# 相关命令

Ssh-keygen -t rsa -b 4096(4096为指定生成大小)

git clone把远程仓库内容复制本地

进入远程仓库提交文件git commit -m“”双引号内容为注释讲解提交内容

Git ls-files 查看仓库状态

Git push 把本地修改提交给远程仓库

Git pull 把远程仓库修改拉到本地

Git remote add 添加一个远程仓库

Origin为远程仓库别名

Git remote -v 查看当前仓库对应的远程仓库别名和地址

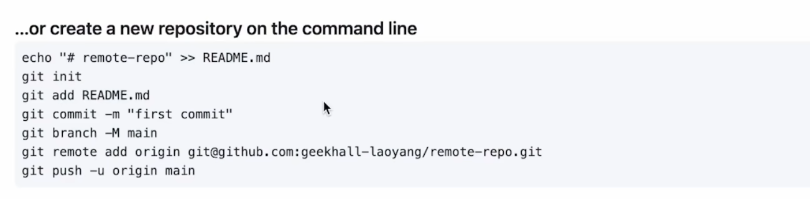
Git branch -M main 指定分支名称为main

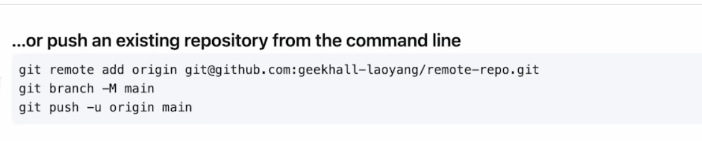
Git push -u origin main 把本地的main 分支和远程仓库的main 分支关联

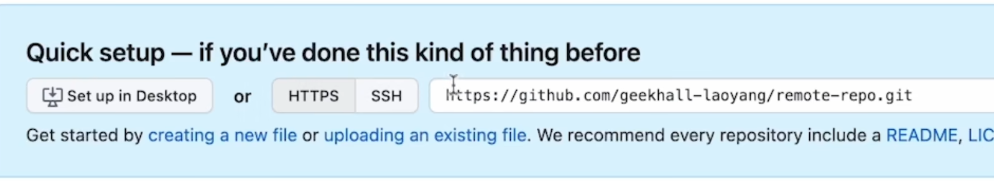
Pull命令：git pull <远程仓库名> <远程分支名>:<本地分支名>

# 本地仓库和远程仓库关联

1. 本地创建仓库







https和SSH区别为：前者把本地代码push到远程仓库的时候要用户名和密码

后者需要公钥配置（前者目前失效）

Ssh-keygen -t rsa -b 4096(4096为指定生成大小)

如果之前生成密钥遇到Enter file in which to save the key步骤的时候需要输入一个新的文件名

输入密码

生成的以.hub为结尾的是公钥文件，vi xxx.pub把公钥文件添加到github上

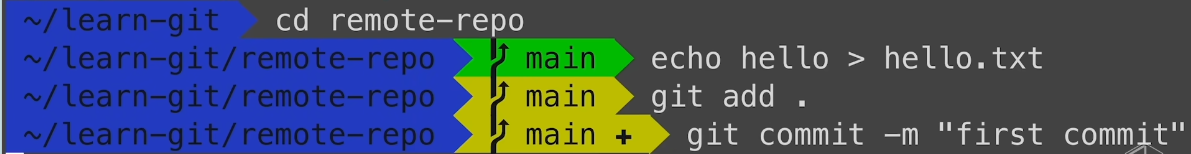
如果是新生成的文件名需要添加一个config文件把以下内容放入文件里



当访问github.com指定使用ssh下的test密钥

git clone

把远程仓库内容复制本地



进入远程仓库提交文件git commit -m“”双引号内容为注释讲解提交内容

Git ls-files 查看仓库状态

Git push 把本地修改提交给远程仓库

Git pull 把远程仓库修改拉到本地

# 把本地修改提交到远程仓库



Git remote add 添加一个远程仓库

Origin为远程仓库别名

Git remote -v 查看当前仓库对应的远程仓库别名和地址

Git branch -M main 指定分支名称为main

Git push -u origin main 把本地的main 分支和远程仓库的main 分支关联

Pull命令：git pull <远程仓库名> <远程分支名>:<本地分支名>

实际操作中直接用git pull（把远程仓库和本地仓库进行合并）就行，只要远程仓库和本地仓库没有冲突，git pull命令就会成功

Fetch命令：fetch命令只是获取远程仓库的修改，但是并不会合并到本地仓库中，需要手动合并