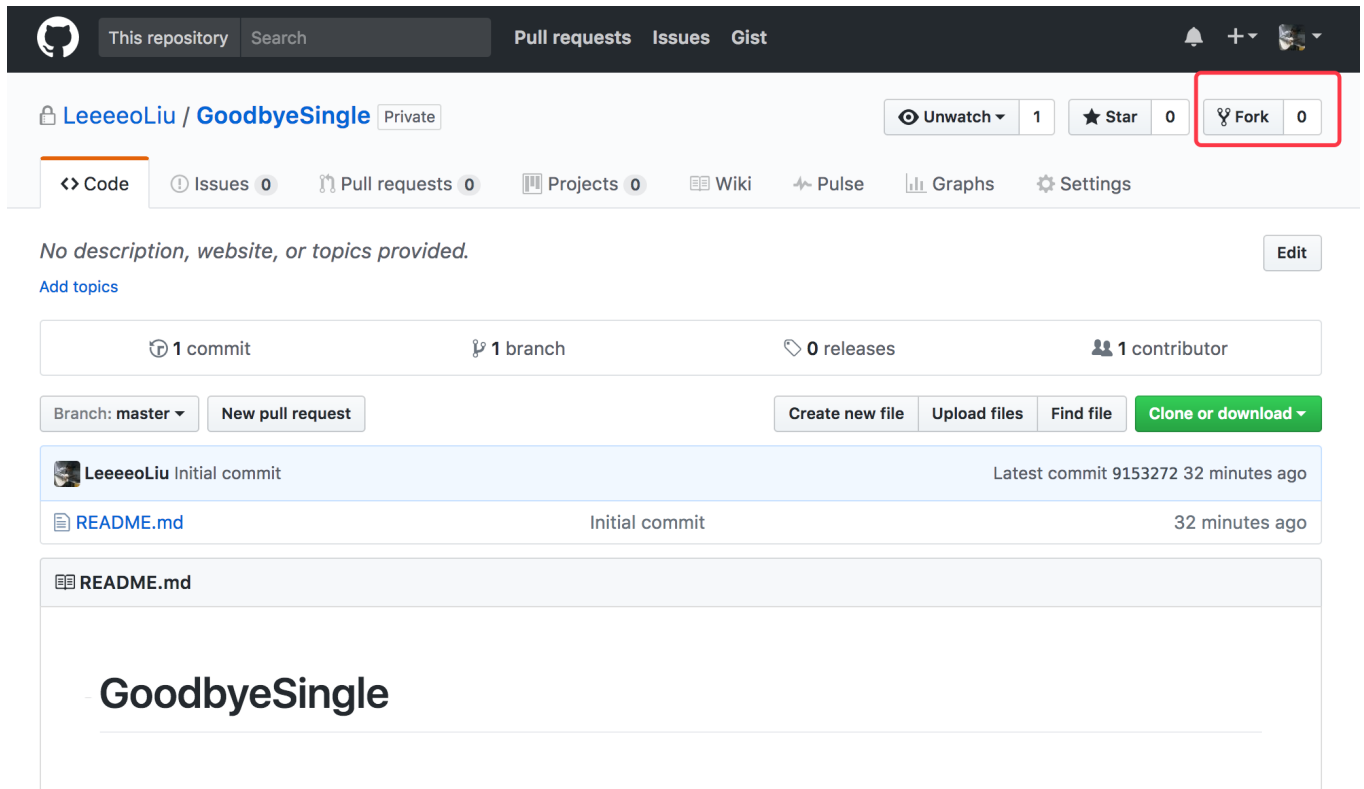


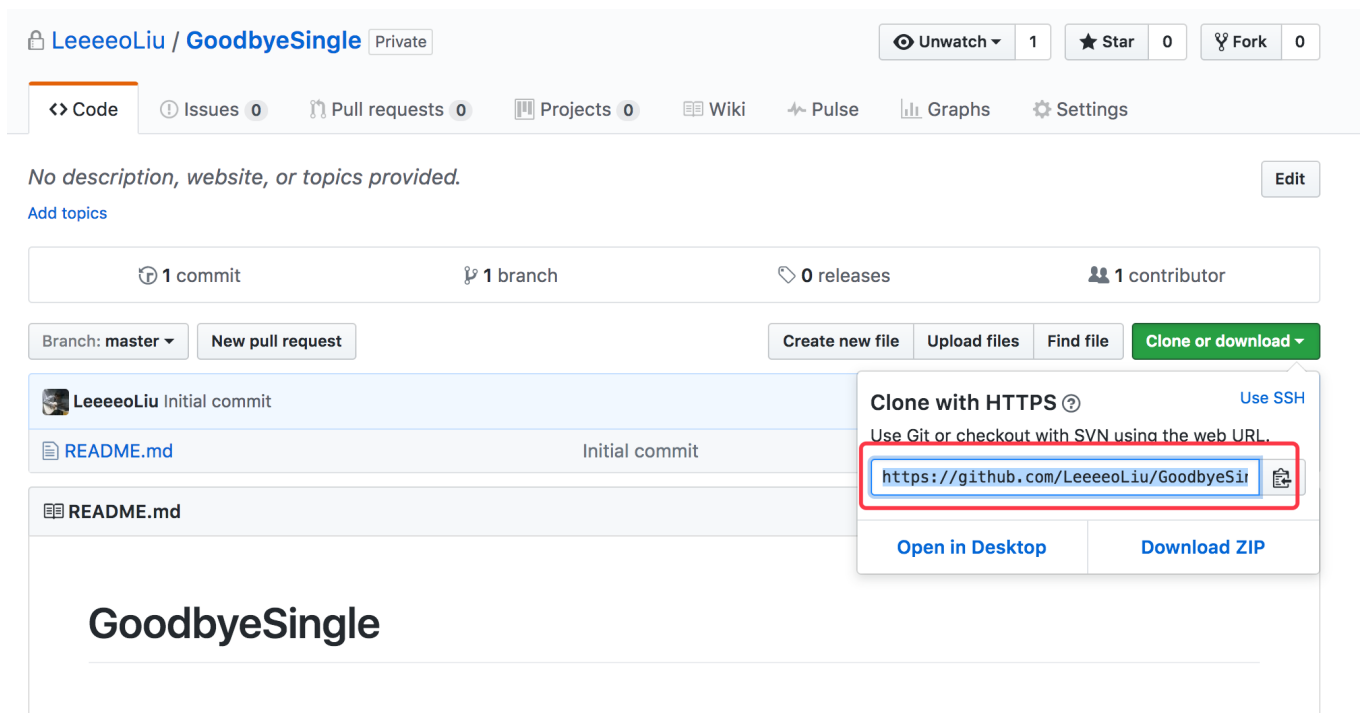
将Github项目部署到本地

1.fork到你的repository里



This screenshot shows the GitHub repository page for 'GoodbyeSingle' by user 'LeeeeeLiu'. The repository is private. At the top right, the 'Fork' button is highlighted with a red box, indicating the next step in the process. Below the repository name, there are tabs for 'Code', 'Issues', 'Pull requests', 'Projects', 'Wiki', 'Pulse', 'Graphs', and 'Settings'. The 'Code' tab is selected. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. A 'Clone or download' button is visible. The repository content shows a 'README.md' file with the text 'GoodbyeSingle'.

2.clone你fork的项目到本地



This screenshot shows the same GitHub repository page, but with the 'Clone or download' button clicked, opening a dialog box. The dialog box is titled 'Clone with HTTPS' and provides the URL 'https://github.com/LeeeeeLiu/GoodbyeSingle'. The URL is highlighted with a red box. There are also buttons for 'Open in Desktop' and 'Download ZIP'.

3.打开git，使用cd命令切换到你想要保存项目的目录下，执行 `git clone` 你刚刚复制的地址

```
# Leeeeo @ LeeeeoLius-MacBook-Pro in ~/Code/JavaWeb/GoodbyeSingle on git:master x [21:21:28] C:1
$ git clone https://github.com/LeeeeoLiu/GoodbyeSingle.git
```

4.设置remote，执行 `git remote add goodbye`

`https://github.com/LeeeeoLiu/GoodbyeSingle.git` 然后执行 `git remote -v` 应该可以看到你刚刚设置的

```
# Leeeeo @ LeeeeoLius-MacBook-Pro in ~/Code/JavaWeb/GoodbyeSingle on git:master x [21:25:56]
$ git remote add goodbye https://github.com/LeeeeoLiu/GoodbyeSingle.git

# Leeeeo @ LeeeeoLius-MacBook-Pro in ~/Code/JavaWeb/GoodbyeSingle on git:master x [21:27:21]
$ git remote -v
goodbye https://github.com/LeeeeoLiu/GoodbyeSingle.git (fetch)
goodbye https://github.com/LeeeeoLiu/GoodbyeSingle.git (push)
```

基本的就完成了，下面说一下更新项目的步骤(这个很重要，一定要理解，不然你可能会重复将项目部署到本地)

1.执行 `git fetch goodbye master` 这个是把主版本的代码拿下来

```
# Leeeeo @ LeeeeoLius-MacBook-Pro in ~/Code/JavaWeb/GoodbyeSingle on git:master x [21:28:28] C:1
$ git fetch goodbye master
From https://github.com/LeeeeoLiu/GoodbyeSingle
 * branch                master          -> FETCH_HEAD
 * [new branch]           master          -> goodbye/master
```

2.执行 `git rebase FETCH_HEAD` 这个是把刚刚拿下来的代码和你本地的代码合并，如果没有冲突的话，应该是下图这种，类似你代码版本更新了的消息，如果有冲突，有点麻烦，直接找我吧

```
# Leeeeo @ LeeeeoLius-MacBook-Pro in ~/Co
$ git rebase FETCH_HEAD
Current branch master is up to date.
```

3.最后执行 `git push` 然后就行了，总的来说来说就这么三步，但是，往往第二步会出问题，所以请在敲代码之前以及我通知主版本库更新之后更新你的本地代码库

下面说一说一种比较常见的情况，如果你正在修改某处代码，我通知代码更新了

1.不要惊慌失措，执行 `git stash` ,这条指令会保存你没有提交的修改，就好比把你做的修改压栈，

2.当你更新完之后，执行 `git stash pop` 可以把你刚刚压的修改，弹出来。然后你就可以继续修改了。

当然，在你更新代码的时候也会出现冲突，但是这时候处理起来就相对简单了。现在没案例，遇到了再说。

最后一步，你改的代码都没问题了，准备提交

1.执行 `git add 文件` 保存你的修改

2.执行 `git commit -m "提交信息"` 提交

团队开发中用到的一些 Git 基本操作（请自行百度）

`git log`

查看版本日志

git status

查看当前分支状态

git add

添加当前某个文件的修改

git commit -m “message”

提交

git push (仓库 分支)

推到自己的远程仓库

git fetch XXX 分支

拉取一个人的一个分支

git rebase FETCH_HEAD

应用 FETCH_HEAD

git reset —hard HEAD

版本回退

发起 Pull Request

git remote -v

显示远端仓库

git remote add XXX Url

添加远端仓库