

## Maven 构建 & 项目测试

在上一章节中我们学会了如何使用 Maven 创建 Java 应用。接下来我们要学习如何构建和测试这个项目。

进入 C:\MVN 文件夹下，打开 consumerBanking 文件夹。你将看到有一个 pom.xml 文件，代码如下：

```
<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/maven-v4_0_0.xsd"
>
<modelVersion>4.0.0</modelVersion>
<groupId>com.companyname.bank</groupId>
<artifactId>consumerBanking</artifactId>
<packaging>jar</packaging>
<version>1.0-SNAPSHOT</version>
<name>consumerBanking</name>
<url>http://maven.apache.org</url>
<dependencies>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>3.8.1</version>
<scope>test</scope>
</dependency>
</dependencies>
</project>
```

从以上 xml 代码中，可知 Maven 已经添加了 JUnit 作为测试框架。

默认情况下 Maven 添加了一个源码文件 C:\MVN\consumerBanking\src\main\java\com\companyname\bank\App.java 和一个测试文件 C:\MVN\consumerBanking\src\test\java\com\companyname\bank\AppTest.java。

打开命令控制台，跳转到 C:\MVN\consumerBanking 目录下，并执行以下 mvn 命令开始构建项目：

```
C:\MVN\consumerBanking>mvn clean package
[INFO] Scanning for projects...
[INFO] -----
[INFO] Building consumerBanking
[INFO]    task-segment: [clean, package]
[INFO] -----
[INFO] [clean:clean {execution: default-clean}]
[INFO] Deleting directory C:\MVN\consumerBanking\target
...
...
...
[INFO] [jar:jar {execution: default-jar}]
[INFO] Building jar: C:\MVN\consumerBanking\target\
consumerBanking-1.0-SNAPSHOT.jar
[INFO] -----
```

```
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 2 seconds
[INFO] Finished at: Tue Jul 10 16:52:18 IST 2012
[INFO] Final Memory: 16M/89M
[INFO] -----
```

执行完后，我们已经构建了自己的项目并创建了最终的 jar 文件，下面是要学习的关键概念：

- 我们给了 maven 两个目标，首先清理目标目录（clean），然后打包项目构建的输出为 jar（package）文件。
- 打包好的 jar 文件可以在 consumerBanking\target 中获得，名称为 consumerBanking-1.0-SNAPSHOT.jar。
- 测试报告存放在 consumerBanking\target\surefire-reports 文件夹中。
- Maven 编译源码文件，以及测试源码文件。
- 接着 Maven 运行测试用例。
- 最后 Maven 创建项目包。

```
C:\MVN\consumerBanking\target\classes>java com.companyname.bank.App
```

你可以看到结果：

```
Hello World!
```

添加 Java 源文件

接下来我们看看如何添加其他的 Java 文件到项目中。打开 C:\MVN\consumerBanking\src\main\java\com\companyname\bank 文件夹，在其中创建 Util 类 Util.java。

#### Util.java

```
package com.companyname.bank;
public class Util
{
    public static void printMessage(String message){
        System.out.println(message);
    }
}
```

更新 App 类来使用 Util 类:

#### App.java

```
package com.companyname.bank;
/**
 * Hello world!
 *
 */
public class App
{
```

```
public static void main( String[] args )
{
    Util.printMessage("Hello World!");
}
```

现在打开命令控制台，跳转到 C:\MVN\consumerBanking 目录下，并执行下面的 mvn 命令。

```
C:\MVN\consumerBanking>mvn clean compile
```

在 Maven 构建成功之后，跳转到 C:\MVN\consumerBanking\target\classes 目录下，并执行下面的 java 命令。

```
C:\MVN\consumerBanking\target\classes>java -cp . com.companyname.bank.App
```

你可以看到结果：

```
Hello World!
```

[← Maven 构建 Java 项目](#)

[Maven 引入外部依赖 →](#)

[✎ 点我分享笔记](#)