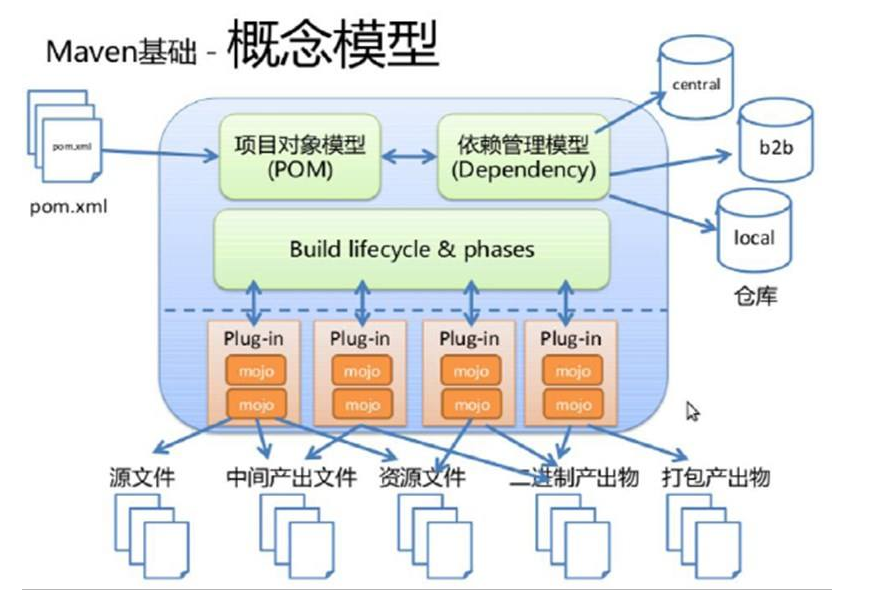
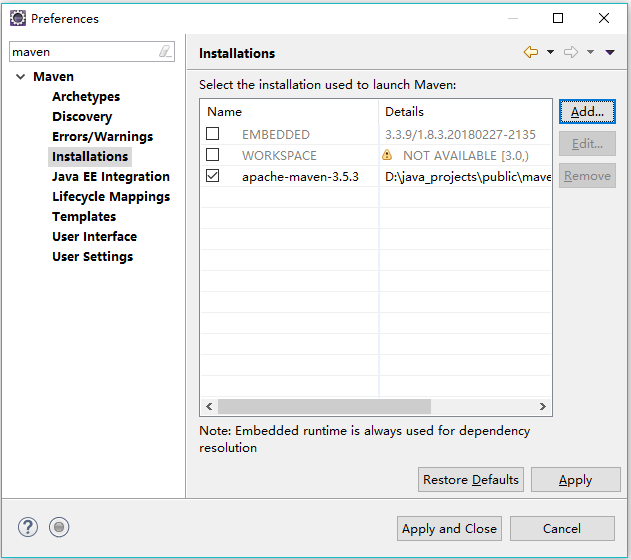
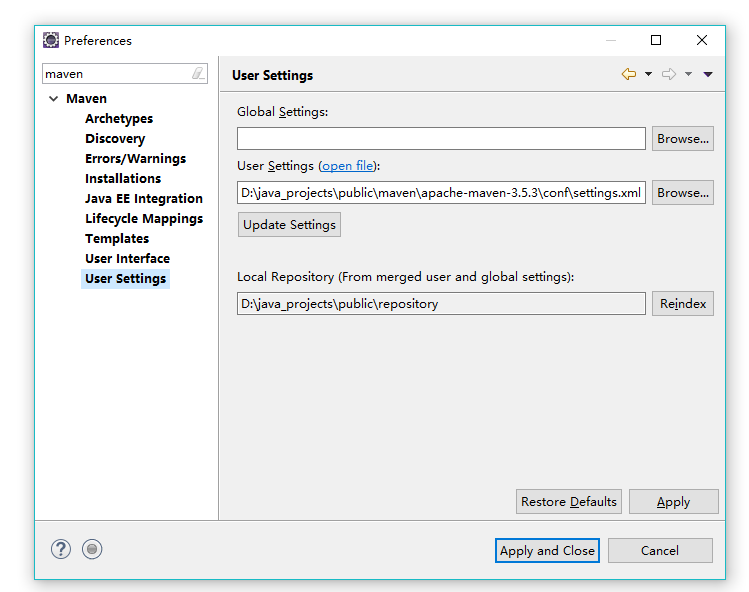
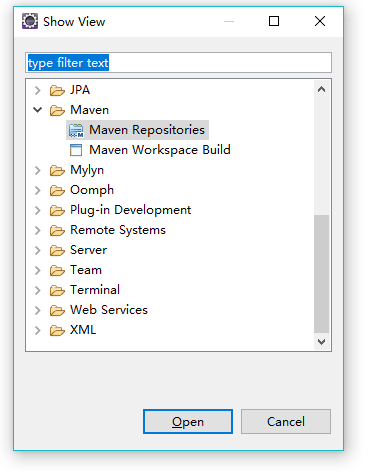
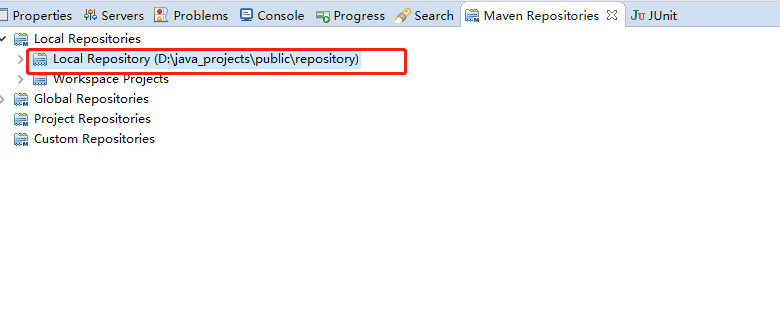
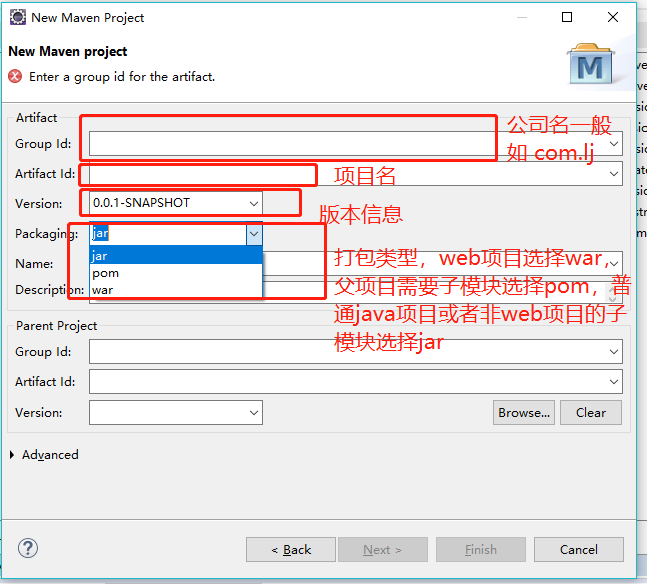
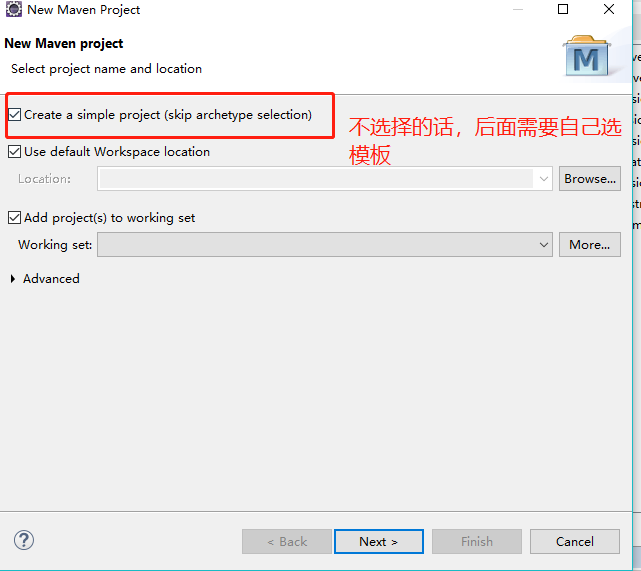
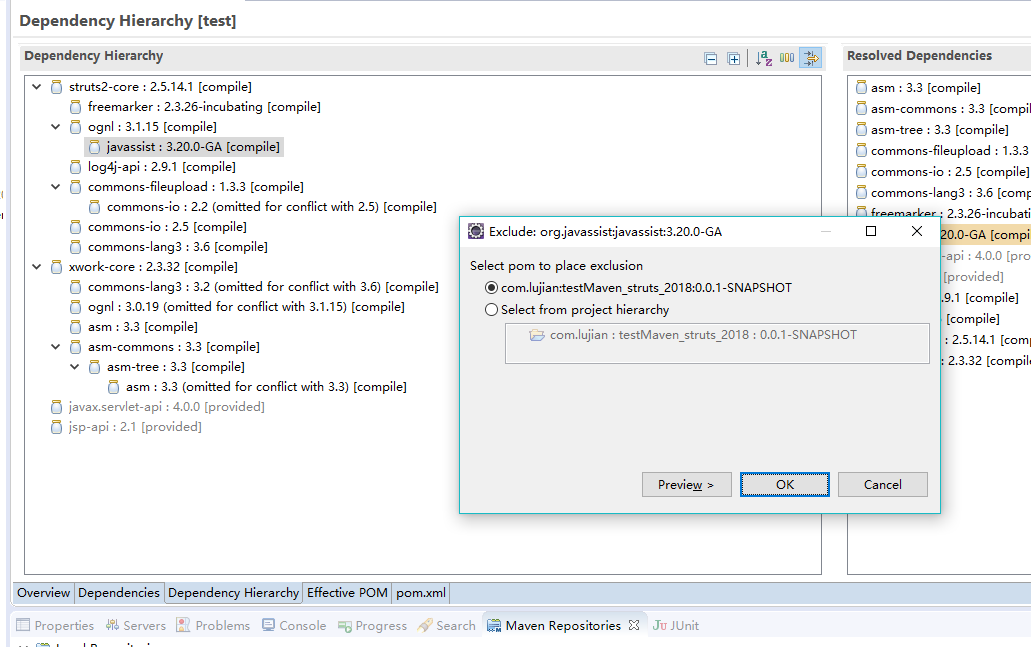
2018年2月20日星期二

maven学习总结

* Maven概念及作用
  + 基于项目对象模型（pom project - object - model）的项目管理工具
  + 主要作用
    - 依赖管理
      * 使用pom.xml配置对jar包等文件统一管理，不用再复制粘贴jar包，并且可以看源码，使项目开发管理更容易
    - 项目构建
      * 可以使用maven插件对项目进行编译测试打包等操作



* 程序安装
  + 直接去官网下载解压即可
  + 配置一下环境变量，和jdk类似
* 配置仓库
  + Maven安装目录下的config文件夹中的settings.xml中配置本地仓库
* 仓库类型
  + 本地仓库（自己电脑上）
  + 私服（公司或局域网服务器上）
  + 中央仓库（Apache管理）
  + Maven项目会先从本地找，找不到再去私服，最后再去中央仓库找，并且会下载到本地仓库
* maven项目目录结构
  + Src
    - Main
      * Java
      * Resource
      * WebApp
    - Test
      * Java
      * Reource
  + Target
* 常用命令
  + mvn tomcat:run（运行tomcat，默认是tomcat6）
  + mvn complie（编译）
  + mvn test（测试）
  + mvn package（打包，默认在target目录下）
  + Mvn install（打包到本地仓库）
  + Mvn clean（清理掉编译好的class文件，即target目录）
  + ...
* 生命周期（三套生命周期互不干扰，但是在同套周期中会按顺序都执行，比如default中运行install命令，他会自动执行前面的编译 测试等命令）
  + Clean
  + Default（包括上面常用的命令）
  + Site（不常用）
* eclipse中配置流程
  + Window-preferences-Maven-配置maven插件
  + 配置本地仓库（需要在settings.xml文件中修改路径，不修改则是默认路径在c盘）
  + Window-showView-打开Maven控制台
  + 构建索引，右键rebuildIndex
* Eclipse中创建maven项目，查找依赖
* 依赖范围
  + Complie 编译 测试 运行
  + Runtime 运行（像jdbc驱动包)
  + Provide 编译 (像servlet-api这种tomcat自带的，项目中需要设置成provide，否则容易jar冲突)
  + test:，测试依赖范围，测试时需要。编译和运行不需要。如Junit
* 传递依赖冲突解决
  + 项目A → jarB（由项目B打包而成）→jarC（项目B里的jar）,那么C就是A的传递依赖
  + Maven自己调节
    - 多个不同版本的传递依赖遵循就近原则，谁排前面用谁
    - 有直接依赖优先直接依赖
  + 自己调节
    - Eclipse中排除依赖,右键exclude maven Artifact
    - 版本指定（常用）

其他：

* 学习了搭建maven 整合ssh项目
* 学习了创建父工程，并且拆分子模块
* 了解了下私服的使用