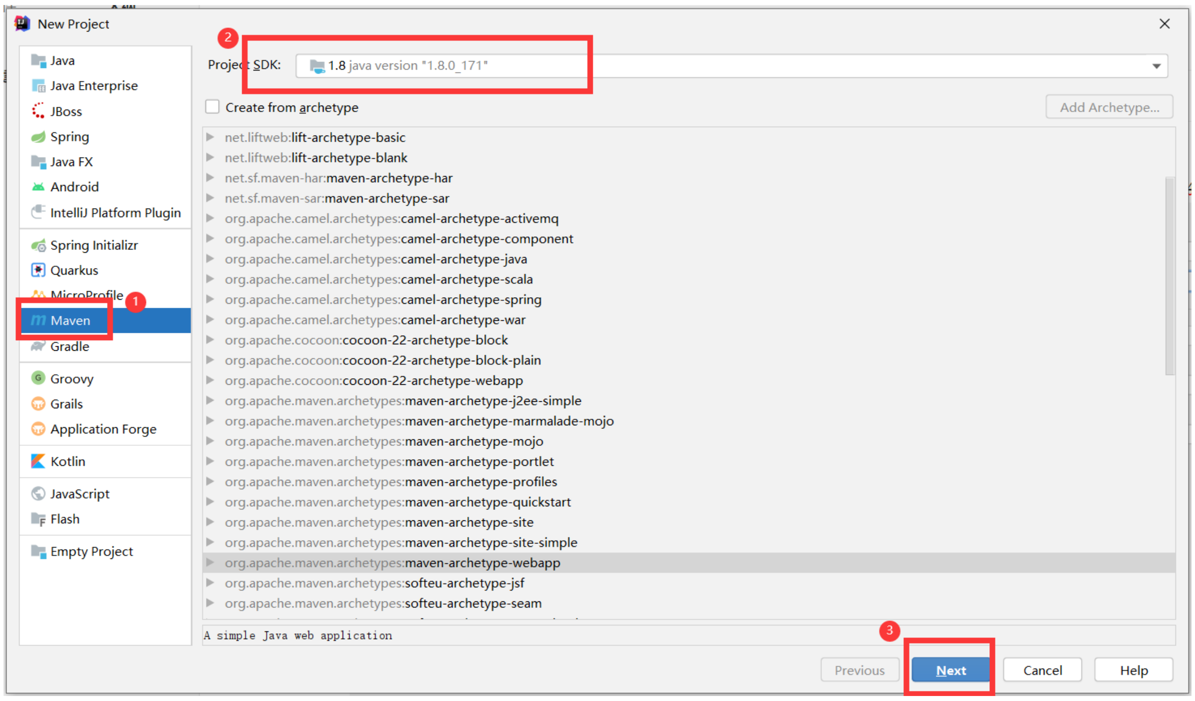
小米商城

一、新建项目

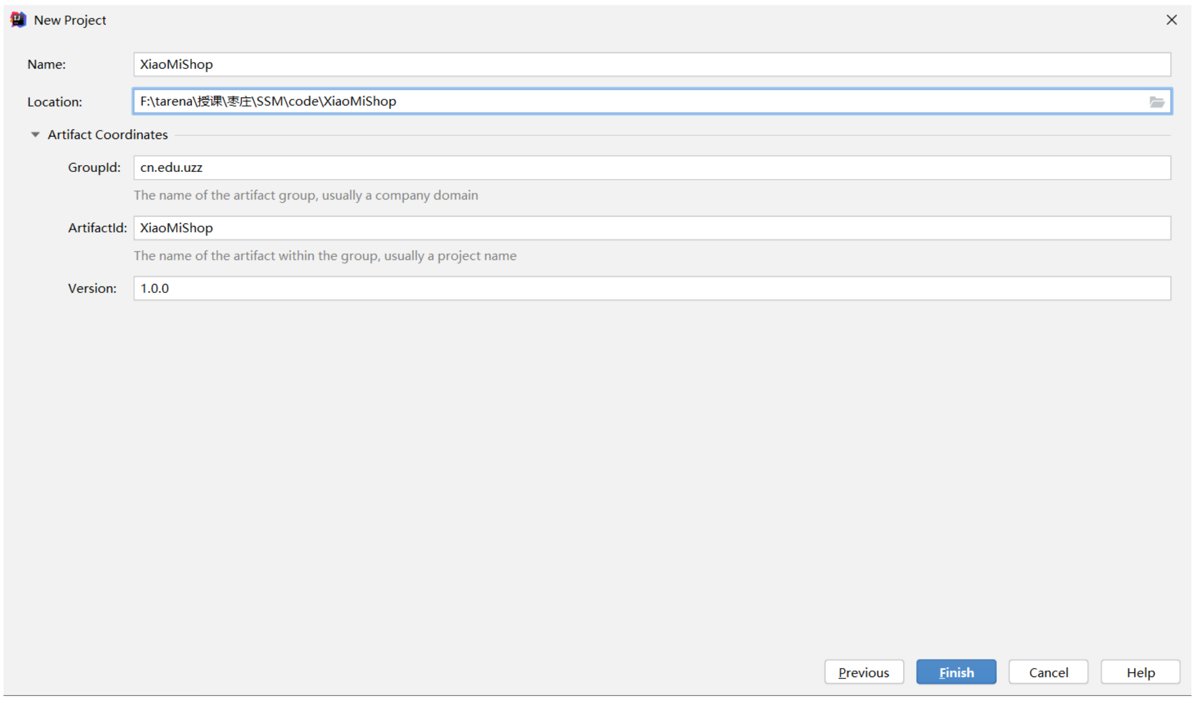


Project SDK: 1.8 java version "1.8.0_171"

Create from archetype

- net.liftweb:lift-archetype-basic
- net.liftweb:lift-archetype-blank
- net.sf.maven-har:maven-archetype-har
- net.sf.maven-sar:maven-archetype-sar
- org.apache.camel.archetypes:camel-archetype-activemq
- org.apache.camel.archetypes:camel-archetype-component
- org.apache.camel.archetypes:camel-archetype-java
- org.apache.camel.archetypes:camel-archetype-scala
- org.apache.camel.archetypes:camel-archetype-spring
- org.apache.camel.archetypes:camel-archetype-war
- org.apache.cocoon:cocoon-22-archetype-block
- org.apache.cocoon:cocoon-22-archetype-block-plain
- org.apache.cocoon:cocoon-22-archetype-webapp
- org.apache.maven.archetypes:maven-archetype-j2ee-simple
- org.apache.maven.archetypes:maven-archetype-marmalade-mojo
- org.apache.maven.archetypes:maven-archetype-mojo
- org.apache.maven.archetypes:maven-archetype-portlet
- org.apache.maven.archetypes:maven-archetype-profiles
- org.apache.maven.archetypes:maven-archetype-quickstart
- org.apache.maven.archetypes:maven-archetype-site
- org.apache.maven.archetypes:maven-archetype-site-simple
- org.apache.maven.archetypes:maven-archetype-webapp
- org.apache.maven.archetypes:softeu-archetype-jsf
- org.apache.maven.archetypes:softeu-archetype-seam

```
A simple Java web application
```



Name: XiaoMiShop

Location: F:\tarena\授课\枣庄\SSM\code\XiaoMiShop

GroupId: cn.edu.uzz

The name of the artifact group, usually a company domain

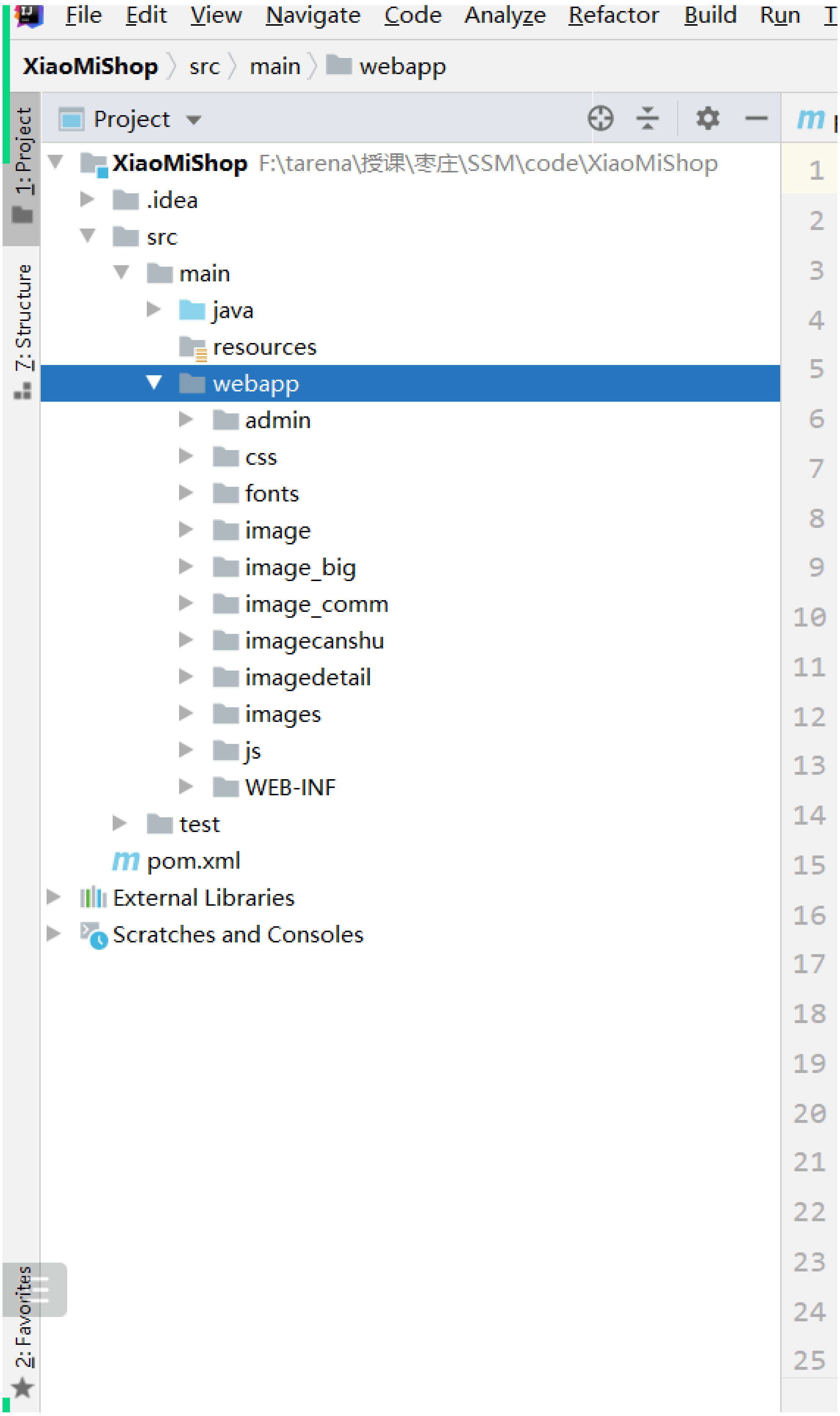
ArtifactId: XiaoMiShop

The name of the artifact within the group, usually a project name

Version: 1.0.0

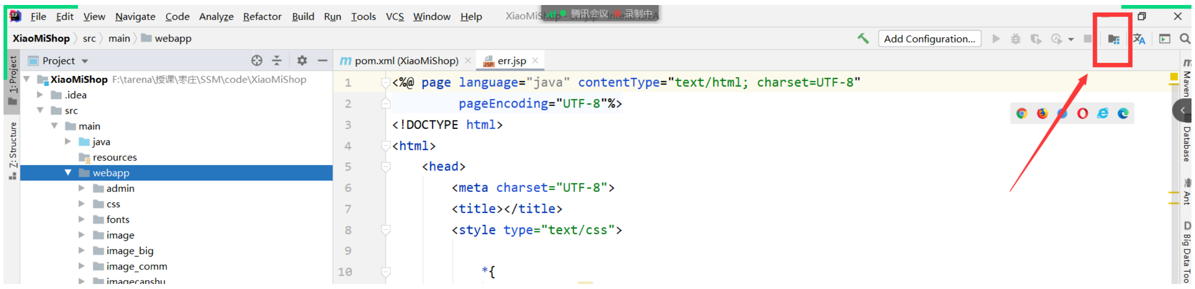
二、修改项目目录

创建webapp目录并导入页面资源

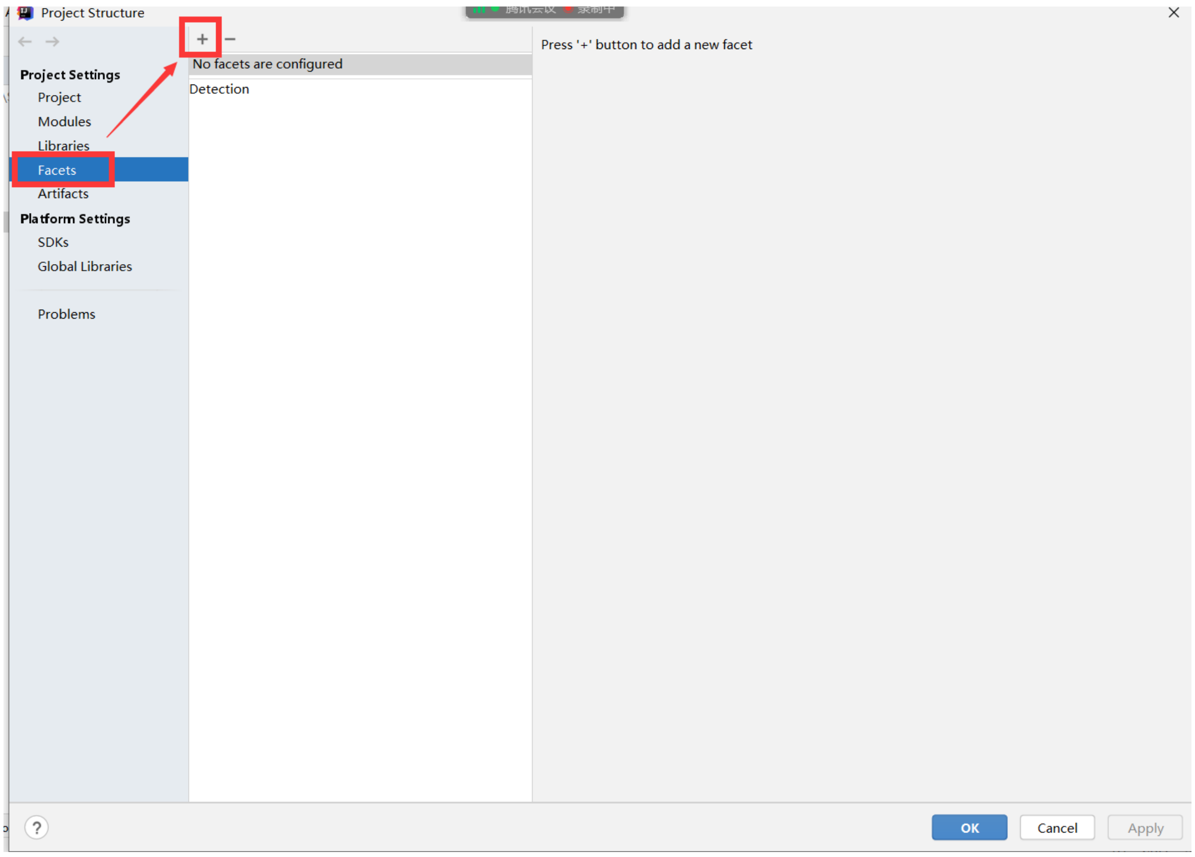

File Edit View Navigate Code Analyze Refactor Build Run
XiaoMiShop src main webapp
Project
1: Project
7: Structure
2: Favorites

```
XiaoMiShop F:\tarena\授课\枣庄\SSM\code\XiaoMiShop
  .idea
  src
    main
      java
      resources
      webapp
        admin
        css
        fonts
        image
        image_big
        image_comm
        imagecanshu
        imagedetail
        images
        js
        WEB-INF
    test
  pom.xml
External Libraries
Scratches and Consoles
```

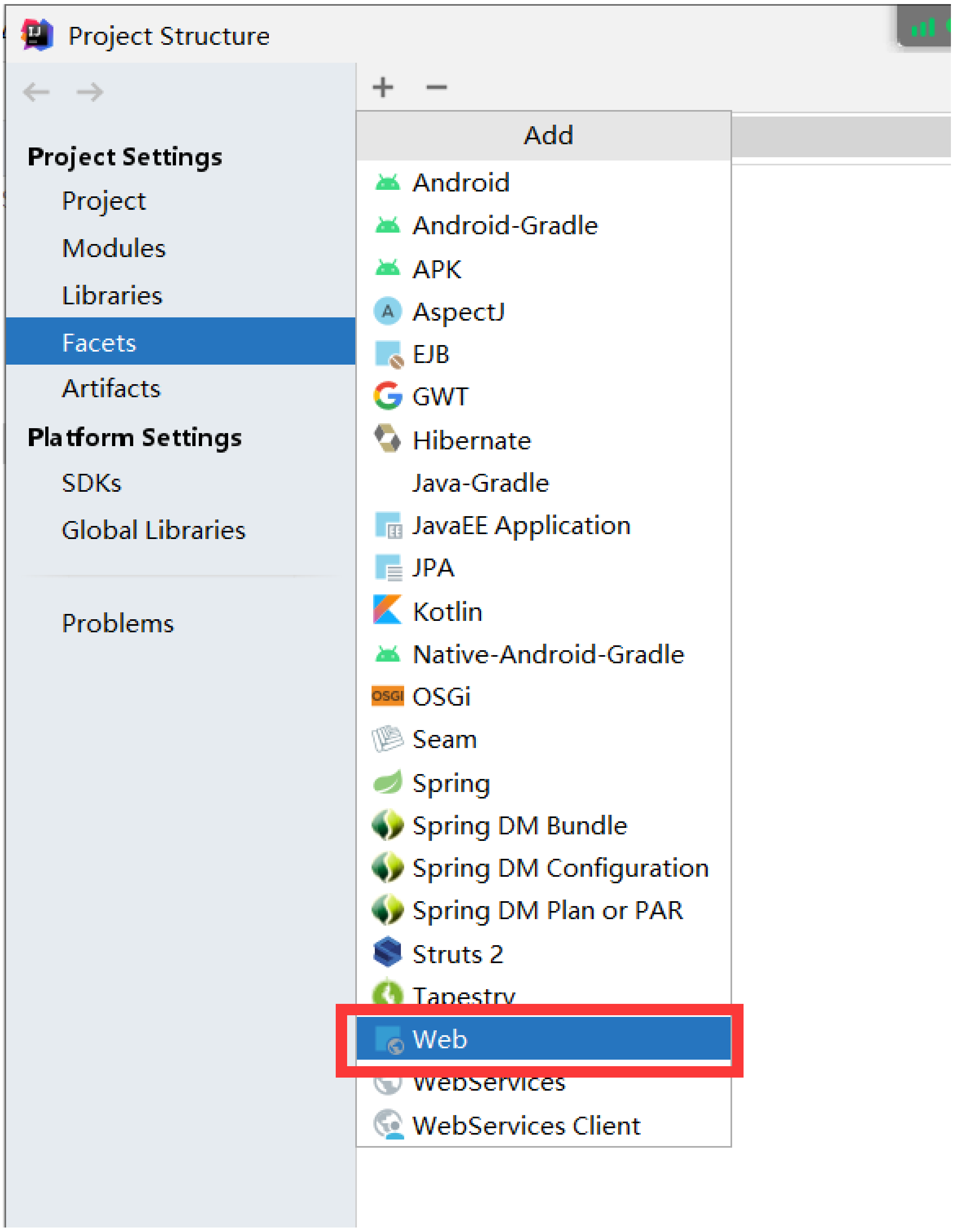
找到右上角图标

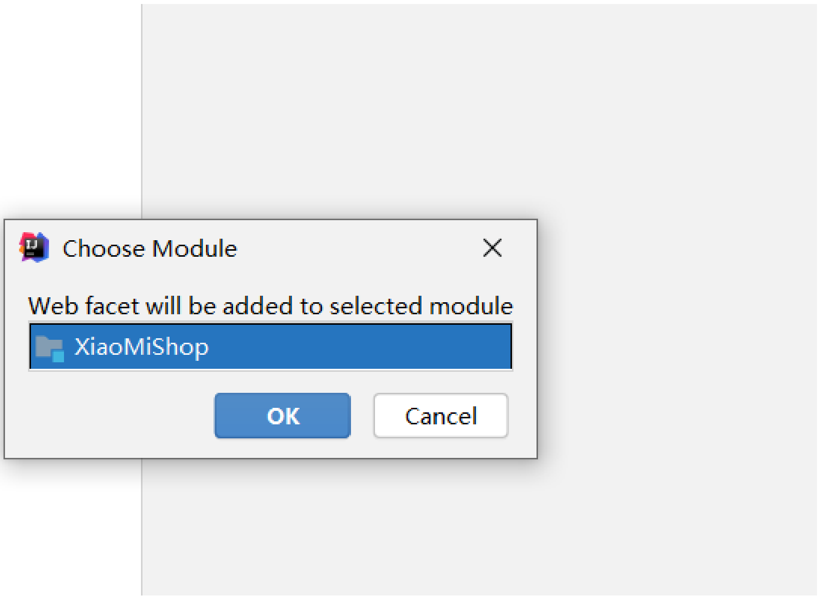

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
XiaoMiShop 〉 src 〉 main 〉 webapp
Add Configuration...
Project
pom.xml (XiaoMiShop)
err.jsp
XiaoMiShop F:\tarena\授课\枣庄\SSM\code\XiaoMiShop
.idea
src
main
java
resources
webapp
admin
css
fonts
image
image_big
image_comm
imagecanshu

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
        pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title></title>
        <style type="text/css">

            *{
```



Press '+' button to add a new facet

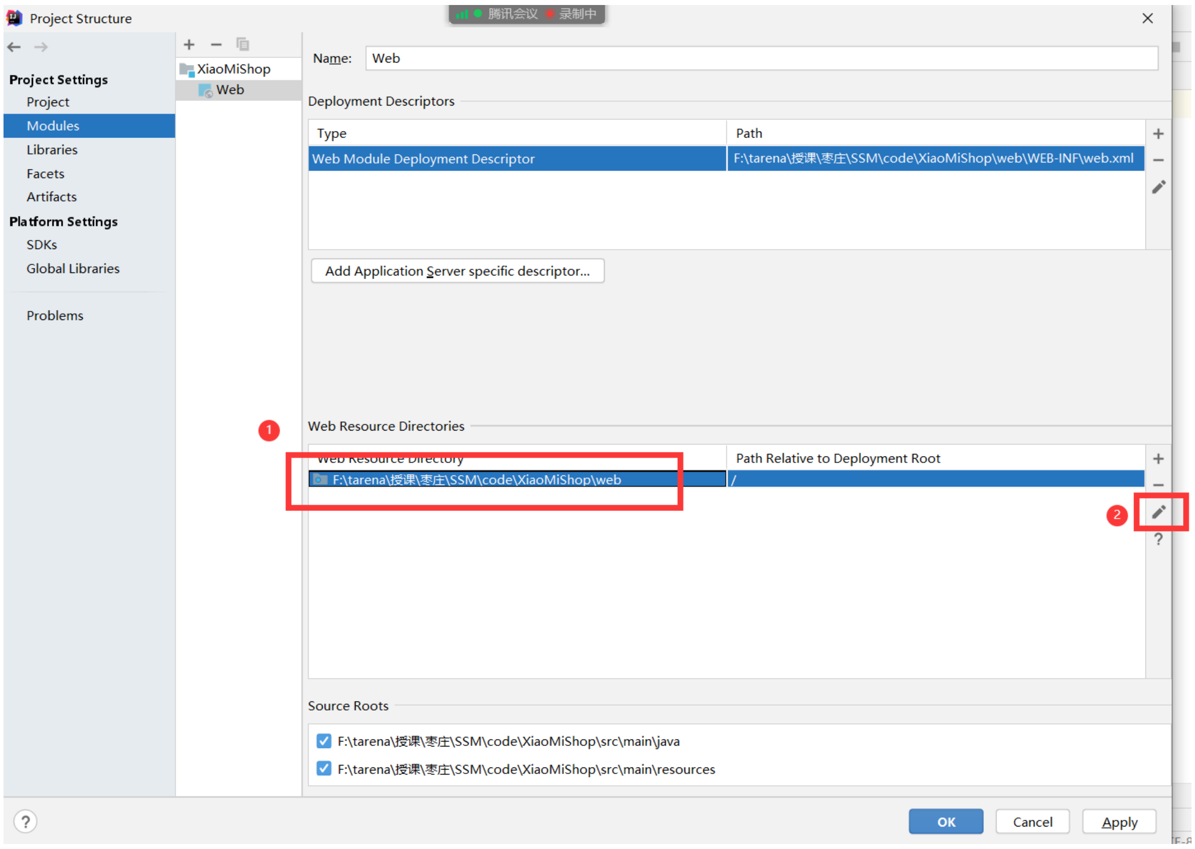

Project Structure
Project Settings
Project
Modules
Libraries
Facets
Artifacts
Platform Settings
SDKs
Global Libraries
Problems
Add
Android
Android-Gradle
APK
AspectJ
EJB
GWT
Hibernate
Java-Gradle
JavaEE Application
JPA
Kotlin
Native-Android-Gradle
OSGi
Seam
Spring
Spring DM Bundle
Spring DM Configuration
Spring DM Plan or PAR
Struts 2
Tapestry
Web
WebServices
WebServices Client


Choose Module

Web facet will be added to selected module

XiaoMiShop

OK Cancel


Project Structure

Project Settings
- Project
- Modules
- Libraries
- Facets
- Artifacts

Platform Settings
- SDKs
- Global Libraries

Problems

XiaoMiShop
Web

Name: Web

Deployment Descriptors

Type	Path
Web Module Deployment Descriptor	F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\web\WEB-INF\web.xml

Add Application Server specific descriptor...

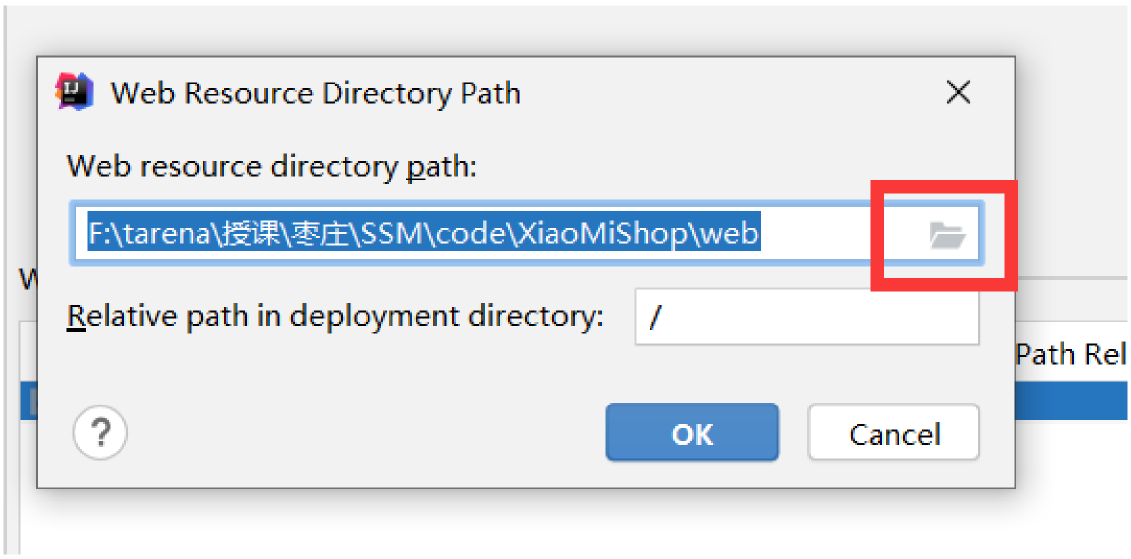
Web Resource Directories

Web Resource Directory	Path Relative to Deployment Root
F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\web	/

Source Roots

☑ F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\java
☑ F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\resources

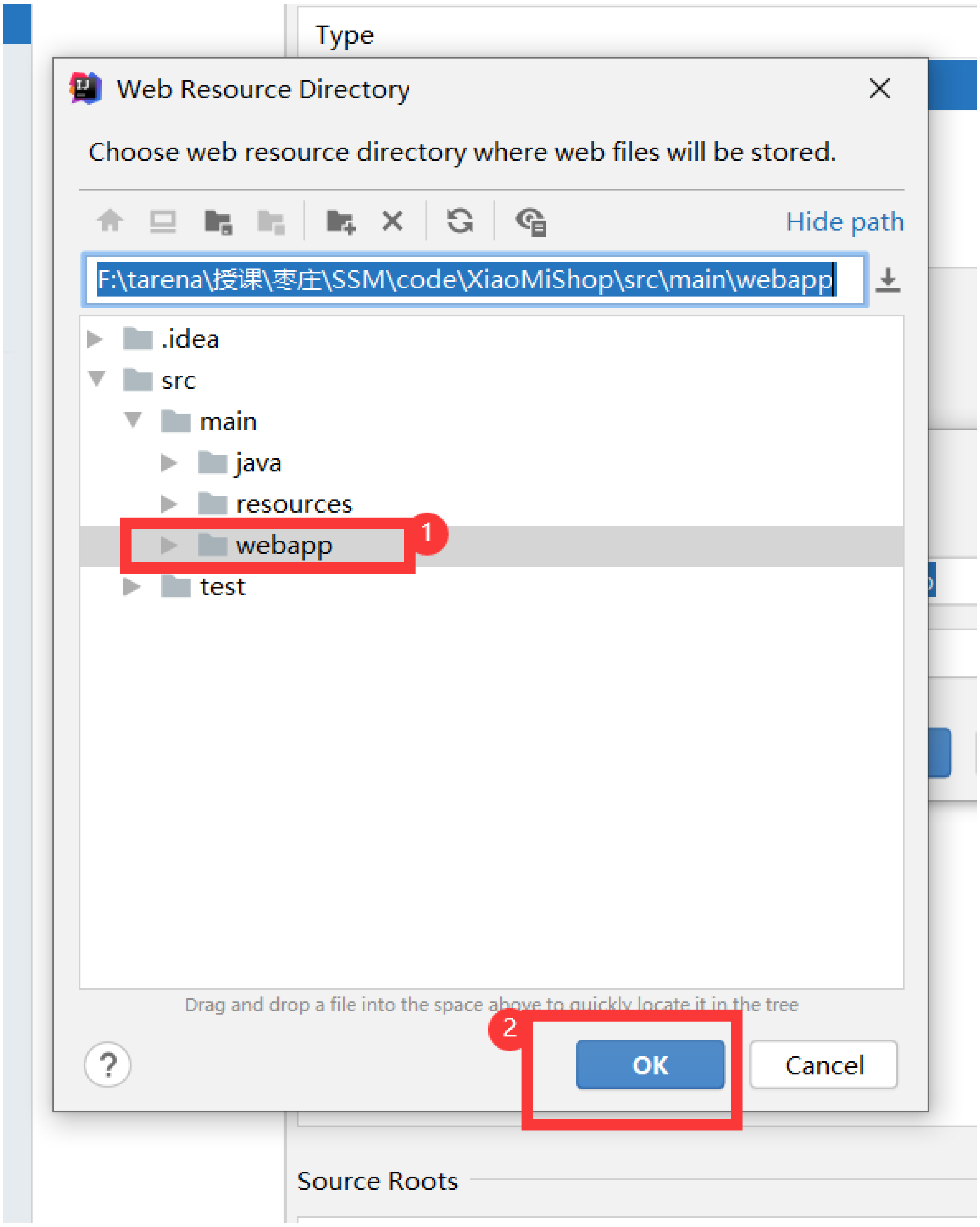
OK Cancel Apply


Web Resource Directory Path

Web resource directory path: F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\web

Relative path in deployment directory: /

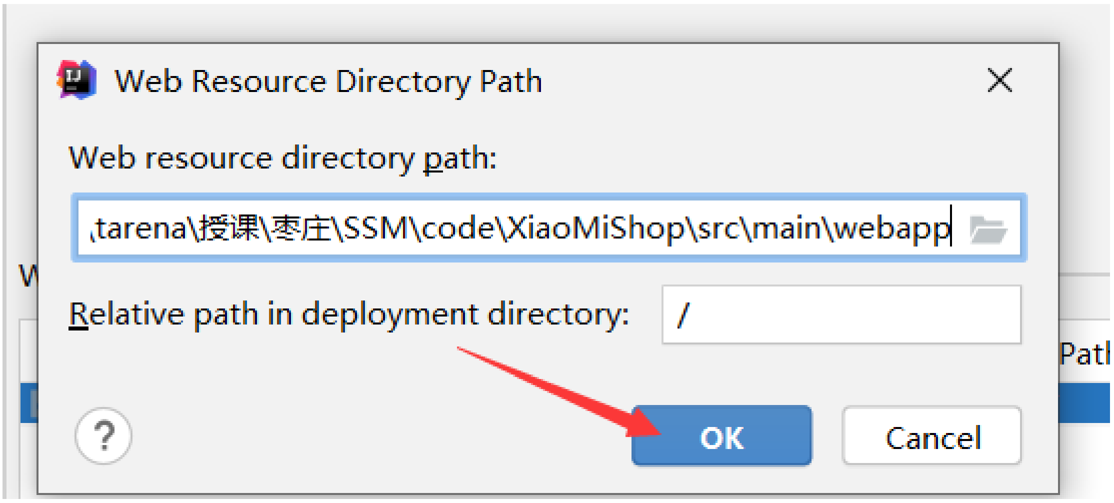
OK Cancel



Choose web resource directory where web files will be stored.

F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\webapp

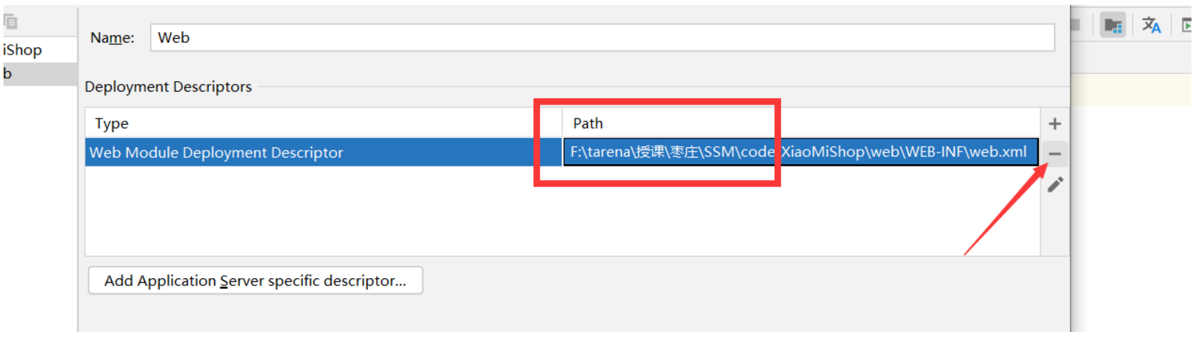
Drag and drop a file into the space above to quickly locate it in the tree



Web resource directory path: tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\webapp

Relative path in deployment directory: /

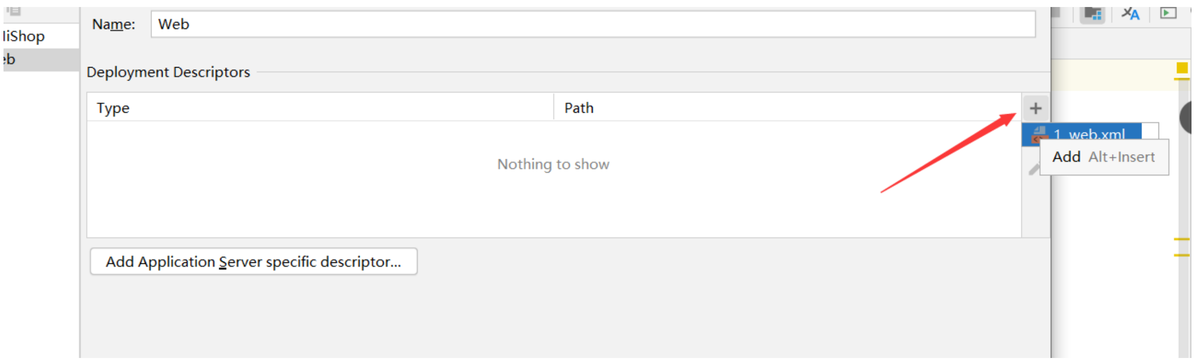
选中方框内的内容，然后点击“-”


iShop
b
Name: Web
Deployment Descriptors

Type	Path
Web Module Deployment Descriptor	F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\web\WEB-INF\web.xml

Add Application Server specific descriptor...

然后点击上面的+，选择web.xml


liShop
ᵢb
Name: Web
Deployment Descriptors

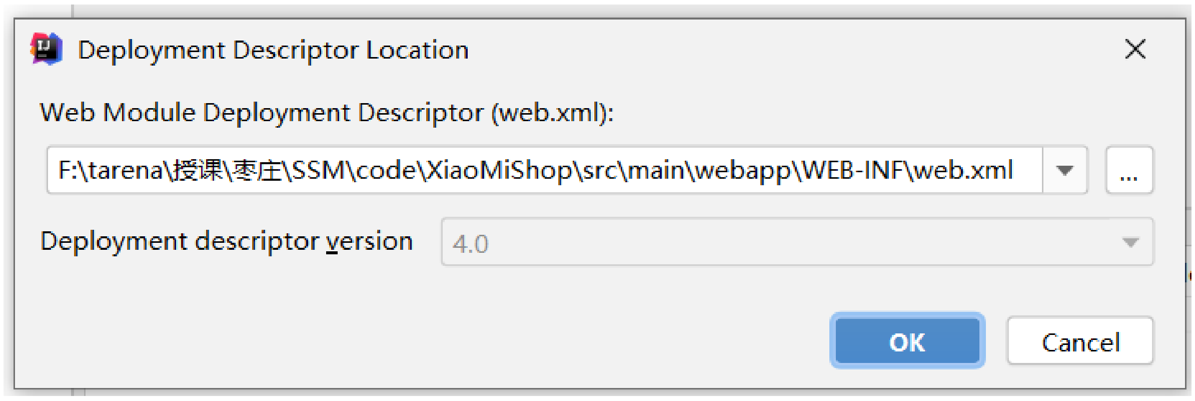
Type	Path

Nothing to show

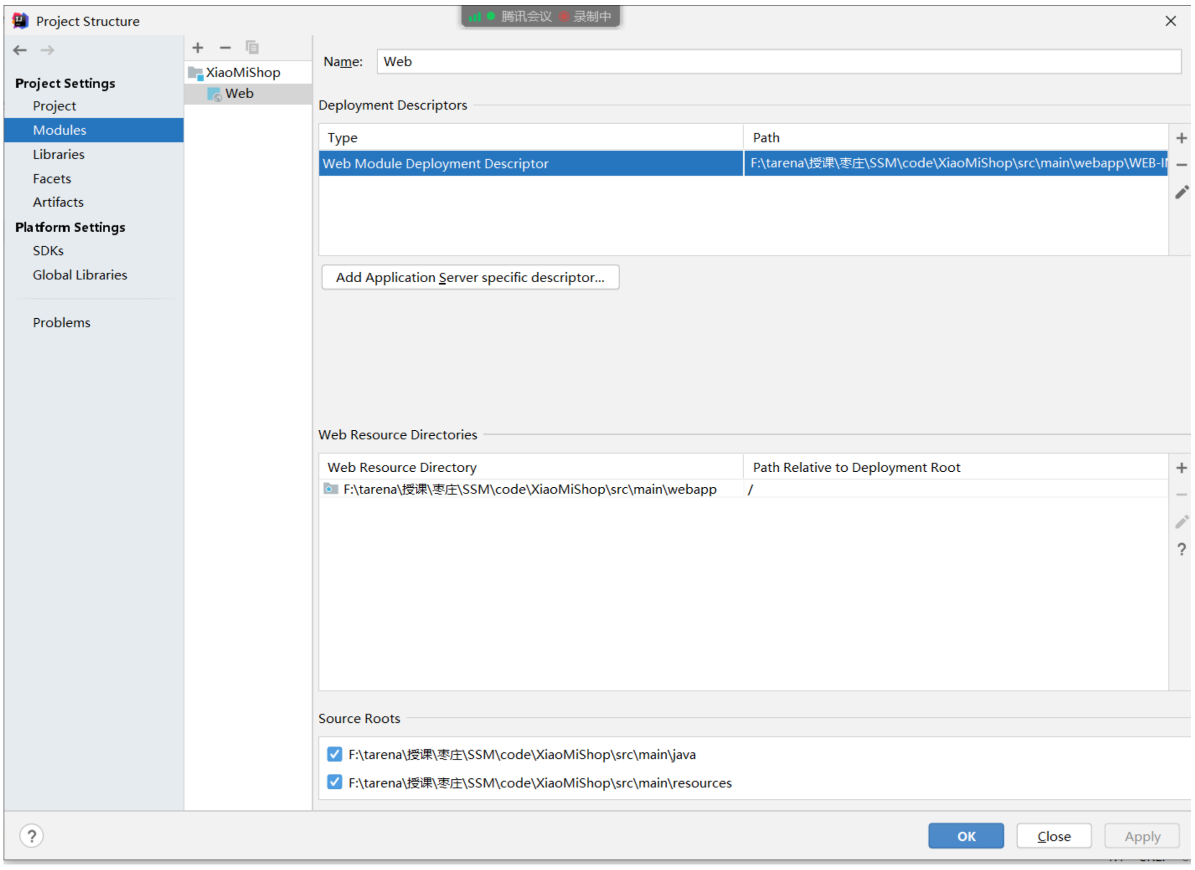
1 web.xml
Add Alt+Insert

Add Application Server specific descriptor...

选择路径


Deployment Descriptor Location
Web Module Deployment Descriptor (web.xml):
F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\webapp\WEB-INF\web.xml
Deployment descriptor version 4.0
OK Cancel

更改之后


Project Structure

腾讯会议 录制中

Project Settings
- Project
- Modules
- Libraries
- Facets
- Artifacts

Platform Settings
- SDKs
- Global Libraries

Problems

XiaoMiShop
Web

Name: Web

Deployment Descriptors

Type	Path
Web Module Deployment Descriptor	F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\webapp\WEB-I

Add Application Server specific descriptor...

Web Resource Directories

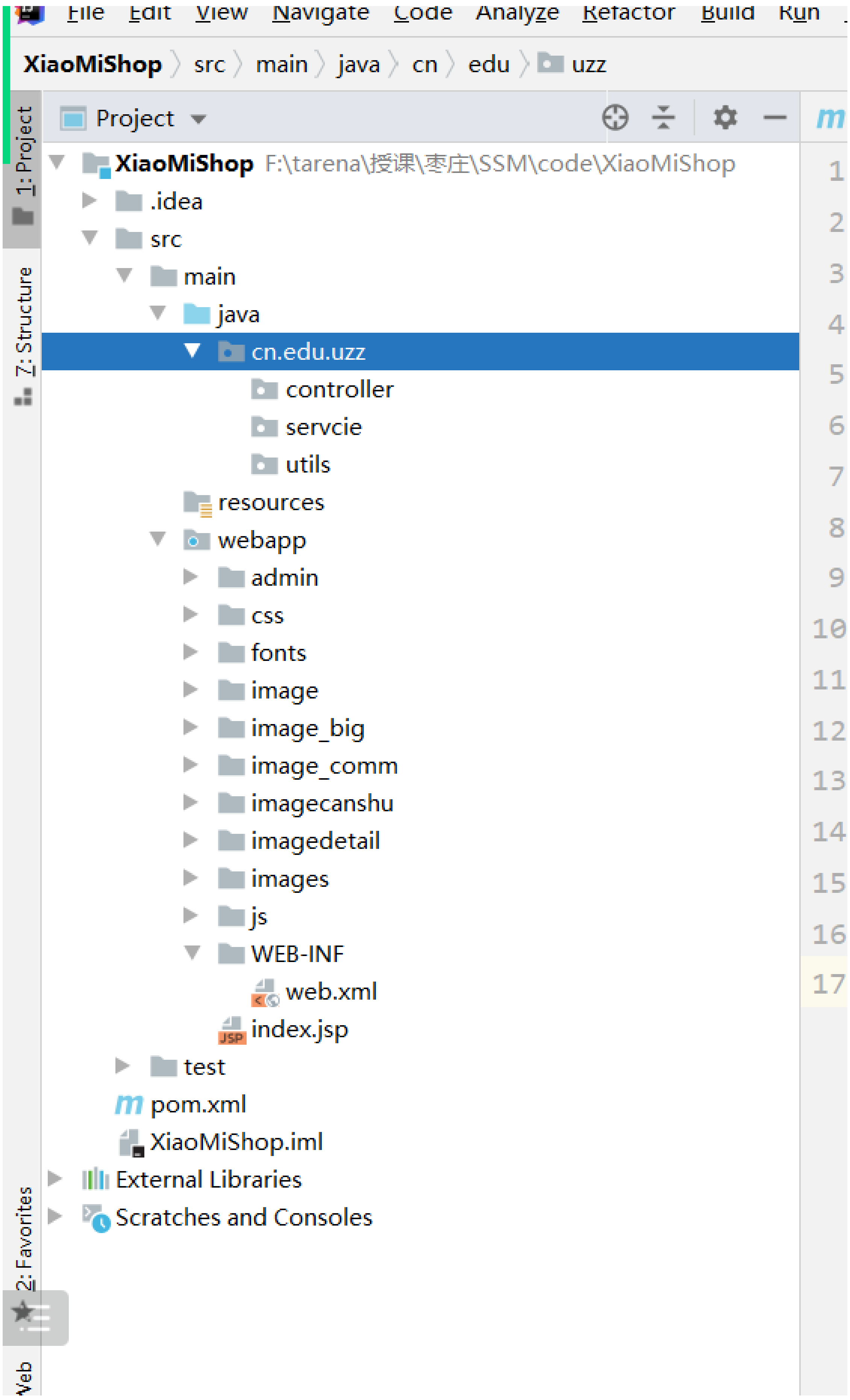
Web Resource Directory	Path Relative to Deployment Root
F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\webapp	/

Source Roots

- [✓] F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\java
- [✓] F:\tarena\授课\枣庄\SSM\code\XiaoMiShop\src\main\resources

OK Close Apply

目录结构


```
XiaoMiShop F:\tarena\授课\枣庄\SSM\code\XiaoMiShop
  .idea
  src
    main
      java
        cn.edu.uzz
          controller
          servcie
          utils
      resources
      webapp
        admin
        css
        fonts
        image
        image_big
        image_comm
        imagecanshu
        imagedetail
        images
        js
        WEB-INF
          web.xml
        index.jsp
    test
  pom.xml
  XiaoMiShop.iml
External Libraries
Scratches and Consoles
```

三、添加依赖

```
<!-- 集中定义依赖版本号 -->
<properties>
    <junit.version>4.12</junit.version>
```

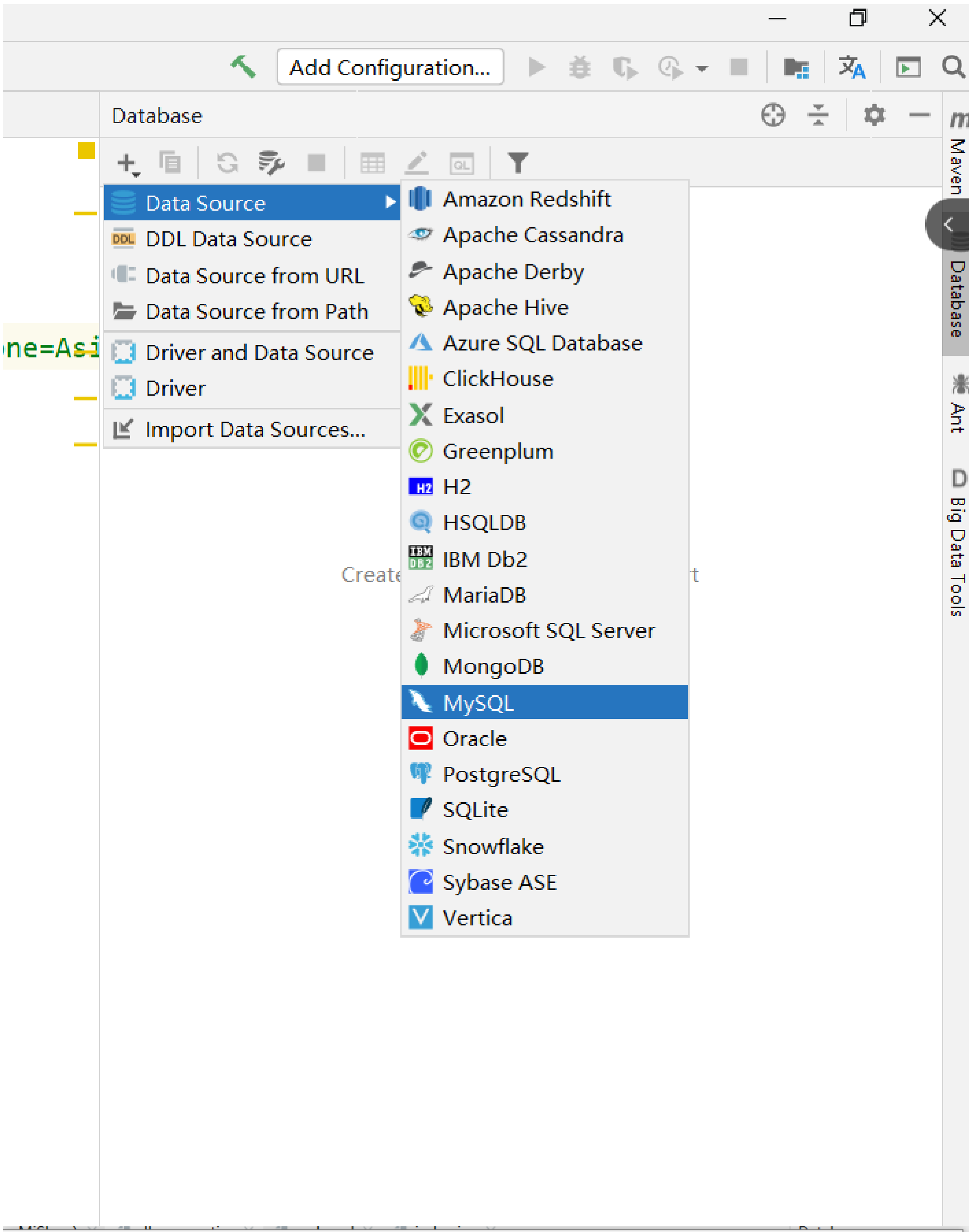
```xml
        <spring.version>5.2.5.RELEASE</spring.version>
        <mybatis.version>3.5.1</mybatis.version>
        <mybatis.spring.version>1.3.1</mybatis.spring.version>
        <mybatis.paginator.version>1.2.15</mybatis.paginator.version>
        <mysql.version>8.0.22</mysql.version>
        <slf4j.version>1.6.4</slf4j.version>
        <druid.version>1.1.12</druid.version>
        <pagehelper.version>5.1.2</pagehelper.version>
        <jstl.version>1.2</jstl.version>
        <servlet-api.version>3.0.1</servlet-api.version>
        <jsp-api.version>2.0</jsp-api.version>
        <jackson.version>2.9.6</jackson.version>
    </properties>
    <dependencies>

        <!-- spring -->
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-beans</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-webmvc</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-jdbc</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-aspects</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-jms</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-context-support</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <dependency>
            <groupId>org.springframework</groupId>
            <artifactId>spring-test</artifactId>
            <version>${spring.version}</version>
        </dependency>
        <!-- Mybatis -->
        <dependency>
```

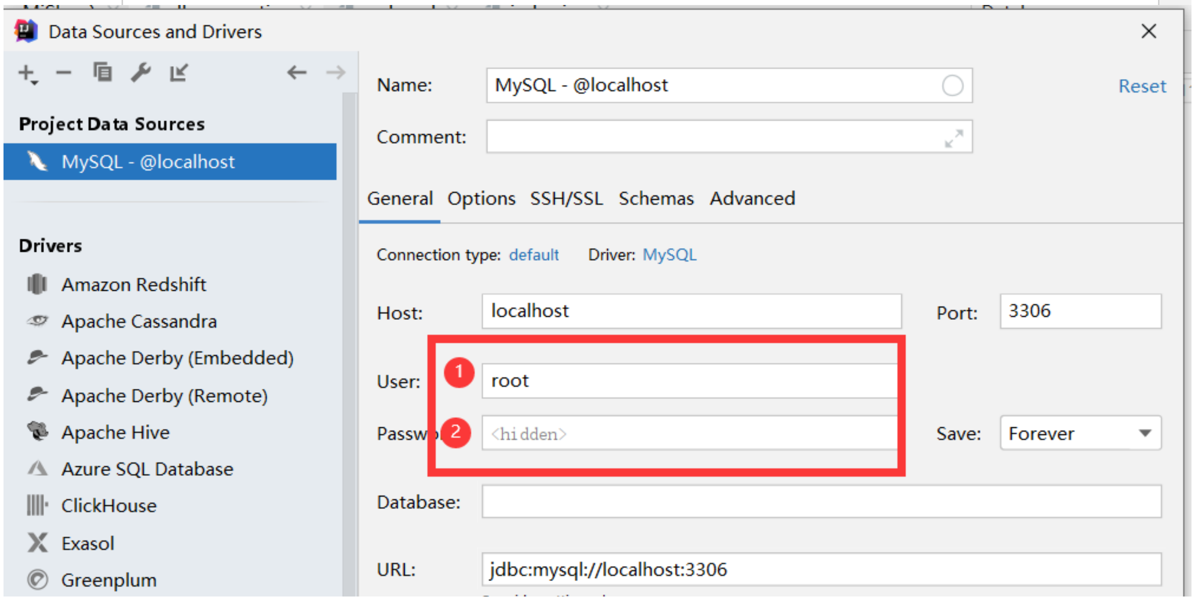
```xml
            <groupId>org.mybatis</groupId>
            <artifactId>mybatis</artifactId>
            <version>${mybatis.version}</version>
        </dependency>
        <dependency>
            <groupId>org.mybatis</groupId>
            <artifactId>mybatis-spring</artifactId>
            <version>${mybatis.spring.version}</version>
        </dependency>
        <dependency>
            <groupId>com.github.miemiedev</groupId>
            <artifactId>mybatis-paginator</artifactId>
            <version>${mybatis.paginator.version}</version>
        </dependency>
        <dependency>
            <groupId>com.github.pagehelper</groupId>
            <artifactId>pagehelper</artifactId>
            <version>${pagehelper.version}</version>
        </dependency>
        <!-- MySql -->
        <dependency>
            <groupId>mysql</groupId>
            <artifactId>mysql-connector-java</artifactId>
            <version>${mysql.version}</version>
        </dependency>
        <!-- 连接池 -->
        <dependency>
            <groupId>com.alibaba</groupId>
            <artifactId>druid</artifactId>
            <version>${druid.version}</version>
        </dependency>

        <!-- junit -->
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>${junit.version}</version>
            <scope>test</scope>
        </dependency>


        <!-- JSP相关 -->
        <dependency>
            <groupId>jstl</groupId>
            <artifactId>jstl</artifactId>
            <version>${jstl.version}</version>
        </dependency>
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>javax.servlet-api</artifactId>
            <version>3.0.1</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>jsp-api</artifactId>
            <scope>provided</scope>
            <version>${jsp-api.version}</version>
```

```xml
    </dependency>
    <!-- Jackson Json处理工具包 -->
    <dependency>
        <groupId>com.fasterxml.jackson.core</groupId>
        <artifactId>jackson-databind</artifactId>
        <version>${jackson.version}</version>
    </dependency>
    <dependency>
        <groupId>commons-io</groupId>
        <artifactId>commons-io</artifactId>
        <version>2.4</version>
    </dependency>
    <dependency>
        <groupId>commons-fileupload</groupId>
        <artifactId>commons-fileupload</artifactId>
        <version>1.3.1</version>
    </dependency>
</dependencies>

<!-- 插件配置 -->
<build>
    <plugins>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
            <configuration>
                <source>1.8</source>
                <target>1.8</target>
                <encoding>UTF-8</encoding>
            </configuration>
        </plugin>
    </plugins>
    <!--识别所有的配置文件-->
    <resources>
        <resource>
            <directory>src/main/java</directory>
            <includes>
                <include>**/*.properties</include>
                <include>**/*.xml</include>
            </includes>
            <filtering>false</filtering>
        </resource>
        <resource>
            <directory>src/main/resources</directory>
            <includes>
                <include>**/*.properties</include>
                <include>**/*.xml</include>
            </includes>
            <filtering>false</filtering>
        </resource>
    </resources>
</build>
```

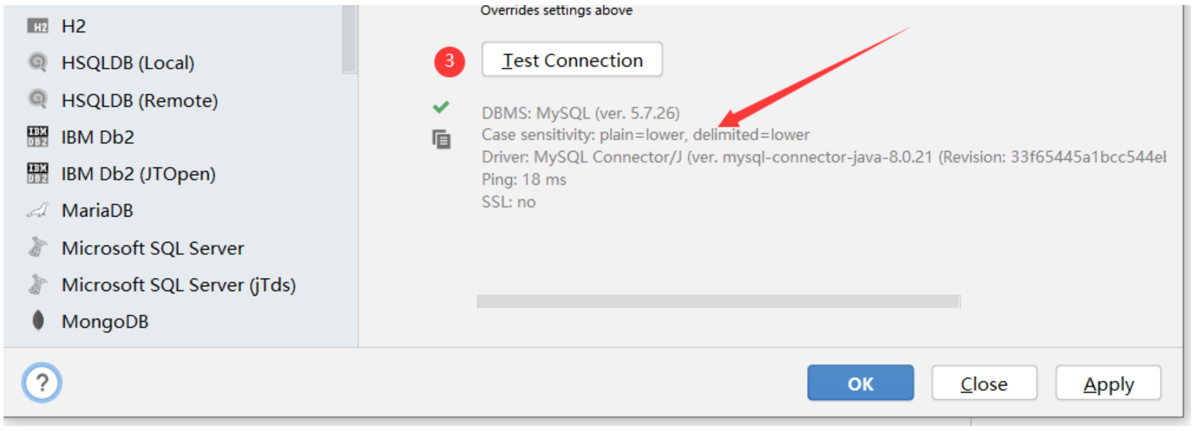
四、添加数据库配置文件

```
# MySQL8的驱动
jdbc.driver=com.mysql.cj.jdbc.Driver
# MySQL5的驱动
# jdbc.driver=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql://localhost:3306/xiaomissm?
useSSL=false&serverTimezone=Asia/Shanghai&allowPublicKeyRetrieval=true
jdbc.username=root
jdbc.password=root
```





Name: MySQL - @localhost
Comment:
Connection type: default Driver: MySQL
Host: localhost Port: 3306
User: 1 root
Passw 2 <hidden> Save: Forever
Database:
URL: jdbc:mysql://localhost:3306



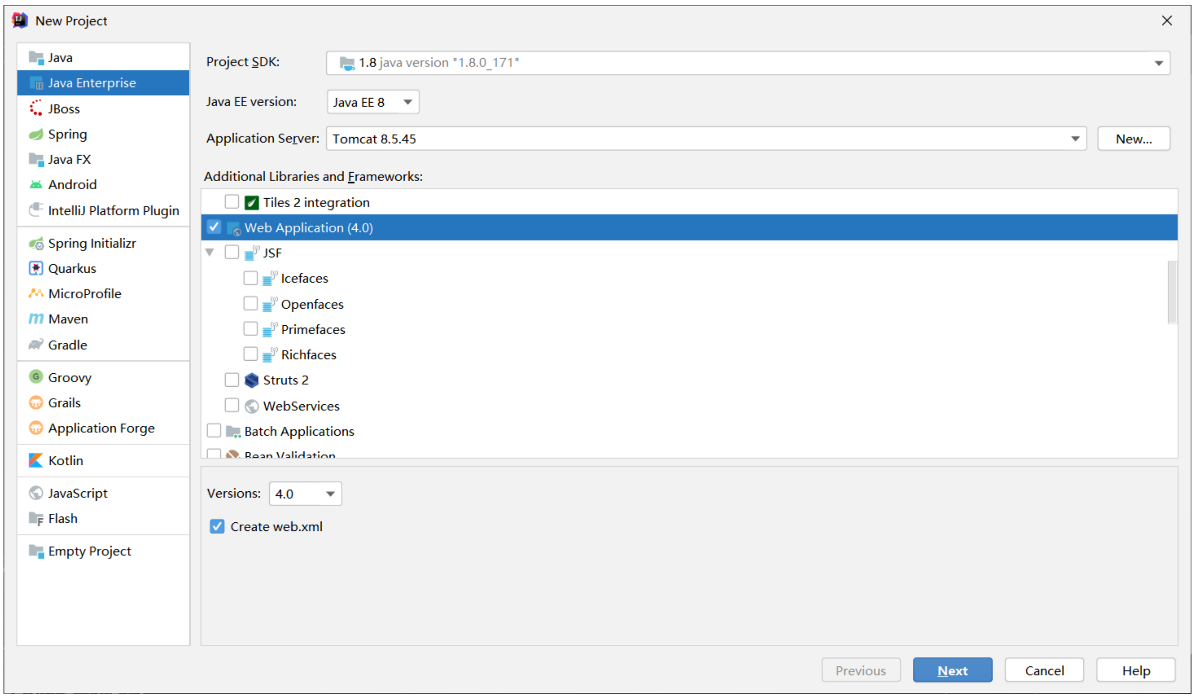
```
DBMS: MySQL (ver. 5.7.26)
Case sensitivity: plain=lower, delimited=lower
Driver: MySQL Connector/J (ver. mysql-connector-java-8.0.21 (Revision: 33f65445a1bcc544el
Ping: 18 ms
SSL: no
```

五、添加mybatis配置文件

SqlMapConfig.xml

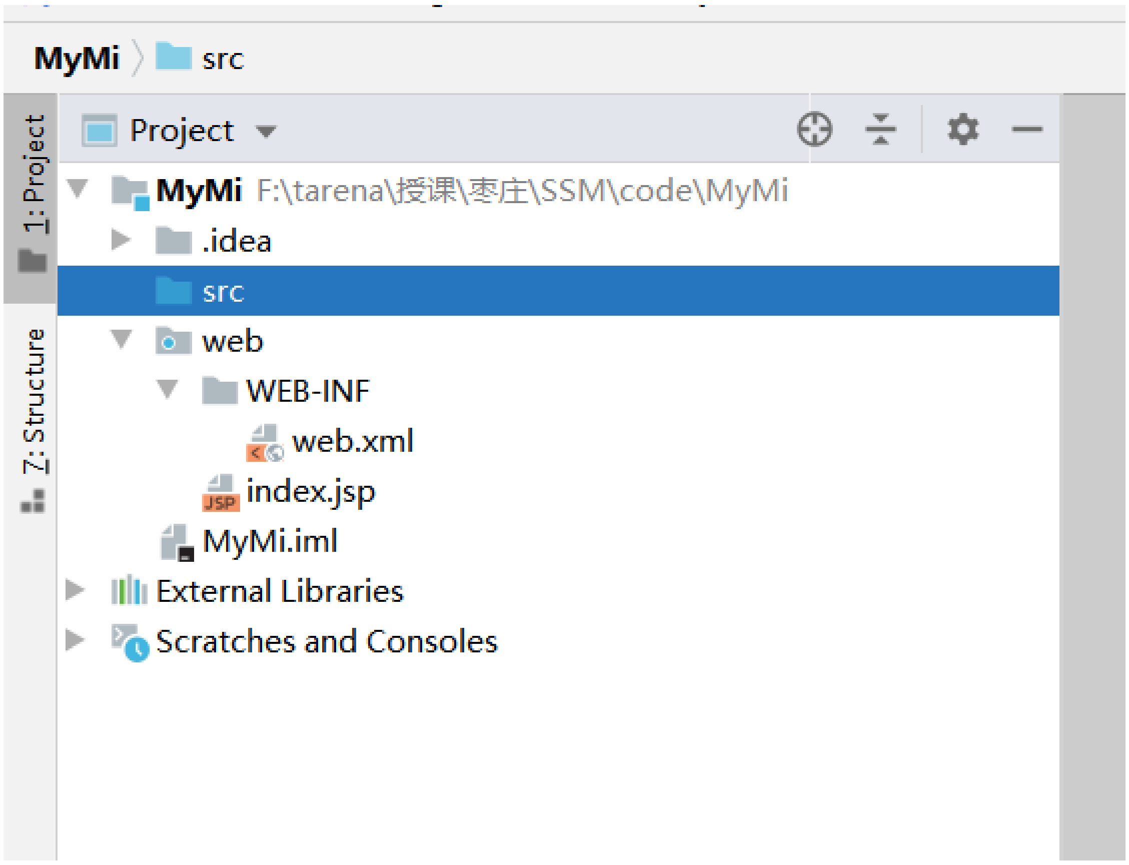
```xml
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
"http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
    <!-- 加载分页插件的配置 -->
    <plugins>
        <plugin interceptor="com.github.pagehelper.PageInterceptor"></plugin>
    </plugins>
</configuration>
```

六、另一种创建项目的方法



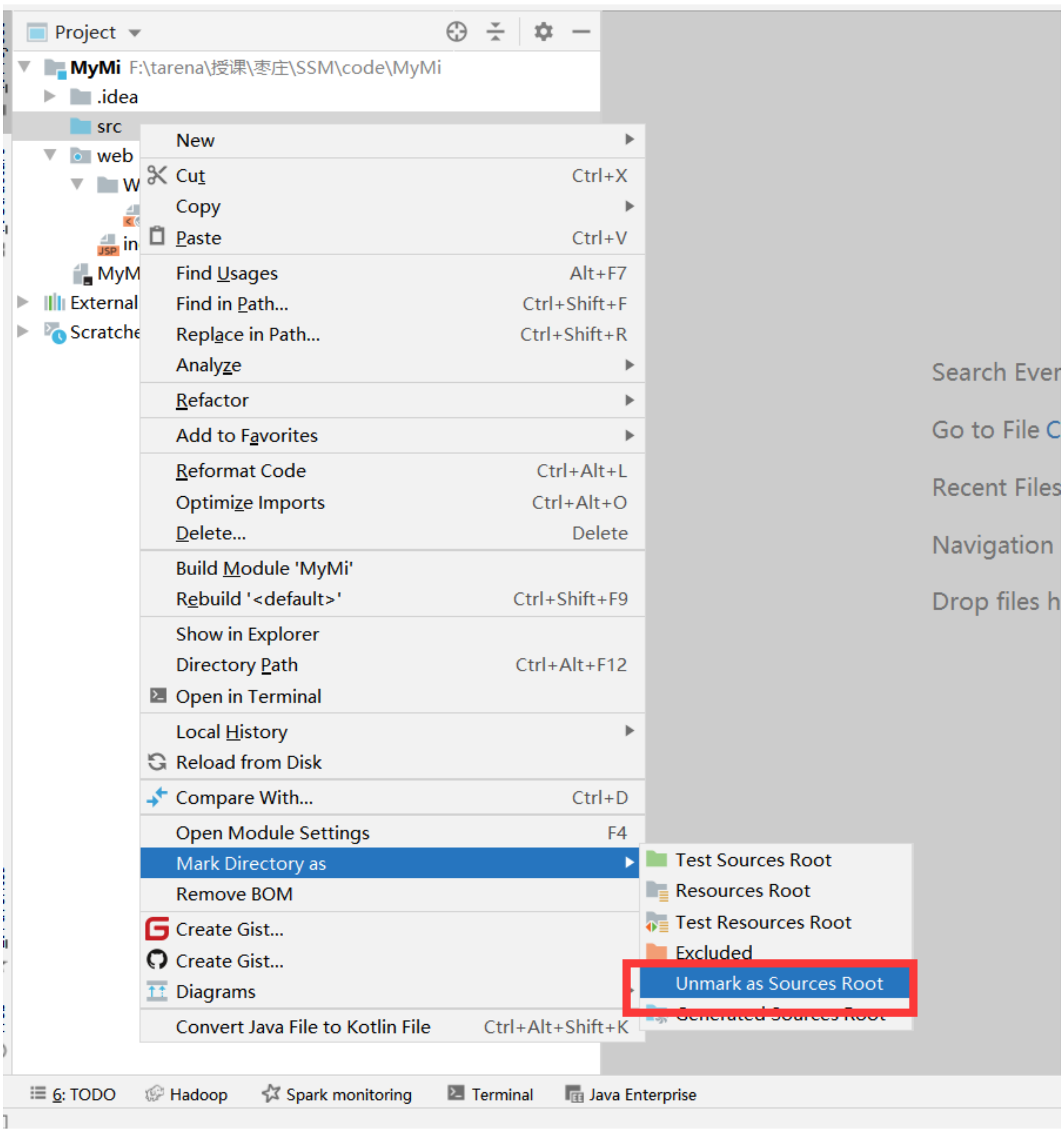
Project SDK: 1.8 java version "1.8.0_171"
Java EE version: Java EE 8
Application Server: Tomcat 8.5.45
[✓] Web Application (4.0)
Versions: 4.0
[✓] Create web.xml

项目初始目录

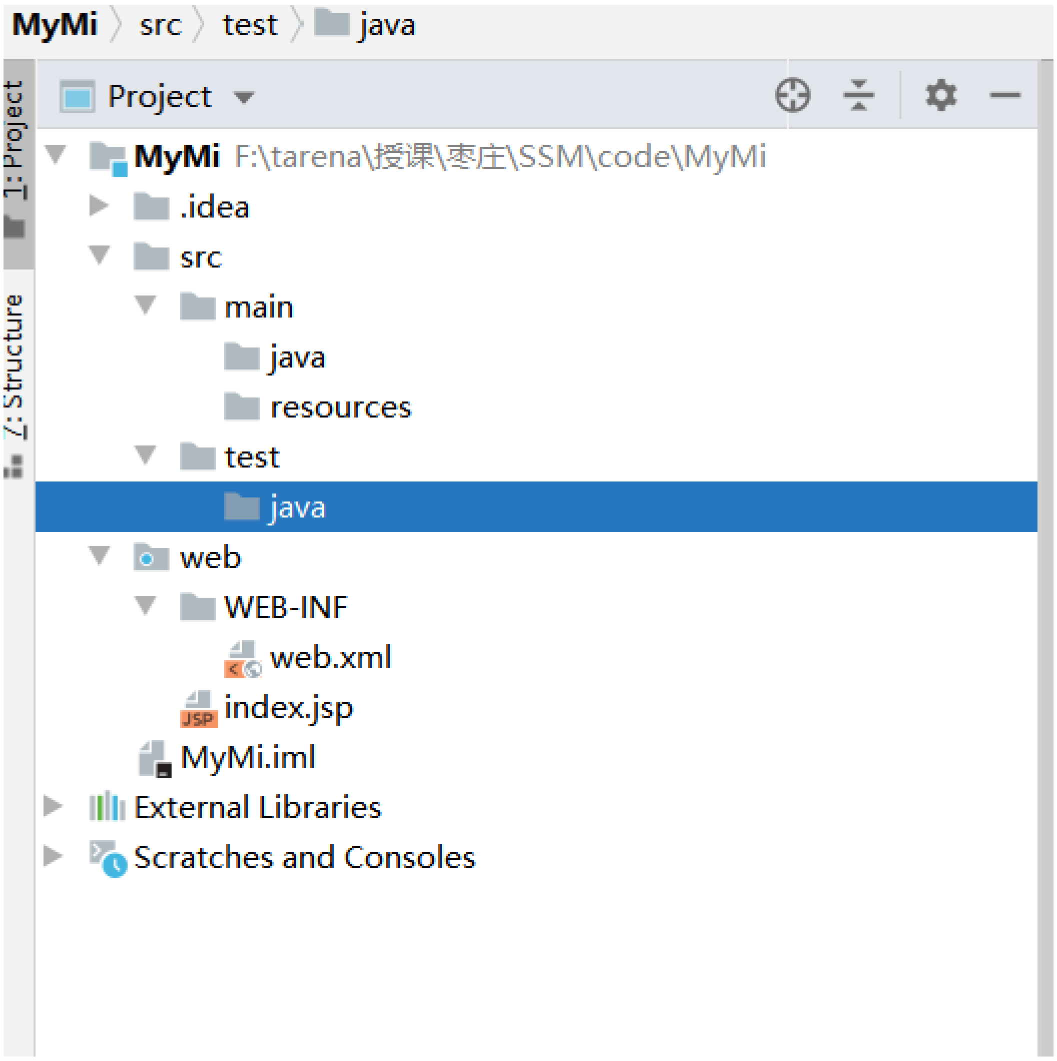


```
▼ MyMi  F:\tarena\授课\枣庄\SSM\code\MyMi
  ▶ .idea
    src
  ▼ web
    ▼ WEB-INF
        web.xml
      index.jsp
    MyMi.iml
▶ External Libraries
▶ Scratches and Consoles
```

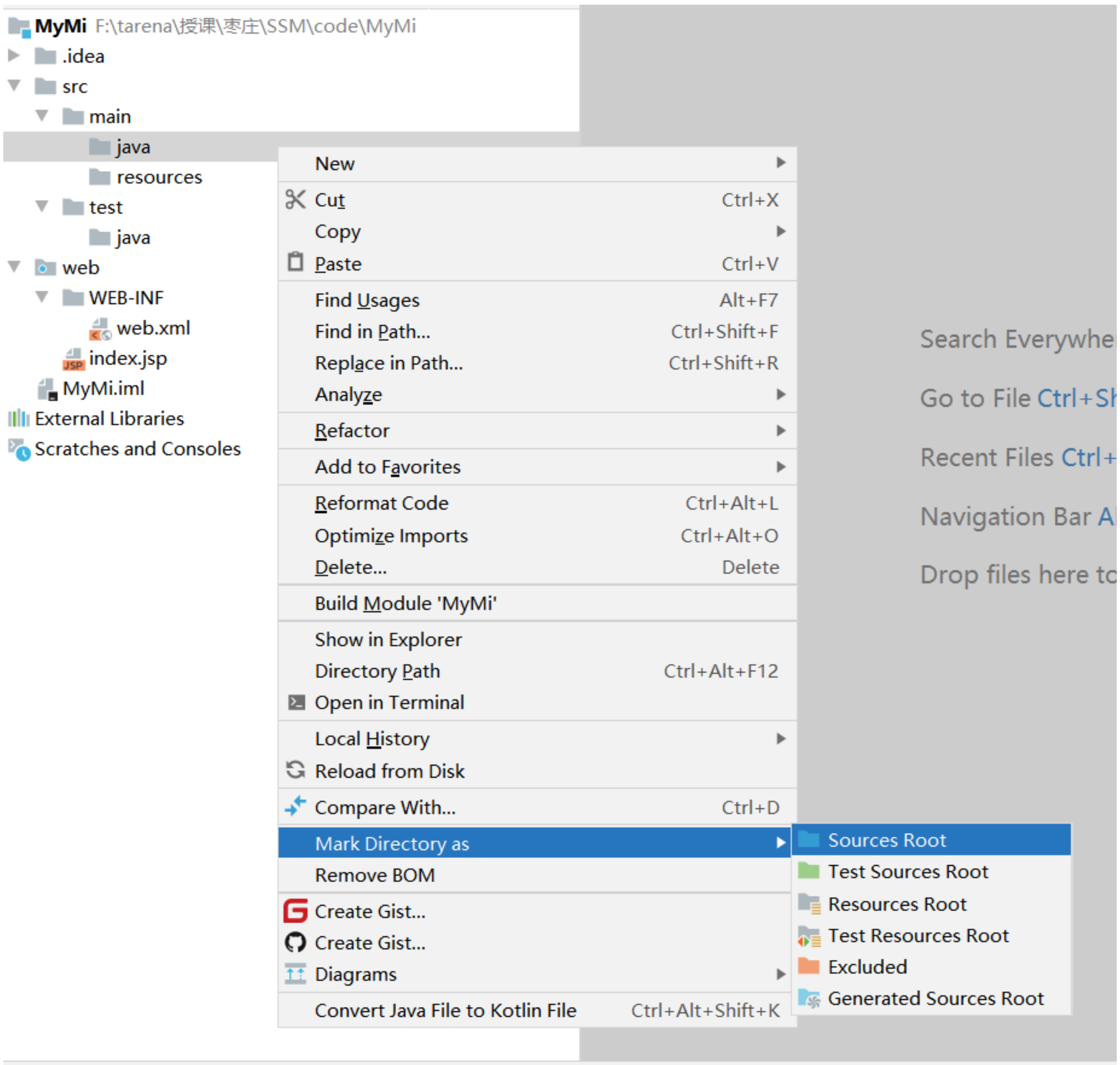
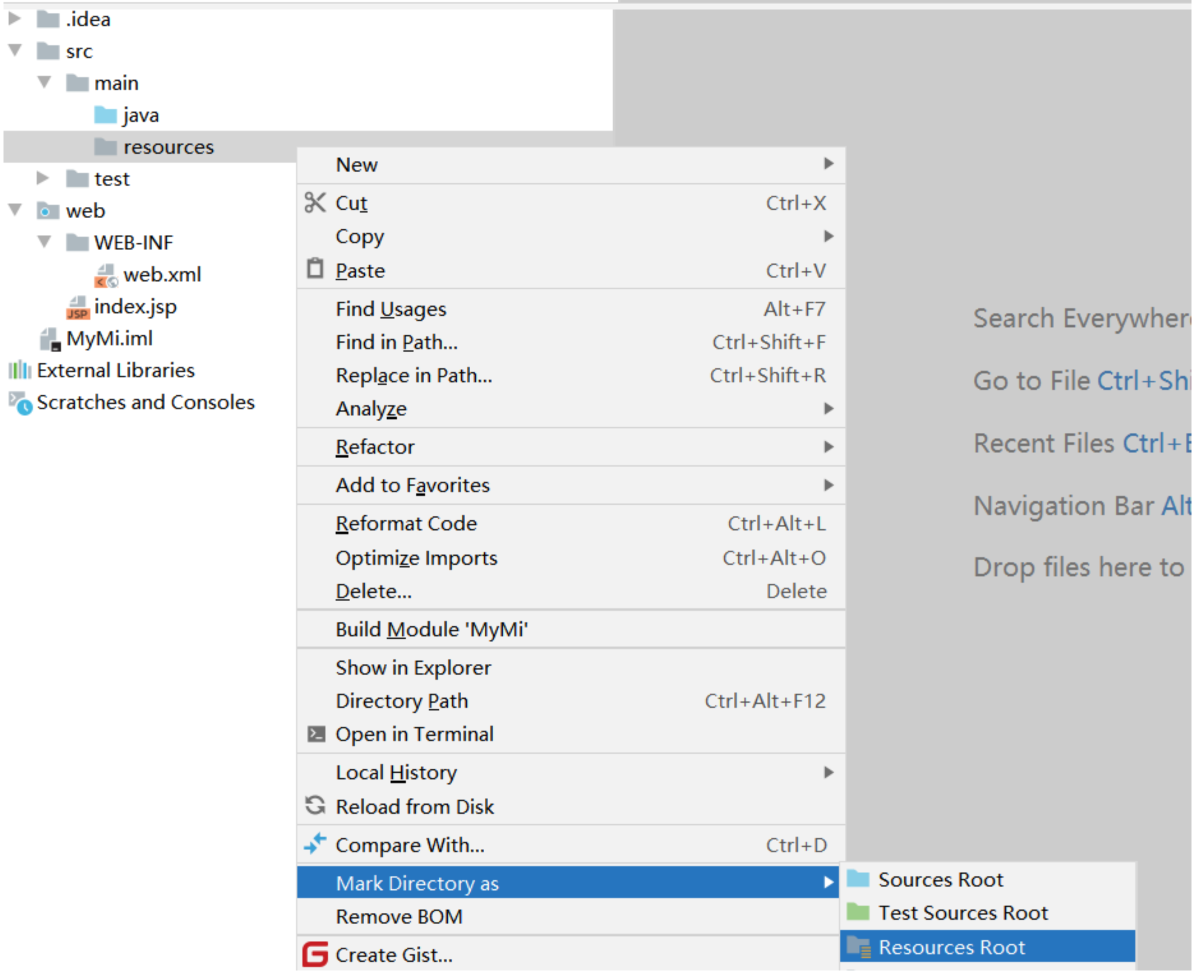
开始目录修改

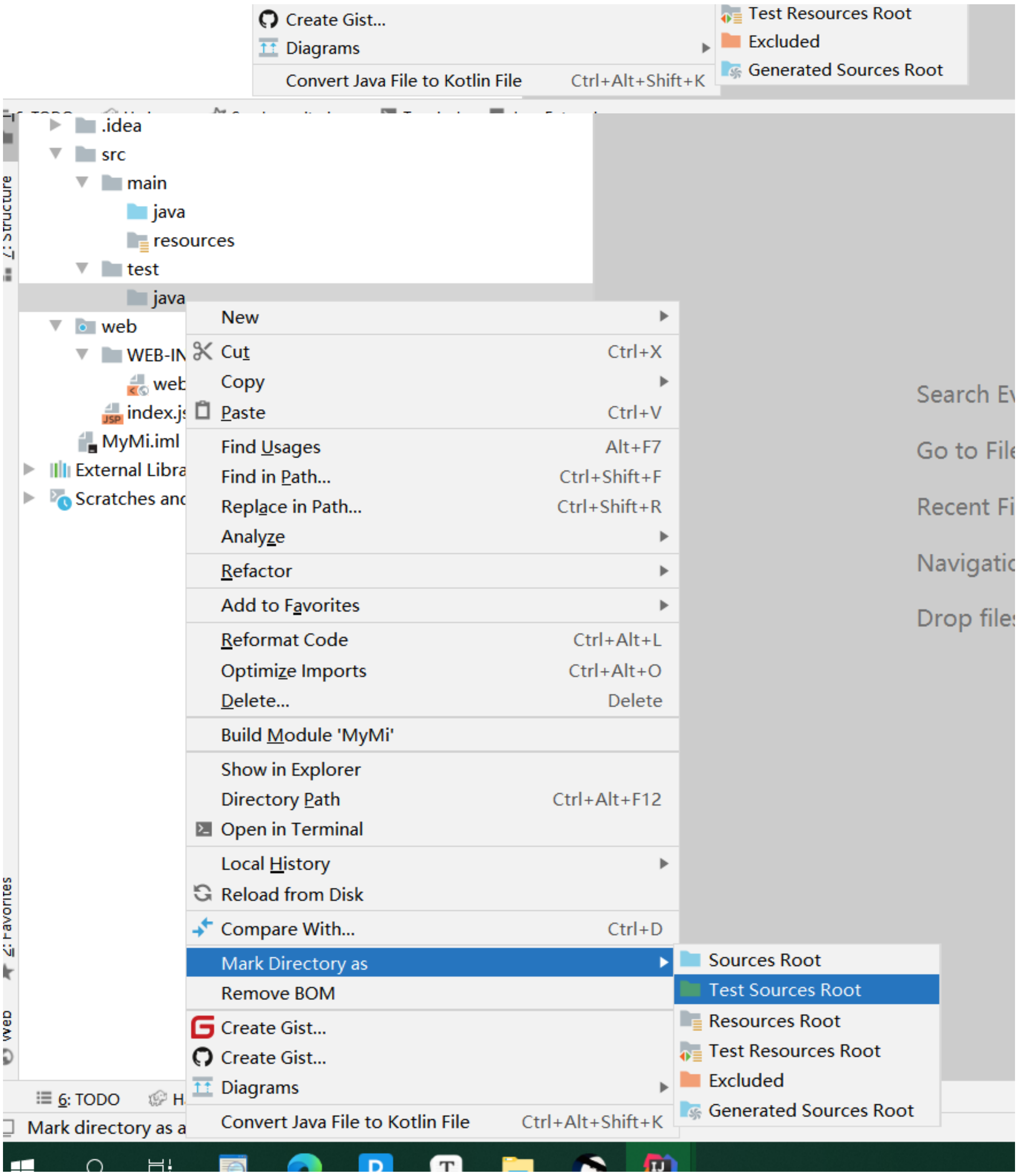

Project
MyMi F:\tarena\授课\枣庄\SSM\code\MyMi
.idea
src
web
External
Scratche
New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'MyMi'
Rebuild '<default>' Ctrl+Shift+F9
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Open Module Settings F4
Mark Directory as
Remove BOM
Create Gist...
Create Gist...
Diagrams
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Test Sources Root
Resources Root
Test Resources Root
Excluded
Unmark as Sources Root
Search Ever
Go to File
Recent Files
Navigation
Drop files h
6: TODO
Hadoop
Spark monitoring
Terminal
Java Enterprise

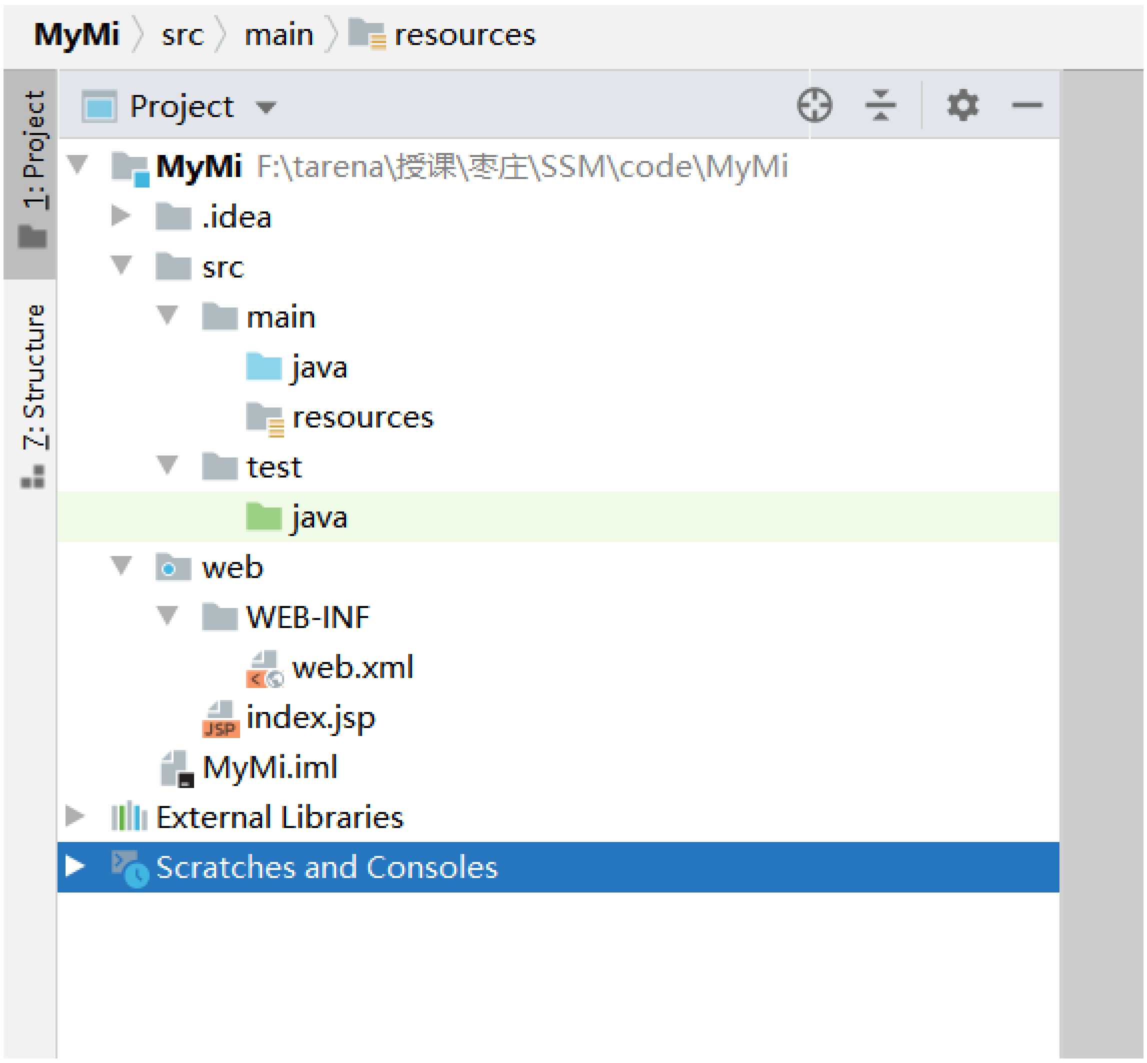
先新建文件夹



```
MyMi F:\tarena\授课\枣庄\SSM\code\MyMi
  .idea
  src
    main
      java
      resources
    test
      java
  web
    WEB-INF
      web.xml
    index.jsp
  MyMi.iml
External Libraries
Scratches and Consoles
```


MyMi F:\tarena\授课\枣庄\SSM\code\MyMi
.idea
src
main
java
resources
test
java
web
WEB-INF
web.xml
index.jsp
MyMi.iml
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'MyMi'
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...
Create Gist...
Diagrams
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Sources Root
Test Sources Root
Resources Root
Test Resources Root
Excluded
Generated Sources Root
Search Everywhe
Go to File Ctrl+Sh
Recent Files Ctrl+
Navigation Bar A
Drop files here to

.idea
src
main
java
resources
test
web
WEB-INF
web.xml
index.jsp
MyMi.iml
External Libraries
Scratches and Consoles
New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'MyMi'
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...
Sources Root
Test Sources Root
Resources Root
Search Everywher
Go to File Ctrl+Shi
Recent Files Ctrl+E
Navigation Bar Alt
Drop files here to


Create Gist...
Diagrams
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Test Resources Root
Excluded
Generated Sources Root
.idea
src
main
java
resources
test
java
web
WEB-IN
web
index.j
MyMi.iml
External Libra
Scratches and
New
Cut Ctrl+X
Copy
Paste Ctrl+V
Find Usages Alt+F7
Find in Path... Ctrl+Shift+F
Replace in Path... Ctrl+Shift+R
Analyze
Refactor
Add to Favorites
Reformat Code Ctrl+Alt+L
Optimize Imports Ctrl+Alt+O
Delete... Delete
Build Module 'MyMi'
Show in Explorer
Directory Path Ctrl+Alt+F12
Open in Terminal
Local History
Reload from Disk
Compare With... Ctrl+D
Mark Directory as
Remove BOM
Create Gist...
Create Gist...
Diagrams
Convert Java File to Kotlin File Ctrl+Alt+Shift+K
Sources Root
Test Sources Root
Resources Root
Test Resources Root
Excluded
Generated Sources Root
Search Ev
Go to File
Recent Fi
Navigatio
Drop files
6: TODO
Mark directory as a
7: Structure
2: Favorites
Web



七、Spring配置文件

applicationContext_dao.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context.xsd">

    <!-- 读取连接数据库的配置文件 db.properties -->
    <context:property-placeholder location="classpath:db.properties">
</context:property-placeholder>

    <!-- 创建数据源 -->
    <bean id="dataSource" class="com.alibaba.druid.pool.DruidDataSource">
        <property name="driverClassName" value="${jdbc.driver}"/>
        <property name="url" value="${jdbc.url}"/>
        <property name="username" value="${jdbc.username}"/>
        <property name="password" value="${jdbc.password}"/>
    </bean>

    <!-- 创建SqlSessionFactoryBean -->
    <bean class="org.mybatis.spring.SqlSessionFactoryBean">
```

```
        <!-- 配置数据源 -->
        <property name="dataSource" ref="dataSource"></property>
        <!-- 配置mybatis的核心配置文件 -->
        <property name="configLocation" value="classpath:SqlMapConfig.xml">
</property>
        <!-- 配置实体类 -->
        <property name="typeAliasesPackage" value="cn.edu.uzz.pojo"></property>
    </bean>
</beans>
```

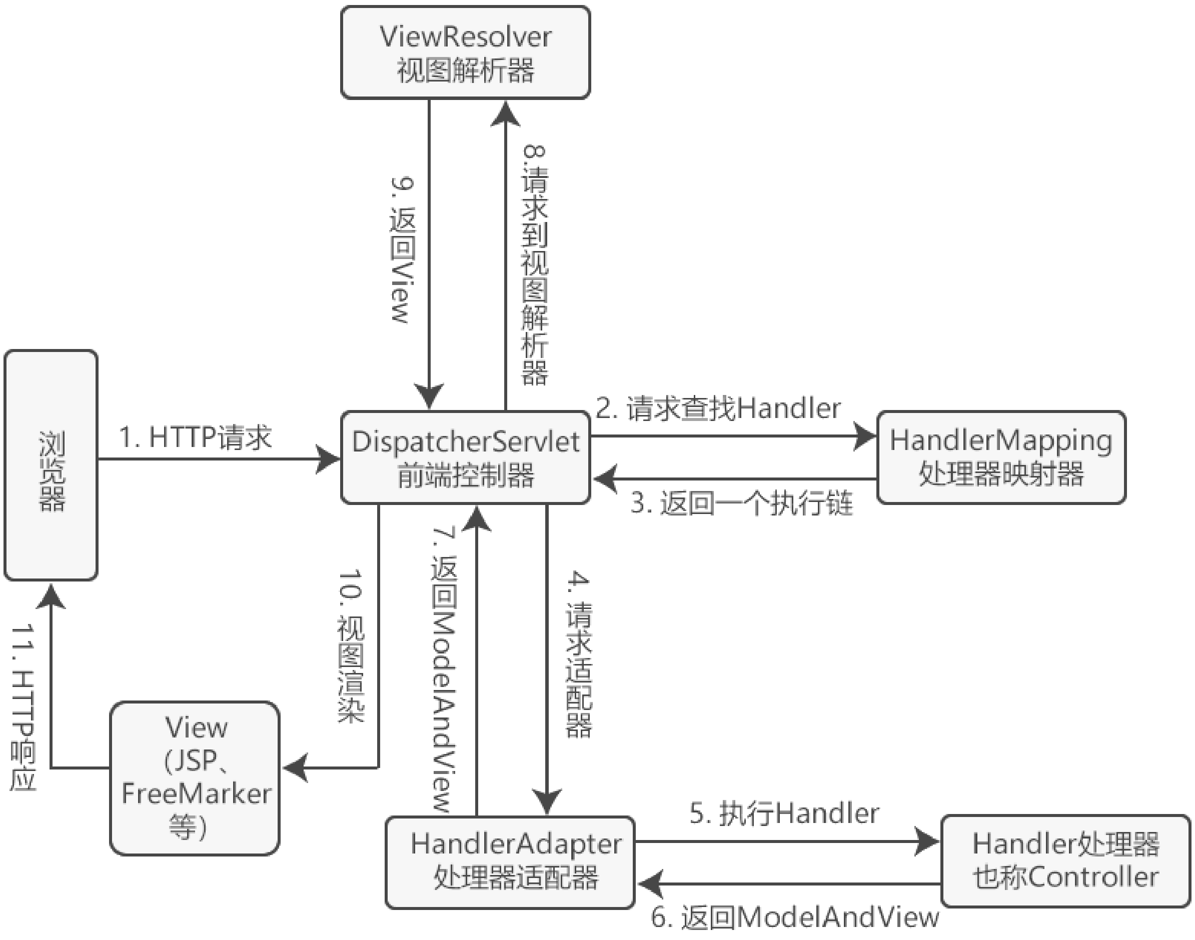
applicationContext_service.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xmlns:aop="http://www.springframework.org/schema/aop"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context.xsd
       http://www.springframework.org/schema/tx
       http://www.springframework.org/schema/tx/spring-tx.xsd
       http://www.springframework.org/schema/aop
       https://www.springframework.org/schema/aop/spring-aop.xsd">

    <!--
        设置业务逻辑的包扫描器，目的是在指定的路径下，使用@Service注解的类，
        Spring负责创建对象，并加载依赖
    -->
    <context:component-scan base-package="cn.edu.uzz.service">
</context:component-scan>

    <!-- 设置事务管理器 -->
    <bean id="transactionManager"
class="org.springframework.jdbc.datasource.DataSourceTransactionManager">
        <!--
            可能会有报错，因为applicationContext-dao和applicationContext-service文件
内容不在一起的原因
            但是容器在加载配置文件的时候，会加载到
        -->
        <property name="dataSource" ref="dataSource"></property>
    </bean>
    <!-- 添加事务的切面 -->
    <tx:advice id="myAdvice" transaction-manager="transactionManager" >
        <tx:attributes>
            <!-- 设置所有的查询方法为只读，在查询的时候，其他的操作会回避 -->
            <tx:method name="*select*" read-only="true"/>
            <tx:method name="*find*" read-only="true"/>
            <tx:method name="*get*" read-only="true"/>
            <tx:method name="*search*" read-only="true"/>
            <!-- 增删改设置为支持事务 -->
            <tx:method name="*insert*" propagation="REQUIRED"/>
            <tx:method name="*save*" propagation="REQUIRED"/>
            <tx:method name="*add*" propagation="REQUIRED"/>
            <tx:method name="*delete*" propagation="REQUIRED"/>
            <tx:method name="*remove*" propagation="REQUIRED"/>
            <tx:method name="*clear*" propagation="REQUIRED"/>
            <tx:method name="*update*" propagation="REQUIRED"/>
            <tx:method name="*modify*" propagation="REQUIRED"/>
            <tx:method name="*change*" propagation="REQUIRED"/>
            <tx:method name="*set*" propagation="REQUIRED"/>
            <!-- 如果上面的都不匹配的话，匹配*，设置支持事务 -->
            <tx:method name="*" propagation="REQUIRED"/>
        </tx:attributes>
    </tx:advice>

    <!-- 完成切面的切入点织入（把切入点和切面绑定在一起） -->
    <aop:config>
        <!--
            切入点表达式
            * 返回值类型：任意
            cn.edu.uzz.service.*.*(..))：service包下的所有的类和所有的方法的任意参数，都
追加事务处理
        -->
        <aop:pointcut id="myPointcut" expression="execution(*
cn.edu.uzz.service.*.*(..))"/>
        <!-- 绑定 -->
        <aop:advisor advice-ref="myAdvice" pointcut-ref="myPointcut">
</aop:advisor>
    </aop:config>
</beans>
```

八、SpringMVC配置文件

```
M：model，service

V：view，jsp

C：controller，***Action、***Controller、***Dao
```

SpringMVC执行流程



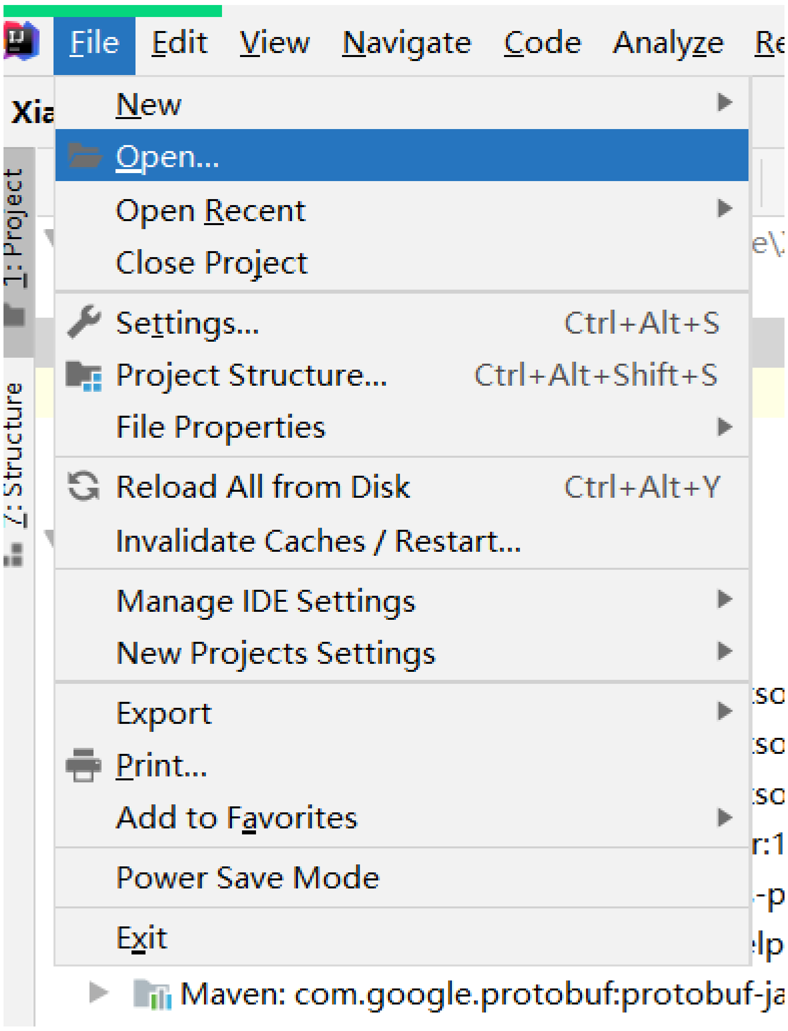
```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:mvc="http://www.springframework.org/schema/mvc"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context.xsd
       http://www.springframework.org/schema/mvc
       http://www.springframework.org/schema/mvc/spring-mvc.xsd">

    <!-- 设置包扫描，扫描控制层包 -->
    <context:component-scan base-package="cn.edu.uzz.controller" />

    <!--
    视图解析器 ViewResolver，设置浏览器拦截的请求
    localhost:8080/admin/login.jsp
     -->
    <bean id="viewResolver"
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/admin/"></property>
        <property name="suffix" value=".jsp"></property>
    </bean>

    <!-- 文件上传插件的配置，注意，这里的id要固定，必须是multipartResolver -->
    <bean id="multipartResolver"
class="org.springframework.web.multipart.commons.CommonsMultipartResolver" />


    <!-- 基于注解开发，设置注解驱动 -->
    <mvc:annotation-driven />

</beans>
```

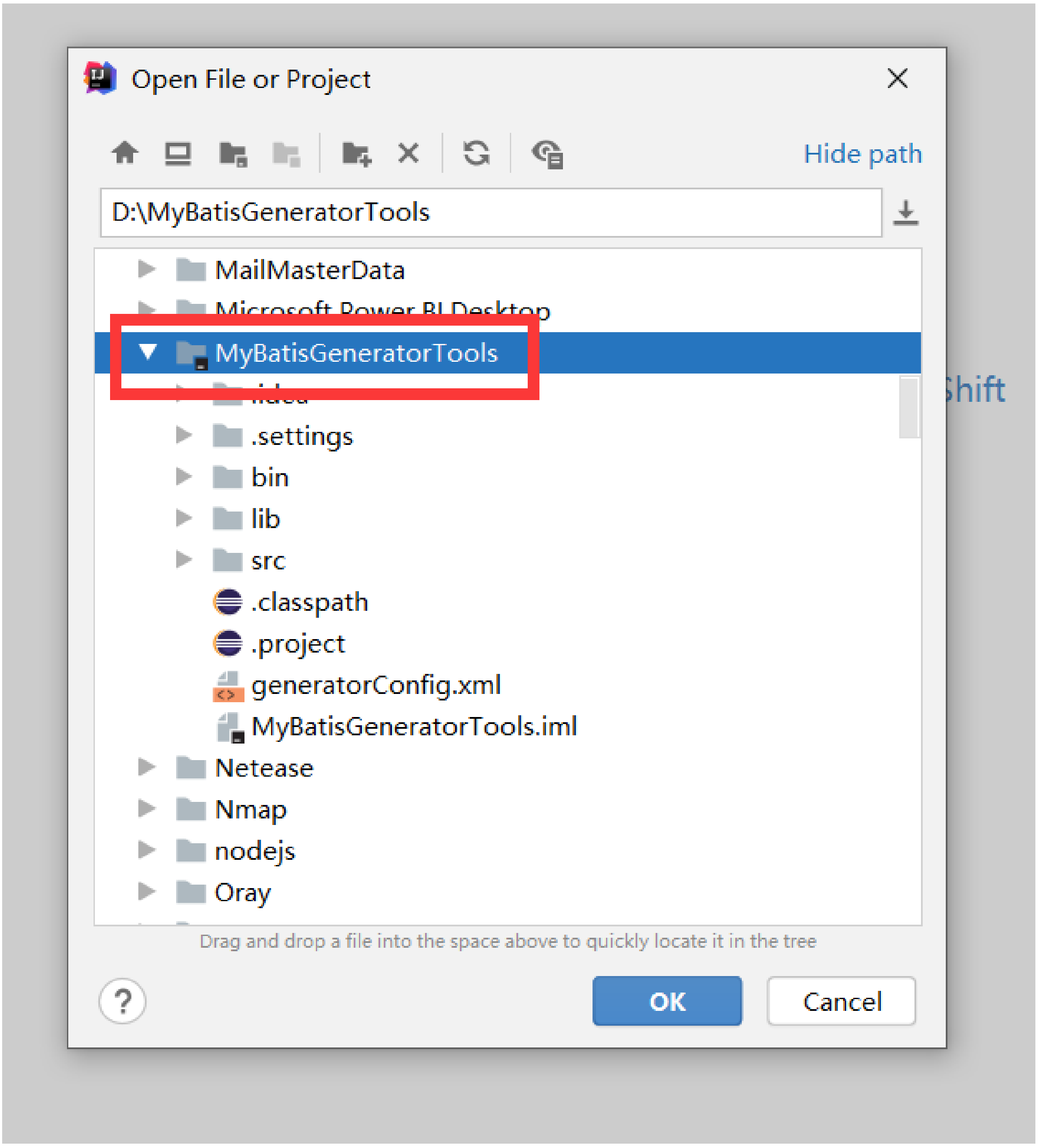
九、配置web.xml描述部署文件

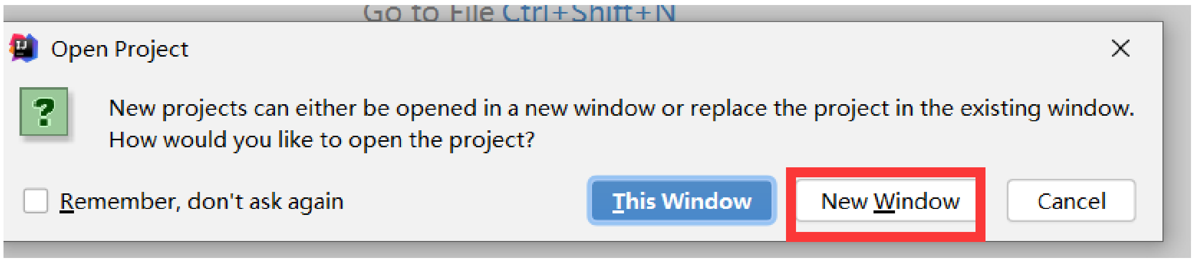
```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
         version="4.0">
    <!-- 配置过滤器，做字符编码的过滤器，放在第一个写 -->
    <filter>
        <filter-name>encode</filter-name>
        <filter-
class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>
        <init-param>
            <param-name>encoding</param-name>
            <param-value>UTF-8</param-value>
        </init-param>
        <init-param>
            <param-name>forceRequestEncoding</param-name>
            <param-value>true</param-value>
        </init-param>
        <init-param>
            <param-name>forceResponseEncoding</param-name>
            <param-value>true</param-value>
        </init-param>
    </filter>
    <filter-mapping>
        <filter-name>encode</filter-name>
        <!-- 拦截所有的请求 -->
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <!-- 注册springmvc -->
    <servlet>
        <servlet-name>springmvc</servlet-name>
        <servlet-
class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
        <init-param>
            <param-name>contextConfigLocation</param-name>
            <param-value>classpath:springmvc.xml</param-value>
        </init-param>
    </servlet>
    <servlet-mapping>
        <servlet-name>springmvc</servlet-name>
        <!-- 当客户端发起请求的时候，会发给DispatcherServlet进行处理，其他的请求我们不拦截
-->
        <url-pattern>*.action</url-pattern>
    </servlet-mapping>

    <!-- 监听器 -->
    <listener>
        <listener-
class>org.springframework.web.context.ContextLoaderListener</listener-class>
    </listener>
    <context-param>
        <param-name>contextConfigLocation</param-name>
        <!-- 通配 -->
        <param-value>classpath:applicationContext-*.xml</param-value>
    </context-param>
</web-app>
```

十、逆向工程生成Pojo和Mapper文件

```
首先把逆向工程MyBatisGeneratorTools复制到D盘根路径下，然后我们用idea打开项目，jdk版本最好是8
```







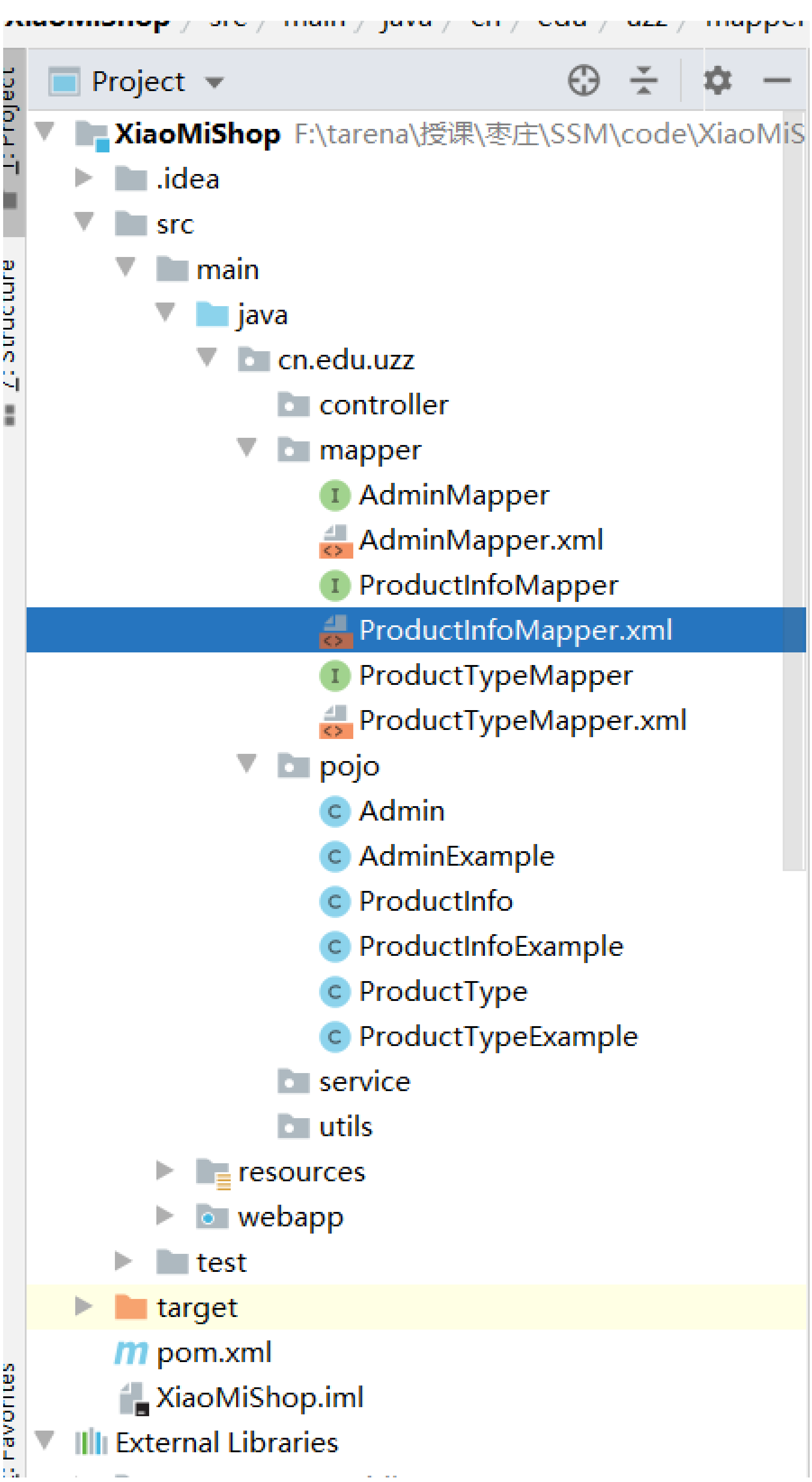
New projects can either be opened in a new window or replace the project in the existing window.
How would you like to open the project?

更改配置文件generatorConfig.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE generatorConfiguration
  PUBLIC "-//mybatis.org//DTD MyBatis Generator Configuration 1.0//EN"
  "http://mybatis.org/dtd/mybatis-generator-config_1_0.dtd">

<generatorConfiguration>
    <context id="testTables" targetRuntime="MyBatis3">
        <commentGenerator>
            <!-- 是否去除自动生成的注释 true：是 ： false:否 -->
            <property name="suppressAllComments" value="true" />
        </commentGenerator>
        <!--数据库连接的信息：驱动类、连接地址、用户名、密码 -->
<!--        <jdbcConnection driverClass="com.mysql.cj.jdbc.Driver"-->
        <jdbcConnection driverClass="com.mysql.jdbc.Driver"
            connectionURL="jdbc:mysql://localhost:3306/xiaomissm?
useSSL=false&amp;serverTimezone=Asia/Shanghai&amp;allowPublicKeyRetrieval=true"
                        userId="root" password="root">
        </jdbcConnection>
        <!-- 默认false，把JDBC DECIMAL 和 NUMERIC 类型解析为 Integer，为 true时把JDBC
DECIMAL 和
            NUMERIC 类型解析为java.math.BigDecimal -->
        <javaTypeResolver>
            <property name="forceBigDecimals" value="false" />
        </javaTypeResolver>

        <!-- targetProject:生成PO类的位置 -->
        <javaModelGenerator targetPackage="cn.edu.uzz.pojo"
            targetProject=".\src">
            <!-- enableSubPackages:是否让schema作为包的后缀 -->
            <property name="enableSubPackages" value="false" />
            <!-- 从数据库返回的值被清理前后的空格 -->
            <property name="trimStrings" value="true" />
        </javaModelGenerator>
        <!-- targetProject:mapper映射文件生成的位置 -->
        <sqlMapGenerator targetPackage="cn.edu.uzz.mapper"
            targetProject=".\src">
            <!-- enableSubPackages:是否让schema作为包的后缀 -->
            <property name="enableSubPackages" value="false" />
        </sqlMapGenerator>
        <!-- targetPackage：mapper接口生成的位置 -->
        <javaClientGenerator type="XMLMAPPER"
            targetPackage="cn.edu.uzz.mapper"
            targetProject=".\src">
            <!-- enableSubPackages:是否让schema作为包的后缀 -->
            <property name="enableSubPackages" value="false" />
        </javaClientGenerator>
        <!-- 指定数据库表 -->
        <table schema="" tableName="admin"></table>
        <table schema="" tableName="product_info"></table>
        <table schema="" tableName="product_type"></table>



    </context>
</generatorConfiguration>
```

> 在生成mapper和pojo前，一定要把逆向工程里之前生成的内容删除掉

```
执行GeneratorSqlmap.java
然后把生成的文件复制到我们的项目中
```



```
▼ XiaoMiShop  F:\tarena\授课\枣庄\SSM\code\XiaoMiS
  ▶ .idea
  ▼ src
    ▼ main
      ▼ java
        ▼ cn.edu.uzz
            controller
          ▼ mapper
              AdminMapper
              AdminMapper.xml
              ProductInfoMapper
              ProductInfoMapper.xml
              ProductTypeMapper
              ProductTypeMapper.xml
          ▼ pojo
              Admin
              AdminExample
              ProductInfo
              ProductInfoExample
              ProductType
              ProductTypeExample
            service
            utils
      ▶ resources
      ▶ webapp
    ▶ test
  ▶ target
    pom.xml
    XiaoMiShop.iml
▼ External Libraries
```