<https://www.yiibai.com/git/git_pull.html>

<http://www.runoob.com/git/git-workspace-index-repo.html>

<https://www.liaoxuefeng.com/wiki/0013739516305929606dd18361248578c67b8067c8c017b000>

Github登录网址

<https://github.com/login?return_to=https%3A%2F%2Fgithub.com%2Fnew>

Github账号

# [PerfictGrassland](https://github.com/PerfictGrassland)

3416890246@qq.com

abc822912

HappyMoment

2395209435@qq.com

abc822912

基本概念

* **工作区：**就是你在电脑里能看到的目录。
* **暂存区：**英文叫stage, 或index。一般存放在 ".git目录下" 下的index文件（.git/index）中，所以我们把暂存区有时也叫作索引（index）。
* **版本库：**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。



图中左侧为工作区，右侧为版本库。在版本库中标记为 "index" 的区域是暂存区（stage, index），标记为 "master" 的是 master 分支所代表的目录树。

"HEAD" 实际是指向 master 分支的一个"游标"，所以图示的命令中出现 HEAD 的地方可以用 master 来替换。

**图中的 objects 标识的区域为 Git 的对象库，实际位于 ".git/objects" 目录下，里面包含了创建的各种对象及内容。**

A,当对工作区修改（或新增）的文件执行 **"git add"** 命令时，暂存区的目录树**（master）**被更新，同时工作区修改（或新增）的文件内容被写入到**对象库中的一个新的对象中（objects）**，而该对象的ID被记录在**暂存区的文件索引**中（**index，stag**e）。

工作区修改---> 暂存区的目录树**（master）更新（主分支）-->同时，更新内容会被写入对象库一个对象库里--->**该对象的ID被记录在**暂存区的文件索引**中（**index，stag**e）

B,当执行提交操作（git commit）时，暂存区的目录树写到版本库（对象库）中，master 分支会做相应的更新。即 master 指向的目录树就是提交时暂存区的目录树。

C,当执行 "git reset HEAD" 命令时，暂存区的目录树会被重写，被 master 分支指向的目录树所替换，但是工作区不受影响

D,当执行 "git rm --cached <file>" 命令时，会直接从暂存区删除文件，工作区则不做出改变。

**E**,当执行 "git checkout ." 或者 "git checkout -- <file>" 命令时，会用暂存区全部或指定的文件替换工作区的文件。**这个操作很危险，会清除工作区中未添加到暂存区的改动**。

**F,**当执行 "git checkout HEAD ." 或者 "git checkout HEAD <file>" 命令时，会用 HEAD 指向的 master 分支中的全部或者部分文件替换暂存区和以及工作区中的文件。这个命令也是极具危险性的，因为不但会清除工作区中未提交的改动，也会清除暂存区中未提交的改动

暂存区index,stage

版本库

目录树，主分支，master，

对象库