

# 1、maven约定的项目目录结构

## 1) 约定：就是大家都遵从的一个规则

--每一个maven项目在磁盘中都是一个文件夹：假设这里项目名为 **Hello**，目录结构如下

```
Hello/
  ---/src
  -----/main          #存放主程序java代码和配置文件
  -----/java          #程序中创建的包和包中的Java代码文件
  -----/resources      #Java程序中需要用到的配置文件（如SQL映射文件）

  -----/test          #存放测试的程序代码和配置文件（不强制要求一定要有）
  -----/java          #测试程序的包和包中的Java代码文件
  -----/resources      #测试Java程序中所需要的配置文件

  ---/pom.xml           #maven的核心文件（maven项目必须要有）
```

maven启动的时候先找pom文件，知道要干什么，然后到main中找程序，到test中找测试程序

## 2) 在Projects文件中按照上面的目录结构新建一个Hello项目

注意src目录与pom.xml文件同级，pom.xml文件内容如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<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/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <!--上面的东西都是固定的-->

  <!--下面三个称之为坐标，三个值来指定项目的名称的唯一标识-->

  <!-- 公司或者组织的唯一标志，并且配置时生成的路径也是由此生成，这里先简写-->
  <groupId>com.studymyself</groupId>
  <!--项目名字-->
  <artifactId>maven-01</artifactId>
  <!--项目的版本号-->
  <version>1.0-SNAPSHOT</version>

</project>
```

在main目录的java文件夹中创建com.studymyself包，其中创建一个HelloMaven类，代码如下

## com.studymyself.HelloMaven.java

```
package com.myselfstudy

public class HelloMaven{
```

```

public int add(int a,int b){

    return a+b;

}

public static void main(String[] args){

    HelloMaven hello = new HelloMaven();

    int res = hello.add(10,20);

    System.out.println("10+20="+res);

}

}

```

在Hello项目的根目录下，即pom.xml文件所在目录下启动cmd，在dos命令窗口中执行命令：mvn compile  
意思是编译main目录下的java目录下的所有java文件

**出现编译错误，重新下载了maven3.3.9，将原来的覆盖，后发现还是解决不了问题，找出错误原因是maven工具所用的settings文件是系统盘.m2目录中的settings文件，为什么？如下面所示**

```

F:\Git_Repositories\Maven\Projects\Hello>mvn compile
[ERROR] Error executing Maven.
[ERROR] 1 problem was encountered while building the effective settings
[FATAL] Non-parseable settings C:\Users\钟荣杰\.m2\settings.xml: Duplicated tag:
'mirrorOf' (position: START_TAG seen ...</url> \ua0\n    <mirrorOf>... @172:15)
@ C:\Users\钟荣杰\.m2\settings.xml, line 172, column 15

```

**1.Maven**的配置文件（Maven的安装目录/conf/settings.xml）和 Maven仓库下（默认Maven仓库的是用户家目录下的.m2文件，可以另行制定）的settings.xml文件  
在Maven中提供了一个settings.xml文件来定义Maven的全局环境信息。--这个文件会存在于Maven的安装目录的conf子目录下面，或者是用户家目录的.m2子目录下面。我们可以通过这个文件来定义本地仓库、远程仓库和联网使用的代理信息等。

其实相对于多用户的PC机而言，在Maven安装目录的conf子目录下面的settings.xml才是真正的全局的配置。而--用户家目录的.m2子目录下面的settings.xml的配置只是针对当前用户的。

--当这两个文件同时存在的时候，那么对于相同的配置信息用户家目录下面的settings.xml中定义会覆盖Maven安装目录下面的settings.xml中的定义。

用户家目录下的settings.xml文件一般是不存在的，但是Maven允许我们在这里定义我们自己的settings.xml，如果需要在本地定义我们自己的settings.xml的时候就可以把Maven安装目录下面的settings.xml文件拷贝到用户家目录的.m2目录下，然后改成自己想要的样子。

**修改.m2目录中的settings用conf目录中的替换，再次编译**

```

F:\Git_Repositories\Maven\Projects\Hello>mvn compile

```

```

[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building maven-01 1.0-SNAPSHOT
[INFO] -----
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ maven-01 -
--
[WARNING] Using platform encoding (GBK actually) to copy filtered resources,
i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ maven-01 ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding GBK, i.e.
build is platform dependent!
[INFO] Compiling 1 source file to
F:\Git_Repositories\Maven\Projects\Hello\target\classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.201 s
[INFO] Finished at: 2021-01-26T19:47:46+08:00
[INFO] Final Memory: 14M/207M
[INFO] -----

```

--注意:

如果是第一次使用maven工具，会自动下载maven工具需要使用的插件（java类），形式都是jar文件  
从中央仓库地址下载：<https://repo.maven.apache.org>

存储到上面所讲的.m2目录的repository目录中:

默认仓库位置（可修改）：C:\Users\（登录操作系统的用户名）\.m2\repository

--因为之前用过，已经下载好了

--编译成功的结果是在项目的目录中生成target目录，编译的class文件都放在这里  
直接java HelloMaven执行

--修改本机存放资源的仓库位置

通过修改settings文件，来修改仓库位置，怎么修改在配置文件中是有注释的，如下:

```

<!-- localRepository
    | The path to the local repository maven will use to store artifacts.
    |
    | Default: ${user.home}/.m2/repository
<localRepository>/path/to/local/repo</localRepository>
-->

```

可以看到上面的注释中告诉了我们默认仓库位置：\${user.home}/.m2/repository

怎么修改位置：<localRepository>/path/to/local/repo</localRepository>

我们在settings文件中有上面这段注释的下方添加:

```

<!-- localRepository
    | The path to the local repository maven will use to store artifacts.
    |
    | Default: ${user.home}/.m2/repository
<localRepository>/path/to/local/repo</localRepository>
-->
<localRepository>E:/maven/myrepository</localRepository>

```

这样保存就修改成功了，要注意路径是左斜杠/

如果.m2目录中有**settings.xml**文件，则在该目录下修改这个配置文件，否者就修改**conf**目录中的**settings**文件

以后如果要配置什么，就直接看里面的注释，按注释中的步骤修改添加。修改前必须备份，放在原目录，等待以后备用

还有就是路径必须是非中文的，无论在什么开发中一定杜绝使用中文路径！！！！