# Java 9 模块系统

Java 9 最大的变化之一是引入了模块系统（Jigsaw 项目）。

模块就是代码和数据的封装体。模块的代码被组织成多个包，每个包中包含Java类和接口；模块的数据则包括资源文件和其他静态信息。

Java 9 模块的重要特征是在其工件（artifact）的根目录中包含了一个描述模块的 module-info.class 文 件。 工件的格式可以是传统的 JAR 文件或是 Java 9 新增的 JMOD 文件。这个文件由根目录中的源代码文件 module-info.java 编译而来。该模块声明文件可以描述模块的不同特征。

在 module-info.java 文件中，我们可以用新的关键词module来声明一个模块，如下所示。下面给出了一个模块com.mycompany.mymodule的最基本的模块声明。

module com.runoob.mymodule {}

## 创建模块

接下来我们创建一个 **com.runoob.greetings** 的模块。

**第一步**

创建文件夹 C:\>JAVA\src，然后在该目录下再创建与模块名相同的文件夹 com.runoob.greetings。

**第二步**

在 C:\>JAVA\src\com.runoob.greetings 目录下创建 module-info.java 文件，代码如下：

module com.runoob.greetings { }

module-info.java 用于创建模块。这一步我们创建了 com.runoob.greetings 模块。

**第三步**

在模块中添加源代码文件，在目录 C:\>JAVA\src\com.runoob.greetings\com\runoob\greetings 中创建文件 Java9Tester.java，代码如下：

package com.runoob.greetings;

public class Java9Tester {

public static void main(String[] args) {

System.out.println("Hello World!");

}}

**第四步**

创建文件夹 C:\>JAVA\mods，然后在该目录下创建 com.runoob.greetings 文件夹，编译模块到这个目录下：

C:/>JAVA> javac -d mods/com.runoob.greetings

src/com.runoob.greetings/module-info.java

src/com.runoob.greetings/com/runoob/greetings/Java9Tester.java

**第五步**

执行模块，查看输出结果：

C:/>JAVA> java --module-path mods -m com.runoob.greetings/com.runoob.greetings.Java9TesterHello World!

**module-path** 指定了模块所在的路径。

**-m** 指定主要模块。