**往一个已经存在的仓库中push code 步骤如下：**

1. Git init
2. git clone (server)
3. git pull server
4. git add \*
5. git commit –m “information”
6. git remote add origin (server)
7. git push origin master (master可以是你想推送的任何分支----git push是推送到远程仓库作用)

**创建本地的分支进行开发，步骤如下：**

通常情况下master分支是不做处理的

1. git checkout –b brachname
2. git checkout master (切换分支)
3. git push origin branchname (如果不推送到远程仓库，这个分支只有你自己可见，其他人无法知道)----远程仓库就显示了该branchname

Example:

git checkout –b develop (开发主分支)

git push origin develop (推送develop分支到远程仓库)

git checkout –b dev1 (开发1分支)

----add and edit some file at local

Git add \* （add到index缓存区）

Git commit –m “information” (提交到heap)

Git push origin dev1 (开发1的分支推送到远程仓库，执行code review，code ok的话，接下来执行合并分支到develop)

Git checkout develop

Git pull origin develop （拉最新develop的代码）

Git merge dev1 （dev1的代码合并到develop）---如果遇到冲突请手动解决

Git push origin develop （done）

git branch –d branchname (Dev1分支的使命完成之后就可以delete改分支了)

**在远程仓库有code的更新，如何merge冲突， 步骤如下：**

1. git pull （拉下最新code）
2. git merge branchname
3. 手动解决冲突
4. Git add (解决冲突的文件)
5. Git commit –m “information”
6. Git push origin develop

得到所有的分支---git fetch