

# Maven简介和安装

---

## 本节目标

1.了解Maven的特点 2.掌握Maven的基本使用

## 1.简介

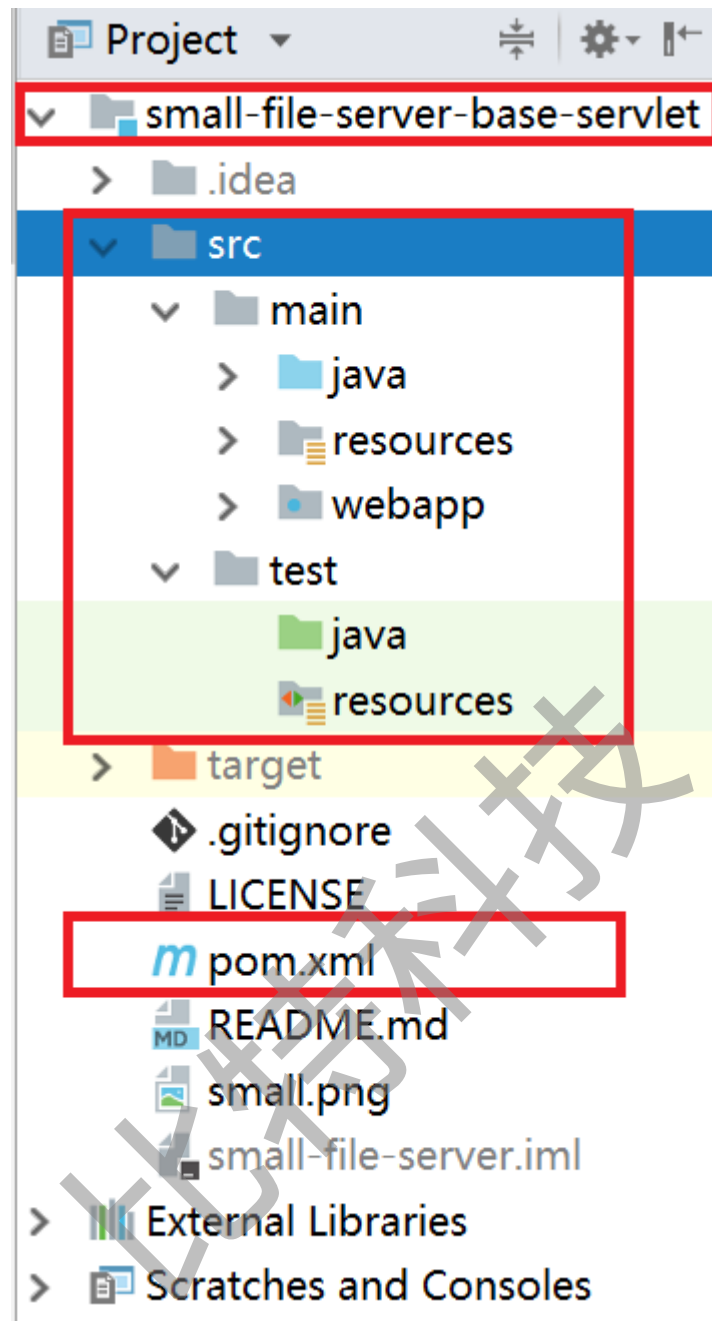
---



### 1.1 Maven是什么

[Apache Maven](http://maven.apache.org/)是一个项目管理和综合工具。Maven提供了开发人员构建一个完整的生命周期框架。开发团队可以自动完成项目的基础工具建设，Maven使用标准的目录结构和默认构建生命周期。

基于Maven管理的项目结构如下图所示（红框部分是项目目录）：



在多个开发环境时，Maven可以按标准设置，并且在非常短的时间里完成配置工作。由于大部分项目的设置都很简单，并且可重复使用，Maven让开发人员的工作更轻松，同时创建报表，检查，构建和测试自动化设置。

Maven提供了开发人员的方式来管理：

- Builds (构建)
- Documentation (文档)
- Reporting (报告)
- Dependencies (依赖)
- SCM (软件配置管理)
- Releases (发布)
- Distribution (分发)
- Mailing list (邮件列表)

概括地说，Maven简化和标准化项目建设过程。处理编译，分配，文档，团队协作和其它任务的无缝连接。Maven增加可重用性并负责建立相关的任务。

## 1.2 Maven历史

Maven最初设计，是以简化Jakarta Turbine项目的建设。经历了Maven-> Maven2 -> Maven3的发展，成为了Java领域项目构建的事实上的标准化管理工具。

## 1.3 为什么要用Maven

Maven之前我们经常使用Ant来进行Java项目的构建，然后Ant仅是一个构建工具，它并未对项目的中的工程依赖以及项目本身进行管理，并且Ant作为构建工具未能消除软件构建的重复性，因为不同的项目需要编写对应的Ant任务。

Maven作为后来者，继承了Ant的项目构建功能，并且提供了依赖关系，项目管理的功能，因此它是一个项目管理和综合工具，其核心的依赖管理，项目信息管理，中央仓库，约定大于配置的核心功能使得Maven成为当前Java项目构建和管理工具的标准选择。

**学习Maven的理由是非常多：**

- 主流IDE (Eclipse,IDEA,Netbean) 都内置了Maven
- Spring Framework已经不再提供jar的下载，直接通过Maven进行依赖下载
- 在github，开源社区几乎所有流行的Java项目都是通过Maven进行构建和管理的
- 主流软件公司也在使用Maven

## 2.安装

课程采用最新版**Maven 3.5.2**

### 2.1 准备环境

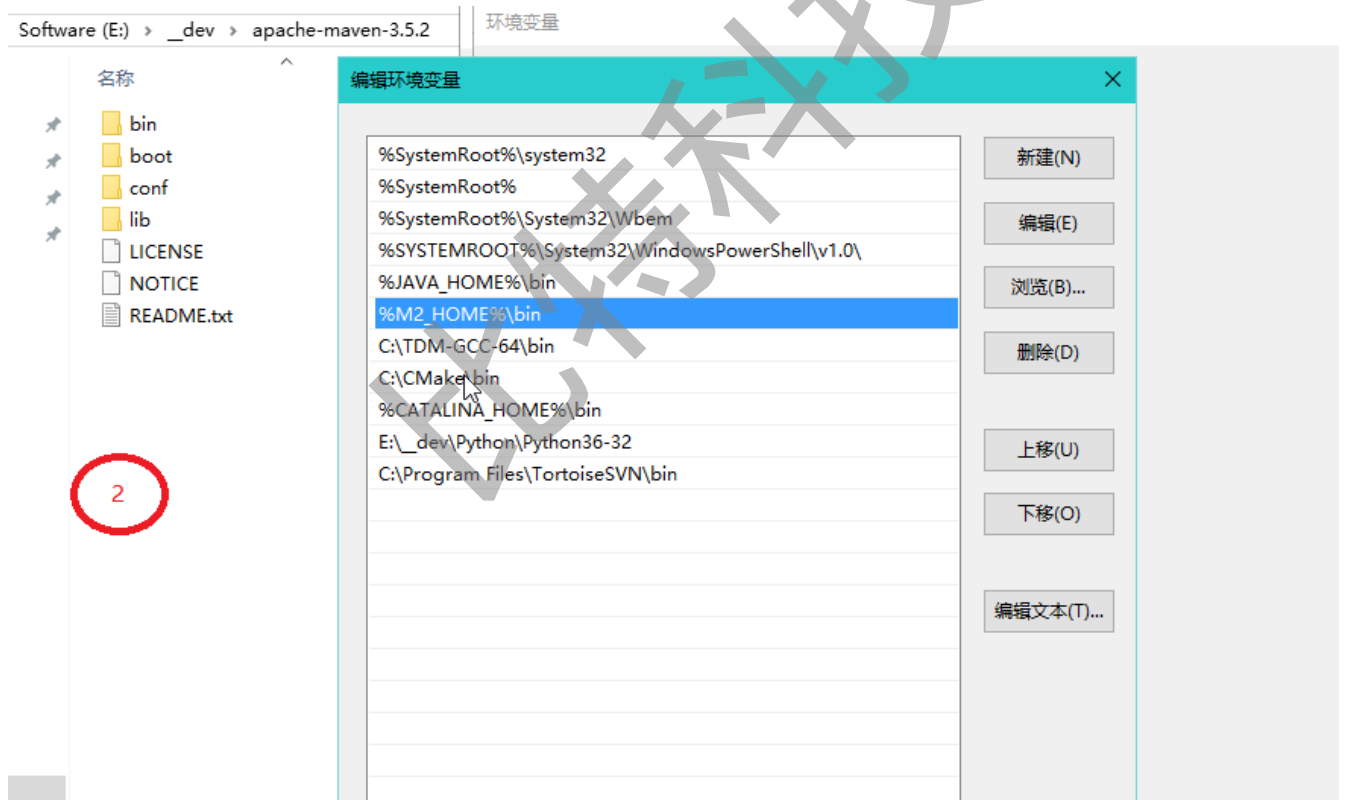
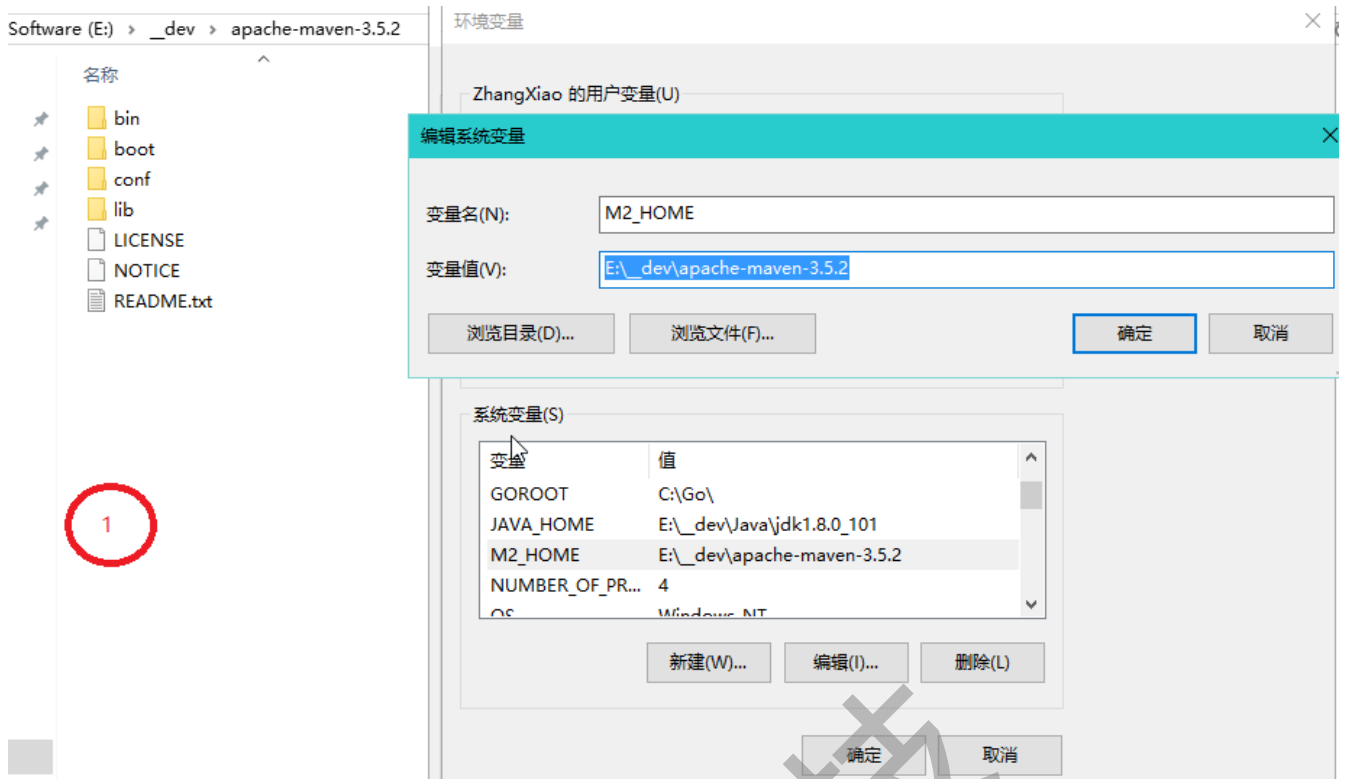
- JDK: 1.7+(Maven3.3+要求JDK是1.7以上) 配置JAVA\_HOME环境变量
- 磁盘要求：安装Maven需要10MB，本地仓库需要至少500MB

### 2.2 下载

- 下载地址: <http://maven.apache.org/download.cgi>

### 2.3 安装

- 安装包解压
- 配置环境变量M2\_HOME和path环境变量



- 在命令行窗口执行 `mvn -v` 输出Maven版本信息表明安装成功

```
C:\Windows\system32\cmd.exe

D:\>mvn -v
D:\
Apache Maven 3.5.2 (138ed61fd100ec658bfa2d307c43b76940a5d7d; 2017-10-18T15:58:13+08:00)
Maven home: E:\_dev\apache-maven-3.5.2\bin\
Java version: 1.8.0_101, vendor: Oracle Corporation
Java home: E:\_dev\Java\jdk1.8.0_101\jre
Default locale: zh_CN, platform encoding: GBK
OS name: "windows 10", version: "10.0", arch: "amd64", family: "windows"
```

## 2.4 配置（可选）

- 配置用户环境（例如：C:\Users\Deletereus\.m2，如果没有可以创建）下的Maven配置 settings.xml
- 本地仓库 localRepository
- 插件组 pluginGroups
- 代理服务 proxies
- 私服配置 servers
- 镜像配置 mirrors
- 规则配置 profile

## 3.基本使用

usage: mvn [options] [<goal(s)>] [<phase(s)>]

Options:

- |                              |  |
|------------------------------|--|
| -D,--define <arg>            | 定义系统属性   |
| -f,--file <arg>              | 强制指定pom.xml文件，默认为当前目录下的pom.xml   |
| -h,--help                    | 显示帮助信息   |
| -P,--activate-profiles <arg> | 逗号分割活动的profiles列表  |
| -q,--quiet                   | 静默退出，只输出错误信息   |
| -s,--settings <arg>          | 修改用户的setting.xml文件，<br>读取顺序是 参数指定->用户目录/.m2/settings.xml-><br>>M2_HOME/conf/settings.xml |
| -v,--version                 | 显示版本信息   |
| -X,--debug                   | DEBUG模式输出处理日志  |

## 课后作业

- 完成个人机器的Maven安装