Lesson2

**版本回退**

1. 查看状态

要随时掌握工作区的状态，使用git status命令。git status可以告诉你中否有文件被修改过。

1. 提交问题

修改某文件内容后，要先 git add 进行添加，然后可以采取其他操作。 比如 git diff，查看修改的内容，。然后 git commit 进行提交。

1. 版本回退

当你觉得文件修改到一定程度的时候，就可以“保存一个快照”，这个快照在Git中被称为commit。一旦你把文件改乱了，或者误删了文件，还可以从最近的一个commit恢复，然后继续工作。

在Git中，我们用git log命令查看历史记录：

git log命令显示从最近到最远的提交日志

如果嫌输出信息太多，看得眼花缭乱的，可以试试加上--pretty=oneline参数

其中，看到的一大串类似3628164...882e1e0的是commit id（版本号）

首先，Git必须知道当前版本是哪个版本，在Git中，用HEAD表示当前版本，也就是最新的提交3628164...882e1e0（注意我的提交ID和你的肯定不一样），上一个版本就是HEAD^，上上一个版本就是HEAD^^，当然往上100个版本写100个^比较容易数不过来，所以写成HEAD~100。

回退时，可以使用git reset 命令（hard参数的意义后面讲）



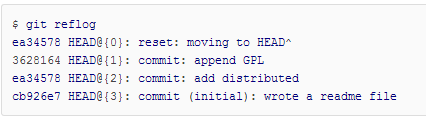
若还想再回去，若刚才的命令行没关闭，刚可以找到版本号，利用



可以回到老版本。

版本号没必要写全，前几位就可以了，Git会自动去找。当然也不能只写前一两位，因为Git可能会找到多个版本号，就无法确定是哪一个了。

若版本号无法找到，则使用 git reflog 命令来查找每一次输入的命令。



### 小结

现在总结一下：

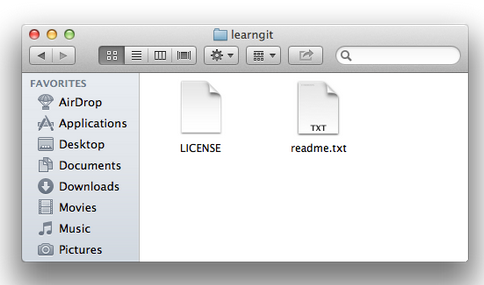
* HEAD指向的版本就是当前版本，因此，Git允许我们在版本的历史之间穿梭，使用命令git reset --hard commit\_id。
* 穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。
* 要重返未来，用git reflog查看命令历史，以便确定要回到未来的哪个版本。

**工作区和暂存区**

Git和其他版本控制系统如SVN的一个不同之处就是有暂存区的概念。

#### 工作区（Working Directory）

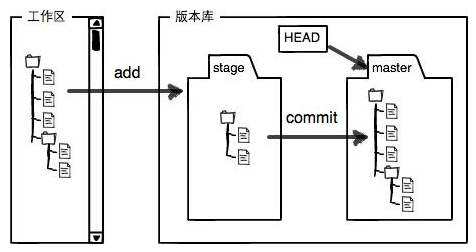
就是你在电脑里能看到的目录，比如我的learngit文件夹就是一个工作区



#### 版本库（Repository）

工作区有一个隐藏目录.git，这个不算工作区，而是Git的版本库。

Git的版本库里存了很多东西，其中最重要的就是称为stage（或者叫index）的暂存区，还有Git为我们自动创建的第一个分支master，以及指向master的一个指针叫HEAD。



分支和HEAD的概念我们以后再讲。

把文件往Git版本库里添加的时候，是分两步执行的：

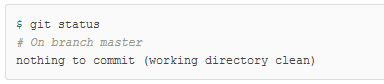
第一步是用git add把文件添加进去，实际上就是把文件修改添加到暂存区；

第二步是用git commit提交更改，实际上就是把暂存区的所有内容提交到当前分支。

因为我们创建Git版本库时，Git自动为我们创建了唯一一个master分支，所以，现在，git commit就是往master分支上提交更改。

git add命令实际上就是把要提交的所有修改放到暂存区（Stage），然后，执行git commit就可以一次性把暂存区的所有修改提交到分支。

一旦提交后，如果你又没有对工作区做任何修改，那么工作区就是“干净”的：



**修改**

1. 管理修改

每次修改，如果不add到暂存区，那就不会加入到commit中。

1. 撤销修改

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，参考版本回退一节，不过前提是没有推送到远程库。

1. 删除文件

在本地删除文件后，工作区和版本库就不一致了。有两个选择：

一是确实要从版本库中删除该文件，那就用命令git rm删掉，并且git commit：

另一种情况是删错了，因为版本库里还有呢，所以可以很轻松地把误删的文件恢复到最新版本：(先用git status 查看哪些文件被删除了)



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”。

**小结**

命令git rm用于删除一个文件。如果一个文件已经被提交到版本库，那么你永远不用担心误删，但是要小心，你只能恢复文件到最新版本，你会丢失**最近一次提交后你修改的内容**。