

[置顶] Github 使用教程(一)--搭建 Github 环境

Git 是一个分布式的版本控制系统，最初编写用作 Linux 内核代码的管理。在推出后，Git 在其它项目中也取得了很大成功。Github 目前已经有 100 多万的开发者，并且对于公共项目的托管免费。Github 使得版本控制更加容易，多人协作编码变得简单。使用过 SVN 的同学应该知道，当没有网络的支持，开发者并没有办法提交已完成代码。Github 支持离线提交，其实就是提交到本地仓库。每一个开发者 clone 下的项目代码都是一个完整的仓储，当远程或其他协作开发者代码无法恢复时，都可以方便使用本地的仓储进行恢复。当需要对代码进行分支时，其他版本控制软件在大量文件时会很慢，而 Github 几乎是瞬间完成这个操作，而且只需要一个命令即可。Github 的便利性和与其他类似软件的比较不是本文重点。下面我们一步一步来搭建 Github 的环境（以 windows 操作系统为例）。

首先，需要[注册](#)一个 Github 的账号。

然后，下载 [Github windows 客户端](#)。安装后得到下面两个快捷方式：



其中 GitHub(蓝色图标)为 windows 的图形界面操作，Git Shell 为命令行操作界面(本文以 Git Shell 为例搭建)。

1.需要告诉 Github，你要使用的用户名

```
[plain] view plaincopy C:\> 1. git config --global user.name "Your Name Here"
```

2.设置邮箱

```
[plain] view plaincopy C:\> 1. git config --global user.email "your_email@example.com"
```




3.在 github 上创建项目



打开[网址](#)，然后填写项目名称和其他信息，如图：

PUBLIC

Owner

 gavincook

Repository name


test


✓

Great repository names are short and memorable. Need inspiration? How about **freezing-octo-**

Description (optional)

the description for test


☒  **Public**
Anyone can see this repository. You choose who can commit.

☐  **Private**
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**
This will allow you to `git clone` the repository immediately.

Add .gitignore: **None**

Add a license: **None**



Create repository

点击 create repository 创建项目,跳转到如下页面：

Quick setup — if you've done this kind of thing before



Set up in Desktop or

HTTP

SSH

https://github.com/gavincook/test.git



We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

Create a new repository on the command line

```
touch README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/gavincook/test.git
git push -u origin master
```

Push an existing repository from the command line

```
git remote add origin https://github.com/gavincook/test.git
git push -u origin master
```

4. clone 项目到本地

打开 Git Shell,执行以下代码：

[plain] [view plaincopy](#)

1. `git clone https://github.com/gavincook/test.git`

这里的 `https://github.com/gavincook/test.git` 来自于上图的创建项目后的地址。

第一次使用时会提示输入用户名和密码，用于获取相关权限。

5.提交代码

步骤 4 后，我们会在我们的工作区得到一个 test 的文件夹，我们新建一个 test.txt 文件，内容是 Github test!

在 Git Shell 中使用 `git status`,我们可以看到相应的改动：



然后使用 `git add .` 将所有的改动加入缓存区

```
E:\workspace\github\test [master +1 ~0 -0 !]> git add .
E:\workspace\github\test [master +1 ~0 -0]>
```

然后使用 `git commit -m "备注信息"` 将改动提交到本地仓库

```
E:\workspace\github\test [master +1 ~0 -0]> git commit -m "提交测试"
[master (root-commit) aa6353c] 提交测试
 1 file changed, 1 insertion(+)
 create mode 100644 test.txt

Warning: Your console font probably doesn't support Unicode. If you experience s
trange characters in the output, consider switching to a TrueType font such as L
ucida Console!
E:\workspace\github\test [master]>
```

最后将代码提交到远程服务器

```
E:\workspace\github\test [master]> git push origin master
Counting objects: 3, done.
Writing objects: 100% (3/3), 233 bytes | 0 bytes/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/gavincook/test.git
 * [new branch]      master -> master
E:\workspace\github\test [master]>
```

登录到 Github 上，打开相应的 repository 的页面，可以看到刚刚提交的内容：

branch: master test / test.txt

gavincook 3 minutes ago 提交测试

1 contributor

file | 1 lines (1 sloc) | 0.012 kb

Edit Raw

1 Github test!

到此为止，就已经完成了 Github 的环境搭建和简单的一次代码提交!