Maven入门（含实例教程）

Maven这个项目管理和构建自动化工具，越来越多的开发人员使用它来管理项目中的jar包。接下来小宝鸽，将从下面几个方向介绍maven：

**（1）Maven简单介绍**   
**（2）Maven安装与配置**   
**（3）Maven携手eclipse构建项目**   
**（4）Maven的常用命令**   
**（5）Maven实例**

猿友门抓紧了，马上起飞了：

（1）Maven的简单介绍

Maven这个个项目管理和构建自动化工具，越来越多的开发人员使用它来管理项目中的jar包。但是对于我们程序员来说，我们最关心的是它的项目构建功能。所以这里我们介绍的就是怎样用 maven 来满足我们项目的日常需要。Maven 使用惯例优于配置的原则 。它要求在没有定制之前，所有的项目都有如下的结构：



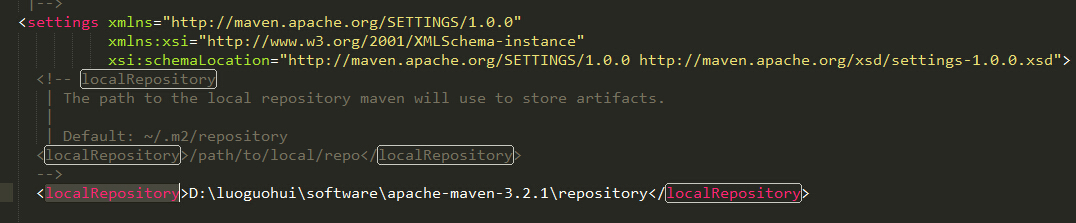
一个 maven 项目在默认情况下会产生 JAR 文件，另外 ，编译后 的 classes 会放在 basedir/target/classes 下面， JAR 文件会放在 ${basedir}/target 下面。这时有人会说了 ， Ant 就没有那么多要求 ，它允许你可以自由的定义项目的结构。在这里不想引起口水战哈， 我个人觉得 maven 的这些默认定义很方便使用。好了 ，接下来我们来安装 maven 。

（2）Maven安装与配置

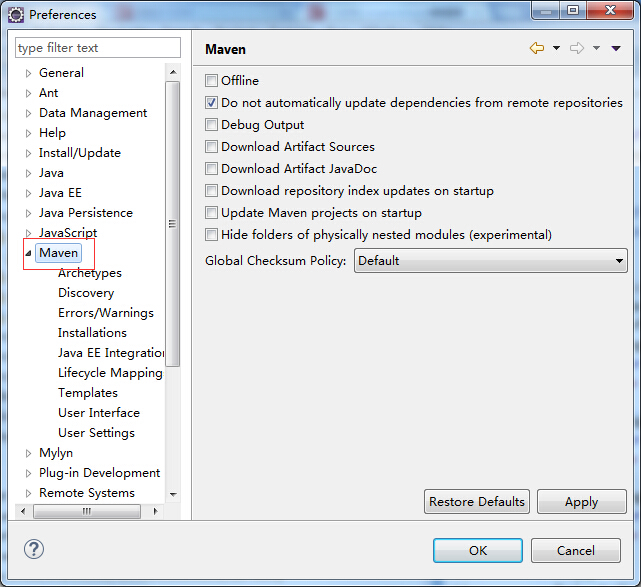
2.1、下载maven安装包：apache-maven-3.2.1 <http://download.csdn.net/detail/u013142781/9355367>，解压即可使用。

2.2、maven环境变量配置，配置方式跟jdk有些类似。新建环境变量MAVEN\_HOME（值为maven的根目录）、然后在PATH环境变量里加入%MAVEN\_HOME%\bin;即可。

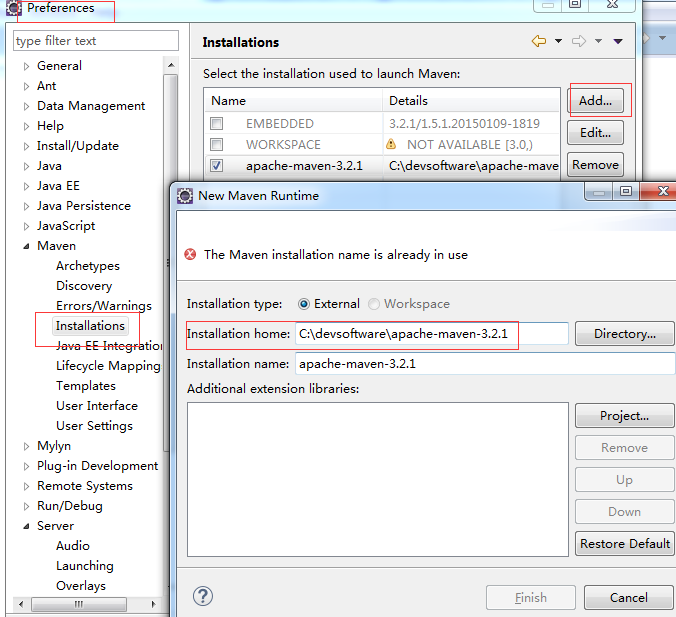
2.3、设置setting.xml文件，配置本地仓库。apache-maven-3.2.1\conf下有个setting.xml文件，打开，localRepository的值设置成你本地仓库的路径。（博主在apache-maven-3.2.1根目录下创建了个repository文件夹作为本地仓库）。如下图



2.4、maven配置到eclipse上，下载插件eclipse-maven3-plugin <http://download.csdn.net/detail/u013142781/9355661>, 解压后把对应的features和plugins这两个文件夹的内容复制到eclipse相同文件夹里面即可，然后重启eclipse，当你打开Window→ Preferences…的时候，你应该能够在一个选项列表中看到一个Maven选项：



然后，Windows–>Prefrences–>Installations–>Add。installation name选maven的根目录，然后一直按确定就OK啦。



至此 maven安装配置完成！如果需要了解jdk+eclipse+svn+maven+mysql+tomcat7.0的环境搭建，可阅读小宝鸽的博客<http://blog.csdn.net/u013142781/article/details/50300233>，非常详细而且还提供安装包哦。

（3）Maven携手eclipse构建项目

具体的创建方式可以查看我好友的一篇博客：Spring+Mybatis+SpringMVC+Maven+MySql搭建实例，<http://blog.csdn.net/evankaka/article/details/48785513>，非常非常详细。当然小宝鸽以后也会写一篇“Spring+Mybatis+SpringMVC+Maven+MySql搭建实例”，希望多多关注支持啊。

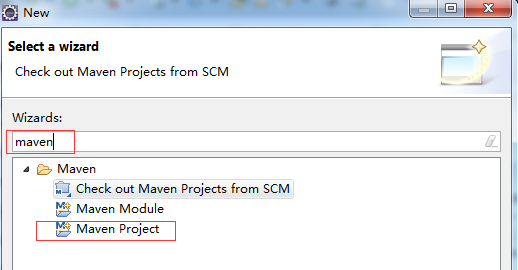
（4）Maven的常用命令

Maven常用命令：

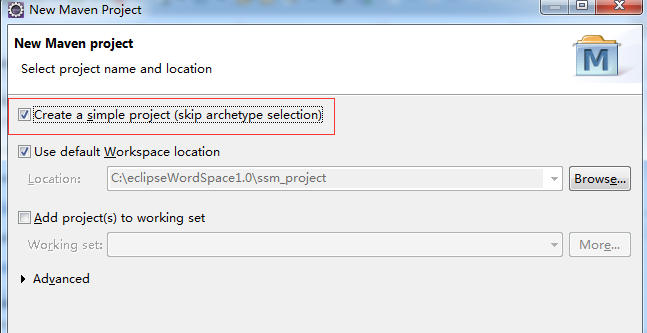
1. 创建Maven的普通java项目：   
   mvn archetype:create -DgroupId=packageName -DartifactId=projectName
2. 创建Maven的Web项目：   
   mvn archetype:create -DgroupId=packageName -DartifactId=webappName-DarchetypeArtifactId=maven-archetype-webapp
3. 编译源代码： mvn compile
4. 编译测试代码：mvn test-compile
5. 运行测试：mvn test
6. 产生site：mvn site
7. 打包：mvn package
8. 在本地Repository中安装jar：mvn install
9. 清除产生的项目：mvn clean
10. 生成eclipse项目：mvn eclipse:eclipse
11. 生成idea项目：mvn idea:idea
12. 组合使用goal命令，如只打包不测试：mvn -Dtest package
13. 编译测试的内容：mvn test-compile
14. 只打jar包: mvn jar:jar
15. 只测试而不编译，也不测试编译：mvn test -skipping compile -skipping test-compile   
    ( -skipping 的灵活运用，当然也可以用于其他组合命令)
16. 清除eclipse的一些系统设置:mvn eclipse:clean

（5）Maven实例

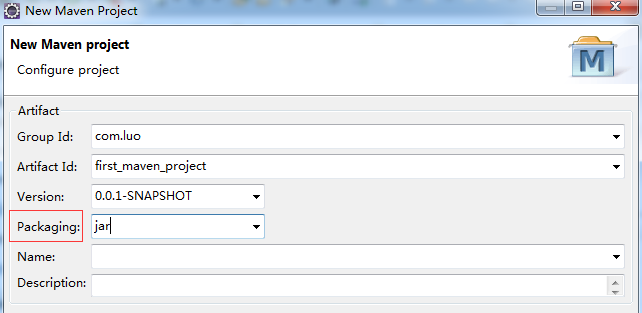
1、Maven工程创建



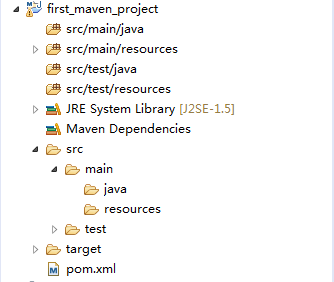
2、选择快速框架



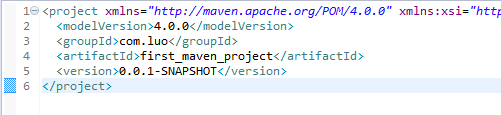
3、输出项目名，包（Packaging，如果只是普通的项目，选jar就好了，如果是web项目就选war，这里我们选择jar）



4、创建好的目录如下：



5、刚开始的pom.xml是这样的



6、现在我们添加mysql驱动包的依赖,编辑pom.xml如下：

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.luo</groupId>

<artifactId>first\_maven\_project</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<!-- mysql驱动包 -->

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

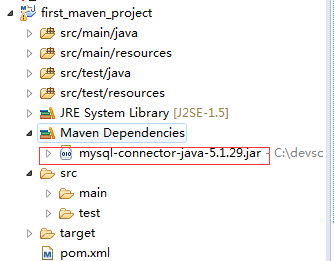
<version>5.1.29</version>

</dependency>

</dependencies>

</project>

7、如上编辑pom.xml，然后保存，那么就会自动从中央仓库下载包，会发现这里多了一个包：



然后就可以正常使用这个包啦

8、其他想要依赖的包，对应的dependency，可以在这个网站找：   
<http://mvnrepository.com/>

