maven这个词翻译为（专家，内行）时跨平台的项目管理工具

maven主要服务于基于java平台的 项目构建，依赖管理，项目信息管理。

（最主要的就是解决相关jar包的下载）

（在联网的情况下自动下载，下载到本地仓库，下载一次永久享用）

使用maven必须要连接网络

1.依赖管理指的就是使用maven来管理项目中使用到的jar包。

maven管理的方式就是“自动下载项目所需要的jar包，统一管理jar包之间的依赖关系”。

2.下载，配置maven，首先要配置好java，（因为maven是java项目，必须要依赖于java环境）

在 maven.apache.org 网站中下载maven压缩包

解压后 目录结构：bin:控制maven 命令 执行的目录

boot:

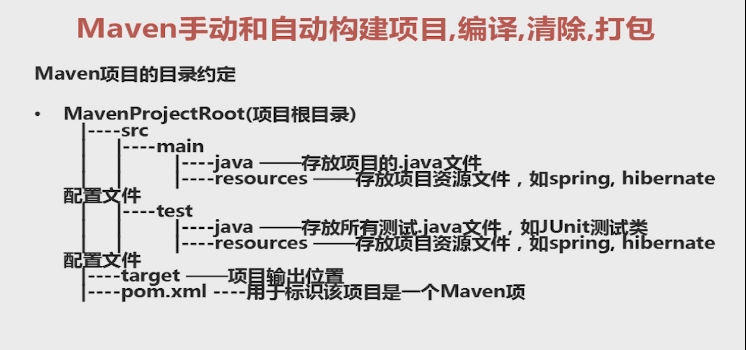
conf:里面放的是一些配置，未来的仓库也是在这里配置

lib: maven本身依赖的一些jar包，负责maven构建一些 命令的依赖

配置环境变量：path中添加 C:\Program Files\apache-maven-3.6.0\bin

在命令行中 输入mvn -v 查看是否配置成功

3.maven 手动和自动 构建项目的2种方式 （构建项目，编译，清除，打包，和发布）



Pom.xml 是maven的核心 配置文件。

手动编译：以前要到 java类的目录下 通过javac，java。

使用maven 只需要在工程的根目录下执行命令。（pom.xml文件所在的目录）

**！！！maven是自动集成的编译，命令的执行不需要切换到类的目录里面，是需要在pom.xml文件所在的目录下即可**

**（工具Eclipse 只是将maven的命令进行集成）**

一切的命令都是通过 mvn ….. 执行

mvn compile 编译 （最重要的就是 会在工程目录下产生一个target目录）

mvn install 编译，运行，和发布（安装maven本地仓库的安装）

mvn package 编译，和运行（不负责发布）

使用package命令 不会将target内容发布到本地仓库

mvn clean 清理（清理的原因是要重新编译，打包，部署，**清理不会把本地仓 库中的清理掉**）

mvn test 测试（编译，如果test包下面有相关的java也编译和运行，**但是也 一样不会发布到本地仓库中**）

自动生成mvn骨架（包括pom）

1.使用mvn archetype:generate 命令

mvn archetype:generate –DgroupId=com.mycompany.app –DartifactId=myapp –DarchetypeArtifactId=maven-archetype-quickstart –DinteractiveMode=false

2.使用mvn archetype:create 命令创建一个java项目的过程