## 工作日志

学号：1830090115

姓名：李响

* 学习内容：

|  |  |
| --- | --- |
| 序号 | 内容 |
| 1 | 了解商城系统功能模块和系统架构。 |
| 2 | 掌握maven工程的环境搭建。 |
| 3 | Svn的使用和管理。 |

* 心得体会：

maven pom元素

依赖范围：<scope>test</scope>

compile:默认的范围，编译测试运行都有效

provided:在编译和测试时有效，如j2ee的jsp依赖包，需要配置这个避免，部署时运行项目与tomcat冲突

runtime:在测试和运行时有效,jdbc驱动实现

test: 在测试时有效，如junit包

system:与本机系统相关联，可移植性差

import:导入的范围，它只使用在dependencyManagement中，表示从其他的pom中导入dependency的配置

依赖传递，如A依赖B，B依赖C，配置好依赖关系，A将依赖C

排除依赖-如A依赖B，B依赖C，配置好依赖关系，A将依赖C，如果不想A依赖C，可以下配置，如<exclusions>

<exclusion>

<groupId>com.hongxing</groupId>

<artifactId>hongxing-bige</artifactId>

</exclusion>

</exclusions>

依赖冲突：有两条原则：

1、短路优先，如A->B->C->X(jar),A->D->X(jar)(优先，因为路径最短)

2、先声明先优先，如果路径长度相同，则谁先声明，先解析谁，根据dependency声明顺序

聚合和继承：聚合：如果在maven中多个项目，进行install安装到本地，而且一次install，为所有的项目install,如 <modules>

<module>../hongxing-bige</module>

<module>../hongxing-nange</module>

<module>../hongxing-shanji</module>

</modules>

继承：如果有多个jar在多个项目复用，可以用继承实现一次配置， 如<parent>

<groupId>com.hongxing</groupId>

<artifactId>hongxing-parent</artifactId>

<version>0.0.1-SNAPSHOT</version>

</parent>

jetty方式启动web服务：找到jetty-maven-plugin坐标(注：有时报错是jetty版本不兼容当前jdk版本)，配置<plugins>

<plugin>

<groupId>org.eclipse.jetty</groupId>

<artifactId>jetty-maven-plugin</artifactId>

<version>9.0.7.v20131107</version>

<executions>

<execution>

<!-- 在打包成功后使用jetty:run启动jetty服务 -->

<phase>package</phase>

<goals>

<goal>run</goal>

</goals>

</execution>

</executions>

</plugin>

</plugins>

tomcat启动web服务：http://tomcat.apache.org/网站找到Maven Plugin,找到2.2版本，找到例子，复制坐标，如：<groupId>org.apache.tomcat.maven</groupId>

<artifactId>tomcat7-maven-plugin</artifactId>

<version>2.2</version>

配置默认jdk版本，可在settings.xml的<profiles></profiles>配置,如 <id>jdk-1.7</id>

<activation>

<activeByDefault>true</activeByDefault>

<jdk>1.7</jdk>

</activation>

<properties>

<maven.compiler.source>1.7</maven.compiler.source>

<maven.compiler.target>1.7</maven.compiler.target>

<maven.compiler.compilerVersion>1.7</maven.compiler.compilerVersion>

</properties>

SVN：

macos自带了svn环境，可通过"svn help"查看。

我使用git作为版本控制工具，具体配置过程如下：

lixiang@192 ~ % git --version

git version 2.24.3 (Apple Git-128)

lixiang@192 ~ % git config --global user.name "lixiang"

lixiang@192 ~ % git config --global user.email "lixiang007top@163.com"

lixiang@192 ~ % cd ~

lixiang@192 ~ % ssh-keygen

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

图形用户界面, 文本, 应用程序, 表格, 电子邮件

描述已自动生成

图形用户界面, 应用程序, Word

描述已自动生成

* 遇到问题：

|  |  |
| --- | --- |
| 序号 | 问题 |
| 1 | 项目加载速度慢。 |
| 2 | fatal: not in a git directory |
| 3 | Push failed |

* 解决方案：

图形用户界面, 文本, 应用程序

描述已自动生成

|  |  |
| --- | --- |
| 序号 | 方案 |
| 1 | 镜像仓库、更改仓库的位置。 |
| 2 | 路径更换到git目录。 |
| 3 | 不验证 SSL : 执行 git config http.sslVerify "false" 命令 ;  GitHub 上重新生成 SSH Key ;  取消代理 : git config --global --unset http.proxy , git config --global --unset https.proxy |

图片包含 表格

描述已自动生成

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成