



# Java 核心技术(进阶)

## 第一章 Maven

### 第二节 Maven基础概念和实战

华东师范大学 陈良育

# 概要



- Maven安装
- Maven基础概念
- Maven实战(面向开发配置)



# 检查和安装Maven(1)

- 最新版本的Eclipse 自动集成Maven
- 检查是否正确安装Maven
  - File菜单->New->Other, 在弹出窗口中查找是否有Maven项目
  - Window菜单->Preferences, 在弹出窗口左侧是否有Maven子菜单
  - Help菜单->Install New Software, 在Work with窗口中输入Maven, 下面显示出All items are installed

## Available Software

Check the items that you wish to install.

Work with:

Name	
<input type="checkbox"/>	Maven Integration for Eclipse
<input type="checkbox"/>	All items are installed



## 检查和安装Maven(2)

- 在旧版本的Eclipse中安装Maven插件
  - Help菜单->Install New Software, 在Work with窗口中输入maven, 打勾下面的软件, 并一路Next。

### Available Software

Check the items that you wish to install.

Work with:	maven	http://download.eclipse.org/technology/m2e/releases
type filter text		
Name		
>	<input checked="" type="checkbox"/>	Maven Integration for Eclipse

- 或者前往 <https://www.eclipse.org/m2e/> 进行下载

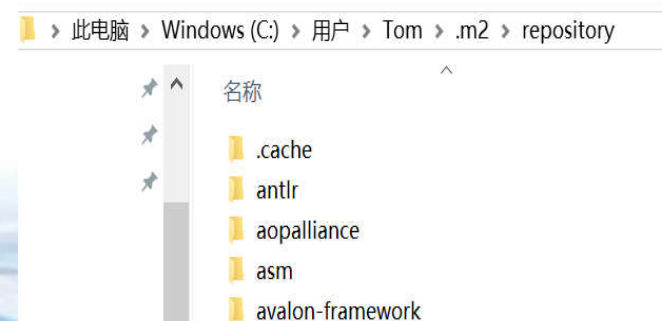
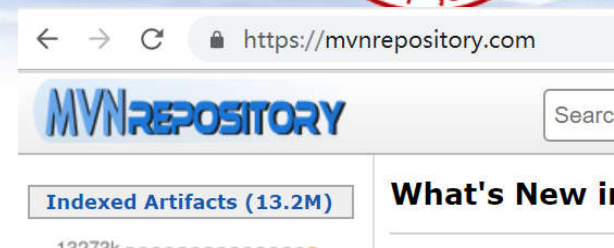
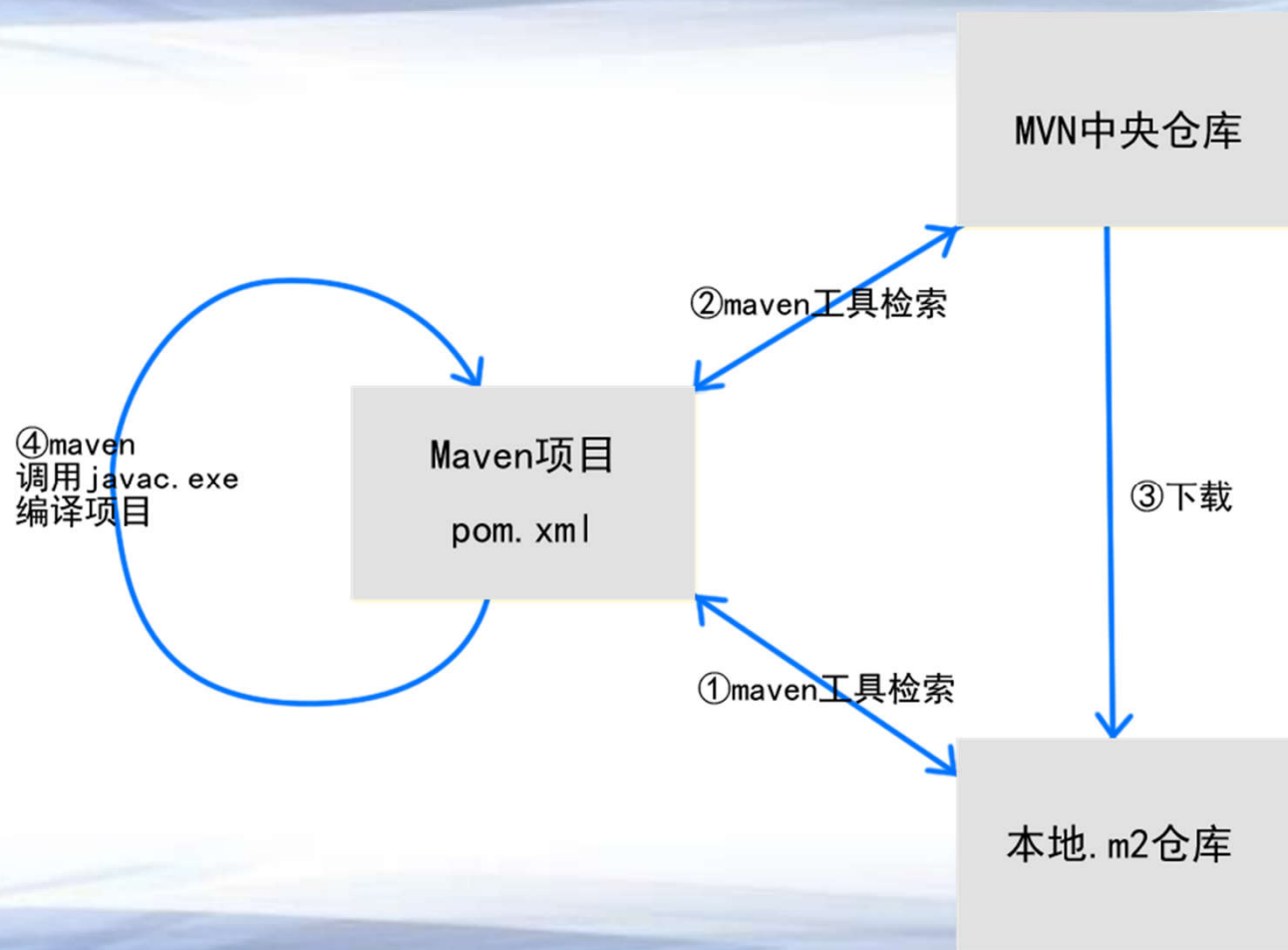


# Maven基础概念

- Maven开发流程
  - 新建Maven项目
  - 在中央仓库查找第三方jar的依赖文本
  - 拷贝依赖文本至项目的pom.xml
  - 执行maven build, 编译/构建整个项目
- Maven是一个软件, <http://maven.apache.org/> 下载
- Maven也有一个中心仓库, <https://mvnrepository.com/>。
  - 包含很多第三方软件。可以有很多第三方的Maven仓库。
- Maven是一个构建工具, 自动下载中心仓库的jar文件, 存在本地进行管理, **编译、测试、运行**、和打包发布Java项目。



# Maven编译工作流程





# POM( Project Object Model )

- XML格式
- 包含了项目信息、**依赖信息**、构建信息
- 构件信息(artifact)
  - groupId: 组织
  - artifactId: 产品名称
  - version: 版本

```
<dependency>  
  <groupId>org.apache.commons</groupId>  
  <artifactId>commons-math3</artifactId>  
  <version>3.6.1</version>  
</dependency>
```



# Maven repository(仓库)

- Maven仓库存放和管理各种构件

- 本地仓库(本地用户的.m2文件夹)

- 远程仓库

- 中央仓库
    - 阿里云仓库
    - 谷歌仓库
    - ...

此电脑 > Windows (C:) > 用户 > Tom > .m2 > repository

名称
.cache
antlr
aopalliance
asm
avalon-framework





# Maven项目的目录结构

- 基本目录结构

- src

- main

- java/ 存放java文件

- resources/ 存放程序资源文件

- test/

- java/ 存放测试程序

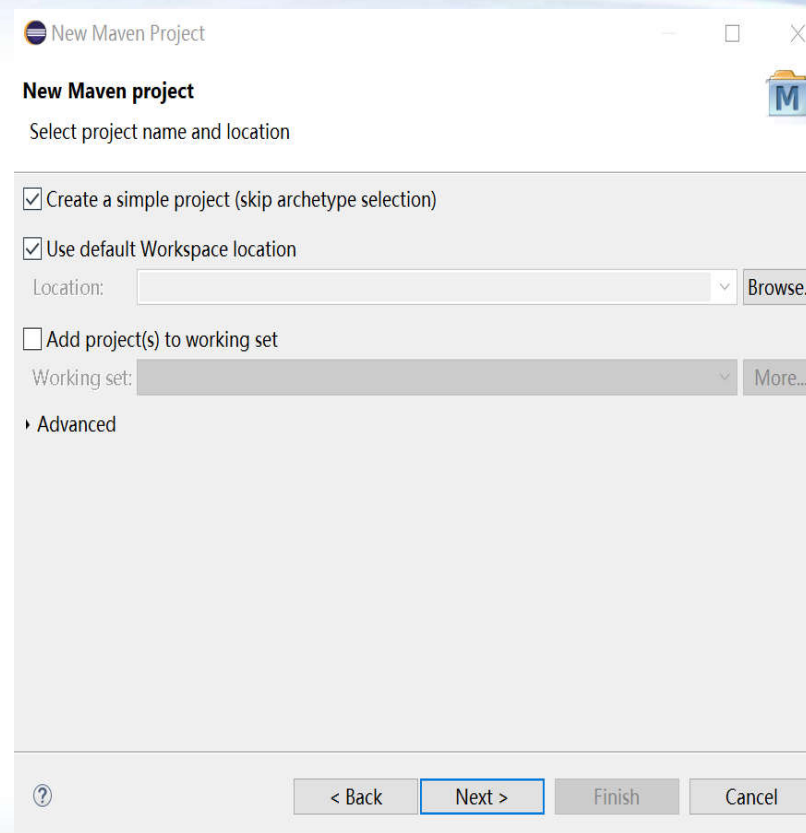
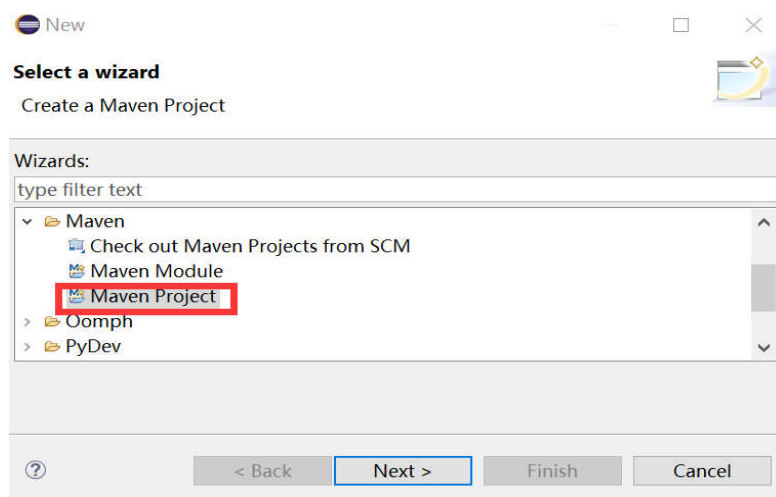
- resources/ 存放测试程序资源文件

- pom.xml

# 创建Maven项目(1)



- 直接创建Maven Project





# 创建Maven项目(2)

## • 设置信息

- groupId: 组织名
- artifactId: 作品(项目)名
- Name: 别名(optional)
- Description: 描述(optional)

A screenshot of the 'New Maven Project' dialog box in an IDE. The dialog is titled 'New Maven Project' and 'Configure project'. It contains several fields for project configuration. The 'Artifact' section is highlighted with a red box, showing 'group id: com.test' and 'Artifact id: MOOC13-02Maven'. Other fields include 'Version: 0.0.1-SNAPSHOT', 'Packaging: jar', 'Name: MOOC13-02Maven', and 'Description: MOOC13-02Maven'. There is also a 'Parent Project' section with empty fields for 'Group Id', 'Artifact Id', and 'Version'. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel'. A 'Browse...' button is next to the 'Version' field in the 'Parent Project' section.

New Maven Project

New Maven project  
Configure project

Artifact

group id: com.test

Artifact id: MOOC13-02Maven

Version: 0.0.1-SNAPSHOT

Packaging: jar

Name: MOOC13-02Maven

Description: MOOC13-02Maven

Parent Project

Group Id:

Artifact Id:

Version:

Browse... Clear

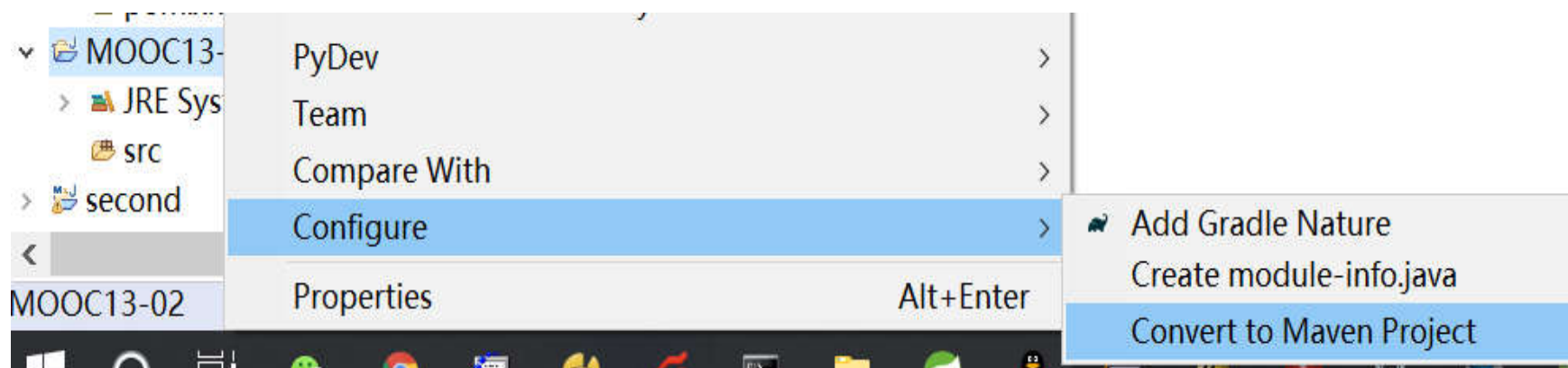
Advanced

< Back Next > Finish Cancel

# 创建Maven项目(3)



- 创建普通Java Project，再转化为Maven项目，修改pom.xml





## 创建Maven项目(4)

- 创建普通Java Project，再转化为Maven项目，修改pom.xml

The image shows a 'Create new POM' dialog box in an IDE. The title bar says 'Create new POM'. Below the title bar, it says 'Maven POM' and 'This wizard creates a new POM (pom.xml) descriptor for Maven.' The form contains several fields: 'Project:' with the value '/MOOC13-02', 'Artifact' section with 'Group Id:' and 'Artifact Id:' both set to 'MOOC13-02' (these two fields are highlighted with a red rectangle), 'Version:' set to '0.0.1-SNAPSHOT', 'Packaging:' set to 'jar', 'Name:' (empty), and 'Description:' (empty). At the bottom, there is a question mark icon, a 'Finish' button, and a 'Cancel' button.





# 修改pom.xml，添加jar依赖

```
<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.test</groupId>
  <artifactId>MOOC13-02Maven</artifactId>
  <version>0.0.1-SNAPSHOT</version>

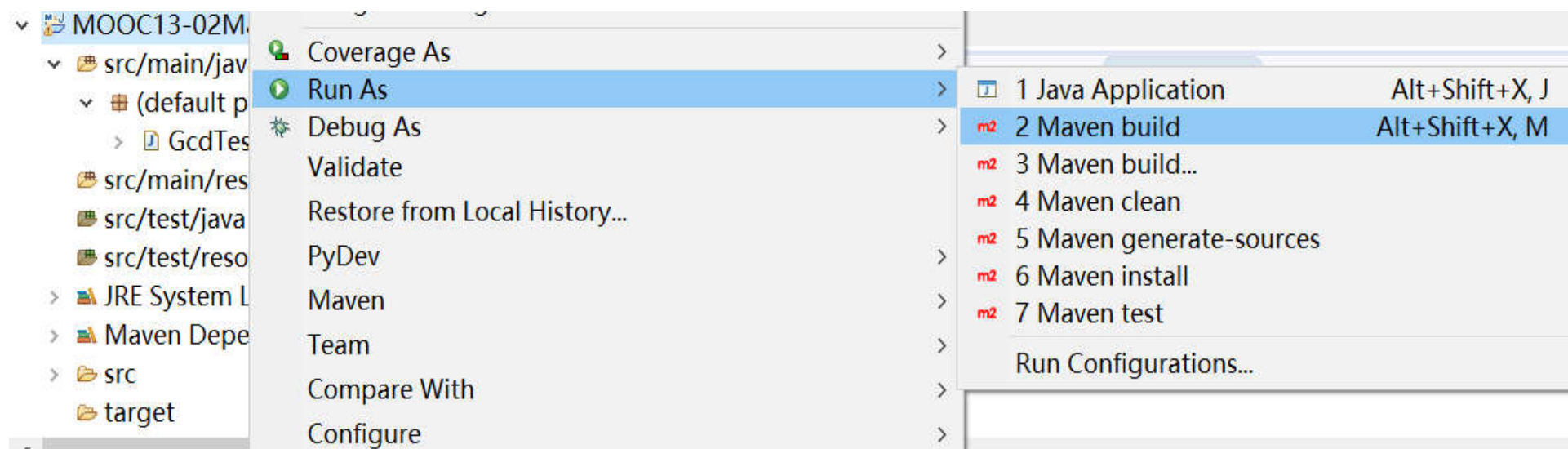
  <dependencies>
    <dependency>
      <groupId>org.apache.commons</groupId>
      <artifactId>commons-math3</artifactId>
      <version>3.6.1</version>
    </dependency>
  </dependencies>

</project>
```



# Maven 编译(1)

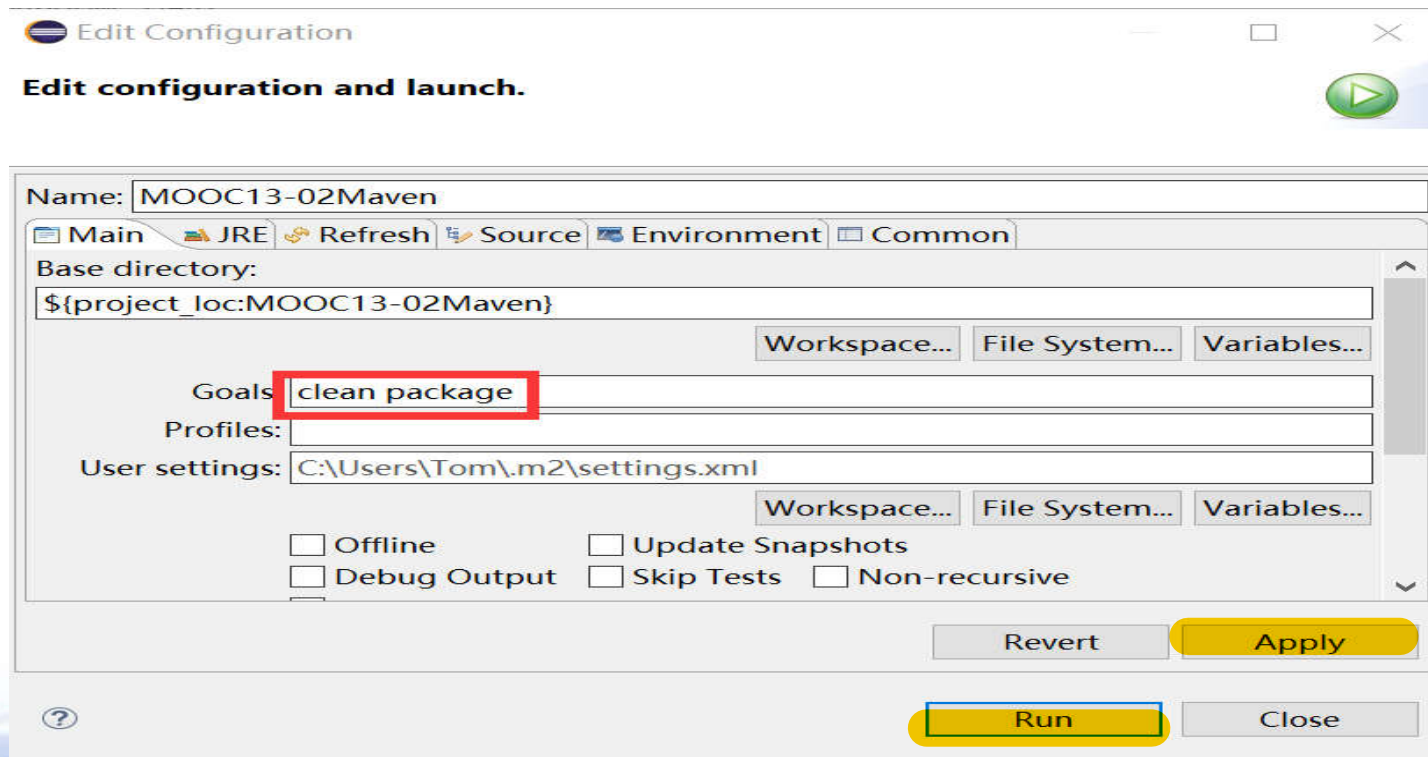
- 右键项目，Run As ->Maven build



# Maven 编译(2)



- 配置Maven build的内容(第一次配置, 后续不用配置)





# Maven 编译(3)



- 编译成功，控制台输出Build Success

```
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ MOOC13-02Maven ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e. build is platform
[INFO] Compiling 1 source file to E:\java\source\MOOC13-02Maven\target\classes
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ MOOC13-02Maven
[WARNING] Using platform encoding (GBK actually) to copy filtered resources, i.e. build is pl
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ MOOC13-02Maven ---
[INFO] Nothing to compile - all classes are up to date
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ MOOC13-02Maven ---
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ MOOC13-02Maven ---
[INFO] Building jar: E:\java\source\MOOC13-02Maven\target\MOOC13-02Maven-0.0.1-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.413 s
[INFO] Finished at: 2018-12-13T20:21:39+08:00
[INFO] -----
```

# Maven构建生命周期



- 清理
- 编译
- 测试
- 打包
- 安装
- 部署





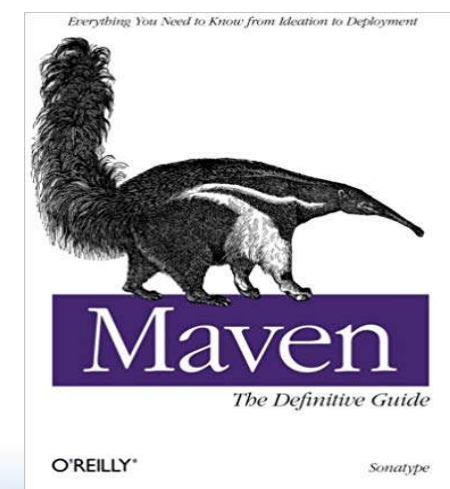
# 进一步研究

- 图书

- 《Maven实战》 许晓斌著，机械工业出版社。
- 《Maven: The definitive Guide》, Sonatype Company, O'Reilly.

- 同类工具

- Gradle
- Ivy
- Buildr
- Ant



# Maven总结



- 总结

- 理解Maven基础概念，如构件、仓库等
- 理解Maven在项目中的作用
- 掌握Maven项目创建、编译和运行



## 代码(1) GcdTest.java

```
import org.apache.commons.math3.util. ArithmeticUtils;

public class GcdTest {

    public static void main(String[] args) {
        //计算两个整数的公约数
        int a = ArithmeticUtils.gcd(361, 285);
        System.out.println(a);
    }
}
```



## 代码(2) ChineseText.java

```
import com.github.houbb.opencc4j.util.ZhConverterUtil;

public class ChineseText {

    public static void main(String[] args) {
        String original = "生命在于运动";

        //简体中文转化为繁体中文
        String result = ZhConverterUtil.convertToTraditional(original);

        System.out.println(result);
    }
}
```



# 代码(3) pom.xml



```
<dependencies>
  <dependency>
    <groupId>org.apache.commons</groupId>
    <artifactId>commons-math3</artifactId>
    <version>3.6.1</version>
  </dependency>

  <dependency>
    <groupId>com.github.houbb</groupId>
    <artifactId>openc4j</artifactId>
    <version>1.0.2</version>
  </dependency>
</dependencies>
```





谢谢!