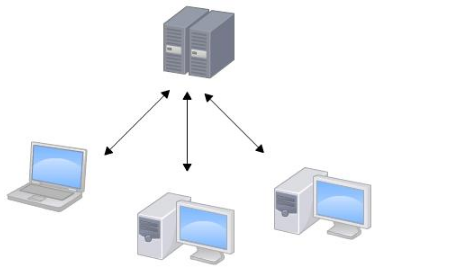
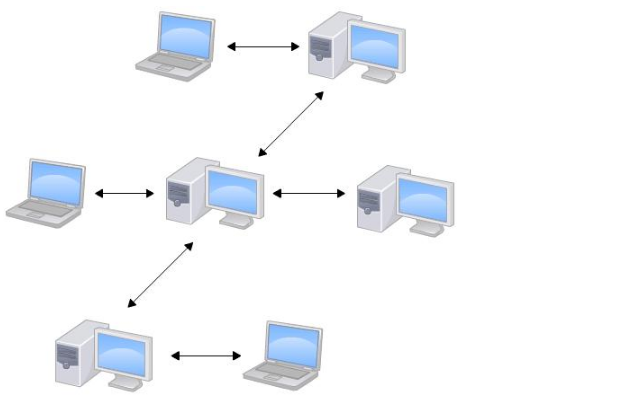
集中式版本控制系统，版本库是集中存放在中央服务器的，而干活的时候，用的都是自己的电脑，所以要先从中央服务器取得最新的版本，然后开始干活，干完活了，再把自己的活推送给中央服务器。

中央服务器就好比是一个图书馆，你要改一本书，必须先从图书馆借出来，然后回到家自己改，改完了，再放回图书馆。



Git的优势不单是不必联网这么简单，后面我们还会看到Git极其强大的分支管理，把SVN等远远抛在了后面!!!

在实际使用分布式版本控制系统的时候，其实很少在两人之间的电脑上推送版本库的修改，因为可能你们俩不在一个局域网内，两台电脑互相访问不了，也可能今天你的同事病了，他的电脑压根没有开机。因此，分布式版本控制系统通常也有一台充当“中央服务器”的电脑，但这个服务器的作用仅仅是用来方便“交换”大家的修改，没有它大家也一样干活，只是交换修改不方便而已。



Git版本回退:



然后顺便把工作区的文件更新了, 所以你让HEAD指向哪个版本号, 你就把当前版本定位在哪!!!

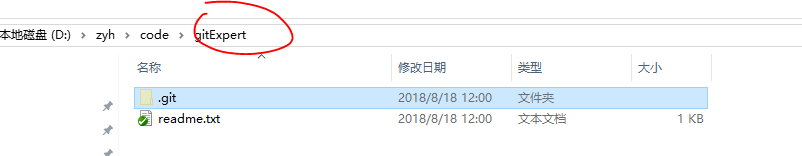
工作区(Working Directory)

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!!!

前面讲了我们把文件往Git版本库里添加的时候, 是分两步执行的:

(1).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;

(2).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!!!

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:

$ git status

On branch master

nothing to commit, working tree clean

现在版本库变成了这样, 暂存区就没有任何内容了:



总结:

(1).暂存区是Git非常重要的概念, 弄明白了暂存区, 就弄明白了Git的很多操作到底干了什么!!!