**准备工作：**

**目的：eclipse使用[Git](http://lib.csdn.net/base/git" \o "Git知识库" \t "_blank)提交本地项目，提交至远程github上**

**eclipse版本：eclipse4.5  64位**

**jdk版本：jdk-1.7 64位**

**项目类型：maven web项目**

**备注：eclipse高版本自带git插件，而低版本则需自己安装git插件，这里使用eclipse高版本，低版本安装git插件这里不进行介绍。**

**eclipse使用git提交项目有2种方式：一种是git命令窗口，另一种是eclipse自带git插件（跟svn插件一样使用）**

**一、使用git命令窗口提交项目**

1.首先官网下载git并安装，然后配置用户信息（任意目录下右击“Git Bash Here”打开git命令窗口）

git config --global user.name "you name"

git config --global user.email "you@youdomain.example.com"

2.登录github官网，注册个人github账号，创建SSH Key 用于将本地项目上传至远程github上

 创建SSH Key：ssh-keygen -t rsa -C "youemail@example.com"

把生成的公钥复制到github上，具体操作参考：[http://blog.csdn.net/u014079773/article/details/49555989](http://blog.csdn.net/u014079773/article/details/49555989" \t "_blank)  中“六：添加远程仓库”

3.创建本地仓库如“testGit”，然后选中该本地仓库右击“Git Bash Here” 执行如下命令:**(注意仓库名称用英文)**

首先在github上创建一个仓库，仓库名为“testGit”(远程仓库名必须与本地仓库名一致，否则提交失败，报仓库不存在)，创建远程仓库具体操作参考：

<http://blog.csdn.net/u014079773/article/details/49555989>  中“六：添加远程仓库”

在“testGit”工程上右击“Git bash Here”输入如下命令：

git init          初始化仓库“testGit”即把这个目录变成Git可以管理的仓库,在该目录下会生成一个隐藏文件“.git”

touch .gitignore    git提交忽略不必要的文件夹或文件（在本地仓库中生成一个隐藏文件“.gitignore”,文本编辑器打开设置要忽略的文件或文件夹如maven项目的target文件夹）

git add .       添加所有文件(add与“.”之间有空格，否则不识别的语法，提交所有文件)

git commit -a -m 'version 1.2'    提交所有修改文件

git remote add origin git@github.com:somenone/testGit.git      提交远程仓库(用SSH的形式，关联一个远程仓库)

git push -u origin master       推送远程仓库

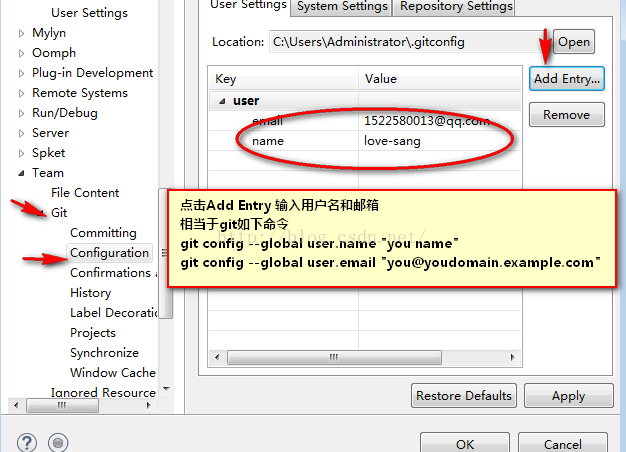
备注：由于远程库是空的，我们第一次推送master分支时，加上了-u参数，Git不但会把本地的master分支内容推送的远程新的master分支，还会把本地的master分支和远程的master分支关联起来，在以后的推送或者拉取时就可以简化命令。

之后执行就是 git push origin master

到此我们本地项目已经提交完成，登录远程github查看发现上传成功。

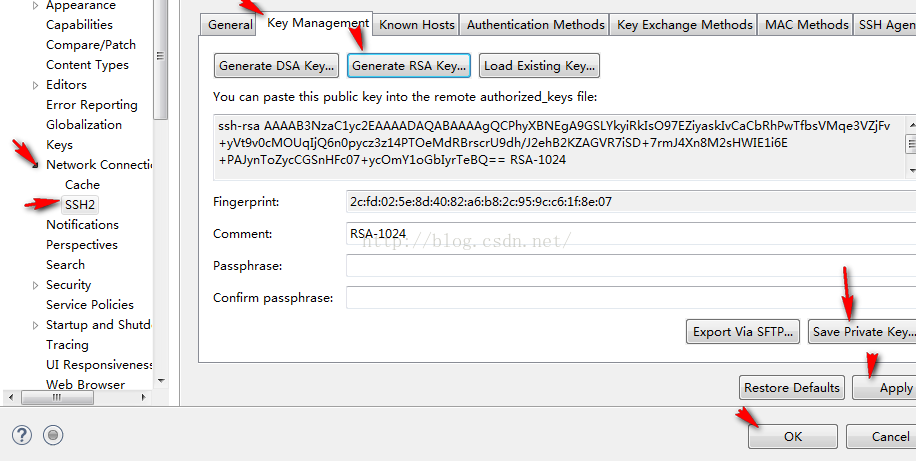
**二、使用eclipse自带插件提交项目**

1.自带git插件进行配置我们的用户名和密码，即是自己github注册用户。windows--perferences--Team--Git--Configuration



2.eclipse生成SSH2 key：windows -- perferences--General--Network Commection--SSH2--key  management--General

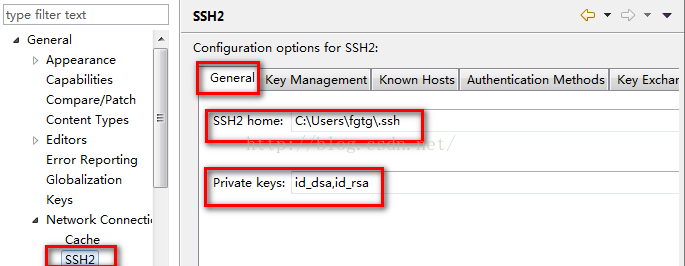
RSA Key



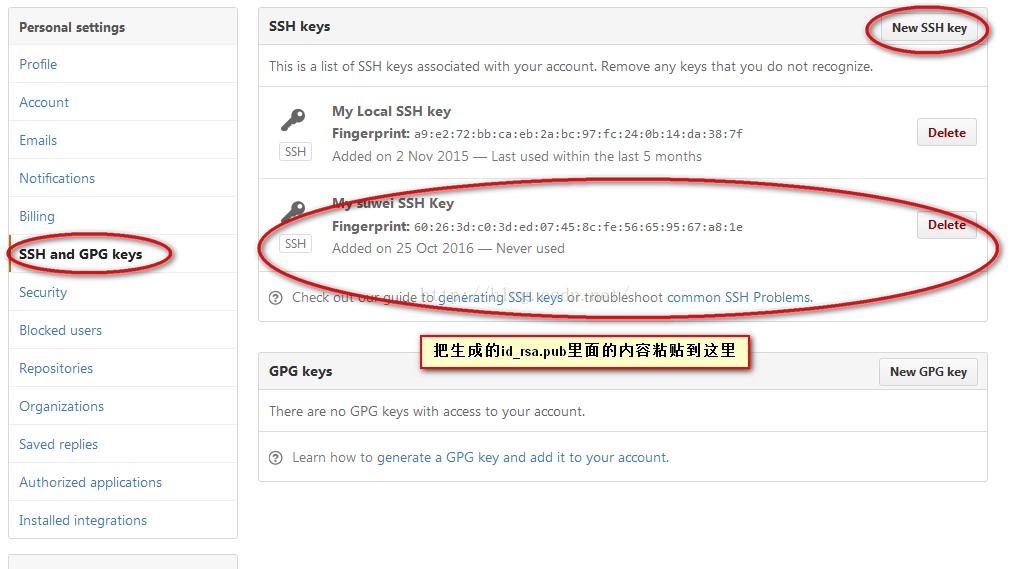
然后把生成的SSH Key 注册到github上，登录github--settings--add把上诉生成的SSH Key粘贴即可。

备注：该方法就相当于命令行方法，在公司电脑和自己笔记本电脑把生成的SSH Key粘贴即可，可以添加很多个SSH Key

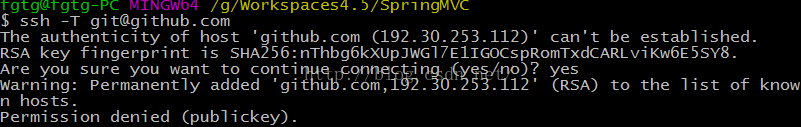
查看生成的秘钥是否在在默认系统盘（C盘），有些人奇葩会安装在其他盘。



然后把生成的公钥粘贴到github上：**（该步不能省略，一定要用命令检查下生成的公钥是否正常）**

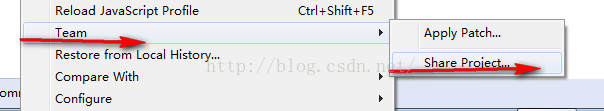


若该步没有操作，执行命令：$ ssh -T git@github.com报这样的错误：

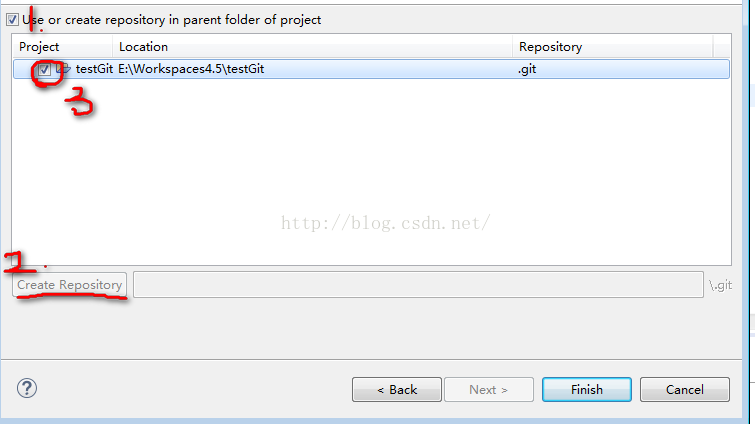


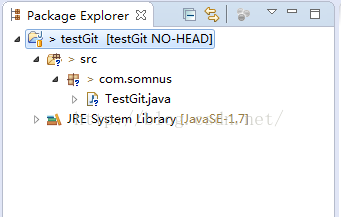
3.本地创建项目如**“testGit**”和本地git仓库

打开eclipse，创建项目“testGit”里面随便写点什么,然后"File"--"Team"--"Share Project"--"Git"--"next"



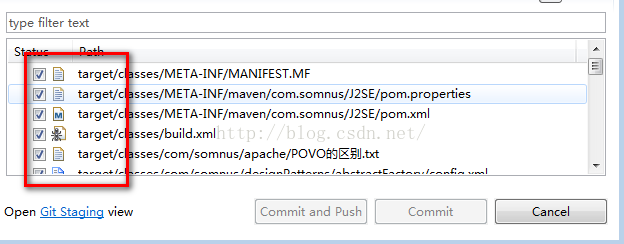




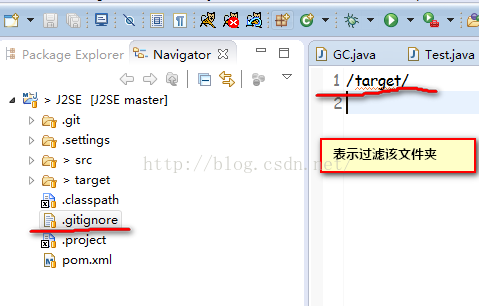


如上图所示到此该项目“testGit”就变成git仓库，到workspace下打开该项目，发现多了一个**隐藏文件“.git”,**但该项目处于untracked状态（文件夹中的符号”?”表示），下面我们需要提交代码到本地仓库，如下图：

首先我们要忽略不必要的文件如maven项目target文件夹



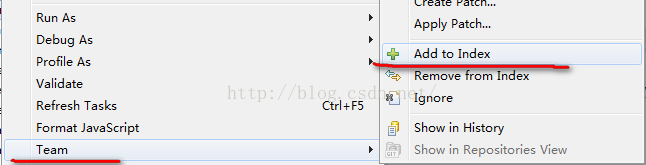
配置.gitignore来排除这个文件夹，打开Navigator窗口，在project根目录中添加.gitignore文件，将需要排除控制的目录写入.gitignore文件中：



这样git提交就过滤了一些不必要的文件，若是要过滤其他文件夹或文件直接在“.gitignore”中写，具体写法参考：

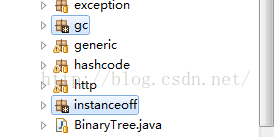
[http://blog.csdn](http://blog.csdn.net/u014079773/article/details/51602344)[**.NET**](http://lib.csdn.net/base/dotnet)/u014079773/article/details/51602344

然后提交修改的文件必须先add然后commit故通过Team -> Add to index可以将文件加入git索引，进行版本监控：

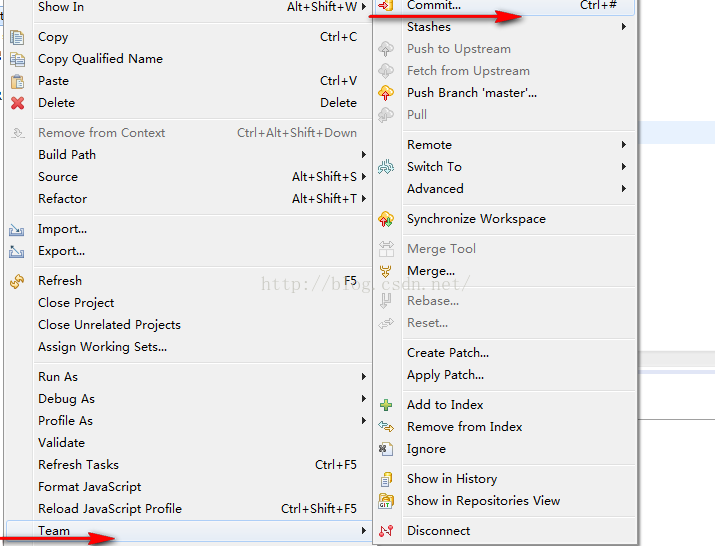


若是不想添加到版本库中则“Remove from Index”该命令前提是为未commit，若是已经commit那么该命令不起作用。添加成功后文件状态发生变化由之前的“？”变成“\*”：

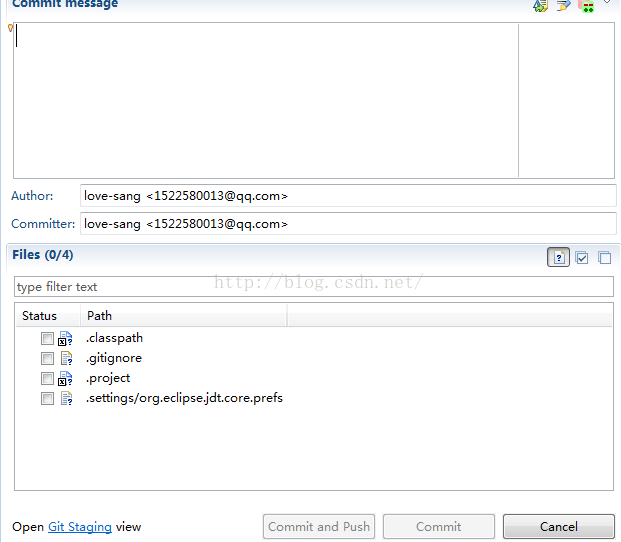
**细节：EGIT中只要Commit就可以默认将untracked的文件添加到索引再提交更新，不需要分开操作**



最后执行commit：（首次提交后，会自动生成master分支）

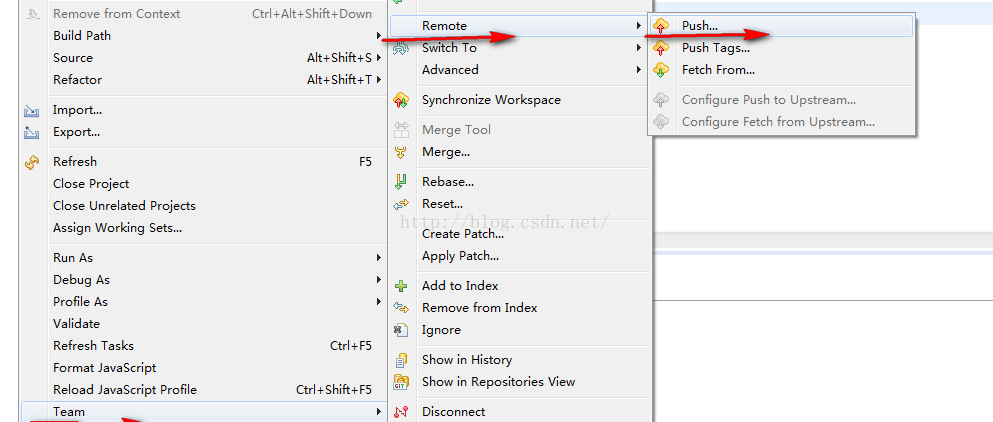


在注释中输入提交信息：

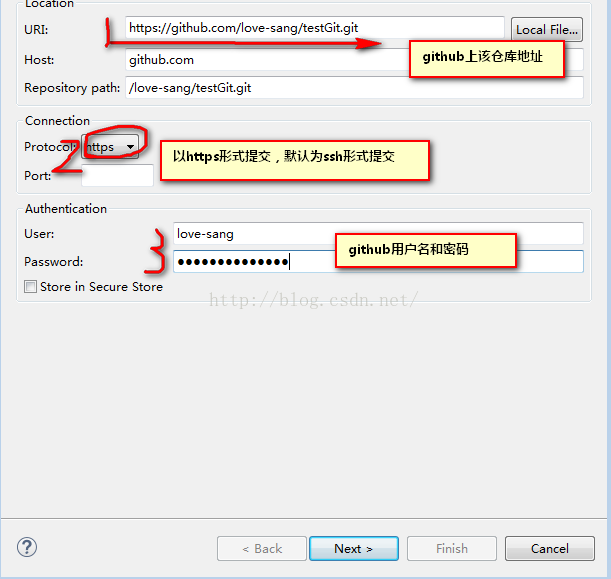


到此项目“testGit”就本地提交成功。接下来我们把本地项目提交至远程远程仓库github上：

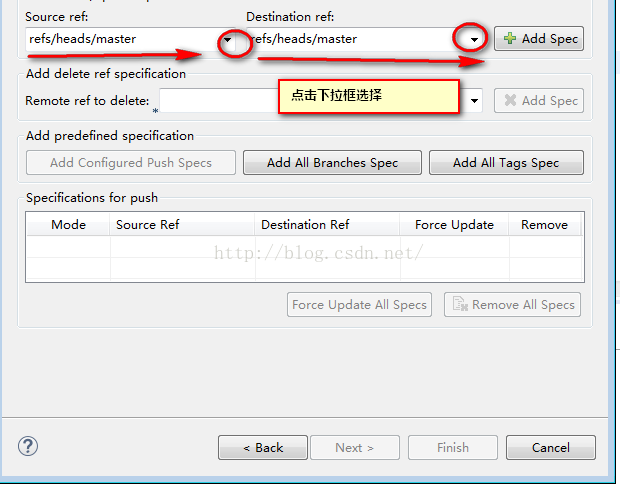
首先在github官网上登录并创建仓库“testGit”（该名称与本地仓库名称一致，表示本地仓库提交到远程仓库并与之关联），创建完毕后执行如下操作：



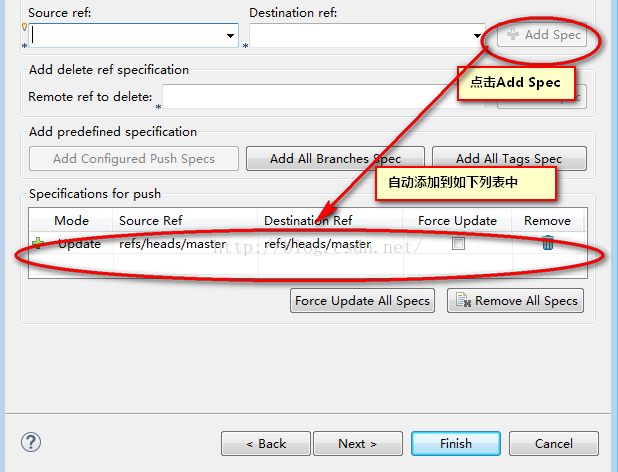
进行如下设置：



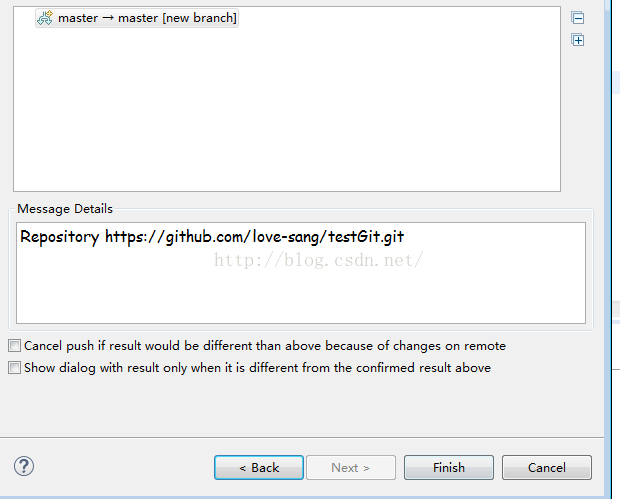
下拉框选择如下：



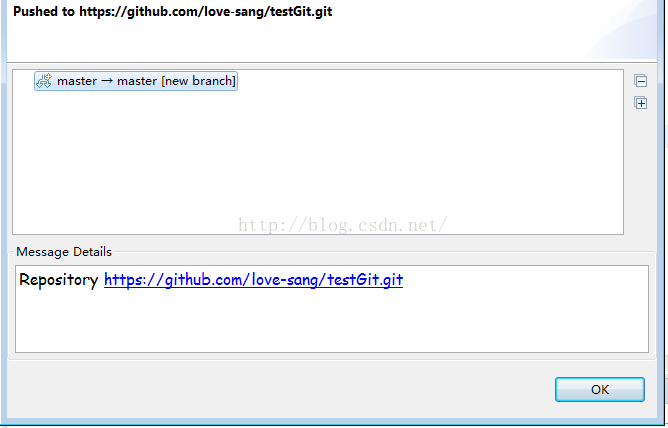
然后进入如下截图操作：**注意：“Force Update”若是勾选，表示覆盖提交**



点击“next”出现如下图所示：



最后点击finis如下图，登录github查看是否上传成功。

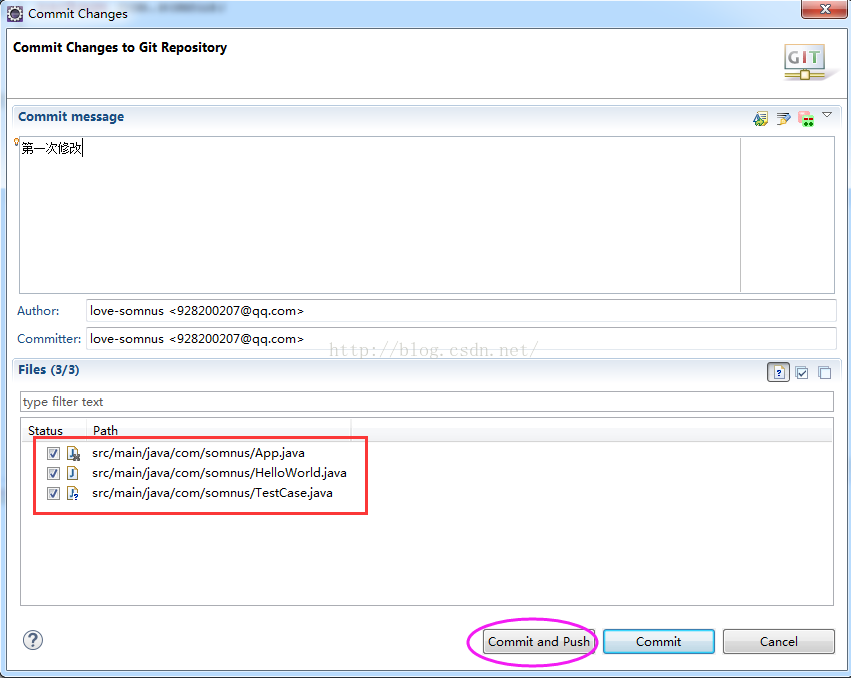


出现如下截图表示上传成功。



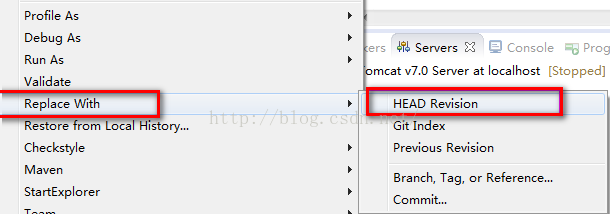
补充：

1.如何提交已经修改的项目，已修改的项目文件夹出现“\*”号标志，选择带“\*”号标志的文件右击“Team”--"commit"

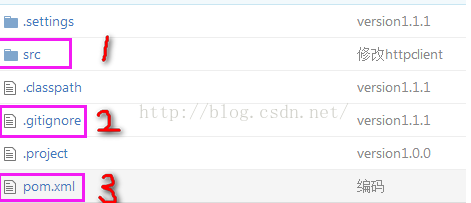


点击“commit and push”表示提交并推送，即把修改的文件提交至本地，推送到远程仓库。

2.如何恢复未提交的修改文件：



3.实际开发中提交项目只需要提交如下三个文件即可：



4.如何用命令查看，修改remote url：

如果你在连接github仓库的时候，用的https协议的地址，还需要切换为git协议  
可以用git remote -v 查看你当前的remote url

$ git remote -v  
origin https://github.com/love-somnus/[**spring**](http://lib.csdn.net/base/javaee).git (fetch)  
origin https://github.com/love-somnus/Spring.git (push)

可以看到是使用https协议进行访问的。

这时，你可以使用 git remote set-url 来调整你的url。

git remote set-url origin git@github.com:love-somnus/Spring.git

完了之后，你便可以再用 git remote -v 查看一下。至此OK。

5.查看本地SSH Key是否正常：

$ ssh -T git@github.com  
# Attempts to ssh to github

出现如下信息表示设置成功：

Hi username! You've successfully authenticated, but GitHub does not # provide shell access.

**备注：**

**1.使用eclipse插件提交项目至远程必须先本地提交，最后提交至远程仓库**

**2.查看谁提交了代码，项目右击“team”--“show history”**

**3.查看提交记录：Team--Show in History   (命令行为：git log)**

**4.撤销恢复修改：Replace With--HEAD Revision (命令行为 git checkout head .)**

**5.更新项目远程仓库：Team-- Pull (命令行：git pull https://github.com/someone/xxx.git)**

**6.查看哪些文件被修改了：Team--Synchronize Workspace**

**7.提交项目时注意忽略不必要的文件或文件夹如maven项目下的target文件夹忽略**

**8.git提供了多重协议用来连接git服务器，其中最常见的就是https和git，git就是可以免用户名和密码的，不同的协议，项目连接地址是有区别的，如下**

https形式：https://github.com/love-somnus/Spring.git  git形式：git@github.com:love-somnus/Spring.git