**Clone Case:**

Clone repositories: git clone

Check all branches: git branch -a

Delete remote branch: cd repositoriesname -> git push origin: branchname

Delete local branch: git branch –d branchname

Commit change: git commit –a -> git push origin master

Commit new fiels: git add filename/. -> git commit –a –m “update files” -> git push origin master

Merge remote branch to remote master: git checkout –b localbranchname origin/remotebranchname -> git checkout master -> git pull origin remotebranchname -> git merge remotebranchname -> git push origin master

|  |  |
| --- | --- |
|  | 往一个已经存在的仓库中push code 步骤如下：  通常情况下master是不做开发的，所以要创建本地的分支进行开发，并推送到远程仓库，分支合并到开发主干后，可以删除该分支 |
|  |  |
|  | 1. Open git bash in any folder 2. git clone url （比如克隆的分支有master，develop---开发主干） 3. cd repositoryname (切换到克隆的仓库名下进入master) 4. git checkout –b branchname1 (在本地创建一个分支用于开发功能，并且切换到新建的分支) 5. Add new file or update code 6. git add \* 7. git commit –m “information” -a 8. git push origin branchname1（把本地分支推送到远程仓库，并且得到一个新的远程分支branchname1；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） 9. git checkout develop (切换到远程的开发主干develop，并且把新建的开发分支branchname1合并到开发主干develop上) 10. git pull origin develop (获取远程开发主干的最新代码，因为其他人可能会merge新代码到开发主干) 11. git merge branchname1 (开发分支branchname1合并到开发主干develop，这里可能会产生冲突，因为别人的改动会可能和你的改动有冲突) 12. 手动解决冲突文件 13. Git add fiel (解决冲突的文件) 14. Git commit –m “information” 15. git merge branchname1 16. git push origin develop (将merge后的develop 推送到远程仓库) 17. git branch –d branchname (删除本地分支) 18. git push origin :localdev1 (删除远程分支) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  | 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 |