//game start with no way home

git init

git add 很多文件

git commit -m “说明，很重要”

git status 看当前分支是什么状态，比如你之前的文件做了什么修改都可以通过这个查看

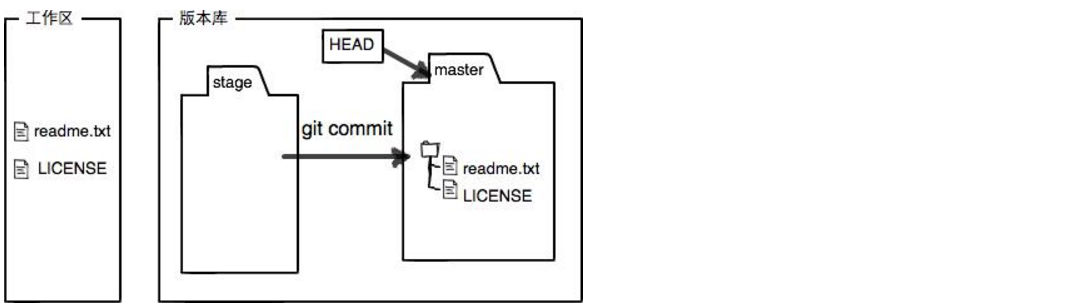
git diff 查看现在文件和版本库中不同点

HEAD 切换版本:

1.git reflog 查看git id 然后继续

2.git reset --hard 编号

暂存区的理解



当你修改了工作区的代码后需要先提交到暂存区（ git add . )然后再将暂存区的代码提交到master分支，也就是(git commit -m "")

管理修改

git diff HEAD -- readme.txt 查看现在的版本和工作区的readme.txt版本有什么区别。

如果修改好不add到暂存区，那么commit的时候不会提交到master分支。

修改撤销

git checkout -- test5.txt

如果你现在的版本已经add，那么checkout到暂存区的版本

如果你没有add，那么就checkout到最新的版本，也就是工作区的

如果已经add，没有commit，那么就git reset HEAD test1.txt 将文件从master分支覆盖掉暂存区

删除

git rm test1.txt

git commit -m "remove file" //从版本库中删除文件

\*\*\*远程仓库

（

\*\*创建本地的秘钥和公钥

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

生成git密钥和公钥

ssh-keygen -t rsa -C "574182446@qq.com"

然后按三下回车就可以了

上传到git,进入到自己的项目下

git init . 注意这里有“."

git pull git@git.oschina.net:chris\_zhj/spider.git

<https://my.oschina.net/158/blog/530112>将项目上传oscChina

）

目前，在GitHub上的这个learngit仓库还是空的，可以从这个仓库克隆出新的仓库，也可以把一个已有的本地仓库与之关联，然后，把本地仓库的内容推送到GitHub仓库。

现在，我们根据GitHub的提示，在本地的oscGit仓库下运行命令：

$ git remote add origin git@github.com:ChrisBamaofan/oscGit.git

请千万注意，把上面的michaelliao替换成你自己的GitHub账户名，但是你以后否则，你在本地关联的就是我的远程库，关联没有问题，推送是推不上去的，因为你的SSH Key公钥不在我的账户列表中。

接下去就是将本地的master分支提交到远程的默认分支origin分支

git push -u origin master //push data to remote repository

现在开始只要本地有什么修改，都可以通过下面的指令，将本地代码提交到远程

git push origin master //将本地的最新修改提交到origin



\*\*\*克隆一个仓库

克隆一个仓库，

首先init一个本地的仓库，将公钥添加到GitHub的sshkey中，

首先必须知道仓库的地址，然后使用git clone命令克隆。

然后就是staged和master的区别，以及远程的origin分支

\*\*\*创建dev分支

git checkout -b dev //这一步相当于创建分支并且切换到分支，

git commit -m "dev" //这一步就是将Dev分支添加repository，

git checkout master

git merge dev //git merge 命令用于合并指定分支到当前分支

git branch -d dev



\*\*\*分支管理



\*\*\*如何参与Github的开源项目，



文件系统：词典的目录索引就是文件系统帮助我们查找文件具体在什么位置。

git仓库，用于存放备份文件的地方。

HEAD：通过git log 查看版本编号 head指针切换版本每个版本关联一份快照，每个快照关联一系列文件，就是“HEAD->快照->仓库文件”这样一个关系链，我们可以很轻松的通过移动HEAD指针来改变我们的工作区文件。