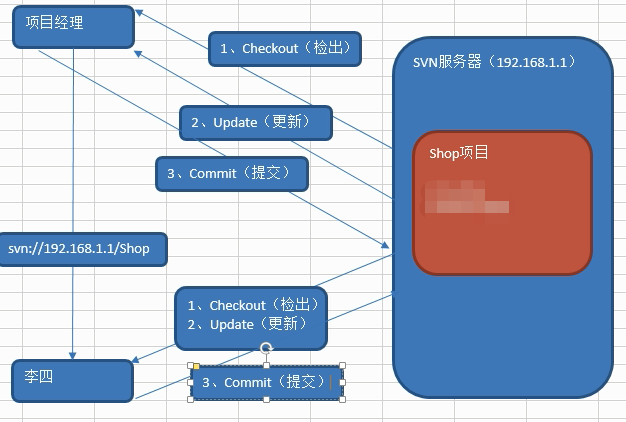
# SVN是什么

Apache Subversion 通常被缩写成 SVN，是一个开放源代码的版本控制系统，Subversion 在 2000 年由 CollabNet Inc 开发，现在发展成为 Apache 软件基金会的一个项目，同样是一个丰富的开发者和用户社区的一部分。

svn 🡪 代码仓库

Subversion(SVN) 是一个开源的**版本控制**系統, 也就是说 Subversion 管理着随时间改变的数据。 这些数据放置在一个中央资料档案库(repository) 中。 这个档案库很像一个普通的文件服务器, 不过它会记住每一次文件的变动。 这样你就可以把档案恢复到旧的版本, 或是浏览文件的变动历史。



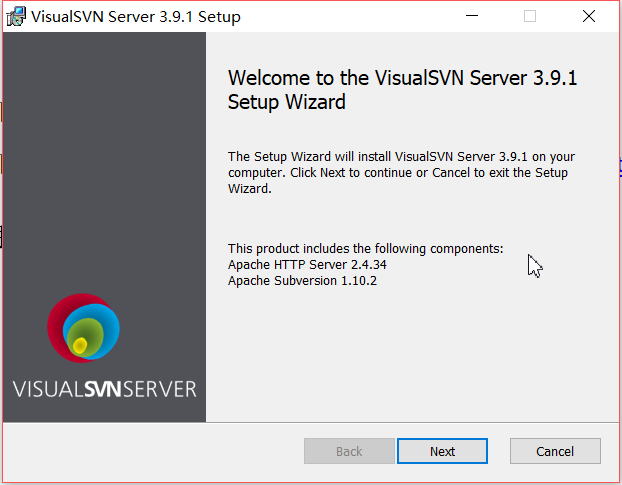
# 二、SVN下载安装

## 2.1下载

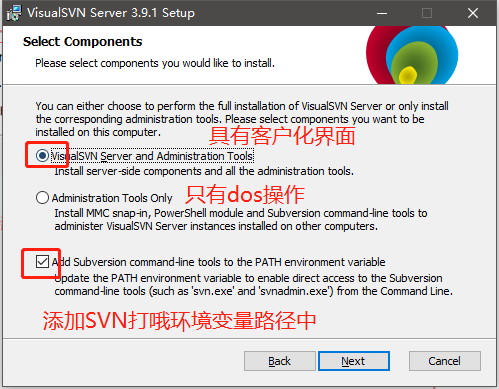
服务端下载地址：<https://www.visualsvn.com/server/download/>

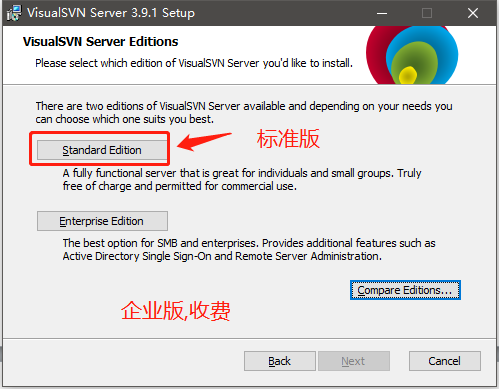
客户端下载地址：<https://www.visualsvn.com/visualsvn/download/tortoisesvn/>

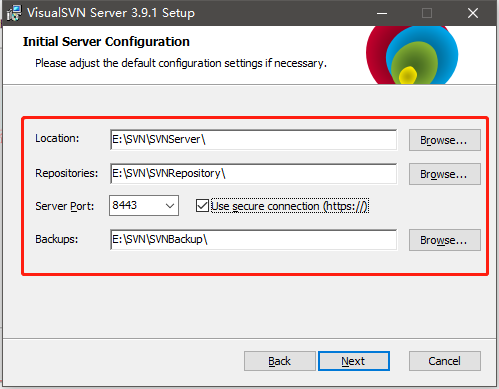
## 2.2 安装服务端



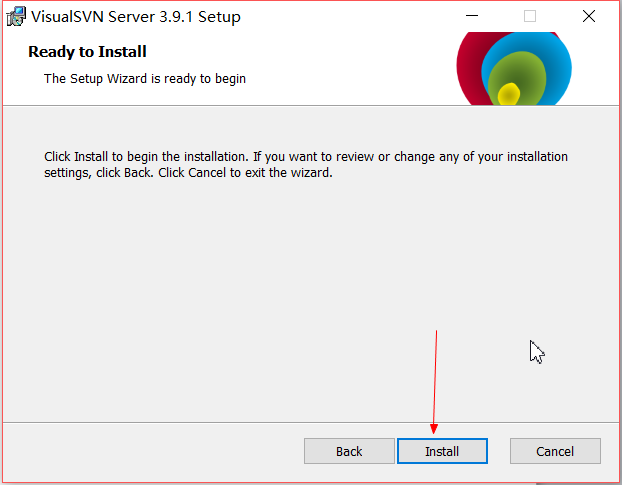


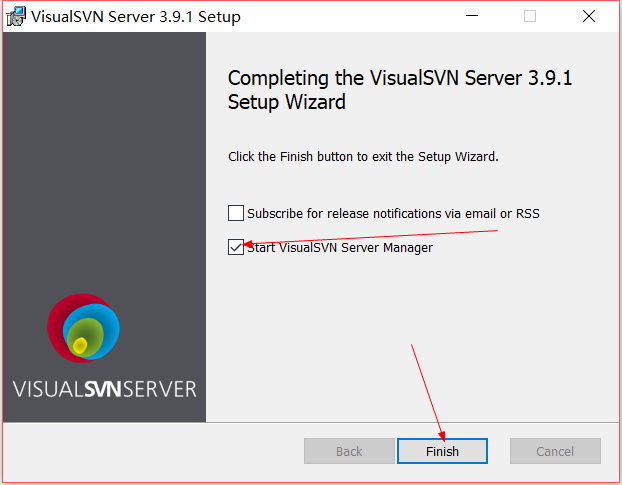






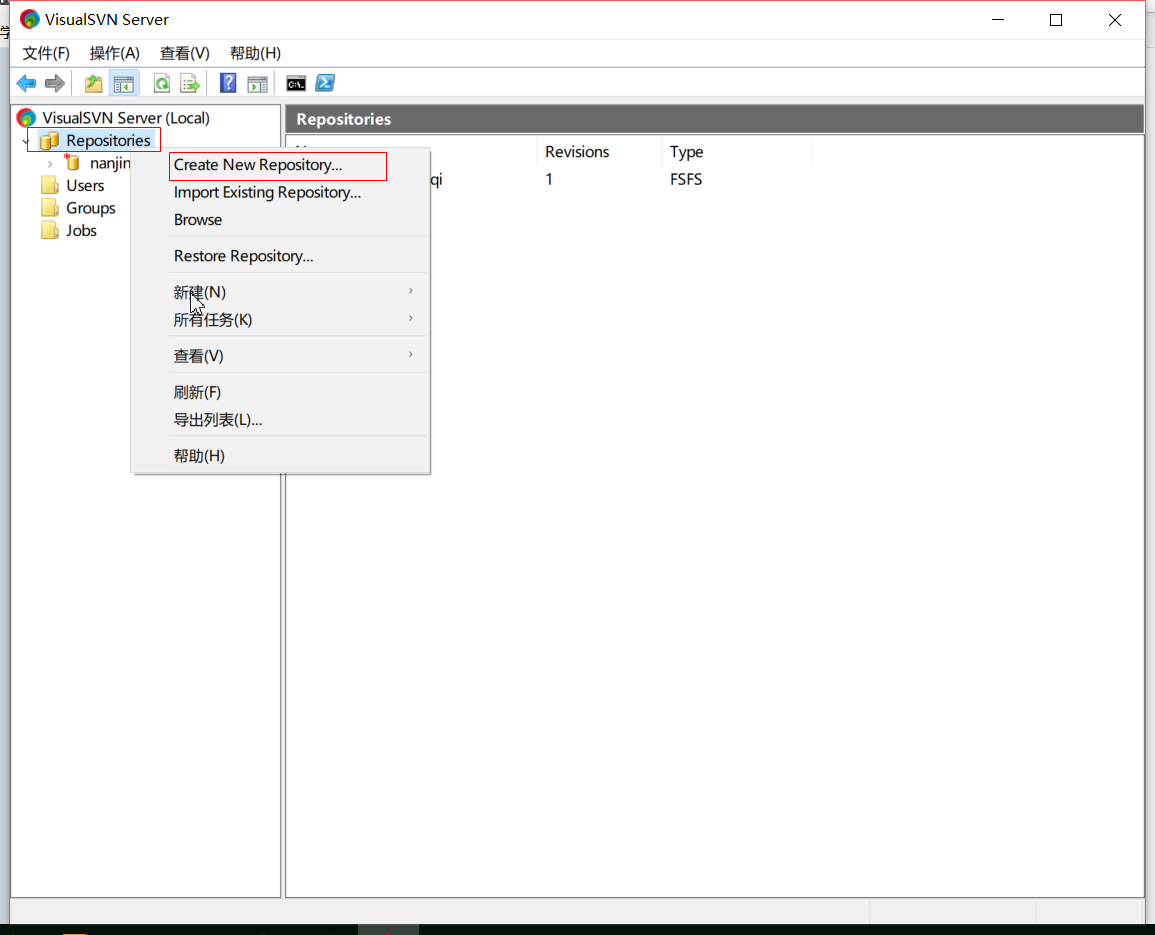
注意:安装路径不能出现中文,空格,中文字符

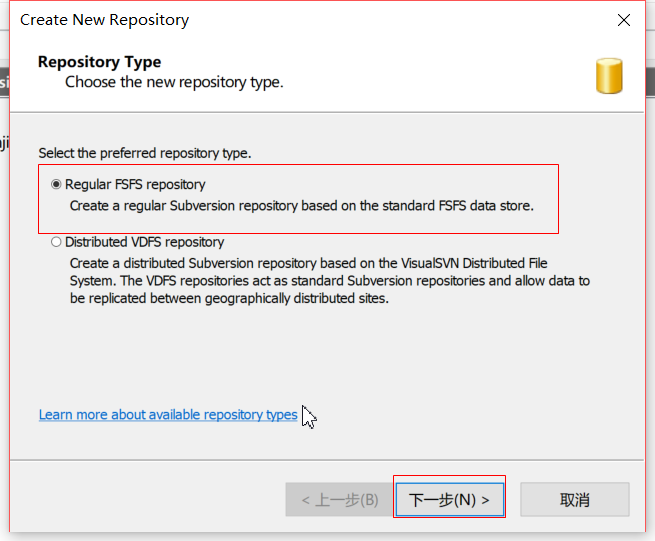


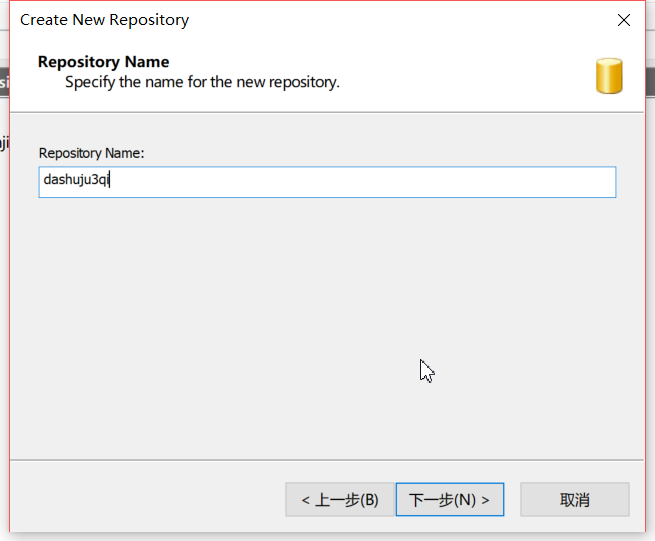


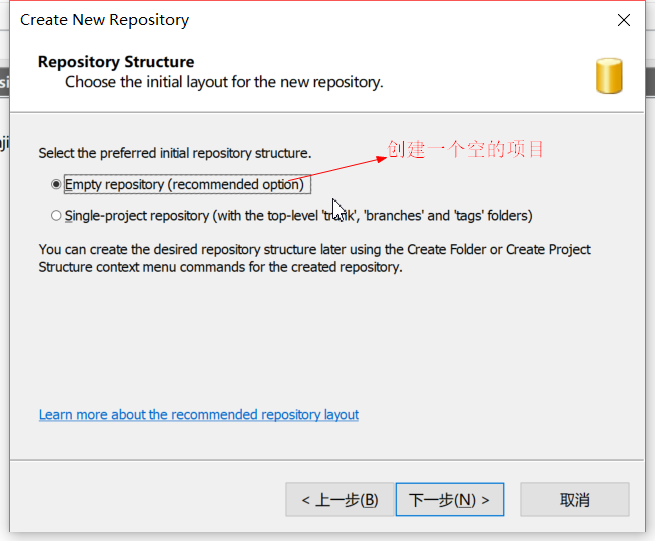
## 2.3服务器端配置(图形界面配置)

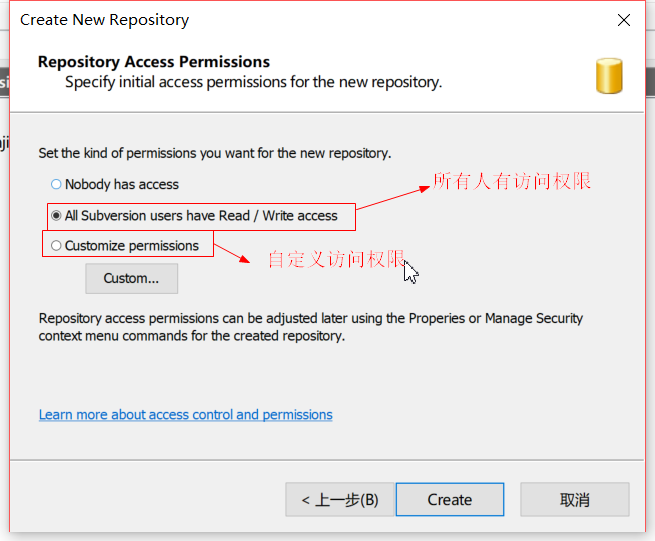
-- 创建仓库

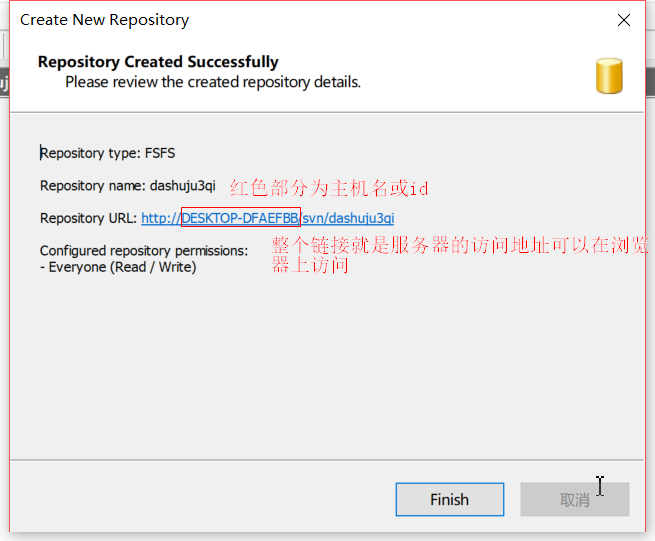






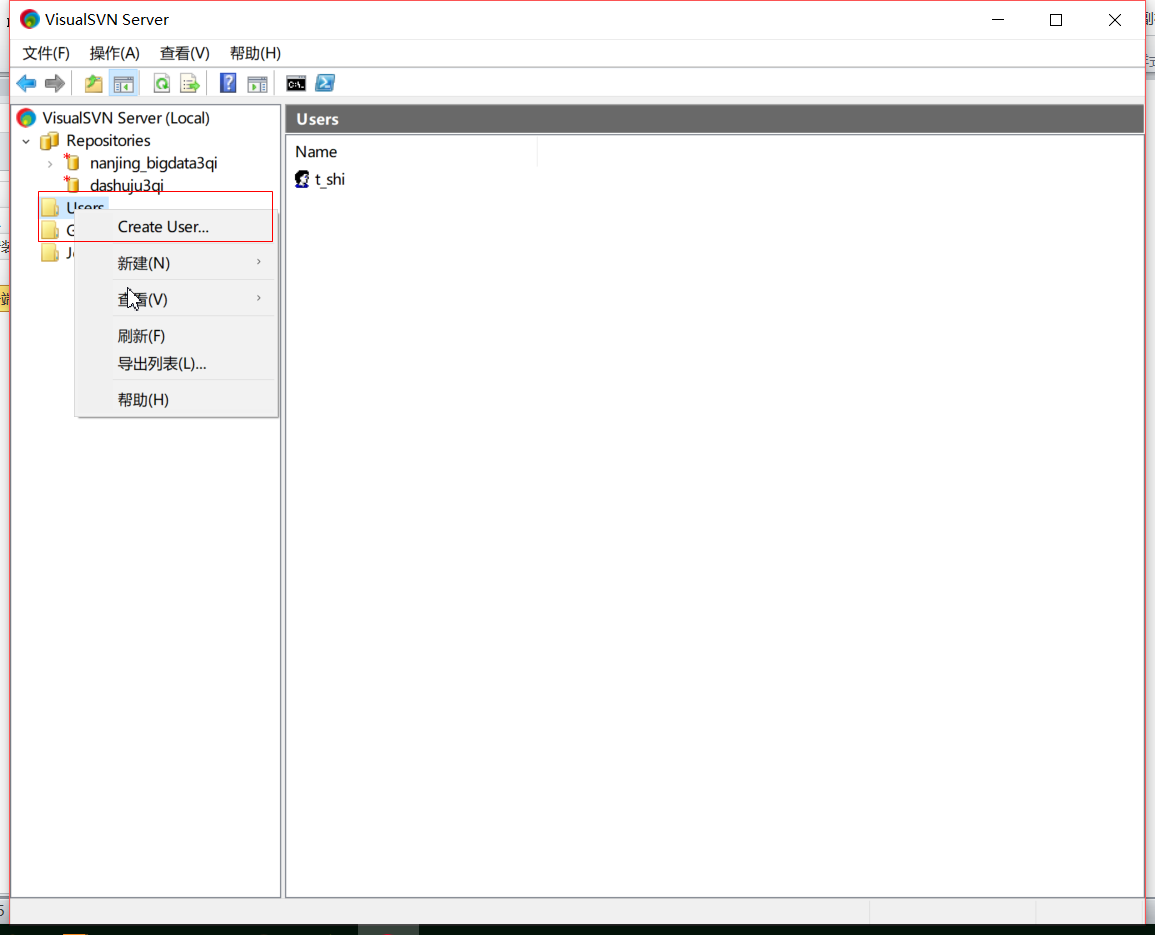


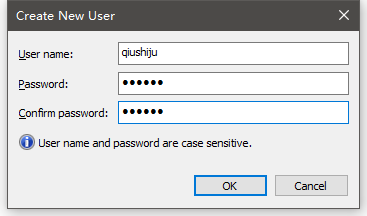


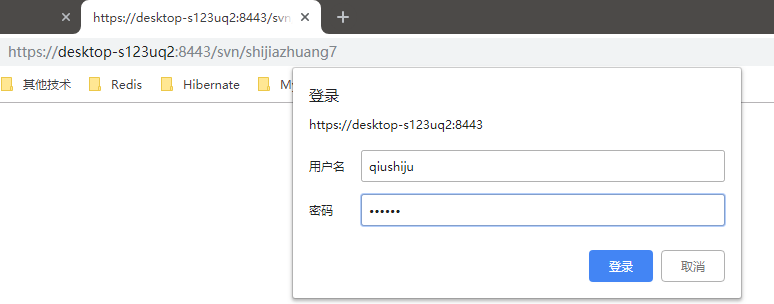


创建的项目就在E:\SVN\SVNRepository ,SVN的仓库中

至此服务器搭建完成，但是还没有用户，需要创建用户





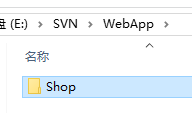


## 2.4服务端配置(使用命令配置)

1. 创建一个项目

1.1 首先在SVN服务端创建一个共有目录WebApp作为项目目录

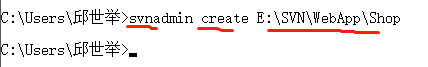
1.2 在WebApp目录下创建Shop项目文件夹,作为Shop项目的仓库,在SVN中,每一个项目就是一个仓库

  
 1.3 创建版本仓库

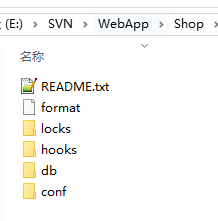
即让我们自己常见的Shop文件夹是一个仓库

语法:(需要打开cmd命令行)

svnadmin create 路径



成功后会看到如下

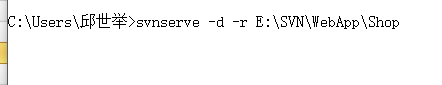


2. 服务器监管

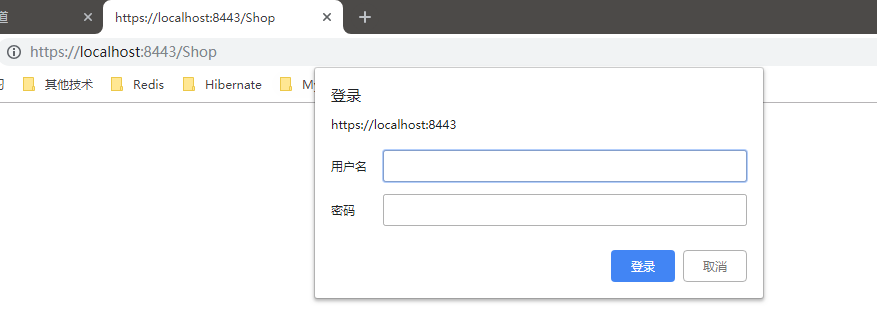
svn://localhost或(ip地址)访问到相关数据仓库(如Shop仓库)

dos语法:

svnserve –d(后台运行) –r(监管目录) 仓库路径



这样以后我们在客户端就可以使用svn://localhost(或IP地址)就可以访问我们的shop仓库

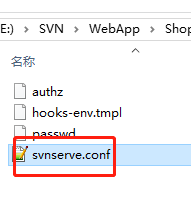


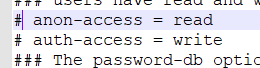
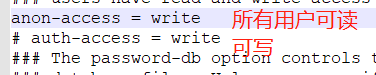
3.权限控制

默认SVN默认不允许匿名访问.

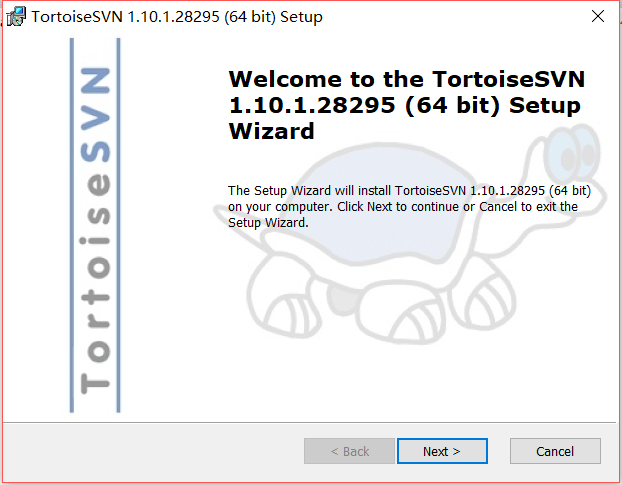
现在我们修改配置使任何人可以访问

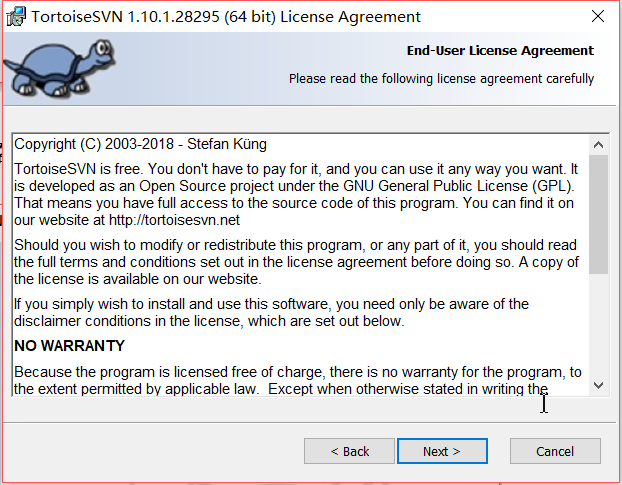
找到Shop仓库的conf文件夹打开

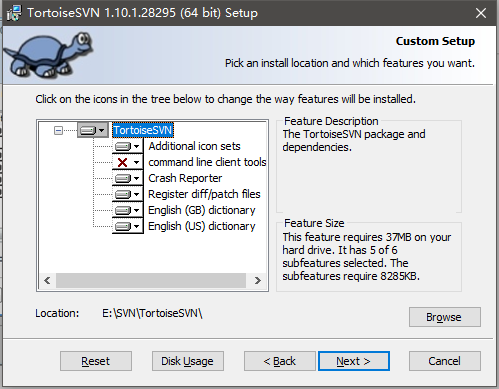


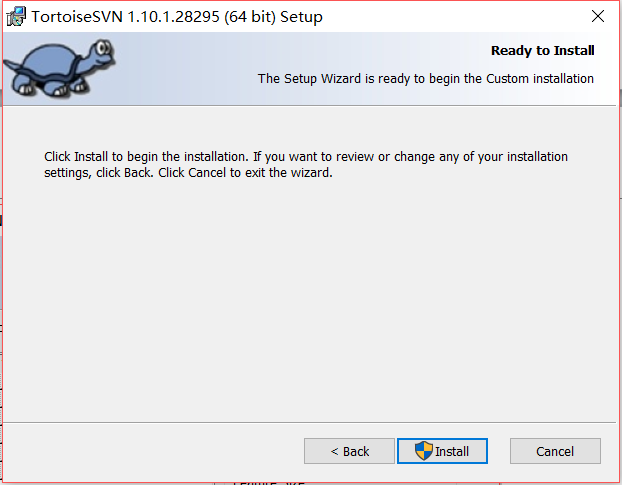
然后修改为

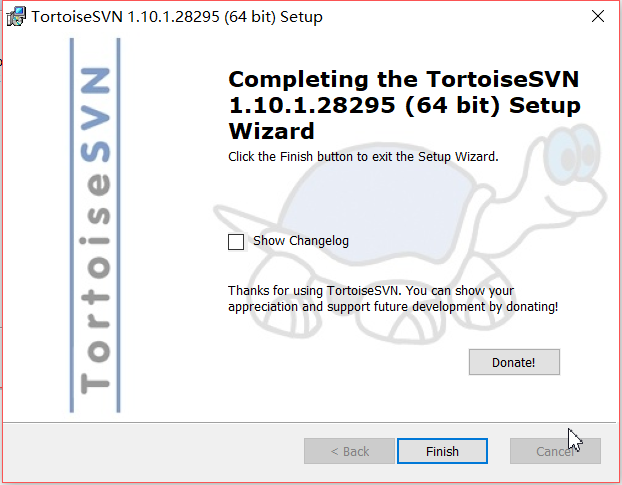
## 2.5 安装客户端



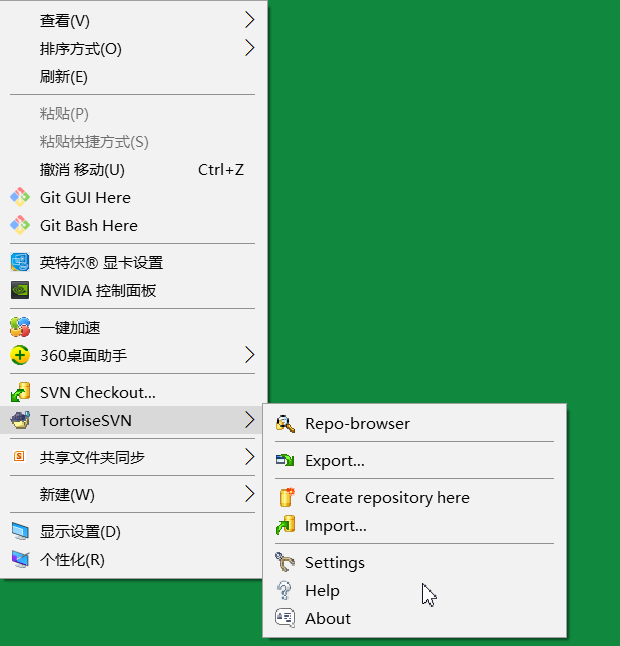








如果在桌面空白处右击出现以下情况说明安装成功



**使这些操作生效需要重启电脑.**

# 三 SVN使用详解

## checkout 检出:

在需要检出的目录中右键checkout,

填入路径:

svn:localhost即可(前提是开启svn服务器监控)

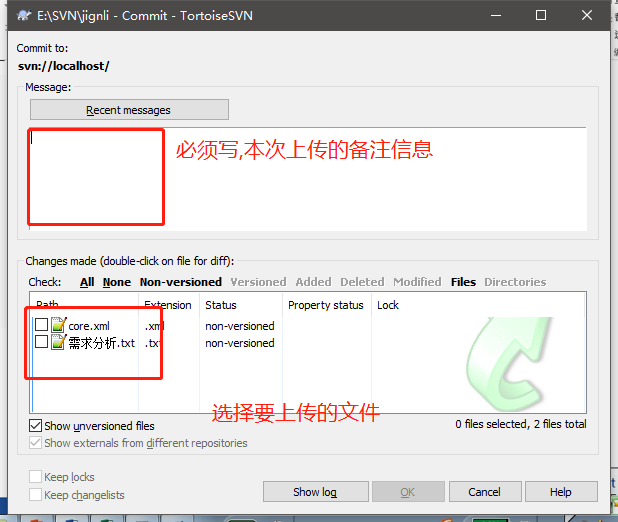
<https://QiuShiju:8443/svn/xxxx/> (图形界面创建的仓库)

* 1. 用于客户端连接服务端
  2. 第一次从服务端拉取数据(更新数据到本地)
  3. 注意只有第一次从服务器更新到本地是checkout,以后都是update

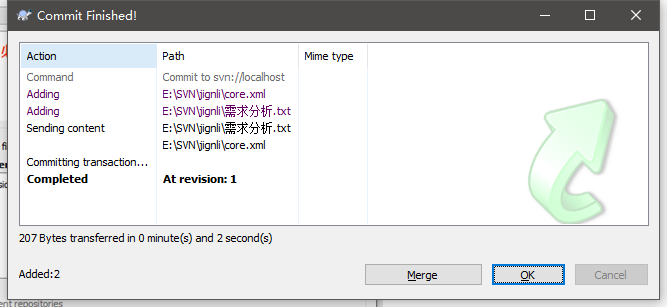
## commit 提交

* 1. 提交本地数据到服务端



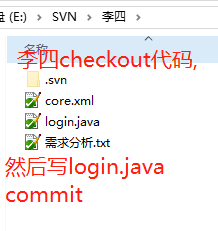


提交成功显示如下



如果新进员工,就新建一个自己的文件,在其中右键checkout检出代码.然后再写自己模块的代码,

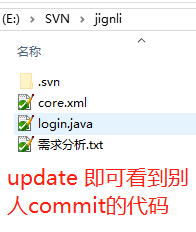
写完之后commit即可



## update

* 将服务器的最新代码更新到本地

其他组员想要最新的项目信息,只需要在自己的文件夹下右键update,即可看到被人commit提交到服务器的代码



## 常用图标解释



### 常规图标



客户端文件与服务器端文件完全一致时,显示此图标

### 修改图标



当客户端文件有修改,但未提交时,系统显示以上图标

### 增加图标



当我们编写的文件已经提交到提交队列时(未提交),会显示以上图标

### 冲突图标



当客户端提交的文件与服务端文件有冲突,系统会显示以上图标

### 删除图标



当服务端数据已删除,客户端会显示该图标

### 忽略图标



客户端文件不需要上传时,显示以上图标

### 无版本控制图标



我们自己在客户端编写的新代码没有提交到提交队列时,系统显示以上图标

### 只读图标



当客户端文件已只读形式存在时,系统显示以上图标

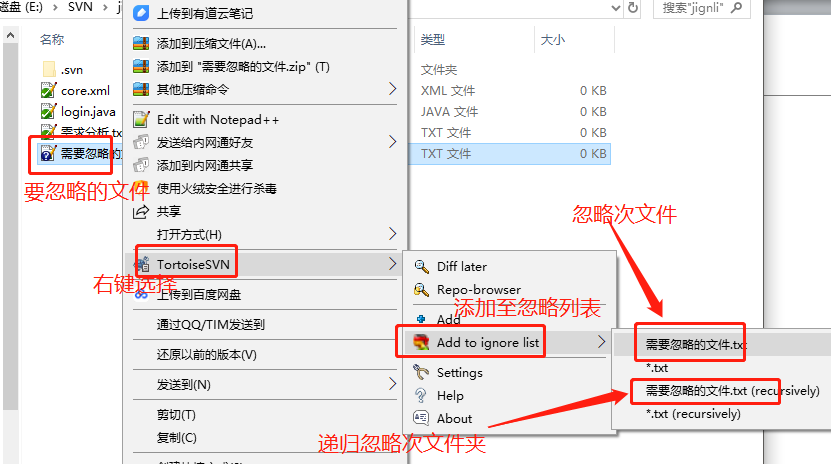
### 锁定图标



当服务端文件以及锁定,客户端的该文件就会显示如上图标

## 5 忽略功能

当某些文件我们不需要上传至服务端就可以采用忽略功能.



忽略成功



## 6 版本回退

### 版本说明

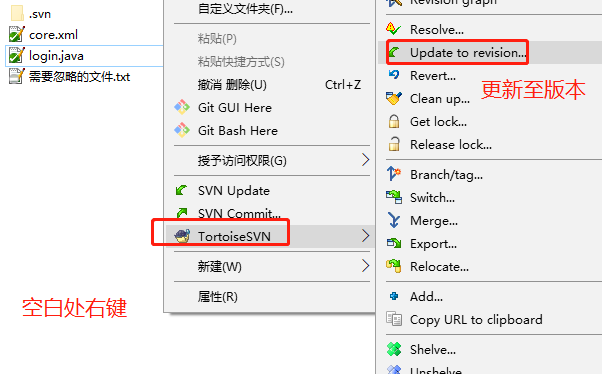
|  |
| --- |
| 版本号: 主版本号.次版本号.修订版本号  例如： 1.13.23  版本说明  Base(SNAPSHOT)：此版本表示该软件仅仅是一个假页面链接，通常包括所有的功能和页面布局，但是页面中的功能都没有做完整的实现，只是做为整体网站的一个基础架构。  Alpha ：软件的初级版本，表示该软件在此阶段以实现软件功能为主，通常只在软件开发者内部交流，一般而言，该版本软件的Bug较多，需要继续修改，是测试版本。测试人员提交Bug经开发人员修改确认之后，发布到测试网址让测试人员测试，此时可 将软件版本标注为alpha版。  Beta ：该版本相对于Alpha 版已经有了很大的进步，消除了严重错误，但还需要经过多次测试来进一步消除，此版本主要的修改对象是软件的UI。修改的的Bug 经测试人员测试确认后可发布到外网上，此时可将软件版本标注为 beta版。  RC ：该版本已经相当成熟了，基本上不存在导致错误的Bug，与即将发行的正式版本相差无几。  Release：该版本意味“最终版本”，在前面版本的一系列测试版之后，终归会有一个正式的版本，是最终交付用户使用的一个版本。该版本有时也称标准版。  例如：  1.13.23 Release |



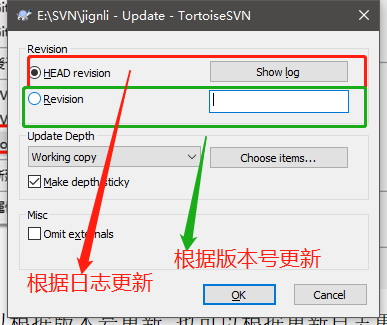
### 版本回退

* 先在jingli文件夹下删除一个文件,然后提交到服务器,提交时写了备注.
* 现在发现,文件删除错了,不应该删除,那该如何办?就是版本回退功能

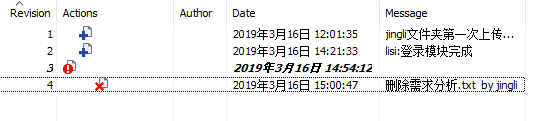
在文件夹下空白区域右键



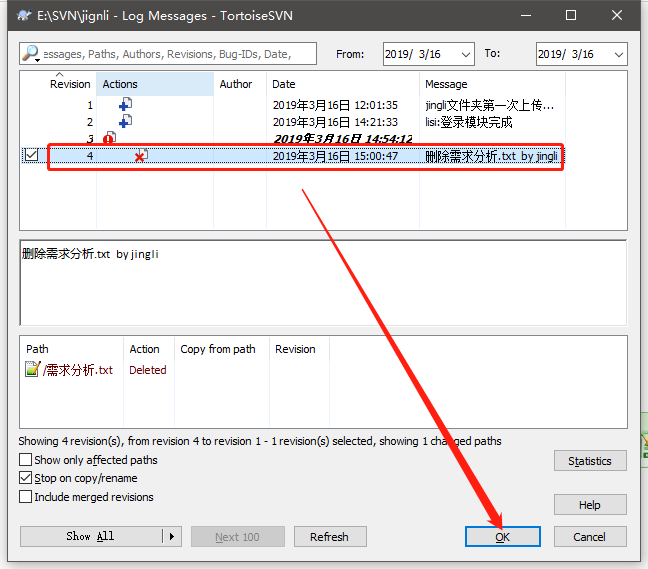
选择更新至哪一版:可以根据版本号更新 也可以根据更新日志更新



选择根据日志更新,出现如下页面



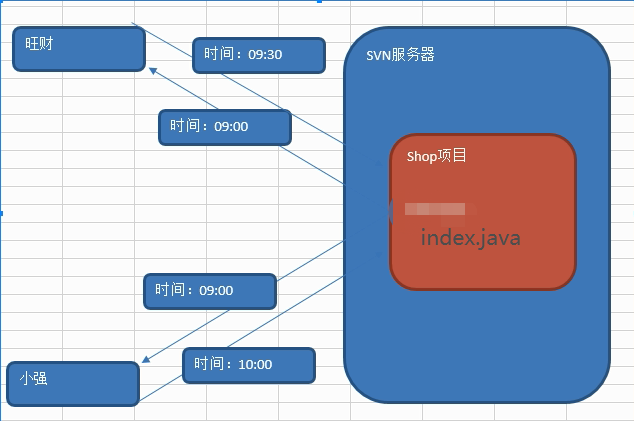
单击需要回退那一版,确定即可



刷新就可看到删除的文件又回来了….

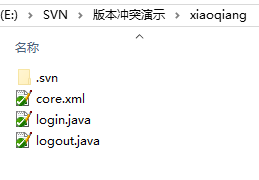
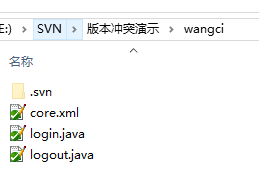
## 7 版本冲突

实际项目中,两个人同时修改某一个文件,后一个人就会对前一个人文件有覆盖/删除等冲突操作.

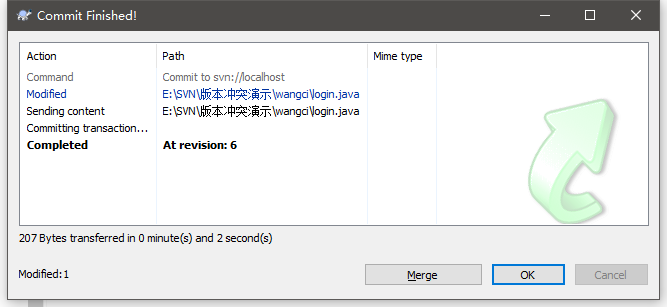
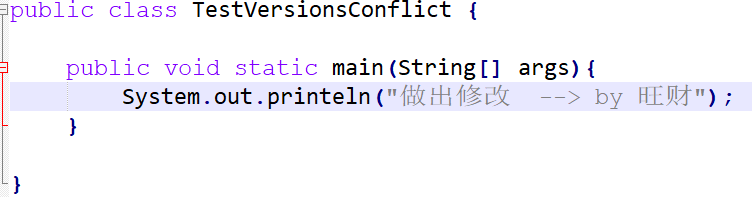


### 模拟演示

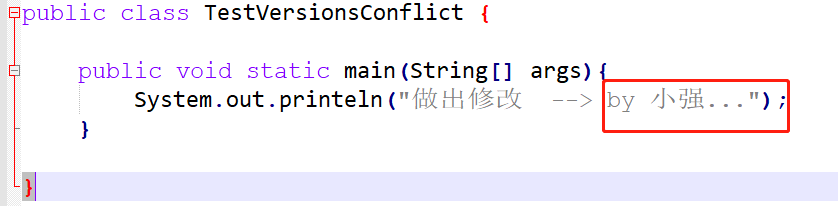
1. 早上9:00上班,旺财和小强都更新数据到本地电脑

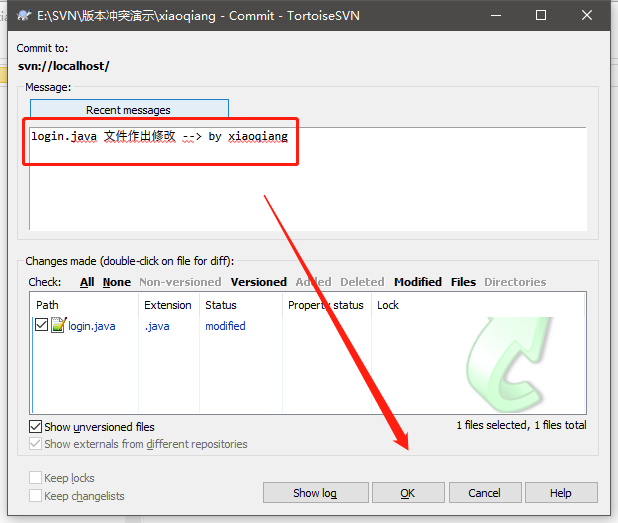


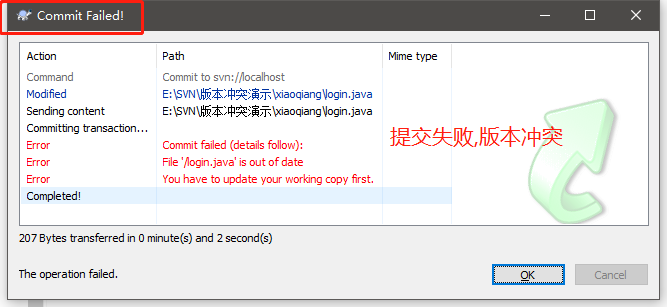
1. 旺财开始修改login.java文件.并在9:30时**提交**了该文件



1. 小强也对login.java文件作出修改,并在10:00**提交**该文件

修改 ..

准备提交.. 

提交失败,版本冲突..

### 解决版本冲突

1. 合理分配项目开发时间

小强上午,旺财下午

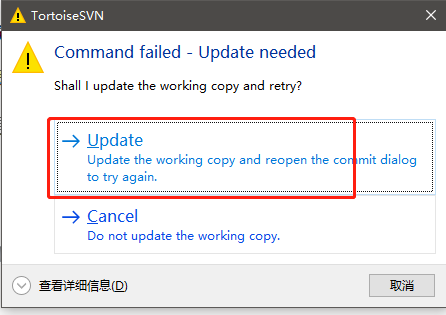
保证不同时对一个项目就行修改…

1. 合理分配项目模块

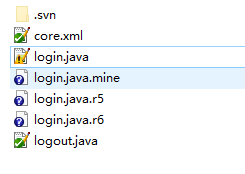
小强做登录模块,旺财做登出模块,就不会的对同一个文件进行修改..

1. 使用svn提供方法解决冲突

3.1) 出现版本冲突,先更新服务端数据到本地



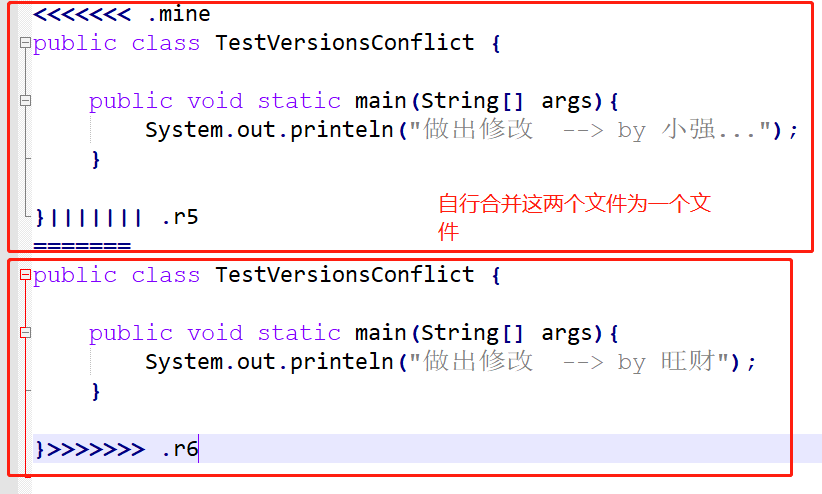
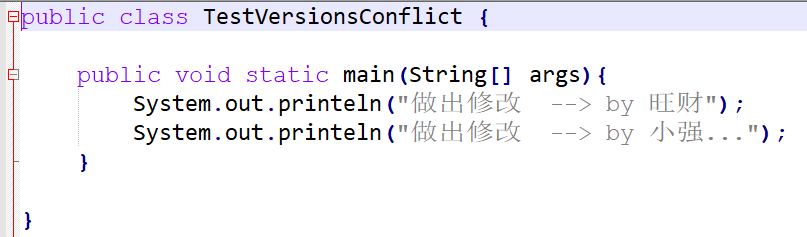
然后发现本地文件夹中文件多出几个



* login.java 旺财和小强整合后的文件
* login.java.mine 我们自己(小强)修改后的文件
* login.java.r5 这个文件在旺财和小强更新前的状态 (起始状态文件)
* login.java.r6 之前提交的(旺财修改后提交的)文件

3.2) 删除除login.java 文件之外的所有文件

3.3) 自行修改整合的login.java文件

3.4) 重新提交,(提交成功)

