**1. 安装**

**2. 创建版本库 git init**

切换到工程目录下，使用 git init，如果有.git的文件夹，说明成功

**3. 提交代码，回退版本，撤销修改，删除代码，对比文件**

**提交：**

将代码存放到暂存区：git add code.txt

将代码提交到当前分支：git commit -m '这是第一个版本'

必须先add再commit，add后再修改的代码不会被提交

**查看版本历史：**

git log

**回退： 回退后git log将查看不到之后的版本号，可以使用git reflog查看**

回退到前一个版本：git reset --hard HEAD~1

或者指定版本号：git reset --hard 版本号 版本号是指git log里面commit后面的值

**查看哪些代码有修改：**

git status

红色表示未添加到暂存区，绿色表示已添加到暂存区

**撤销修改：**

撤销本地工作区代码修改（其实就是重新检出）：git checkout -- code.txt

已添加到暂存区撤销：git reset HEAD code.txt

已提交到版本库撤销：直接回退版本

**代码对比：**

本地与当前版本对比：git diff HEAD -- code.txt

对比两个版本的不同：git diff HEAD HEAD^ -- code.txt

**删除代码：**

先把本地的文件删除：git rm code.txt

再提交：git commit -m '代码没用了，删除'

**4. 分支管理**

git初始化以后就是默认有一个master分支

**查看当前分支：**

git branch

**创建分支：**

git branch dev

**切换分支：**

git checkout dev

切换到dev分支后，后续对代码的操作都是在dev分支上

**合并dev分支到master：**

先切换到master，git checkout master

合并dev分支：git merge dev

**删除分支：**

git branch -d dev

**创建+切换：**

git checkout -b dev

**解决冲突：**

如果两个分支同时修改了同一个代码，就会产生冲突

没有关系，先git merge dev，然后再手动解决冲突，最后git add code.txt，git commit -m '解决'

**查看分支信息：**

git log --pretty=oneline --graph

**禁用Fast forward模式合并分支：**

git merge --no-ff -m '禁用快进模式合并分支' dev

**BUG分支处理：**

如果当前分支代码未完成，且不能提交，但是需要处理一个BUG，再回来继续修改代码，可以使用：

（1）储存工作现场：git stash

（2）切换到master分支，创建临时分支git checkout -b bug001

（3）修复bug

（4）切换到master，然后合并分支

（5）删除bug001分支

（6）返回dev分支，恢复工作现场：

git checkout dev git stash list git stash pop

**5. github使用：**

**克隆项目：**

git clone git@github.com:smart/text.git

如果克隆出错，eval "$(ssh-agent -s)" ssh-add

**推送分支，将本地分支推送到远程库中，如果远程库不存在该分支会自动创建：**

git push origin dev

**本地分支跟踪服务器分支：**

git branch --set-upstream-to=origin/远程分支名称 本地分支名称

例：git branch --set-upstream-to=origin/dev dev

**从远程库拉取分支并合并到本地分支：**

git pull origin dev

**mysql**

首先删除Mysql服务，接着新建一份my.ini文件，最后使用命令行重新初始化mysql服务，同时指定新建的my.ini作为服务默认的配置文件。以下是详细步骤：

sc delete MySql

新建my.ini文件

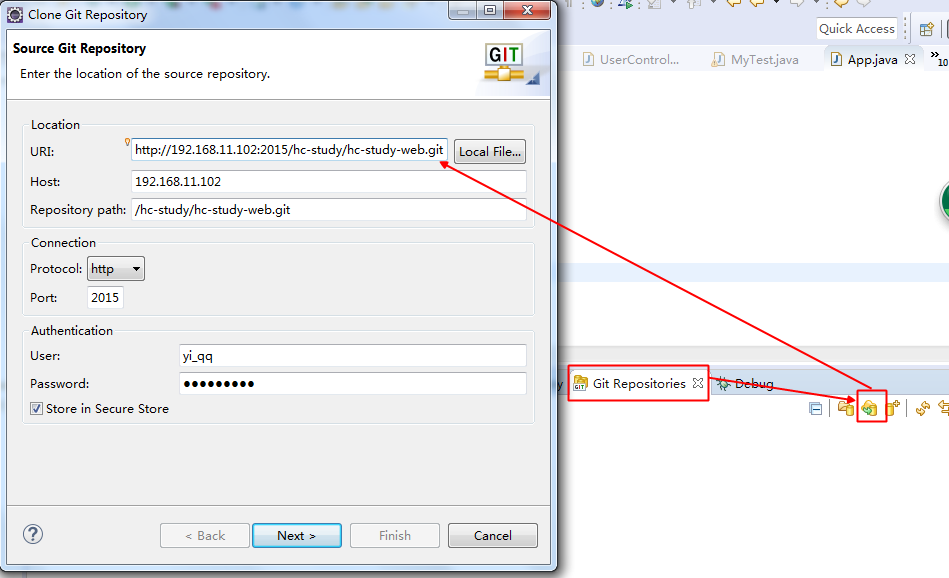
mysqld --initialize

mysqld --install MySQL --defaults-file="D:\software\mysql-5.7.25-winx64\my.ini“

eclipse git插件使用

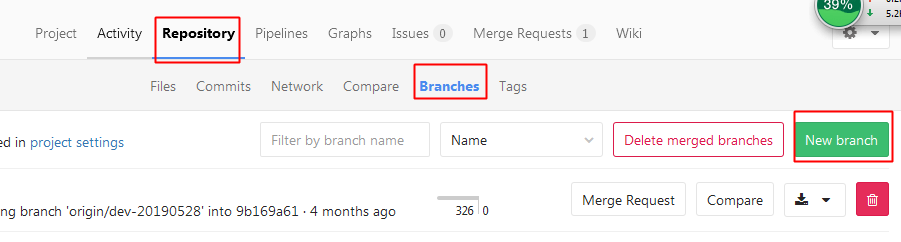
整体步骤：克隆master代码-->到服务器新建自己分支b-yiqq-->eclipse里面fetch，然后切换到b-yiqq-->开始写代码-->写完后pull自己分支b-yiqq(其他人可能也在用此分支)，切换到master并pull最新代码-->切换到b-yiqq并merge本地的master-->解决冲突-->push到自己分支b-yiqq

克隆代码

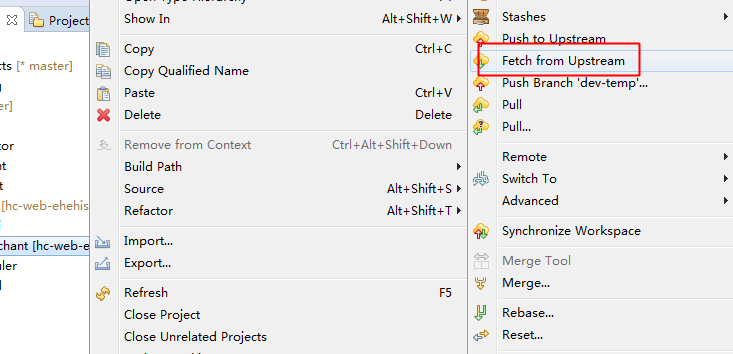


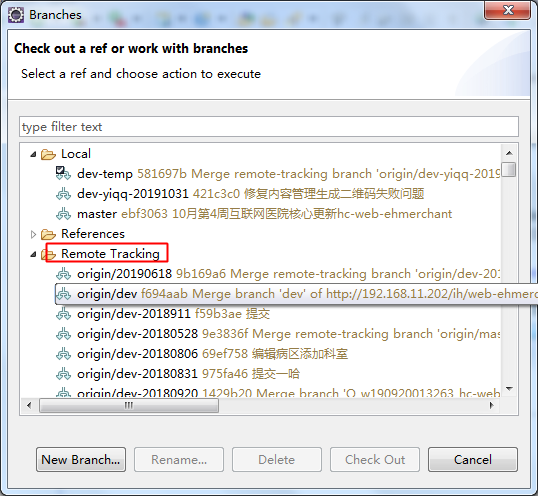
创建分支

一般先到git服务器新建分支

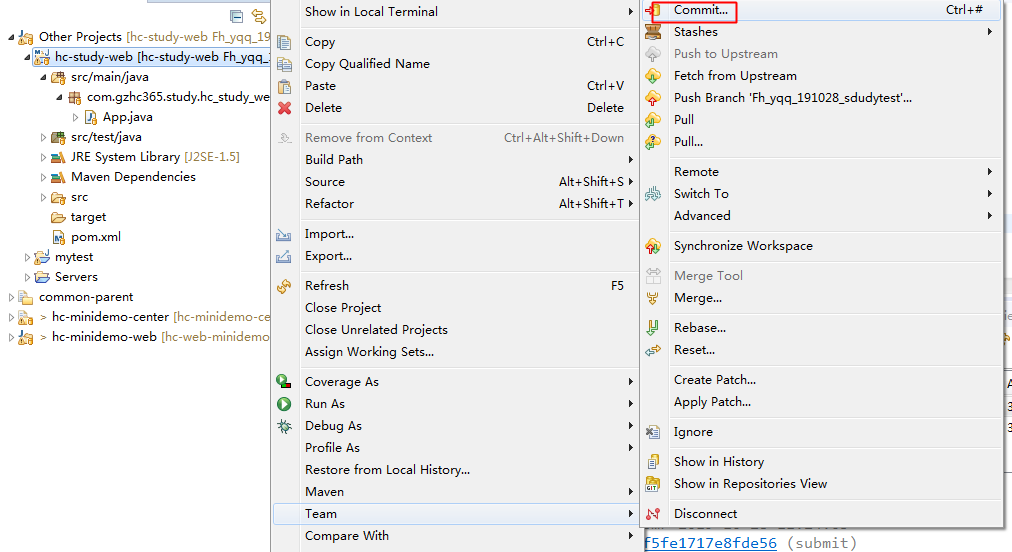


本地切换到刚刚新建的分支，需要先fetch

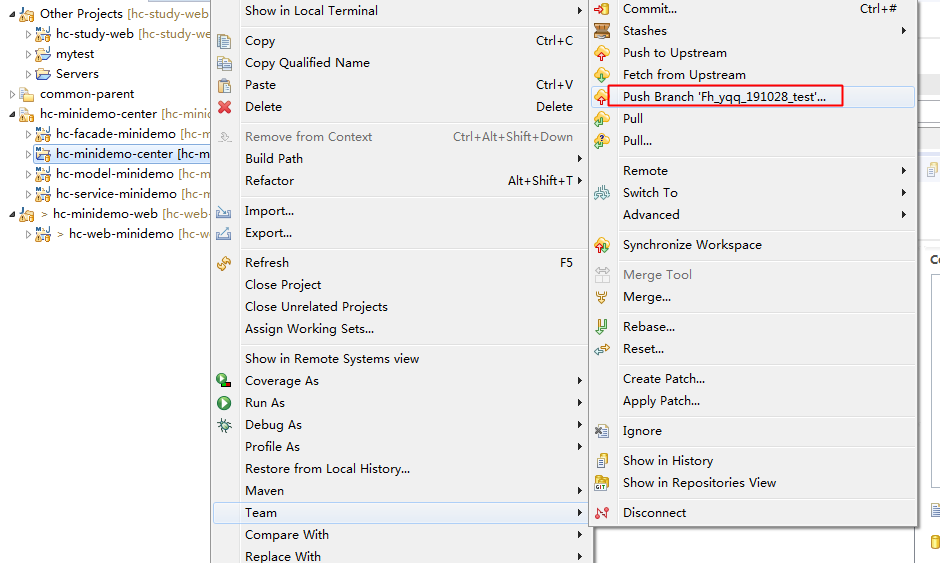




提交代码，如果有新增的代码记得add index，也可以先点synchronise workspace对比下代码



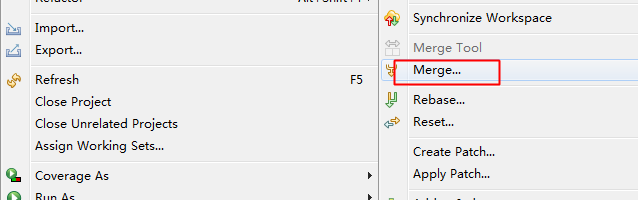
push到自己的分支

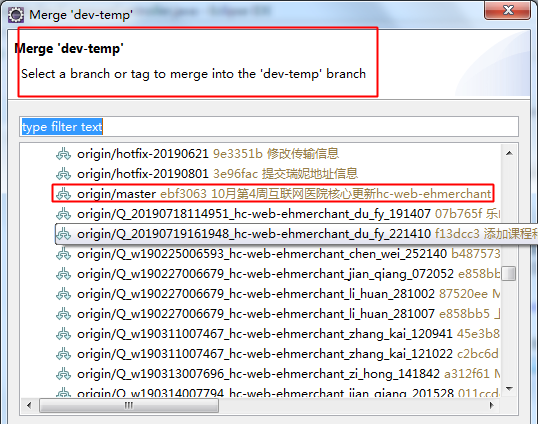


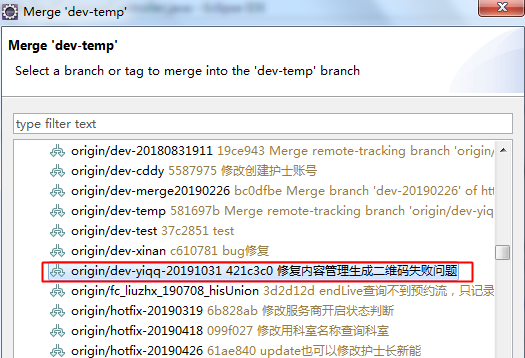
合并分支

在自己分支上merge下master分支，然后再dev-temp上merge自己分支，或者

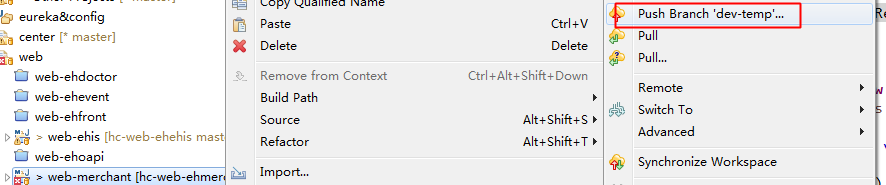
切换到dev-temp，先把master分支merge下来，再把自己的分支merge下来





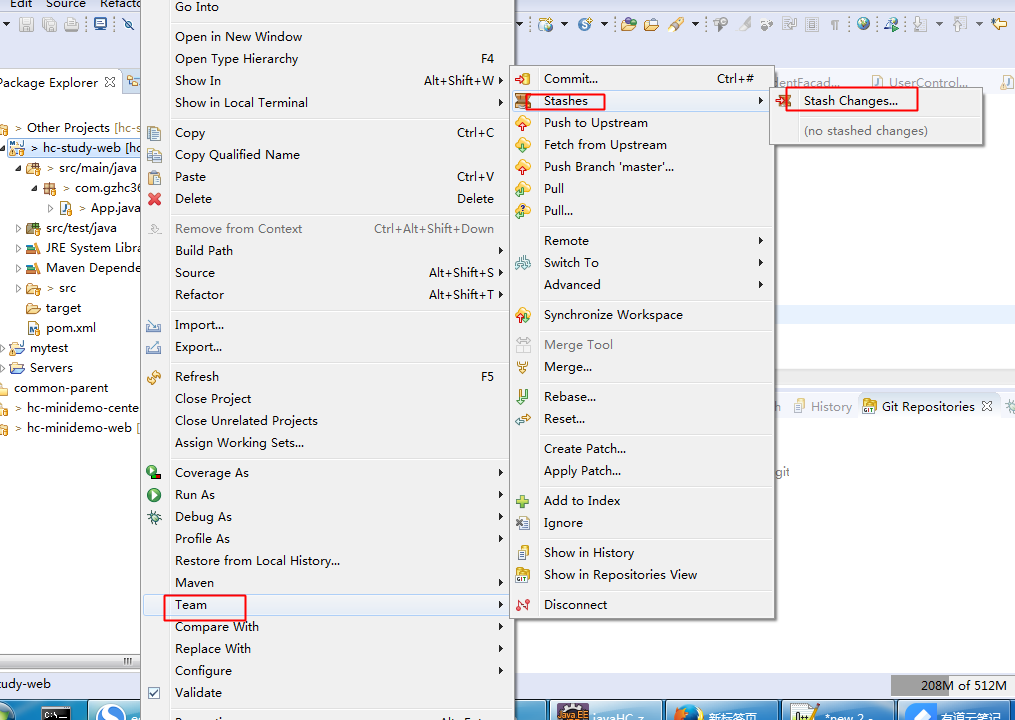


push代码

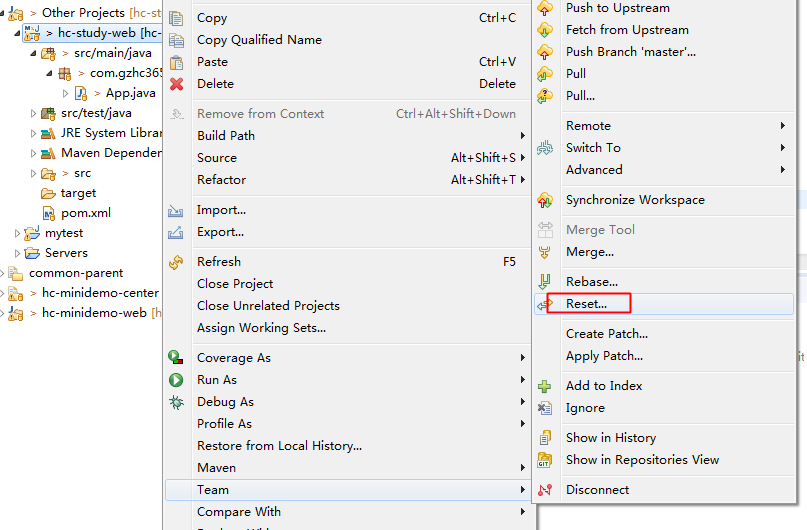


代码写错分支了怎么办

还没有提交：先stash切换分支后在pop

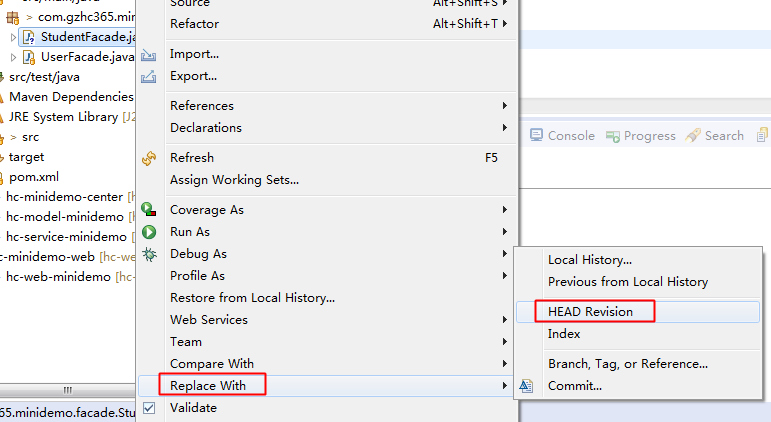


如果已经commit了：先reset，选择mix，会清空暂存区，然后再stash，pop



如果已经push到远程仓库了，那就只能git revert,然后切换分支重新提交

代码撤销修改



git状态图标

