

## Github 使用指南！（下文针对 linux 系统而言，特指 ubuntu 系统）

### 第一步：

下载安装Git。

1. 使用新立得软件包管理工具(Synaptic Package Manager)安装最新版本的 git。  
推荐选择安装 git-core, git-gui, git-doc。

### 第二步：设置SSH Keys

github 使用 ssh keys 来确保你的电脑与 github 的连接有安全性。设置过程很简单。

但是有几个步骤。

步骤一：检查已有的 ssh keys。

```
$ cd ~/.ssh
```

如果提示说，没有这样的文件或者目录。(No such file or directory) 则跳到步骤三。否则继续。

步骤二：备份已有的 ssh keys。

步骤三：产生一个新的 ssh key。

我的产生过程如下：

```
banxi1988@banxi:~$ ssh-keygen -t rsa -C "banxi1988@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/banxi1988/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Passphrases do not match. Try again.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/banxi1988/.ssh/id_rsa.
Your public key has been saved in /home/banxi1988/.ssh/id_rsa.pub.
The key fingerprint is:
0c:a5:4d:a7:d8:e8:42:b2:5c:75:d0:4c:e5:ff:6a:7f banxi1988@gmail.com
The key's randomart image is:
+--[ RSA 2048 ]-----+
|      o==.o      |
|     .Xo+       |
|    .o = + .    |
|   . = . o .   |
|  o . . S .    |
|   . . .       |
|    .. E|      |
|     .....     |
+-----+
banxi1988@banxi:~$
```

### 步骤四：

在 gitHub 里单击账户设置。点击 SSH 公钥，点击添加另一个公钥。

打开前面生成的 id\_rsa.pub 文件。（可能要显示隐藏文件才能看到用 gedit 或者其它的编辑器打开。  
注意不要编辑，添加空格或者换行。）

然后把里面的内容复制粘贴到下面的 key 输入栏中。

### 步骤五：测试一下。

```
banxi1988@banxi:~$ ssh git@github.com
/etc/ssh/ssh_config: line 20: Bad configuration option: X11Forwrding
/etc/ssh/ssh_config: terminating, 1 bad configuration options
banxi1988@banxi:~$
```

After modify the ssh\_config.中间要求输入上面输入的 passphrase。

```
banxi1988@banxi:~$ ssh git@github.com
The authenticity of host 'github.com (207.97.227.239)' can't be established.
RSA key fingerprint is 16:27:ac:a5:76:28:2d:36:63:1b:56:4d:eb:df:a6:48.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,207.97.227.239' (RSA) to the list of known hosts.
PTY allocation request failed on channel 0
Hi banxi1988! You've successfully authenticated, but GitHub does not provide shell access.
```

```
connection to github.com closed.  
banxi1988@banxi:~$
```

C

步骤六：设置 git 的个人信息。

1. 设置用户名和 email。git 使用这个来验证每个提交者。除此之外 github 会使用这些信息来与你的 github 账号相联系。下面的名字应该是你真实的名字。而不是要 GitHub 的用户名。

```
banxi1988@banxi:~$ git config --global user.name "Li HaiZhen"  
banxi1988@banxi:~$ git config --global user.email "banxi1988@gmail.com"  
banxi1988@banxi:~$
```

2. 设置 GitHub 的指令环。

有些工具不使用 ssh 来连接。为了使用这些工具。你应该找出配置好你的 API Token。  
在 GitHub 上点击。账号设置 (Account Settings)，然后点击 账号管理(Account Admin)。  
在 API Token 一栏里有你的 API Token。  
在命令行下运行下面的代码。

```
banxi1988@banxi:~$ git config --global github.user banxi1988  
banxi1988@banxi:~$ git config --global github.token e5ebe68d43d9820ed8d05a3d2633d7f3  
banxi1988@banxi:~$
```

上面使用的 user 是指 GitHub 的用户名了。

最后：恭喜你！你已经正确的设置了 git 和 gitHub。

接下来。要做的就是。1. 创建一个仓库。2. Fork 一个仓库。 3.社区化。

## 第二节:创建一个仓库(Create A Repo Repositories)

直接在自己的登录后进入 github.com 首页就可以看到, 下面一栏有第四步.用来创建 Repository.

直接填入项目名称就可以了.其它的可以不填.要填,这个表单也足够自解释了.

创建后之后.会跳转到一个页面.其中有指示接下来该怎么做的.

如下:

[git@github.com:banxi1988/tasteHibernate.git](https://github.com/banxi1988/tasteHibernate.git)

接下来给你自己的项目创建一个基本的 Readme 文件吧.

详细操作过程如下:

Global setup:

Download and install [Git](#)

```
git config --global user.name "banxi1988"  
git config --global user.email banxi1988@gmail.com
```

Next steps:

```
mkdir tasteHibernate  
cd tasteHibernate  
git init  
touch README  
git add README  
git commit -m 'first commit'  
git remote add origin git@github.com:banxi1988/tasteHibernate.git  
git push -u origin master
```

Existing Git Repo?

```
cd existing_git_repo  
git remote add origin git@github.com:banxi1988/tasteHibernate.git  
git push -u origin master
```

## Importing a Subversion Repo?

[Click here](#)

When you're done:

[Continue](#)

```
banxi1988@banxi:~/github/tasteHibernate$ git init
Initialized empty Git repository in /home/banxi1988/github/tasteHibernate/.git/
banxi1988@banxi:~/github/tasteHibernate$ touch README
banxi1988@banxi:~/github/tasteHibernate$ vi README
banxi1988@banxi:~/github/tasteHibernate$ git add README
banxi1988@banxi:~/github/tasteHibernate$ git commit -m 'first commit'
[master (root-commit) 6ec8aae] first commit
 1 files changed, 6 insertions(+), 0 deletions(-)
 create mode 100644 README
banxi1988@banxi:~/github/tasteHibernate$ git remote add origin git@github.com:banxi1988/tasteHibernate.git
banxi1988@banxi:~/github/tasteHibernate$ git push origin master
ERROR: banxi1988/tasteHibernate.git doesn't exist. Did you enter it correctly?
fatal: The remote end hung up unexpectedly
banxi1988@banxi:~/github/tasteHibernate$ git push -u origin master
Counting objects: 3, done.
Delta compression using up to 2 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 383 bytes, done.
Total 3 (delta 0), reused 0 (delta 0)
To git@github.com:banxi1988/tasteHibernate.git
 * [new branch]    master -> master
Branch master set up to track remote branch master from origin.
banxi1988@banxi:~/github/tasteHibernate$
```

关于 Git 的命令请参见 Git 手册.

现在我们已经可以创建了一个库了.创建了一个文件,并且提交了.并且把它推向了 github.  
接下来我们将做什么呢?

### 第三节: Fork A Repo

有些时候你发现自己想要为别人的项目做贡献.或者希望来使用别人的项目做为自己的起点.也就称之为 Fork.

1. Fork 一个项目. 在你想 fork 的项目的首页.找到 fork 按钮.点击.
2. 接下来设置你本地仓库.

A. 克隆项目.

```
$ git clone git@github.com:username/projectname.git
```

B. 远程配置.

当你克隆了一个项目之后.它有一个默认的 remote.叫做.origin.这是指你是在 github 上 fork 的.而不是在原来的仓库.为了跟踪原本的仓库,你需要添加另一个叫做 upstream 的选项.

```
$ cd projectname
$ git remote add upstream git://github.com/username/projectname.git
$ git fetch upstream
```

3. 接下来.你要做的就是.

A. 推送提交.

一旦你做出了某些提交到你 fork 的仓库里,你可能想要将其推送到你 fork 的项目去.你要做就跟平常的项目一样.

```
$ git push origin master
```

4. 接收 upstream 变更.

如果你 fork 的那个原来的仓库改变了,你可以使用下面的命令来更新你 fork 到本地的仓库.

```
$ git fetch upstream
$ git merge upstream/master
```

后面的更多使用指南请参考相关文档.例如创建分支等.

本指南是由 **banxi1988** 根据 **github** 提供的帮助.进行的实践之后,翻译过来的.  
由于本人水平有限.所以翻译得不好,你可以向我提意见或者直接到 **github** 里面的去看帮助.  
(;banxi-iyān ~\_~)