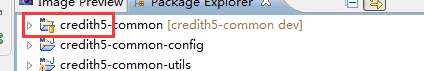
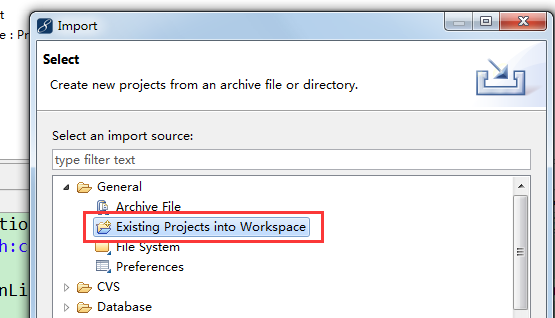
1.Myeclipse导入已存在的git仓库中的项目，使其关联

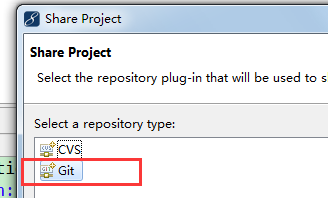
1. 本地已有的git仓库中的项目，想把它导入MyEclipse中，并使其关联（就是有仓库的小图标）

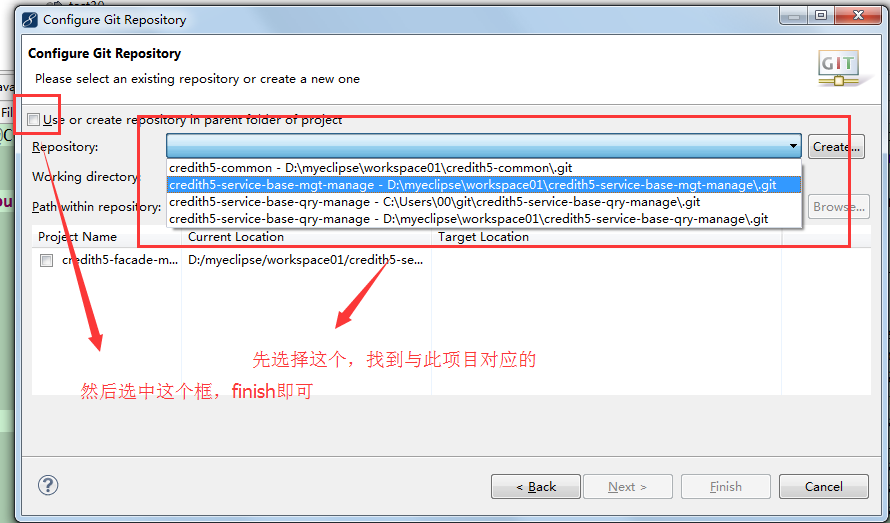


1. 先这样直接导入项目即可（最普通的方式导入）



1. 然后右键导入的项目 Team🡪Share Project..-->选git 点击next （如下图）



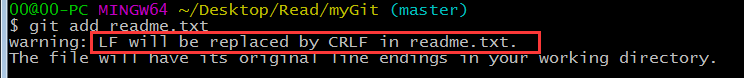
1. 按照图中步骤操作即可
2. 

# 2.git 命令

git stash –提交的时候先隐藏本地文件

git pull

git stash pop –将隐藏的显示出来然后看文件

1. git status --查看仓库状态，看看那些文件修改了
2. git log --查看操作日志
3. git reflog --查看所有的操作日志，可以用于回到未来的版本
4. git commit -m “对提交的描述”
5. git push --推送到远程
6. git diff --查看不同
7. git reset --hard 后面跟commit id 前几位即可 --重置为某个版本
8. git merge命令用于合并指定分支到当前分支
9. 查看分支：git branch
10. 创建分支：git branch <name>
11. 切换分支：git checkout <name>
12. 创建+切换分支：git checkout -b <name>
13. 合并某分支到当前分支：git merge <name>
14. 删除分支：git branch -d <name>
15. 当Git无法自动合并分支时，就必须首先解决冲突。解决冲突后，再提交，合并完成。
16. 用git log --graph命令可以看到分支合并图。
17. 合并分支时，加上--no-ff参数就可以用普通模式合并，合并后的历史有分支，能看出来曾经做过合并，而fast forward合并就看不出来曾经做过合并。
18. 如果要丢弃一个没有被合并过的分支，可以通过git branch -D <name>强行删除。
19. 现在，你的小伙伴要在dev分支上开发，就必须创建远程origin的dev分支到本地，于是他用这个命令创建本地dev分支：
20. $ git checkout -b dev origin/dev
21. git pull也失败了，原因是没有指定本地dev分支与远程origin/dev分支的链接，根据提示，设置dev和origin/dev的链接：
22. $ git branch *--set-upstream dev origin/dev*
23. 如果git pull提示“no tracking information”，则说明本地分支和远程分支的链接关系没有创建，用命令git branch --set-upstream branch-name origin/branch-name。这就是多人协作的工作模式，一旦熟悉了，就非常简单
24. 要关联一个远程库，使用命令git remote add origin [git@server-name:path/repo-name.git](mailto:git@server-name:path/repo-name.git)；
25. 关联后，使用命令git push -u origin master第一次推送master分支的所有内容
26. 此后，每次本地提交后，只要有必要，就可以使用命令git push origin master推送最新修改
27. git push origin local\_branch:remote\_branch 例：将本地新建的分支dev推送到远程，如果远程没有此分支就新建一个dev（用此让远程建立分支）
28. 这个操作，local\_branch必须为你本地存在的分支，remote\_branch为远程分支，如果remote\_branch不存在则会自动创建分支。
29. 
30. 

**以上问题解决办法：git config core.autocrlf false**

1. 删除远程分支
2. git branch -r -d origin/dev
3. git push origin :dev

&lt; 是< [less](javascript:void(0);) [than](javascript:void(0);)

&gt; 是> [greater](javascript:void(0);) [than](javascript:void(0);)

# 3.github账号

guomaoyangeheh

[229406891@qq.com](mailto:229406891@qq.com)

886xiaolaji

# 4.开发问题

1. <http://localhost:8080/credith5-web-boss/swagger/index.html> **swagger地址**

**2.** <http://blog.csdn.net/linlzk/article/details/50728264> **swagger 整合springmvc 教程**

**Git release 111**

**3.配置tomcat配置文件解决get乱码问题**



4.单元测试时要使用这个东西判断，（涨涨姿势）

# 5.当git识别不了新增文件时

方法：

git add 1.cpp

返回无法添加，说1.cpp已经处于.gitignore中。但是iganore中明明没有忽略这些cpp文件。

解决方法：

git add -f 1.cpp

这样就可以将1.cpp添加到git的版本控制中。

如果直接git add -f 1.cpp

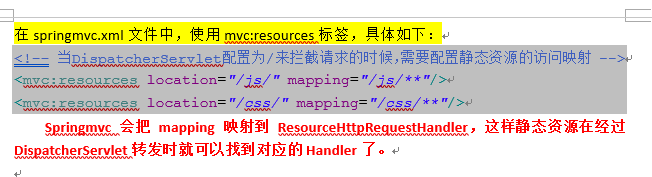
报错，无法找到文件1.cpp。

那么就将1.cpp的完整路径输入即可。比如D:/jingmai\_git/TradePlugin/src/Trade/1.cpp就可以了。

1. 首先先看一下文件中是否忽略掉了该文件，或者用以上方法解决
2. 异常处理，在service层处理异常，抛至controller层，在controller层配置了ExceptionHandler捕获到异常，在web层设置了一个

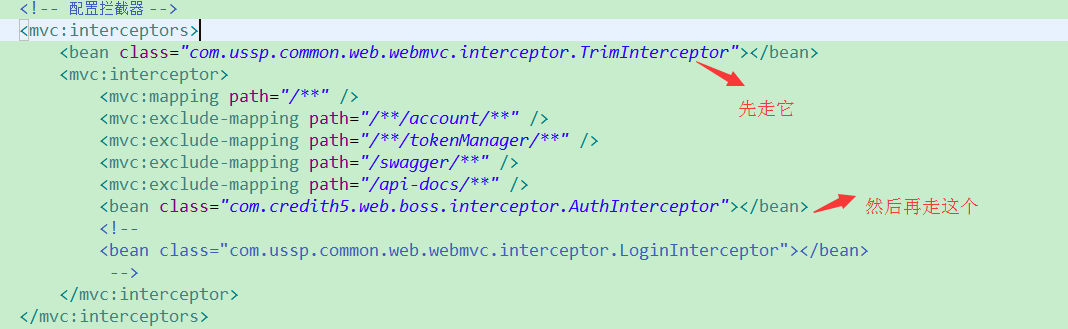
**异常处理思路，dao层抛至service层，service层抛至web层在web层设置异常处理器，所有的处理器都会继承它，通过注解@ExceptionHandler 可以抓取到抛至处理器中的异常，日志输出异常信息，并将错误信息返还至前端。**

**Spring MVC使用：**



**Spring MVC中的拦截器是实现HandlerIntercept接口实现的**





**Public class** HandlerInterceptor1 **implements** HandlerInterceptor{

/\*\*

\* controller执行前调用此方法

\* 返回true表示继续执行，返回false中止执行

\* 这里可以加入登录校验、权限拦截等

\*/

@Override

**Public boolean** preHandle(HttpServletRequest request,

HttpServletResponse response, Object handler) **throws** Exception {

**O** Auto-generated method stub

**Return false**;

}

/\*\*

\* controller执行后但未返回视图前调用此方法

\* 这里可在返回用户前对模型数据进行加工处理，比如这里加入公用信息以便页面显示

\*/

@Override

**Public void** postHandle(HttpServletRequest request,

HttpServletResponse response, Object handler,

ModelAndView modelAndView) **throws** Exception {

}

/\*\*

\* controller执行后且视图返回后调用此方法

\* 这里可得到执行controller时的异常信息

\* 这里可记录操作日志，资源清理等

\*/

@Override

**Public void** afterCompletion(HttpServletRequest request,

HttpServletResponse response, Object handler, Exception ex)

**throws** Exception {

}

}

# Checked异常一Unchecked异常(CreditH5项目中异常问题)

除了Error与RuntimeException，其他剩下的异常都是你需要关心的，而这些异常类统称为Checked Exception,至于Error与RuntimeException则被统称为Unchecked Exception.

Error和RuntimeException及其子类是unchecked exception.其他exception是checked exception

Error和RuntimeException属于unchecked（不受检查）异常，其余继承Exception的异常属于checked（受检查）异常

CreditH5项目中遇到的在service层抛异常被Dubbo的ExceptionFilter拦截Controller层抓取到的是时候该异常被包了一层RuntimeException

**详细见：**

<http://blog.csdn.net/mj158518/article/details/51228649>

**御魂**

<http://www.18183.com/yys/201610/725955.html>

# 7.Http协议详解

<http://www.ruanyifeng.com/blog/2016/08/http.html>