# [git创建分支并提交项目](http://www.cnblogs.com/YingYue/p/5894680.html)

**git 创建分支, 切换分支, 合并分支, 删除分支及提交[commit提交到本地仓库push名利提交到远程服务器], 检出[pull], 冲突修改, 本地仓库同步远程服务器[pul和push命令l]操作**

**开发中常用操作**

删除文件：git rm 文件名[删除本地git仓库文件, 提交后远程服务器上的文件才会消失]

查看状态：git status

添加记录：git add 文件名 或 git add . [将文件提交到远程git服务器上]

添加描述：git commit -m "描述或备注类似svn提交时的注释"

同步数据：git pull  [获取git远程服务器上的数据也可以具体到某一分支如: git pull origin 分支名]

提交数据：git push origin 分支名

**分支操作**

查看分支：git branch

创建分支：git branch 分支名

切换分支：git checkout 分支名

创建+切换分支：git checkout -b 分支名

合并某分支到当前分支：git merge 分支名

　　[比如，如果要将开发中的分支（deve），合并到稳定分支（master），  
     首先切换的master分支：git checkout master。  
     然后执行合并操作：git merge deve。  
     如果有冲突，会提示你，调用git status查看冲突文件。  
     解决冲突，然后调用git add或git rm将解决后的文件暂存。  
     所有冲突解决后，git commit 提交更改。]

　　注意点:

　　分支衍合  
     分支衍合和分支合并的差别在于，分支衍合不会保留合并的日志，不留痕迹，而 分支合并则会保留合并的日志。  
     要将开发中的分支（dev），衍合到稳定分支（master）。  
     首先切换的master分支：git checkout master。  
     然后执行衍和操作：git rebase deve  
     如果有冲突，会提示你，调用git status查看冲突文件。  
     解决冲突，然后调用git add或git rm将解决后的文件暂存。  
     所有冲突解决后，git rebase --continue 提交更改。

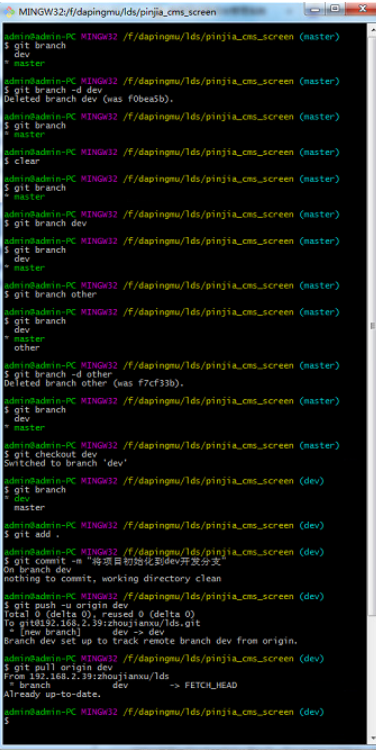
删除分支：git branch -d 分支名

　　删除分支执行git branch -d 分支名  
     如果该分支没有合并到主分支会报错，可以用以下命令强制删除git branch -D 分支名

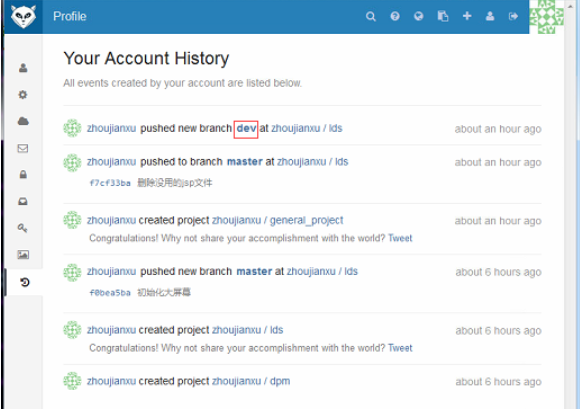
删除远程分支：git push origin :name

撤销修改：git checkout -- file

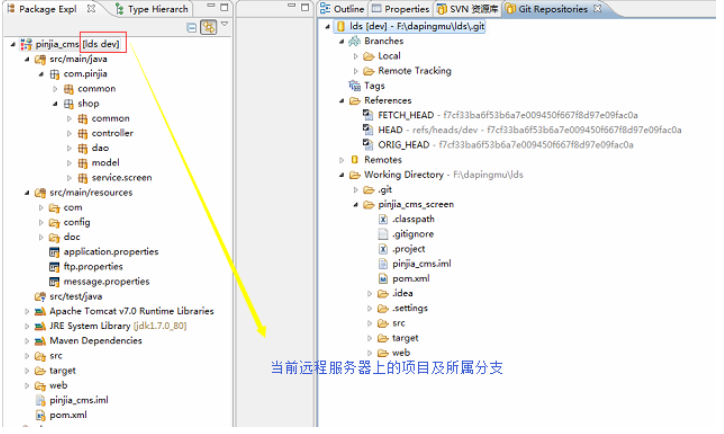
对应操作



GitLab远程服务器上的历史记录



本地仓库



也可以不用命令的模式操作, 可以使用egit插件来进行图像化的commit, update及创建分支;

 git的eclipse或myeclipse插件org.eclipse.egit-updatesite-1.3.0.201202151440-r-site建议采用link方式

http://files.cnblogs.com/files/YingYue/git%E7%9A%84eclipse%E6%88%96myeclipse%E6%8F%92%E4%BB%B6org.eclipse.egit-updatesite-1.3.0.201202151440-r-site.zip