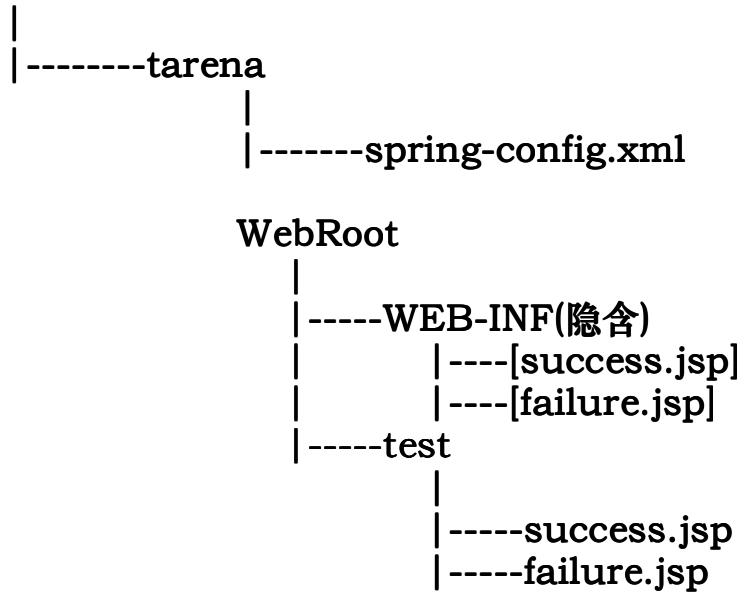


实验一.txt

实验一，Spring2WebMVC框架使用

SpringMVCDemo(一个简单的登录功能)



spring-webmvc.jar--> <!--加入Spring2.0 WebLibrary Jar包，使用

步骤一，在web.xml文件中修改为以下信息

固定写法：

<servlet>

<servlet-name>dispatcherServlet</servlet-name>
<servlet-class>

实验一.txt

```
org.springframework.web.servlet.DispatcherServlet
    </servlet-class>
        <init-param>
            <param-name>
contextConfigLocation
            </param-name>
            <param-value>
/WEB-INF/classes/tarena/spring-config.xml
            </param-value>
        </init-param>
    </servlet>

    <servlet-mapping>

<servlet-name>dispatcherServlet</servlet-name>
        <url-pattern>*.do</url-pattern>
    </servlet-mapping>apping>
```

步骤二，创建login.jsp页面

```
<%@ page language="java" pageEncoding="GBK"%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN">
<html>
    <head>
        <title>用户登录页面</title>
    </head>

    <body>
        <form action="login.qq" method="post">
            <table border="5" align="center">
                <tr>
                    <td>
                        用户名

```

实验一.txt

```

type="text" name="username" />
type="submit" value="登录" />
</html>
</body>
</form>
</table>
</tr>
<tr>
<td>
<input
type="text" name="username" />
</td>
</tr>
<tr>
<td>
<input
type="submit" value="登录" />
</td>
</tr>
</table>
</form>
</body>
</html>
```

步骤三，创建LoginAction类实现Controller接口

```
public class LoginAction implements Controller {

    private String success_view;
    private String failure_view;

    public String getSuccess_view() {
        return success_view;
    }

    public void setSuccess_view(String success_view) {
        this.success_view = success_view;
    }

    public String getFailure_view() {
        return failure_view;
    }
}
```

实验一.txt

```
public void setFailure_view(String failure_view) {
    this.failure_view = failure_view;
}

public ModelAndView handleRequest(HttpServletRequest
request,
                                HttpServletResponse response) throws
Exception {

    String username =
request.getParameter("username");

    if ("admin".equals(username)) {
        System.out.println("登录成功");
        //success为要转发到的JSP名字
        return new ModelAndView("success");
    } else {
        System.out.println("登录失败");
        return new ModelAndView("failure");
    }
}
}
```

步骤四，配置spring-config.xml文件

```
<?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
ns
http://www.springframework.org/schema/beans/spring-beans-2.0.x
```

实验一.txt

sd">

<!-- 视图处理 -->

<bean id="InternalResourceViewResolver"

class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix">

<!--/表示WebRoot目录-->

<value>/test/</value>

</property>

<property name="suffix">

<value>.jsp</value>

</property>

</bean>

<!-- 控制器处理 -->

<bean name="/login.do" class="tarena.LoginAction">

<!--success为JSP页面的名字-->

<property name="success_view">

<value>success</value>

</property>

<property name="failure_view">

<value>failure</value>

</property>

</bean>

</beans>

步骤五，部署到Tomcat服务器上，在浏览器中访问

<http://localhost:8080/SpringMVCDemo/login.jsp>

填入：

用户名：admin

实验一.txt

返回:

登录成功
success.jsp

步骤六，分析总结