# 采用ssh-api进行交互

## 1、有库没有分支：

git init：初始化本地文件夹

git add：添加需要上传的文件

git commit -m “”：提交，后面跟上备注

git branch -m main：给分支命名

git remote add origin github@....：在github特定的库创建分支

git push -u origin main：上传文件到github中创建的分支

## 2、换文件夹进行操作同一个库时：

git clone github@.....：将想要操作的库的分支连接起来。如果clone到一个文件夹时，可以取出含有.git文件的全部，其他的都可以删（可以用https也可以用SSH）。

后面进行git add、git commit -m “”、git push origin main。

## 3、删除：

git rm 文件名：删除本地和远程以及缓存区

git rm --cached 文件名：删除远程和缓存区

git rm -f 文件名：如果删除之前修改过并且已经放到暂存区域的话则必须要用强制删除-f。

再进行：git commit -m “”、git push origin main

## 4、复原

git reset 文件名 HEAD~n：将文件回到n次提交前

git checkout 文件名：确认是这个文件

此时已经本地文件已经回到n次以前了。但现在本地和远程文件版本不同步，需要把复原的文件拷贝到其他地方防止被破坏，然后先：

git pull origin main：拉去远程到本地，并融合本地，一般会失败，此时打开要上传的文件，其中《《《《本地内容====远程内容》》，《《、==、》》是标记的冲突的符号，删除其中一个并删除标记符号，再进行下面的内容。

git add 刚才复原的文件名：向上提交

git commit -m “冲突已解决”：提交

git push origin main：推到远程，此时可能会失败。

再次尝试一边顺序。如果还是不行，需要给git push -f origin mian强制上推（同一起的开发者沟通好）

## 5、注意事项

所进行的所有操作都需要建立在对的分支上才可以。

git switch 分支：可以用于调整现在的指向分支，是HEAD还是branch。

## **6、删除仓库**

本地 git branch -d 分支名，如果删除被拒绝可以用git branch -D 分支名。删除远程仓库：git remote prune origin。

## 7、操作

一般不要一个文件夹操控多个仓库，不易操作，并且如果上一个库和下一个库中任意一个库存在与之相对应远程库文件版本存在不一致时，就会出现报错，所以不要这么用。

假设buff为本地仓库名，那么远程仓库为remote origin/a。除此以外，对于同一仓库的不同分支，可以用同一个文件夹绑定。但是在上传文件时，注意分支的选择。

## 8、删除操作

对于特殊情况的其他进程终端可以使用rm -f .git/index.lock。对于先删除一个文档，仅仅在远端和缓存区删除，需要用git rm -r –cached 文件名：-r为递归选项。删除本地和远程以及缓存区都需要用-r(递归操作)。