GIT分布式版本控制工具

不但能自动帮我记录每次文件的改动，还可以让同事协作编辑

Git安装完成时 还需要设置：

$ git config --global user.name "Your Name"$ git config --global user.email "email@example.com"

因为Git是分布式版本控制系统，所以，每个机器都必须自报家门：你的名字和Email地址。你也许会担心，如果有人故意冒充别人怎么办？这个不必担心，首先我们相信大家都是善良无知的群众，其次，真的有冒充的也是有办法可查的。

Pwd命令显示当前目录

Git init把Git变成可以管理的仓库

把文件添加到版本库

所有的版本控制系统，其实只能跟踪文本文件的改动，比如TXT文件，网页，所有的程序代码等等 word是二进制文件不行

注意：不要用window编写utf-8

$ git add readme.txt

git commit告诉Git，把文件提交到仓库

$ git commit -m "wrote a readme file"

Git status

git diff

git log

回退

$ git reset --hard HEAD^

HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，使用命令git reset --hard commit\_id。

穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。

要重返未来，用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

就是让这个文件回到最近一次git commit或git add时的状态。

git reset HEAD readme.txt

命令git reset HEAD file可以把暂存区的修改撤销掉（unstage），重新放回工作区：

场景1：当你改乱了工作区某个文件的内容，想直接丢弃工作区的修改时，用命令git checkout -- file。

场景2：当你不但改乱了工作区某个文件的内容，还添加到了暂存区时，想丢弃修改，分两步，第一步用命令git reset HEAD file，就回到了场景1，第二步按场景1操作。

场景3：已经提交了不合适的修改到版本库时，想要撤销本次提交，参考[版本回退](https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/0013744142037508cf42e51debf49668810645e02887691000" \t "https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000/_blank)一节，不过前提是没有推送到远程库。

远程仓库：

ssh key

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

为什么GitHub需要SSH Key呢？因为GitHub需要识别出你推送的提交确实是你推送的

你在本地关联的就是我的远程库s

$ git remote add origin git@github.com:michaelliao/learngit.git

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

关联后，使用命令git push -u origin master第一次推送master分支的所有内容；

### …or create a new repository on the command line

echo "# Spring-Hibernate" >> README.mdgit initgit add README.mdgit commit -m "first commit"git remote add origin https://github.com/zhaoxiaowu/Spring-Hibernate.gitgit push -u origin master

### …or push an existing repository from the command line

git remote add origin https://github.com/zhaoxiaowu/Spring-Hibernate.gitgit push -u origin master

### …or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.