系统开发环境搭建手册

# 1. 准备工作

1.1 前置环境准备——安装Maven

若依基于Maven管理项目的构建，需要先安装好相应的版本。

1.2 开发工具

若依系统采用Eclipse作为开发工具。但不局限于Eclipse。此处仅介绍在Eclipse搭建开发环境所需的操作。

# 2. 开发环境搭建

2.1 开发工具的配置

2.1.1 配置Maven

进入Window->Preferences->Maven->Installations页面，设置已经安装好的Maven



2.1.2 配置Maven仓库路径

进入Window->Preferences->Maven->User Settings页面，配置仓库路径



2.1.4 关闭校验

进入Window->Preferences->Validation页面，勾选“Suspend all validators”，关闭校验



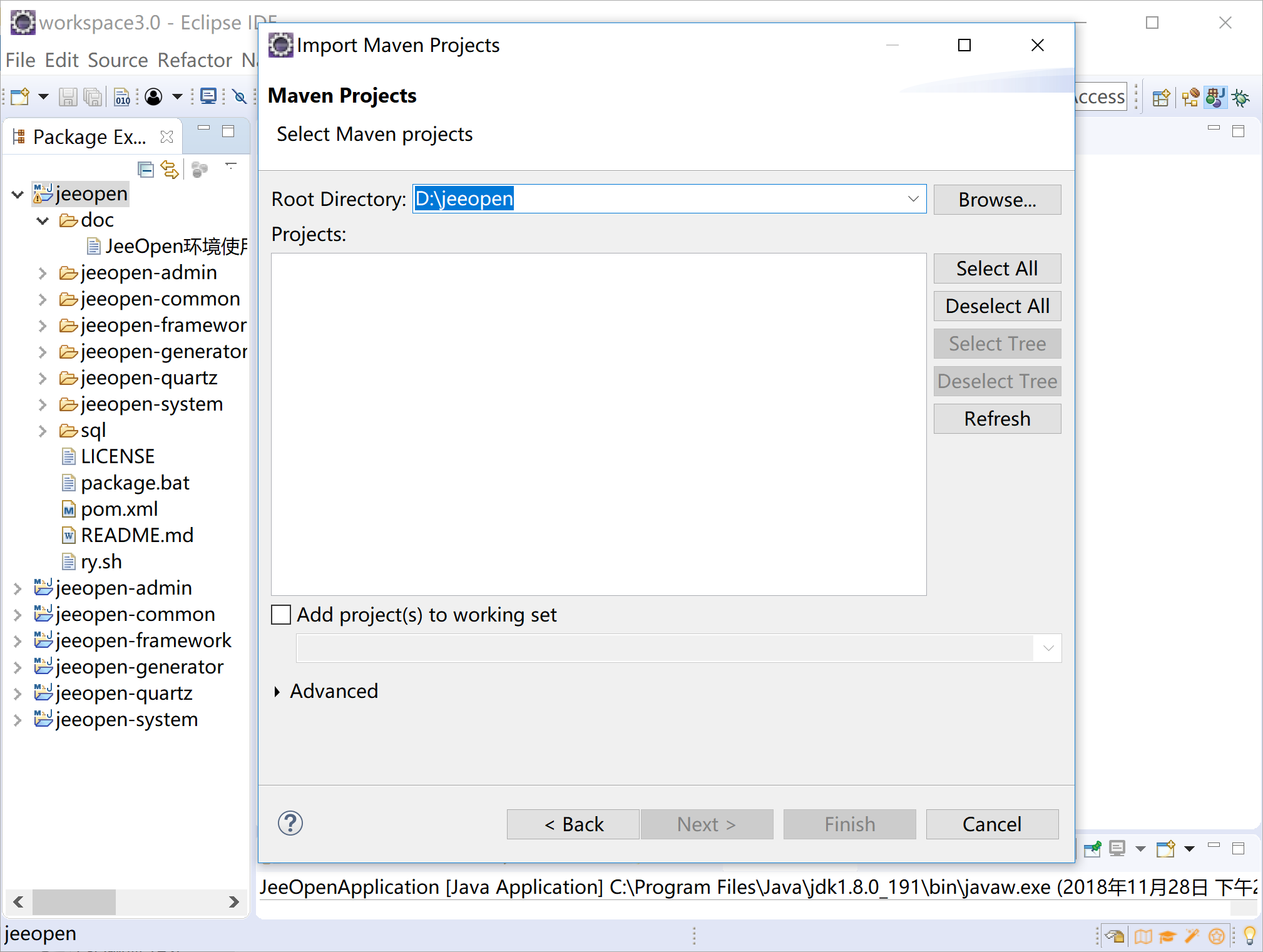
2.2 导入工程

通过Eclipse导入工程，步骤如下：

(1)点击左侧项目区域—>Import...

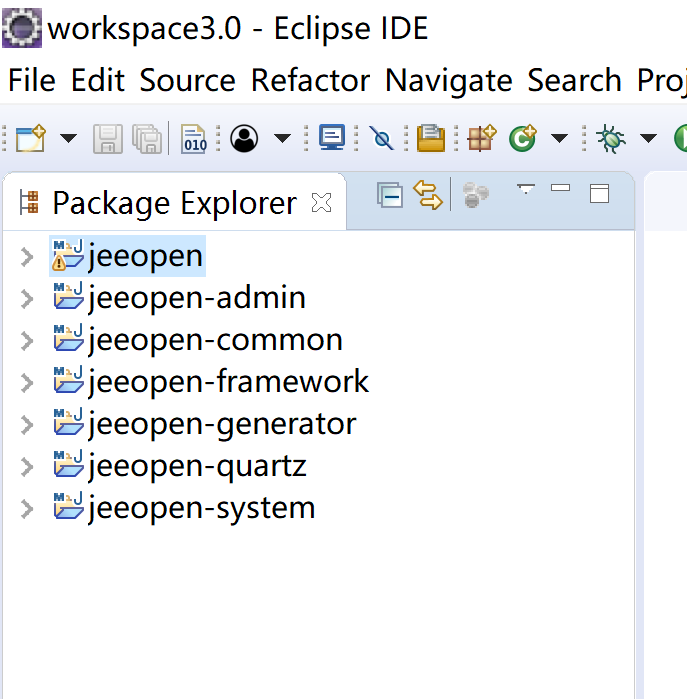


(2)选择jeeopen



(3)点击Finish

(4)jeeopen的代码就被导出到Eclipse中了，此时可以在工程视图中看到。



# 3. 运行jeeopen系统

3.1 必要的配置

3.1.1 修改数据库连接

编辑src/main/ resources目录下的application-druid.yml 文件，修改数据库地址账号信息。

执行sql/ jeeopen\_20181124.sql，quartz.sql 两个文件。

3.1.2 开发环境配置

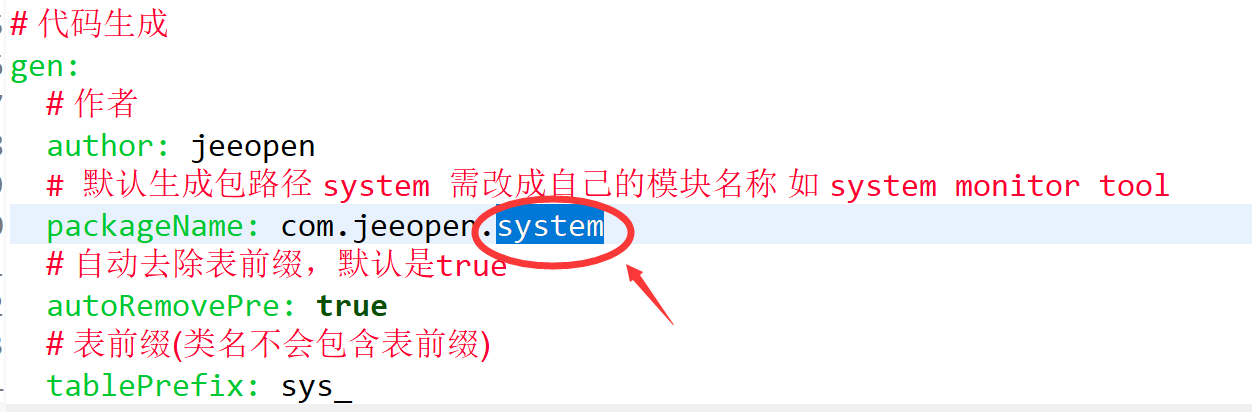
编辑src/main/ resources目录下的application.yml 文件，

默认端口为80



3.1.3 代码生成配置

编辑src/main/ resources目录下的application.yml 文件，



默认为system，根据实际情况修改即可。生成的表要有注释

3.2 启动及验证

启动JeeOpenApplication.java 出现如下图表示启动成功



打开浏览器，输入：http://localhost:80/

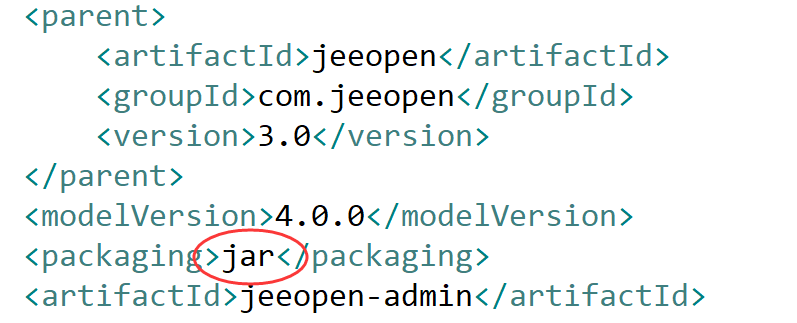
若能正确展示登录页面，并能成功登录，登录后菜单及页面展示正常，则表明环境搭建成功。

默认密码为 t1/123456

# 4. 部署系统

4.1 war部署方式

4.1.1 修改pom.xml文件。将jar修改为war,jeeopen-admin



4.1.2 在spring-boot-starter依赖中移除tomcat模块



<exclusions>

<exclusion>

<artifactId>spring-boot-starter-tomcat</artifactId>

<groupId>org.springframework.boot</groupId>

</exclusion>

</exclusions>

4.1.3 部署到tomcat的webapps目录下面

默认为jeeopen.war

4.1.4 启动及验证

运行startup.bat 出现如下图即部署成功



4.2 Jar方式部署

执行命令：java –jar jeeopen.jar

脚本执行：ry.sh start 启动stop 停止