# 关于sts上egit的主要几个相关操作

[ 添加并更新文件 1](#_Toc8901)

[ 删除并更新文件 4](#_Toc14683)

[ 修改并更新文件(用的最多) 6](#_Toc13035)

[ 从远端更新本地文件 8](#_Toc9393)

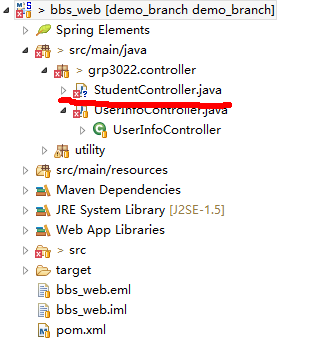
[ 关于版本回退 9](#_Toc12457)

[ 关于作用域（即在哪个文件夹上进行操作） 11](#_Toc13012)

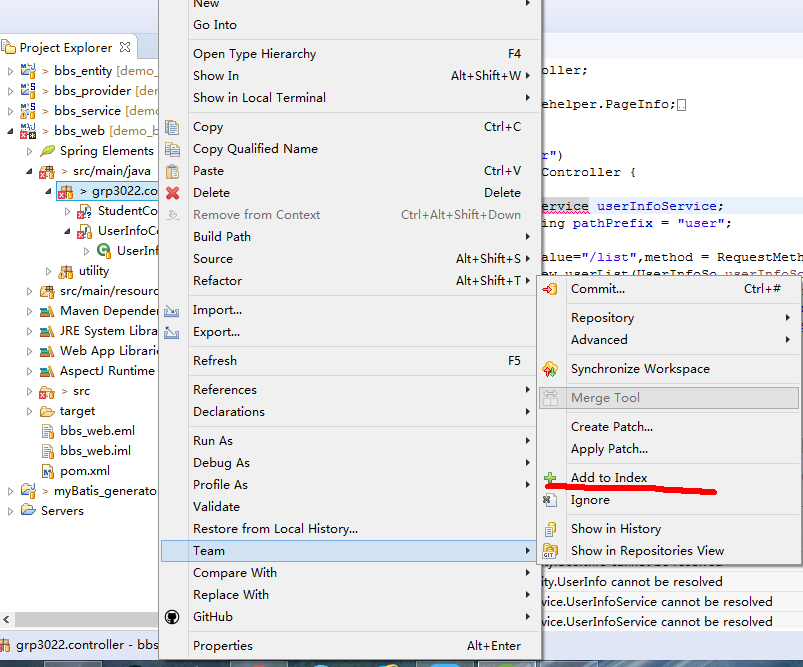
[ 关于Commit message的书写格式规范的建议 14](#_Toc7243)

[ 几个原则 15](#_Toc10608)

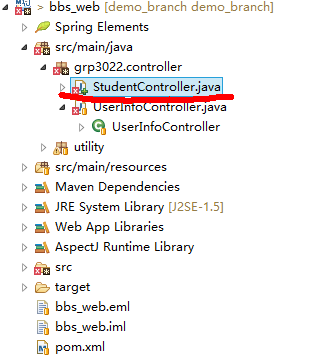
* **添加并更新文件**



在你新添加的文件右键角会有问号标识这是一个新添加的文件

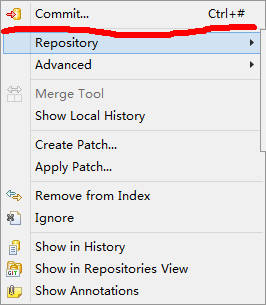


点击add to index

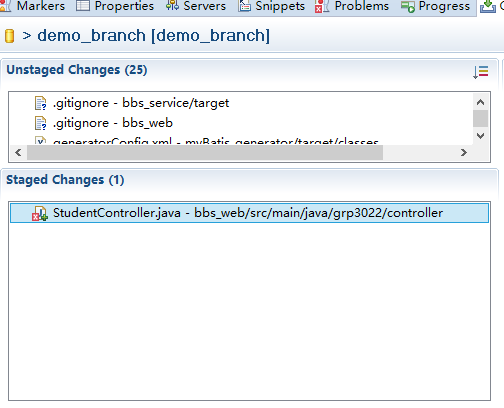


此时图标会显示已添加到本地缓存

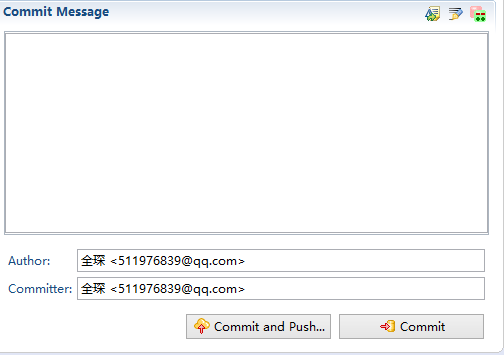
添加之后再右键项目点击team->commit file



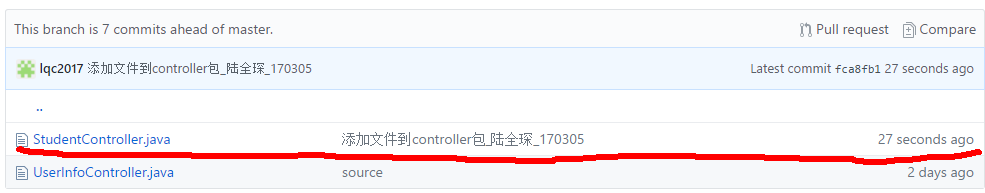
将会看到新添加的文件已经添加到staged strange中，stage strange中的文件是在更新缓存中等待提交的文件，而unstage strange为已更新没有提交到缓存的文件，右键点击即可选择文件是否添加到更新更新缓存中。



右侧的有两个提交按钮，输入框为填写提交信息，提交信息填写格式会在后边做补充[（关于Commit message的书写格式](#书签2)），提交信息是必须要填写的，不然无法提交。

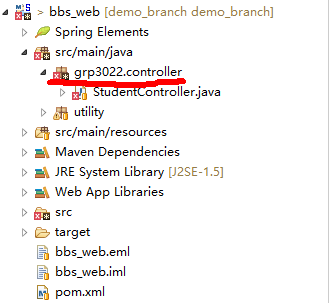


第一个commit and push按钮为提交更新到本地及远端(github网站上的可视化内容)，第二个commit按钮为只更新本地。一般来说我们只要点击第一个按钮就可以了。

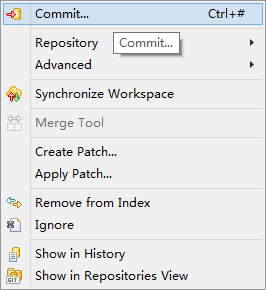


远端所看到的更新。

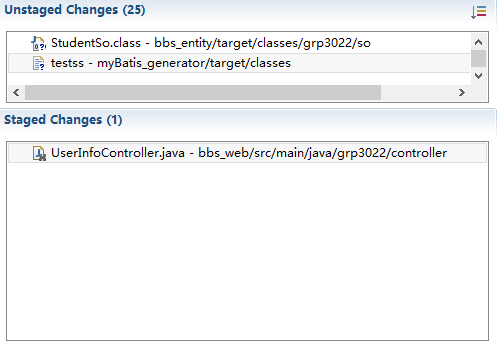
* **删除并更新文件**



删除文件后，父文件夹会有更新提示的标识



此时直接右键父文件夹team->commit

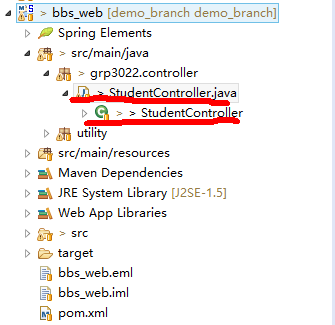


他与添加文件的图标不同，很容易就看出来。之后的更新操作和添加文件相同。

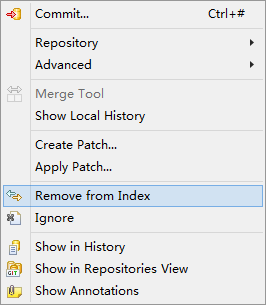


远端同步更新

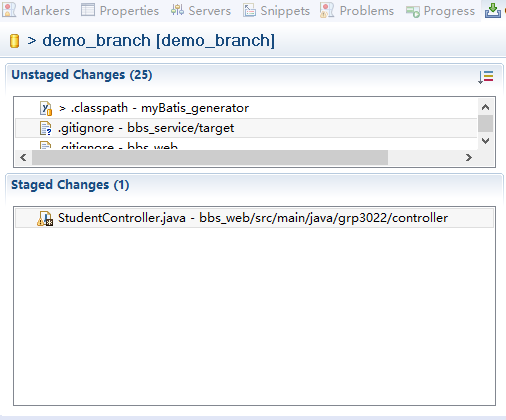
* **修改并更新文件(用的最多)**



修改文件后修改的文件，和父目录都会文件名前都会有>或>>的重定向标识（注意要ctrl+s保存文件后才会显示）



同删除操作一样直接点击commit即可



显示的修改文件。

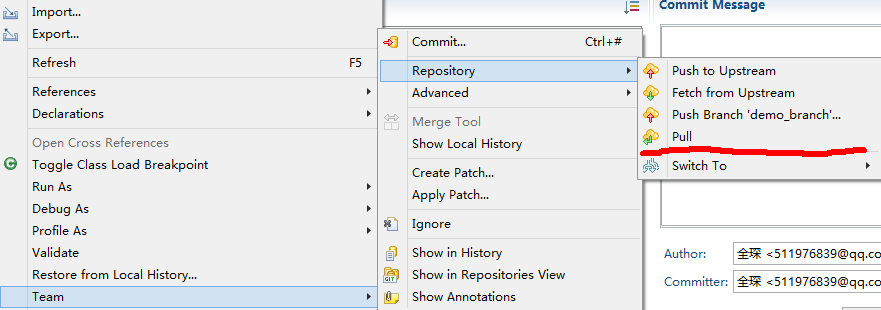
提交修改步骤相同。

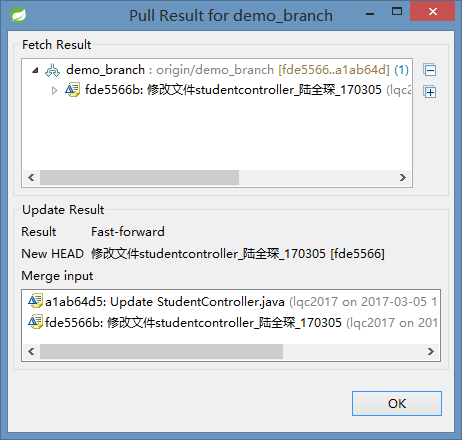
远端：



* **从远端更新本地文件**

右键需要跟新的文件team->repository->pull





Fetch result为远端最新的版本（一般是整合别人提交后的最新版本），merge input中上面的版本为你目前本地的版本，下面版本为远端版本。点击ok即可更新。

* **关于版本回退**

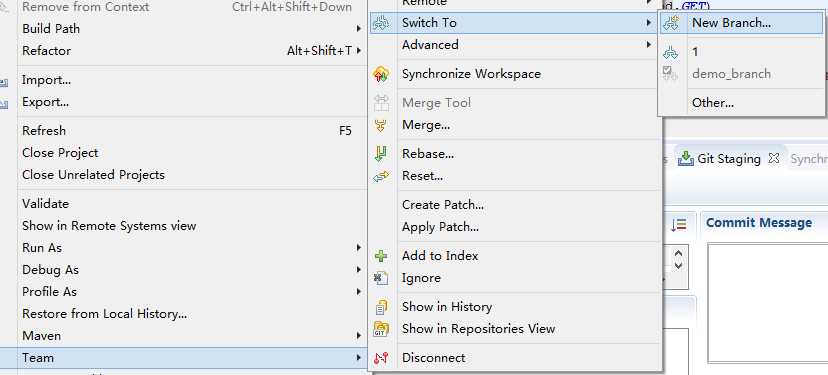
回退情况分析

* 提交了错误的更新。
* 提交了未知的更新。
* 本地从远程更新之后项目无法运行
* 其他Collaborators有需求

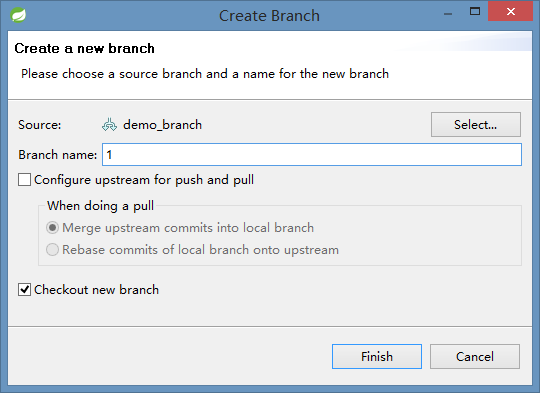
回退操作

1. 本地备份（跟据实际需求看需不需要备份，如果当前版本都无法运行，那就没有备 份的必要）。

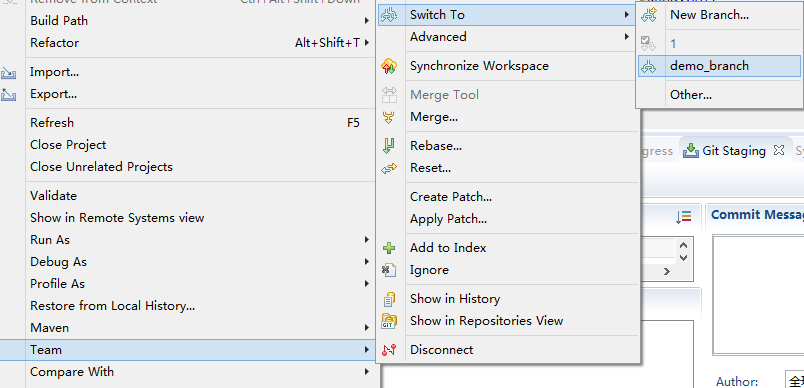
新建一个新的local branch（本地分支）



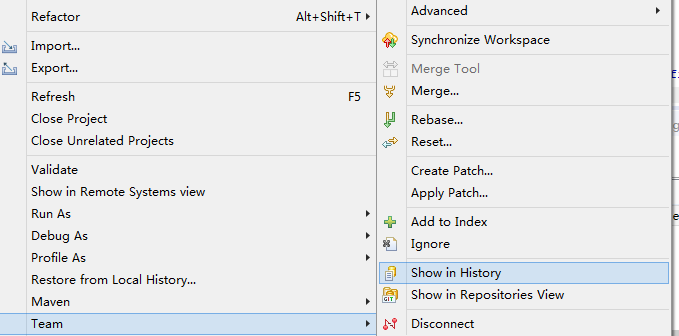
填写好分支名称点击finish



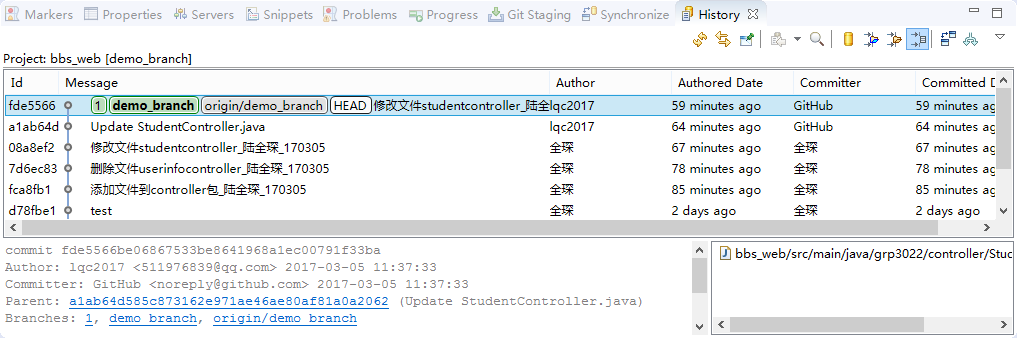
1. 切换回要回退的分支



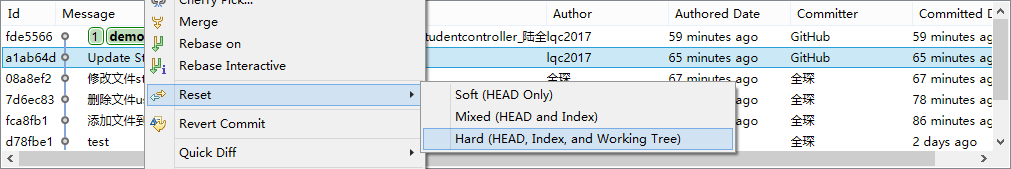
1. 右键项目的根目录点击show in history



历史版本面板



右键要回退的版本rest->hard



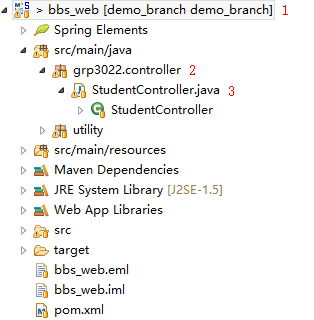
PS:回退版本之后，之前的版本均会丢失，如果没有回退很久以前的版本需求，回 退至能够正常使用的最近版本即可。

如果想撤销回退，重新[从远端跟新本地文件](#书签1)即可。

回退之后可以进行相应的修改并重新提交到远端。

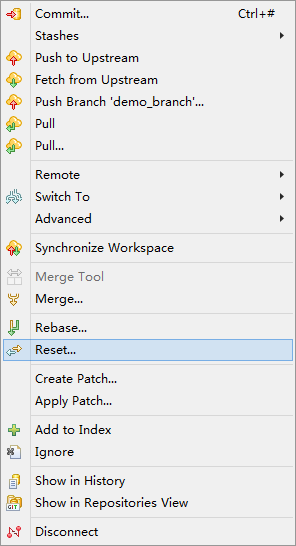
备份的作用在于回退错误或失败之后进行对照并还原。

* **关于作用域（即在哪个文件夹上进行操作）**



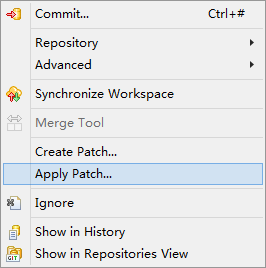
1. 项目目录操作

team面板：



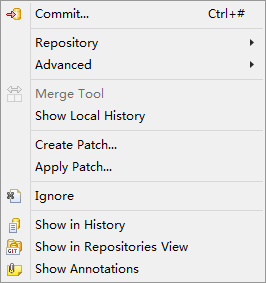
1. 包目录操作

team面板：



1. 对应文件操作

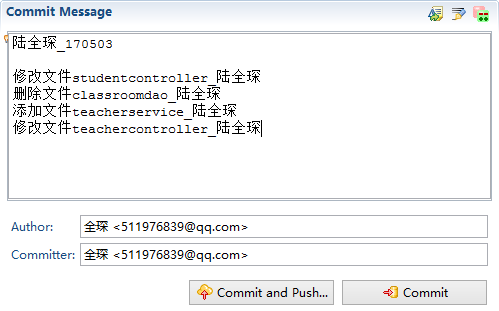
team面板：



建议

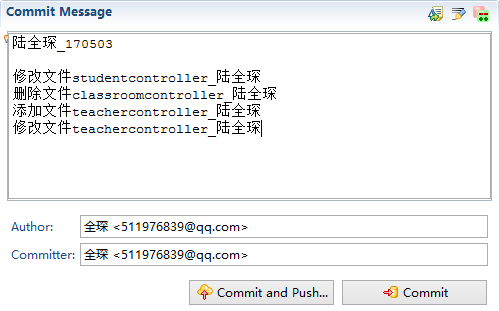
1. 确定要添加的文件在对应文件上操作。
2. 回退版本在哪个目录上操作都是一样的
3. 当然你动不动有些修改就提交更新到远端是非常繁琐的，有些人往往会在结束工作前把所有的更改一起提交更新，但是这种方式写commit massage的时候往往会很多而且不易描述

类似于：



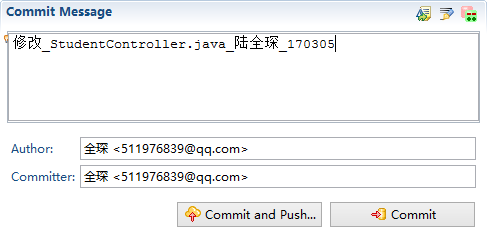
所以我推荐方式是不同项目下的文件分批次提交

类似于：

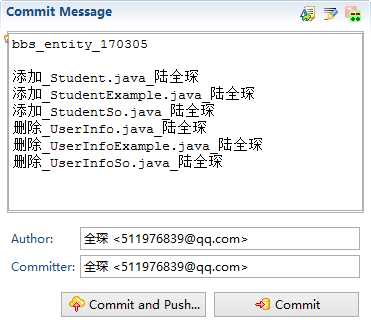


* **关于Commit message的书写格式规范的建议**

1. 单文件提交时：操作类型\_文件名称\_操作人\_操作时间



1. 多文件提交时：
   1. 第一行：操作模块\_操作时间。第一行是标题，第三行以后是内容，中间要空出一行供提交的时候区分。
   2. 第三行：操作类型\_文件名称\_操作人



* **几个原则**

1. 工作前一般要先[更新](#更新)再工作，工作后一定要先提交再下班。
2. 尽量避免[回退](#回退)操作。
3. 只提交要更新的文件到staged Changes其他文件千万不要放进去。
4. 积极[本地备份](#本地备份)，保质不保量，时常清理那些没用的备份。