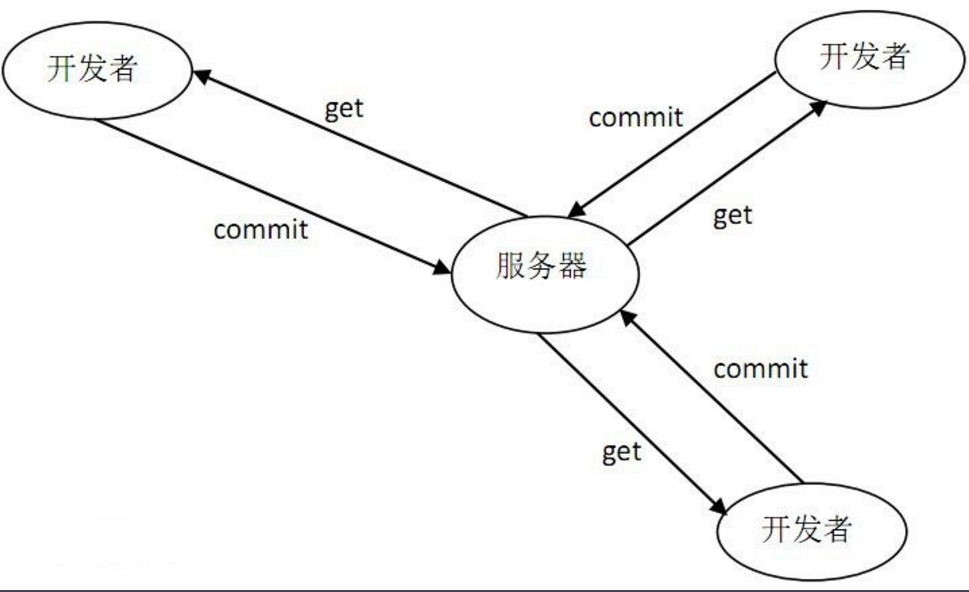
# SVN

## SVN概述

SVN是**Subversion**的简称，是一个开放**源代码的版本控制系统**，相较于RCS、CVS，它采用了分支管理系统，它的设计目标就是取代CVS。互联网上很多版本控制服务已从CVS迁移到Subversion。说得简单一点SVN就是用于多个人共同开发同一个项目

CVS 即 Concurrent Versions SystemCVS 即 Concurrent Versions System

## 1.2 SVN工作原理



**准备：**

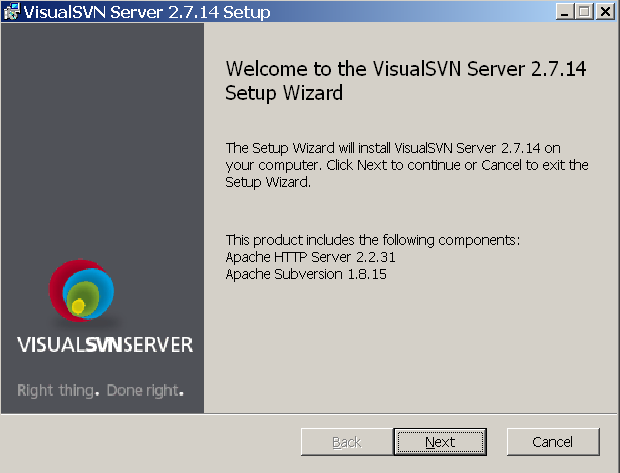
**在桌面创建zhangsan和lisi两个文件夹，内部都放着各自的eclipse和workspace**

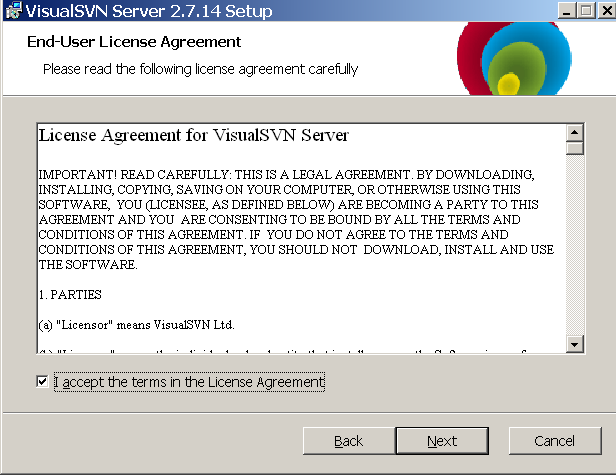
**安装SVN服务端程序**

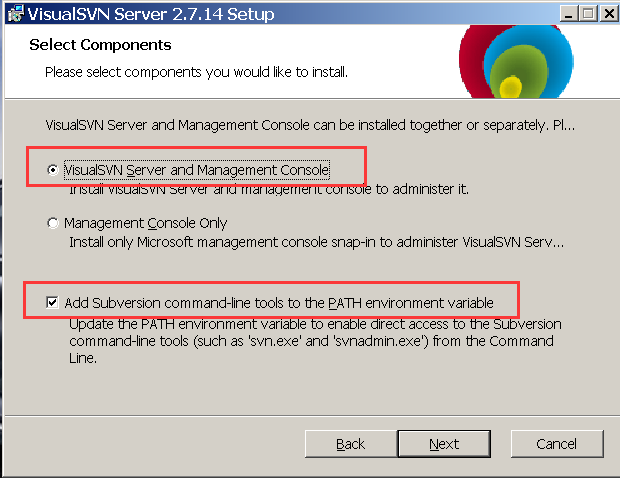
## 1.3 SVN服务器安装

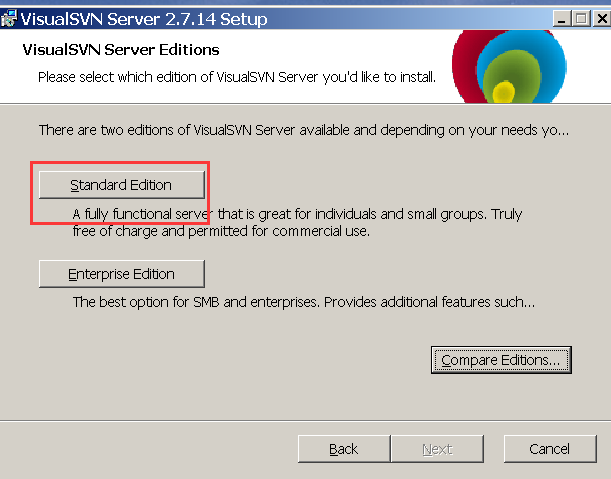


双击，下一步下一步安装即可

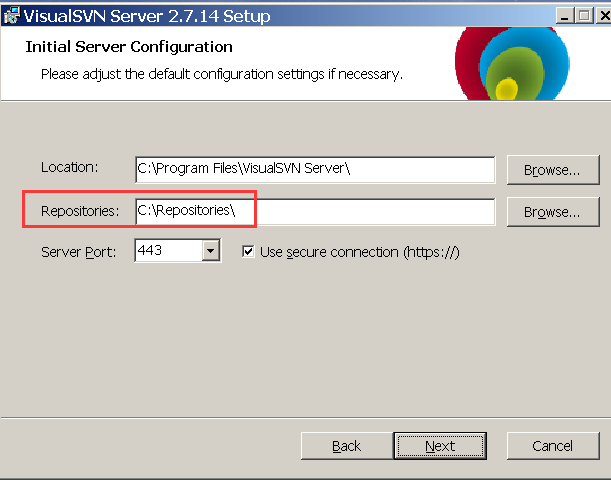


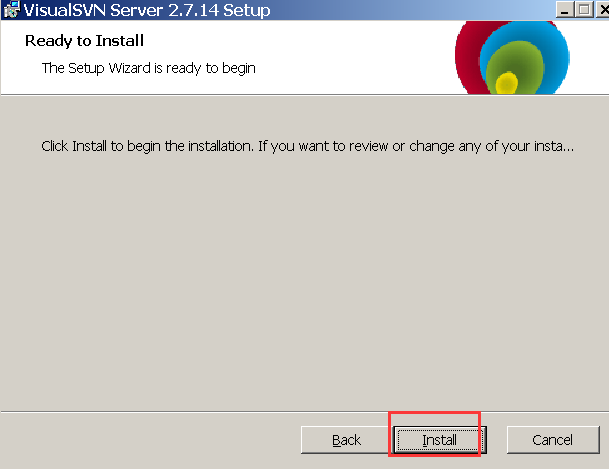


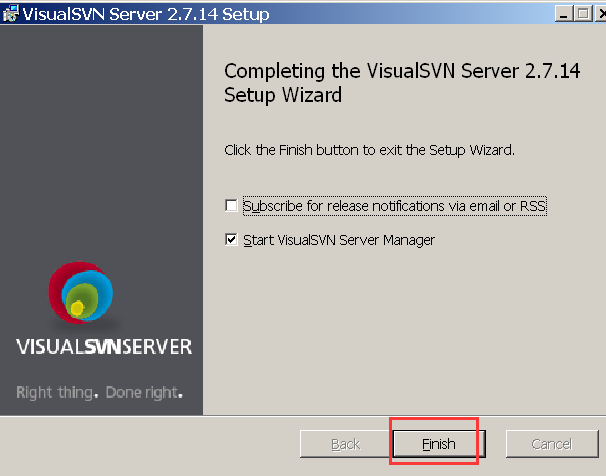


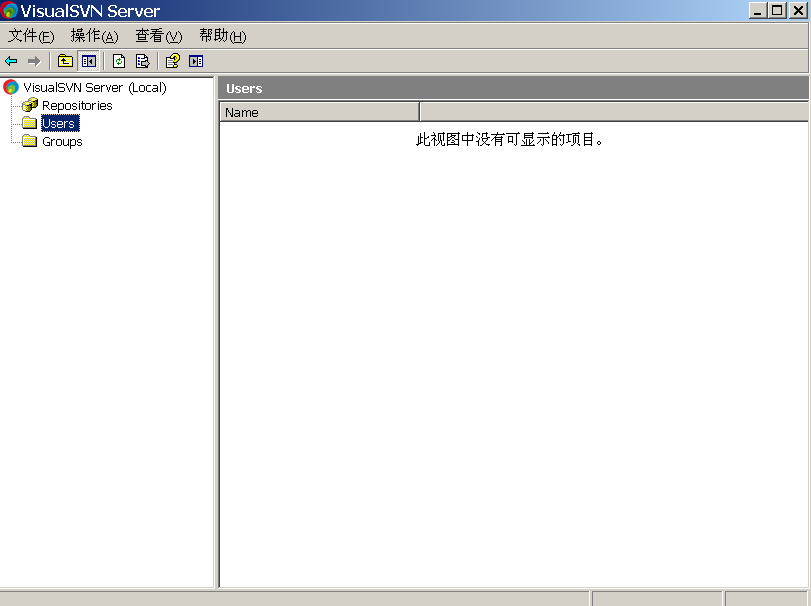


* 下面红色的是服务器代码仓库，下面的https不要选







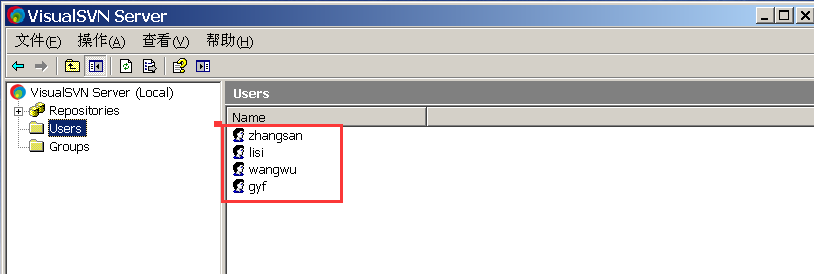


## 1.4 SVN帐户配置

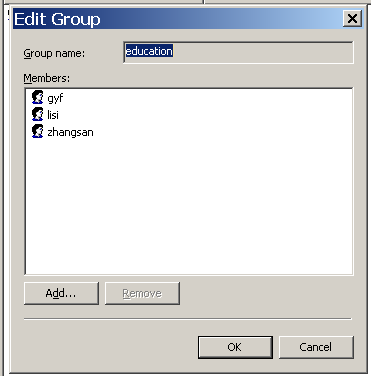
打开VisualSVN Servler



创建账户,创建zhangsan,lisi,wangwu,gyf 4个帐户



添加成员项目到组【创建一个**教育组**-成员有zhangsan,lisi,wangwu】

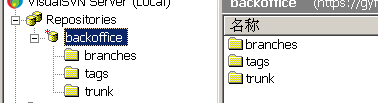


## 1.5 SVN创建代码目录结构

trunk为主开发目录 =trunk放正在开发的版本

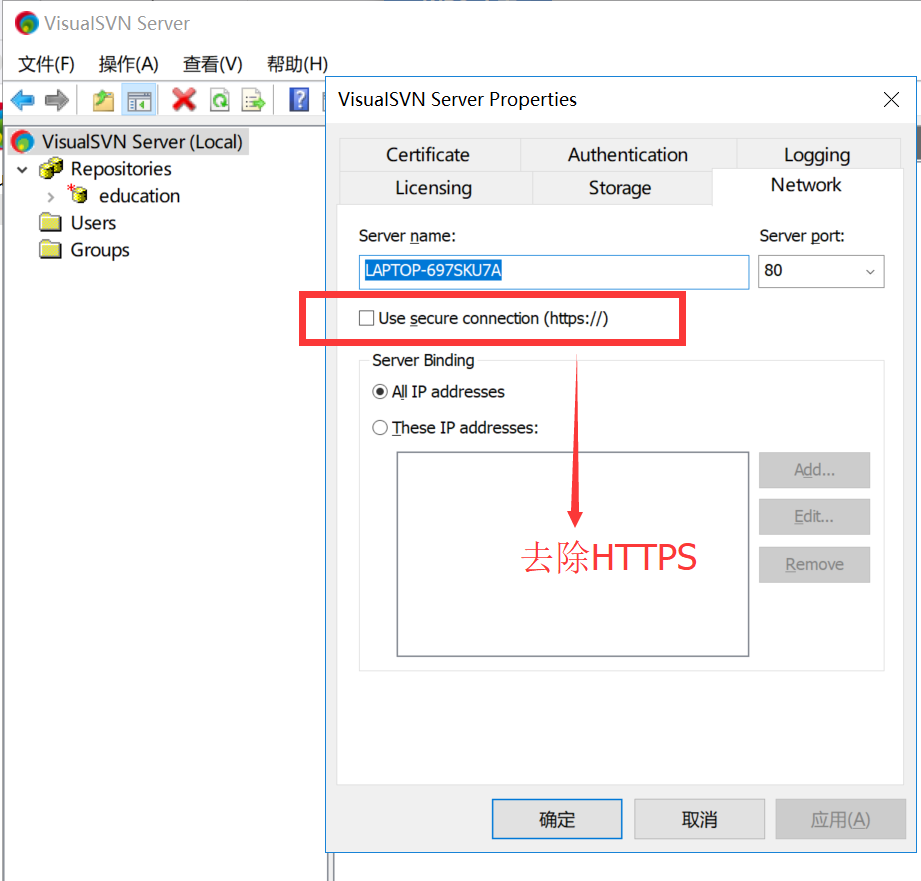
branches为分支开发目录 =branches放以前发布的版本

tags为tag存档目录（不允许修改）



## 1.6 去除HTTPS

点击Server右键属性



## 1.7 在浏览器访问truck目录

访问路径：http://192.168.1.128/svn/education/trunk

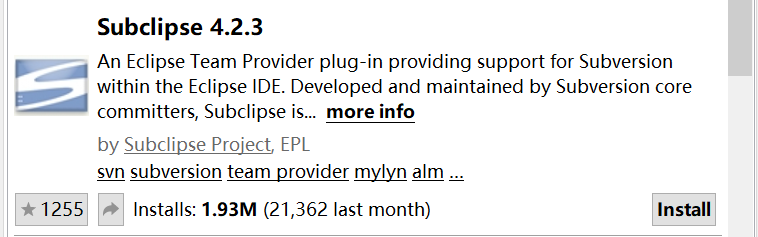
注意IP地址改成你本机的IP地址

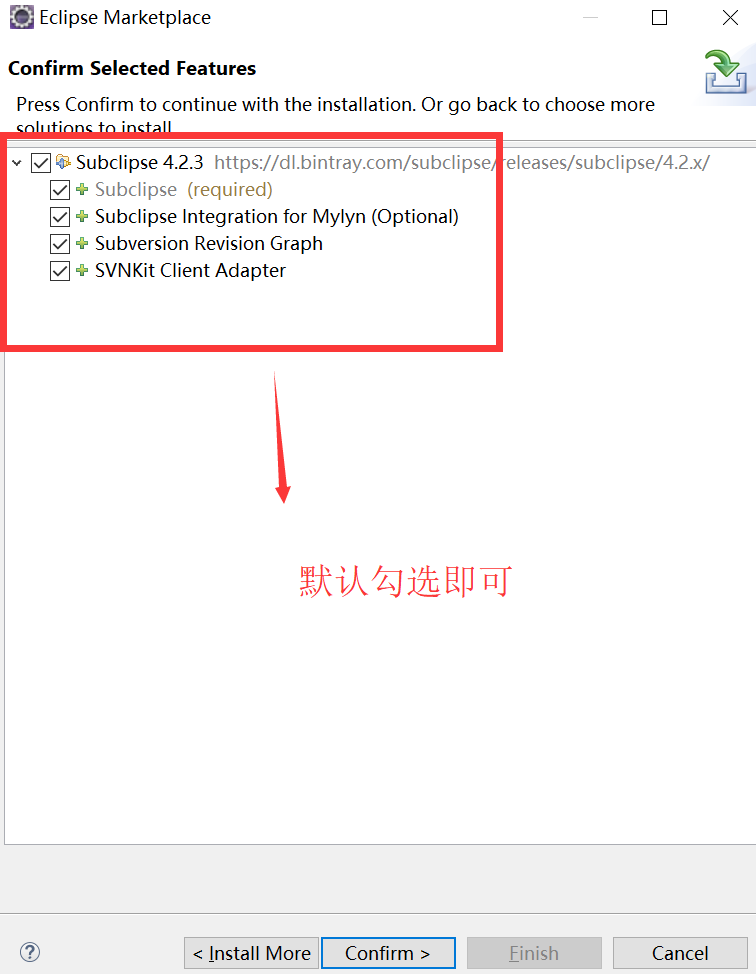


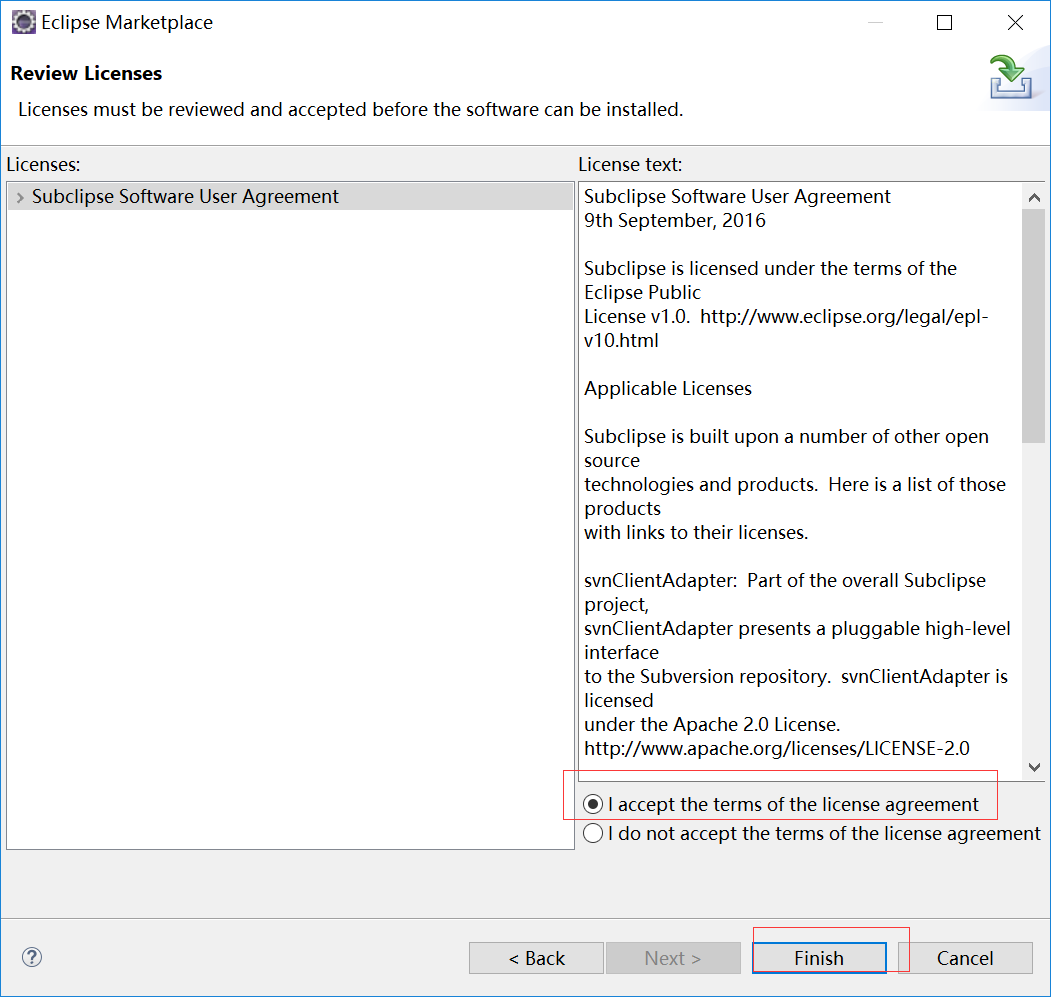
# 2 SVN开发流程使用

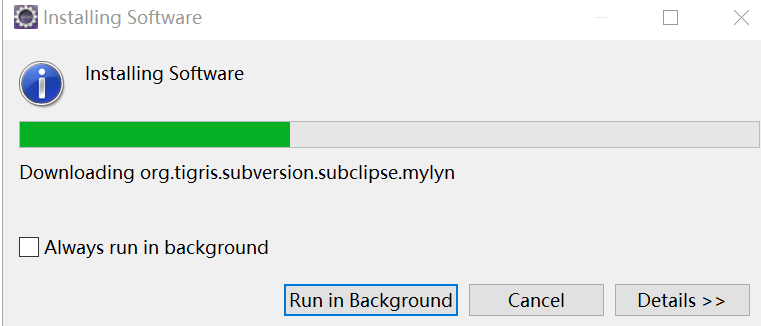
## 2.1 安装subclipse

* 在Eclipse安装个subclipse插件
* 找到Eclipse的应用市场进行搜索subclipse
* 点击Eclipse的Help菜单，可以看到Eclipse Marketplace
* 安装完后重启Eclipse

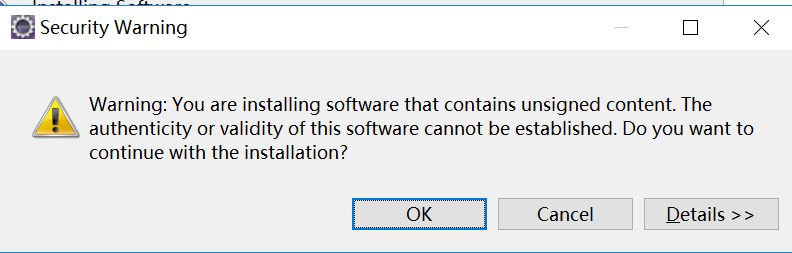




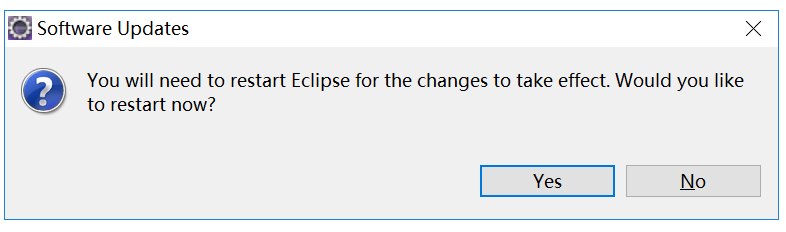




点继续，这里提示有未签名的内容，不用管，继续点OK就行了

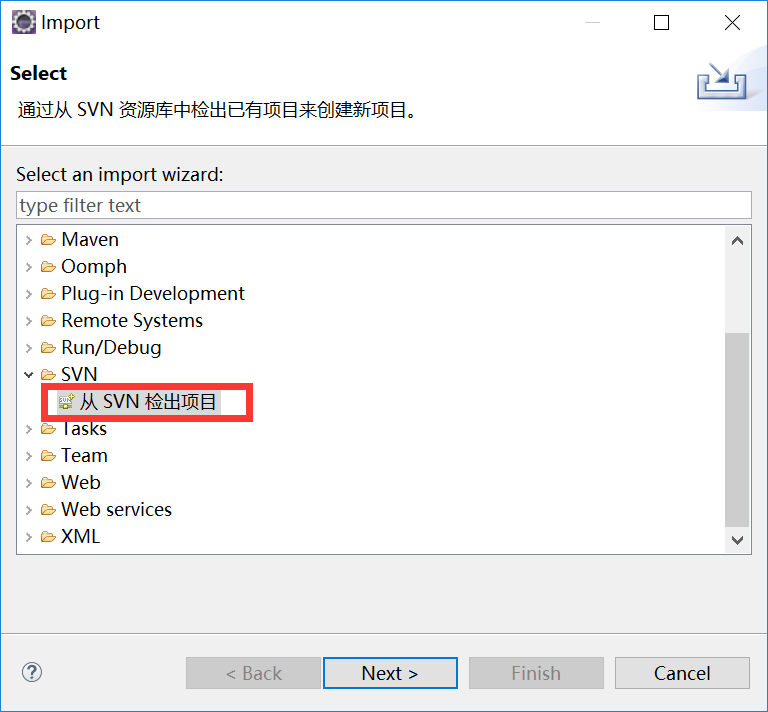


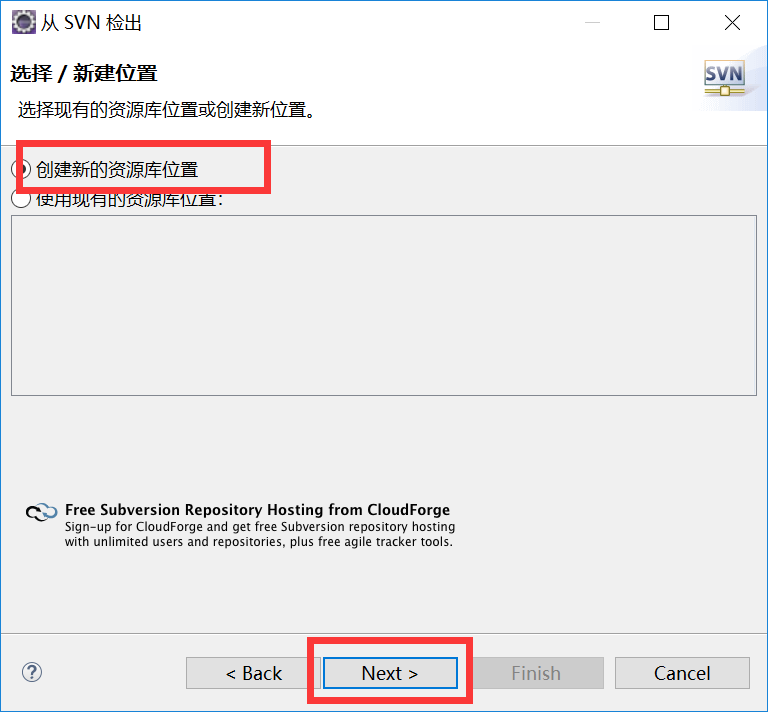
重启

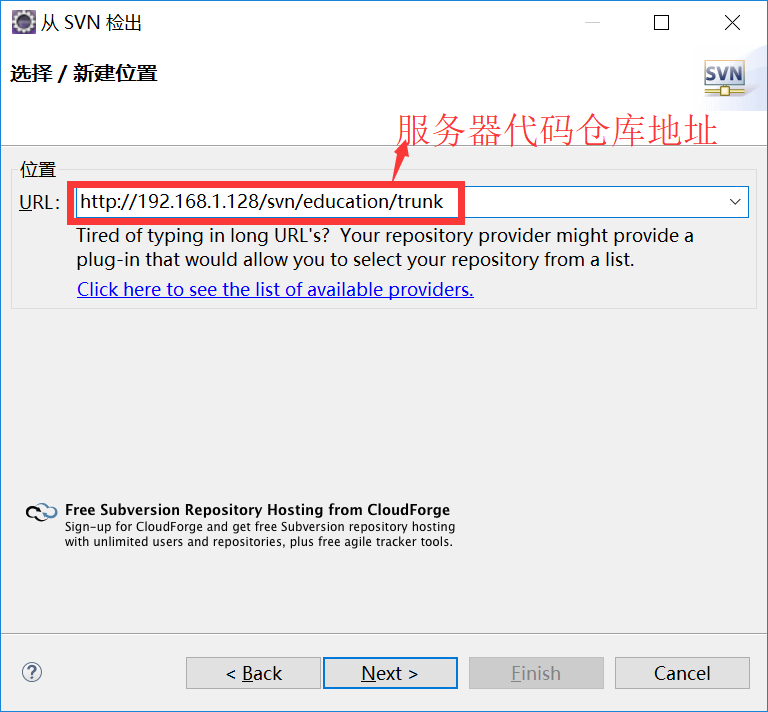


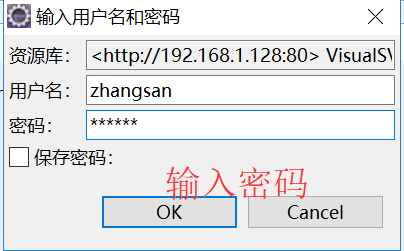
## 2.2 把trunk目录下载到本地来

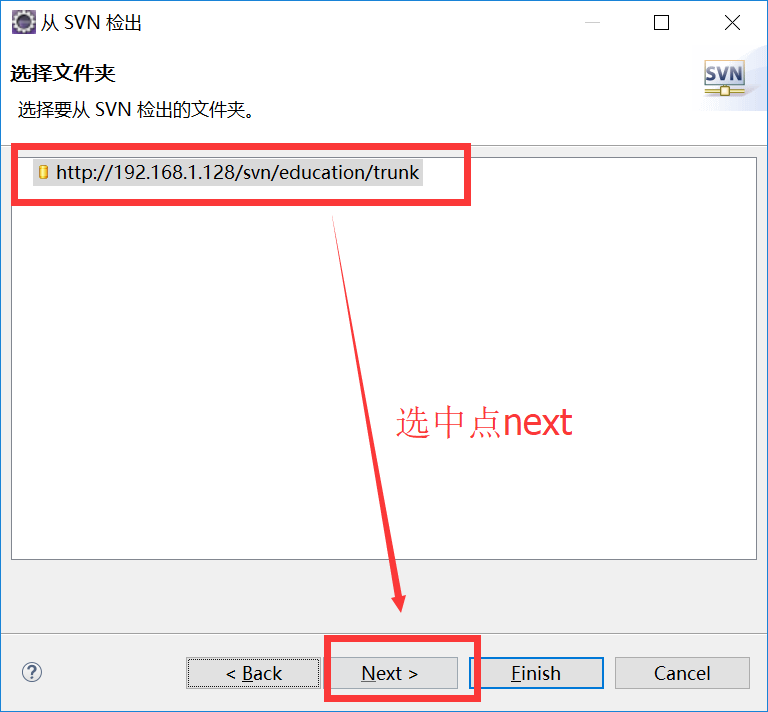
从SVN检出项目

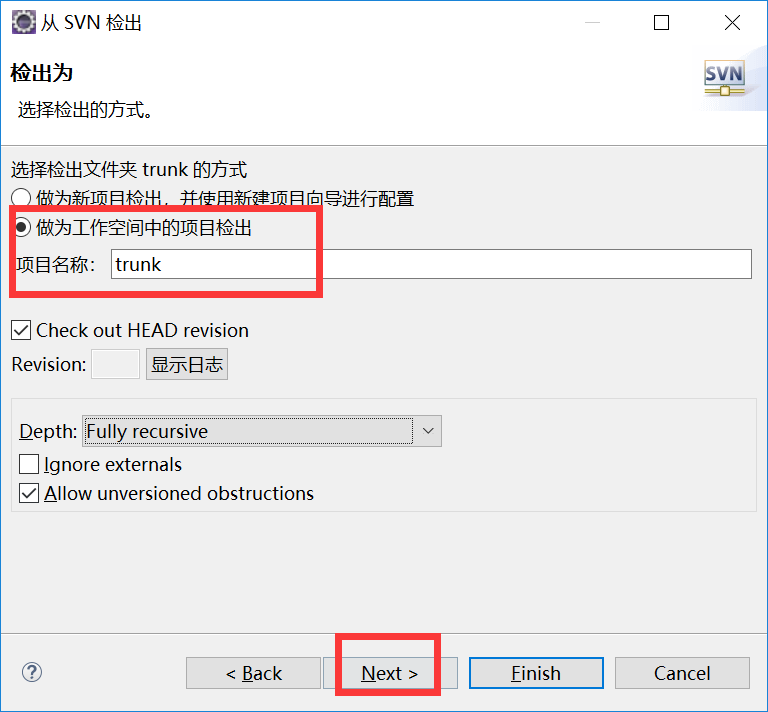


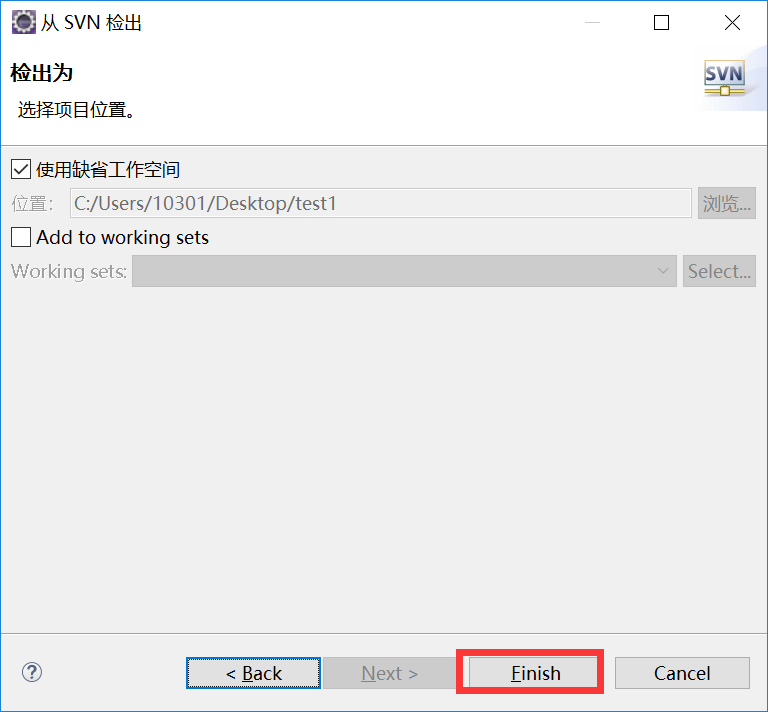


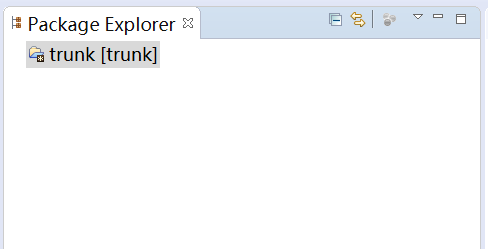






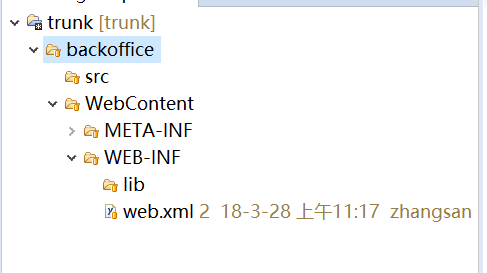




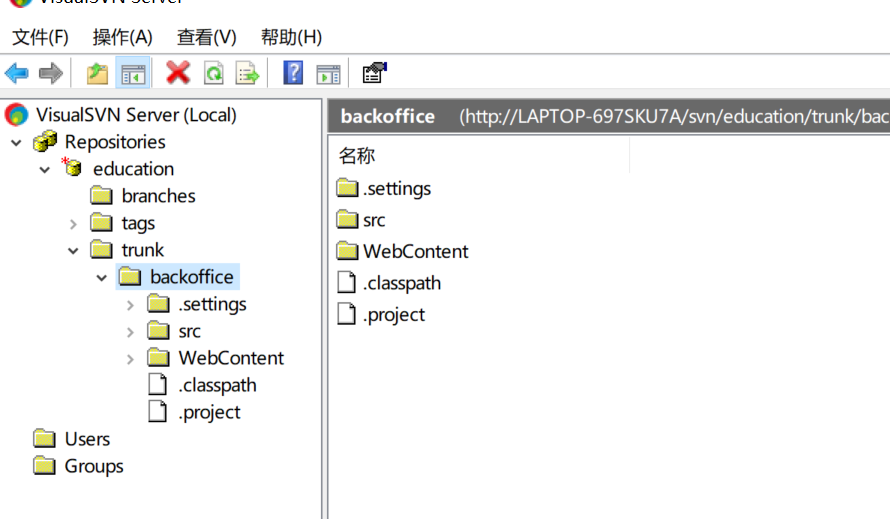


## 2.3 在trunk中放个项目代码

* 创建一个JavaWeb项目放在trunk中，然后提交
* **提交是右键trunk -> Team ->提交**



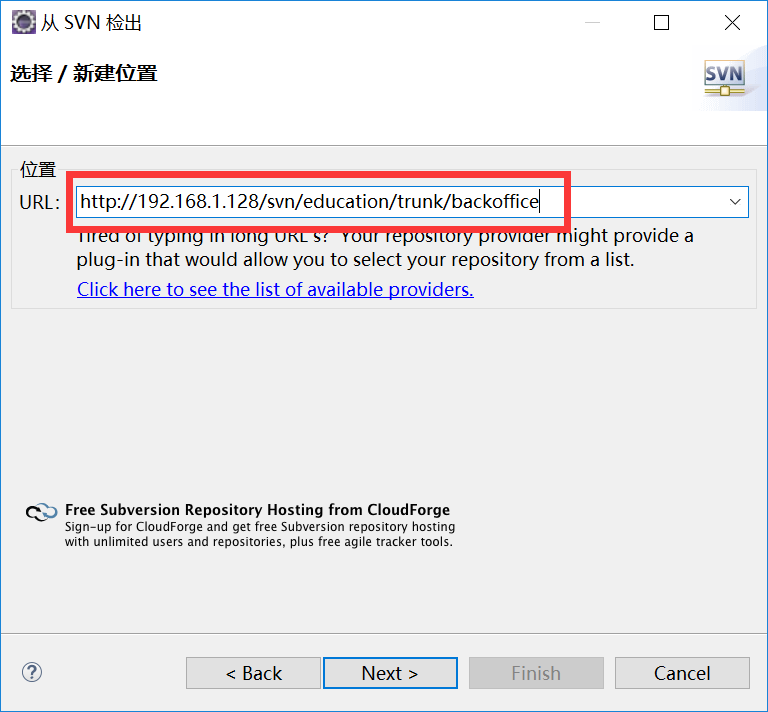
* 提交后在服务器能看到代码

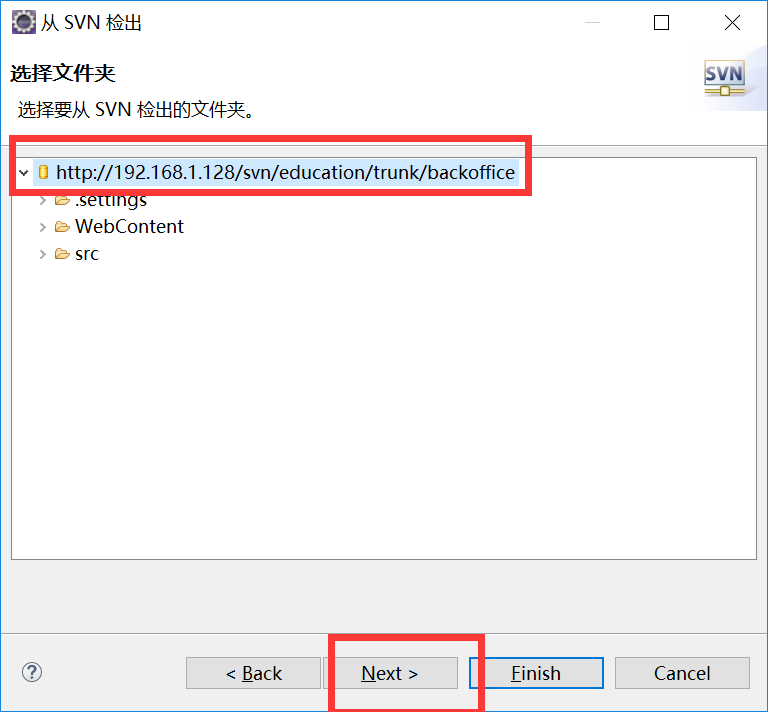


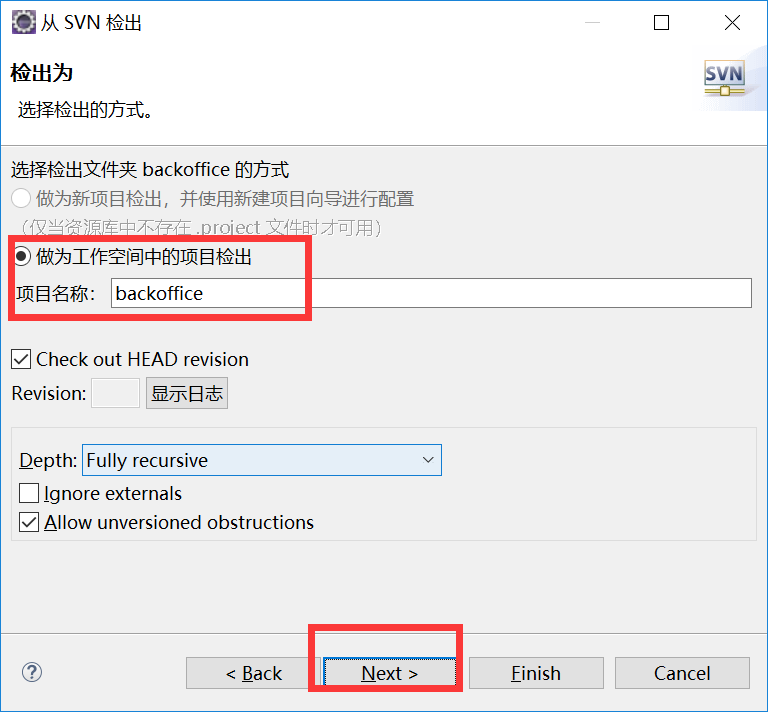
## 2.4 删除eclipse中的trunk 【前面操作-一般是项目负责，经理操作】

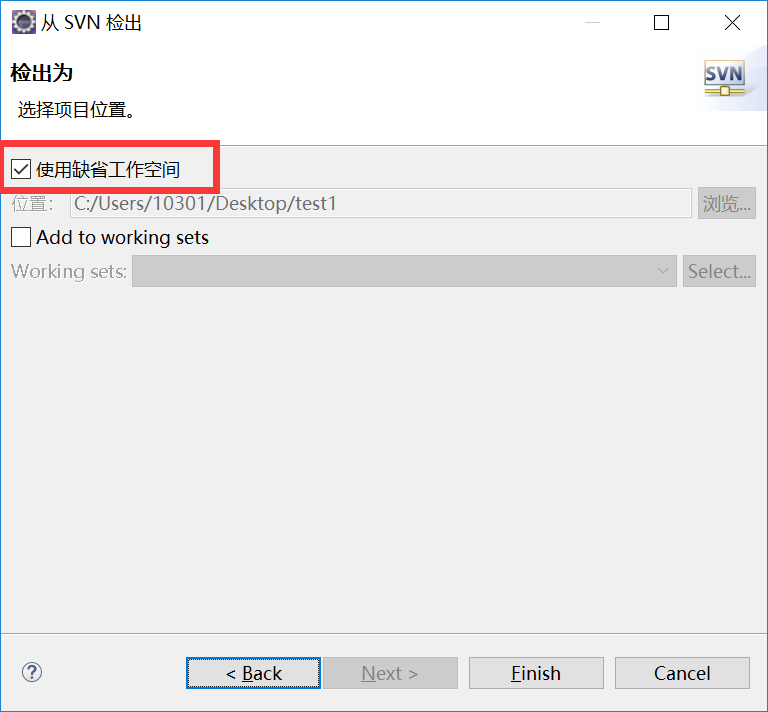
## 2.5 重新从服务器下载项目代码

* http://192.168.1.128/svn/education/trunk/backoffice

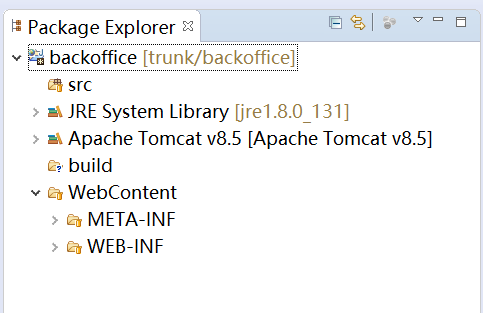






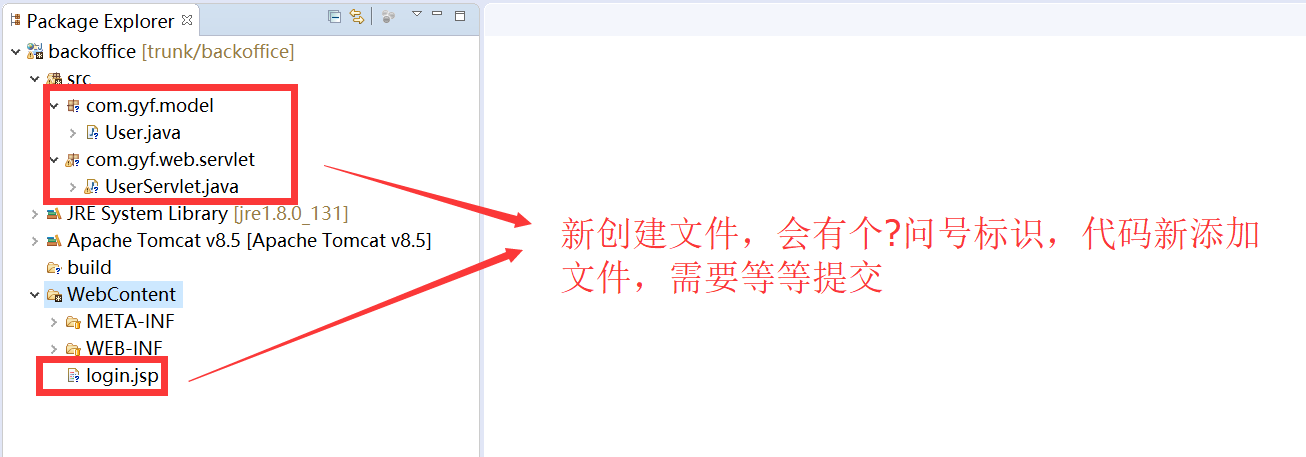


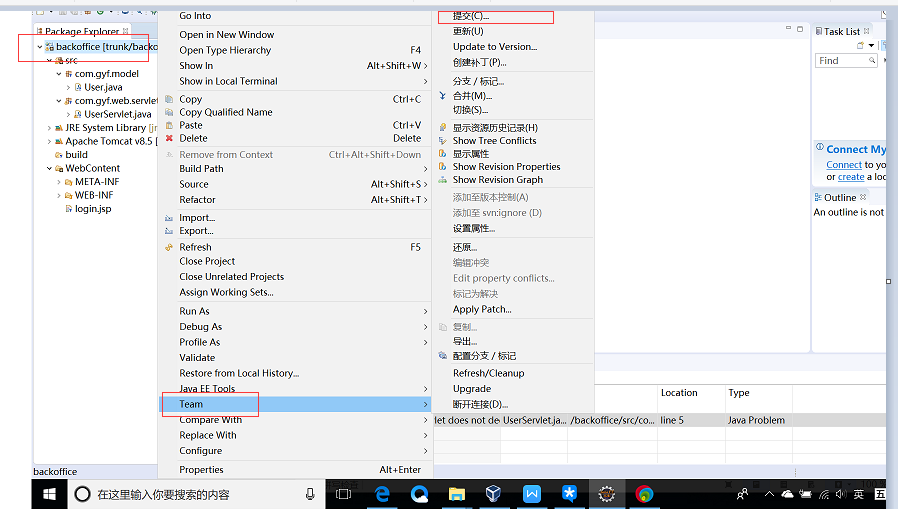
最后效果



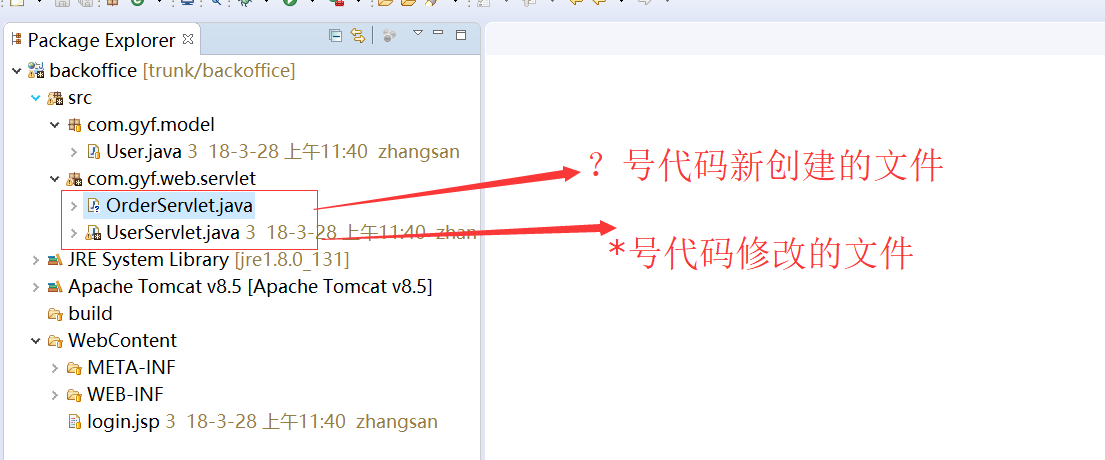
## 2.6 进行项目开发

* 写些代码然后提交到服务器

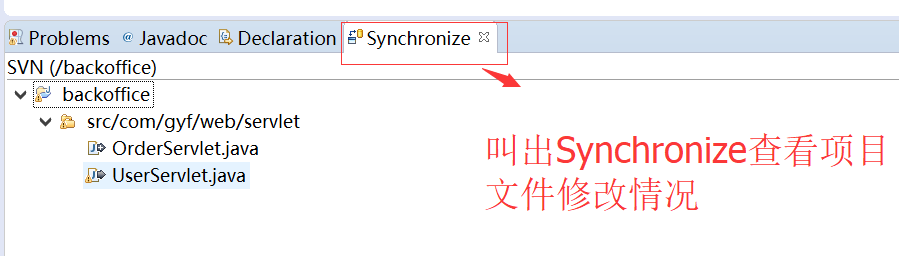




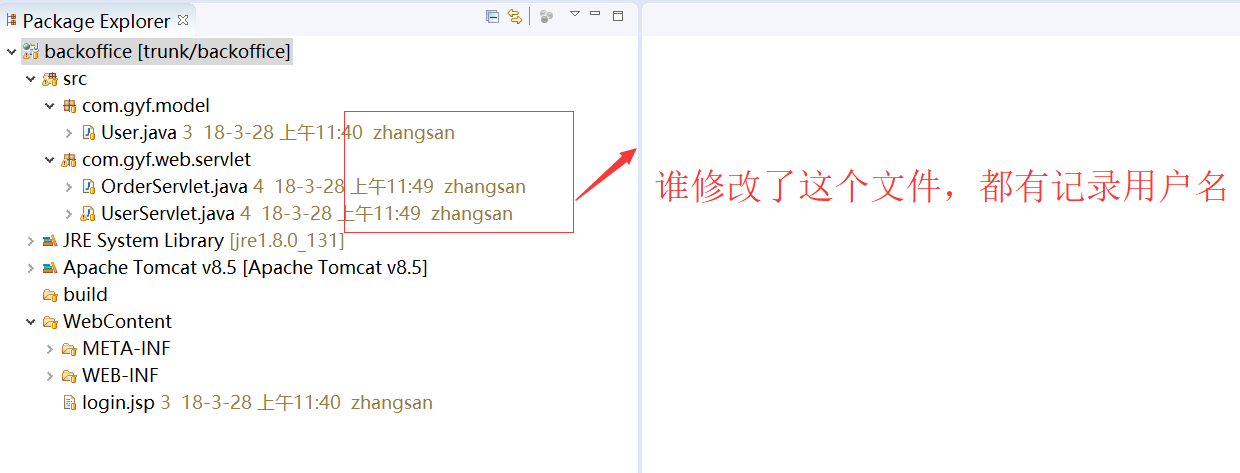
* 修改些代码后提交到服务器



* 叫出Synchronize查看项目情况

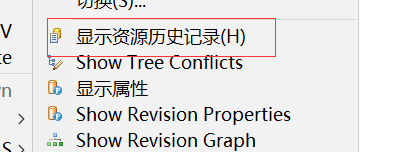


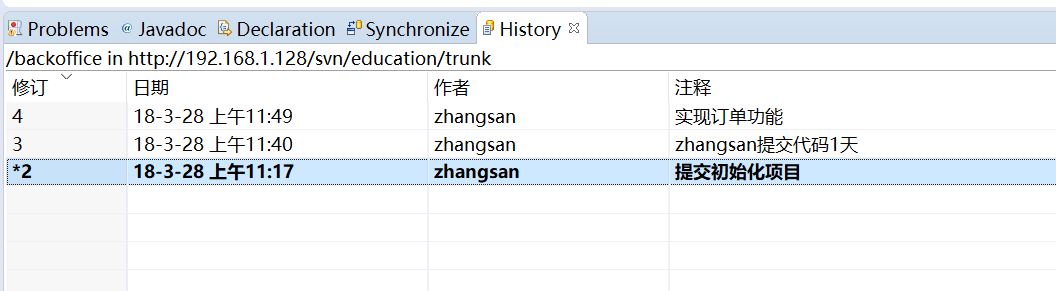
* 2.7 文件修改都查看



## 2.7 显示资源历史记录

右键项目- 查找Team - 显示资源历史记录

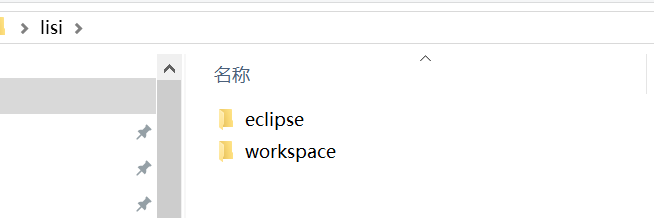




# 3 合作开发

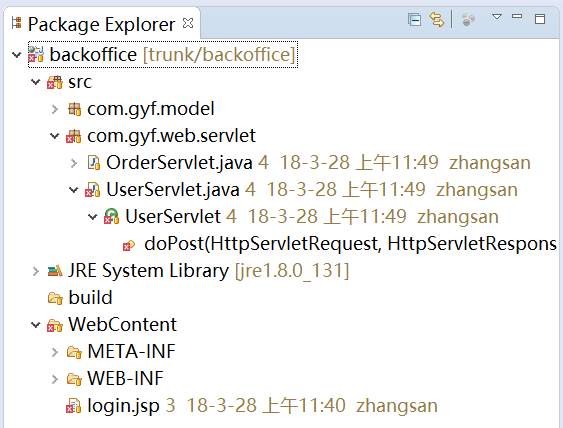
## 3.1 在桌面新创建一个文件夹lisi

* 里面有Eclipse，和workspace
* 在Eclipse中也安装个subclipse

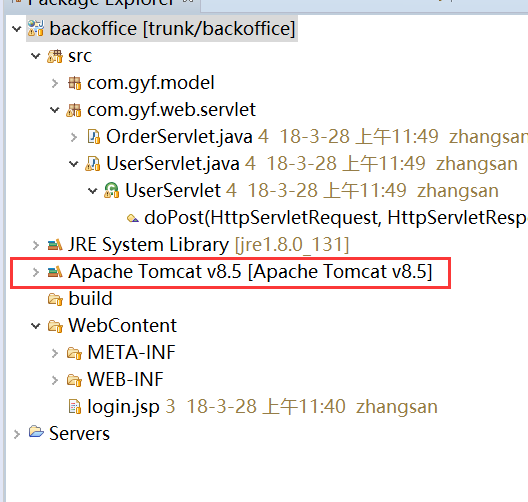


## 3.2 在Eclipse2中以lisi帐号下载【checkout】项目代码

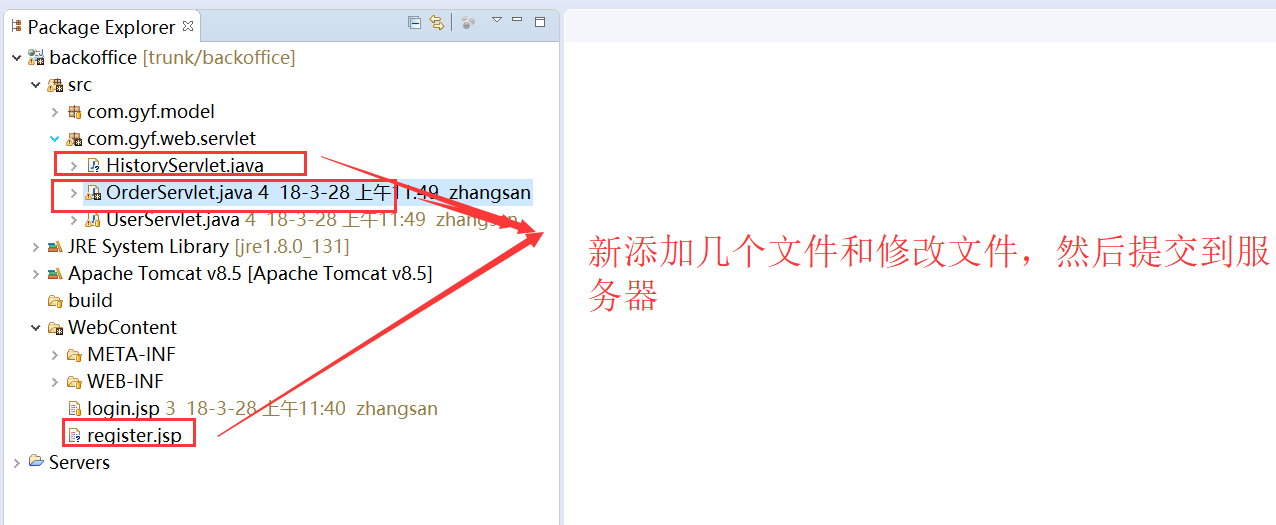
* 导出后会报错，是因为没有Tomcat中的java包

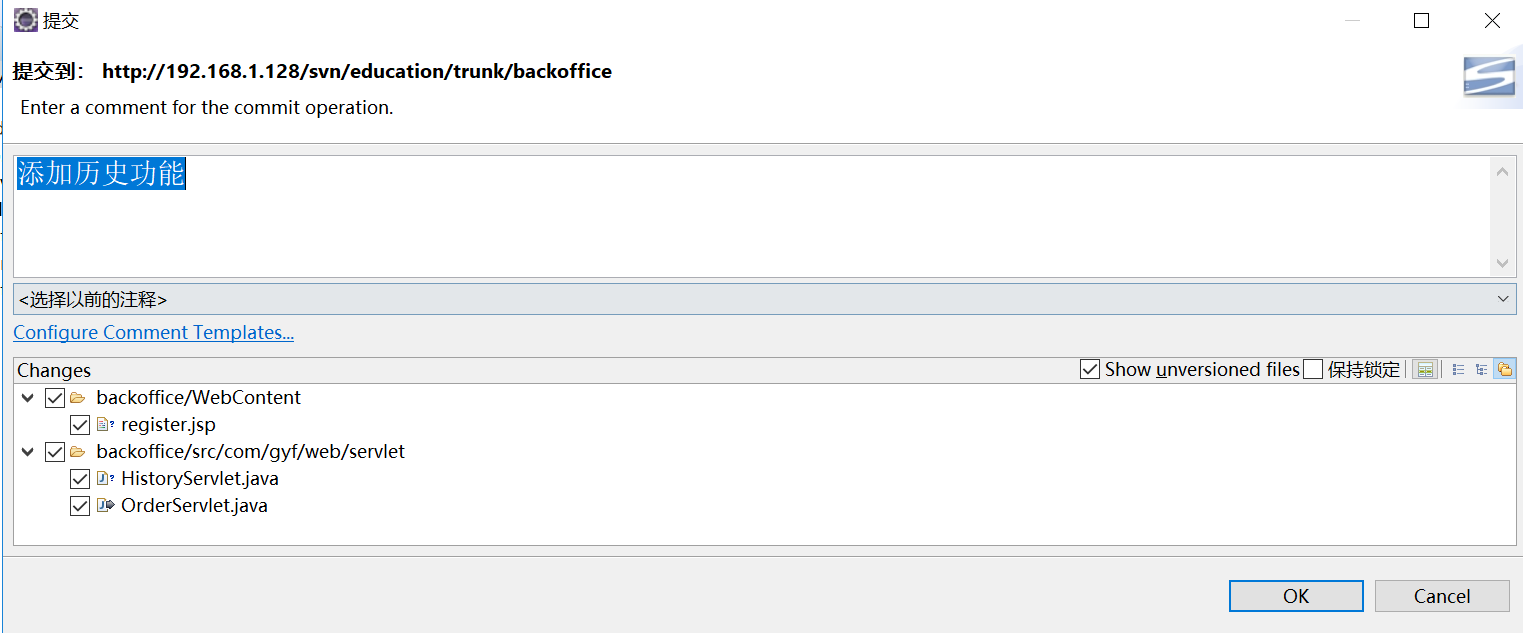


* 然后右键运行项目到tomcat后，会自动添加Tomcat的jar包

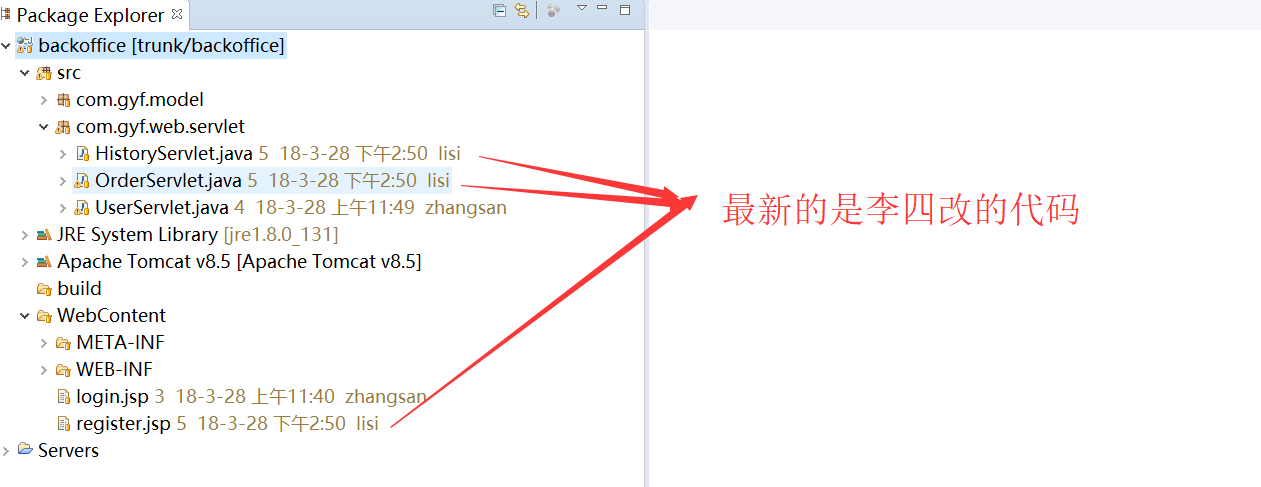


## 3.3 lisi 添加修改文件再提交服务器

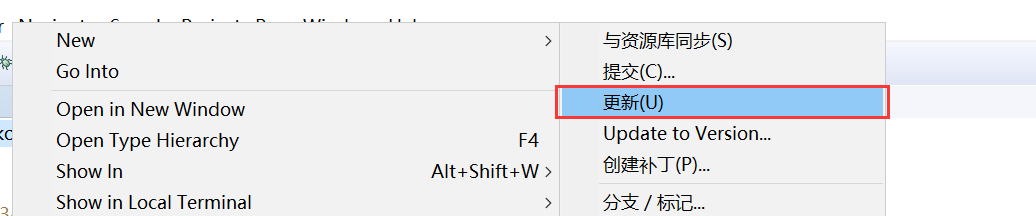


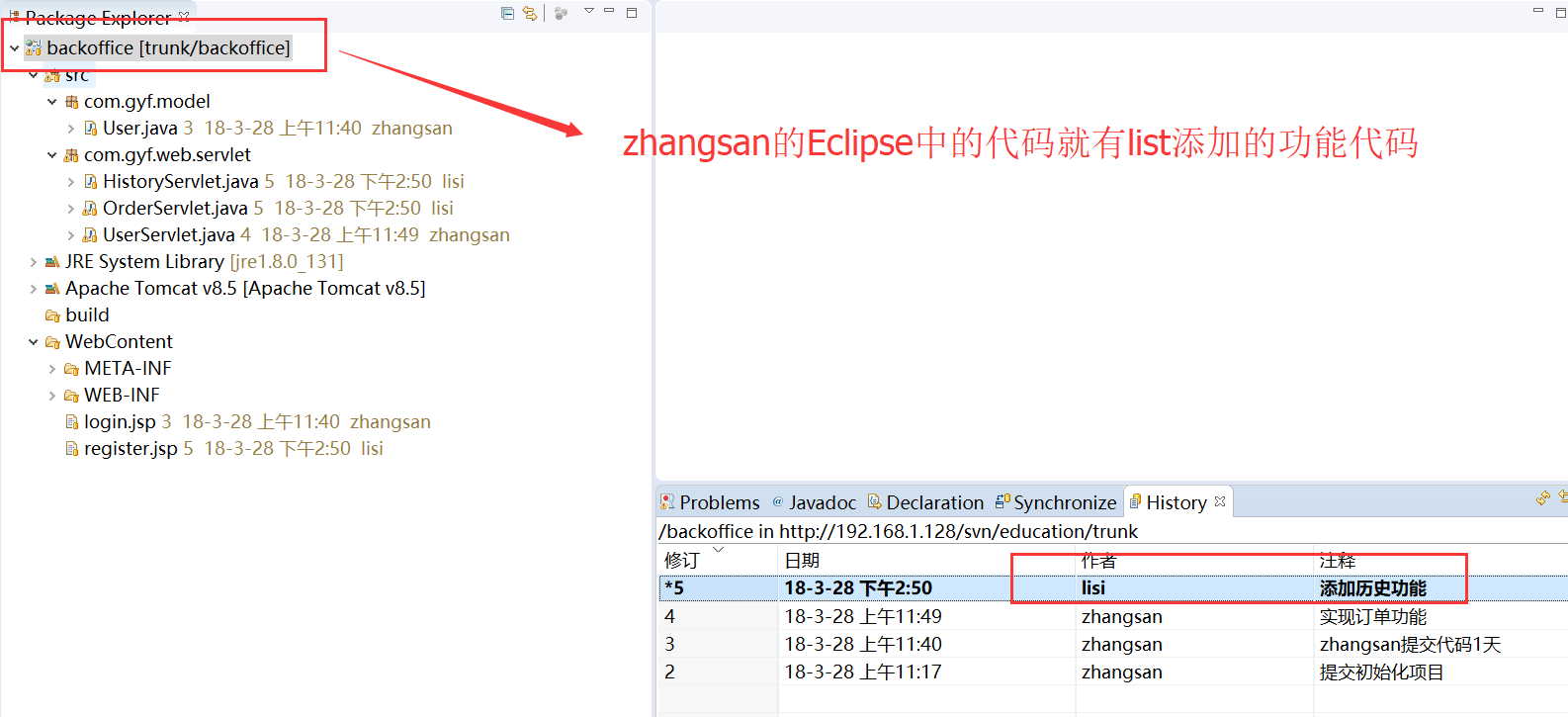


* 提交后就看到下面李四修改代码的情况



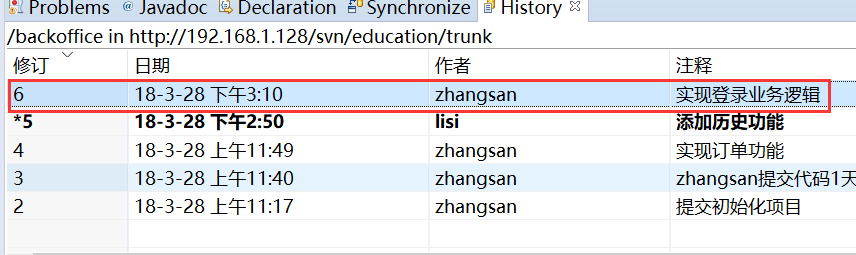
## 3.4 张三从服务器更新最新代码





## 3.5 模拟文件冲突【记住-练习3次】

* 确保张三和李四的代码都是服务器最新的代码
* 张三和李四同时修改某一个文件
* 张三先上传,**【比如张三实现了某个业务逻辑】**

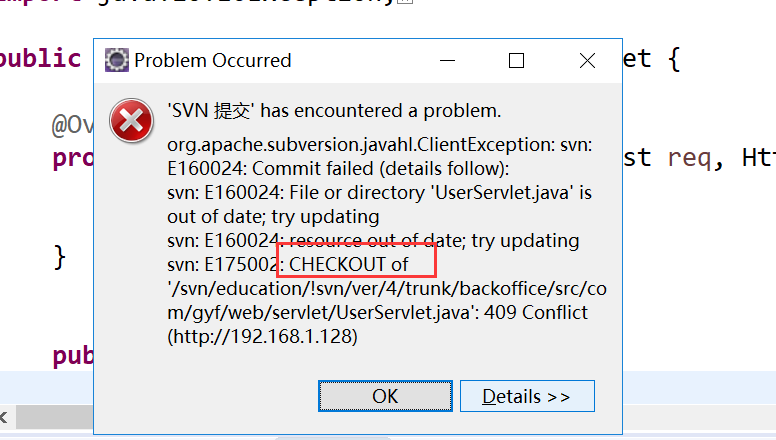


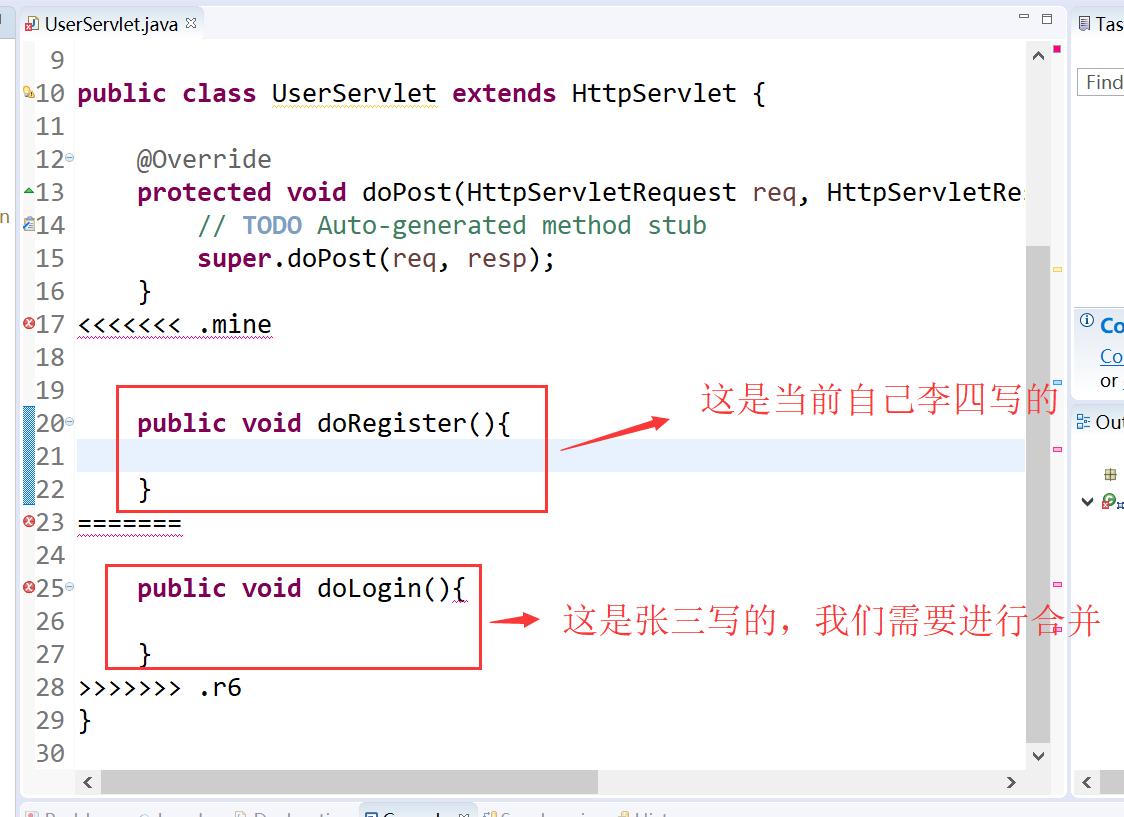
* 然后李四修改同一个文件，然后直接上传，这时不会成功

因为李四的代码版本是5，服务器的代码版本是6

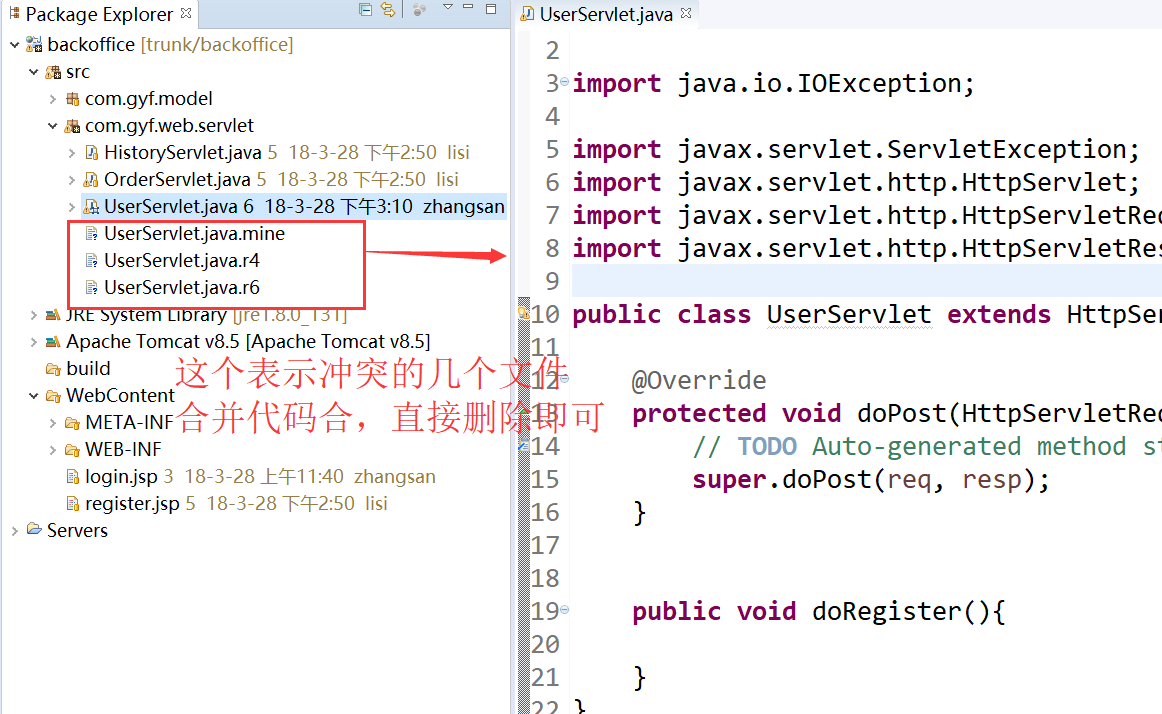
低版本代码不能提交到版本的代码中

先把李四的代码更新到最新版本，然后进行代码冲突的合并





* **删除冲突生成的文件，然后就可以提交了**



你在公司使用什么来管理代码版本控制

SVN-Java开发,SVN用的多

使用SVN时候，你觉得什么事情是比较重要/提交时要注意什么事项

提交时，先更新服务器代码

为了可以提交成功，解决代码冲突

GIT

Eclipse中SVN帐号的保存路径：

C:\Users\"你的用户名"\AppData\Roaming\Subversion\auth

如果提交代码时，记住了密码，把auth下面文件删除即可

## 3.6 张三从服务器更新最新的代码

# 4 创建分支

分支是用来备份已经发布的版本

一个软件功能是不断完善

一开始就开发1.0版本

如果1.0的版本开发完，上线（发布），给客户用，给1.0代码进行备份

进行2.0

Trunk 正在开发版本

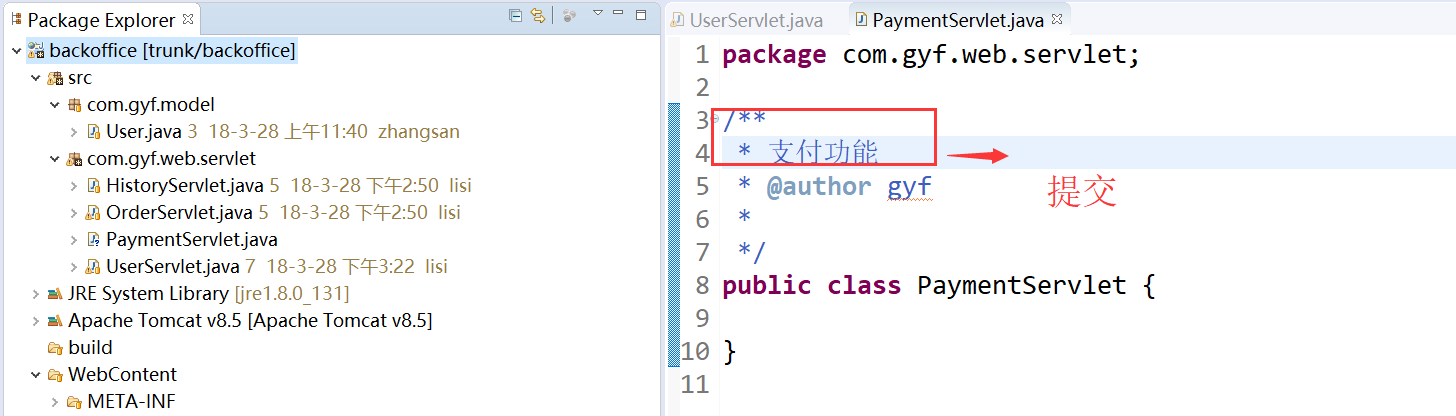
## 4.1 张三创建个1.0版本分枝

* 先在branches下面创建个1.0的文件夹
* 然后选择Team - 分支/标记创建分支

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |

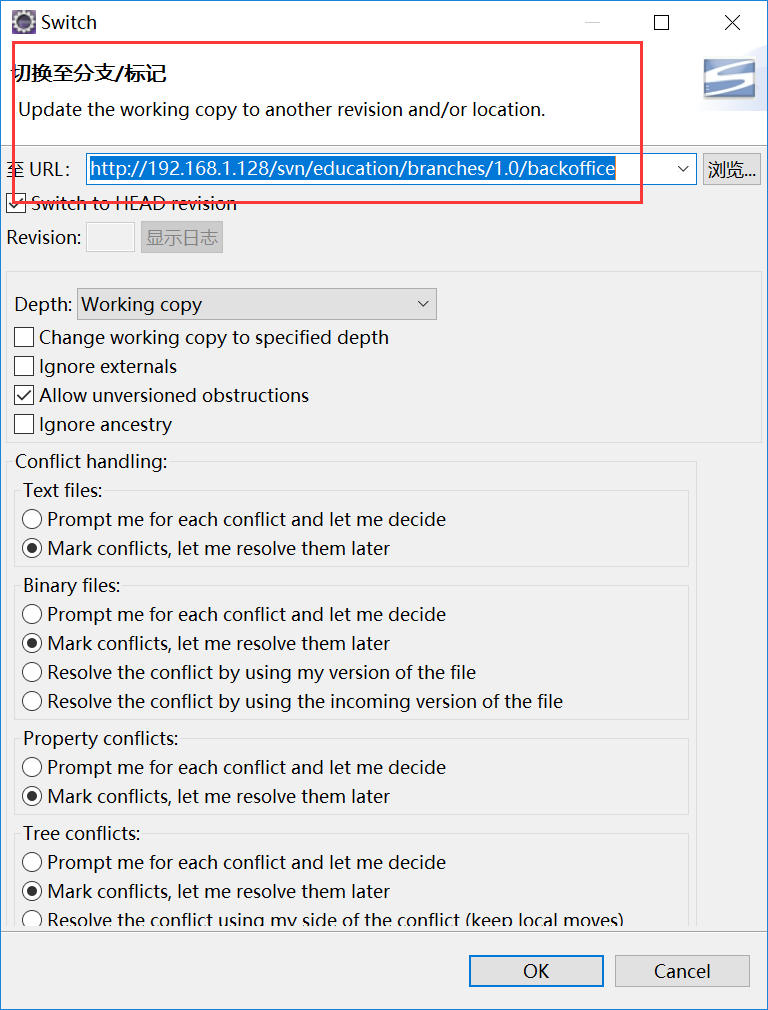
## 4.2 张三在trunk下进行2.0版本的开发

* 随意添加个功能然后提交



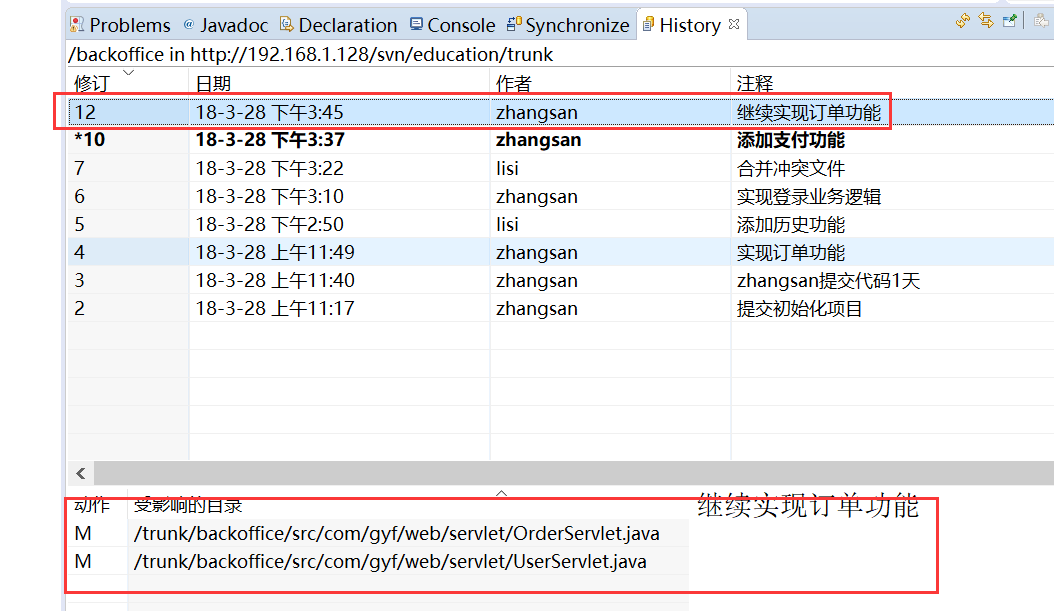
## 4.3 然后切换到1.0分支进行bug修复并提交

* 如修改

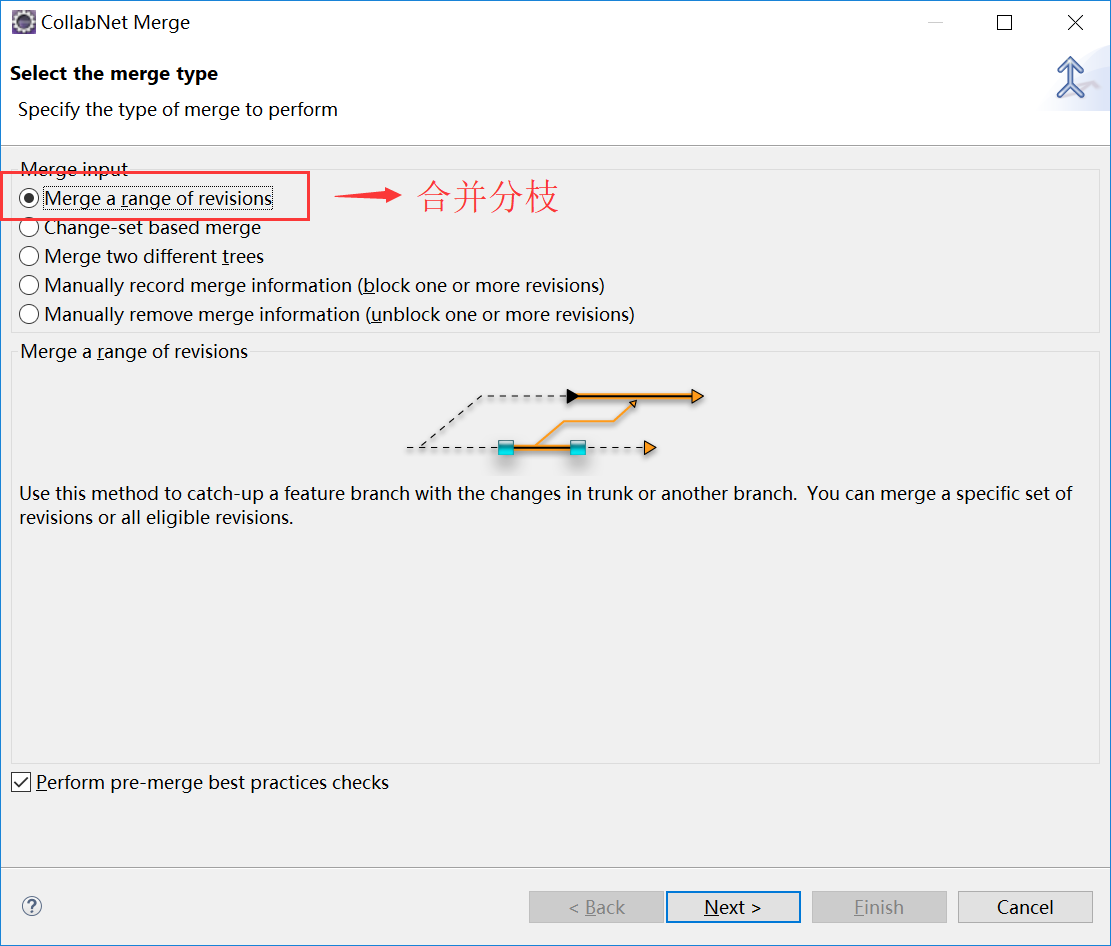


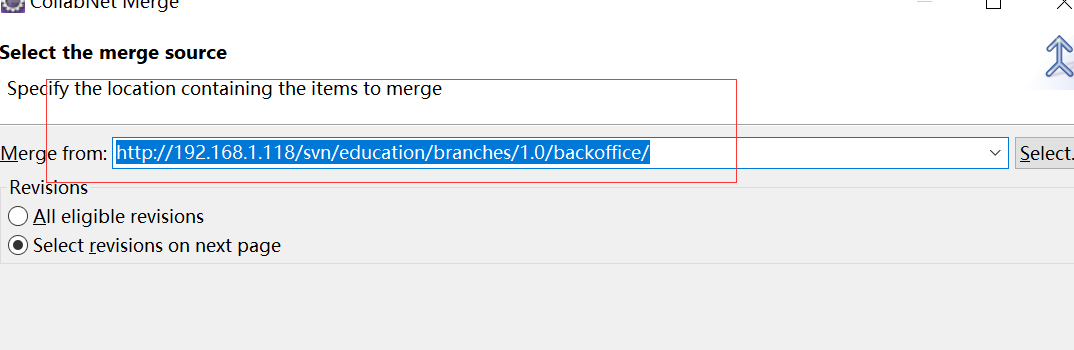
## 4.4 然后继续切换到主杆trunk进行2.0的开发

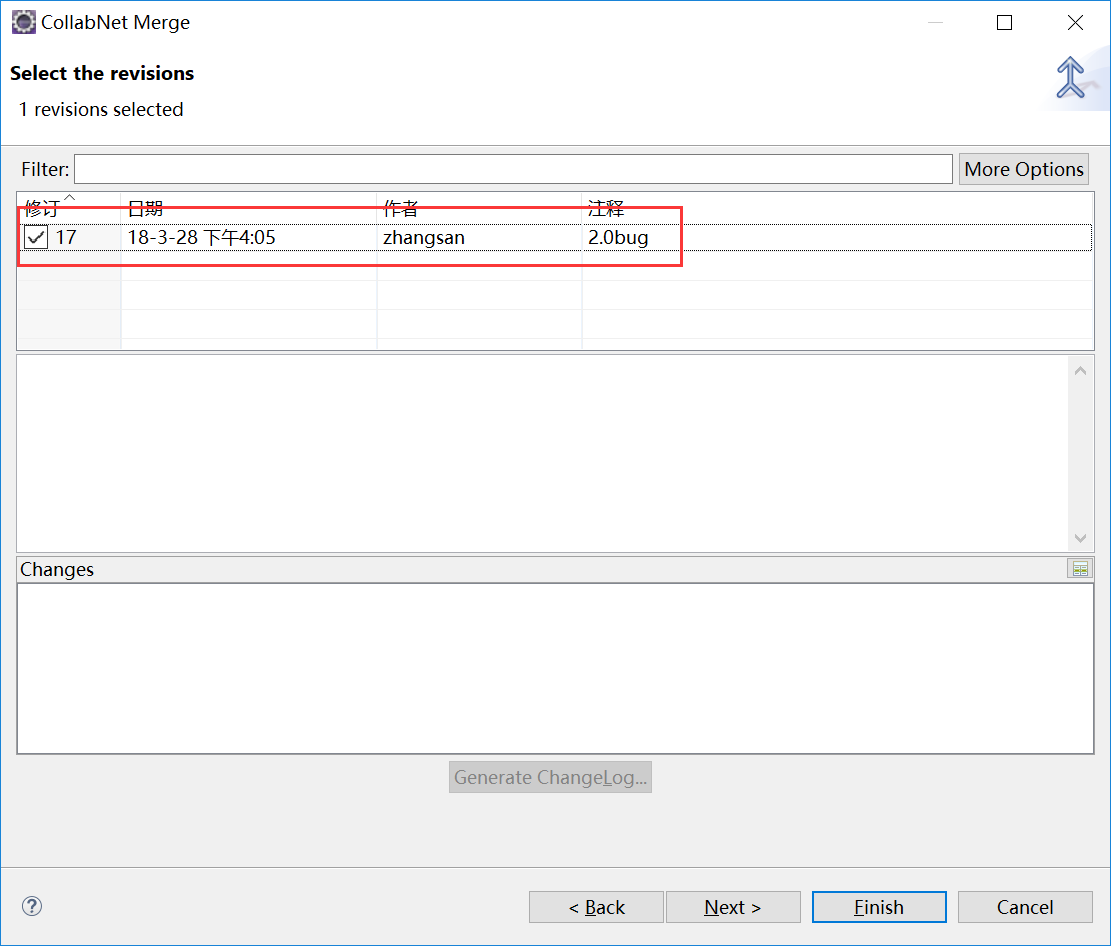
* 在前面bug修改的文件，任意修改下代码，然后提交

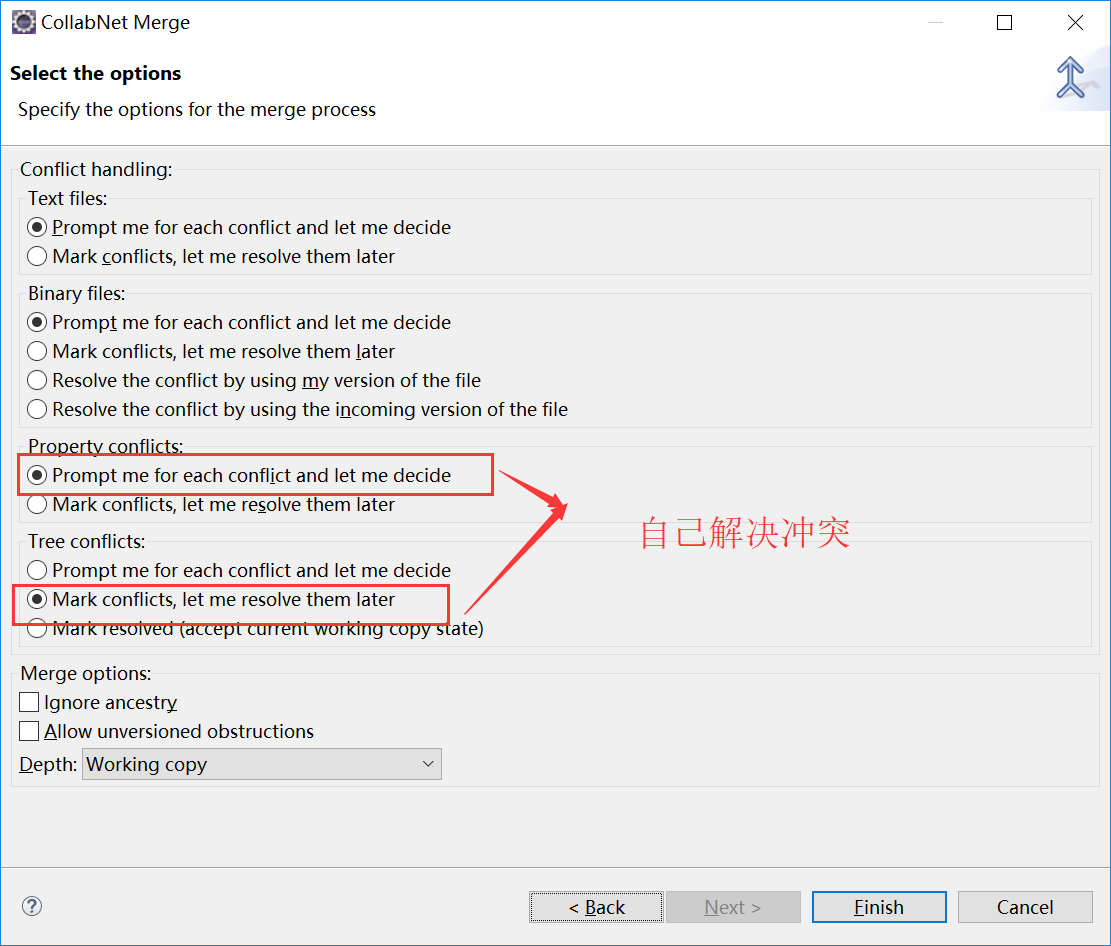


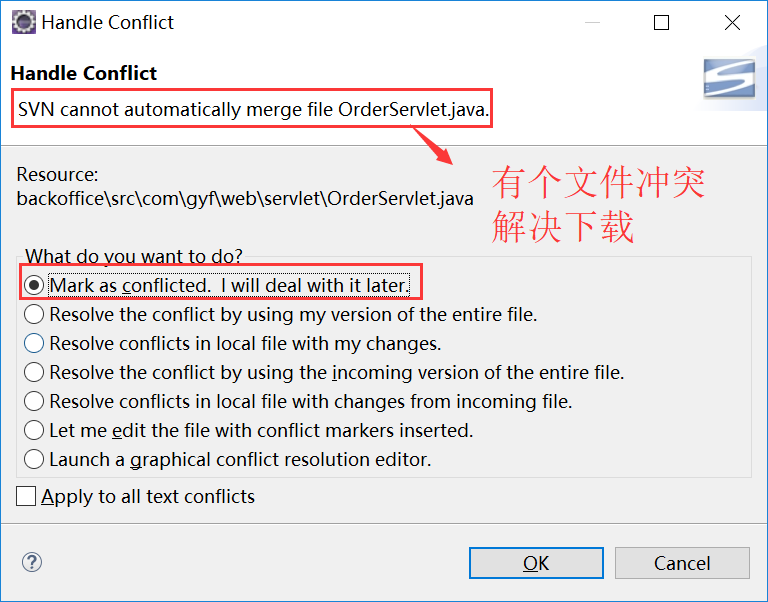
## 4.5 合并1.0分枝

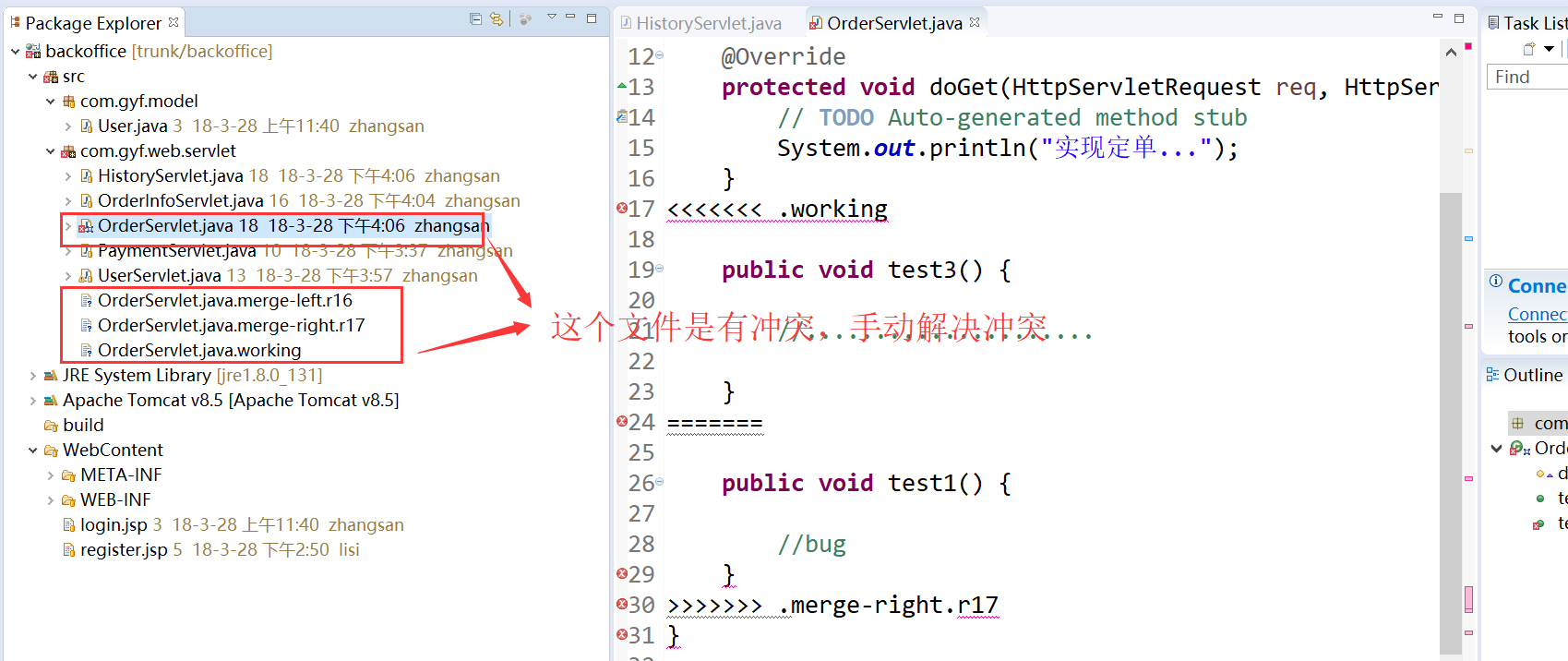












合并完后删除多余的文件，再提交上传即可

开发时SVN使用时间

下午下班提交一次代码到服务

上班更新服务器最新代码

解决冲突:

A,B两个开发

A下午先提交，B就要更新代码，解决冲突，才能提交