# branch的简单使用

本地仓库创建：git init

添加文件：git add .所有文件 filename文件名

查看分支：git branch

创建分支：git branch <name>

切换分支：git checkout <name>

创建+切换分支：git checkout -b <name>

合并某分支到当前分支：git merge <name>

删除本地分支：git branch -d <name>

删除远程分支：

git branch -r -d origin/branch-name

git push origin :branch-name

提交：git commit -m "first commit"

添加到远程分支：git remote add origin git @ url 地址

提交到远程分支：git push -u origin master

创建Tag 命令

git tag -a v1.0 -m "对Tag的描述信息"

提交本地的Tag 到远程服务器的命令

git push origin --tags

删除本地的tag 命令

git tag -d v1.0

要删除远程服务器上的tag，可以使用如下的命令：

git push origin --delete tag v1.0。

# tag 的简单使用

以下命令都是我使用 tag 过程中一般会使用到的，可以说都是常用命令。

## 1.创建标签

需要说明的是，创建 tag 是基于本地分支的 commit，而且与分支的推送是两回事，就是说分支已经推送到远程了，但是你的 tag 并没有，如果把 tag 推送到远程分支上，需要另外执行 tag 的推送命令。

git tag <tagName> //创建本地tag

git push origin <tagName> //推送到远程仓库

* 1
* 2

若存在很多未推送的本地标签，你想一次全部推送的话，可以使用一下的命令：

git push origin --tags

* 1
* 2

以上是基于本地当前分支的最后的一个 commit 创建的 tag ，但是如果不想以最后一个，只想以某一个特定的提交为 tag ，也是可以的，只要你知道 commit 的 id。

git log --pretty=oneline //查看当前分支的提交历史，里面包含 commit id

git tag -a <tagName> <commitId>

* 1
* 2

## 2.查看标签

查看本地某个 tag 的详细信息：

git show <tagName>

* 1

查看本地所有 tag ：

//下面两个命令都可以

git tag

git tag -l

* 1
* 2
* 3

查看远程所有 tag：

git ls-remote --tags origin

* 1
* 2

## 3.删除标签

本地 tag 的删除：

git tag -d <tagName>

* 1

远程 tag 的删除：

git push origin :<tagName>

* 1

## 4.重命名标签

这个本质上是删除掉旧名字 tag ，然后再新建新名字 tag ,然后实现重命名的作用。

如果 tag 只存在本地，那么只需要删除本地的旧名字 tag ，然后新建新名字 tag：

git tag -d <oldTagName>

git tag <newTagName>

git push origin <newTagName> //推送到远程仓库

* 1
* 2
* 3

若已经推送到远程了，那么不仅要删除本地的，还要删除远程的，再重新创建和推送：

git tag -d <oldTagName>

git push origin :<oldTagName>

git tag <newTagName>

git push origin <newTagName> //推送到远程仓库

* 1
* 2
* 3
* 4
* 5

## 5.检出标签

命令如下：

git checkout -b <branchName> <tagName>

* 1

因为 tag 本身指向的就是一个 commit，所以和根据 commit id 检出分支是一个道理。   
但是需要特别说明的是，如果我们想要修改 tag 检出代码分支，那么虽然分支中的代码改变了，但是 tag 标记的 commit 还是同一个，标记的代码是不会变的，这个要格外的注意。

## 6. git 根据tag创建分支

原创 2016年04月07日 10:15:58

在项目中我们需要根据tag创建分支.现将创建步骤总结一下.假设在你的主分支上有一个tag为v1.0,主分支的名字为master.

1.执行:git fetch 获得最新.

2.通过:git branch <new-branch-name> <tag-name> 会根据tag创建新的分支.

例如:git branch newbranch v1.0 . 会以tag v1.0创建新的分支newbranch;

3.可以通过git checkout newbranch 切换到新的分支.

4.通过 git push origin newbranch 把本地创建的分支提交到远程仓库.

现在远程仓库也会有新创建的分支啦.