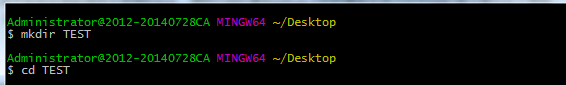
第一步：我们需要先创建一个本地的版本库（其实也就是一个文件夹）。

       你可以直接右击新建文件夹，也可以右击打开Git bash命令行窗口通过命令来创建。

       现在我通过命令行在桌面新建一个TEST文件夹（你也可以在其他任何地方创建这个文件夹），并且进入这个文件夹

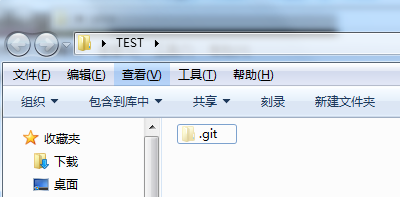




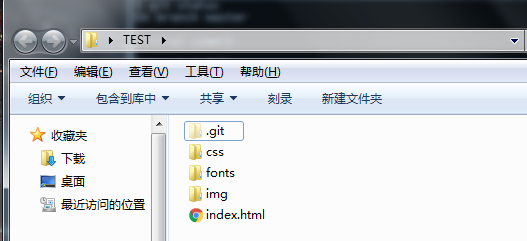
       第二步：通过命令git init把这个文件夹变成Git可管理的仓库

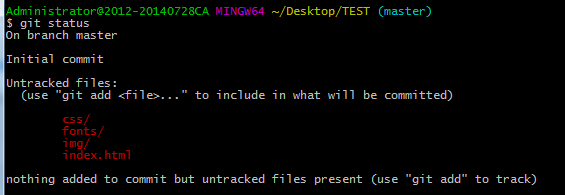
IMG_258

       这时你会发现TEST里面多了个.git文件夹，它是Git用来跟踪和管理版本库的。如果你看不到，是因为它默认是隐藏文件，那你就需要设置一下让隐藏文件可见。

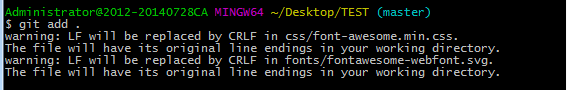


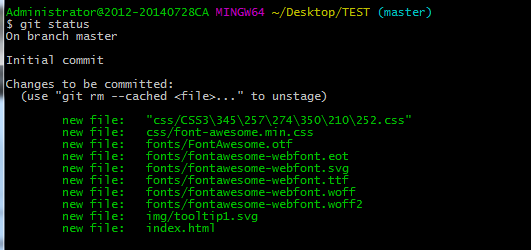
       第三步：这时候你就可以把你的项目粘贴到这个本地Git仓库里面（粘贴后你可以通过git status来查看你当前的状态），然后通过git add把项目添加到仓库（或git add .把该目录下的所有文件添加到仓库，注意点是用空格隔开的）。在这个过程中你其实可以一直使用git status来查看你当前的状态。



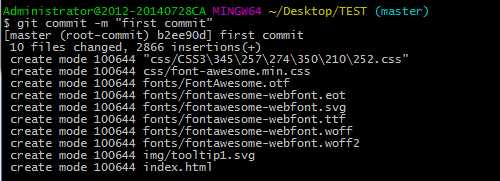


       这里提示你虽然把项目粘贴过来了，但还没有add到Git仓库上，然后我们通过git add .把刚才复制过来的项目全部添加到仓库上。





        第四步：用git commit把项目提交到仓库。



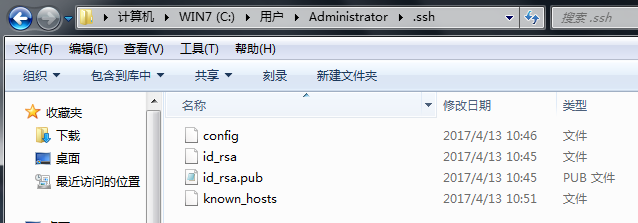
        -m后面引号里面是本次提交的注释内容，这个可以不写，但最好写上，不然会报错，详情自行Google。 好了，我们本地Git仓库这边的工作做完了，下面就到了连接远程仓库（也就是连接Github）

      由于本地Git仓库和Github仓库之间的传输是通过SSH加密的，所以连接时需要设置一下：

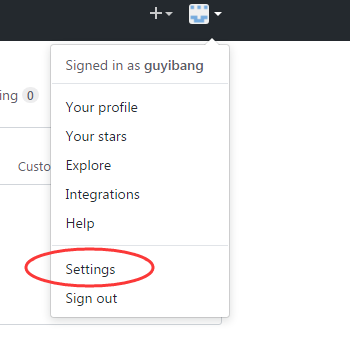
      第五步：创建SSH KEY。先看一下你C盘用户目录下有没有.ssh目录，有的话看下里面有没有id\_rsa和id\_rsa.pub这两个文件，有就跳到下一步，没有就通过下面命令创建

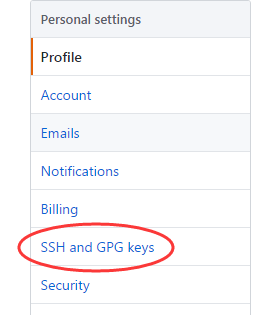
$ ssh-keygen -t rsa -C "youremail@example.com"

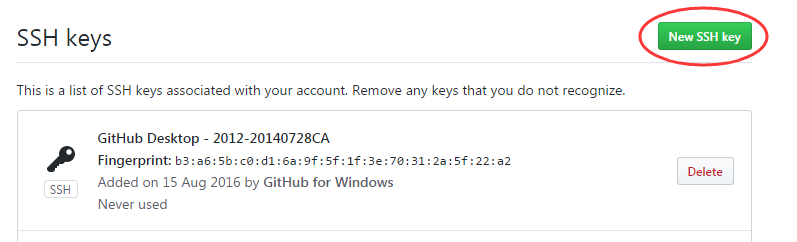
       然后一路回车。这时你就会在用户下的.ssh目录里找到id\_rsa和id\_rsa.pub这两个文件

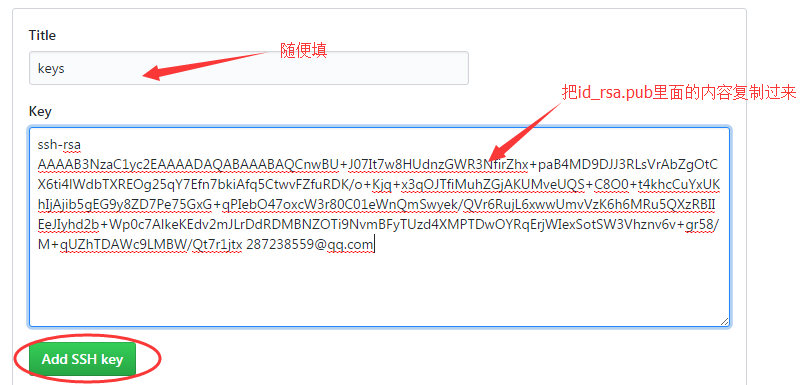


      第六步：登录Github,找到右上角的图标，打开点进里面的Settings，再选中里面的SSH and GPG KEYS，点击右上角的New SSH key，然后Title里面随便填，再把刚才id\_rsa.pub里面的内容复制到Title下面的Key内容框里面，最后点击Add SSH key，这样就完成了SSH Key的加密。具体步骤也可看下面：



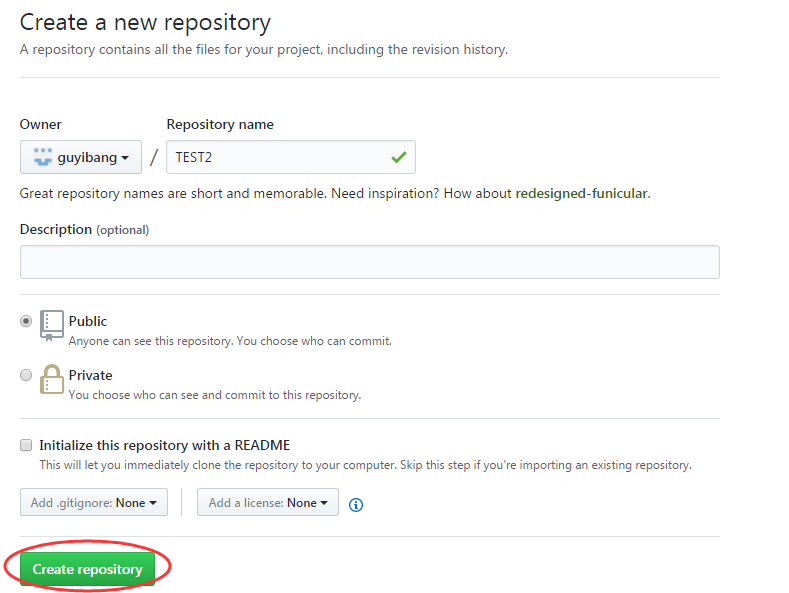






        第七步：在Github上创建一个Git仓库。

     你可以直接点New repository来创建，比如我创建了一个TEST2的仓库（因为我里面已经有了一个test的仓库，所以不能再创建TEST仓库）。



        第八步：在Github上创建好Git仓库之后我们就可以和本地仓库进行关联了，根据创建好的Git仓库页面的提示，可以在本地TEST仓库的命令行输入：

$ git remote add origin https:*//github.com/guyibang/TEST2.git*

IMG_271

        注意origin后面加的是你Github上创建好的仓库的地址。



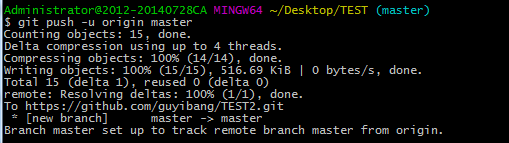
      第九步：关联好之后我们就可以把本地库的所有内容推送到远程仓库（也就是Github）上了，通过：

$ git push -u origin master

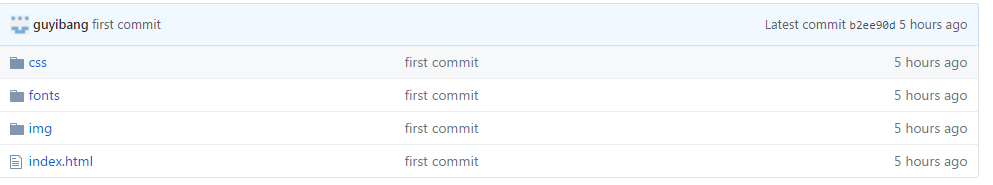
       由于新建的远程仓库是空的，所以要加上-u这个参数，等远程仓库里面有了内容之后，下次再从本地库上传内容的时候只需下面这样就可以了：

$ git push origin master

        上传项目的过程可能需要等一段时间，完成之后是这样的：



        这时候你再重新刷新你的Github页面进入刚才新建的那个仓库里面就会发现项目已经成功上传了：



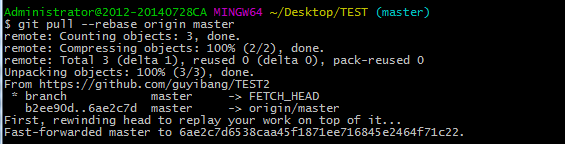
        至此就完成了将本地项目上传到Github的整个过程。

      另外，这里有个坑需要注意一下，就是在上面第七步创建远程仓库的时候，如果你勾选了Initialize this repository with a README（就是创建仓库的时候自动给你创建一个README文件），那么到了第九步你将本地仓库内容推送到远程仓库的时候就会报一个failed to push some refs to https://github.com/guyibang/TEST2.git的错。



      这是由于你新创建的那个仓库里面的README文件不在本地仓库目录中，这时我们可以通过以下命令先将内容合并以下：

$ git pull *--rebase origin master*



       这时你再push就能成功了。

     总结：其实只需要进行下面几步就能把本地项目上传到Github

     1、在本地创建一个版本库（即文件夹），通过git init把它变成Git仓库；

     2、把项目复制到这个文件夹里面，再通过git add .把项目添加到仓库；

     3、再通过git commit -m "注释内容"把项目提交到仓库；

     4、在Github上设置好SSH密钥后，新建一个远程仓库，通过git remote add origin https://github.com/guyibang/TEST2.git将本地仓库和远程仓库进行关联；

     5、最后通过git push -u origin master把本地仓库的项目推送到远程仓库（也就是Github）上；（若新建远程仓库的时候自动创建了README文件会报错，解决办法看上面）。