## 1. Github账号信息

**用户名：**ivyzvh

**协议：**HTTPS

**地址：**

https://github.com/ivyzvh/xx.git

**协议：**SSH

**地址：**

git@github.com:ivyzvh/xx.git

2.码云账号信息

<https://gitee.com/ivyzvh/doc.git>

**Quick setup — if you’ve done this kind of thing before**

[**Set up in Desktop**](https://desktop.github.com/)

**or**

窗体顶端

HTTPS

窗体底端

窗体顶端

SSH

窗体底端



We recommend every repository include a [README](https://github.com/ivyzvh/xx/new/master?readme=1), [LICENSE](https://github.com/ivyzvh/xx/new/master?filename=LICENSE.md), and [.gitignore](https://github.com/ivyzvh/xx/new/master?filename=.gitignore).

**…or create a new repository on the command line**

echo "# xx" >> README.md

git init

git add README.md

git commit -m "first commit"

git remote add origin git@github.com:ivyzvh/xx.git

git push -u origin master

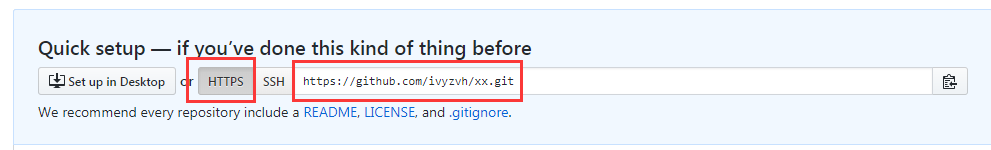
**…or push an existing repository from the command line**

git remote add origin git@github.com:ivyzvh/xx.git

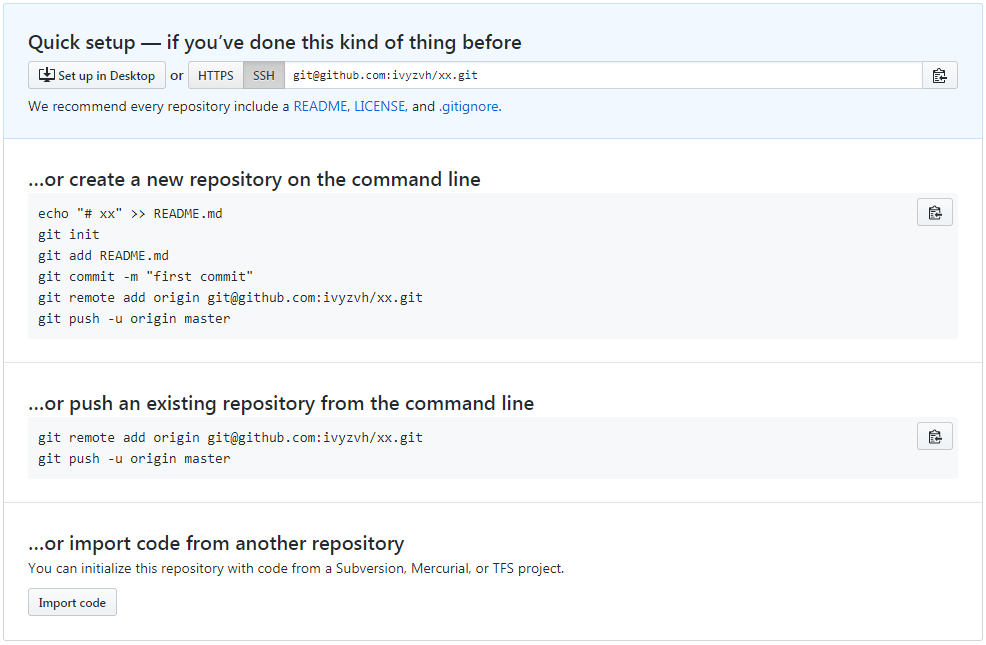
git push -u origin master

**…or import code from another repository**

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.



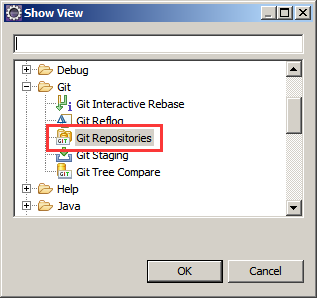


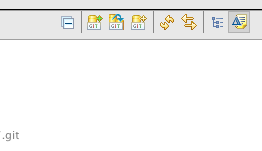


## 2. 将远程仓库的项目克隆到Eclipse中

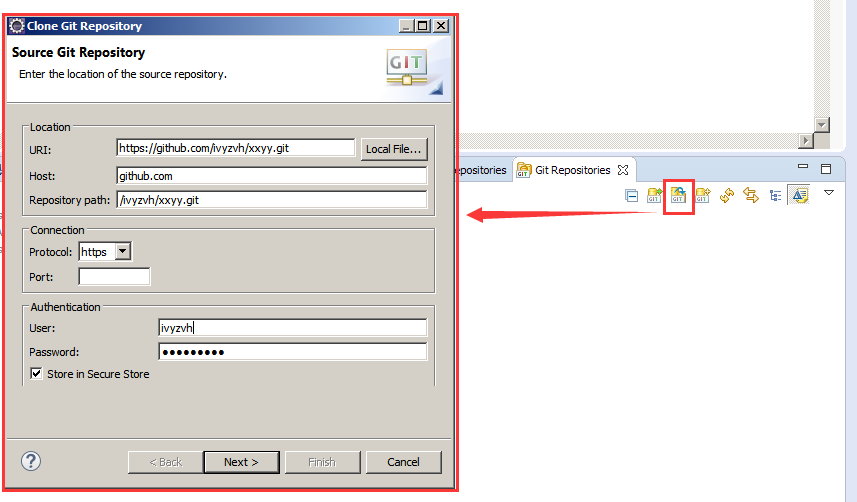
 将Ecpilse切换到git视图

Windows 🡪 Show View 🡪 Other … 🡪 选择Git Repositories

  
    点击Clone Git repository【没找到】



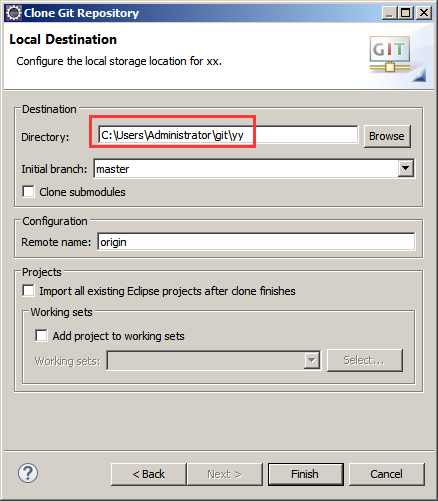
或者点击箭头的图标，如下图。



在里面填入，你需要克隆的项目的url地址：输入账号和密码

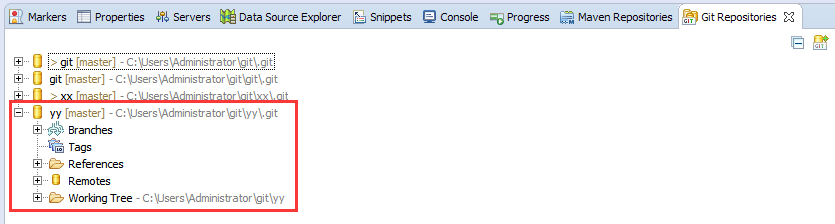
【URL】

>> 【Next】 >> 【Next】

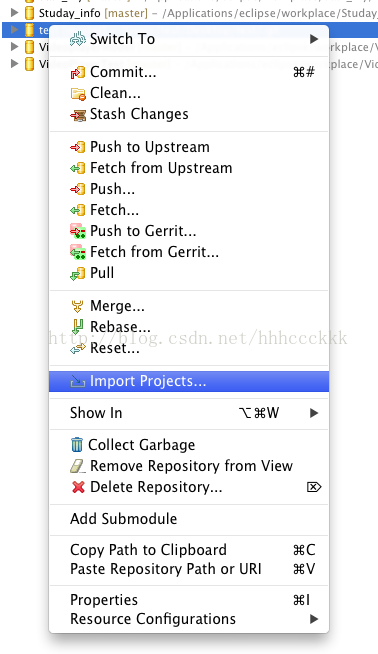


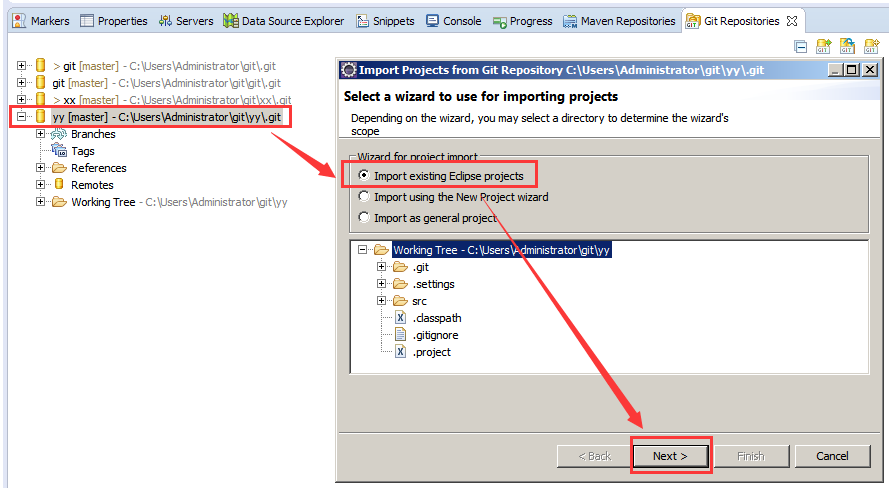
点击【Browse】，选择一个目录放项目文件，不要放在Eclipse工作目录。

点击【Finish】完成Clone操作，结果如下图：



完成后，用import导入项目到Eclipse，方法是：在Git Repositories视图中选中“yy”，右键弹出菜单，点击其中“Import Projects…”项，

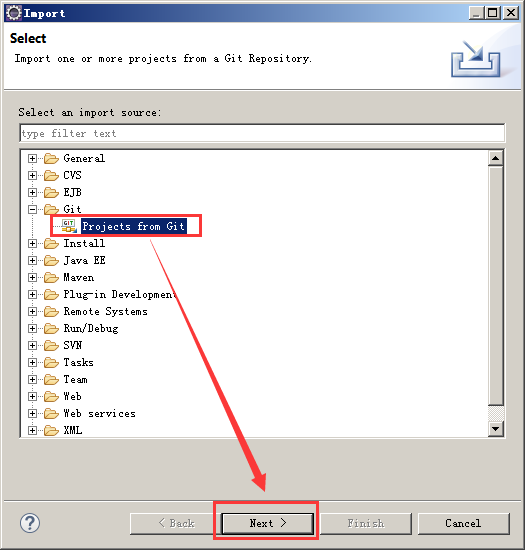


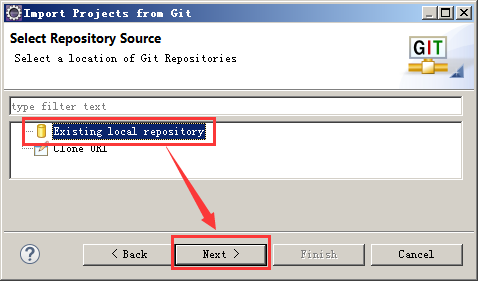


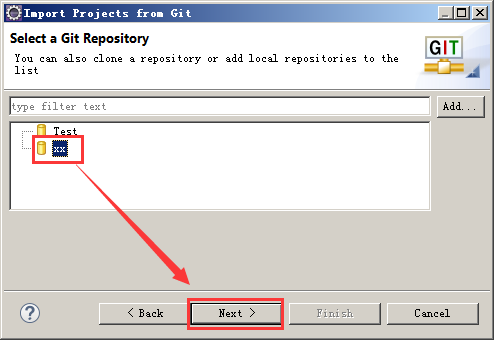
和导入一般工程一样的了

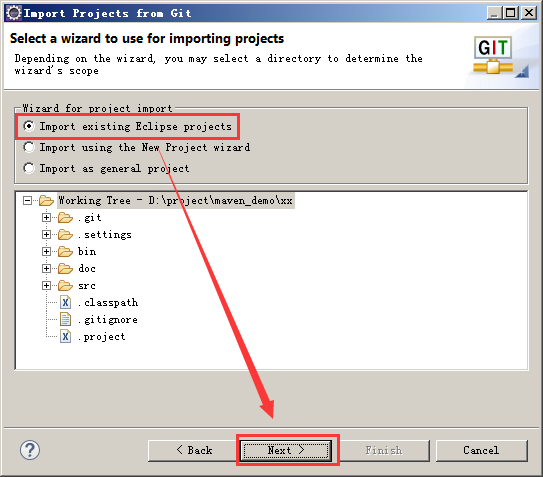
### 2.2导入Git项目方法二

切换到Java EE视图🡪Project Explorer🡪右键弹出菜单，选中其中[Import…]🡪打开Import窗口🡪选择[Git]🡪[Project from Git]，如下图







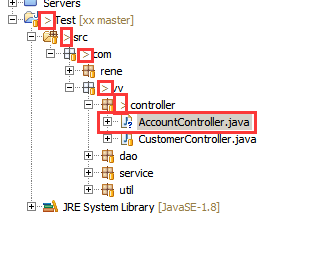


## 3. Eclipse修改后提交到Github网络仓库

[1]在添加AccountController.java文件。

[2]在Project Explorer中选中项目，右键弹出菜单，选中【Team】🡪【Commit…】，弹出窗口Commit Changes，填写注释，选中要提交的文件，点击【Commit and Push】，完成修改提交操作。

【说明】默认情况下，【Commit Changes】窗口并非如图所示，需要进行设置，方法如下图3.3所示：

图3.1

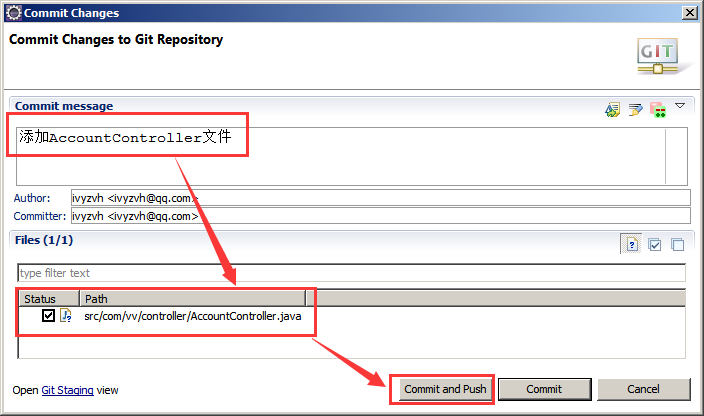


图3.2

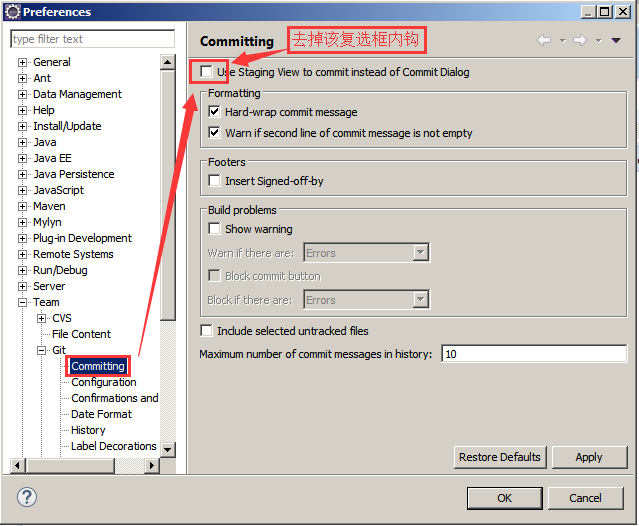
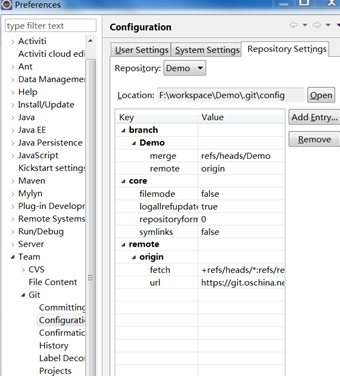


图3.3

## 4 获取网络他人修改

## 4.1 右键项目Team-->Pull

如果出错：就更改设置，如下图：



**点击open添加属性：**

[branch "master"]

         remote = origin

         merge = refs/heads/master

[remote "origin"]

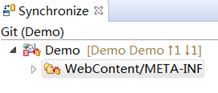
         url = https://git.oschina.net/maofa/test.git

         fetch = +refs/heads/\*:refs/remotes/origin/\*

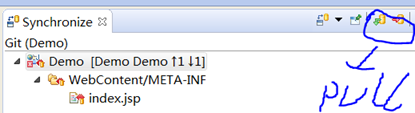
         push = refs/heads/master:refs/heads/master

## 5 更改冲突

## 5.1 右键项目Team—>Synchronize 添加到同步视图

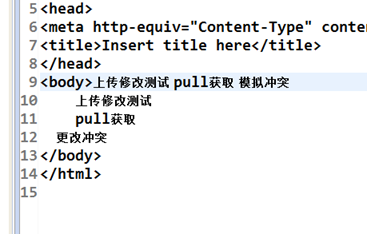


## 5.2 出现红图标证明修改冲突



 先拉下来更新pull

## 5.3 手动解决冲突



## 5.4 右键项目 Team🡪Add to index 出现\*

http://images2015.cnblogs.com/blog/1107610/201702/1107610-20170216182446082-1178511217.png

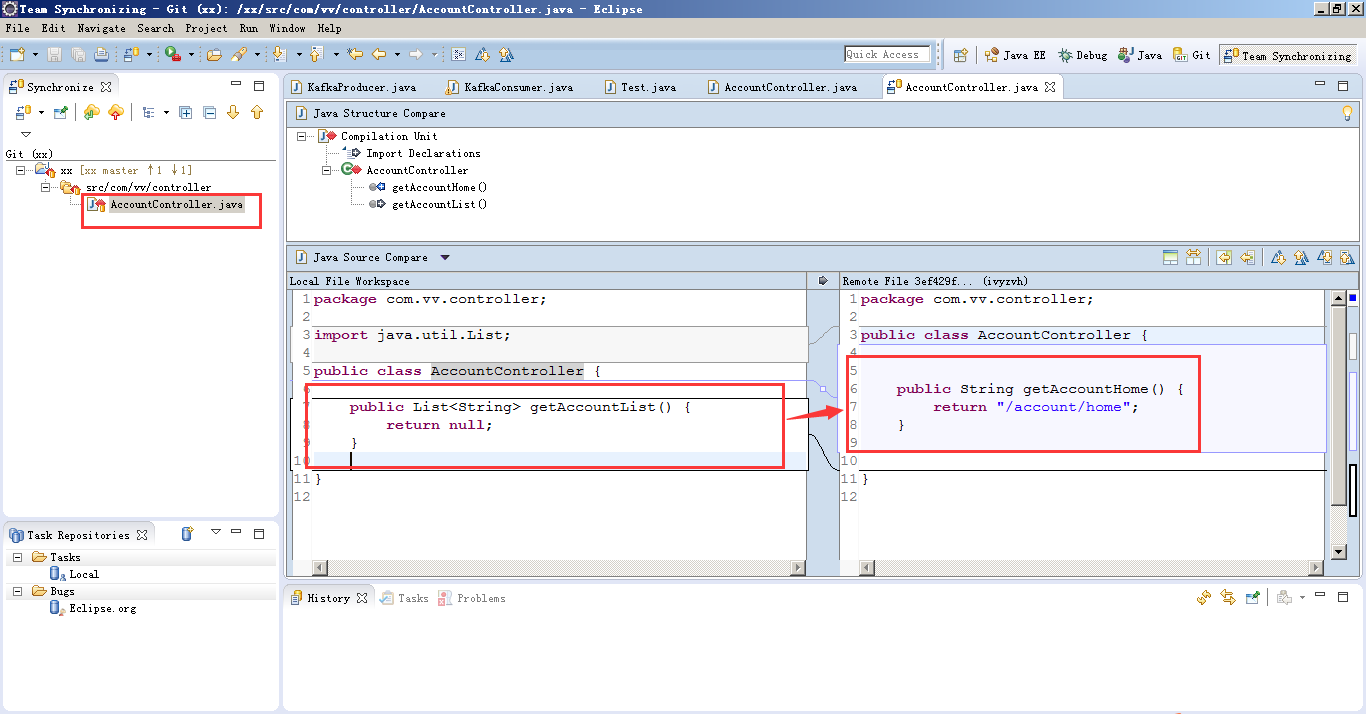
## 5.5 右键项目Team🡪Commit

## 5.6 右键项目Team🡪Remote🡪Push

完成：



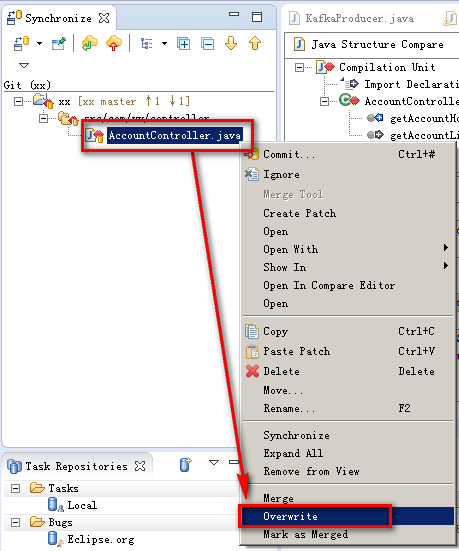
(1)项目右键🡪Team🡪Synchronize Workspace

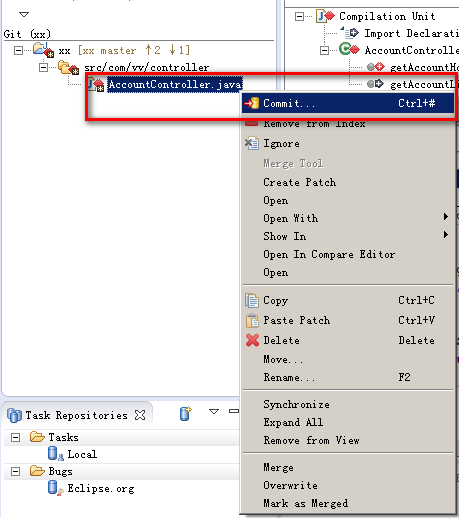


远程文件覆盖本地文件【小心使用，本地文件会被远程文件替换掉】

选择冲突文件🡪右键菜单🡪Overwrite🡪修改文件🡪Mark as Merged🡪完成冲突编辑合并

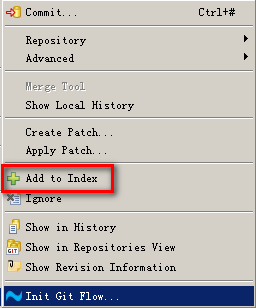
选择冲突文件🡪右键菜单🡪Commit🡪完成提交





【其它说明】

【Add to Index】将要提交的文件的信息添加到索引库中。



## 3 [Eclipse里git提交冲突rejected – non-fast-forward](http://www.cnblogs.com/wuxun1997/p/6952197.html)

　　Eclipse里commit代码，其实只是提交到本地仓库，需要push才会提交到远程的git仓库，这时是一个本地仓库到远程仓库的同步过程。Git是分布式的，每个人在本地仓库维护本地的自己的那一份代码，commit时只提交到本地仓库，不会影响到别人。当你push到远程仓库就可能发生冲突了。假如别人在你push前也push了同一个文件，那么问题就来了，冲突了，rejected – non-fast-forward出现了。

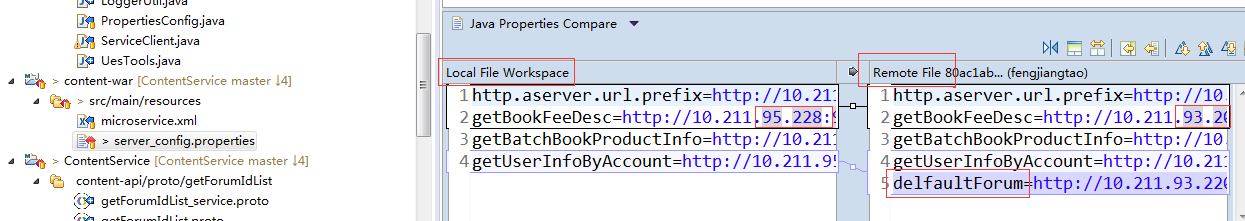
　　解决方法一：每次本地修改代码前先从git仓库pull下想要提交的分支的最新代码，改完了commit并push到git仓库。这里有个前提，修改、commit和push期间没人动你修改的文件。

　　解决方法二：期间文件在git仓库被改了，那么重新从git仓库获取最新的分支代码，然后把自己的修改放上去提交。

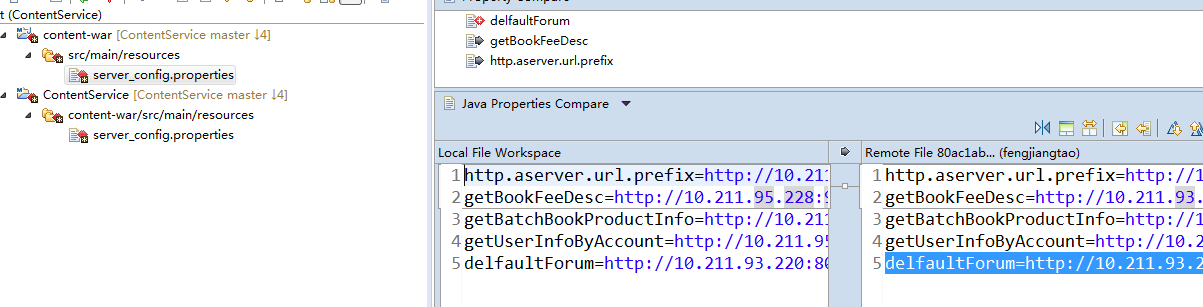
　　解决方法三：通过merge来解决冲突。当出现rejected – non-fast-forward别慌，这时我们可以识别哪些文件有冲突，一个一个改，改好了再pull下最新代码，然后提交到远程仓库。

　　下面具体说明下提交操作：

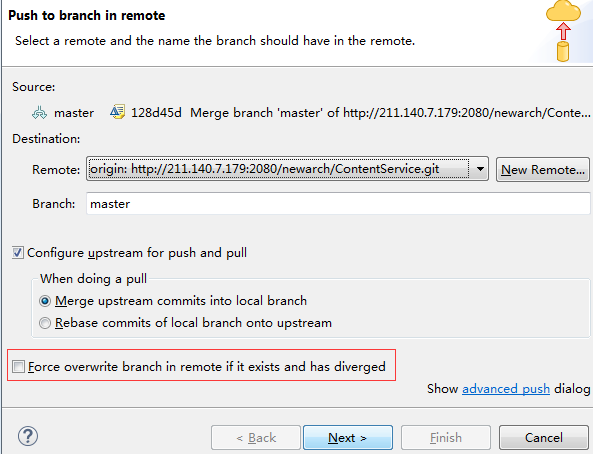
1、单个文件提交：首先跟远程仓库做比较，右键修改的文件和目录->Team->Synchronized Workspace->确保没有冲突后再次右键修改的文件和目录->commit，在Git Staging里选中Unstaged Changes窗口里要提交的文件->Add to Index->在Commit Messge里写提交说明->Commit and Push->Next，注意这里不要选最下面的Force overwrite branch in remote if exists and have diverged，否则本地仓库会覆盖远程仓库，造成覆盖别人代码的严重后果->OK

2、冲突解决：上面在Synchronized Workspace后出现冲突会有鲜艳的红色箭头出现，双击就可以看到哪个地方冲突了，一般是先把本地的改动保存好，从远程取下最新内容，再加上本地修改，最后提交。图示如下：

上面本地文件改动了，远程文件新增了，所以冲突了，右键选中该文件，点击Merge：



可以看到新增的那一条自动复制到本地来了，接着右键点击Commit，输入提交说明，提交到本地仓库。此时再Pull最新远程git仓库里的代码到本地仓库（冲突不解决Pull会报错，Push报rejected – non-fast-forward），然后Push即可：



注意不要点强推，很容易覆盖代码。