

我们以github为例，介绍本地仓库如何与github上仓库联动。

github官网: <https://github.com/>

值得注意的是，国内对GitHub限速，除魔法上网之外的方法有时候会出现短暂掉线，连接不上GitHub的情况，这是正常的，多刷新几次，或者等一会就可以了。

(魔法上网大家就各显神通吧，没法说)

首先注册一个账户，略。

一、SSH配置

为了让本地仓库和github账号上的仓库之间建立连接，我们需要在本地配置ssh-key。

ssh简要来说是一种常见的安全连接协议。

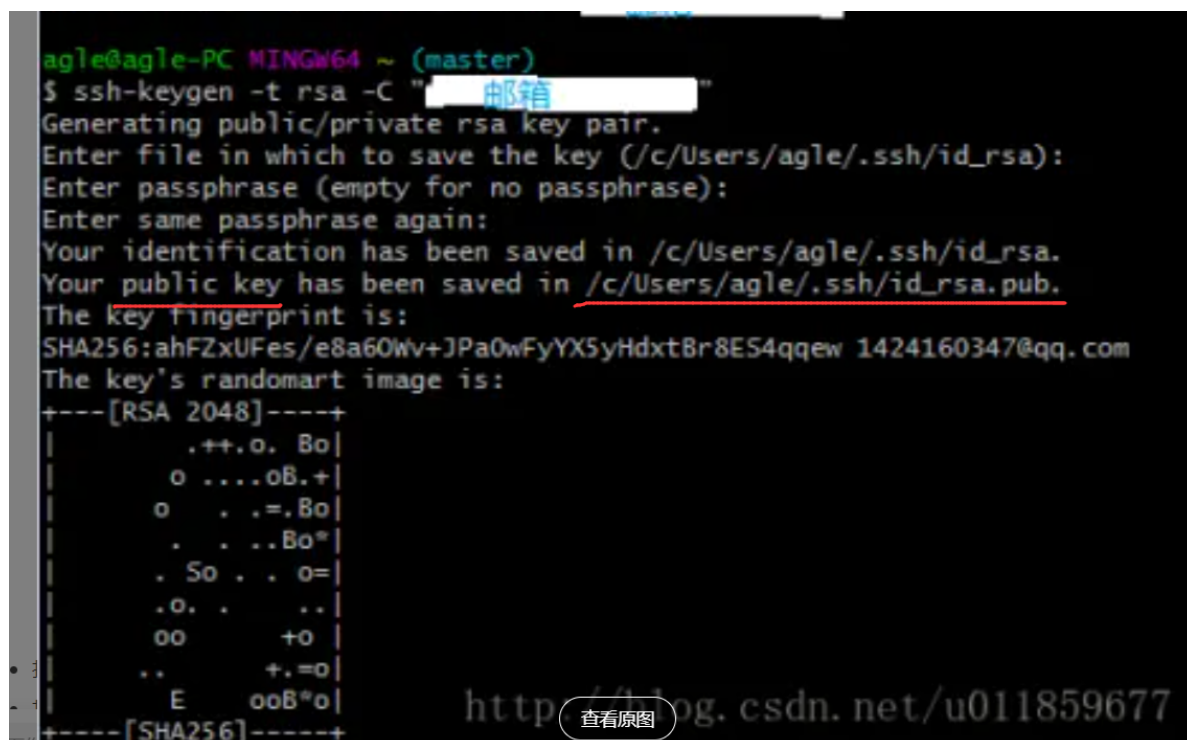
我们在任意位置右击 -> Open Git Bash here, 输入命令。

```
ssh-keygen -t rsa -C "你的账户邮箱（写全，包括后面@...com）"
```

(-t rsa 指的是采用RSA方式加密)

如果你了解ssh，请在下面的弹框中自行配置passphrase，否则不输入一路回车。

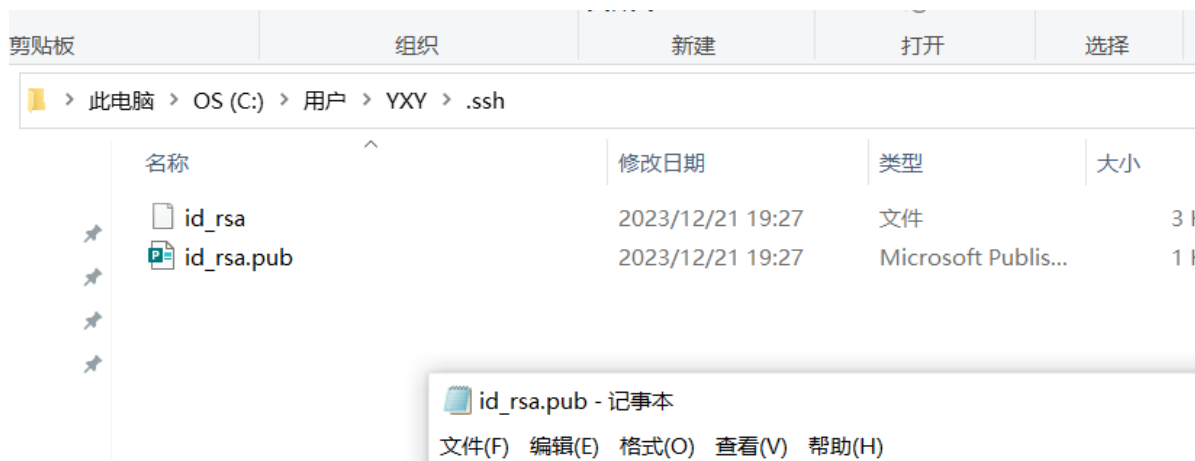
(我这边已经创建过了，这是截的一个网图)



```
agle@agle-PC MINGW64 ~ (master)
$ ssh-keygen -t rsa -C "[redacted]"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/agle/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/agle/.ssh/id_rsa.
Your public key has been saved in /c/Users/agle/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:ahFZxUFes/e8a60Wv+JPa0wFyYX5yHdxtBr8E54qqew 1424160347@qq.com
The key's randomart image is:
+---[RSA 2048]---+
|      .+++.o. Bo|
|      o ....oB.+|
|      o  .  .=.Bo|
|      .  .  ..Bo*|
|      . So . . o=|
|      .o. .    ..|
|      oo      +o |
|      ..      +.=o|
|      E      ooB*o|
+---[SHA256]-----+
http://blog.csdn.net/u011859677
```

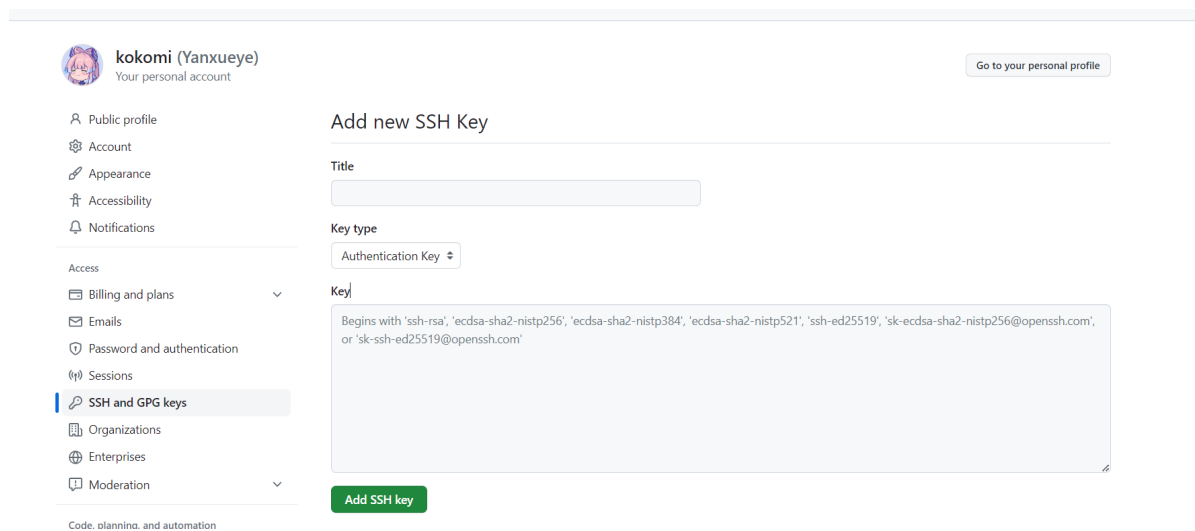
然后我们按照红线所示路径找到这个文件（记得打开显示隐藏文件，agle是系统本地用户名）

用记事本打开文件，复制里面全部内容



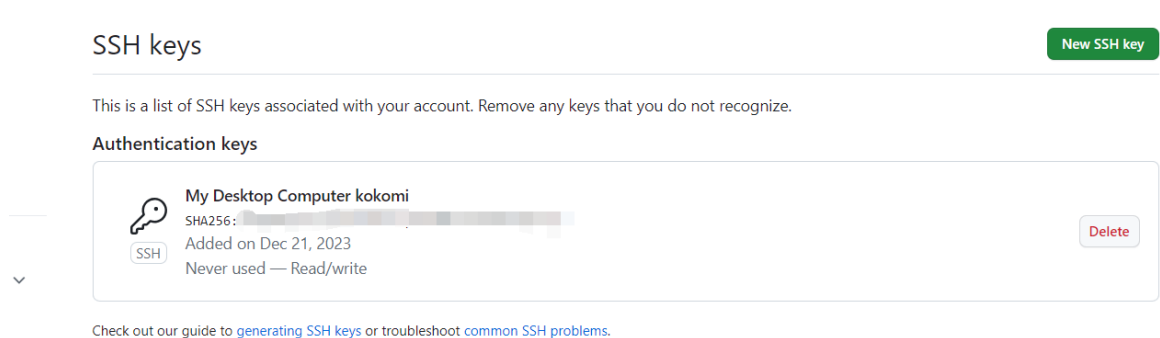
打开GitHub，点击头像，Settings -> SSH and GPG keys

点击右上角 New SSH key，然后进入这个界面。



Title随便写，Key框里粘贴复制内容。

完成后点击Add SSH key，然后输入密码验证一下。完成之后结果如下，然后你写的邮箱里应该会收到ssh-key有关的邮件。



然后再次打开git bash

输入 `ssh -T git@github.com`

随后系统会询问我们是否要继续连接，输入yes然后回车，然后github的信息就被写进hosts文件了，这意味着我们和GitHub账号绑定了。

```
YXY@DESKTOP-PSGQT5E MINGW64 /d/Git/onlineRepo/TestRepository (main)
$ ssh -T git@github.com
The authenticity of host 'github.com (20.205.243.166)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Hi Yanxueye! You've successfully authenticated, but GitHub does not provide shell access.
```

二、Github和本地仓库传输命令

1.将本地仓库上传

我们需要首先自己在GitHub上创建一个Repository（后面用Repo代替，这是大家默认的简写）

头像 -> Your repositories -> 右上角 New

Great repository names are short and memorable. Need inspiration? How about [improved-rotary-phone](#) ?

Description (optional)

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

Initialize this repository with:

☐ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: Java ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: GNU General Public License v3.0 ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

① You are creating a public repository in your personal account.

Create repository

库名和描述好说。

下面选择库的属性，公开或私有，字面意思。

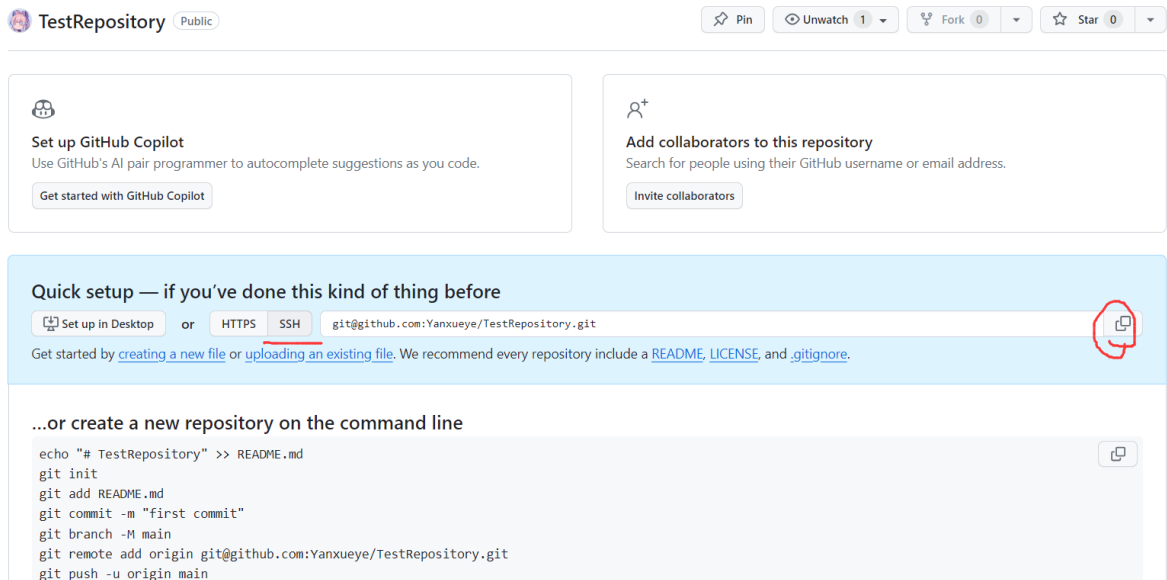
（以下选项可选可不选，如果选择GitHub会对应自动创建相应文件，但是为了在上传时不引起冲突，初学者最好都不要选）

README file是关于你的项目的介绍，大项目一般都有，是markdown文件，可以勾选也可以不勾选，勾选后GitHub会为你创建Readme.md。

.gitignore文件指的是在git工具处理过程中被忽略的文件，这里包含一些模板，我们选择Java模板就可以了，它会帮我们过滤掉*.class, *.log, *.jar等等运行文件或Java配置文件。

license是使用许可证，我们使用GNU General Public License v3.0就足够了，其他的详情自行百度。

什么都不选就会进入这个界面，简单来说GitHub需要我们进行初始化。



你可以按照下面的帮助（..or create a new repo on command line...及以下部分）来初始化。

但是我们现在想要把之前本地的仓库上传，我们走Quick setup的流程。

首先复制SSH地址，在上图红线和红圈位置。

在你本地写好的仓库位置打开Git Bash

输入以下命令

```
git remote add origin 复制的内容（右键Paste）
```

其中 `git remote` 命令是控制远程仓库的命令。

`add` 命令表示在当前本地库的位置添加远程库。

`origin` 是远程仓库别名，可以自己随便取，`origin`是公认默认叫法。

复制的内容是仓库的ssh地址

```
YXY@DESKTOP-PSGQT5E MINGW64 /d/Git/repo2 (master)
$ git remote add origin git@github.com:Yanxueye/TestRepository.git
```

输入完后没有任何反应表示添加成功。

接下来我们使用如下指令上传master分支的全部内容。

```
git push -u origin master
```

如果报错请尝试 `git push -u origin master -f` 进行强制推送

```
YXY@DESKTOP-PSGQT5E MINGW64 /d/Git/repo2 (master)
$ git push -u origin master
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 12 threads
Compressing objects: 100% (22/22), done.
Writing objects: 100% (26/26), 2.41 MiB | 1.22 MiB/s, done.
Total 26 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To github.com:Yanxueye/TestRepository.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

查看GitHub，刷新一下页面。

The screenshot shows the GitHub interface for a repository named 'TestRepository' by user 'Yanxueye'. The repository is public and has 1 branch and 0 tags. The current branch is 'master'. The file list shows: 'Hi' (INIT, 21 minutes ago), 'pdf' (INIT, 21 minutes ago), '.gitignore' (INIT, 21 minutes ago), 'LICENSE.txt' (license, 19 minutes ago), and 'README.md' (INIT, 21 minutes ago). The 'README' file is selected and its content is displayed below the list. On the right side, there are sections for 'About', 'Learning', and 'Releases'.

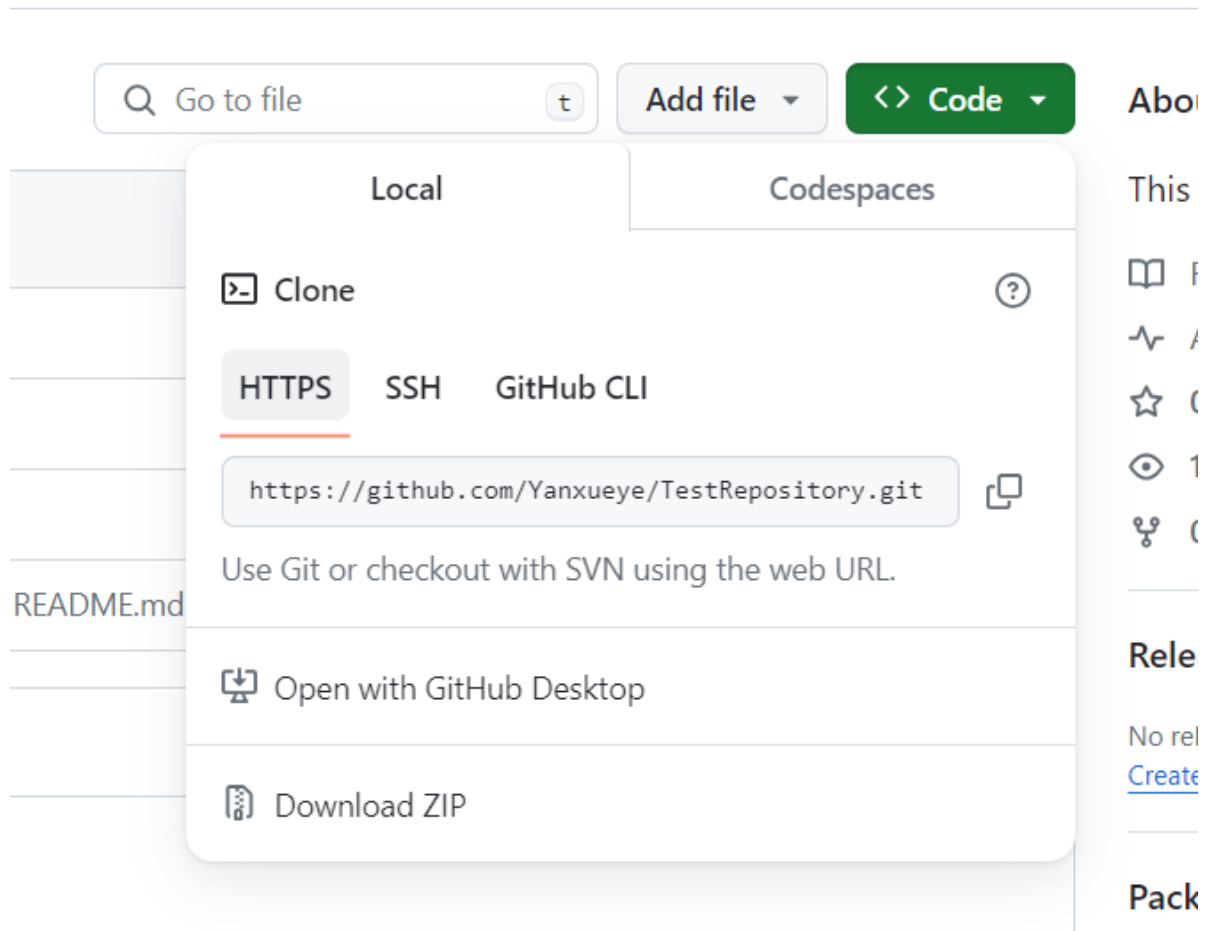
File	Commit	Time
Hi	INIT	21 minutes ago
pdf	INIT	21 minutes ago
.gitignore	INIT	21 minutes ago
LICENSE.txt	license	19 minutes ago
README.md	INIT	21 minutes ago

左上角表示当前为master分支，ok，内容更新完毕。

2.Copy别人的项目到本地

使用 git clone 命令。

点击项目右上角<>Code，切换到HTTPS，然后复制。



然后在本地要存储的位置右击打开Git Bash

输入 `git clone <鼠标右键Paste粘贴网址>`

```
XY@DESKTOP-PSGQT5E MINGW64 /d/Git
$ git clone https://github.com/Yanxueye/TestRepository.git
Cloning into 'TestRepository'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), 12.40 KiB | 470.00 KiB/s, done.
```

TestRepository 2023/12/21 21:4

然后这个项目就被下载下来了，非常简单。

我把pdf的原markdown文件放到这个项目里了，大家可以下载下来查看，也算是一个作业？

<https://github.com/Yanxueye/TestRepository.git>

GitHub本身也有很多有趣的操作，比如云复制fork，评论issues，提交修改申请Pull Requests等等，大家感兴趣可以自己搜索>_<，其实所有的操作GitHub官方也有Help文档，只不过都是英文hhh。

