Maven的依赖范围：

1

2

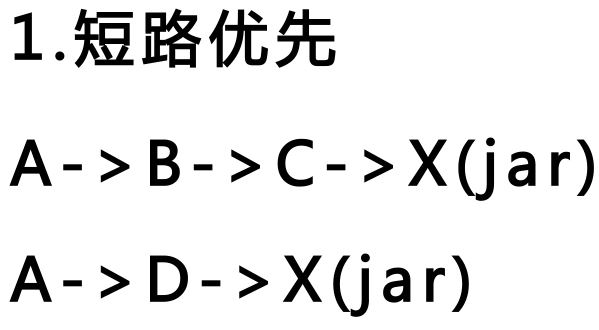
3



5



Maven依赖冲突处理原则：



Maven常见构建命令：



Maven的pom.xml配置简介：

<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"*>

<!-- pom版本。 -->

<modelVersion>4.0.0</modelVersion>

<groupId>com.yx</groupId><!-- 反写的公司网址+项目名。 -->

<artifactId>mavenTest</artifactId><!-- 项目名+项目的模块名。 -->

<version>0.0.1-SNAPSHOT</version>

<!-- 第一个0表示大版本号，第二个0表示分支版本号，第三个0表示小版本号。

snapshot快照，alpha内测版，beta公测版，Release稳定版，GA正式发布版本。 -->

<packaging>jar</packaging><!-- 打包方式。 -->

<name>mavenTest</name><!-- 项目描述名。 -->

<url>http://maven.apache.org</url><!-- 项目的地址。 -->

<developers></developers><!-- 开发人员列表。 -->

<licenses></licenses><!-- 许可证信息 -->

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

<dependencies><!-- 依赖列表。 -->

<dependency>

<groupId>junit</groupId><!-- 坐标 -->

<artifactId>junit</artifactId>

<version>3.8.1</version><!-- 版本 -->

<scope>test</scope><!-- 依赖的范围。 -->

<optional></optional><!-- 依赖是否可选。 -->

<exclusions></exclusions><!-- 排除依赖传递列表。 -->

</dependency>

</dependencies>

<dependencyManagement></dependencyManagement><!-- 依赖管理，为子模块提供依赖列表，本身中的依赖列表不可用。 -->

<build><!-- 构建支持。 -->

<plugins><!-- 插件列表。 -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-source-plugin</artifactId>

<version>2.4</version>

<executions><!-- 在打包时执行run。 -->

<execution>

<phase>package</phase>

<goals>

<goal>run</goal>

</goals>

</execution>

</executions>

</plugin>

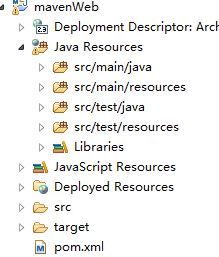
</plugins>

</build>

<!--<parent></parent> 子模块对父模块的继承。-->

<modules></modules><!-- 聚合多个模块。 -->

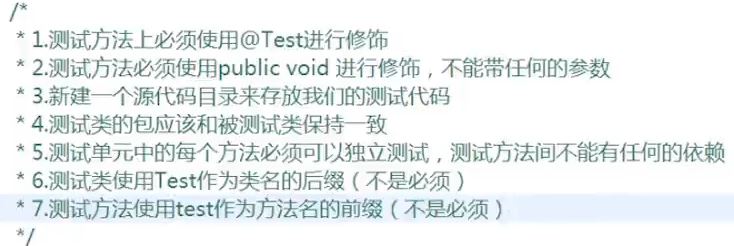
</project>

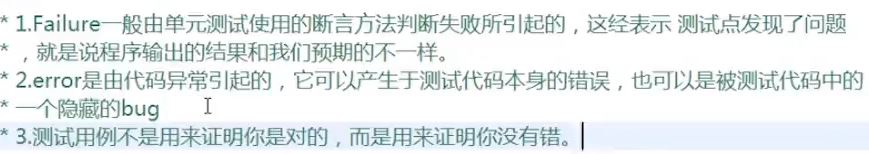


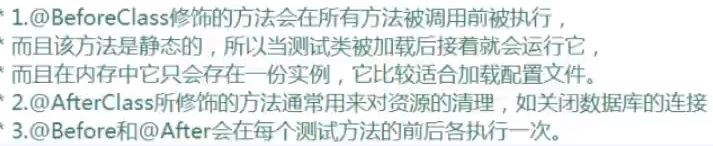
Main是存放主代码存放目录；

Test是测试代码存放目录。

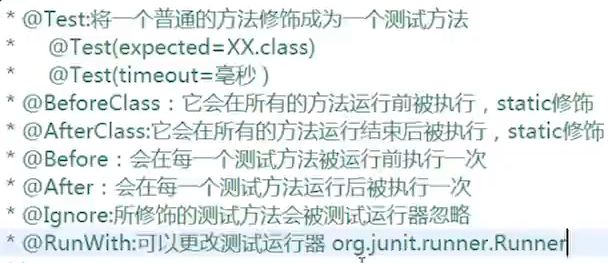
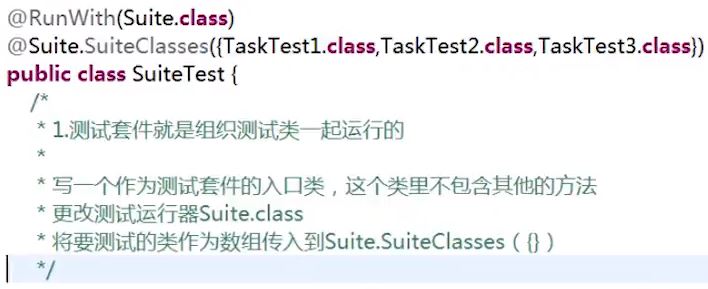
**单元测试：**

****

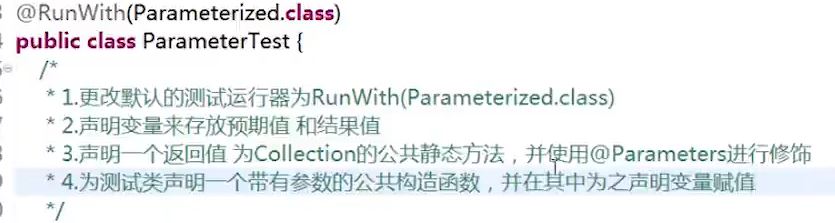
****

****

Before和after执行一定会被执行的代码



参数化设置：



代码见：mavenTest工程com.yx.mavenTest包。