

JEECG-P3 插件开发环境

Maven 版本手把手入门手册

2016/05/23

www.jeecg.org

Jeecg 社区

官方标准开发工具：

1.IDE

Eclipse Java EE IDE for Web Developers.

Version: Mars.2 Release (4.5.2)

Build id: 20160218-06002.JDK1.7

3.Tomcat7

4.Maven 项目构建

5.Mysql

6.执行 Mysql 脚本

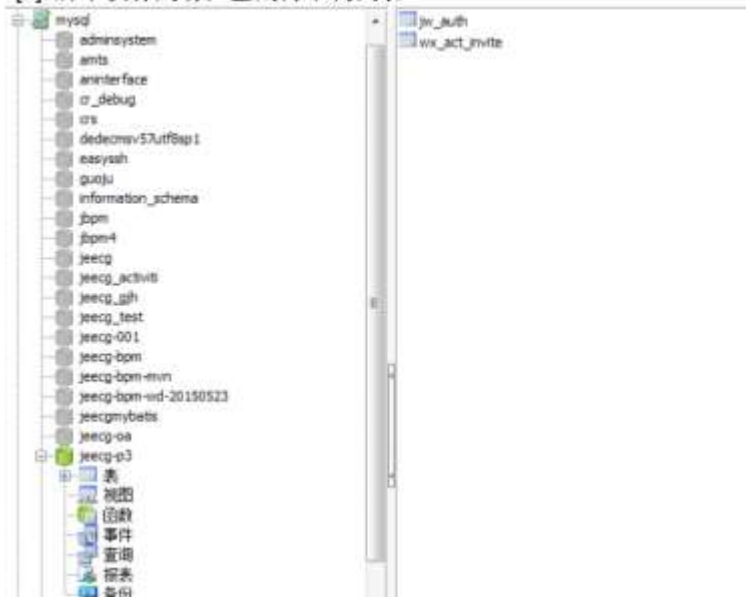
首先创建 Mysql 数据库 jeecg-p3，采用 UTF-8 编码，执行对应插件的数据库脚本

脚本位置： {插件项目名}\doc\db\{插件项目名}-mysql.sql

[1].注意编码选择：

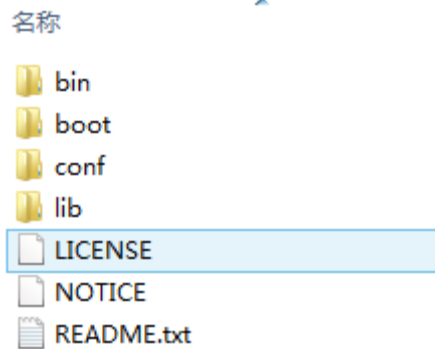


[2].脚本执行成功，生成两个测试表

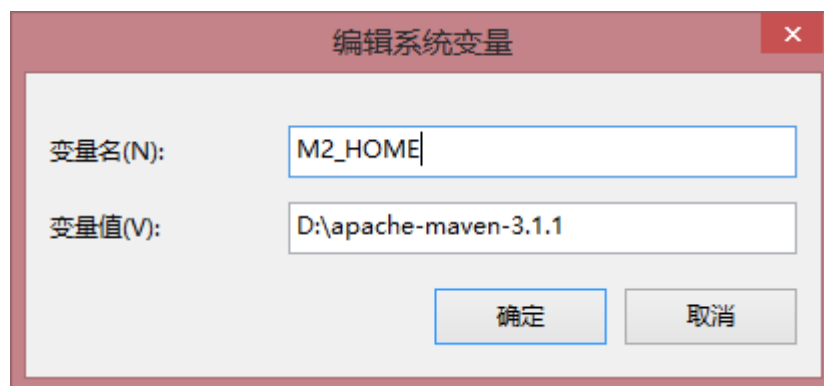


一. Maven 环境搭建

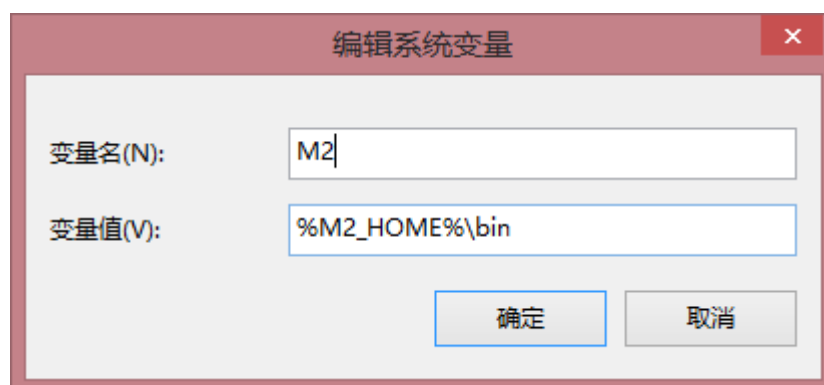
1. 下载: 到 Maven 官方网站 <http://maven.apache.org/download.cgi> 下载目前的最新版本 [apache-maven-3.1.1-bin.zip](#)
2. 解压: 下载到本地的 zip 文件, 只需解压成 X:\apache-maven-3.1.1 即完成安装, 目录结构为如下



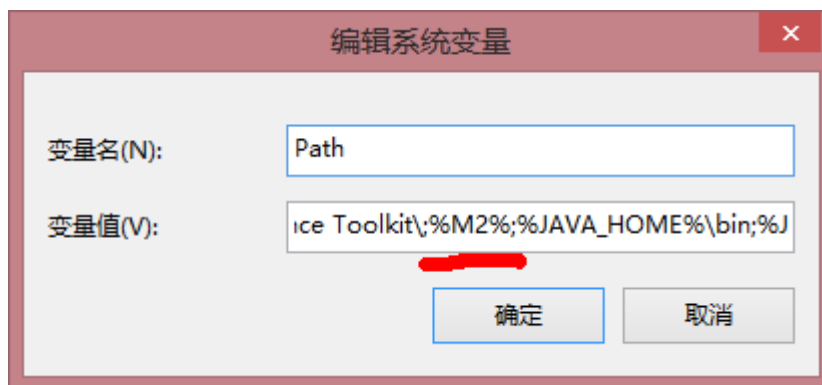
3. 配置环境变量: 这台电脑->鼠标右键->属性->高级环境设置->环境变量-系统变量框下新增变量名 M2_HOME, 值为 X:\apache-maven-3.1.1



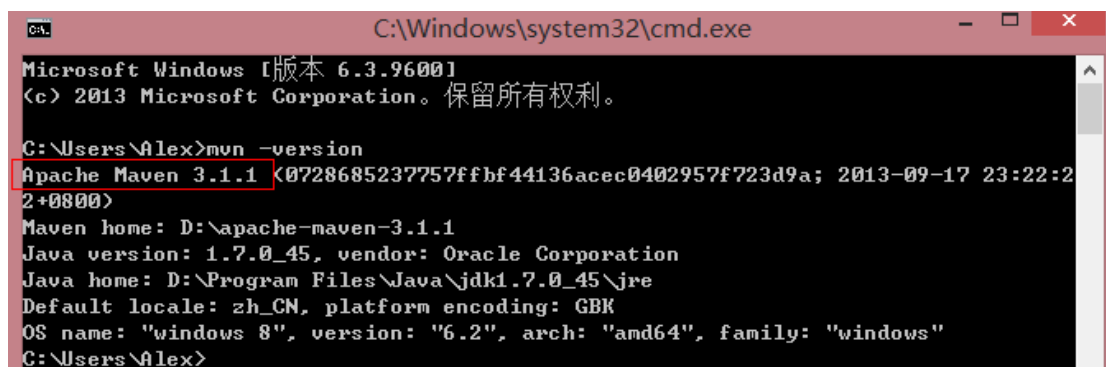
再新建变量 M2, 值为 %M2_HOME%\bin



编辑变量名 Path, 在变量值中后面追加"; %M2%"



4. 测试安装: win+R ->cmd ->在命令行中输入 mvn -version 显示 mvn 版本信息为 Apache Maven 3.1.1 则为安装成功。



5. 修改 maven 仓库配置

[1]、修改本地 maven 仓库位置

找到 maven 下的 conf 下的 settings.xml 配置文件，我的是在 D:\apache-maven-3.1.1\conf\settings.xml。maven 的仓库默认是放在本地用户的临时文件夹下面的 .m2 文件夹下的 repository 下，这很影响 C 盘的空间。现在我们来修改将它指定到我们自己的路径下，我现在要将仓库指定到 D:\Maven\repo\m2（可以自己定义位置）目录下。

```

<!--
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0 http://maven.apache.org/xsd/maven-settings-1.0.0.xsd">
  <!-- localRepository
  | The path to the local repository maven will use to store artifacts.
  |
  | Default: ${user.home}/.m2/repository
  <localRepository>path/to/local/repo</localRepository>
-->
<localRepository>D:/Maven/repo/m2</localRepository>
<!-- interactiveMode
  | This will determine whether maven prompts you when it needs input. If set to false,
  | maven will use a sensible default value, perhaps based on some other setting, for
  | the --interactive option.
-->
-->
</settings>

```

- [2]、配置官方 JeeCG 的方式 maven 镜像（非常重要）

修改 settings.xml 配置文件

在<mirrors>标签内增加阿里云 maven 镜像，同时排除 jeecg 私服，最终结果见

下面：

```
<mirrors>

  <mirror>

    <id>nexus-aliyun</id>

    <mirrorOf>*,!jeecg,!jeecg-snapshots</mirrorOf>

    <name>Nexus aliyun</name>

    <url>http://maven.aliyun.com/nexus/content/groups/public</url>

  </mirror>

</mirrors>
```

[3]、官方 Jeecg 提供的本地仓库下载

如果因为网络等问题导致依赖下载总失败，可以直接下载官方提供的本地仓库覆盖

本地仓库。下载地址：<https://gitee.com/jeecg/jeecg-local-maven>

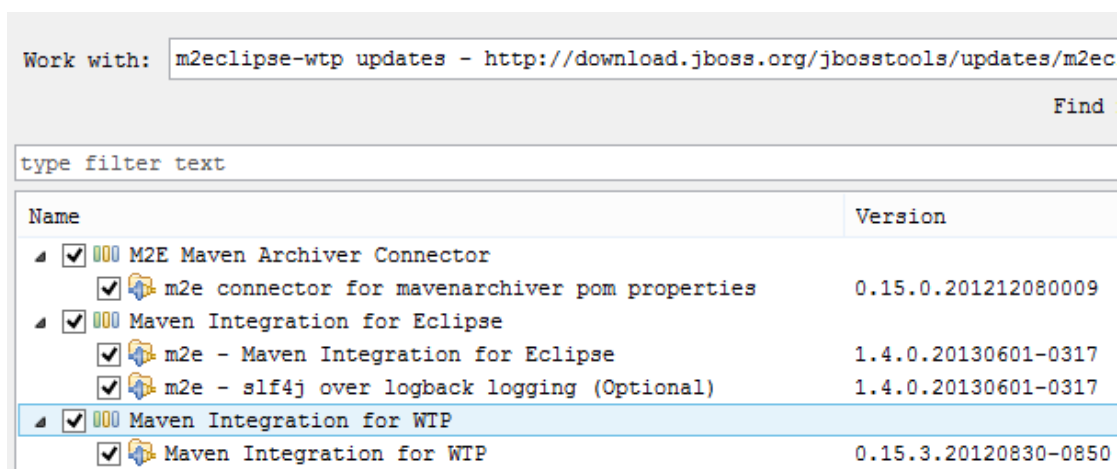
6. OK，先来体会一下 maven，在 cmd 中敲并回车执行：mvnhelp:system 这时候 maven 就会从远程仓库开始下载一大堆的东西，没事，让它下载着，迟早都要下载的。

二. MyEclipse 或 Eclipse 环境搭建

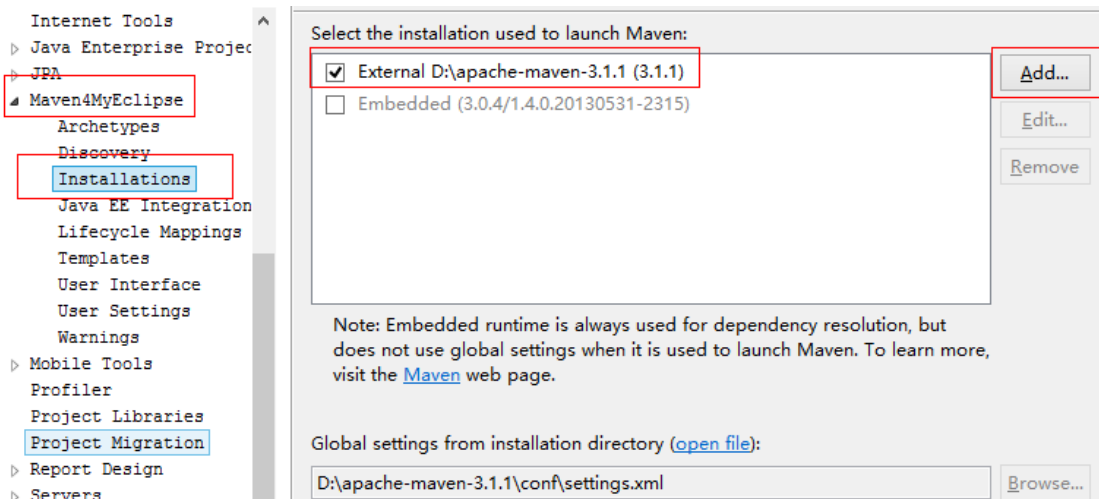
1. 插件安装：Eclipse Mars.2 和 MyEclipse8 到 MyEclipse2014 已经自带 maven 插件，此步可以略过。
2. 在线安装：选择 Help --> Install New Software 输入地址

<http://download.jboss.org/jbosstools/updates/m2eclipse-wtp/>，把选项

勾上，然后等待它下载安装，完成之后重启 eclipse 即可

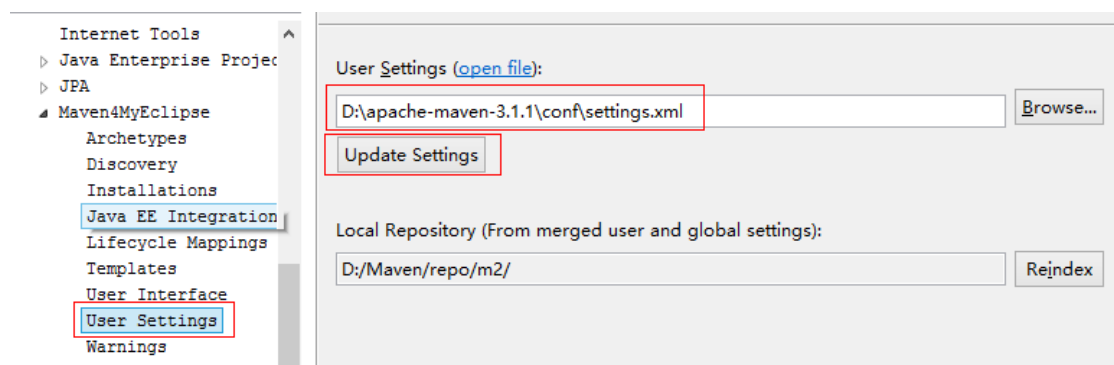


3. 配置 eclipse：找到 Maven4MyEclipse 下的 Installations，添加 maven 的安装目录



找到 User Settings 选择 maven 安装目录下的 settings.xml , 单击 Update

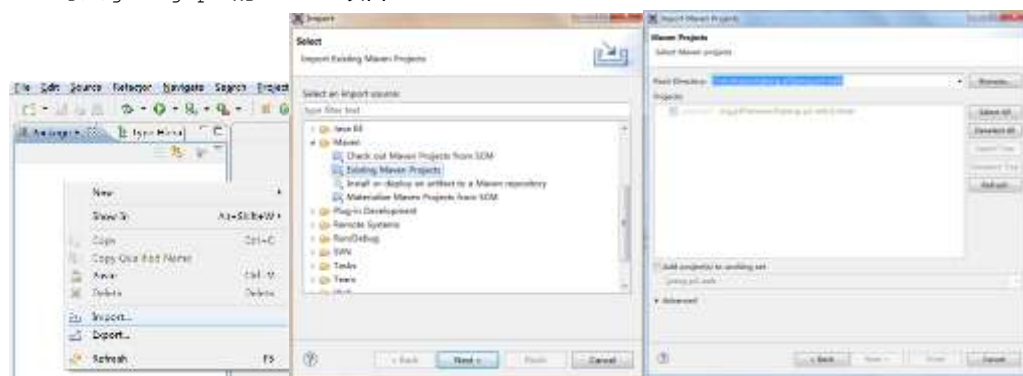
Settings.



OK , eclipse 配置已经完成了。

三. 导入和配置 jee cg-p3 的 maven 项目

1. 导入 jee cg-p3 的 maven 项目

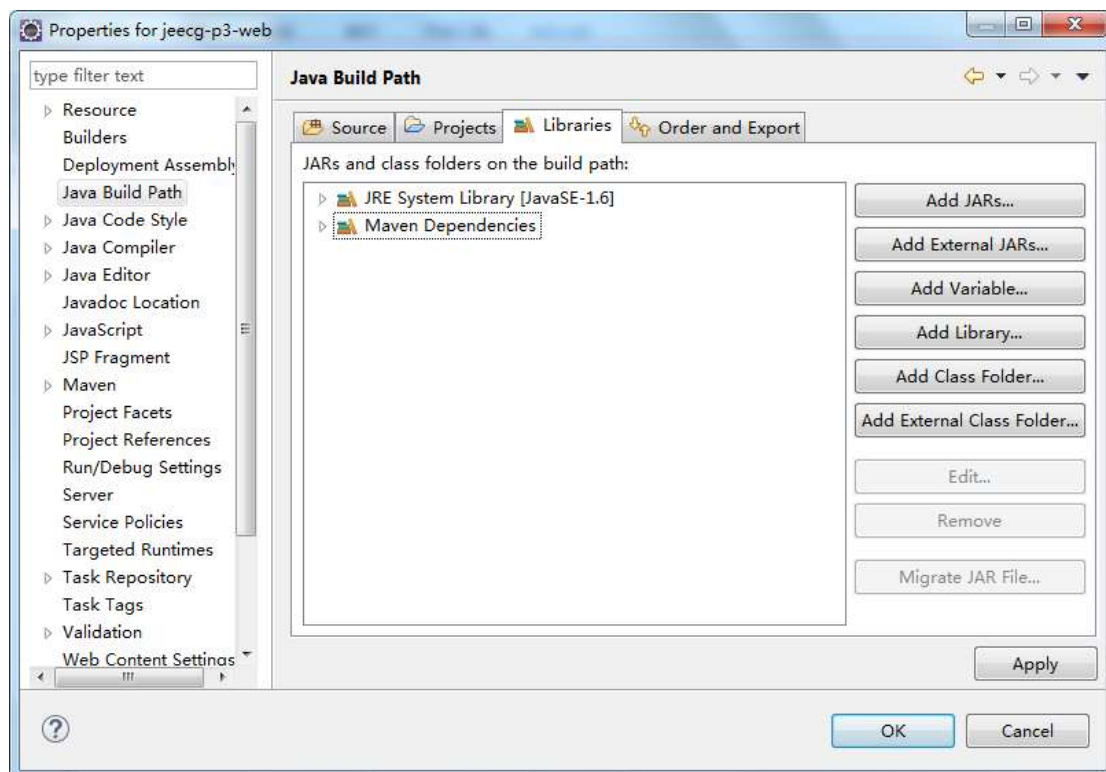


2. 加载 jee cg-p3 项目

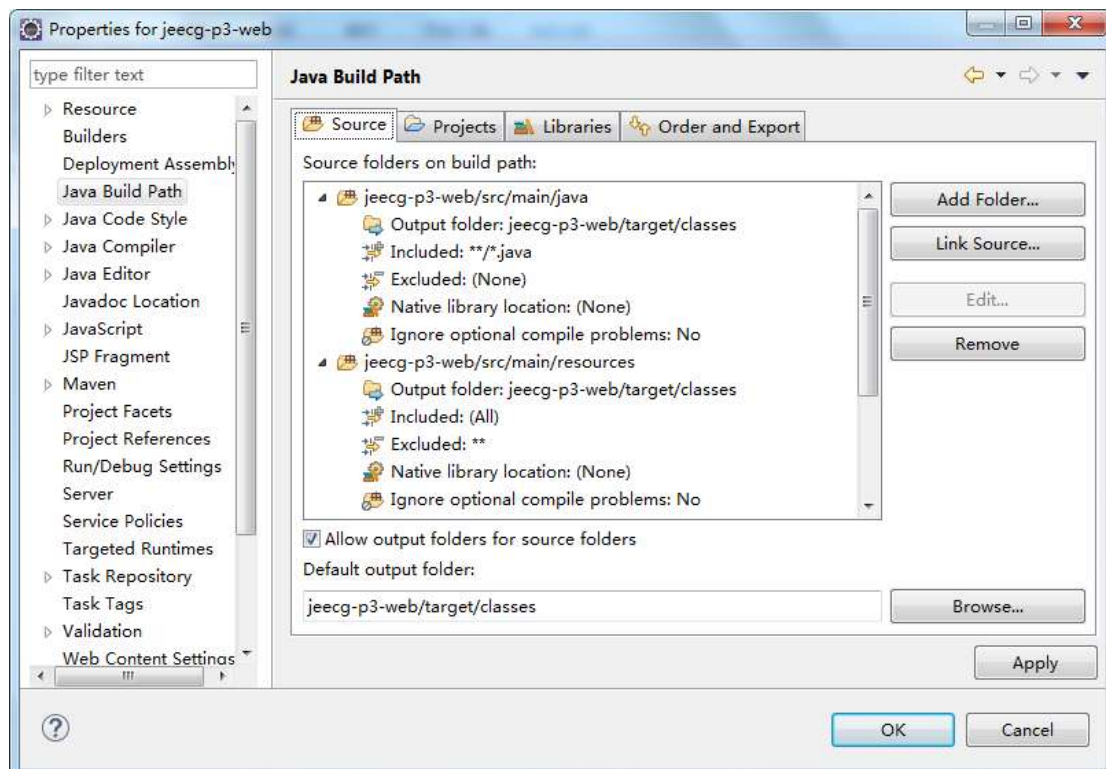
加载 jee cg-p3 项目到 MyEclipse 后 , 选择项目名右键选择 RunAS , 再选择 Maveninstall , 控制台就开始下载项目所需要的包。

3. Java Build Path

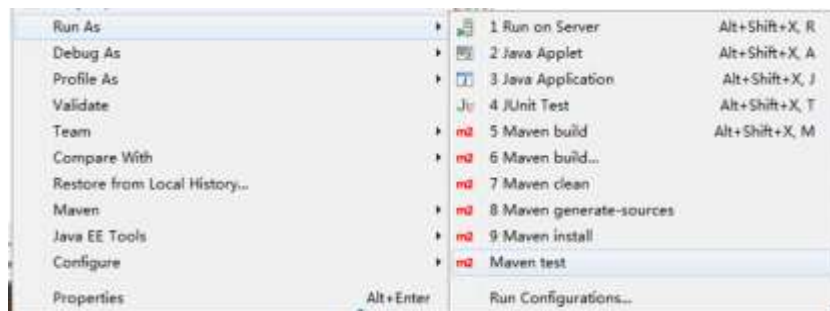
Libraries 如下



Source 如下:



4. 运行 maven 测试
工程上右键:

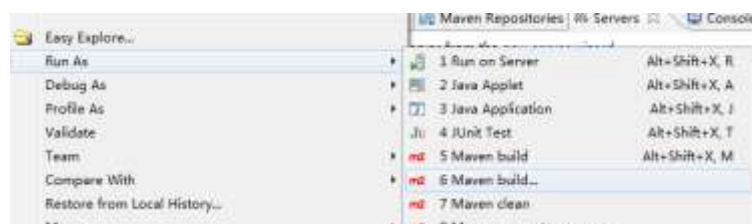


控制台没有错误出现就 ok 啦，下一步介绍 maven 项目的启动。

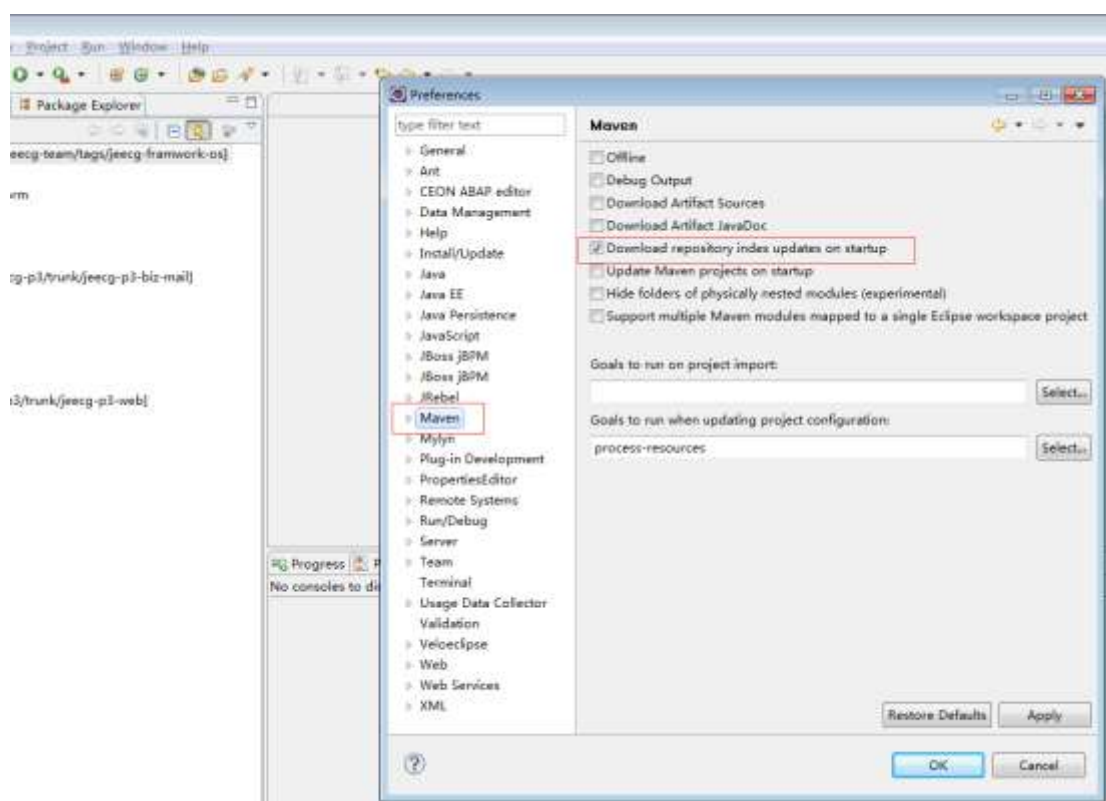
四. 启动 maven 项目

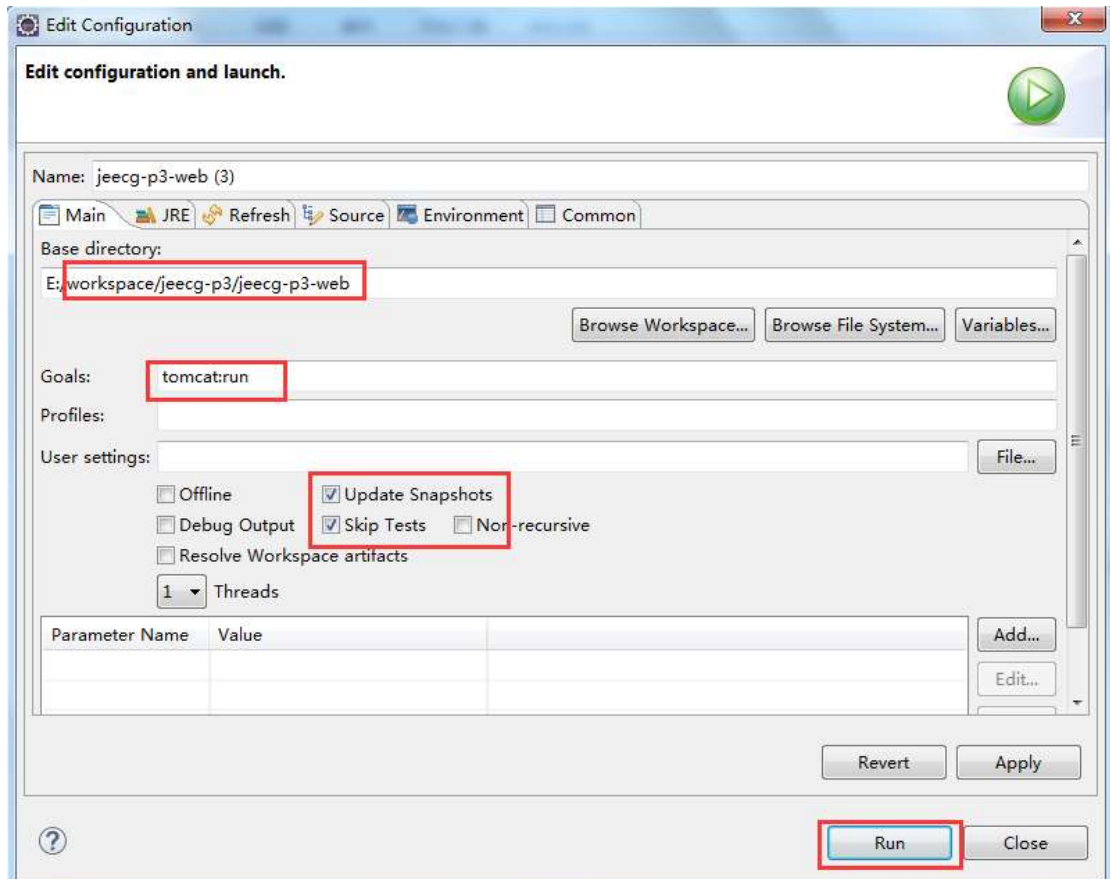
使用 maven 的 build 的方式启动

工程右键，运行 mavenbuild:



配置运行服务的环境变量:





点击 **Run** 按钮即可以启动应用。

浏览器访问：

测试访问地址如下（注意：其他地址可能无效）：

1. <http://localhost/jeecg-p3-web/p3/wxActInvite.do?list>
2. <http://localhost/jeecg-p3-web/p3/auth.do?list>

五. 截图演示

wxActInvite.do?list 效果截图：

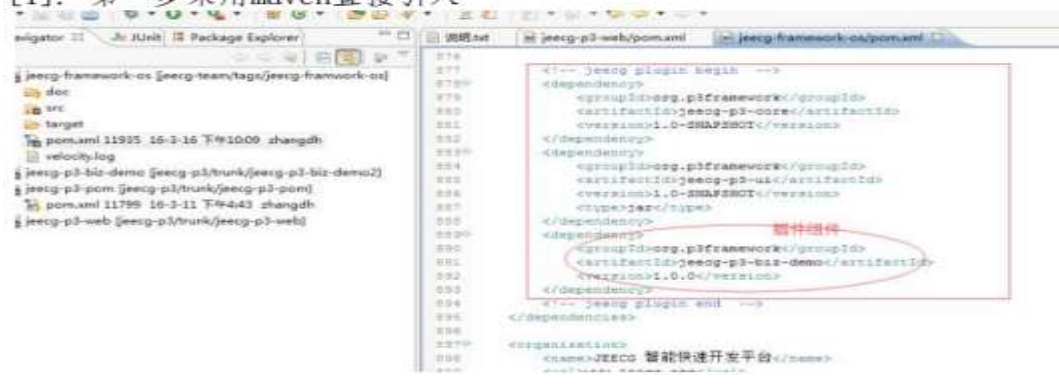


auth.do?list 效果截图：



六. Jeecg 项目集成插件方法

[1]. 第一步采用maven直接引入



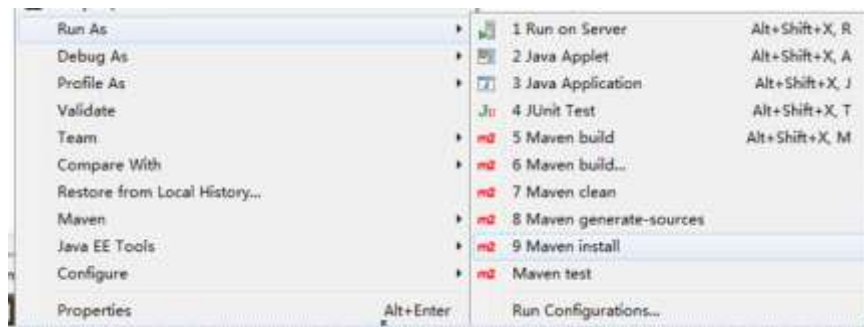
[2]. 在jeecg库，执行插件SQL脚本

[3]. 配置插件功能访问菜单

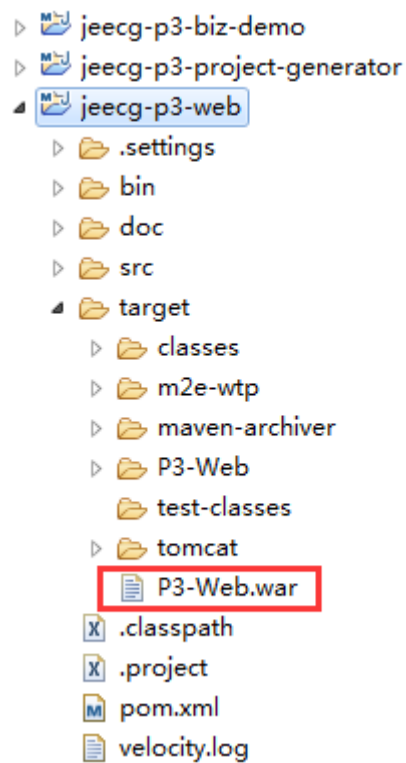


七. 发布 war 包

工程右键:



编译通过后, target 目录下会生成 P3-Web.war 包:



拷贝 P3-Web.war 到 tomcat 的 webapps 下面, 然后运行 tomcat 即可。