# 1.手动部署web项目

1)到tomcat的Lib目录下拷贝servlet-api.jar导入到项目中

2)编写一个普通的java类继承HttpServlet类,覆盖doGet方法

|  |
| --- |
| package com.company;  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 req, HttpServletResponse resp) throws ServletException, IOException {  resp.setContentType("text/html;charset=utf-8");  resp.getWriter().write("hello,world");  }  } |

3) 把上述类生成的包+类文件拷贝到tomcat

的某个项目的WEB-INF/classes目录下，这时路径为:

F:\共享目录\server\apache-tomcat-8.5.23\apache-tomcat-8.5.23\webapps\web\WEB-INF\classes\com\company\ 里有个文件HelloServlet.class

4)在WEB-INF下新建web.xml内容: 注意这个文件要保存为utf-8

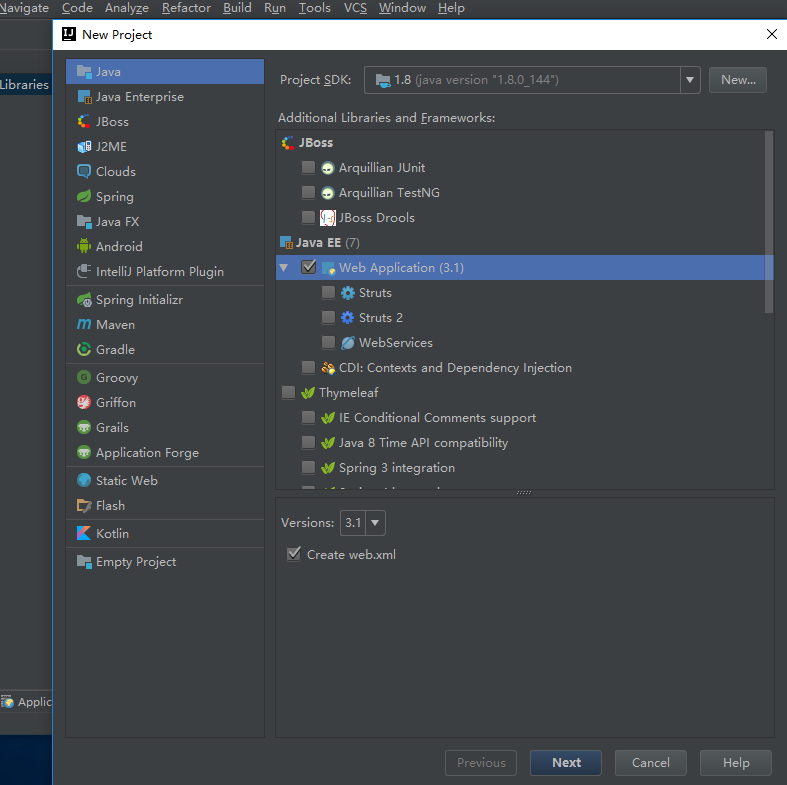
|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee  http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"  version="3.1">  <!--配置一个servlet -->  <servlet>  <!--servlet名称,自定义-->  <servlet-name>HelloServlet</servlet-name>  <!--servlet类的全名:包名+类名-->  <servlet-class>com.company.HelloServlet</servlet-class>  </servlet>  <servlet-mapping>  <!--servlet名称,和上面名称保持一直-->  <servlet-name>HelloServlet</servlet-name>  <!--公开给外部访问这个servlet的名称(路径)-->  <url-pattern>/hello</url-pattern>  </servlet-mapping>    </web-app> |

4)通过url访问:

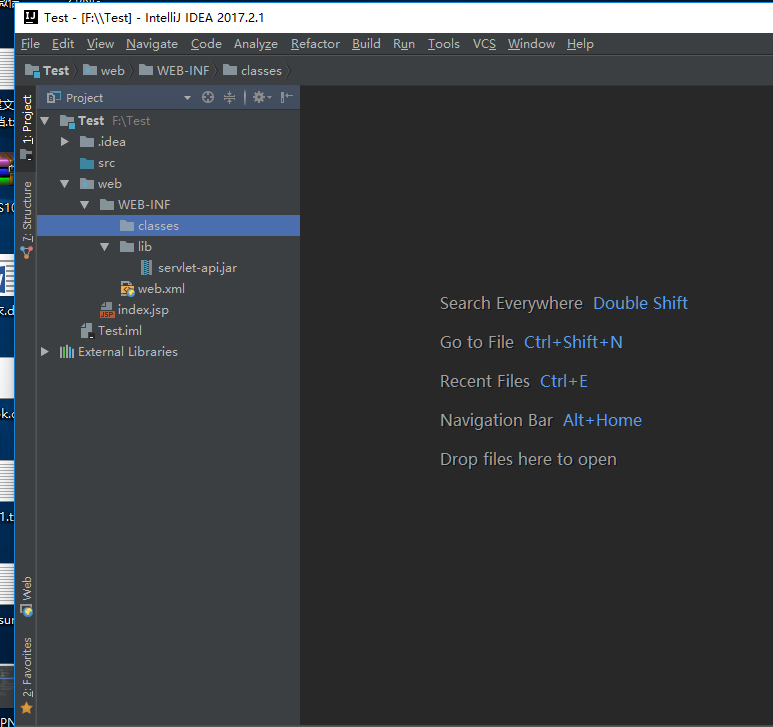
<http://localhost:8080/web/hello>

# 2.idea部署web项目

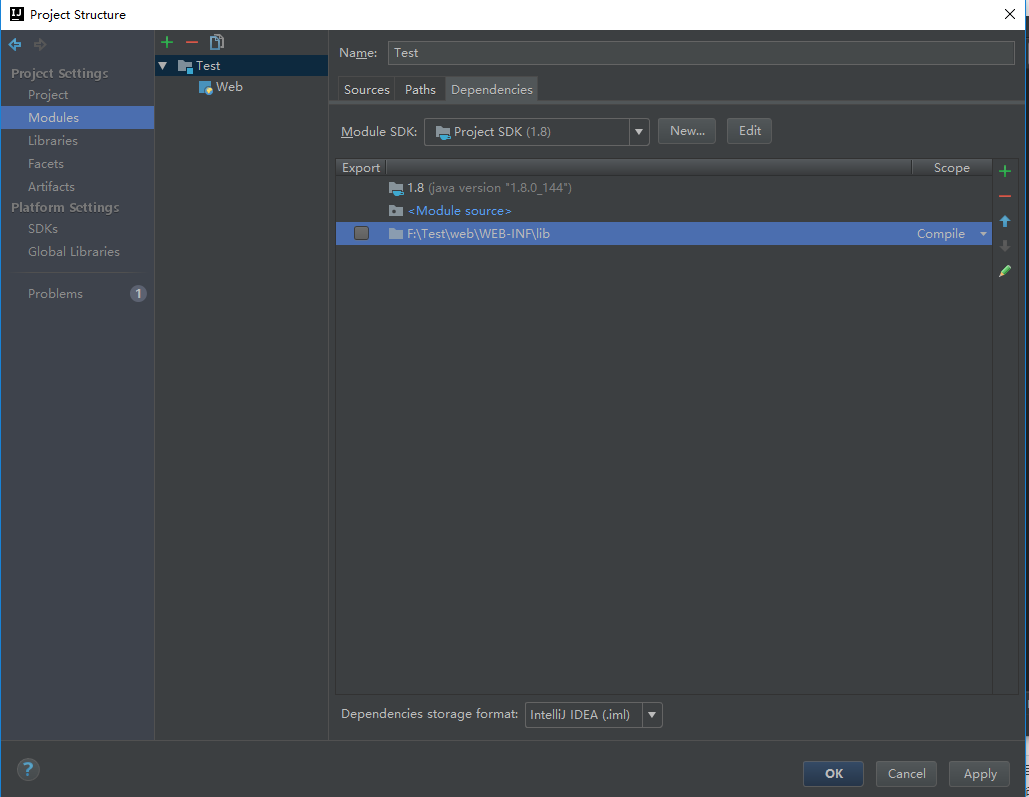
1)新建一个web项目取名为Test



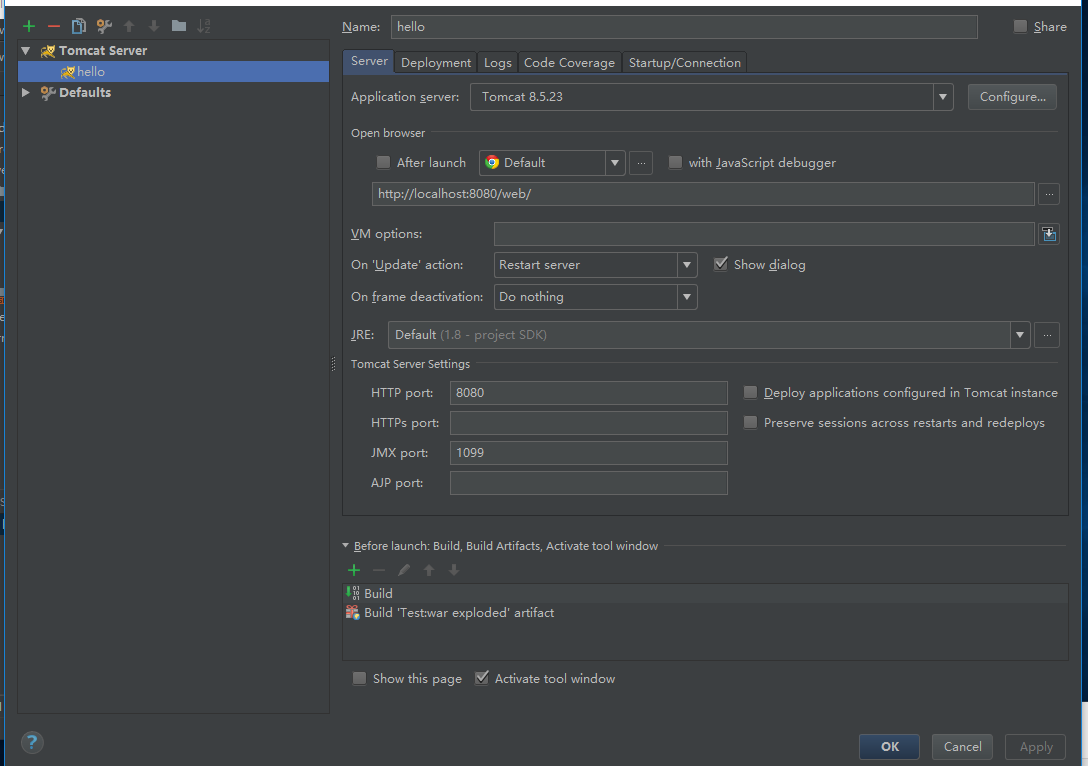
2)在WEB-INF目录下手动新建两个目录分别为classes和lib,在将servlet-api.jar包复制到lib目录下

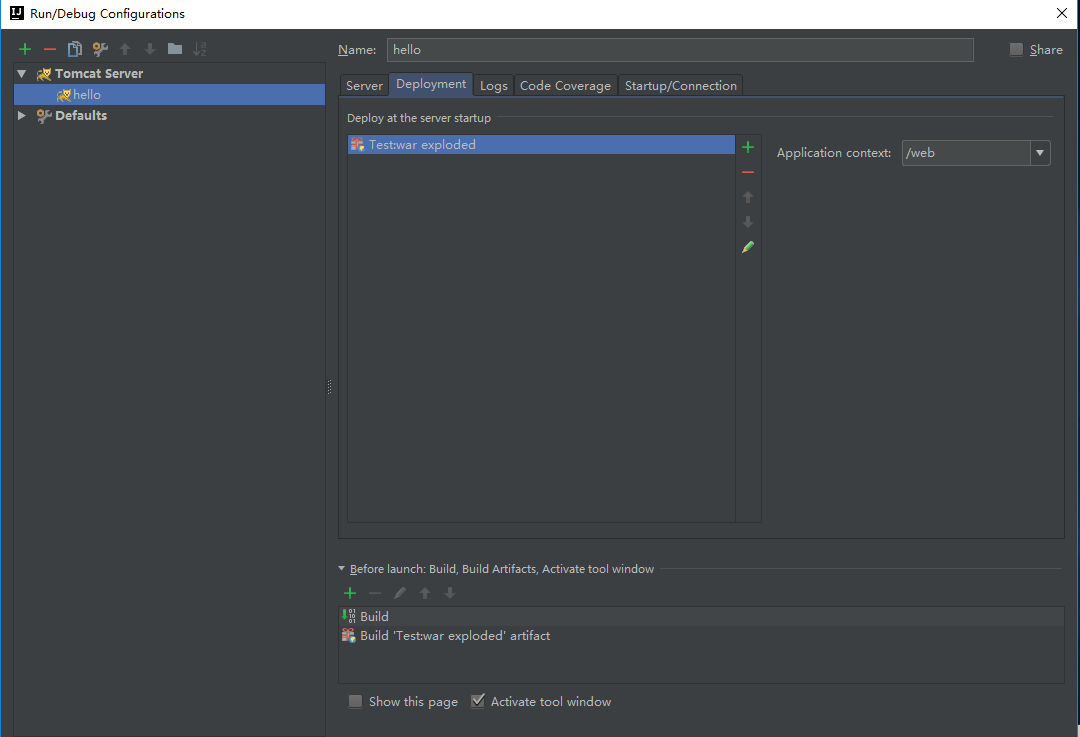


3)给Test项目添加lib库,选择WEB-INF下的lib目录

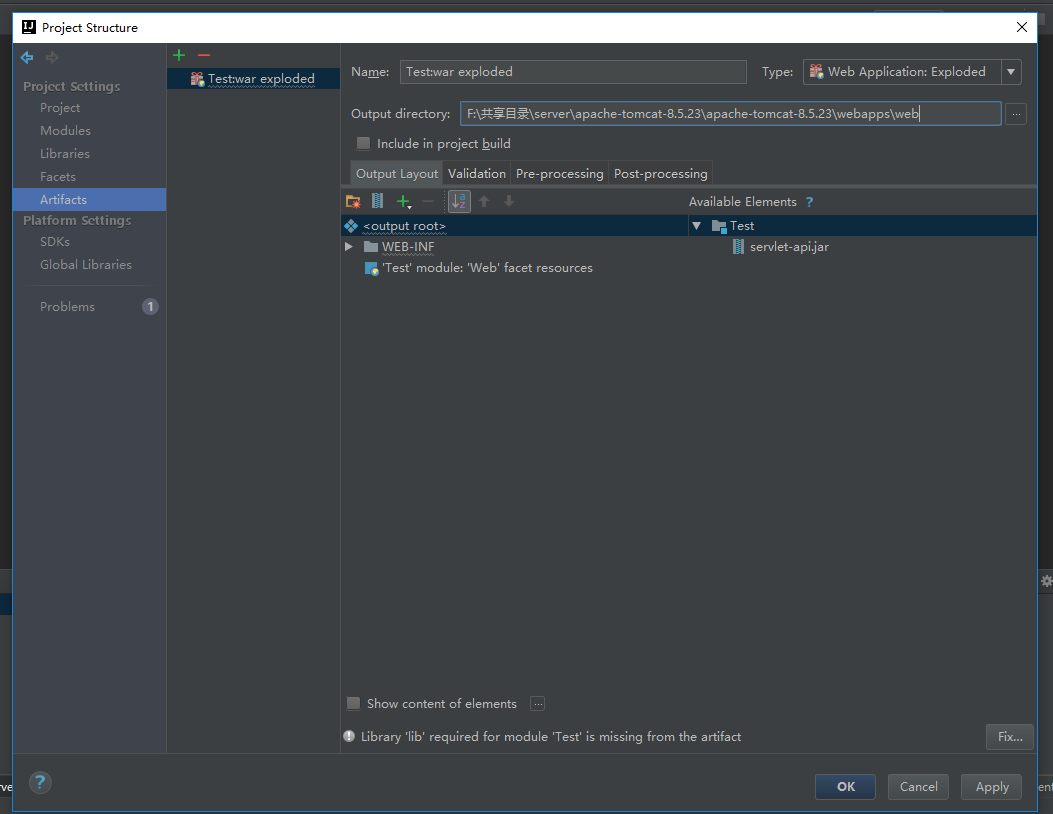


4)新建一个tomcat容器,注意这里的web和webapps目录下的网站名一样



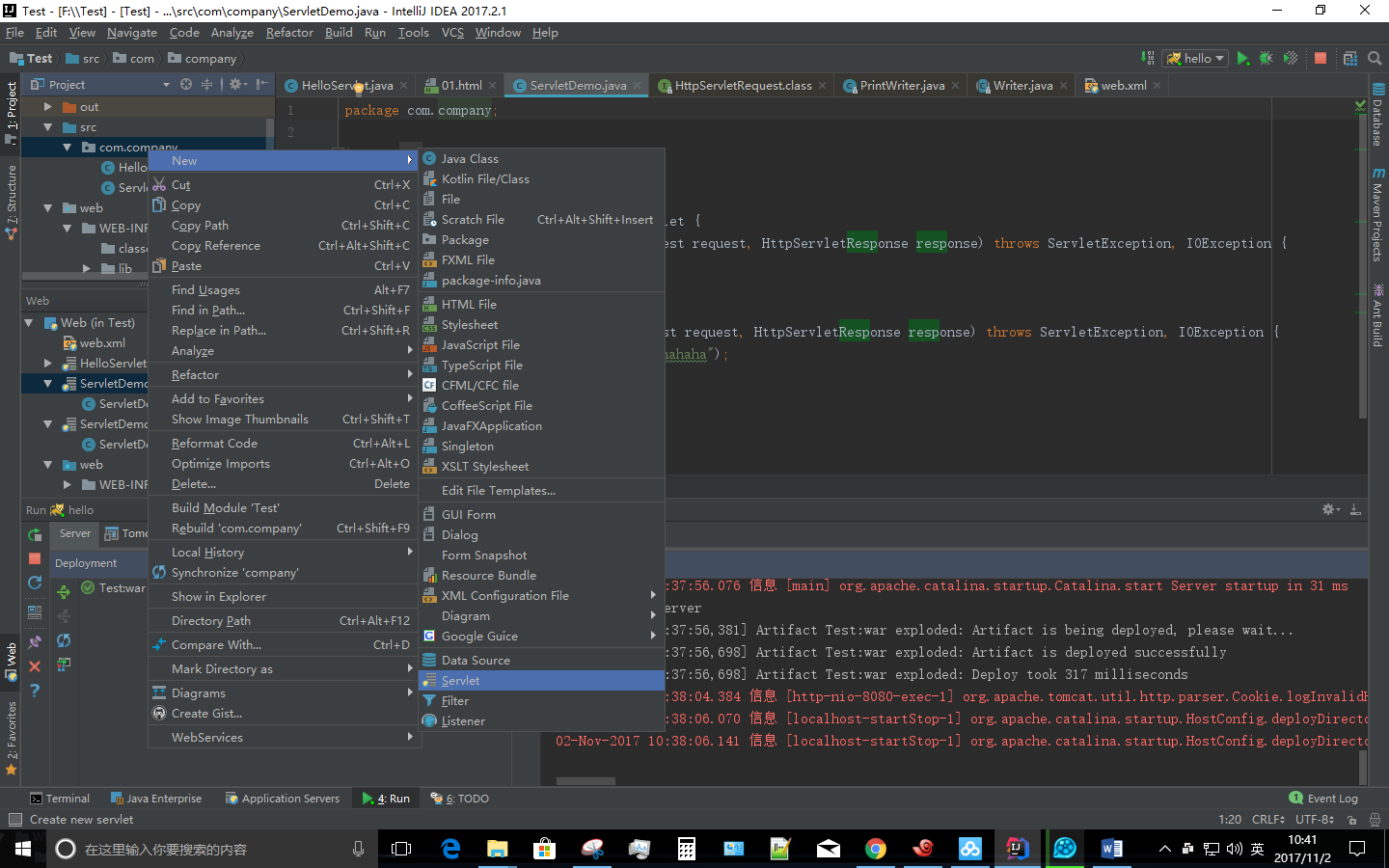


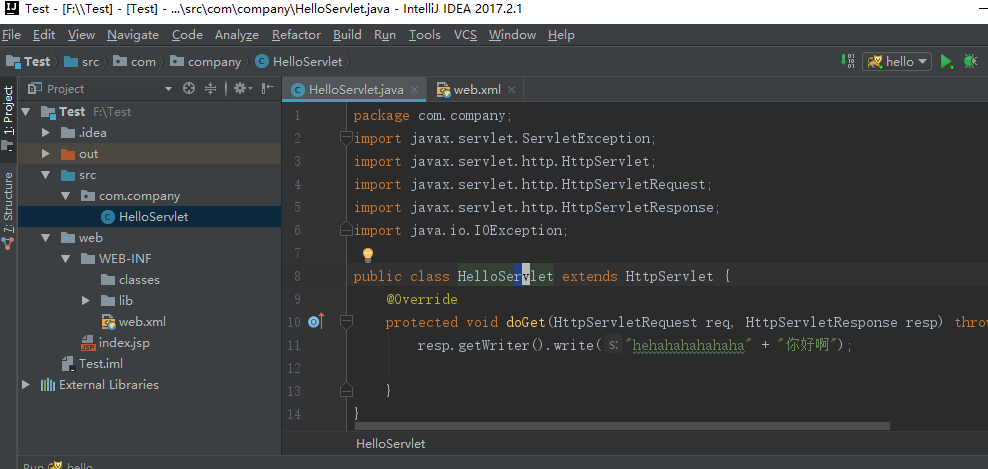
5)将网站部署到tomcat容器里,主要就是将WEB-INF目录复制到webapps目录下

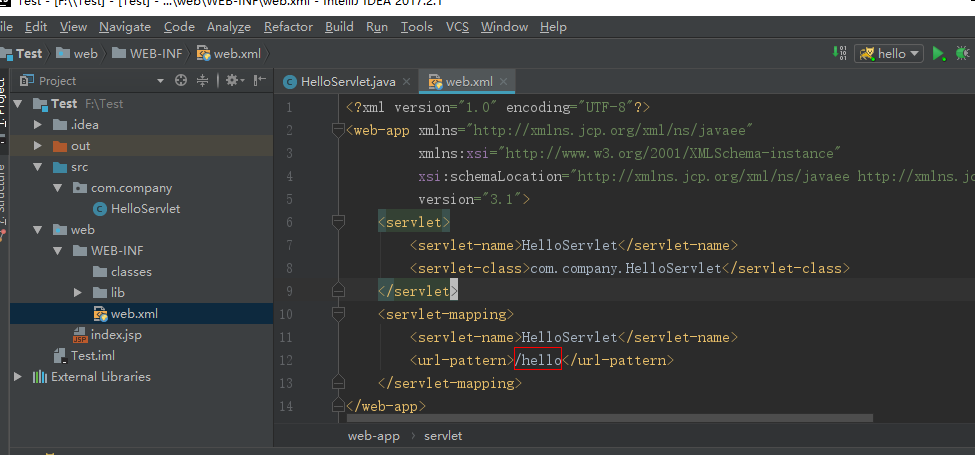


6)新建一个HelloServlet.java文件在src下,再编写WEB-INF下的web.xml文件

注意新建的方式:







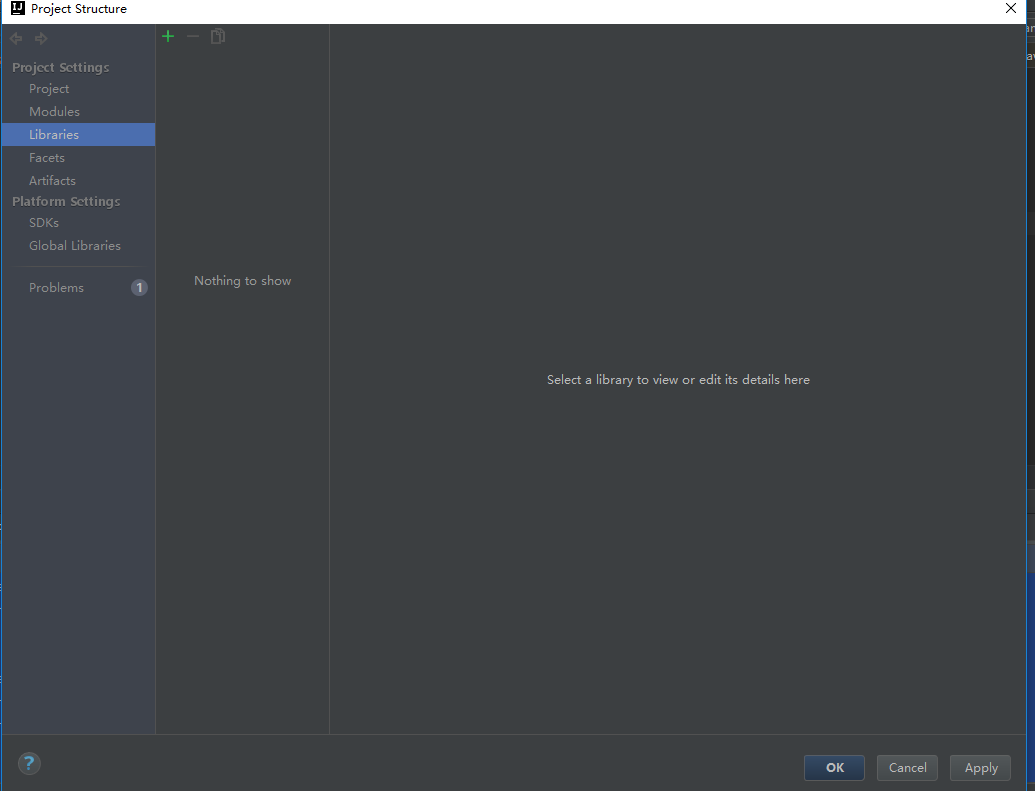
7)直接运行,此时会将WEB-INF目录复制到webapps目录下的web目录里,通过浏览器

<http://localhost:8080/web/hello>就能访问到

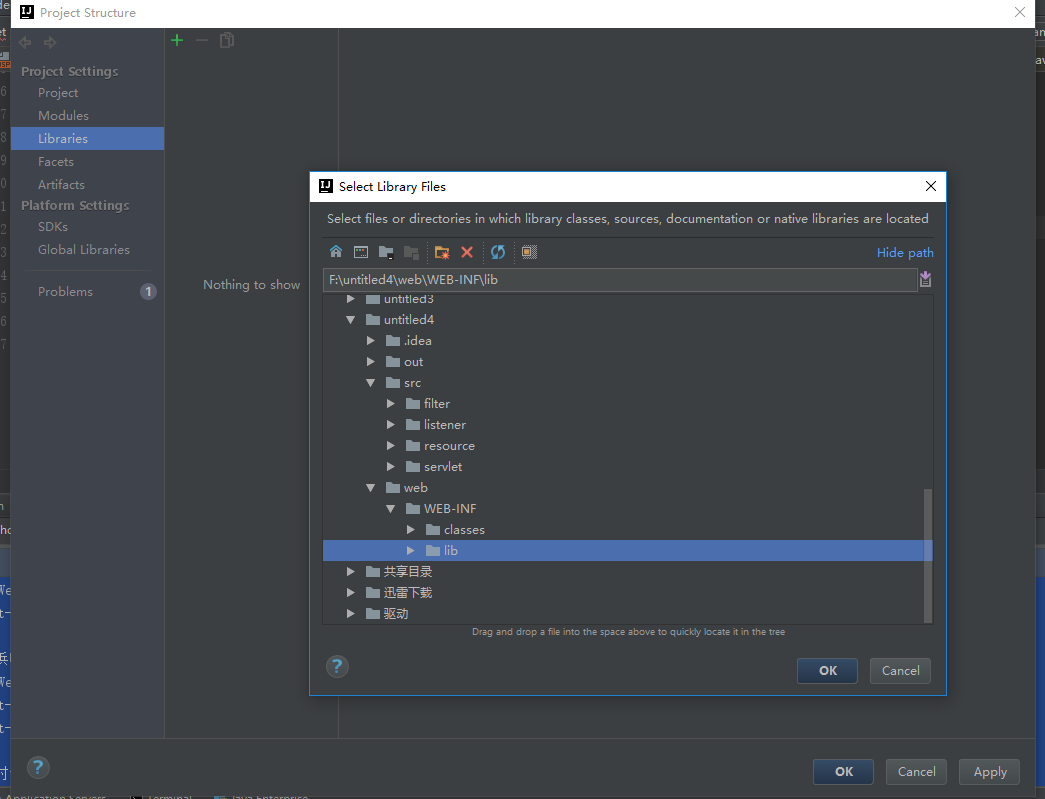
# 3.idea源码和jar包关联

这样就能跳转到源码了

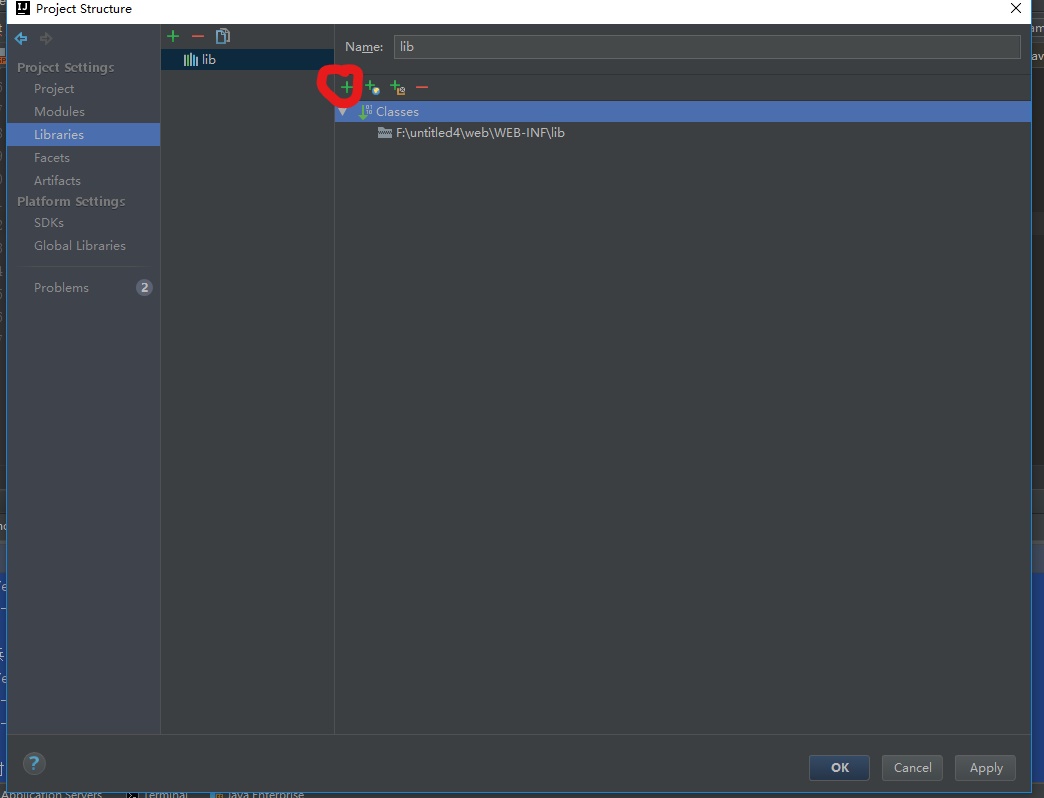
1)首先选择Libraries



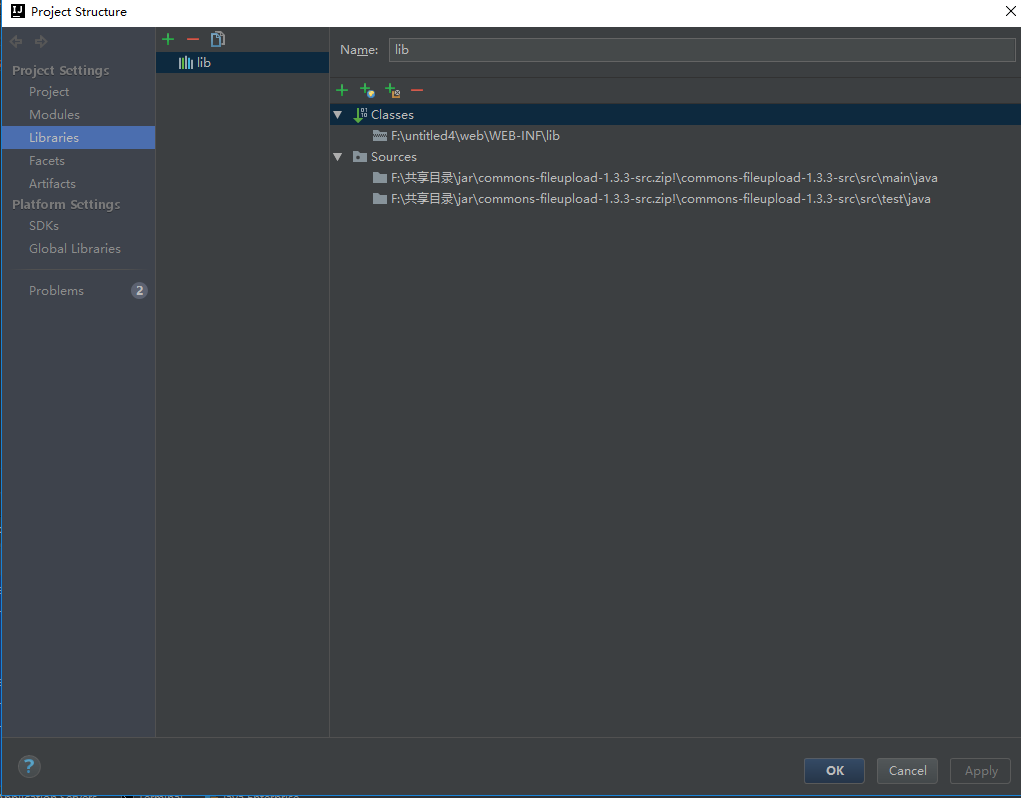
2)点击+,弹出个框框,选择Java,再选择你存放Jar的目录即lib,我们的目的是要将源码关联到这个lib库,这样我们使用lib库里的类时才能跳转到对应的源码



3)然后点击红色圆圈标注的”+”号,选择lib库里的jar对应的源代码文件,这个文件可以为zip压缩文件,目录也行,然后一路确定就行了



4)最后如下图

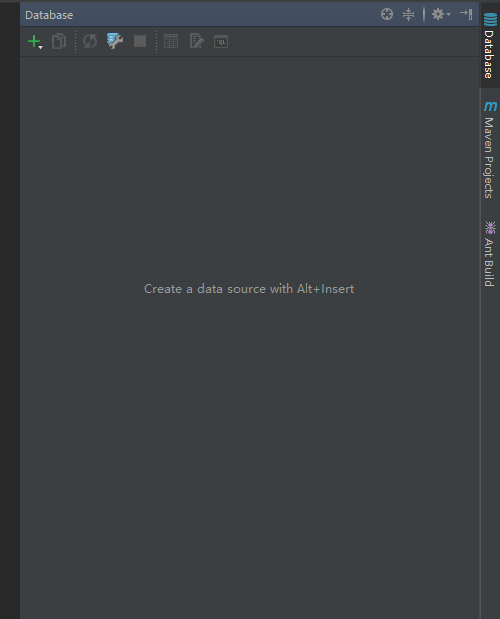


# 4.idea数据库面板

需要下载个插件:Database Tools and SQL

点击View->Tool Windows->Database

出现下面的界面:

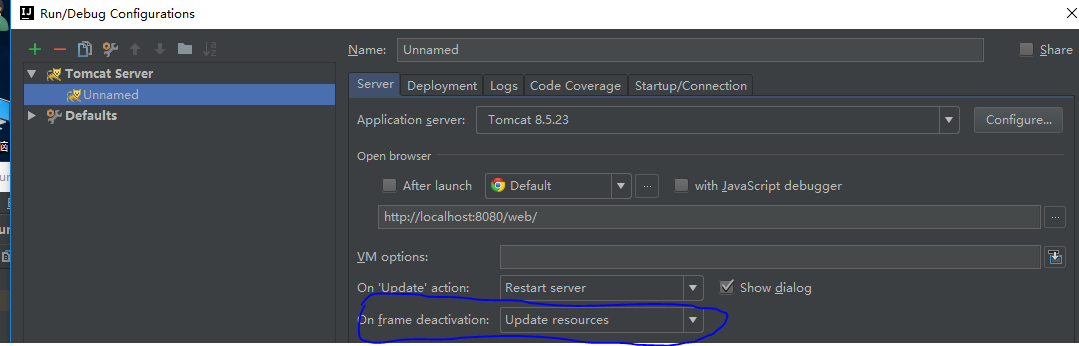


点击”+”号,选择Data Source,再选择Mysql,再添加相关信息就可以了.

Ctrl + Enter 能执行当前行的sql

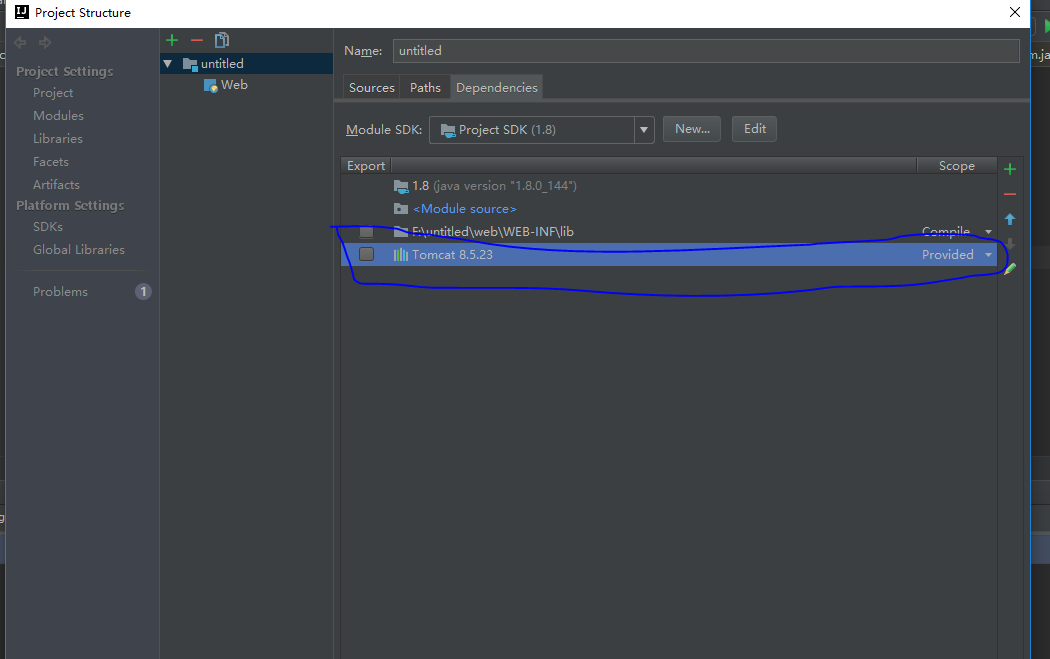
# 5.idea对jsp补全设置和刷新

## 5.1自动刷新



## 5.2jsp补全设置

Dependencies->library->tomcat



# 6.idea修改项目名称

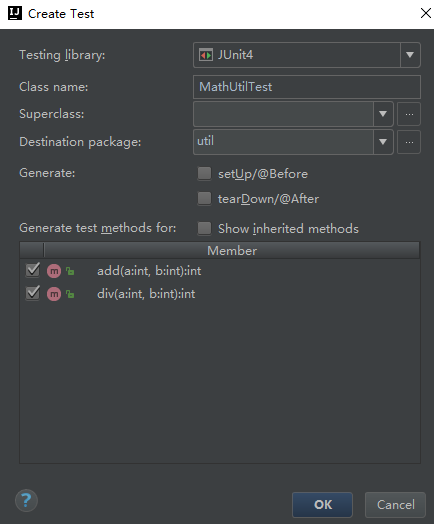
例如要将Demo项目改为Demo2

1. 右击左侧的项目名称->refactor->rename 修改名称为Demo2
2. 将项目下的的.idea 目录下的文件中包含有Demo的字符串全部替换为Demo2
3. 关闭项目,修改目录名称为Demo2

# 7.idea生成测试类

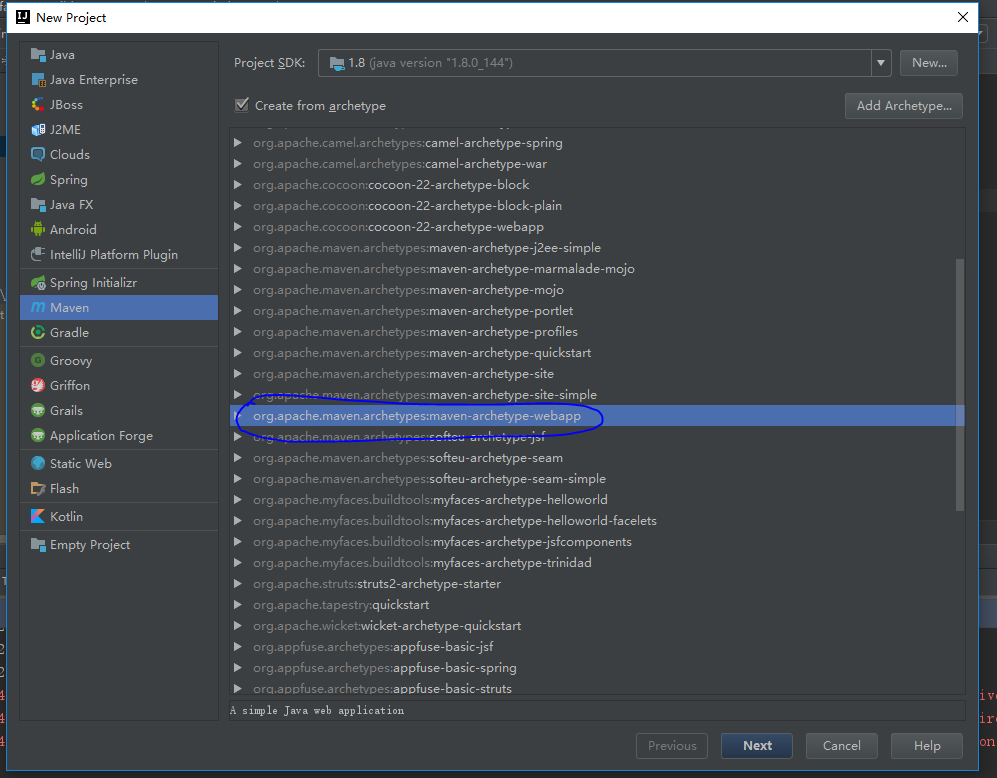
有两种方法:

1. 直接Ctrl + shift + t
2. 光标移动到要测试的类的类名称上,alt + enter->Create Test

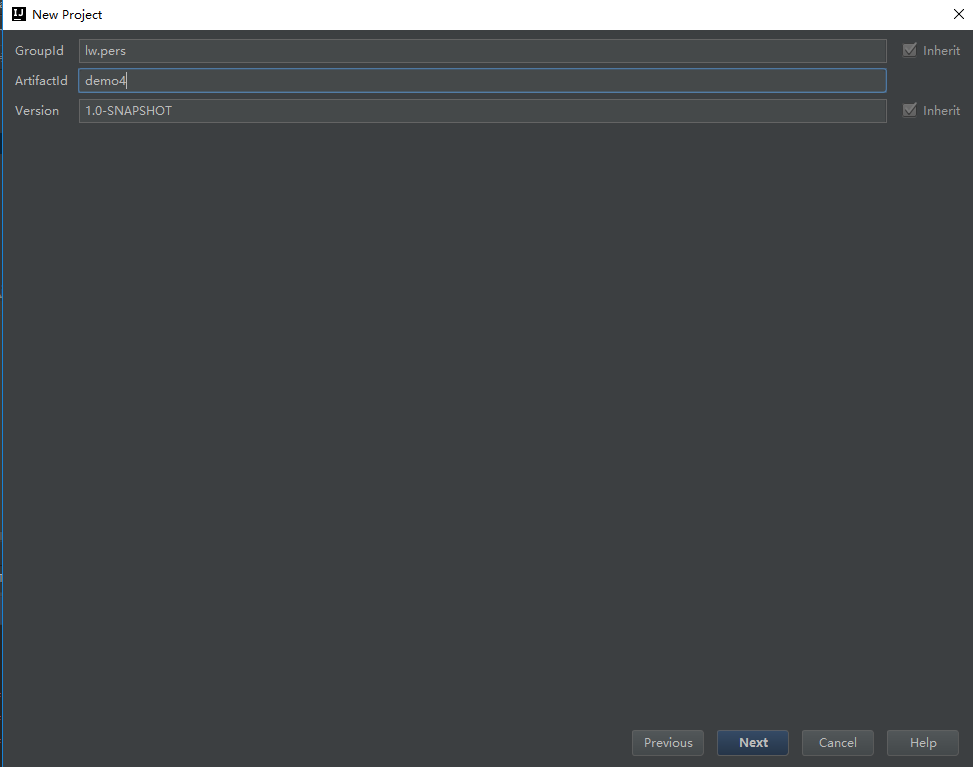


# 8.idea maven创建web项目

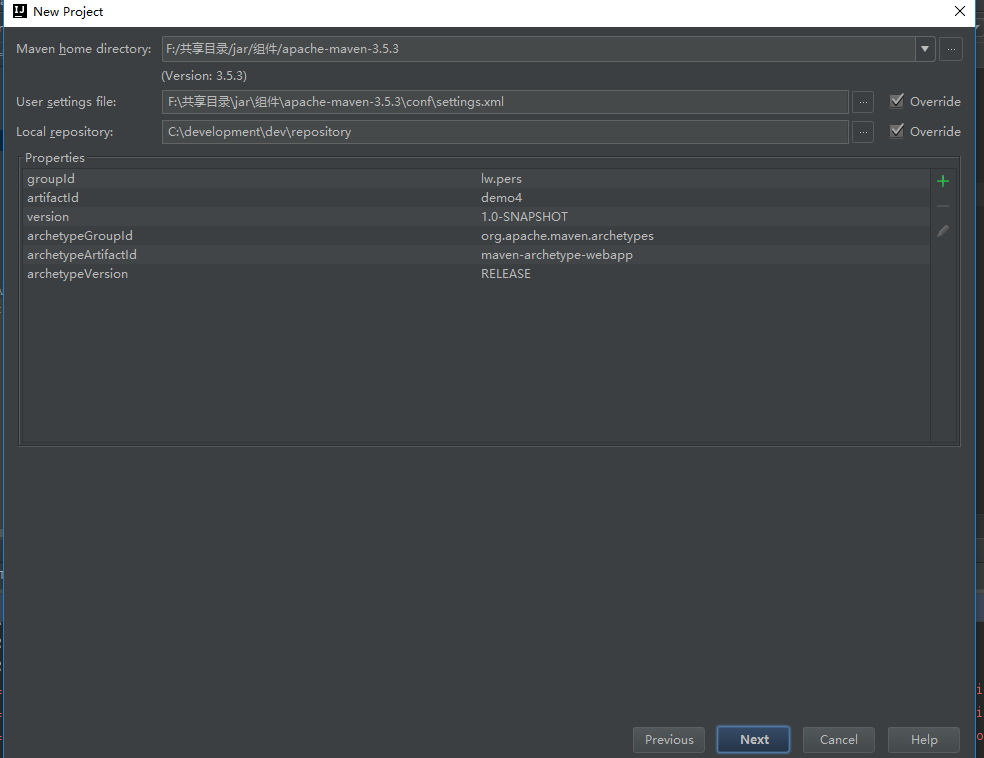
1.选择webapp



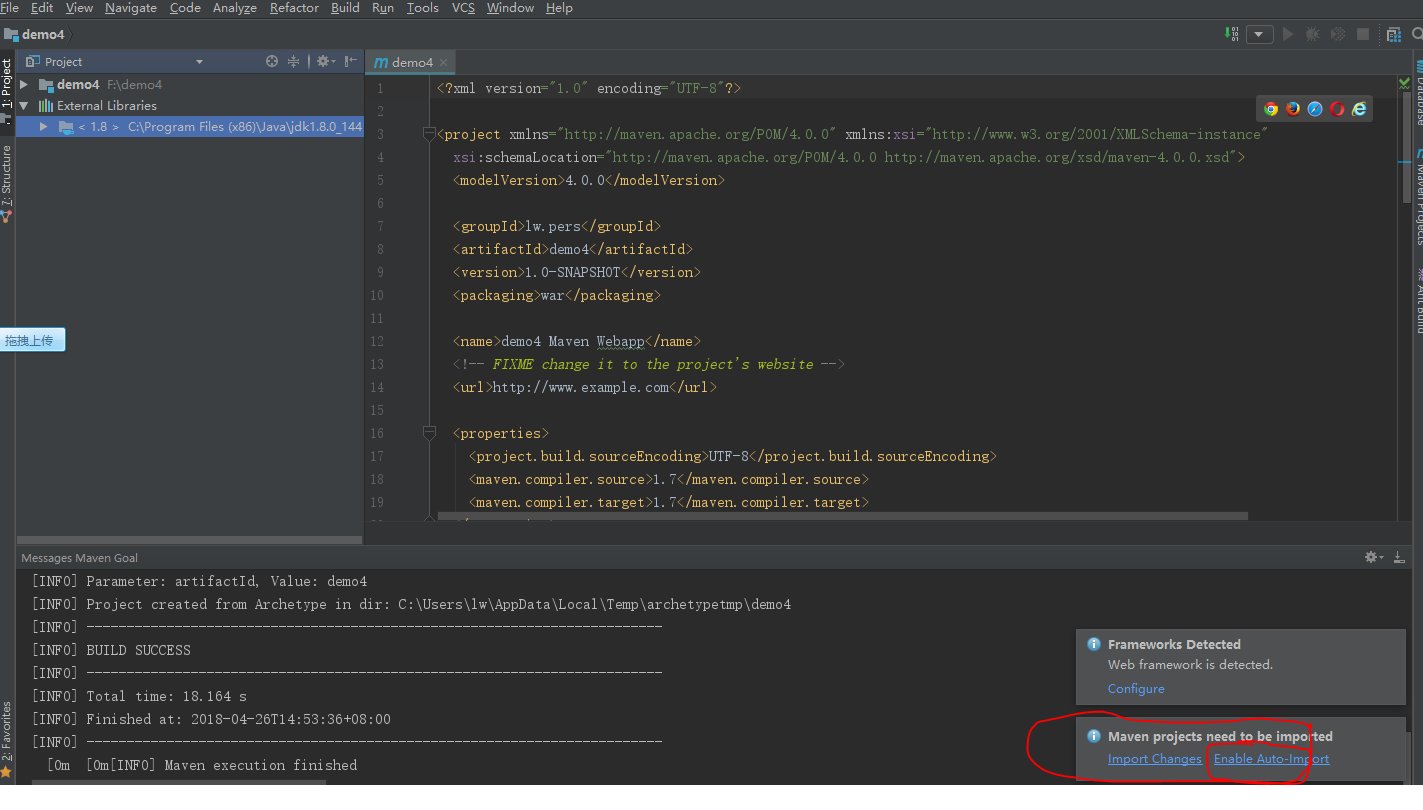
2.坐标



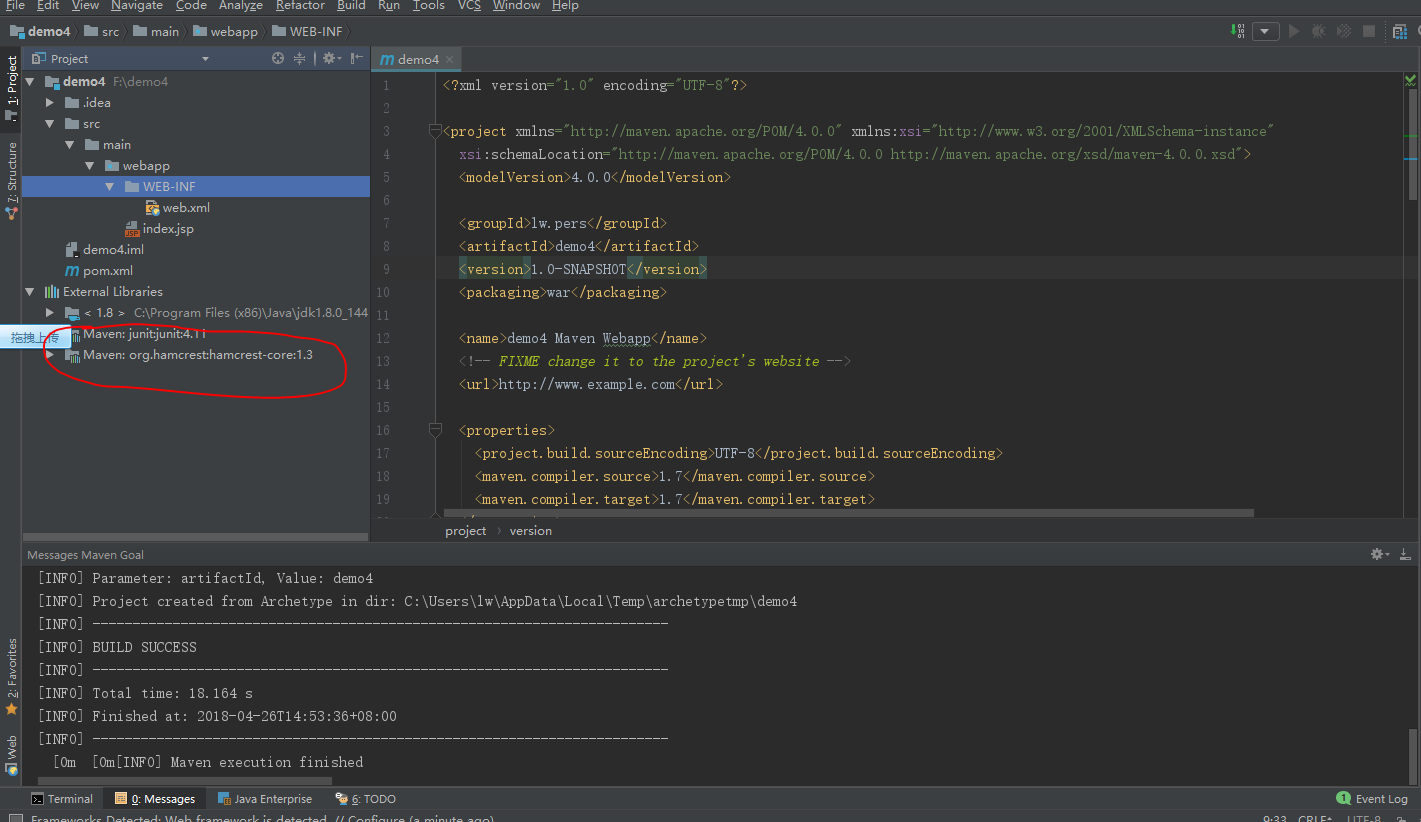
3.maven配置



4.点击右下角的Enable,会自动导入两个包

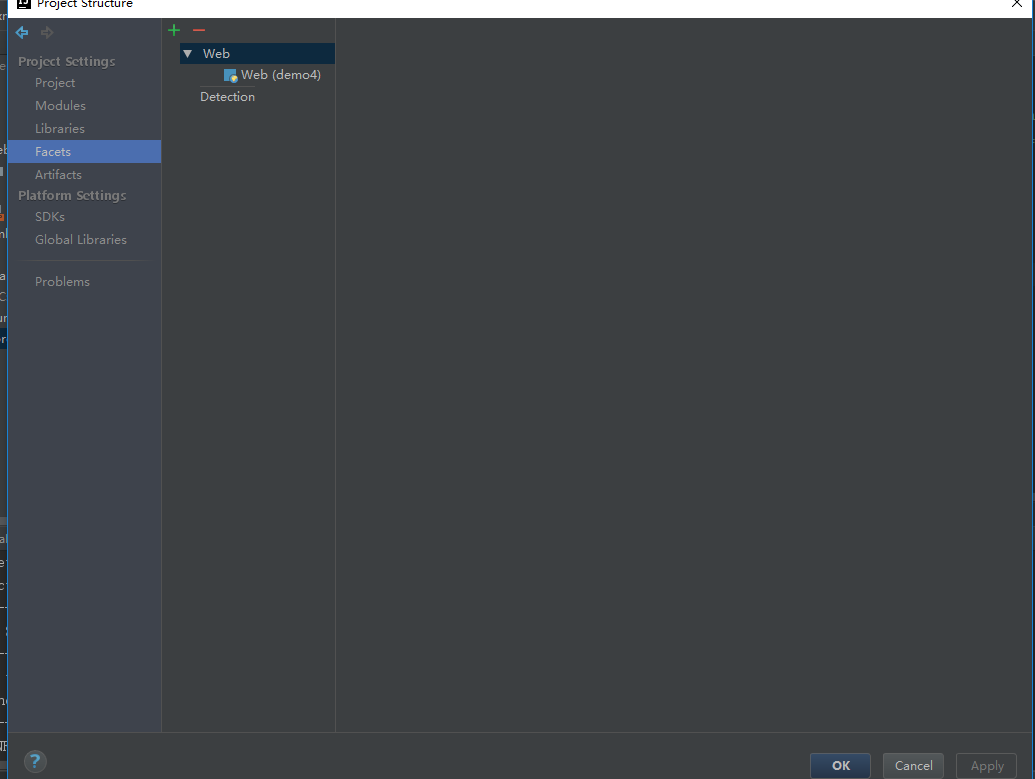


enable后:



5.查看项目的Facets选择web项目,这里由于第4步选择enable自动导入,这时,这里idea

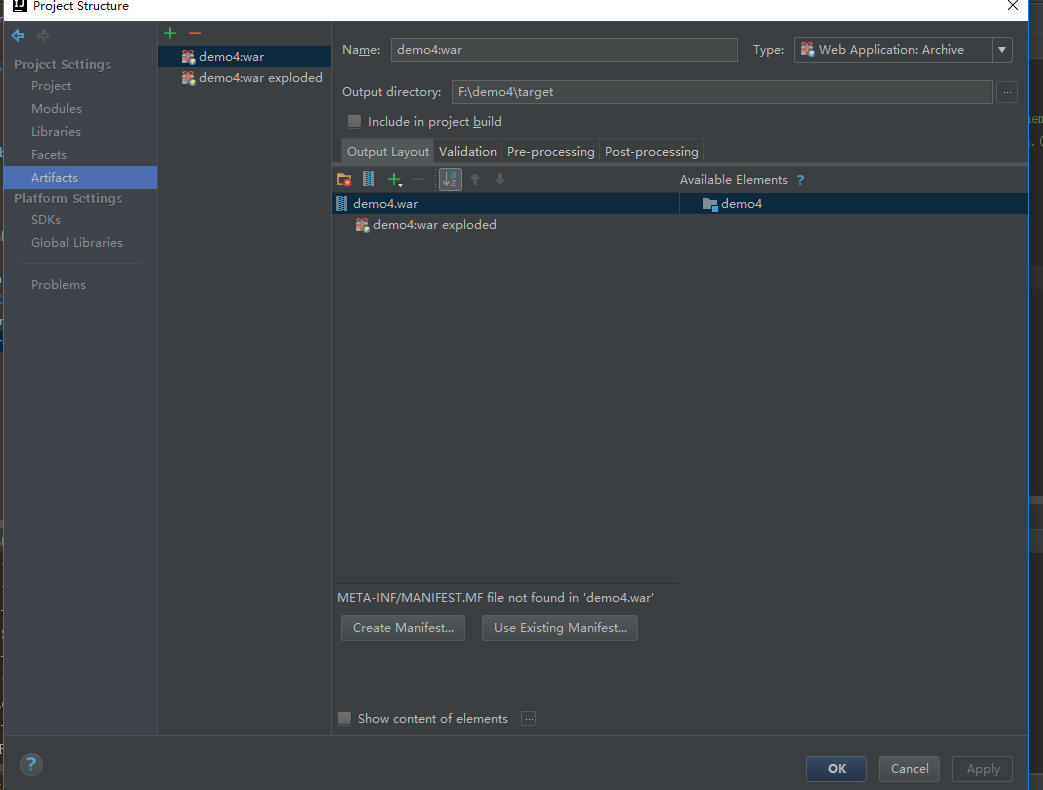
已经自动选择为web项目了.



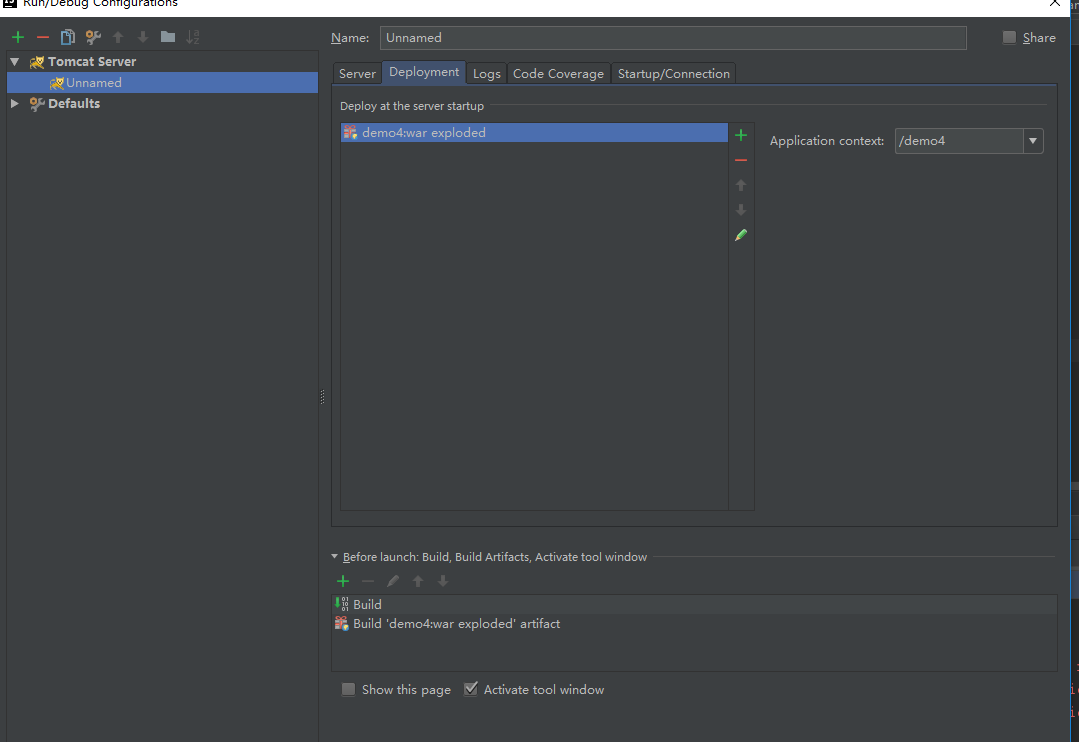
6.查看项目的Artifacts,当然由于第4步选择enable,则系统已经创建了两个包

demo4:war 类型: Web Application:Archive

demo4:war exploded 类型: Web Application:Exploded



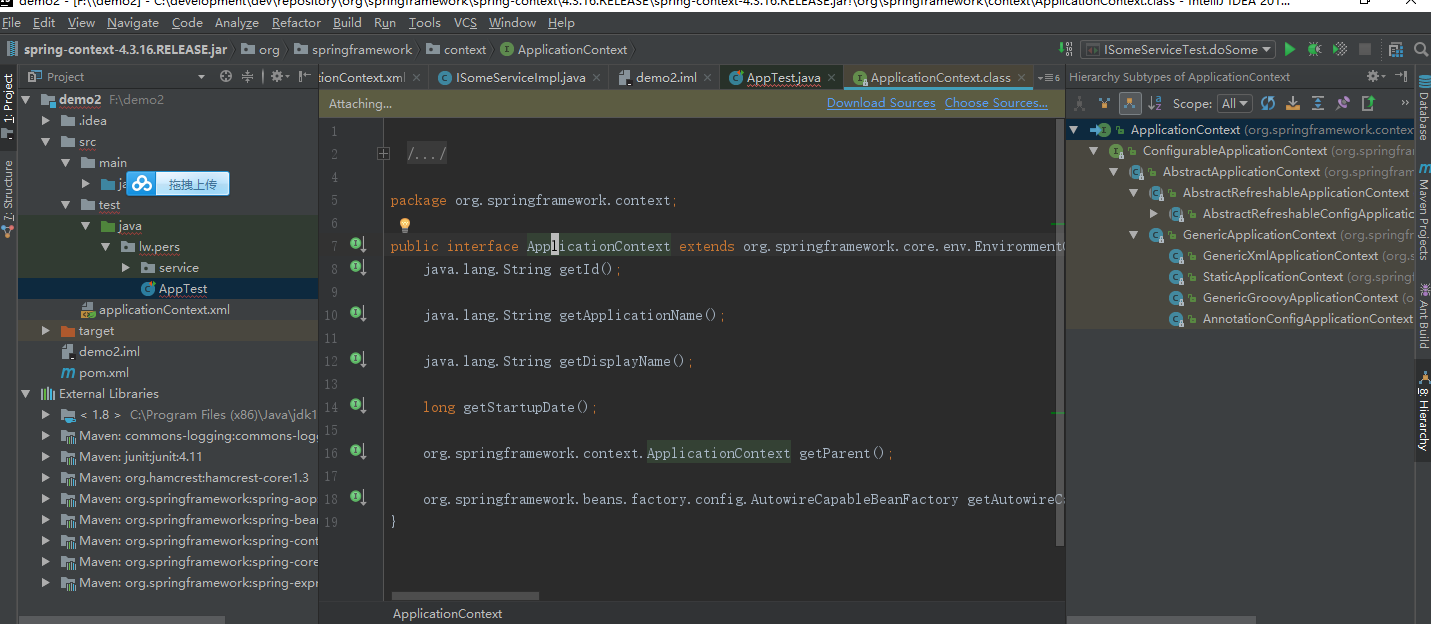
7.配置tomcat 选择demo4:war exploded



9.然后run

访问:localhost:8080/demo4

# 9.查看看接口的实现类的关系图



光标移动到ApplicationContext上,点击右上角Navigate ->Type Hierarchy