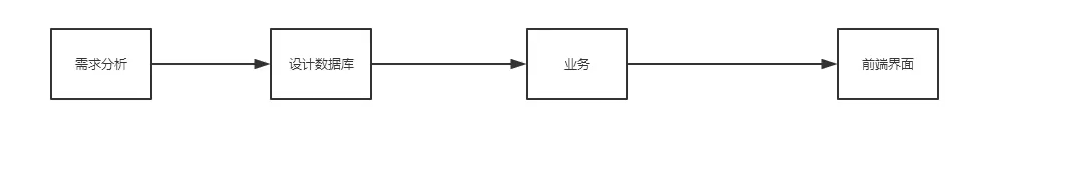
**ssm整合**

步骤：需求分析-->设计数据库-->业务-->前端界面



数据分析完毕后，接下来：

1.首先先在pom文件里面导入依赖：

**依赖: junit，数据库驱动，连接池，servlet，jsp，mybatis， mybattis-spring，spring**

注意，如果我们的mybatis是3.5以上spring是5.0以上，mybatis-spring要用2.0以上。

配置相关问题:

**静态资源导出问题。**

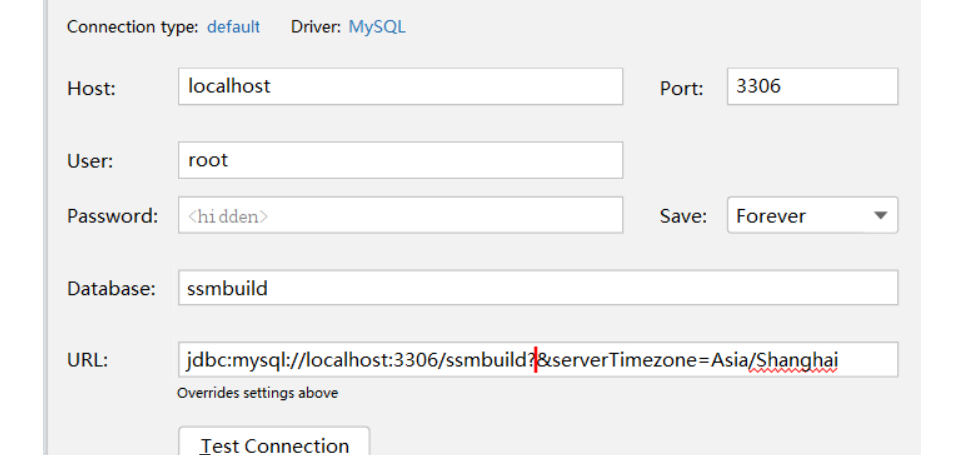
**2.建立数据库：**



然后建立基本结构和配置框架：

即mybatis的配置文件mybatis-config.xml还有applicationContext.xml的配置文件：

！如果使用的是MysQL8.0+，增加一个时区的配置;&serverTimezone=Asia/Shanghai



然后开始写mybatis层：

mybatis层要干的事情不多具体可以分为三块：

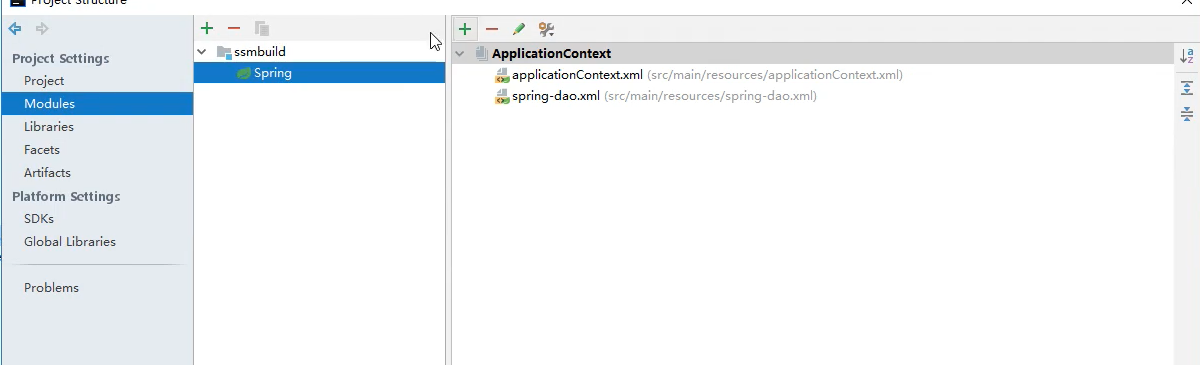
第一块：创建我们需要的类，先创建pojo类---->dao层建立Mapper类和对应的mapper的xml配置文件--->创建service层接口和对应的service层实现类去实现接口调用功能

第二块：这里主要是配置文件的书写，先写上mybatis-config.xml使用typeAliases给我们的mapper类开启重命名--->用mapper绑定上我们的dao层mapper的xml配置文件

第三块：编写我们spring层需要用到的数据库配置文件database.properties，编写applicationContext.xml，这个文件里面可以用import导入我需要的配置文件

案例：<https://blog.csdn.net/qq_42388853/article/details/108577013>

spring层：注意，把spring的配置文件放到这里。



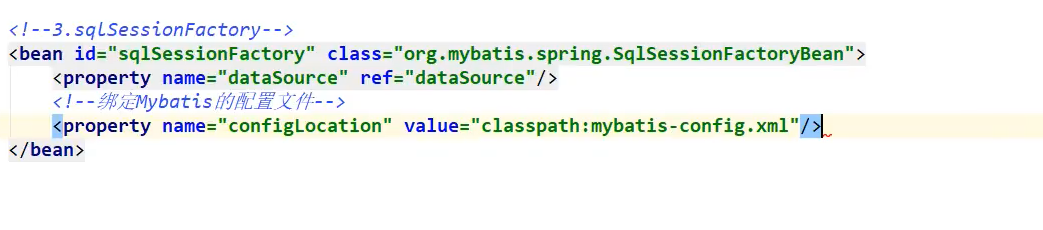
spring-dao.xml用来做关联数据库等操作：注意jdbc数据库连接池这块用的是${}



通过sqlsessionFactory绑定我们的mybatis文件：

干了两件事：引用我们的连接池资源，第二件事绑定我们的mybatis配置文件

（在mybatis里我们需要getsqlsession，这边用这个代替）

配置dao接口扫描包：代替实现类使用了sqlsession，（这里是给我们的mapper绑定上sqlsession，代替我们的getmapper）

同样干了两件事：把sqlsession丢到spring容器里

扫描我们的包，给我们的mapper注入sqlsession



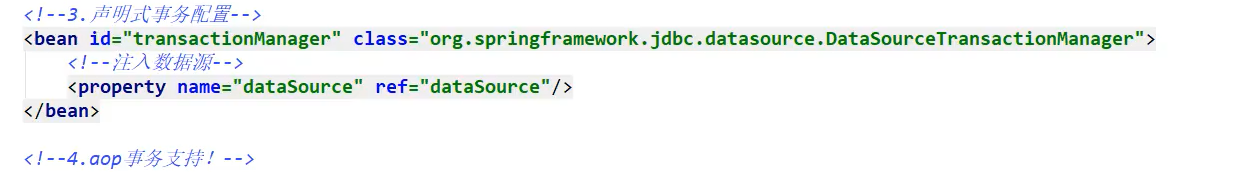
案例：<https://blog.csdn.net/qq_42388853/article/details/108582210>

**spring-service.xml**

注入spring：







步骤1，创建spring-dao.xml文件：关联数据库文件--->创建数据库连接池在里面导入关联数据库文件的各个信息---->配置SqlSessionFactory对象--->扫描我们的dao接口包，把我们创建的sqlSessionFactory注入到dao层。

步骤2，创建spring-service.xml：开启注解扫描--->使用注解或者配置的方式将业务层使用到的业务类注入--->开启事务管理--->aop事务支持

记得将spring-service.xml用import导入我们的applicationContext.xml

案例：<https://blog.csdn.net/qq_42388853/article/details/108582210>

Spring-mvc层：

先右键项目把普通的maven项目转化为web项目，然后接下来的工作也具体分为两步步！

第一步：先到web下的web.xml去配置我们的前置控制器DispatcherServlet---> 配置乱码过滤filter --->配置session的相关配置如session-timeout

第二步：编写spring-mvc.xml文件--->注解驱动--->静态资源过滤

--->扫描包: controlLer--->配置视图解析器

记得将spring-mvc.xml用import导入我们的applicationContext.xml

第三步：去我们配置视图解析器对应的文件下创建文件--->然后去control对应的文件夹里编写文件跳转到视图解析器对应的文件下就可以了

案例：<https://blog.csdn.net/qq_42388853/article/details/108588557>

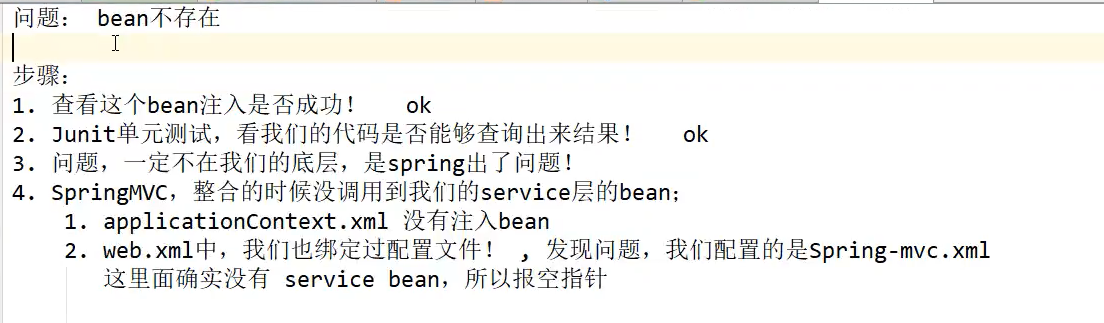
功能的实现：

查询书籍功能

注意！！Jsp的链接跳转尽量取绝对路径：

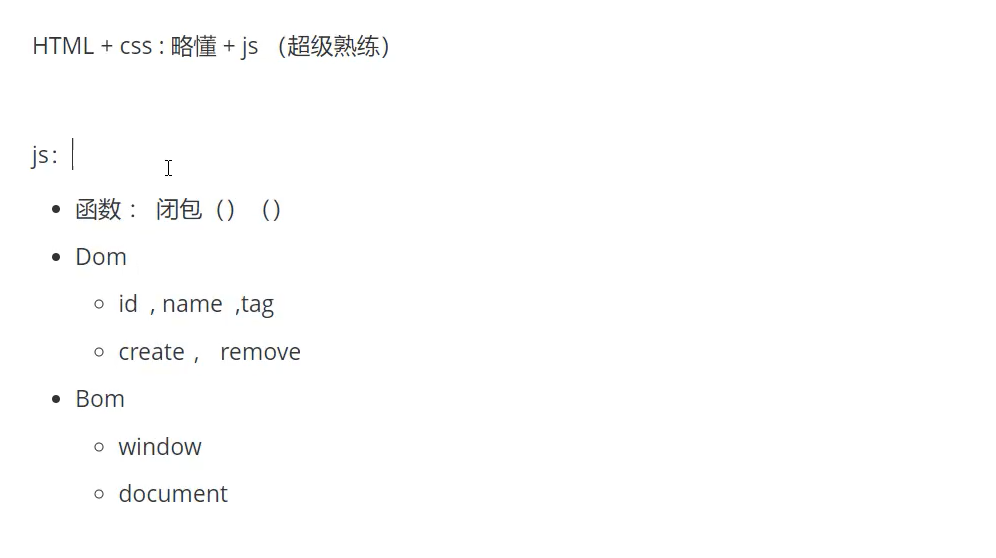


出现问题：问题的解决思路

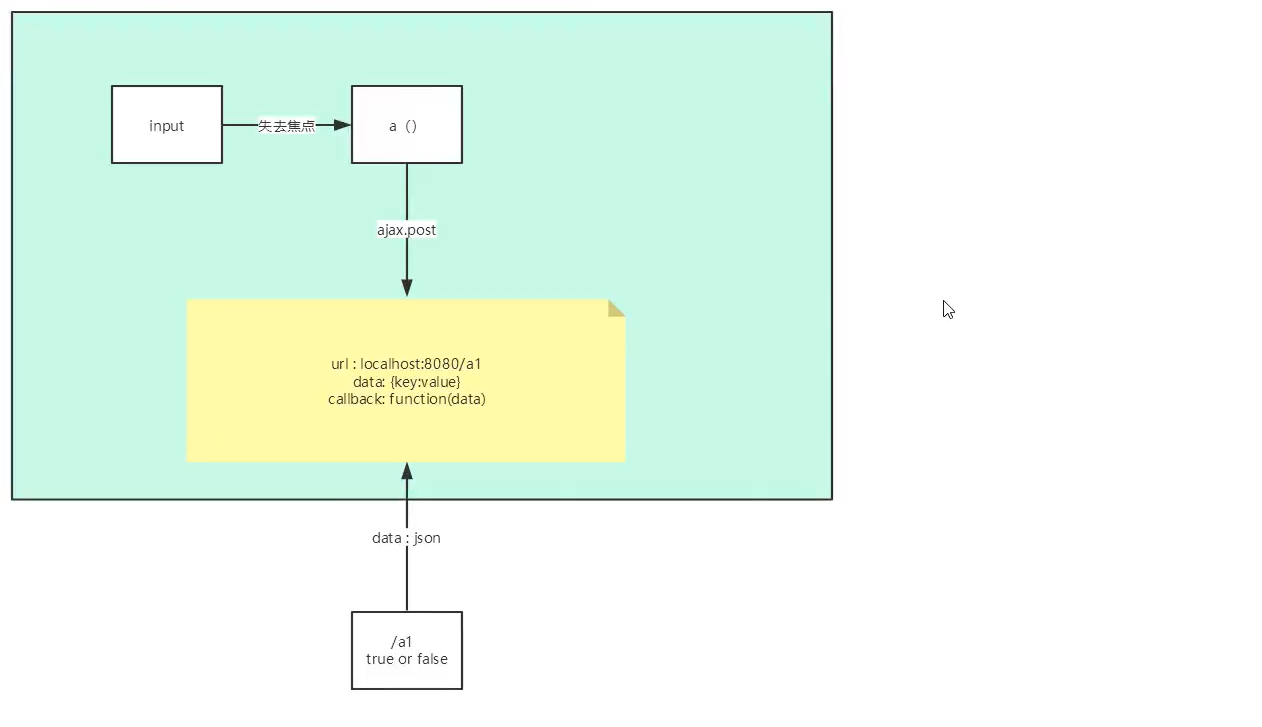


Ajax初体验：

前端需要掌握的事：



图解ajax干的事：青色部分是前端干的事。



引入方法：

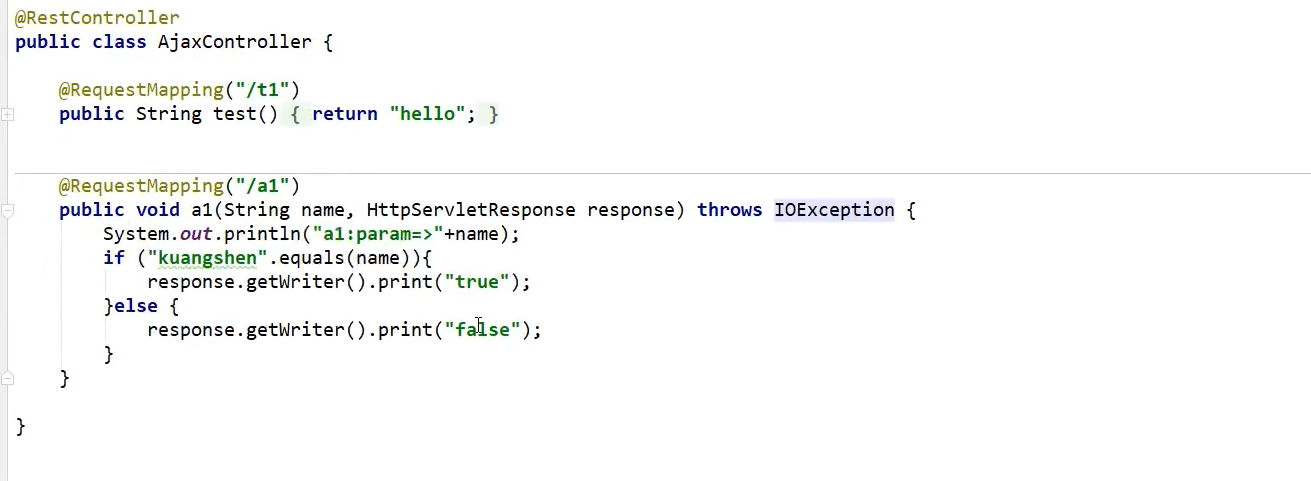
<script src="${pageContext.request.contextPath}/js/jquery-3.1.1.js"></script>

例子：

前端写ajax请求

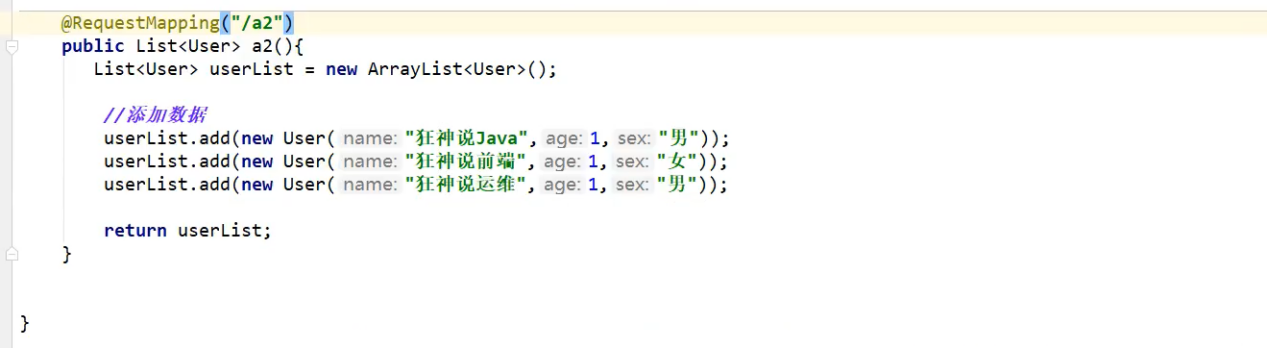


后端提交json：



显示学生数据：





SSM复习思路：



