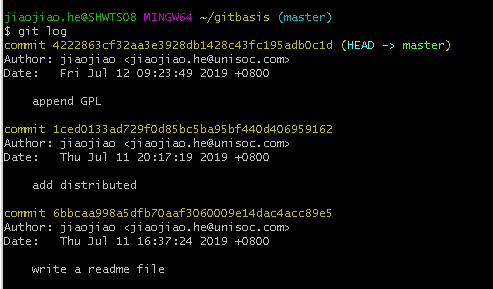
第一、git学习

当文件被多次修改，我们可以使用git lo查看修改详情。git log命令显示从最近到最远的提交日志



其中一大串数字显示的是commit\_id的版本号。Git是分布式的版本控制系统，因此需要使用不同的id进行区分。

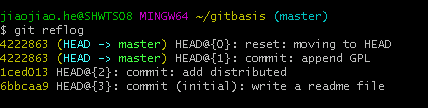
这里可以使用git reset来恢复上一个版本，在Git中，用HEAD表示当前版本，上一个版本就是HEAD^，上上一个版本就是HEAD^^，当然往上100个版本写100个^比较容易数不过来，所以写成HEAD~100



cat readme.txt查看当前版本。



在Git中用$ git reset --hard HEAD^回退到add distributed版本时，再想恢复到append GPL，就必须找到append GPL的commit id。Git提供了一个命令git reflog用来记录你的每一次命令：



总结：

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。

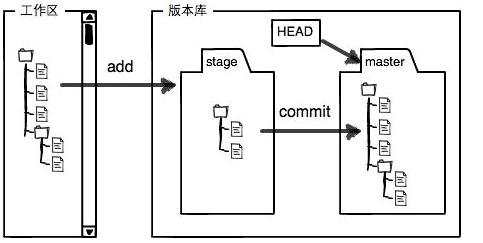
* 穿梭前，用git log可以查看提交历史，以便确定要回退到哪个版本。
* 要重返未来，用git reflog查看命令历史，以便确定要回到未来的哪个版本。

第二、工作区和暂存区

工作区：一般能看到的目录

暂存区：工作区有一个暂存的.git，是git的版本号

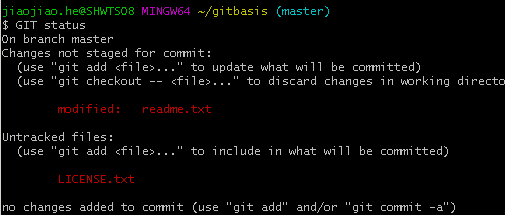
Git的版本库里存了很多东西，其中最重要的就是称为stage（或者叫index）的暂存区，还有Git为我们自动创建的第一个分支master，以及指向master的一个指针叫HEAD



一般地，git添加文件需要两步，

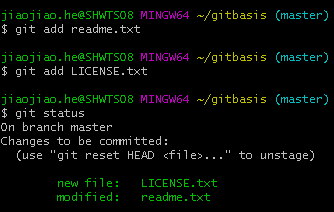
1. git add来增加文件，实际上是将文件增加到暂存区
2. git commit用以提交修改，实际上是将暂存区的内容存放在当前分支master.

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

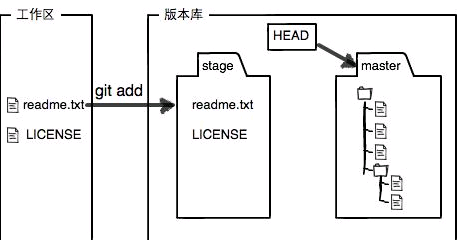


Git非常清楚地告诉我们，readme.txt被修改了，而LICENSE还从来没有被添加过，所以它的状态是Untracked

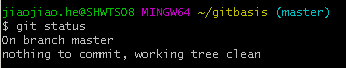
使用两次命令git add，把readme.txt和LICENSE都添加后，用git status再查看一下



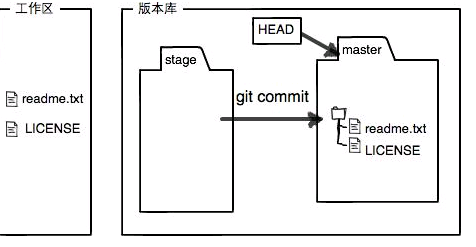
暂存区的状态就变成这样了：



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



现在版本库变成了这样，暂存区就没有任何内容了：



小结

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

第三、

调试知识了解：

Monkey测试是Android自动化测试的一种手段。Monkey测试本身非常简单，就是模拟用户的按键输入，触摸屏输入，手势输入等，看设备多长时间会出异常。

当Monkey程序在模拟器或真实设备运行的时候，程序会产生一定数量或一定时间内的随机模拟用户操作的事件, 如点击，按键，手势等， 以及一些系统级别的事件。通常也称随机测试或者稳定性测试。

Monkey测试是Android自动化测试的一种手段，Monkey测试本身非常简单，就是模拟用户的按键输入，触摸屏输入，手势输入等，看设备多长时间会出异常。  
当Monkey程序在模拟器或设备运行的时候，如果用户出发了比如点击，触摸，手势或一些系统级别的事件的时候，它就会产生随机脉冲，所以可以用Monkey用随机重复的方法去负荷测试你开发的软件.  
最简单的方法就是用用下面的命令来使用Monkey，这个命令将会启动你的软件并且触发500个事件.  
$ adb shell monkey -v -p your.package.name 500

Memleak 是一个C语言内存泄漏检测工具。 原理是利用 C 语言的宏调用来替代原有的函数调用， memleak 维护了一个链表，在这个链表中保存着程序中对内存函数调用的记录，这些函数包括：malloc、calloc、realloc、free。