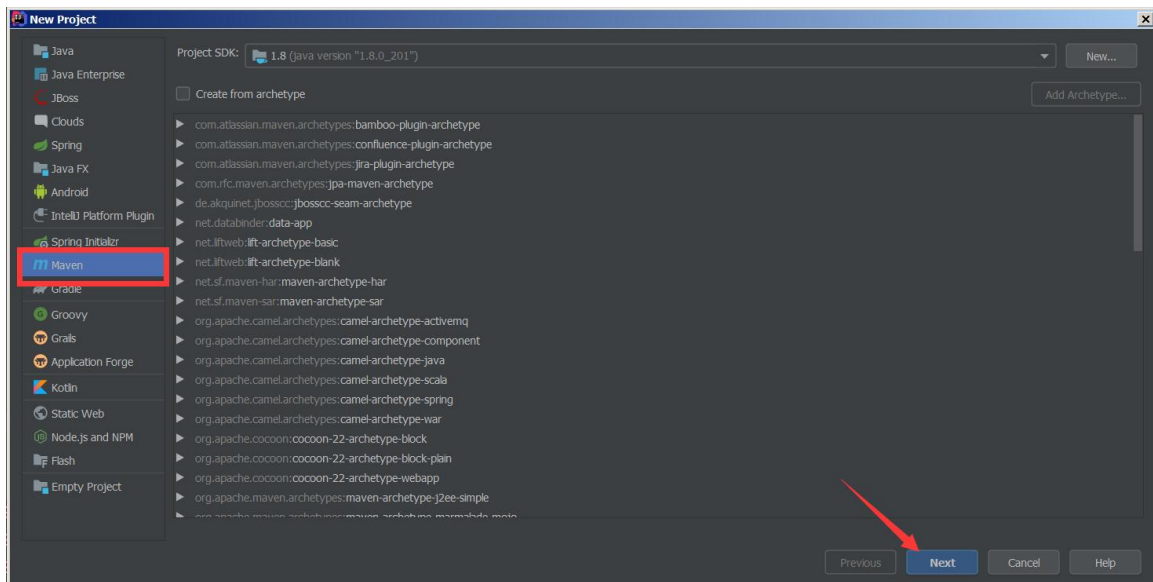


第八讲 SSM 框架集成

一、单一模块构建

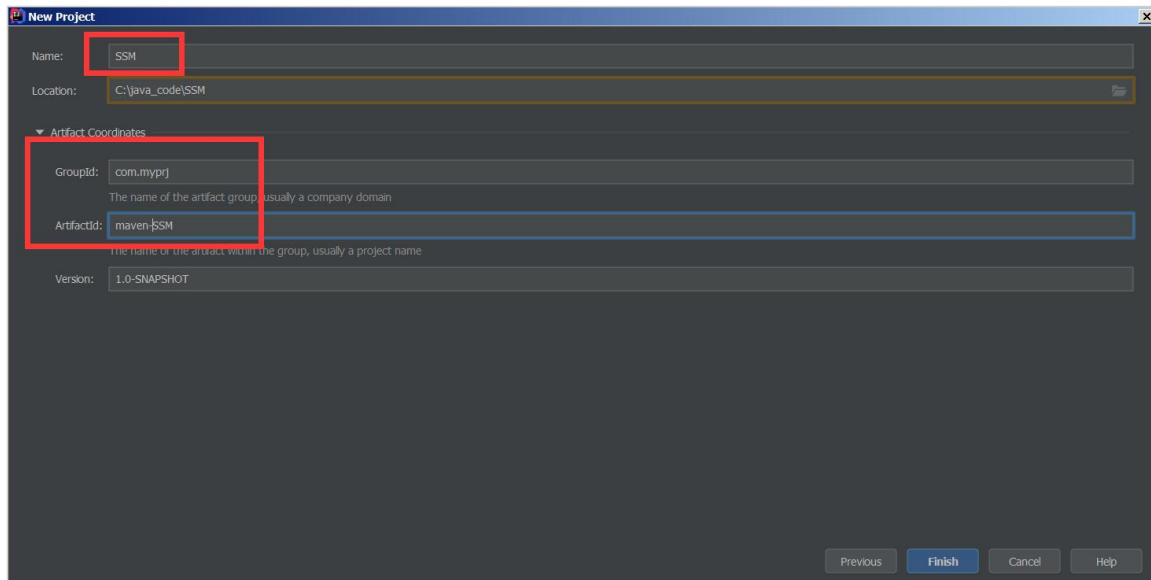
- 新建项目



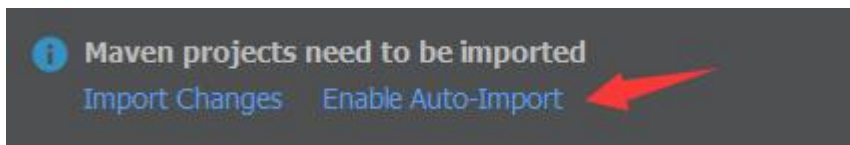
Name: SSM

GroupId: com.myprj

ArtifactId:maven-SSM



开启 Maven 工程自动导入功能



- 编写 pom.xml

```
<?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>

    <groupId>com.myprj</groupId>
    <artifactId>maven-SSM</artifactId>
    <version>1.0-SNAPSHOT</version>
    <packaging>war</packaging>

    <properties>
        <junit.version>4.12</junit.version>
        <spring.version>5.2.7.RELEASE</spring.version>
```

```
<mybatis.version>3.5.5</mybatis.version>
<mybatis.spring.version>2.0.5</mybatis.spring.version>
<!--
<mysql.version>8.0.20</mysql.version>
-->
<mysql.version>5.1.32</mysql.version>
<slf4j.version>1.6.4</slf4j.version>
<jstl.version>1.2</jstl.version>
<servlet-api.version>2.5</servlet-api.version>
<jsp-api.version>2.0</jsp-api.version>
<c3p0.version>0.9.5.2</c3p0.version>
</properties>
```

```
<dependencyManagement>
  <dependencies>
    <!-- 单元测试 -->
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
    <!-- 日志处理 -->
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-log4j12</artifactId>
      <version>${slf4j.version}</version>
    </dependency>
    <!-- Mybatis -->
    <dependency>
      <groupId>org.mybatis</groupId>
      <artifactId>mybatis</artifactId>
      <version>${mybatis.version}</version>
```

```
</dependency>
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis-spring</artifactId>
    <version>${mybatis.spring.version}</version>
</dependency>
<!-- MySql -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>${mysql.version}</version>
</dependency>
<!-- 数据库连接池 -->
<dependency>
    <groupId>com.mchange</groupId>
    <artifactId>c3p0</artifactId>
    <version>${c3p0.version}</version>
</dependency>

<!-- 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>${spring.version}</version>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-context-support</artifactId>
        <version>${spring.version}</version>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-test</artifactId>
        <version>${spring.version}</version>
    </dependency>
    <!-- JSP 相关 -->
    <dependency>
        <groupId>jstl</groupId>
        <artifactId>jstl</artifactId>
        <version>${jstl.version}</version>
    </dependency>
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>servlet-api</artifactId>
        <version>${servlet-api.version}</version>
        <scope>provided</scope>
```

```
        </dependency>
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>jsp-api</artifactId>
            <version>${jsp-api.version}</version>
            <scope>provided</scope>
        </dependency>
    </dependencies>
</dependencyManagement>
```

```
<dependencies>
    <!-- Spring -->
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-context</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-beans</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-jdbc</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-aspects</artifactId>
    </dependency>
    <dependency>
```

```
<groupId>org.springframework</groupId>
<artifactId>spring-context-support</artifactId>
</dependency>
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-test</artifactId>
</dependency>
<!-- JSP 相关 -->
<dependency>
    <groupId>jstl</groupId>
    <artifactId>jstl</artifactId>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>servlet-api</artifactId>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jsp-api</artifactId>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>com.mchange</groupId>
    <artifactId>c3p0</artifactId>
</dependency>
<!-- 单元测试 -->
<dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>${junit.version}</version>
    <scope>test</scope>
</dependency>
```

```
<!-- 日志处理 -->
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-log4j12</artifactId>
</dependency>
<!-- Mybatis -->
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis</artifactId>
</dependency>
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis-spring</artifactId>
</dependency>
<!-- MySql -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
</dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <groupId>org.apache.tomcat.maven</groupId>
            <artifactId>tomcat7-maven-plugin</artifactId>
            <configuration>
                <path>/</path>
                <port>8080</port>
            </configuration>
        </plugin>
    </plugins>
    <resources>
```



```
<resource>
  <directory>src/main/java</directory>
  <includes>
    <include>**/*.xml</include>
  </includes>
</resource>
<resource>
  <directory>src/main/resources</directory>
  <includes>
    <include>**/*.xml</include>
  </includes>
</resource>
</resources>
</build>
```

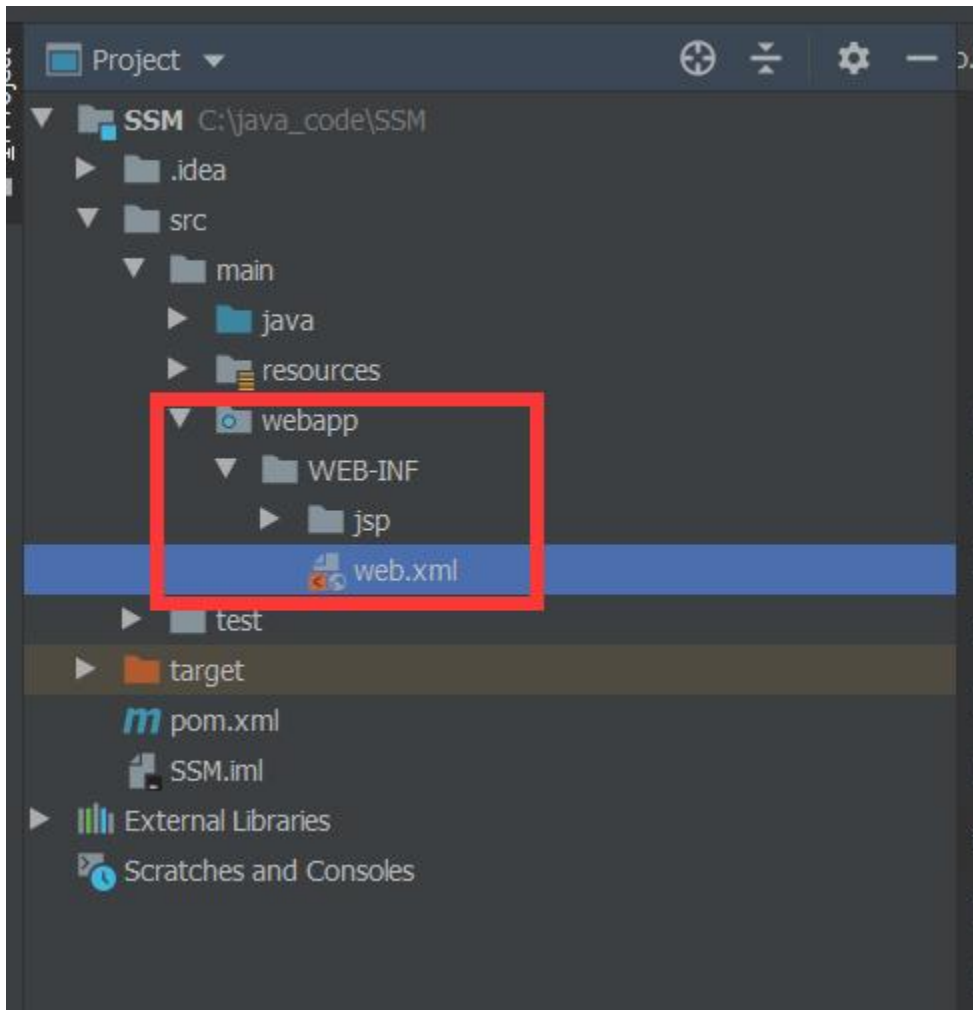
```
</project>
```

- 编写 web.xml

在 main 下创建 webapp 文件夹

在 webapp 下创建 WEB-INF

在 WEB-INF 下添加 web.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xmlns="http://java.sun.com/xml/ns/javaee"
         xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
         http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
         version="2.5">

    <display-name>maven-ssm</display-name>
    <welcome-file-list>
        <welcome-file>/</welcome-file>
    </welcome-file-list>
    <!-- post 乱码过滤器 -->
    <filter>
```

```
<filter-name>CharacterEncodingFilter</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>
</filter>
<filter-mapping>
    <filter-name>CharacterEncodingFilter</filter-name>
    <url-pattern>/*</url-pattern>
</filter-mapping>
<!-- 前端控制器 -->
<servlet>
    <servlet-name>ssmServlet</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
    <!--
contextConfigLocation 不是必须的， 如果不配置 contextConfigLocation,
springmvc 的配置文件默认在: WEB-INF/servlet 的 name+"-servlet.xml"
-->
    <init-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>classpath:applicationContext.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
    <servlet-name>ssmServlet</servlet-name>
    <url-pattern>/</url-pattern>
</servlet-mapping>
</web-app>
```

- 实体 com.mypj.entity.News

```
package com.mypj.entity;

import java.util.Date;

public class News {
    private Integer id;
    private String title;
    private String content;
    private Date createTime;

    public Integer getId() {
        return id;
    }

    public void setId(Integer id) {
        this.id = id;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content) {
        this.content = content;
    }
}
```

```

    }

    public Date getCreateTime() {
        return createTime;
    }

    public void setCreateTime(Date createTime) {
        this.createTime = createTime;
    }
}

```

- DAO 接口 com.myprj.dao.NewsDao

```

package com.myprj.dao;

import com.myprj.entity.News;

import java.util.List;

public interface NewsDao {
    public List<News> getAll();
}

```

- 添加 com/myprj/dao/mapper/NewsMapper.xml

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE mapper
    PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
    "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
<!--namespace:dao 接口路径-->
<mapper namespace="com.myprj.dao.NewsDao">
    <select id="getAll" resultType="News">
        select * from news
    </select>

```

```
</select>
</mapper>
```

- 添加 service 接口及其实现类

com.myprj.service.NewsService

```
package com.myprj.service;

import com.myprj.entity.News;

import java.util.List;

public interface NewsService {
    public List<News> getAll();
}
```

com.myprj.service.impl.NewsServiceImpl

```
package com.myprj.service.impl;

import com.myprj.dao.NewsDao;
import com.myprj.entity.News;
import com.myprj.service.NewsService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;

@Service
public class NewsServiceImpl implements NewsService {
    @Autowired
    private NewsDao newsDao;

    public List<News> getAll() {
```

```
        return newsDao.getAll();
    }
}
```

- 添加 Controller

com.myprj.controller.NewsController

```
package com.myprj.controller;

import com.myprj.entity.News;
import com.myprj.service.NewsService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;

import java.util.List;

@Controller
public class NewsController {
    @Autowired
    private NewsService newsService;

    @RequestMapping("/")
    public String index(Model model){
        List<News> newsList = newsService.getAll();
        System.out.println(newsList.size());
        model.addAttribute("newsList",newsList);
        return "index";
    }
}
```

- 添加 jsp

在 WEB-INF 下创建 jsp 文件夹

在 jsp 文件夹下新建 index.jsp

```
<%--
    Created by IntelliJ IDEA.
    User: Administrator
    Date: 2020/7/5 0005
    Time: 15:08
    To change this template use File | Settings | File Templates.
--%>
<%@ page contentType="text/html;charset=UTF-8" language="java" %>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<html>
<head>
    <title>Title</title>
</head>
<body>
<h1>
    新闻列表:
</h1>
<ul>
<c:forEach items="${requestScope.newsList}" var="news">
    <li>${news.title}----<fmt:formatDate
                                value="${news.createTime}"
                                pattern="yyyy-MM-dd"/></li>
</c:forEach>
</ul>
</body>
</html>
```

- 在 resources 中添加配置

applicationContext-dao.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-4.3.xsd
http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context-4.3.xsd">

    <context:component-scan base-package="com.myprj.dao" />

    <!-- 数据库连接池 -->
    <bean id="dataSource" class="com.mchange.v2.c3p0.ComboPooledDataSource">
        <!--
        <property name="driverClass" value="com.mysql.cj.jdbc.Driver"></property>
        <property name="jdbcUrl" value="jdbc:mysql://localhost:3306/shoppingstreet?useSSL=false&serverTimezone=UTC"></property>
        -->
        <property name="driverClass" value="com.mysql.jdbc.Driver"></property>
        <property name="jdbcUrl" value="jdbc:mysql://localhost:3306/shoppingstreet"></property>

        <property name="user" value="root"></property>
        <property name="password" value="root"></property>
        <!--连接池启动的时候默认创建的连接数量 -->
        <property name="initialPoolSize" value="3"></property>
        <!--连接池最多可以管理的连接对象个数 -->
        <property name="maxPoolSize" value="100"></property>
        <!--连接池中最多能够管理的 statement 对象 -->
        <property name="maxStatements" value="1000"></property>
        <!--一旦连接池中现有的连接数量不够，每次增长的连接数目：5，但是连接池中的连
```

接数量 -->

<!--最多不可超过 maxPoolSize 中设置的连接数目 -->

<property name="acquireIncrement" value="5"></property>

</bean>

<!-- SqlSessionFactory -->

<!-- 让 spring 管理 sqlSessionFactory 使用 mybatis 和 spring 整合包中的 -->

<bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">

<!-- 数据库连接池 -->

<property name="dataSource" ref="dataSource" />

<property name="typeAliasesPackage" value="com.myprj.entity" />

<!-- 加载 mybatis 的全局配置文件 -->

<property name="mapperLocations" value="classpath*:com/myprj/dao/**/*.xml" />

</bean>

<bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">

<!--指定 dao 接口的位置 -->

<property name="basePackage" value="com.myprj.dao" />

<property name="sqlSessionFactoryBeanName"

value="sqlSessionFactory"></property>

</bean>

</beans>

applicationContext-trans.xml

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:tx="http://www.springframework.org/schema/tx"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-4.3.xsd

http://www.springframework.org/schema/aop

<http://www.springframework.org/schema/aop/spring-aop-4.3.xsd>

<http://www.springframework.org/schema/tx>

<http://www.springframework.org/schema/tx/spring-tx-4.3.xsd>

">

<!-- 事务管理器 -->

<bean id="transactionManager"

class="org.springframework.jdbc.datasource.DataSourceTransactionManager">

<!-- 数据源 -->

<property name="dataSource" ref="dataSource" />

</bean>

<!-- 通知 -->

<tx:advice id="txAdvice" transaction-manager="transactionManager">

<tx:attributes>

<!-- 传播行为 -->

<tx:method name="save*" propagation="REQUIRED" />

<tx:method name="insert*" propagation="REQUIRED" />

<tx:method name="add*" propagation="REQUIRED" />

<tx:method name="create*" propagation="REQUIRED" />

<tx:method name="delete*" propagation="REQUIRED" />

<tx:method name="update*" propagation="REQUIRED" />

<tx:method name="find*" propagation="SUPPORTS" read-only="true" />

<tx:method name="select*" propagation="SUPPORTS" read-only="true" />

<tx:method name="get*" propagation="SUPPORTS" read-only="true" />

</tx:attributes>

</tx:advice>

<!-- 切面 -->

<aop:config>

<aop:advisor advice-ref="txAdvice"

pointcut="execution(* com.myprj.service.*(..))" />

</aop:config>

</beans>

applicationContext-service.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
       http://www.springframework.org/schema/beans/spring-beans-4.3.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context-4.3.xsd"
       ">

    <context:component-scan base-package="com.mypj.service"/>
</beans>
```

applicationContext-mvc.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:p="http://www.springframework.org/schema/p"
       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-4.3.xsd
       http://www.springframework.org/schema/mvc
       http://www.springframework.org/schema/mvc/spring-mvc-4.3.xsd
       http://www.springframework.org/schema/context
       http://www.springframework.org/schema/context/spring-context-4.3.xsd">

    <!-- 配置注解驱动 -->
    <mvc:annotation-driven />
```

```

<!-- 视图解析器 -->
<bean
    class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/jsp/" />
    <property name="suffix" value=".jsp" />
</bean>
<!-- 配置包扫描器，扫描@Controller 注解的类 -->
<context:component-scan base-package="com.myprj.controller"/>
<!-- 配置资源映射 -->
<mvc:resources location="/css/" mapping="/css/**"/>
<mvc:resources location="/js/" mapping="/js/**"/>

</beans>

```

applicationContext.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"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-4.3.xsd">

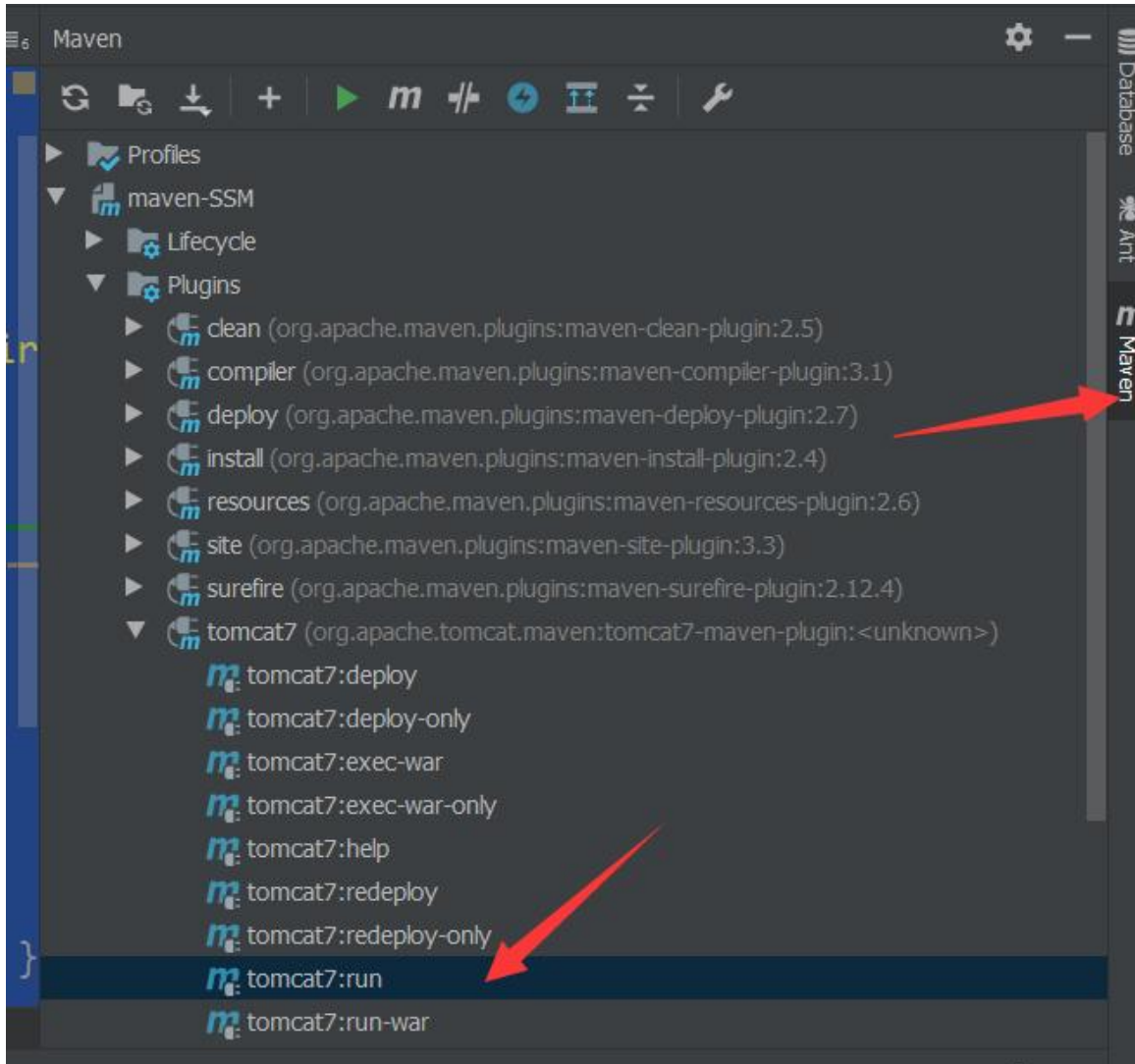
    <import resource="applicationContext-dao.xml"/>
    <import resource="applicationContext-mvc.xml"/>
    <import resource="applicationContext-service.xml"/>
    <import resource="applicationContext-trans.xml"/>

</beans>

```

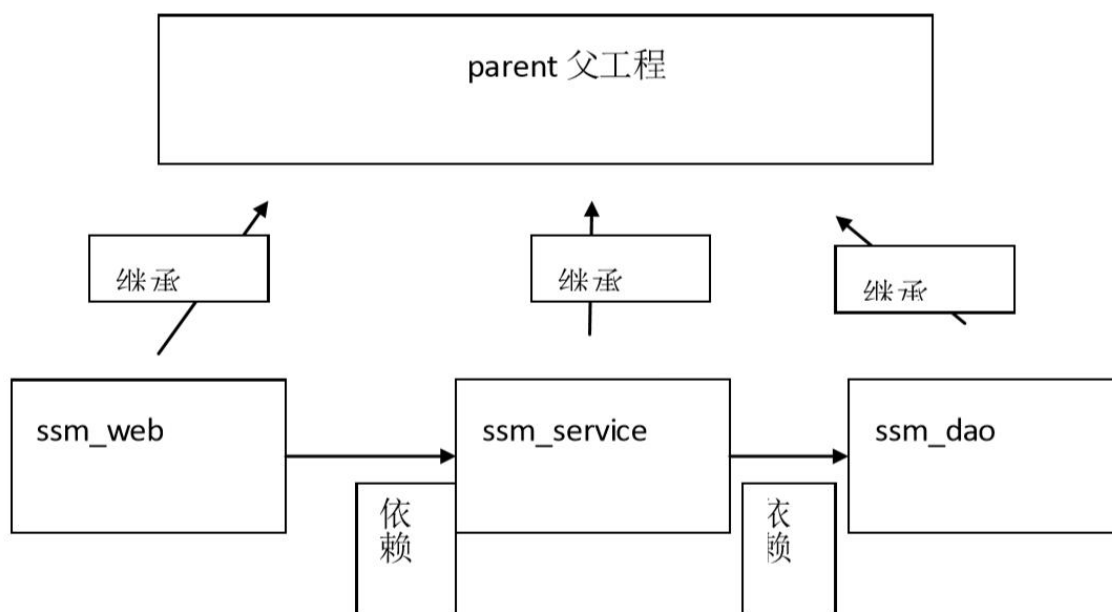
- 测试运行:

双击 tomcat7:run



二、分模块构建

1. 理解继承和聚合



何为继承？

继承是为了消除重复，如果将 `dao`、`service`、`web` 分开创建独立的工程则每个工程的 `pom.xml` 文件中的内容存在重复，比如：设置编译版本、锁定 `spring` 的版本等等，可以将这些重复的配置提取出来在父工程的 `pom.xml` 中定义。

何为聚合？

项目开发通常是分组分模块开发，每个模块开发完成要运行整个工程需要将每个模块聚合在一起运行，比如：`dao`、`service`、`web` 三个工程最终会打一个独立的 `war` 运行。

2. 案例实现

① maven-parent 父模块

- 创建父工程

name:ssm2

groupId:com.myprj

artifactId:maven-SSM2

- 定义 pom.xml

在父工程的 pom.xml 中抽取一些重复的配置的，比如：锁定 jar 包的版本、设置编译版本等。

打包方式设置为 pom

```
<?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>

    <groupId>com.myprj</groupId>
    <artifactId>maven-SSM2</artifactId>
    <version>1.0-SNAPSHOT</version>
    <modules>
        <module>ssm-dao</module>
        <module>ssm-service</module>
        <module>ssm-web</module>
    </modules>
    <packaging>pom</packaging>

    <properties>
        <junit.version>4.12</junit.version>
        <spring.version>5.2.7.RELEASE</spring.version>
        <mybatis.version>3.5.5</mybatis.version>
```



```
<mybatis.spring.version>2.0.5</mybatis.spring.version>
<!--
<mysql.version>8.0.20</mysql.version>
-->
<mysql.version>5.1.32</mysql.version>
<slf4j.version>1.6.4</slf4j.version>
<jstl.version>1.2</jstl.version>
<servlet-api.version>2.5</servlet-api.version>
<jsp-api.version>2.0</jsp-api.version>
<c3p0.version>0.9.5.2</c3p0.version>
</properties>
```

```
<dependencyManagement>
  <dependencies>
    <!-- 单元测试 -->
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
    <!-- 日志处理 -->
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-log4j12</artifactId>
      <version>${slf4j.version}</version>
    </dependency>
    <!-- Mybatis -->
    <dependency>
      <groupId>org.mybatis</groupId>
      <artifactId>mybatis</artifactId>
      <version>${mybatis.version}</version>
    </dependency>
```

```
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis-spring</artifactId>
    <version>${mybatis.spring.version}</version>
</dependency>
<!-- MySql -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>${mysql.version}</version>
</dependency>
<!-- 数据库连接池 -->
<dependency>
    <groupId>com.mchange</groupId>
    <artifactId>c3p0</artifactId>
    <version>${c3p0.version}</version>
</dependency>

<!-- 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-context-support</artifactId>
    <version>${spring.version}</version>
</dependency>
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-test</artifactId>
    <version>${spring.version}</version>
</dependency>
<!-- JSP 相关 -->
<dependency>
    <groupId>jstl</groupId>
    <artifactId>jstl</artifactId>
    <version>${jstl.version}</version>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>servlet-api</artifactId>
    <version>${servlet-api.version}</version>
    <scope>provided</scope>
</dependency>
```

```
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>jsp-api</artifactId>
            <version>${jsp-api.version}</version>
            <scope>provided</scope>
        </dependency>
    </dependencies>
</dependencyManagement>
```

```
<dependencies>
    <!-- Spring -->
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-context</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-beans</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-jdbc</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-aspects</artifactId>
    </dependency>
    <dependency>
        <groupId>org.springframework</groupId>
```

```
<artifactId>spring-context-support</artifactId>
</dependency>
<dependency>
    <groupId>org.springframework</groupId>
    <artifactId>spring-test</artifactId>
</dependency>
<!-- JSP 相关 -->
<dependency>
    <groupId>jstl</groupId>
    <artifactId>jstl</artifactId>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>servlet-api</artifactId>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jsp-api</artifactId>
    <scope>provided</scope>
</dependency>
<dependency>
    <groupId>com.mchange</groupId>
    <artifactId>c3p0</artifactId>
</dependency>
<!-- 单元测试 -->
<dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>${junit.version}</version>
    <scope>test</scope>
</dependency>
<!-- 日志处理 -->
```

```
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-log4j12</artifactId>
</dependency>
<!-- Mybatis -->
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis</artifactId>
</dependency>
<dependency>
    <groupId>org.mybatis</groupId>
    <artifactId>mybatis-spring</artifactId>
</dependency>
<!-- MySql -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
</dependency>
</dependencies>

<build>
    <plugins>
        <plugin>
            <groupId>org.apache.tomcat.maven</groupId>
            <artifactId>tomcat7-maven-plugin</artifactId>
            <configuration>
                <path></path>
                <port>8080</port>
            </configuration>
        </plugin>
    </plugins>
</build>
```

```
</project>
```

② ssm_dao 子模块

- 创建 dao 子模块

在父工程上右击创建 maven 模块:

填写模块名称 ssm_dao

打包方式是 jar

- 定义 pom.xml

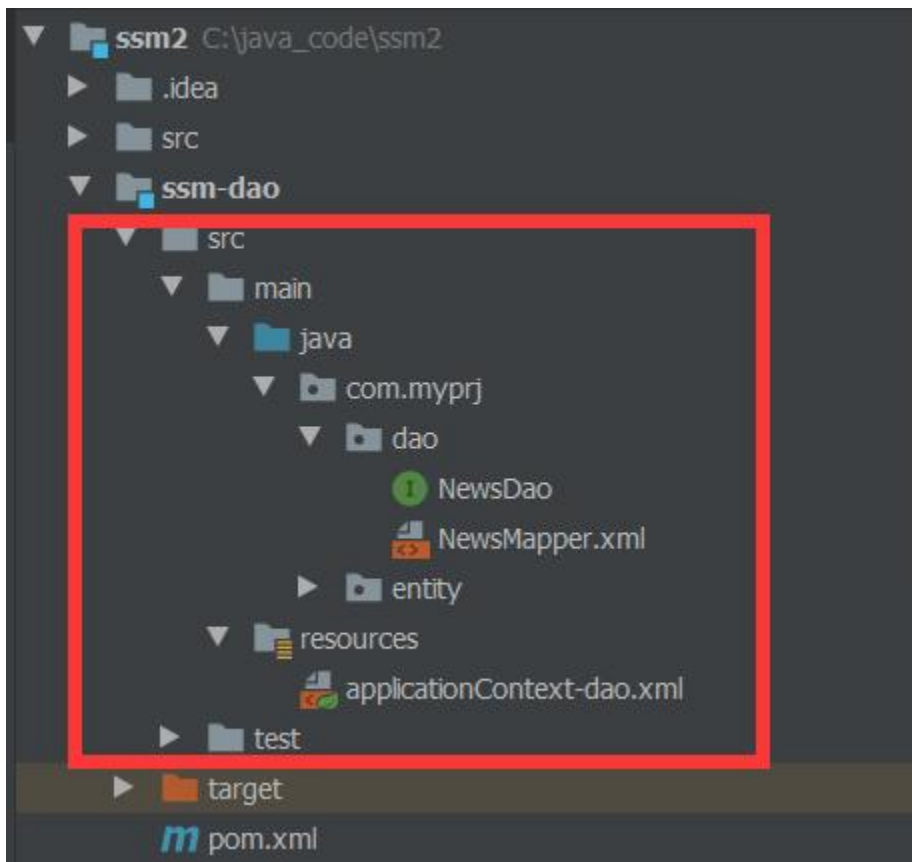
```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <parent>
        <artifactId>maven-SSM2</artifactId>
        <groupId>com.myprj</groupId>
        <version>1.0-SNAPSHOT</version>
    </parent>
    <modelVersion>4.0.0</modelVersion>

    <artifactId>ssm-dao</artifactId>

    <build>
        <resources>
            <resource>
                <directory>src/main/java</directory>
                <includes>
                    <include>**/*.xml</include>
                </includes>
            </resource>
            <resource>
```

```
<directory>src/main/resources</directory>
<includes>
  <include>**/*.xml</include>
</includes>
</resource>
</resources>
</build>
</project>
```

- 代码与配置



③ ssm_service 子模块

- 创建 service 子模块

方法同 ssm_dao 模块创建方法，模块名称为 ssm_service。

- 定义 pom.xml

ssm_service 模块的 pom.xml 文件中需要继承父模块, ssm_service 依赖 ssm_dao 模块, 添加

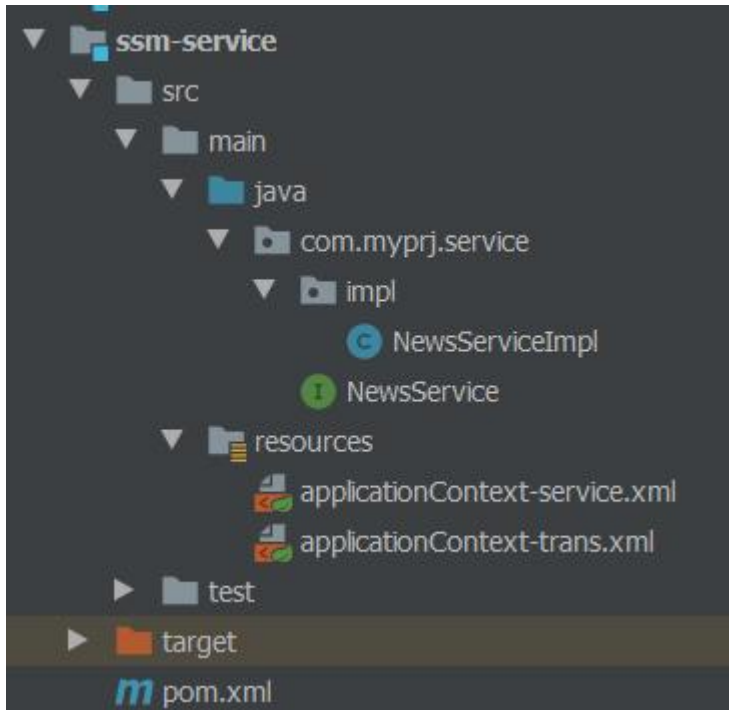
spring 相关的依赖:

```
<?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">
    <parent>
        <artifactId>maven-SSM2</artifactId>
        <groupId>com.myprj</groupId>
        <version>1.0-SNAPSHOT</version>
    </parent>
    <modelVersion>4.0.0</modelVersion>

    <artifactId>ssm-service</artifactId>

    <dependencies>
        <dependency>
            <artifactId>ssm-dao</artifactId>
            <groupId>com.myprj</groupId>
            <version>1.0-SNAPSHOT</version>
        </dependency>
    </dependencies>
</project>
```

- 代码及配置



④ ssm_web 子模块

- 创建 web 子模块

添加打包方式 war

- 定义 pom.xml

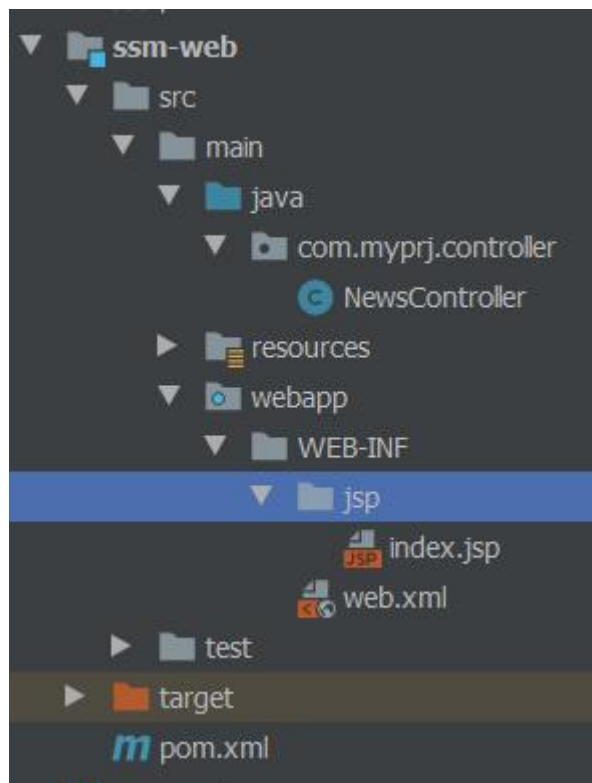
```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <parent>
        <artifactId>maven-SSM2</artifactId>
        <groupId>com.myprj</groupId>
        <version>1.0-SNAPSHOT</version>
    </parent>
    <modelVersion>4.0.0</modelVersion>

    <artifactId>ssm-web</artifactId>
```

```
<packaging>war</packaging>

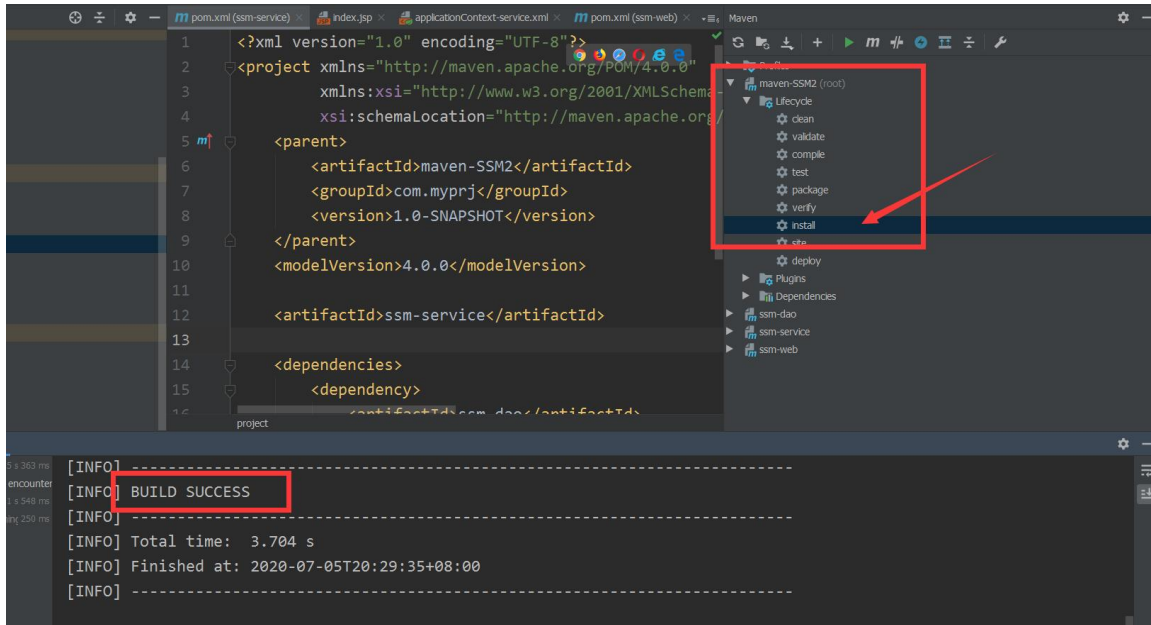
<dependencies>
    <dependency>
        <artifactId>ssm-service</artifactId>
        <groupId>com.myprj</groupId>
        <version>1.0-SNAPSHOT</version>
    </dependency>
</dependencies>
</project>
```

- 代码与配置文件



⑤ 将父工程发布至仓库

父工程执行 `install` 将父工程发布到仓库方便子工程继承:



```
<groupId>com.myprj</groupId>
<artifactId>maven-SSM2</artifactId>
<version>1.0-SNAPSHOT</version>
<modules>
  <module>ssm-dao</module>
  <module>ssm-service</module>
  <module>ssm-web</module>
</modules>
<packaging>pom</packaging>
```

```
[INFO] maven-SSM2 ..... SUCCESS [ 11.194 s]
[INFO] ssm-dao ..... SUCCESS [ 2.070 s]
[INFO] ssm-service ..... SUCCESS [ 0.228 s]
[INFO] ssm-web ..... SUCCESS [ 1.342 s]
[INFO] -----
[INFO] BUILD SUCCESS
```

⑥ 测试运行

双击 tomcat7:run

