**Github简单的配置和使用方法**

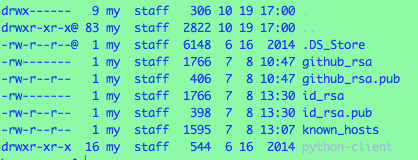
一、使用SSH密钥连接Github

如果不配置SSH密钥的话，每次push代码或改动的时候都要输入用户名和密码，如果使用SSH，就可以记住用户名，下面介绍如何创建SSH Keys并将公钥加到Github账号中，使用SSH Keys在本机和Github之间建立一个安全链接。

方法：

1. 首先，我们需要看看本机是否存在SSH Keys，打开Terminal，执行：ls -al ~/.ssh。

检查列表是否已经存在共有的SSH Key



如果存在，可以跳过第二步直接去第三步。

1. 创建一对新的SSH Keys

复制粘贴以下命令

$ssh-keygen -t rsa -C "your\_email@example.com"

#这将按照你提供的邮箱地址，创建一对密钥

Generating public/private rsa key pair.

Enter file in which to save the key (/Users/you/.ssh/id\_rsa): [Press enter]

直接回车，则将密钥按默认文件进行储存，此时也可以输入特定的文件名，比如/Users/you/.ssh/github\_rsa

接着根据提示输入密码和确认密码

Enter passphrase (empty for no passphrase): [Type a passphrase]

Enter same passphrase again: [Type passphrase again]

官方建议直接点击回车，不用添加密码，输入完成后，屏幕会显示如下信息：

Your identification has been saved in /Users/you/.ssh/id\_rsa.

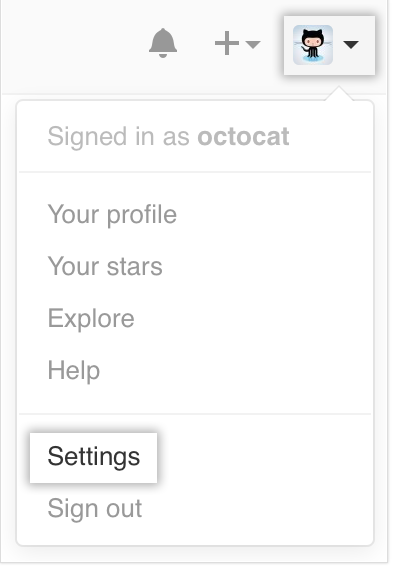
Your public key has been saved in /Users/you/.ssh/id\_rsa.pub.

The key fingerprint is:

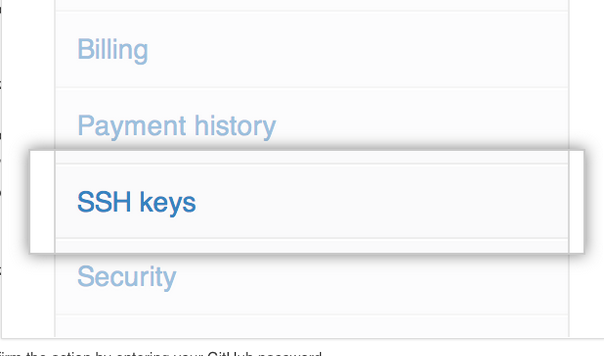
01:0f:f4:3b:ca:85:d6:17:a1:7d:f0:68:9d:f0:a2:db your\_email@example.com

然后在终端中输入：pbcopy < ~/.ssh/id\_rsa.pub可将公钥的内容复制到系统粘贴板

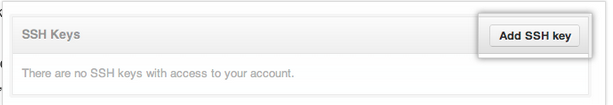
1. 在Github账户中添加你的公钥
   1. 登陆Github，进入Settings



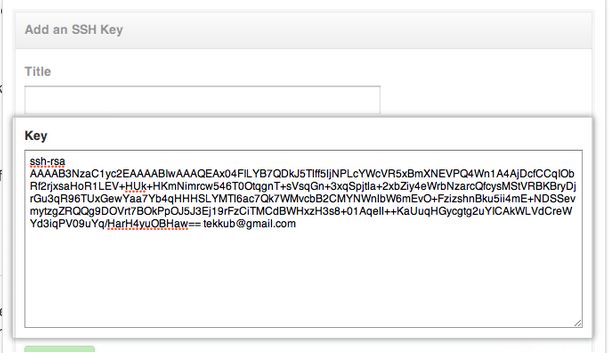
* 1. 在左边菜单，点击“SSH Keys”



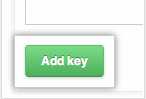
* 1. 点击“Add SSH Key”按钮



* 1. 粘贴你的密钥内容到Key输入框中



* 1. 点击“Add Key”按钮



* 1. 在弹出窗口，输入你的Github密码，点击确定。

1. 测试

为了确认我们可以通过SSH连接github，我们输入下面的命令，输入后，会要求我们提供验证密码，输入之前创建的密码就可以了。

在终端上执行：ssh –T [git@github.com](mailto:git@github.com)

可能会出现如下警示信息：

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)?

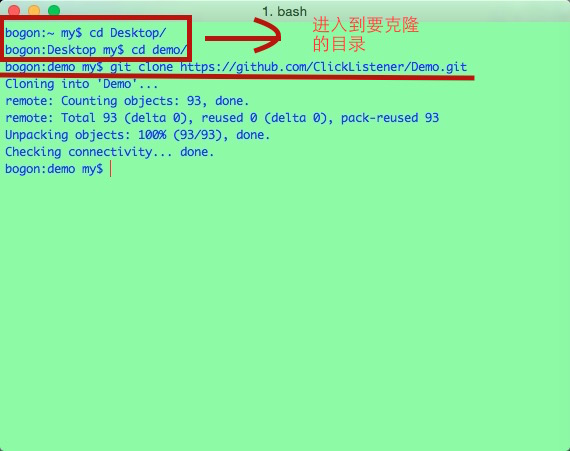
直接输入yes，如果出现如下信息，说明配置成功了。

Hi username! You’ve successfully authenticated, but GitHub does not provide shell access.

二、将github远程服务器的代码clone到本地

方法：执行git clone <server>就可以将远程的代码clone到本地了

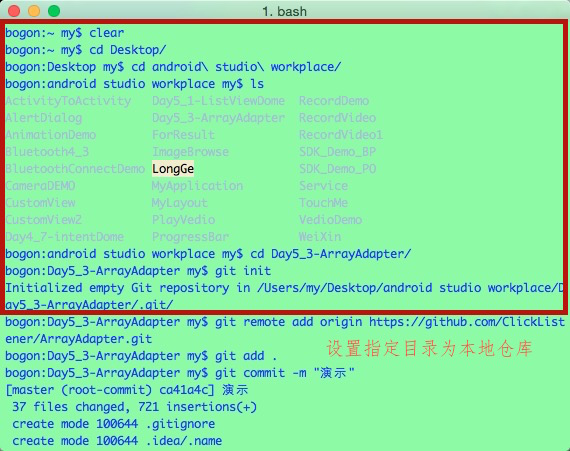
示例：



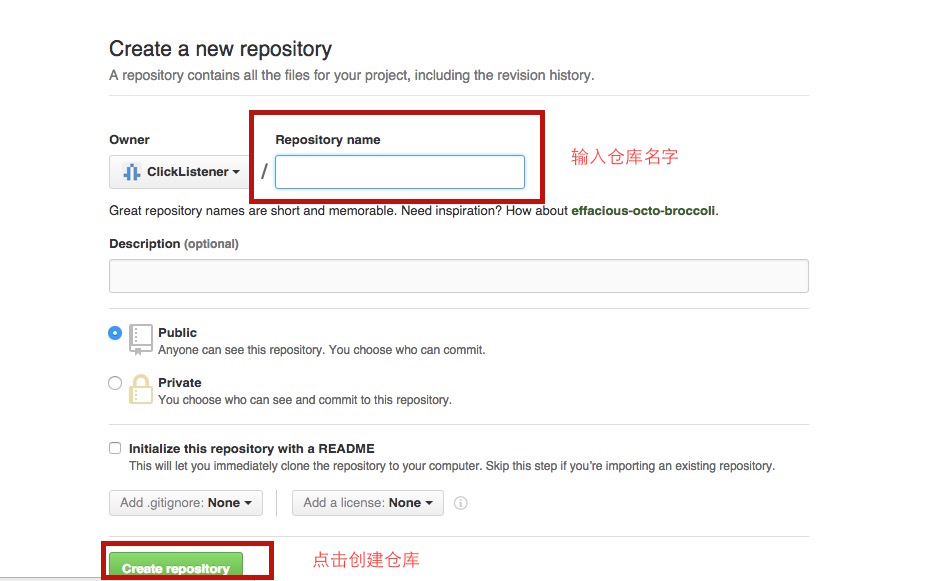
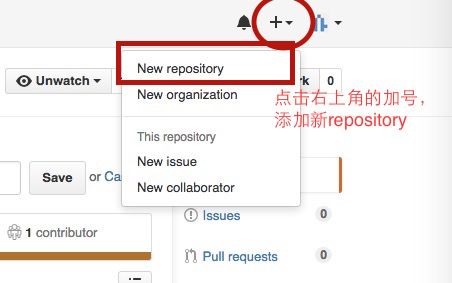
三、 将本地git仓库与某个远程服务器建立连接，并将代码上传到服务器

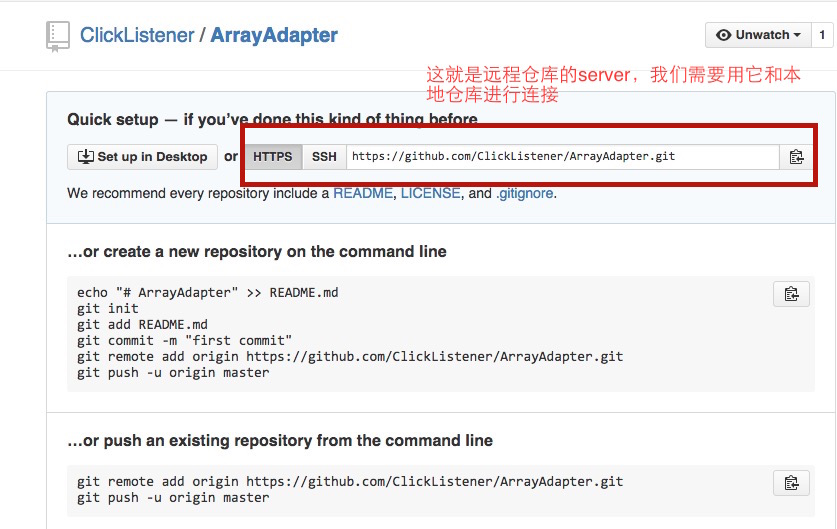
方法：

1、建立本地仓库git init



2、在你的github账号上建立一个repository



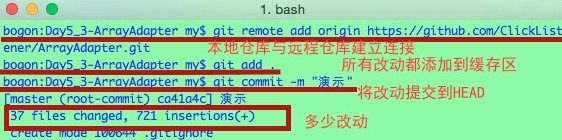


3、本地git仓库与远程仓库建立连接，git remote add origin <server>

备注：你的本地仓库由git维护的三棵“树”组成，第一个是你的工作目录，也就是你存放代码的文件，它持有实际文件；第二个是缓存区，他就像是个缓存区域，临时保存你的改动；最后是HEAD，指向你最近一次提交的结果。

4、执行git add <filename>添加某个文件的改动到缓存区或者git add \*将所有改动都添加到缓存区。

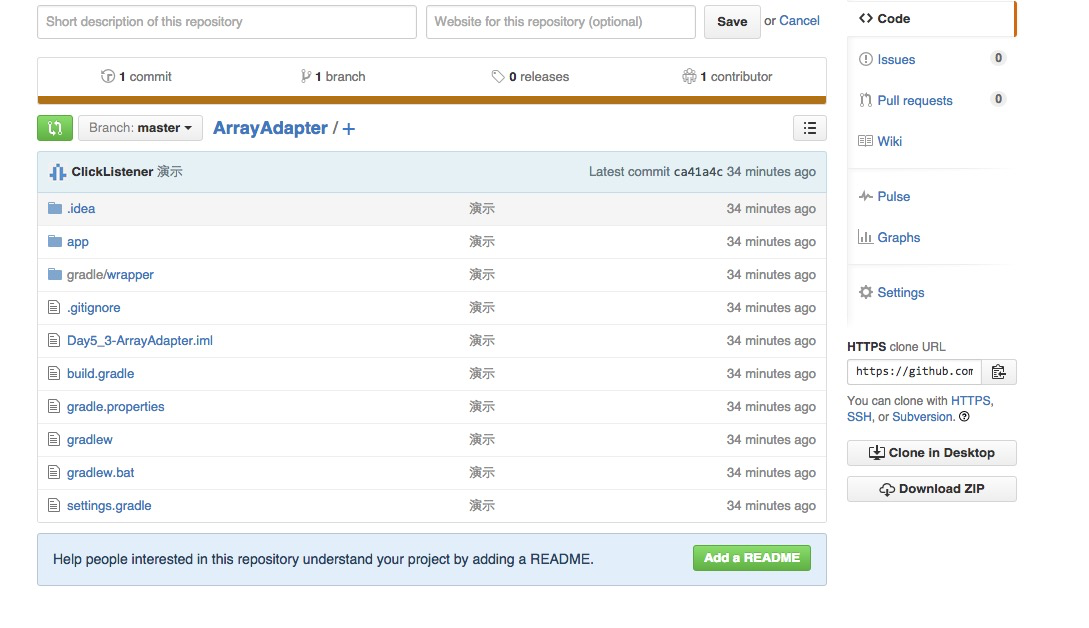
5、执行git commit -m “修改信息说明”将改动提交到HEAD，但是代码还没到你的远程仓库。



6、执行git push origin master将改动推送到远程仓库，注：master可以是任何你想要推送的分支



7、之后你就可以看到你的远程仓库变成这样



**Meteor创建项目和向项目中导入plugin的方法**

1. 安装meteor：执行：https://install.meteor.com | /bin/sh
2. 创建一个项目：meteor create myapp(项目名称)
3. 导入plugin，有两种方式：
4. 如果导入的plugin在github上：

执行meteor add cordova:com.ihealth.plugin.bp.bpmanagercordova@https://github.com/iHealthLab/plugin-ihealth-bp/tarball/3e5db902a9092e28e767095989abc749daebf2a6

1. 如果导入的plugin在本地上：

执行meteor add cordova:com.ihealth.plugin.bp.bpmanagercordova@file:///Users/apple/Documents/plugin-ihealth-bp

1. 运行项目：meteor run android-device –verbose,提示你Please add the Android platform to your project first，先执行meteor add-platform android，然后在执行meteor run android-device –verbose

