

Maven 构建 Java 项目

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。在下面的例子中，我们将在 C:\MVN 文件夹下创建一个基于 maven 的 java 应用项目。

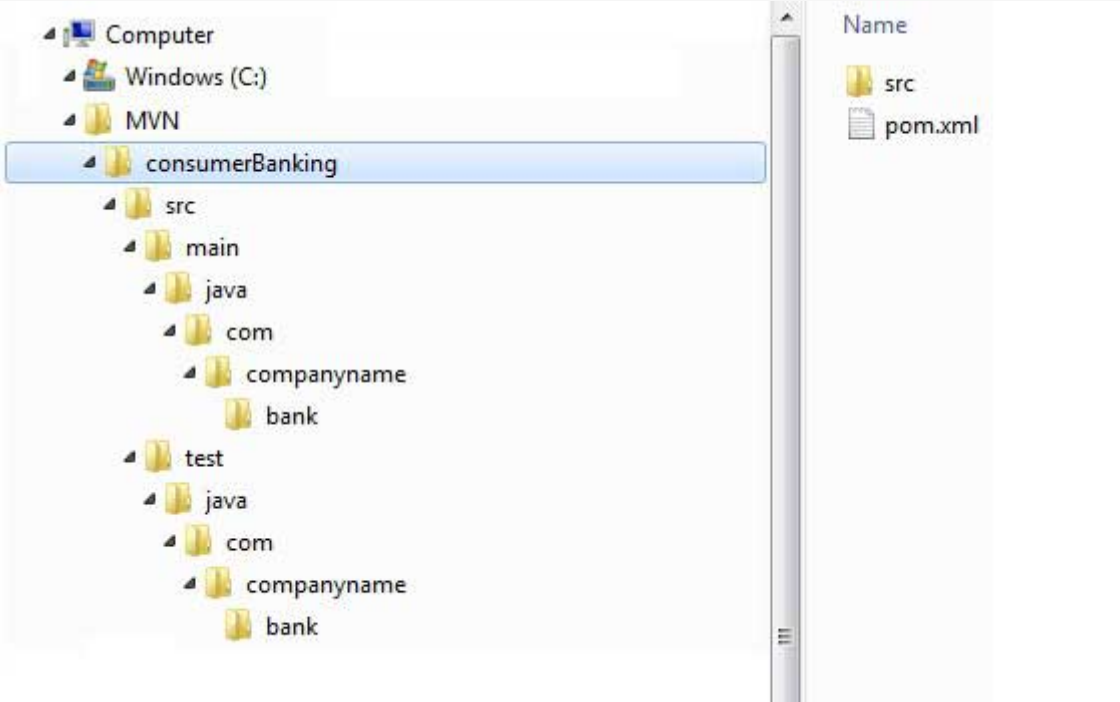
命令格式如下：

```
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：



各个文件夹说明：

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) 。

文件夹结构	描述
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) 。

在 C:\MVN\consumerBanking\src\main\java\com\companyname\bank 文件夹中 , 可以看到一个 App.java , 代码如下 :

App.java

```
package com.companyname.bank;
/**
 * Hello world!
 */
public class App
{
    public static void main( String[] args )
    {
        System.out.println( "Hello World!" );
    }
}
```

打开 C:\MVN\consumerBanking\src\test\java\com\companyname\bank 文件夹 , 可以看到 Java 测试文件 AppTest.java。

AppTest.java

```
package com.companyname.bank;
import junit.framework.Test;
import junit.framework.TestCase;
import junit.framework.TestSuite;
/**
 * Unit test for simple App.
 */
public class AppTest extends TestCase
{
    /**
     * Create the test case
     *
     * @param testName name of the test case
     */
    public AppTest( String testName )
    {
        super( testName );
    }
    /**
     * @return the suite of tests being tested
     */
    public static Test suite()
    {
        return new TestSuite( AppTest.class );
    }
    /**
     * Rigorous Test :-)
     */
}
```

```
public void testApp()  
{  
    assertTrue( true );  
}  
}
```

接下来的开发过程中我们只需要按照上面表格中提到的结构放置好，其他的事情 Maven 帮我们将会搞定。

[← Maven 插件](#)[Maven 构建 & 项目测试 →](#)

1 篇笔记

[写笔记](#)

IDEA 创建 MAVEN 项目卡在 Generating project in Batch mode。是 MAVEN 一直在请求：

```
http://repo1.maven.org/maven2/archetype-catalog.xml
```

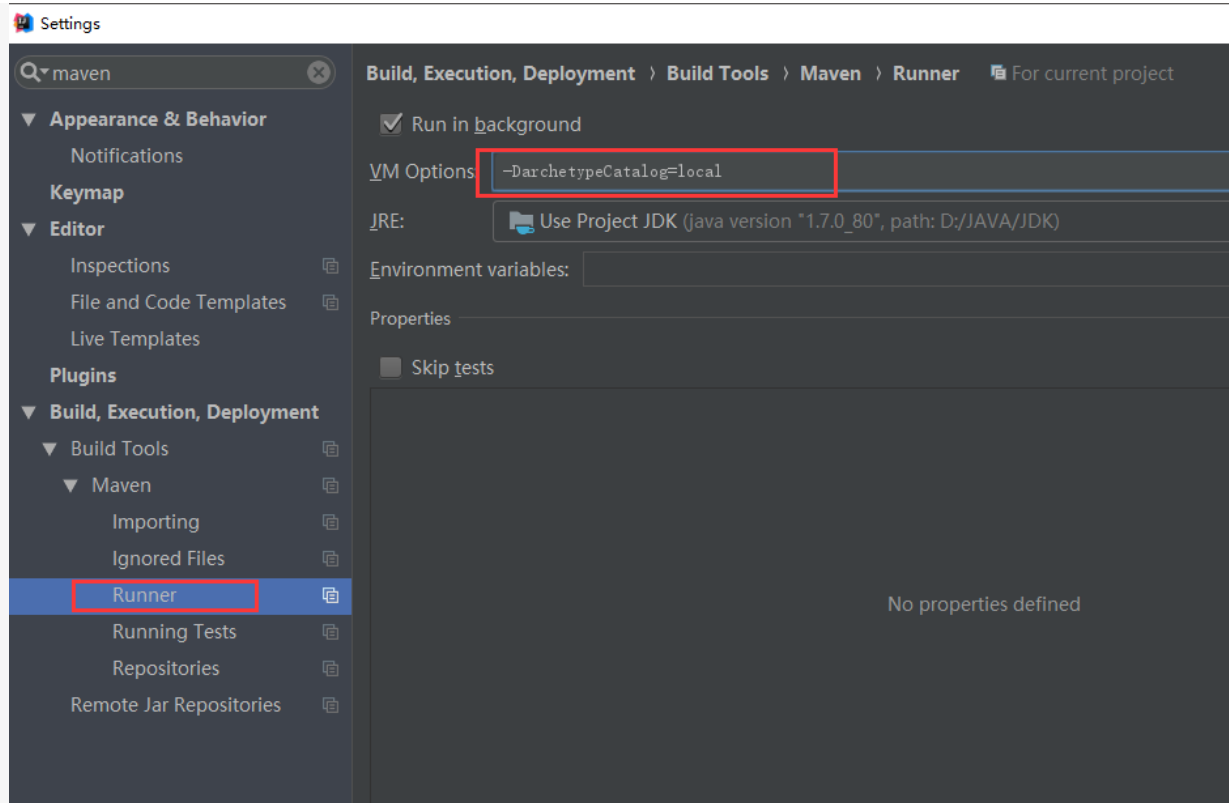
这个项目目录的 xml 文件，由于墙的问题，加载缓慢，等待时间十分漫长。

解决方法：修改 setting.xml 文件，把 mirror 改成阿里的镜像。

- 1、打开 maven 目录下的 conf/setting.xml，注意要是 idea 使用的 maven。
- 2、搜索 <mirrors>；找到 <mirrors>。在 <mirrors> 节点下添加。

```
<mirror>  
    <id>alimaven</id>  
    <name>aliyun maven</name>  
    <url>http://maven.aliyun.com/nexus/content/groups/public/</url>  
    <mirrorOf>central</mirrorOf>  
</mirror>
```

- 3、运行 MVN 命令的时候加上 **-DarchetypeCatalog=local** 这个参数。在 IDEA 中：



完成！

duduupup 3周前 (02-22)