# Java 9 多版本兼容 jar 包

多版本兼容 JAR 功能能让你创建仅在特定版本的 Java 环境中运行库程序时选择使用的 class 版本。

通过 **--release** 参数指定编译版本。

具体的变化就是 META-INF 目录下 MANIFEST.MF 文件新增了一个属性：

Multi-Release: true

然后 META-INF 目录下还新增了一个 versions 目录，如果是要支持 java9，则在 versions 目录下有 9 的目录。

multirelease.jar├── META-INF│ └── versions│ └── 9│ └── multirelease│ └── Helper.class├── multirelease

├── Helper.class

└── Main.class

在以下实例中，我们使用多版本兼容 JAR 功能将 Tester.java 文件生成了两个版本的 jar 包, 一个是 jdk 7，另一个是 jdk 9，然后我们再不同环境下执行。

**第一步**

创建文件夹 c:/test/java7/com/runoob，并在该文件夹下创建 Test.java 文件，代码如下：

package com.runoob;

public class Tester {

public static void main(String[] args) {

System.out.println("Inside java 7");

}}

**第二步**

创建文件夹 c:/test/java9/com/runoob，并在该文件夹下创建 Test.java 文件，代码如下：

package com.runoob;

public class Tester {

public static void main(String[] args) {

System.out.println("Inside java 9");

}}

编译源代码：

C:\test > javac --release 9 java9/com/runoob/Tester.java

C:\JAVA > javac --release 7 java7/com/runoob/Tester.java

创建多版本兼容 jar 包

C:\JAVA > jar -c -f test.jar -C java7 . --release 9 -C java9.Warning: entry META-INF/versions/9/com/runoob/Tester.java,

multiple resources with same name

使用 JDK 7 执行：

C:\JAVA > java -cp test.jar com.tutorialspoint.TesterInside Java 7

使用 JDK 9 执行：

C:\JAVA > java -cp test.jar com.tutorialspoint.TesterInside Java 9