

git 远程仓库

<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/001374385852170d9c7adf13c30429b9660d0eb689dd43a000>

创建SSH Key

打开Git Bash，创建SSH Key：

```
$ ssh-keygen -t rsa -C "youremail@example.com"
```

成功后，可以在用户主目录里找到`.ssh`目录，里面有`id_rsa`和`id_rsa.pub`两个文件，这两个就是SSH Key的秘钥对，`id_rsa`是私钥，不能泄露出去，`id_rsa.pub`是公钥，可以放心地告诉任何人。

在GitHub中“Add SSH Key”，在Key文本框里粘贴`id_rsa.pub`文件的内容。

Git支持SSH协议，所以，GitHub只要知道了你的公钥，就可以确认只有你自己才能推送。

GitHub允许你添加多个Key。假定你有若干电脑，只要把每台电脑的Key都添加到GitHub，就可以在每台电脑上往GitHub推送了。

添加远程库

在GitHub上点击“Create a new repo”按钮，创建一个新的仓库。

把本地仓库的内容推送到远程库

```
$ git remote add origin git@github.com:totoroQ/learngit.git
```

//远程库的名字就是origin，这是Git默认的叫法

```
$ git push -u origin master //把本地库的所有内容（即当前分支master）推送到远程库
```

//由于远程库是空的，我们第一次推送master分支时，加上了-u参数，Git不但会把本地的master分支内容推送的远程新的master分支，还会把本地的master分支和远程的master分支关联起来，在以后的推送或者拉取时就可以简化命令。

Counting objects: 19, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (19/19), done.

Writing objects: 100% (19/19), 13.73 KiB, done.

Total 23 (delta 6), reused 0 (delta 0)

To git@github.com:michaelliao/learngit.git

* [new branch] master -> master

Branch master set up to track remote branch master from origin.

```
$ git push origin master //把本地master分支的最新修改推送至远程库
```

从远程库克隆

```
$ git clone git@github.com:totoroQ/gitskills.git
```

Cloning into 'gitskills'...

remote: Counting objects: 3, done.

remote: Total 3 (delta 0), reused 0 (delta 0)

Receiving objects: 100% (3/3), done.

Git支持多种协议，默认的git://使用ssh，但也可以使用https等其他协议。

使用https除了速度慢以外，还有个最大的麻烦是每次推送都必须输入口令，但是在某些只开放http端口的公司内部就无法使用ssh协议而只能用https。