git pull命令用于从另一个存储库或本地分支获取并集成(整合)。git pull命令的作用是：取回远程主机某个分支的更新，再与本地的指定分支合并，它的完整格式稍稍有点复杂。

**使用语法**

git pull [options] [<repository> [<refspec>…]]

Shell

**描述**

将远程存储库中的更改合并到当前分支中。在默认模式下，git pull是git fetch后跟git merge FETCH\_HEAD的缩写。

更准确地说，git pull使用给定的参数运行git fetch，并调用git merge将检索到的分支头合并到当前分支中。 使用--rebase，它运行git rebase而不是git merge。

**示例**

以下是一些示例 -

$ git pull <远程主机名> <远程分支名>:<本地分支名>

Shell

比如，要取回origin主机的next分支，与本地的master分支合并，需要写成下面这样 -

$ git pull origin next:master

Shell

如果远程分支(next)要与当前分支合并，则冒号后面的部分可以省略。上面命令可以简写为：

$ git pull origin next

Shell

上面命令表示，取回origin/next分支，再与当前分支合并。实质上，这等同于先做git fetch，再执行git merge。

$ git fetch origin

$ git merge origin/next

Shell

在某些场合，Git会自动在本地分支与远程分支之间，建立一种追踪关系(tracking)。比如，在git clone的时候，所有本地分支默认与远程主机的同名分支，建立追踪关系，也就是说，本地的master分支自动”追踪”origin/master分支。

Git也允许手动建立追踪关系。

$ git branch --set-upstream master origin/next

Shell

上面命令指定master分支追踪origin/next分支。

如果当前分支与远程分支存在追踪关系，git pull就可以省略远程分支名。

$ git pull origin

Shell

上面命令表示，本地的当前分支自动与对应的origin主机”追踪分支”(remote-tracking branch)进行合并。

如果当前分支只有一个追踪分支，连远程主机名都可以省略。

$ git pull

Shell

上面命令表示，当前分支自动与唯一一个追踪分支进行合并。

如果合并需要采用rebase模式，可以使用–rebase选项。

$ git pull --rebase <远程主机名> <远程分支名>:<本地分支名>

Shell

**git fetch和git pull的区别**

1. *git fetch*：相当于是从远程获取最新版本到本地，不会自动合并。

$ git fetch origin master

$ git log -p master..origin/master

$ git merge origin/master

Shell

以上命令的含义：

* 首先从远程的origin的master主分支下载最新的版本到origin/master分支上
* 然后比较本地的master分支和origin/master分支的差别
* 最后进行合并

上述过程其实可以用以下更清晰的方式来进行：

$ git fetch origin master:tmp

$ git diff tmp

$ git merge tmp

Shell

2.*git pull*：相当于是从远程获取最新版本并merge到本地

git pull origin master

Shell

上述命令其实相当于git fetch 和 git merge  
在实际使用中，git fetch更安全一些，因为在merge前，我们可以查看更新情况，然后再决定是否合并。