**作业一 自己总结的SSM知识点**，github仓库地址：

https://github.com/EhHover/SSM.git

## **一、Spring部分**

Spring 是一个轻量级的开源框架，是为解决企业应用开发的复杂性而创建的。而在我的理解中，Spring 的主要就解决了两件事情（当然它还解决了数据访问、远程调用、单元测试等问题），分别对应 Spring 的两个设计思想 IOC 和 AOP：

IOC 容器（解耦合）：解决各种 new 对象的问题

AOP（切面编程）：把非业务范畴的功能,提取成一个切面，统一实现

Spring由7个模块组成：

Spring Core: 核心容器提供 Spring 框架的基本功能。核心容器的主要组件是BeanFactory，它是工厂模式的实现。BeanFactory 使用控制反转 （IOC） 模式将应用程序的配置和依赖性规范与实际的应用程序代码分开。

Spring 上下文：Spring 上下文是一个配置文件，向 Spring 框架提供上下文信息。Spring 上下文包括企业服务，例如 JNDI、EJB、电子邮件、国际化、校验和调度功能。

Spring AOP：通过配置管理特性，Spring AOP 模块直接将面向方面的编程功能集成到了 Spring 框架中。所以，可以很容易地使 Spring 框架管理的任何对象支持 AOP。Spring AOP 模块为基于 Spring 的应用程序中的对象提供了事务管理服务。通过使用 Spring AOP，不用依赖 EJB 组件，就可以将声明性事务管理集成到应用程序中。

Spring DAO：JDBC DAO 抽象层提供了有意义的异常层次结构，可用该结构来管理异常处理和不同数据库供应商抛出的错误消息。异常层次结构简化了错误处理，并且极大地降低了需要编写的异常代码数量（例如打开和关闭连接）。Spring DAO 的面向 JDBC 的异常遵从通用的 DAO 异常层次结构。

Spring ORM：Spring 框架插入了若干个 ORM 框架，从而提供了 ORM 的对象关系工具，其中包括 JDO、Hibernate 和 iBatis SQL Map。所有这些都遵从 Spring 的通用事务和 DAO 异常层次结构。

Spring Web 模块：Web 上下文模块建立在应用程序上下文模块之上，为基于 Web 的应用程序提供了上下文。所以，Spring 框架支持与 Jakarta Struts 的集成。Web 模块还简化了处理多部分请求以及将请求参数绑定到域对象的工作。

Spring MVC 框架：MVC 框架是一个全功能的构建 Web 应用程序的 MVC 实现。通过策略接口，MVC 框架变成为高度可配置的，MVC 容纳了大量视图技术，其中包括 JSP、Velocity、Tiles、iText 和 POI。



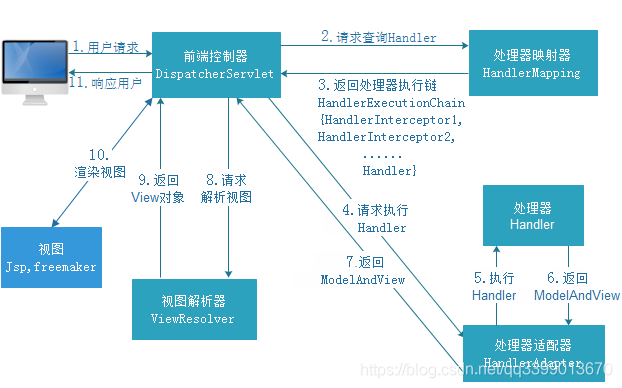
二、SpringMVC部分

1. SpringMVC的含义

概念：SpringMVC是Spring框架中的一个分支，是基于Java实现MVC的轻量级Web框架。

核心：Spring的web框架围绕DispatcherServlet [ 调度Servlet ] 设计的。

总结： SpringMVC 本质上还是在使用Servlet处理，并在其基础上进行了封装简化了开发流程，提高易用性、并使用程序逻辑结构变得更清晰



2.SpringMVC常见注解

@RequestMapping 处理常见的 HTTP 请求类型

5 种常见的请求类型:

GET ：请求从服务器获取特定资源。

POST ：在服务器上创建一个新的资源。

PUT ：更新服务器上的资源（客户端提供更新后的整个资源）。DELETE ：从服务器删除特定的资源。

PATCH ：更新服务器上的资源（客户端提供更改的属性，可以看做作是部分更新），使用的比较少。

三、Mybatis部分

1. Mybatis的含义

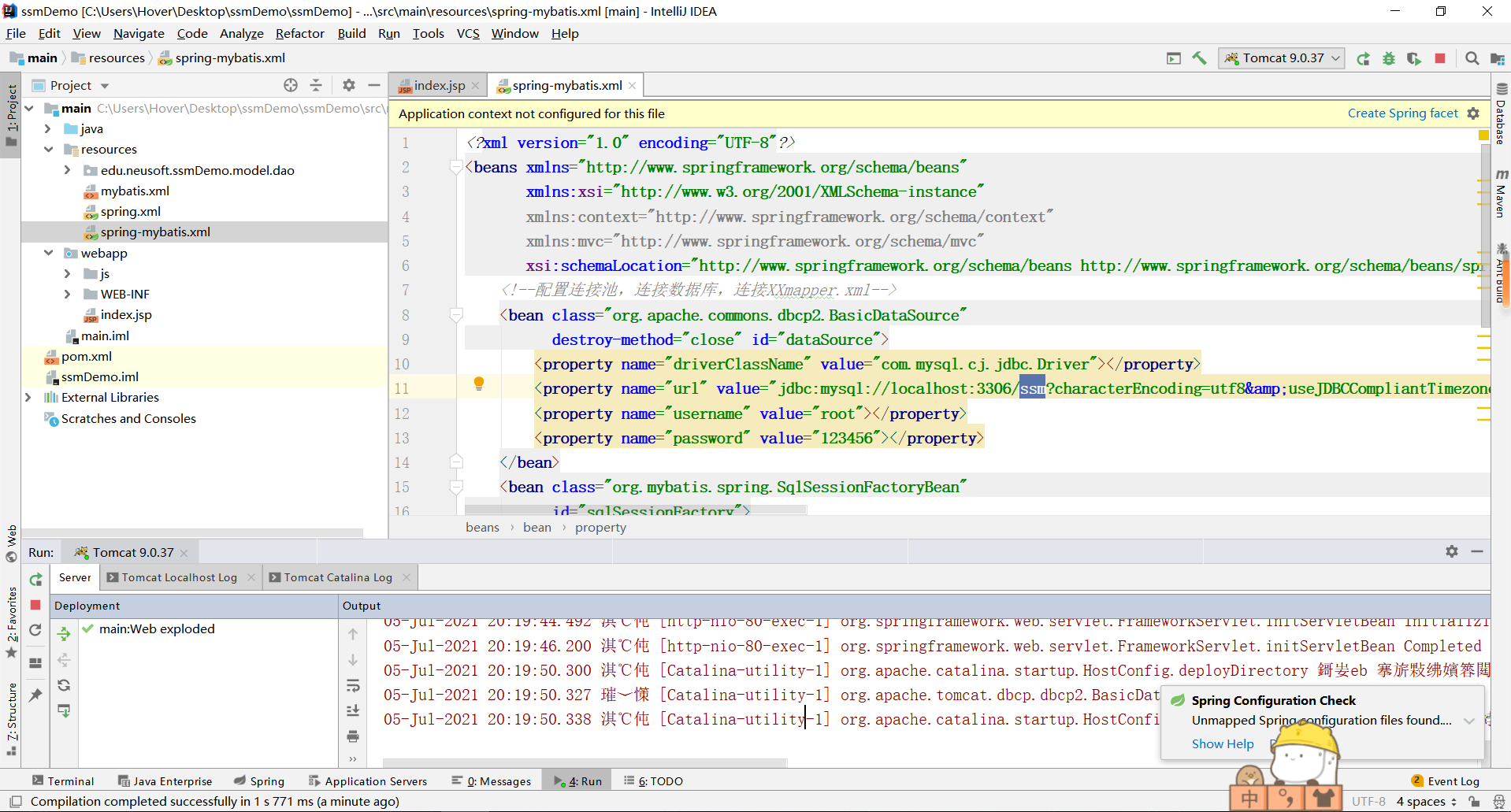
MyBatis是一种半自动化的Java持久层框架，其通过注解或XML 的方式将对象和SQL 关联起来。之所以说它是半自动的，是因为和Hibernate 等一些可自动生成SQL的ORM框架相比，使用MyBatis需要用户自行维护SQL，维护SQL 的工作比较繁琐，但也有好处。

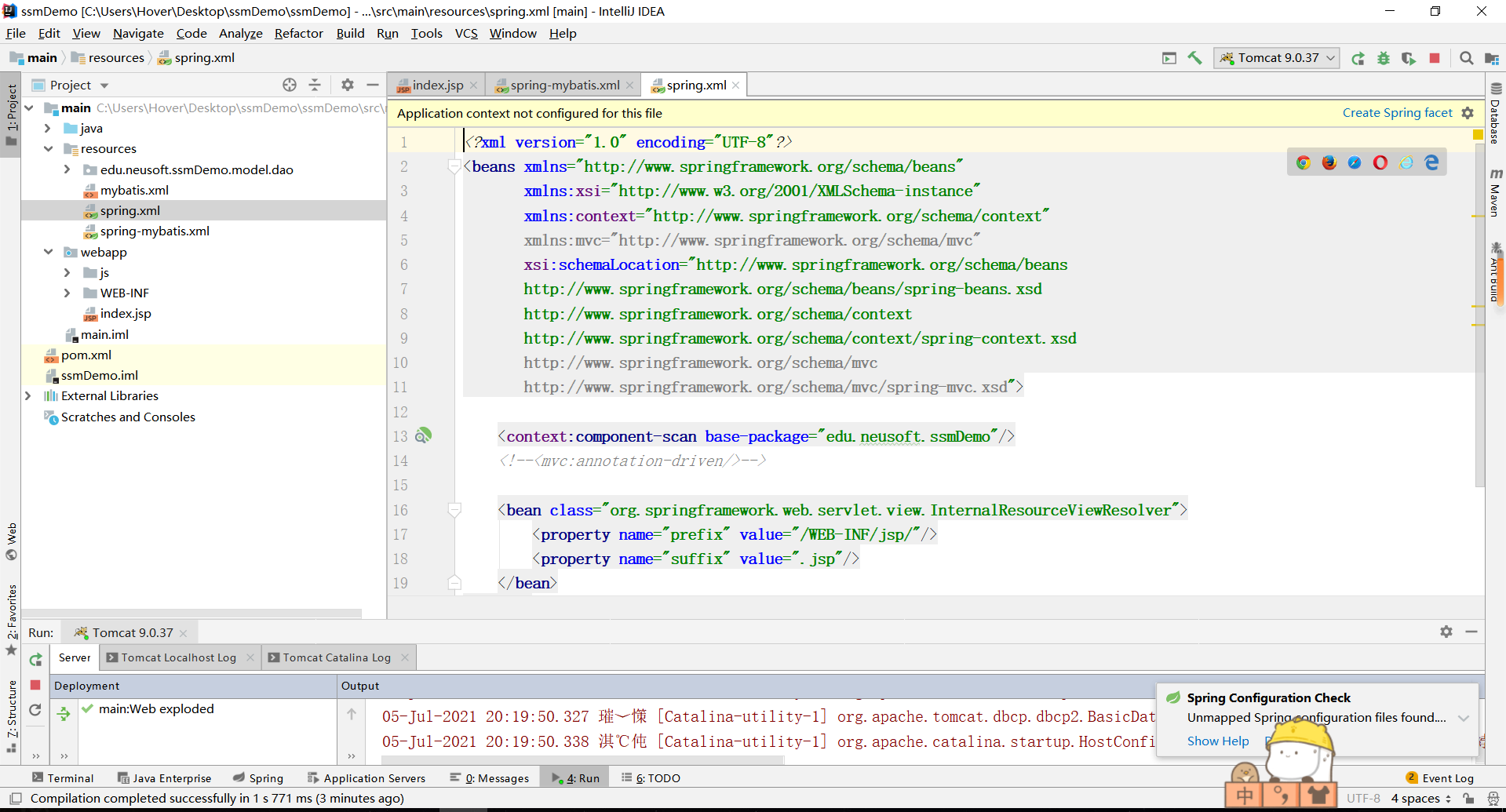
2.Mybatis工作原理

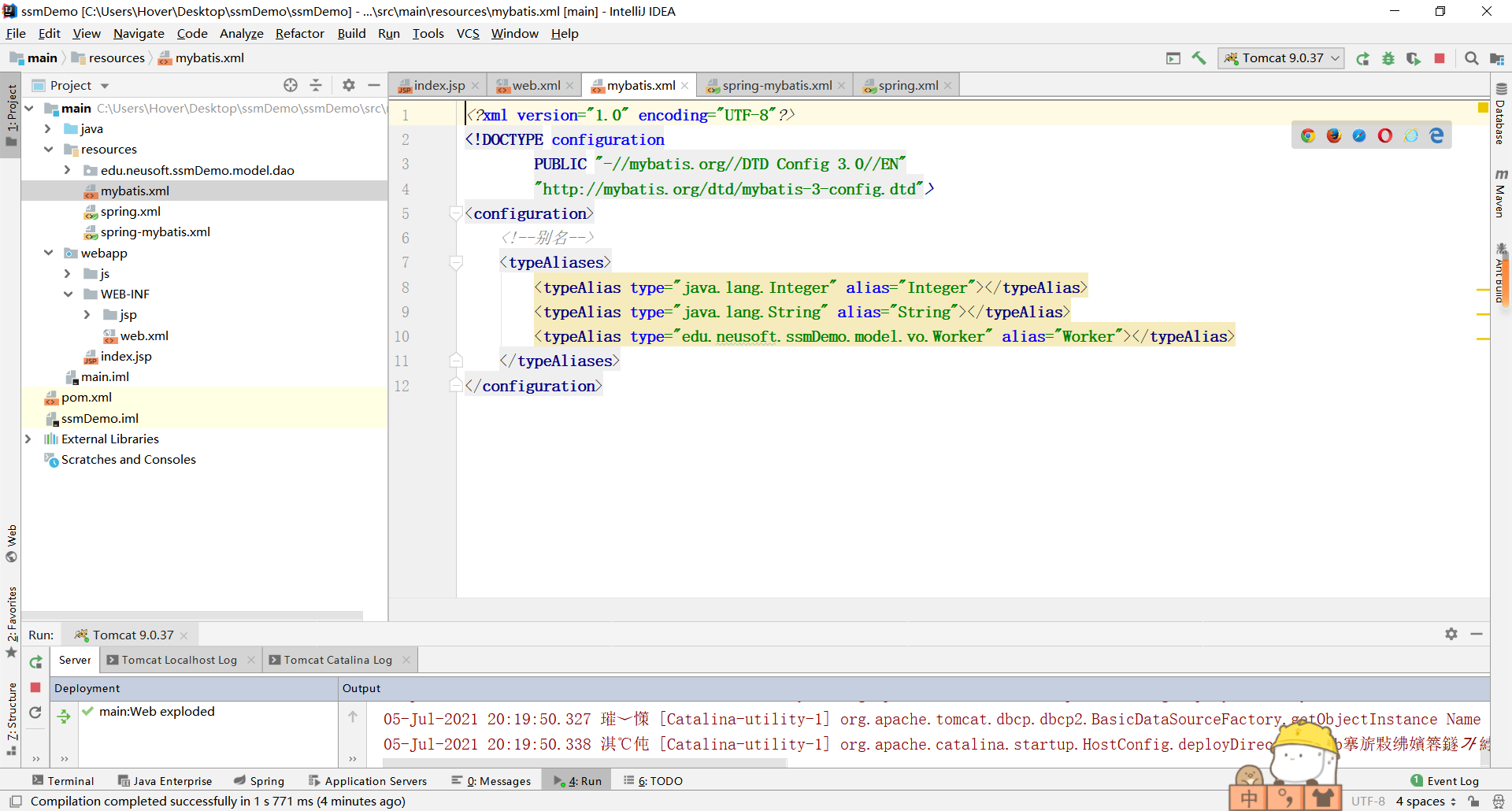
（1）创建mybatis-config.xml配置文件  
（2）创建sqlSessionFactory  
（3）编写数据库表对应的实体类  
（4）创建mybatis的sql映射文件，在这个文件中，把实体类的属性和数据库表的列联系起来，并且可以编写sql语句  
（5）从sqlSessionFactory的实例获取session  
（6）session内部通过Executor执行器执行操作。Executor会用到mapped statement对象，这个对象是对数据库存储的封装，包括：sql语句、输入参数、输出结果类型  
（7）关闭session

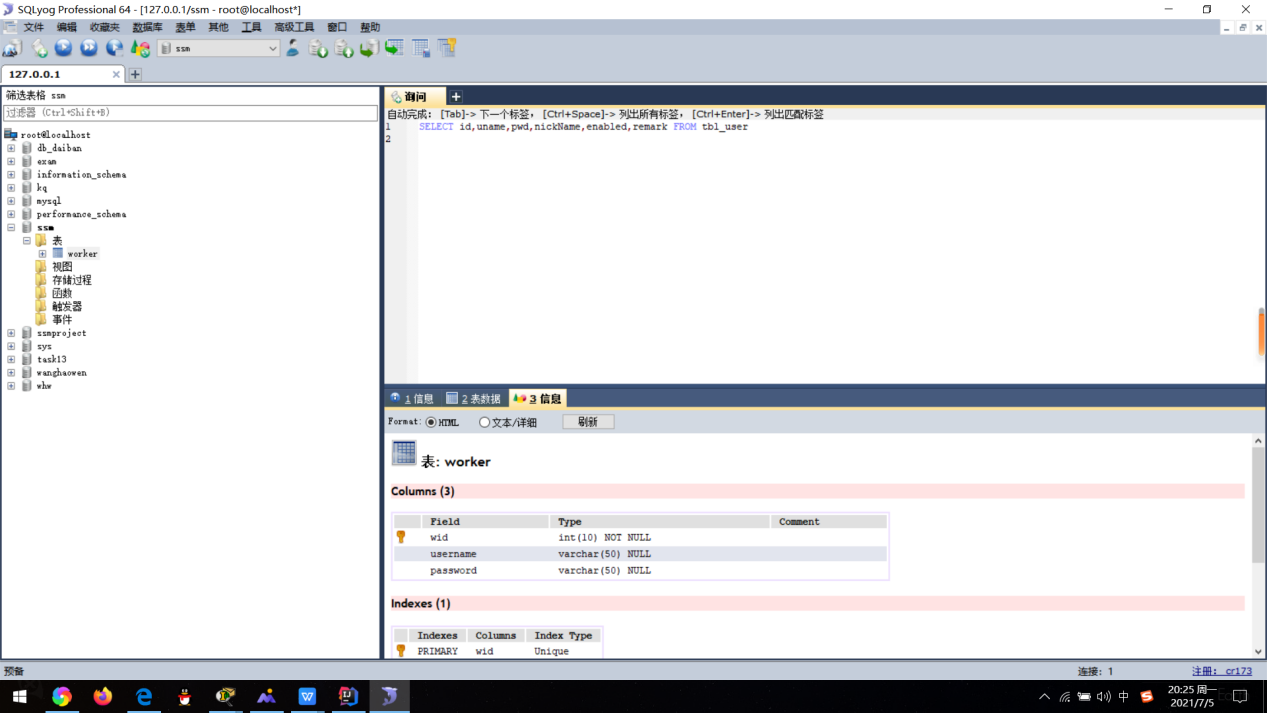
**作业二 报告**

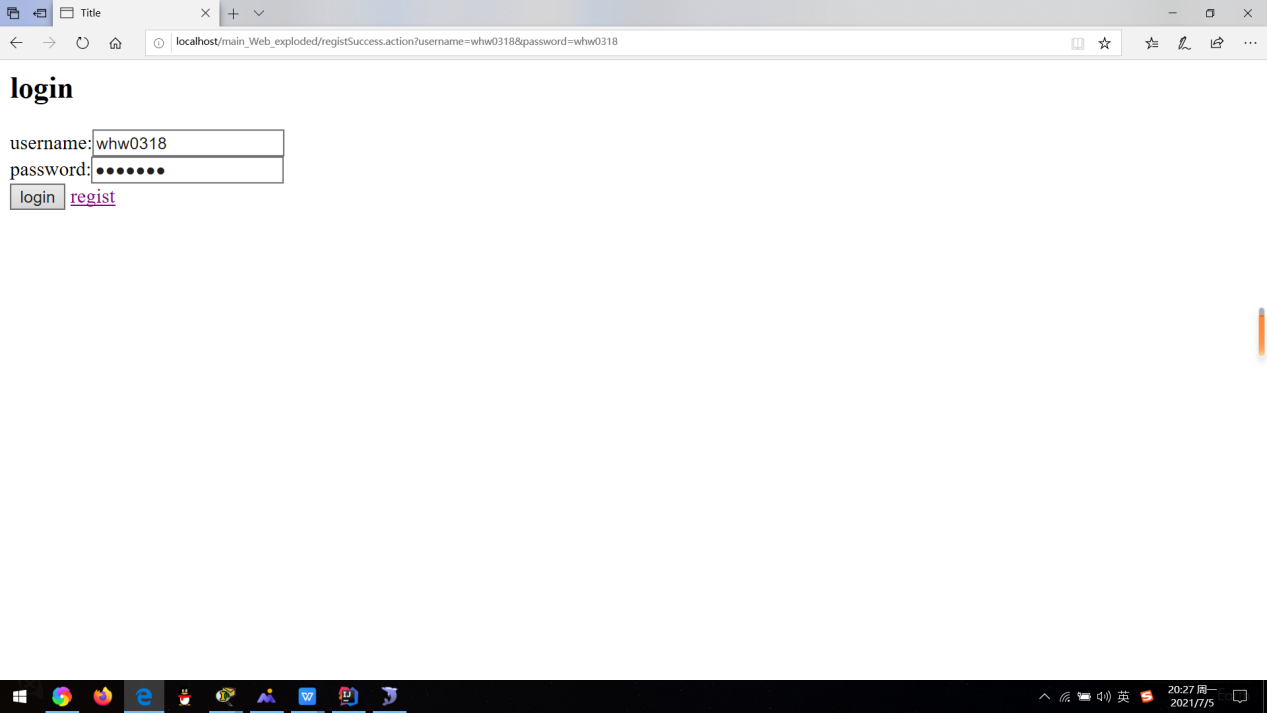
**SSM用户登录报告**

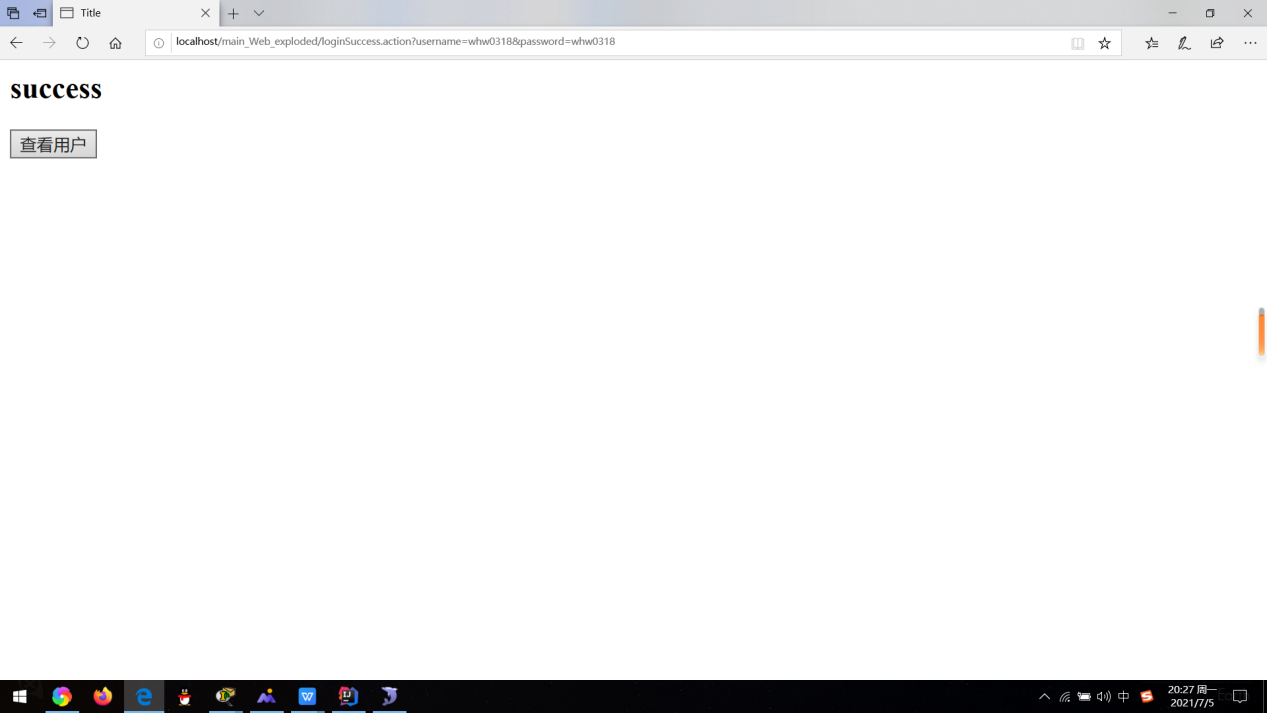


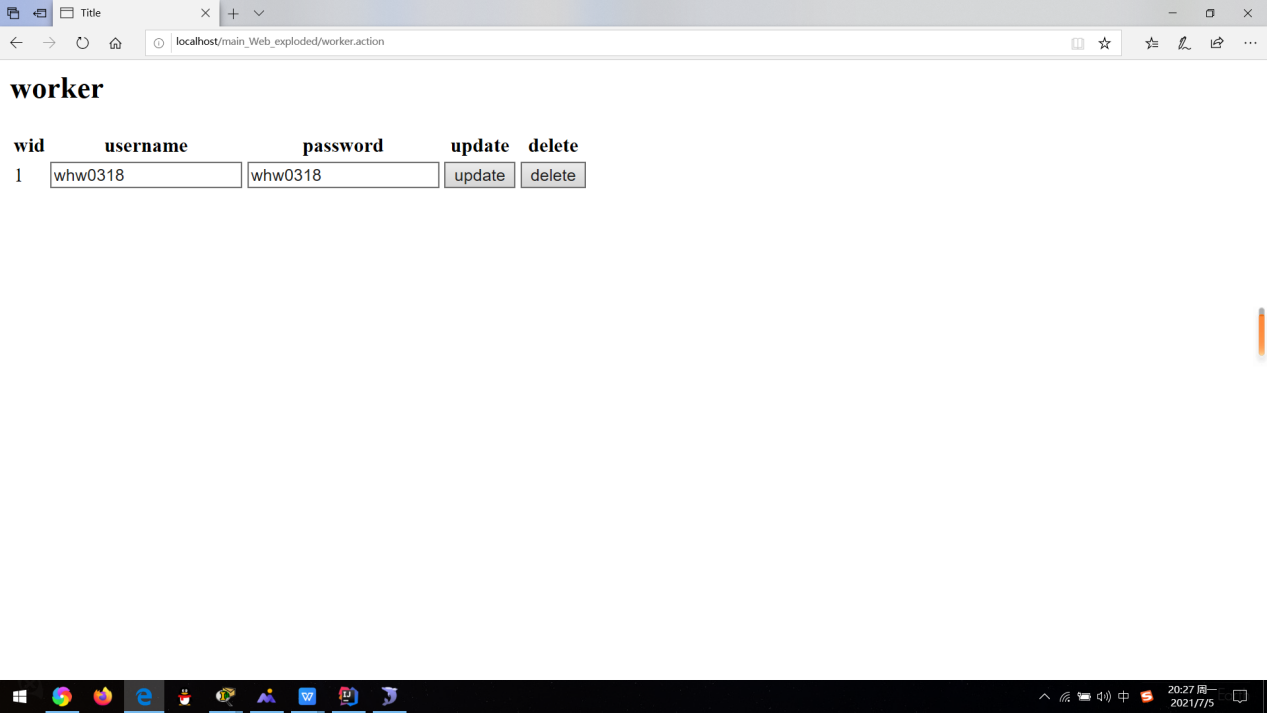












简介：通过SSM框架，对数据库进行增删改查，可以登陆注册并且显示所有用户信息

****需求：输入用户名和密码进行登录验证****

1. 整合思路

①、表现层，也就是 Controller,由 SpringMVC 来控制，而SpringMVC 是Spring 的一个模块，故不需要整合。

②、业务层，也就是 service，通常由 Spring 来管理 service 接口，我们会使用 xml 配置的方式来将 service 接口配置到 spring 配置文件中。而且事务控制一般也是在 service 层进行配置。

③、持久层，也就是 dao 层，而且包括实体类，由 MyBatis 来管理，通过 spring 来管理 mapper 接口，使用mapper的扫描器自动扫描mapper接口在spring中进行注册。

很明显，spring 在三大框架的整合中占据至关重要的地位，类似于一个大管家，将 MyBatis 和 SpringMVC 揉合在一起。

2、准备环境

①、数据库环境

数据库类型：MySQL 5.1

数据库名称：ssm

数据表：user

ID用户id，主键，自增，非空，自动生成

设置Username用户名

设置Password密码

②、开发工具 IDEA

③、JDK 1.8

④、mybatis 3.3

⑤、SpringMVC 4.2.4

⑥、Spring 4.2.4

⑦、数据库连接池 dbcp1.2.2

⑧、数据库驱动包mysql5.1.26

****①、定义 service 接口****

****package com.ys.service.impl;****

****import com.ys.po.User;****

****public interface IUserService {****

****//通过用户名和密码查询User****

****public User selectUserByUsernameAndPassword(User user);****

****}****

****②、编写 service 实现类****

****package com.ys.service;****

****import org.springframework.beans.factory.annotation.Autowired;****

****import com.ys.mapper.UserMapper;****

****import com.ys.po.User;****

****import com.ys.service.impl.IUserService;****

****public class UserServiceImpl implements IUserService{****

****@Autowired****

****private UserMapper userMapper; //通过@Autowired向spring容器注入UserMapper****

****//通过用户名和密码查询User****

****@Override****

****public User selectUserByUsernameAndPassword(User user) {****

****User u = userMapper.selectUserByUsernameAndPassword(user);****

****return u;****

****}****

****}****