第十一章习题答案

1.填空题

（1）@RequestMapping

（2）@RequestParam

（3）@PathVariable

（4）@CookieValue

（5）@RequestHeader

2.选择题

（1）B

（2）B

（3）B

（4）A

（5）B

3.思考题

（1）

答：Spring通过一系列组件实现功能，这些组件包括DispatcherServlet（前端控制器）、HandlerMapping（处理器映射器）、HandlAdapter（处理器适配器）、ViewResolver（视图解析器）等。

（2）

答：1、用户发送请求至前端控制器DispatcherServlet。

2、DispatcherServlet收到请求调用HandlerMapping处理器映射器。

3、处理器映射器找到具体的处理器(可以根据xml配置、注解进行查找)，生成处理器对象及处理器拦截器(如果有则生成)一并返回给DispatcherServlet。

4、DispatcherServlet调用HandlerAdapter处理器适配器。

5、HandlerAdapter经过适配调用具体的处理器(Controller，也叫后端控制器)。

6、Controller执行完成返回ModelAndView。

7、 HandlerAdapter将controller执行结果ModelAndView返回给DispatcherServlet。

8、DispatcherServlet将ModelAndView传给ViewReslover视图解析器。

9、ViewReslover解析后返回具体View。

10、DispatcherServlet根据View进行渲染视图（即将模型数据填充至视图中）。

11、DispatcherServlet响应用户。

4.编程题

数据库表、实体类和dao此处不再编写，详情见例题。

Service.java

1. @Service
2. public class StudentServiceImpl implements StudentService{
3. @Resource
4. private StudentDao studentDao;
5. @Override
6. public Student loginStu(String name, String password) {
7. Student stu = studentDao.selectByName(name);
8. if(stu!=null&&stu.getPassword().equals(password)) {
9. return stu;
10. }else {
11. return null;
12. }
13. }

Controller.java

1. @Controller
2. public class LoginController {
3. @Resource
4. private StudentService StudentService;
5. @RequestMapping(value="/login")
6. public String loginStu(String name, String password,HttpServletResponse response) {
7. Student stu = StudentService.loginStu(name, password);
8. if(stu!=null) {
9. return "success";
10. }else {
11. return "error";
12. }
13. }
14. }

Login.jsp

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7. <title>登录</title>
8. </head>
9. <body>
10. <h3>系统登录</h3>
11. <form action="login">
12. <label>姓名：</label><input name="name"><br />
13. <label>密码：</label><input name="password"><br />
14. <input type="submit" value="确认">
15. </form>
16. </body>
17. </html>

success.jsp

1. <%@ page language="java" contentType="text/html; charset=UTF-8"
2. pageEncoding="UTF-8"%>
3. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
4. <html>
5. <head>
6. <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7. <title>success</title>
8. </head>
9. <body>
10. <!-- <h2>success</h2>
11. <h4 style="color: red">本次操作成功！</h4> -->
12. 欢迎您，xiaoqian
13. </body>
14. </html>

Springmvc.xml

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans xmlns="http://www.springframework.org/schema/beans"
3. xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4. xmlns:context="http://www.springframework.org/schema/context"
5. xmlns:mvc="http://www.springframework.org/schema/mvc"
6. xsi:schemaLocation="http://www.springframework.org/schema/beans
7. http://www.springframework.org/schema/beans/spring-beans.xsd
8. http://www.springframework.org/schema/context
9. http://www.springframework.org/schema/context/spring-context.xsd
10. http://www.springframework.org/schema/mvc
11. http://www.springframework.org/schema/mvc/spring-mvc.xsd">
12. <!-- 启动自动扫描 -->
13. <context:component-scan
14. base-package="com.qfedu.controller" />
15. <!-- 注册MVC注解驱动 -->
16. <mvc:annotation-driven />
17. <!-- 静态资源可访问的设置方式 -->
18. <bean id="viewResolver"
19. class="org.springframework.web.servlet.view.InternalResourceViewResolver">
20. <property name="prefix" value="/" />
21. <property name="suffix" value=".jsp" />
22. </bean>
23. </beans>

Web.xml

1. <?xml version="1.0" encoding="UTF-8"?>
2. <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
3. xmlns="http://xmlns.jcp.org/xml/ns/javaee"
4. xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
5. http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd" version="3.1">
6. <display-name>Archetype Created Web Application</display-name>
7. <!--编码格式过滤器-->
8. <filter>
9. <filter-name>encoding</filter-name>
10. <filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>
11. <init-param>
12. <param-name>encoding</param-name>
13. <param-value>UTF-8</param-value>
14. </init-param>
15. </filter>
16. <filter-mapping>
17. <filter-name>encoding</filter-name>
18. <url-pattern>/\*</url-pattern>
19. </filter-mapping>
20. <!--Spring的配置文件-->
21. <context-param>
22. <param-name>contextConfigLocation</param-name>
23. <param-value>classpath:spring.xml</param-value>
24. </context-param>
25. <!--Spring的监听器-->
26. <listener>
27. <listener-class>
28. org.springframework.web.context.ContextLoaderListener
29. </listener-class>
30. </listener>
31. <!--SpringMVC的前端控制器-->
32. <servlet>
33. <servlet-name>springMVC</servlet-name>
34. <servlet-class>
35. org.springframework.web.servlet.DispatcherServlet
36. </servlet-class>
37. <init-param>
38. <param-name>contextConfigLocation</param-name>
39. <param-value>classpath:spring-mvc.xml</param-value>
40. </init-param>
41. <load-on-startup>1</load-on-startup>
42. </servlet>
43. <!--访问DispatcherServlet对应的路径-->
44. <servlet-mapping>
45. <servlet-name>springMVC</servlet-name>
46. <url-pattern>/</url-pattern>
47. </servlet-mapping>
48. </web-app>