

## 1、把 git 仓库中的文件推送到 GitHub 中的完整流程

(1) 先把本地要上传的所有文件夹及文件都添加到 git 仓库中

首先进入到放置所有要上传的文件及文件夹的目录下，单击右键选择“git bash”进入 git 命令行，然后初始化 git 仓库、添加到暂存区、提交到 git 仓库中，如下图所示：

```
gg@lxx MINGW64 /d/GitTool/gitCSS3
$ git init
Initialized empty Git repository in D:/GitTool/gitCSS3/.git/

gg@lxx MINGW64 /d/GitTool/gitCSS3 (master)
$ git add .

gg@lxx MINGW64 /d/GitTool/gitCSS3 (master)
$ git commit -m 'CSS3新增功能'
[master (root-commit) 8163f04] CSS3新增功能
 2 files changed, 1 insertion(+)
 create mode 100644 "CSS3\346\226\260\345\242\236\347\232\204\345\212\237\350\275\357\274\210\344\270\200\357\274\211.pdf"
 create mode 100644 readme.md
```

(2) 再把本地仓库 master 分支上的所有文件都推送到 GitHub 远程仓库 origin 对应的 master 分支中

先远程添加默认仓库 origin 的 URL 地址，然后将本地 master 分支中的文件推送到远程 origin 仓库中，如下图所示：

```
gg@lxx MINGW64 /d/GitTool/gitCSS3 (master)
$ git remote add origin https://github.com/lixiuxiu2017/CSS3.git

gg@lxx MINGW64 /d/GitTool/gitCSS3 (master)
$ git push origin master
Counting objects: 4, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 703.55 KiB | 8.18 MiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/lixiuxiu2017/CSS3.git
 * [new branch]      master -> master
```

### Summary:

(1) 从本地推送分支的方法是使用 **git push origin branch-name**，推送时 Git 会自动把本地分支和远程的同名分支对应起来，并且远程仓库的默认名称是 origin

(2) 多人协作在各自 dev 分支上工作时，如果使用 **git push origin dev** 推送分支失败，是因为你的推送和其他人的最新提交有冲突，解决办法就是先用 **git pull** 把最新的提交从 origin/dev 抓取下来，然后在本地合并，解决冲突再推送。如果 **git pull** 提示错误“no tracking information”，则说明本地分支和远程分支的连接关系没有创建，可以使用 **git branch --set-upstream dev origin/dev** 建立关联

(3) 并不是一定要把本地分支推送到 GitHub 中，本地分支包括 master 分支、dev 分支、bug 分支等等，其中：

master 分支是主分支，用于直接发布产品，要时刻与远程同步；  
dev 分支是日常开发分支，团队成员都需要在上面工作，所以也需要与远程同步；  
bug 分支只用于在本地修复 bug，就没必要同步到远程。

(4) 克隆代码可以使用 **git clone GitHub 的 URL**，新克隆下来的代码会默认本地的 master 分支和远程的 origin/master 分支关联，而 dev 分支和远程没有任何关联，要在本地构建和远程分支对应的分支，可以使用 **git checkout -b dev origin/dev**

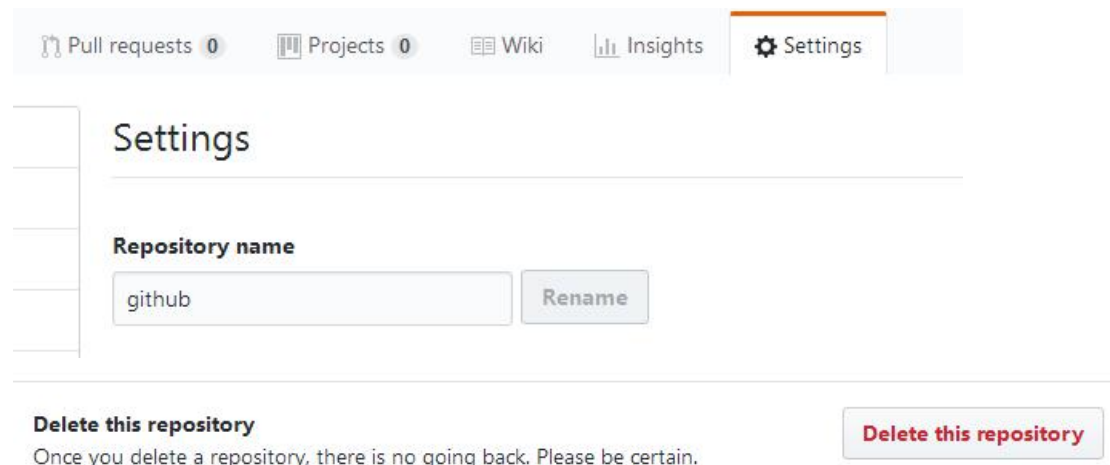
(5) 要查看远程仓库的信息，可以使用 **git remote** 命令，如果要显示更详细的信息，可以用 **git remote -v**

(6) 要查看所有分支，可以使用 **git branch --all**

(7) 如要要从 master 分支切换到 dev 分支进行开发，可以使用 **git checkout dev**

## 2、修改及删除 GitHub 远程仓库的方法

在 GitHub 页面中进入要修改的仓库，在页面上方找到“settings”，即可重命名以及删除已有仓库，如下图：



## 3、推送分支过程中出现错误提示 remote origin already exists 的解决办法

只要在本地删除远程仓库 origin，然后再重新 git remote add origin 即可，如下图：

```
gg@1xx MINGW64 /d/GitTool/gitCSS3 (master)
$ git remote add origin https://github.com/lixiuxiu2017/HTML5.git
fatal: remote origin already exists.

gg@1xx MINGW64 /d/GitTool/gitCSS3 (master)
$ git remote rm origin
```