git config --global user.name ”jiashauifei”

git config --global user.email [”1097697383@qq.com”](mailto:\”1097697383@qq.com\”)

mkdir learngit 创建learngit的文件夹

cd learngit 进入learngit文件夹

pwd 显示当前目录

git init 把当前目录变为可管理的仓库（初始化仓库）

ls 显示文件

ls -ah 显示隐藏的文件

git add readme.txt 把readme.txt文件添加到仓库

git commit -m “wrote a readme.txt file” 告诉git ，把文件提交到仓库，并注释

git status 查看仓库当前状态

git diff 查看修改内容

git log 查看历史版本，提交日志，查看提交历史，以便确定要回退到哪个版本

Git reset 回退版本

Git reset --hard HEAD^ 回退到上一个版本

Git reset --hard HEAD^^ 回退到上上一个版本

Git reset --hard HEAD~100 回退到前一百个版本

Cat readme.txt 查看readme.txt的内容

Git reflog 查看命令历史，以便确定要回到未来的哪个版本

Git checkout -- readme.txt 丢掉工作区的修改

git checkout其实是用版本库里的版本替换工作区的版本，无论工作区是修改还是删除，都可以“一键还原”。

命令git checkout -- readme.txt意思就是，把readme.txt文件在工作区的修改全部撤销，这里有两种情况：

一种是readme.txt自修改后还没有被放到暂存区，现在，撤销修改就回到和版本库一模一样的状态；

一种是readme.txt已经添加到暂存区后，又作了修改，现在，撤销修改就回到添加到暂存区后的状态。

总之，就是让这个文件回到最近一次git commit或git add时的状态。

git checkout -- file命令中的--很重要，没有--，就变成了“切换到另一个分支”的命令，我们在后面的分支管理中会再次遇到git checkout命令。

Git reset HEAD readme.txt 把暂存区的修改撤销掉，放回工作区

（git commit 之前）

rm test.txt 删除test.txt文件

git rm test.txt 从版本库中删除文件

ssh-keygen -t rsa -C ["1097697383@qq.com"](mailto:\"youremail@example.com\") 创建SSH Key

如果一切顺利的话，可以在用户主目录里找到.ssh目录，里面有id\_rsa和id\_rsa.pub两个文件，这两个就是SSH Key的秘钥对，id\_rsa是私钥，不能泄露出去，id\_rsa.pub是公钥，可以放心地告诉任何人。

第2步：登陆GitHub，打开“Account settings”，“SSH Keys”页面：

然后，点“Add SSH Key”，填上任意Title，在Key文本框里粘贴id\_rsa.pub文件的内容：

假定你有若干电脑，你一会儿在公司提交，一会儿在家里提交，只要把每台电脑的Key都添加到GitHub，就可以在每台电脑上往GitHub推送了

git remote add origin <https://github.com/Mrjiajia/test.git> 关联远程库

添加后，远程库的名字就是origin，这是Git默认的叫法，也可以改成别的，但是origin这个名字一看就知道是远程库。

git push -u origin master 把本地库的所有内容推动到远程库上

把本地库的内容推送到远程，用git push命令，实际上是把当前分支master推送到远程。

由于远程库是空的，我们第一次推送master分支时，加上了-u参数，Git不但会把本地的master分支内容推送的远程新的master分支，还会把本地的master分支和远程的master分支关联起来，在以后的推送或者拉取时就可以简化命令。

分支：

Git branch dev 创建分支dev

Git checkout dev 切换到分支dev

Git checkout -b dev 创建分支dev并切换到分支dev

Git branch 查看当前分支，当前分支前会有一个\*号

Git merge dev 合并dev分支到当前分支

Git branch -d dev 删除分支dev