1-maven-项目管理工具

1、Maven介绍

Maven是一个采用纯【Java编写】的开源【项目管理工具】,Maven采用了一种被称之为Project Object Model (POM)概念来管理项目,所有的【项目配置信息】都被定义在一个叫做【POM.xml】的文件中,通过该文件Maven可以管理项目的整个生命周期,包括青徐、编译,测试,报告、打包、部署等等。开发人员的主要任务应该是【关注商业逻辑】并去实现它,而不是把时间浪费在学习如何在不同的环境中去依赖。中心项目部署等。Maven正是为了将开发人员从这些任务中解说出来而诞生的。

2、Maven能够做什么

- a) Jar管理--》自动下载指定的jar,和依赖的jar
- b)项目管理-》项目编译,测试,打包...

3、Maven下载和配置

a) 单独下载使用(通过命令操作)

1)下载

http://maven.apache.org/download.html

2)配置环境

- x) MAVEN_HOME: 配置到maven的根目录
- x) path环境: %MAVEN_HOME%\bin
- x) MAVEN_HOME/conf/settings.xml 这是全局配置信息(一般配置远程仓库地址!)
- 3)测试环境:命令行输入 mvn -version 看是否有版本提示

4)其他常见命令

mvn eclipse:eclipse 生成Eclipse项目结构 mvn eclipse:clean 清除Eclipse项目结构

mvn clean 清理 (删除target目录下编译内容)

mvn compile 编译项目

mvn test 测试项目

mvn install 安装项目

mvn package 打包发布

mvn package -Dmaven.test.skip=ture 打包时跳过测试

b) 无需下载 (在eclipse高版本自带maven程序)

无需配置环境,通过界面操作。

4、ecipse使用maven

1) maven项目的结构约定



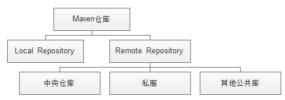
2) pom文件基本功能介绍

- -》参考pom文件的模版
- 3) Maven项目坐标
- ps-》通过坐标可以准确的找到唯一的一个maven仓库的项目
 - a) groupld:定义当前Maven项目隶属项目(实际对应JAVA的包的结构,一般为公司域名)
 - b) artifactId: 定义实际项目中的一个模块(项目的唯一的标识符,实际对应项目的名称就是项目根目录的名称)
 - c) version: 定义当前项目的当前版本

4) Maven仓库

用来统一存储所有Maven项目的位置就是仓库

Maven配置jar包的路径为: groupId/artifactId/version/artifactId-version



a) 本地仓库(~/.m2/repository/): 每个用户只有一个本地仓库

b) 远程仓库

1)中央仓库(Maven默认的远程仓库):访问人多【速度慢】。

http://repo.maven.apache.org/maven2/

- 2)私服:是一种特殊的远程仓库,它是架设在【局域网内的仓库】,主要是为了团队协作开发
- ps-》我们自己架设在内网,通过nexus可以架设。
- 3)镜像(其他公共库):用来替代中央仓库,【速度一般比中央仓库快】
- ps-》阿里云【目前很快】,开源中国等公司提供的-在外网的maven库

http://maven.aliyun.com/nexus/content/groups/public/

http://maven.oschina.net/content/groups/public/

http://repository.jboss.com/maven2/

5) setting文件介绍(用于把pom里的通用配置提取)

ps-》参考setting.xml-基本模版

主要用于,配置本地仓库地址,远程仓库地址。

6) eclipse建立maven项目

a) 方式1:新建

Maven Project

开发web项目需要做很多配置,不方便

b) 方式2:新建

Dynamic Web Project

选中项目configure -》convert maven project

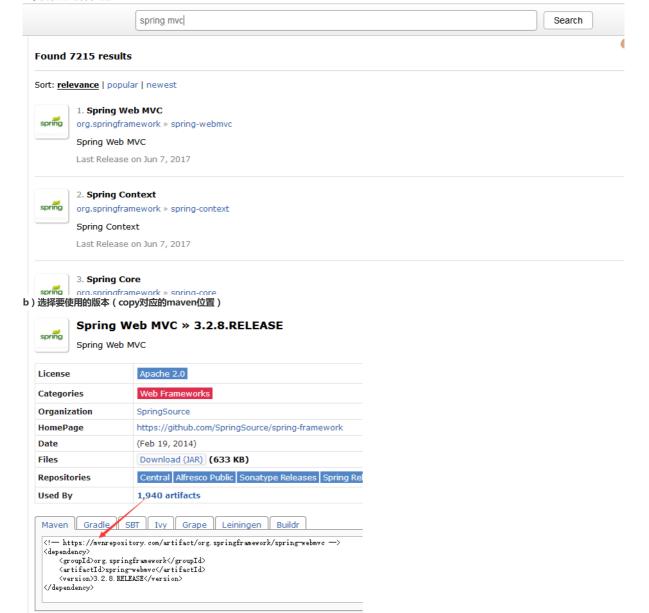
修改pom文件,进行相关项目的配置

- 1)项目信息
- 2)项目依赖jar
- 3) 项目的jre环境 ...

7) jar包查找 (java常用的第三方jar在maven仓库基本都可以找到)

maven仓库-项目在线查看: http://www.mvnrepository.com/

a)搜索并选择合适的包



c) 当前下载的jar所依赖的其他jar

ps-》其中optional表示不是必须(扩展功能需要使用),不会自动下载该jar,其他不是optional都会下载。

Compile Dependencies (31)



Template Engine Apache 2.0	org.freemarker w freemarker (optional)	2.3
Dep Injection Apache 2.0	org.springframework » spring-beans	3.2.8.F
Dep Injection Apache 2.0	org.springframework » spring-context	3.2.8.F
Apache 2.0	org.springframework » spring-context-support (optional)	3.2.8.F

练习:

把ssh框架写的crm项目的jar,用maven的pom配置一遍,远程仓库,阿里云,jdk1.7。