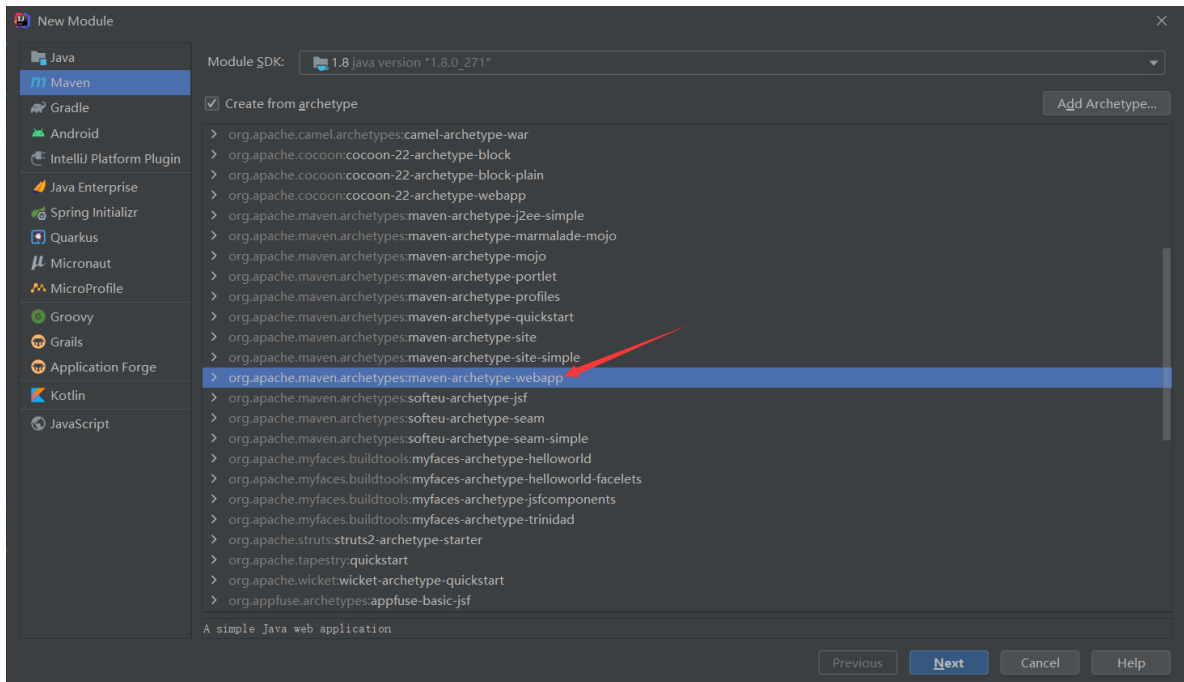
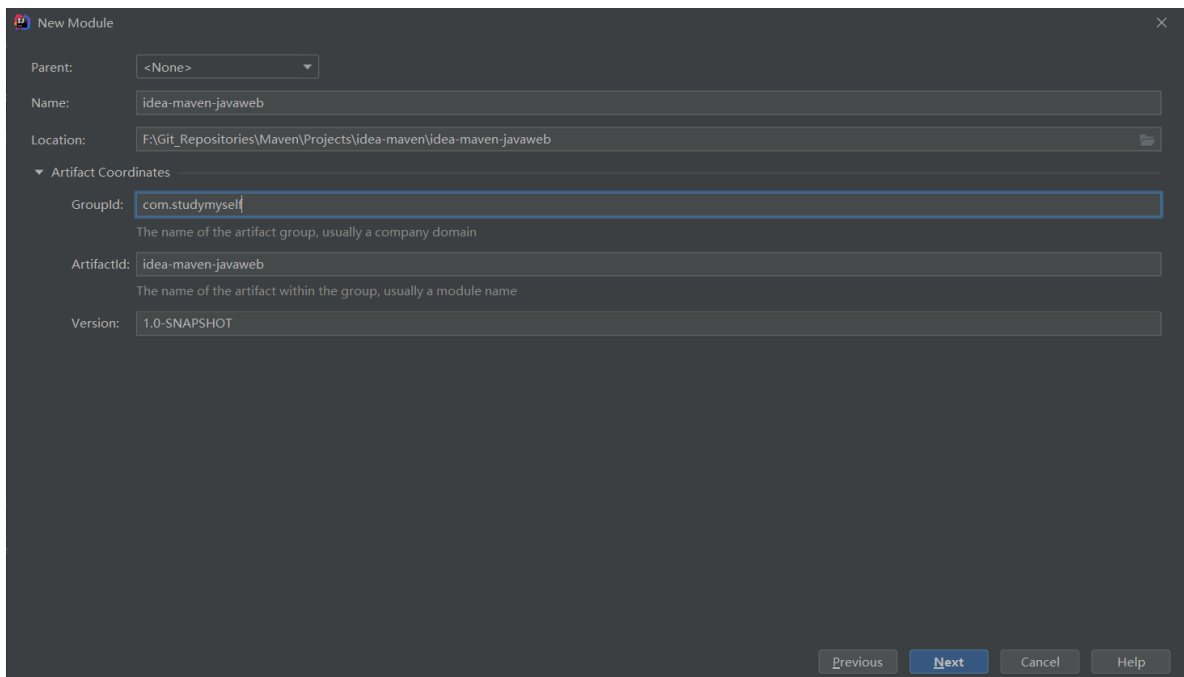


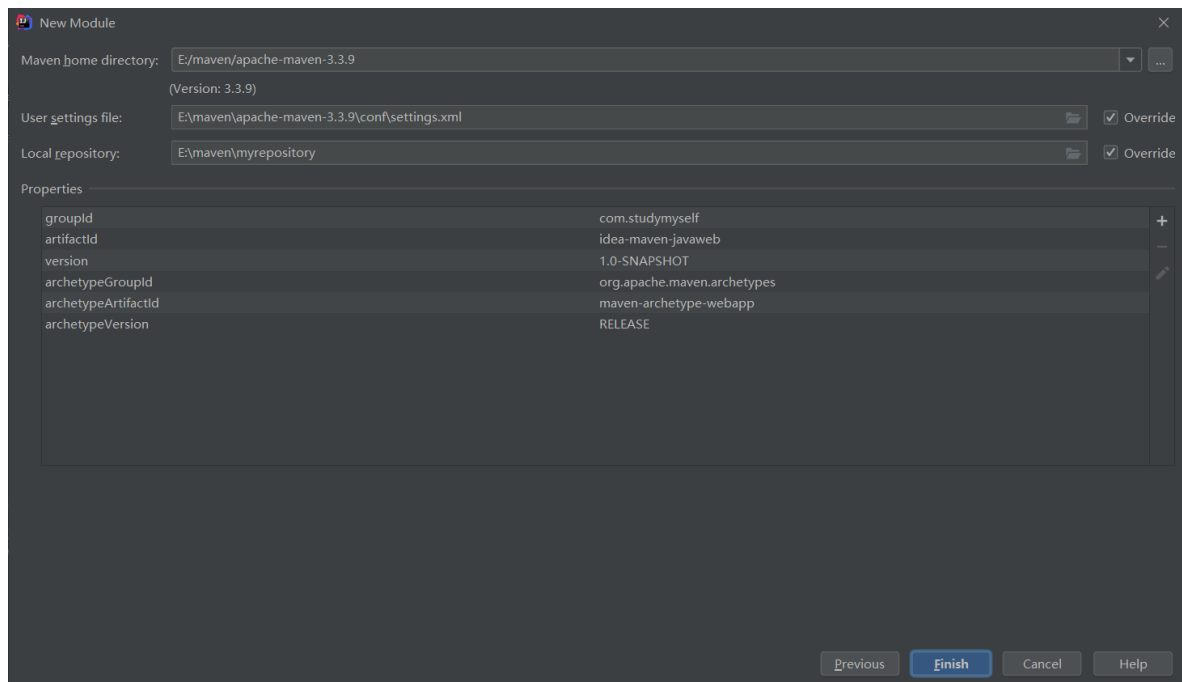
## 1、在工程idea-maven中添加新模块，选择maven，其中勾选模板是maven-archetype-webapp



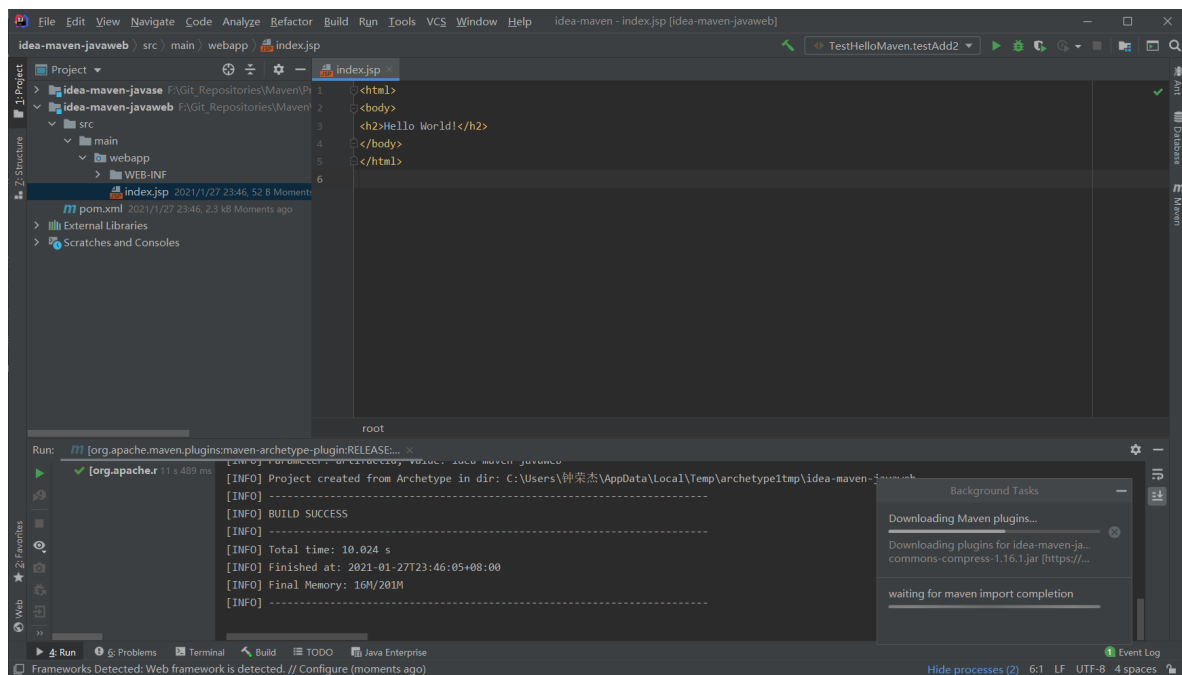
## 2、不继承项目，项目名字idea-maven-javaweb



## 3、检查maven设置是否有误，无误finish，有就改好

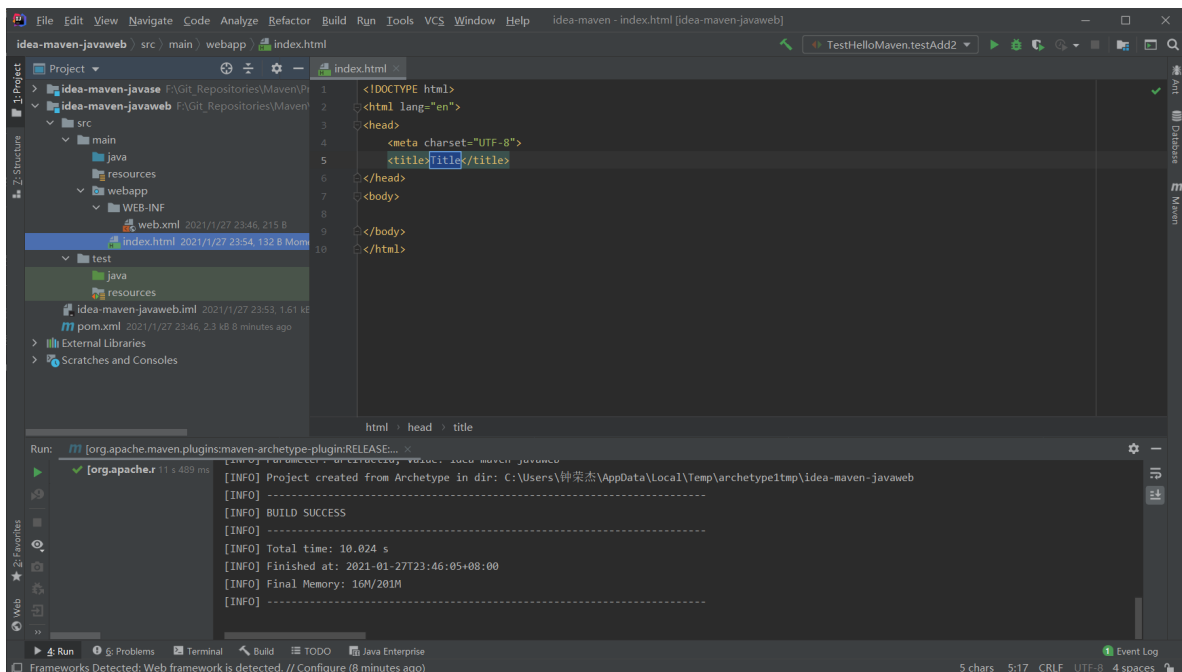


## 4、创建好后完整目录如下

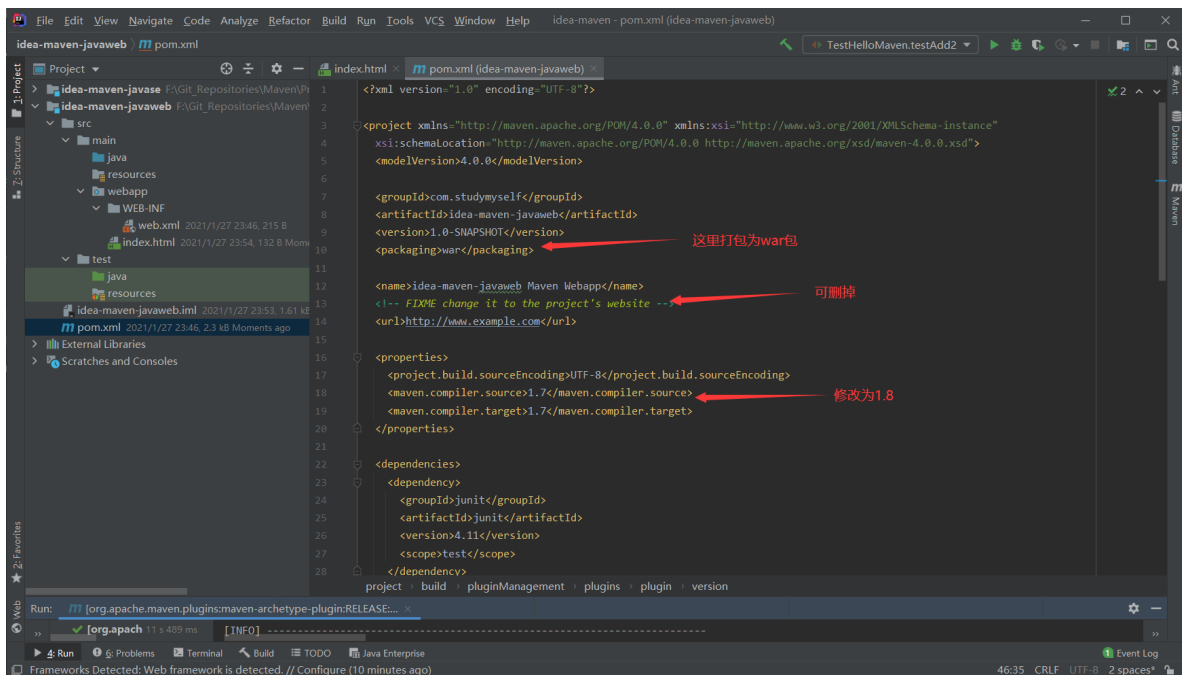


发现main中并没有java源文件夹，自己在main目录下创建java，然后定义为源文件夹Sources Root；再创建一个resources文件夹定义为源资源文件夹Resources Root  
如果以后还需要测试的话，再在src下新建test，然后创建java和resources，各自定义相应类型。

然后删掉jsp文件，新建一个index.html文件  
完整的maven的web项目结构如下：



## 5、打开pom文件



可以看到由于添加的模板不同，是web项目，所以打包被定义为打成war包。  
下面的与之前普通java项目的一样。

## 5、添加Tomcat容器

在maven项目中就不需要在Structure Projects中的Dependencies中选择Tomcat的jar包了，直接在中央仓库搜索Tomcat，找对应版本的坐标依赖代码粘贴到pom文件中的依赖标签中就行了，因为搜索出来的很多，不知道哪一个，就直接百度对应博客，出来的更准确。

~~<!-- 下面这个是错误的-->~~

```

<dependency>
  <groupId>org.apache.tomcat</groupId>
  <artifactId>tomcat-catalina</artifactId>
  <version>9.0.41</version>
</dependency>

```

~~<!-- 这个才是正确的-->~~

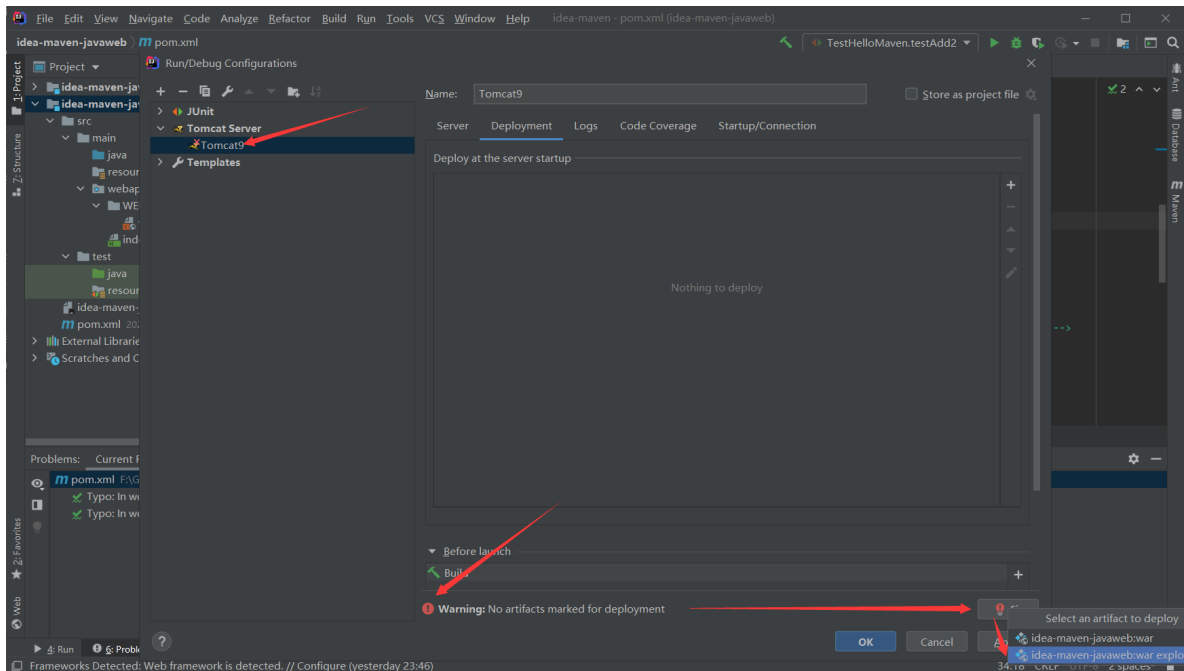
```
<dependency>
  <groupId>org.apache.tomcat</groupId>
  <artifactId>tomcat-servlet-api</artifactId>
  <version>9.0.41</version>
</dependency>
```

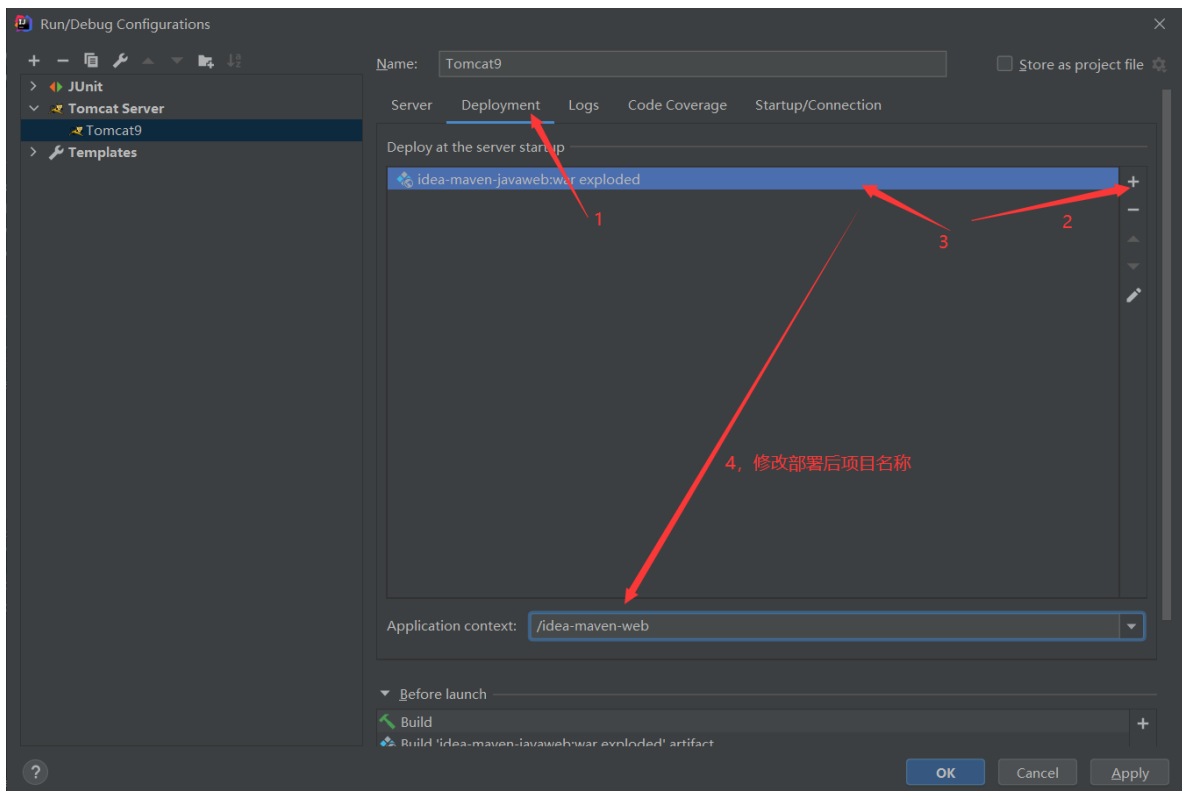
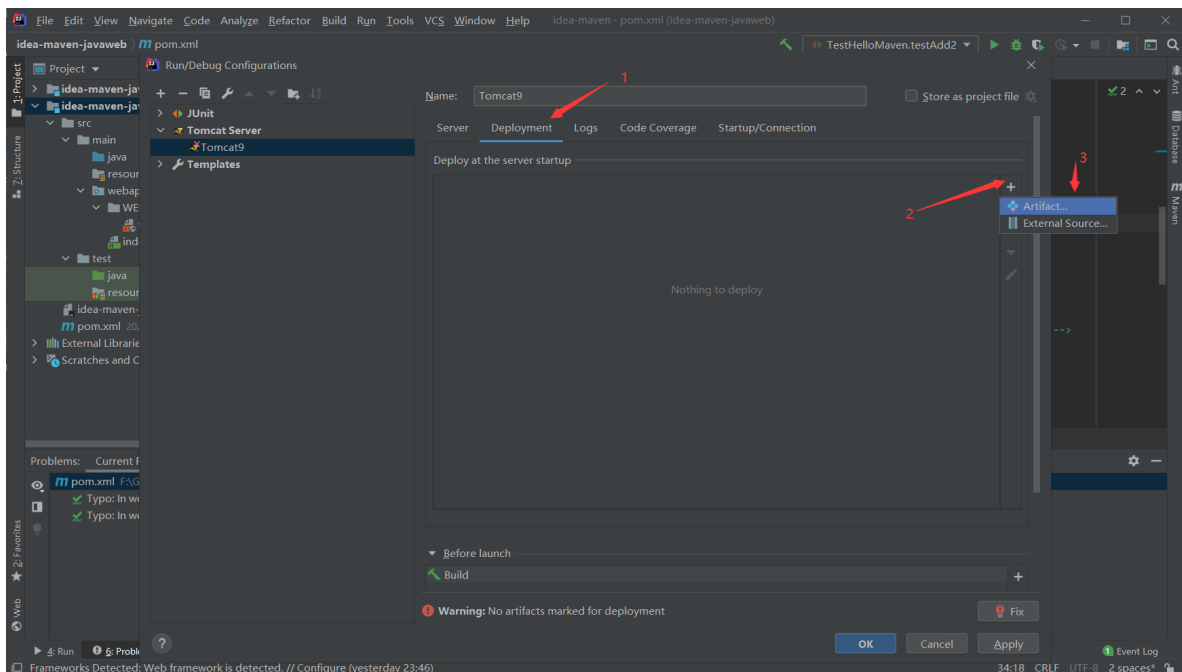
添加进去后，发现版本号是红的，说明没有刷新pom文件加载对应依赖，点击maven出现maven窗口，刷新Reload所有maven工程，如果仓库有的话就可以加载进项目中了，没有就到默认的镜像网址中下载。还有一种方法就是修改了 maven 依赖以后，当前 pom 文件的右上角会出现一个 maven 的小图标，点一下就可以更新依赖了。或者快捷键：Ctrl + Shift + O

## 添加完web容器依赖后，配置选择容器

Edit Configurations->加号+>Tomcat Server->Local->点击下边Warning右边的fix，选择带有explode的项目

或者这样添加部署项目 Local->Deployment->右边加号->Artifacts->选择带有explode的项目





在index.html中编写一个超链接，连接到一个Servlet资源类

新建一个Servlet资源类

com.studymyself.controller.HelloServlet

```
package com.studymyself.controller;

import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.IOException;

public class HelloServlet extends HttpServlet {
```

```

@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

    response.setContentType("text/html;charset=utf-8");

    response.getWriter().print("大家好，这是一个使用Maven构建工具创建的web项目");

}

@Override
protected void doPost(HttpServletRequest req, HttpServletResponse resp)
throws ServletException, IOException {

}
}

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>welcome Page</title>
</head>
<body>
    <h1>欢迎光临! </h1>
    <a href="/idea-maven-web/helloMaven">点击进入</a>
</body>
</html>

```

我们在maven窗口打开该web项目的生命周期，双击package，maven帮我们将项目打包成一个war包：  
idea-maven-javaweb.war，我们从硬盘中将该war文件，改名为myweb.war，放进Tomcat目录中的webapps目录中，然后在命令窗口setup启动服务器，这样这个项目就开始发挥功能了，我们在浏览器中输入http://localhost:8080/myweb/回车，就能到该项目的欢迎页面了。这个war包就是我们的最终成品，交给客户的。

#### --导入maven项目模块

File->New->Module From Existing Sources->选择项目,ok->选择Import module from external model, 选择Maven项目, ok->Apply

也可以在Project Structure的ProjectSettings的Modules中点击加号选择新建还是导入，导入后选中项目，在右边的Dependencies属性中选择Module SDK为JDK1.8，然后应用。