

创建

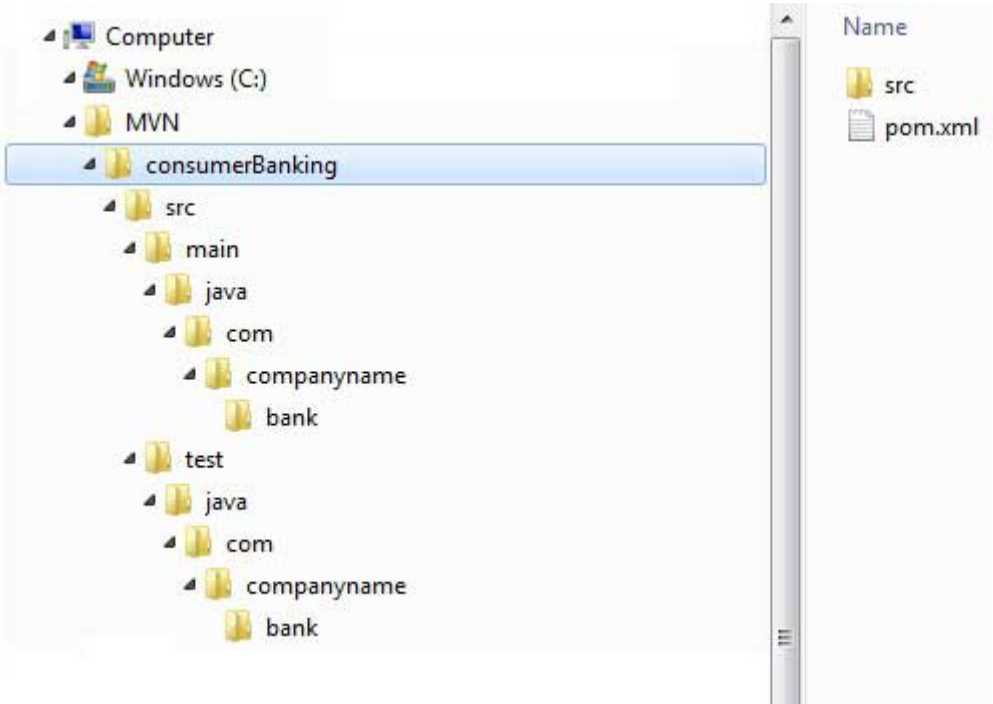
Maven 使用原型 archetype 插件创建项目。要创建一个简单的 Java 应用，我们将使用 maven-archetype-quickstart 插件。

```
mvn archetype:generate -DgroupId=com.companyname.bank -
DartifactId=consumerBanking -DarchetypeArtifactId=maven-archetype-quickstart -
DinteractiveMode=false
```

参数说明：

- **-DgroupId:** 组织名，公司网址的反写 + 项目名称
- **-DartifactId:** 项目名-模块名
- **-DarchetypeArtifactId:** 指定 ArchetypeId，maven-archetype-quickstart，创建一个简单的 Java 应用
- **-DinteractiveMode:** 是否使用交互模式

生成的文件夹结构如下：



文件夹结构	描述
consumerBanking	包含 src 文件夹和 pom.xml
src/main/java contains	java 代码文件在包结构下（com/companyName/bank）。
src/main/test contains	测试代码文件在包结构下（com/companyName/bank）。

src/main/resources | 包含了 图片 / 属性 文件（在上面的例子中，我们需要手动创建这个结构）。

项目模板不止quickstar（默认），执行以下 mvn 命令：

```
C:\MVN> mvn archetype:generate
```

Maven 将开始处理，并要求选择所需的原型。

构建 & 项目测试

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：

- 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 文件可以放在 consumerBanking\target 中，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。

在 C:\MVN\consumerBanking\src\main\java\com\companyname\bank 文件夹中创建 Util 类 Util.java。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!
```