连接的是你 GitHub 上的远程仓库。

2. 开始参与开发前的一些准备

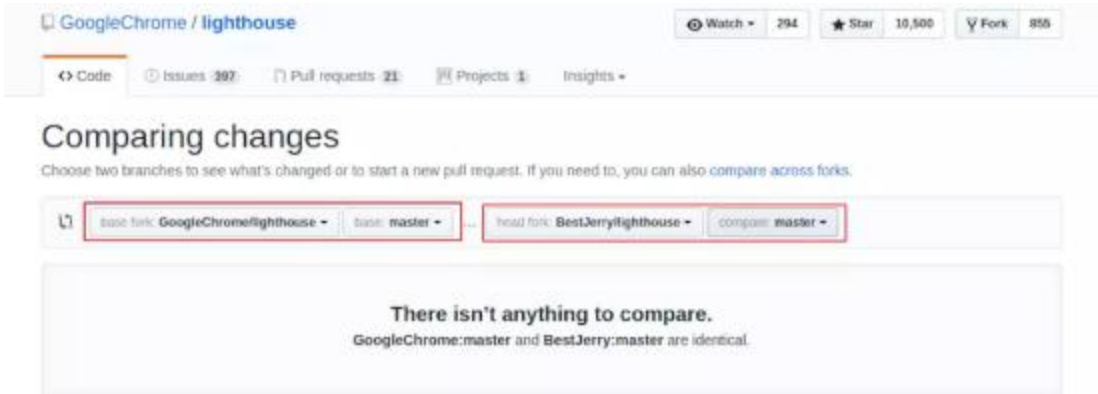
- (1) 在你做任何开发前，最好先详细阅读该项目的 CONTRIBUTING.md 文件。
- (2) 浏览该项目的 Issues（问题）公告，甚至可以自己创建一个 Issue。
- (3) 一般不要将自己做的修改提交到 master branch（主分支）上，而是应该提交到某个明确的 topic branch（主题分支，解决某个bug或者添加某一功能的分支）上。注意，我们应该自己建一个 topic branch，然后在上面作修改，而不在 master 分支直接修改，因为这样更具有直观性。
- (4) 最好每次只提交较小的修改，并写好清晰明确的 Commit Messages（提交说明）。
- (5) 如果有需要，请更新 README 文件。

3. 创建一个 Pull Request

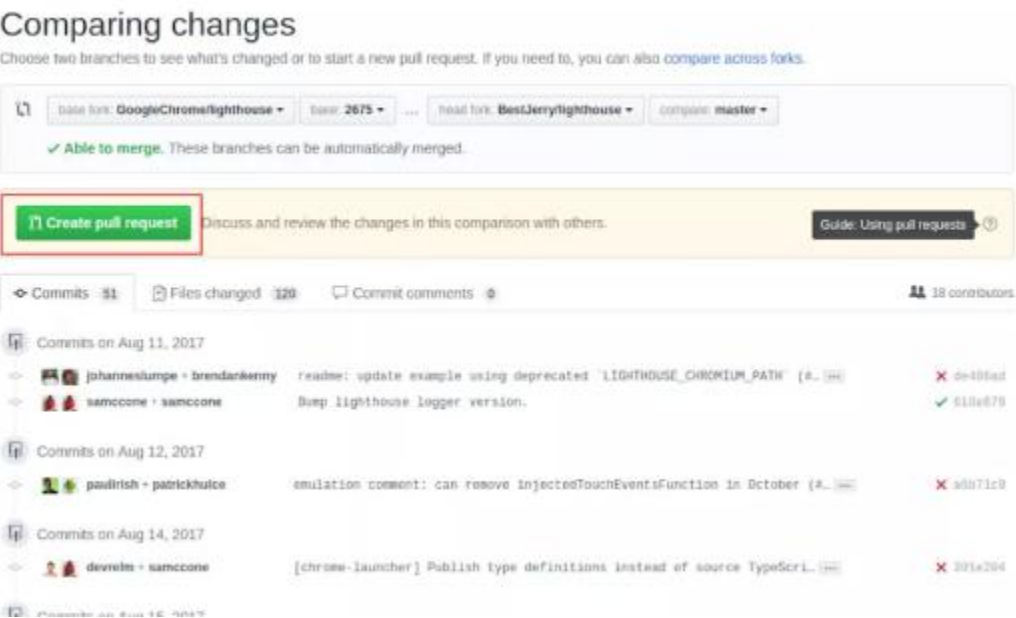
按自己的需要去修改项目内容，然后将所做的修改提交（Add+Commit）到自己的代码仓库，接着到仓库页面，点击 New pull request 按钮。



点击去之后可以看到，左边红框是你“Fork”的原项目仓库和分支，右边红框是你当前选择的项目仓库和分支。其实这里就是将两个红框内的内容作对比，并把不相同的地方显示出来。因为此时两个仓库和分支的内容是完全一样的，所有没有什么信息显示出来（我没有做修改）。不太理解的话，可以自由去尝试四个下拉框里不同的选项，很快你就会知道是怎么回事了。



我这里任意选择了原项目的另一个分支，因为这两个分支不完全相同，所以可以看到出现了一个“Create pull request”的按钮，点击它。

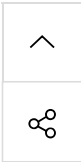


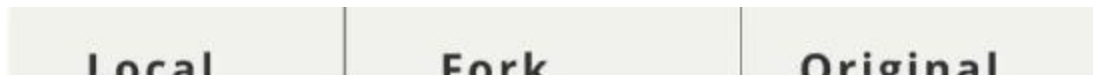
(/apps/redi
utm_sourc
banner-clic

填写你的请求信息，说明你做了些什么之类的。这个请求是发给该项目的维护者（拥有者）的，完成之后点击“Create pull request”按钮即可。



之后项目的维护者会受到你的请求，只要他/她通过了你的请求，你所做的修改就会被整合到原项目的仓库里了。





左边是你的本地仓库，中间是你在 GitHub 上 Fork 建立的仓库，右边是项目的原仓库。

(/apps/redi
utm_sourc
banner-clic

4. 与原仓库保持同步更新

说到合作开发就会有一个问题，如何与他人的代码保持同步？在自己做开发的过程中，难免会遇到你“Fork”的项目已经有了新的更新，这时当然是希望自己仓库中的代码也能同步进行更新。可是，你本地仓库所连接的远程仓库的是你自己的 GitHub 仓库，而不是原作者的仓库。**解决方法其实很简单，为你的本地仓库再添加一个远程仓库源。**

查看当前项目所连接的远程仓库

打开终端，进入到项目的 Git 仓库所在目录，一般就是项目目录，输入：

`git remote -v`

```
Terminal
+ jerry@jerry-ThinkPad-E550:~/lighthouse$ git remote -v
x origin https://github.com/BestJerry/lighthouse.git (fetch)
origin https://github.com/BestJerry/lighthouse.git (push)
```

可以到目前只连接了我自己的远程仓库。

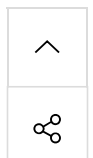
添加原作者的远程仓库连接

`git remote add upstream https://github.com/GoogleChrome/lighthouse.git`

(注意替换原仓库的 http 链接)

```
jerry@jerry-ThinkPad-E550:~/lighthouse$ git remote add upstream https://github.com/GoogleChrome/lighthouse.git
jerry@jerry-ThinkPad-E550:~/lighthouse$ git remote -v
origin https://github.com/BestJerry/lighthouse.git (fetch)
origin https://github.com/BestJerry/lighthouse.git (push)
upstream https://github.com/GoogleChrome/lighthouse.git (fetch)
upstream https://github.com/GoogleChrome/lighthouse.git (push)
```

如图，现在已经成功添加原项目的远程仓库了。



还差一点就大功告成了

(1) 从原仓库获取最新版本到本地

`git fetch upstream master`

(2) 保证当前位于 master 分支上

`git checkout master`

(3) 将最新版本整合到本地 master 分支上

`git merge upstream/master`

(4) 将更新发送到自己的 GitHub 仓库里

`git push origin master`

(/apps/redi
utm_sourc
banner-clic

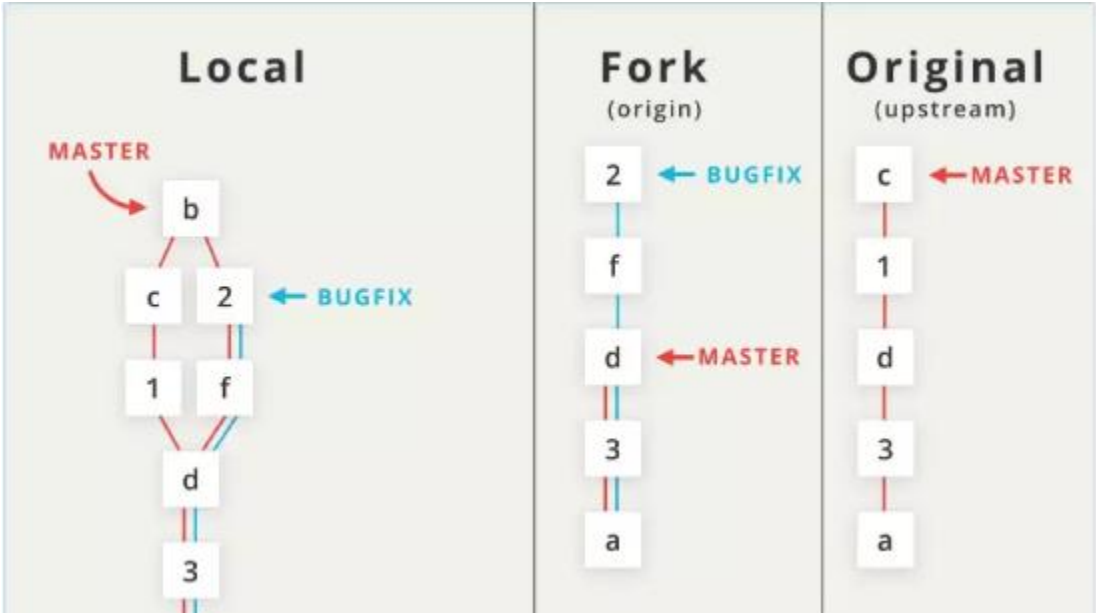
(1) (2) (3) 步可以用

`git pull upstream master`

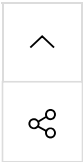
这条命令替代，可以这样不太安全，因为你 **fetch (获取)** 之后可以通过

`gitlog--oneline --graph --decorate --all`

来查看更新的情况，再决定是否 **merge (整合)** 到一起。



小礼物走一走，来简书关注我



赞赏支持

(/apps/redi
utm_sourc
banner-clic

 Git/GitHub (/nb/20738488)

举报文章

© 著作权归作者所有



Little Jerry (/u/7133a0105292) ♂

写了 428019 字，被 83 人关注，获得了 554 个喜欢
(/u/7133a0105292)

+ 关注

喜欢 | 16



更多分享



(/apps/redirect?utm_source=note-bottom-click)




登录后发表评论 (/sign-in?utm_source=desktop&utm_medium=not-signed-in-comr

7条评论

只看作者


按时间倒序 按时间正序




MaLizhi (/u/2f82e0cb8e36)
6楼 · 2019.05.07 23:35
(/u/2f82e0cb8e36)
看了这篇文章，懂了原来不懂的一些知识点

赞 回复


Little Jerry (/u/7133a0105292)：大家一同学习进步~
2019.05.13 15:35 回复

 添加新评论

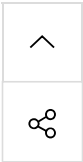


strugglexiang (/u/c63cbdb82e1c)
5楼 · 2019.02.28 22:41
(/u/c63cbdb82e1c)
路过学习

赞 回复



xeonH (/u/e028a20524c7)
4楼 · 2018.09.28 11:34
(/u/e028a20524c7)



写得很好


赞 回复

Little Jerry (/u/7133a0105292)：谢谢 ~
2018.09.30 11:38 回复

添加新评论

(/apps/redi
utm_sourc
banner-clic


 xeonH (/u/e028a20524c7)
3楼 · 2018.09.28 11:33
(/u/e028a20524c7)
写得很好
赞 回复

 Fsquirrel (/u/f25e21344851)
2楼 · 2018.08.20 20:05
(/u/f25e21344851)
赞一个
赞 回复

(/p/561a4fec4a96?




utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendatio
Git学习 (/p/561a4fec4a96?utm_campaign=maleskine&utm_content=no...
今天第一天开始上班，没有任务，于是开始学习Git这一程序猿必须掌握之技能，希望今天的积累过后，对与Git或者GitHub不再是了解，要有更深入的认识。在网上找了很多视频，看了之后版本感觉都太老旧，到现...

 CoderTung (/u/24369473bae2?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendatio

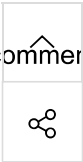
(/p/c0d4b9021aeb?




utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendatio
记正月十三游九古奇村 (/p/c0d4b9021aeb?utm_campaign=maleskine&ut...
金塘望月月无华，青头观龙龙有灵。迷洞萦回无去处，飞石无声鸟兽散。

 唐璜一 (/u/10defe2c8169?
utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendatio

在一朵云下 (/p/759c69b5b5a1?utm_campaign=maleskine&utm_content...
晴空如洗，坐一朵巨大的云。你静立草原，裙裾飞扬，孤单的影子瘦成芒草。及腰的黑精灵笑出声来，沙



哑，苍凉，温柔尽失。风吹草涌，你背影窈窕，如一川河，一道光，在一朵云下。我隔着思念，泪流满...

 司徒镜心 (/u/0afcab12569d?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendatio (/apps/redi

utm_sour

banner-clic


(/p/ce72e4db8d2e?



utm_campaign=maleskine&utm_content=note&utm_medium=seo_notes&utm_source=recommendatio

简书首页怎么啦？ (/p/ce72e4db8d2e?utm_campaign=maleskine&utm_c...


这几天，日更后照样首先投简书首页，意外发现每次都是秒过，简直是神转折，有点不相信自己的眼睛。忽
如一夜春风来，千树万树梨花开，我以为是我的码字水平忽然跃至一个新高峰，心中且喜且狐疑。只是发...

 无花也有果 (/u/83415485bf56?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendatio

运动旋风来了 (/p/eb6ffbd300f0?utm_campaign=maleskine&utm_conten...

2015年收官，并不算完美，各大品类绿意盎然。在这一片绿色之中，竟然还显现着一抹红，额外耀眼。飘红
的品类逆市飞扬，成了一道亮丽的风景线。在前几年它还表现中庸，既无亮点，也没潜力，一直平平淡淡...

 柳二白 (/u/914d3c7f83e8?)

utm_campaign=maleskine&utm_content=user&utm_medium=seo_notes&utm_source=recommendatio

