# Struts

## struts简介

### 什么是框架

框架（Framework）：是整个或部分系统的可重用设计，表现为一组公用组件及组件实例间交互的方式。

通俗的说就是别人写好的程序，来帮你简化程序开发

### struts2是什么

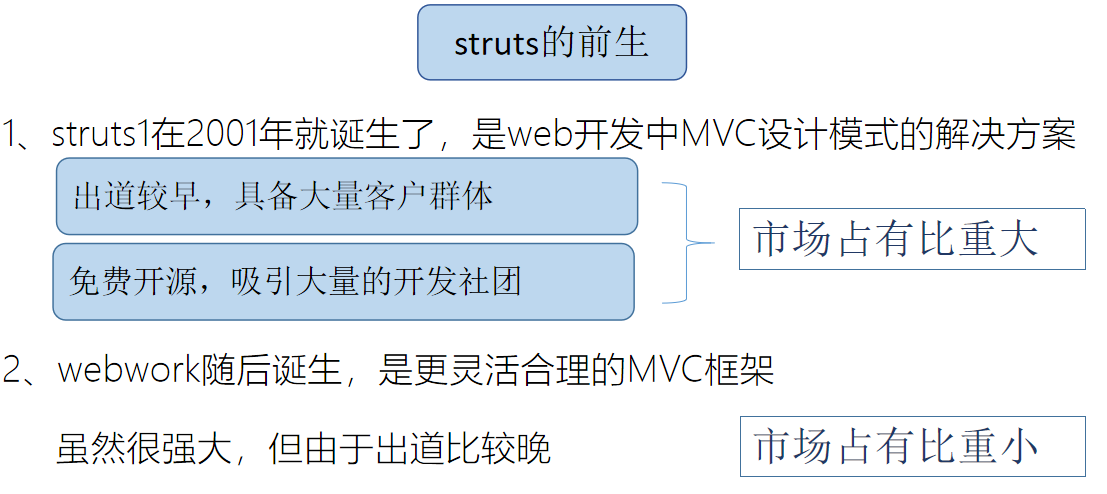
struts2是J2EE的比较流行的框架，对JSP/servlet进行了封装，应用于面向WEB开发的项目。struts是MVC设计模式的一种解决方案

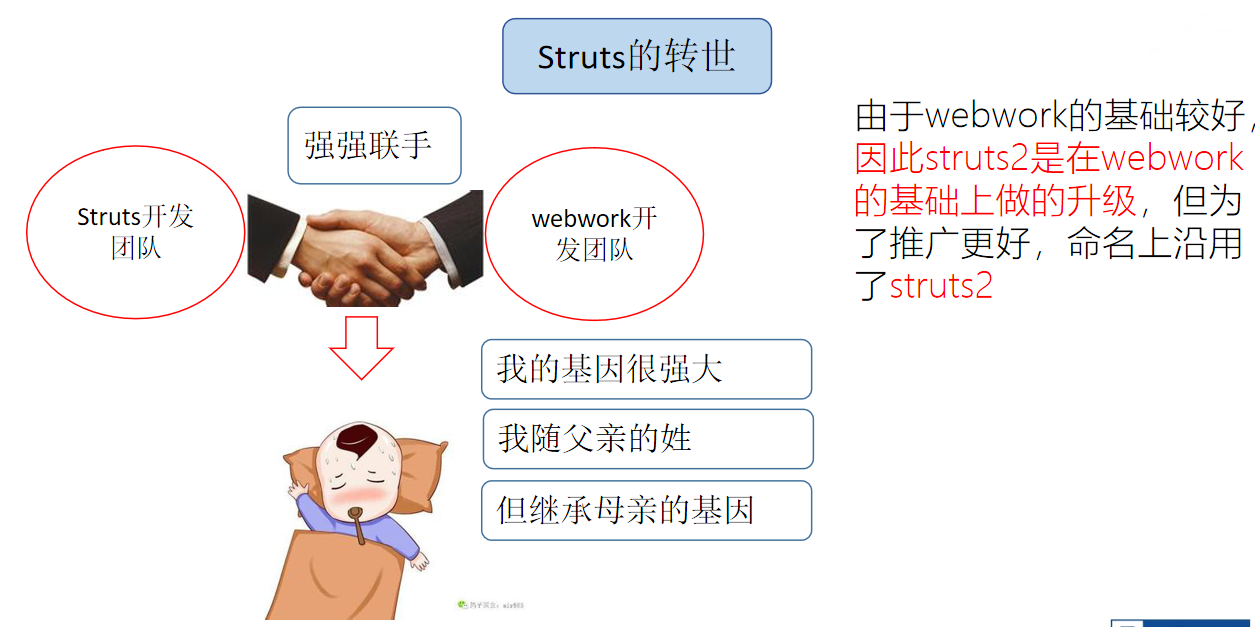
### Struts2的特点

提供了MVC设计模式的解决方案

提供了一组web开发常用功能组件

### struts2的前生今世





### struts2与SpringMVC的区别

(1) springmvc的入口是一个servlet即前端控制器(DispatchServlet),而struts2入口是一个filter过滤器(StrutsPrepareAndExecuteFilter) 。

(2) springmvc是基于方法开发(一个url对应一个方法)，请求参数传递到方法的形参，struts2是基于类开发，传递参数是通过类的属性。

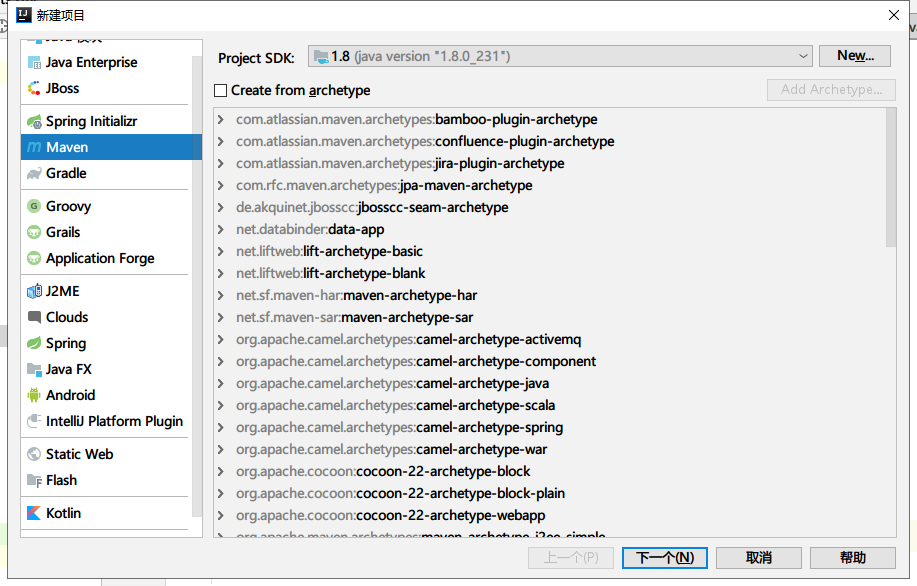
## struts2的官网

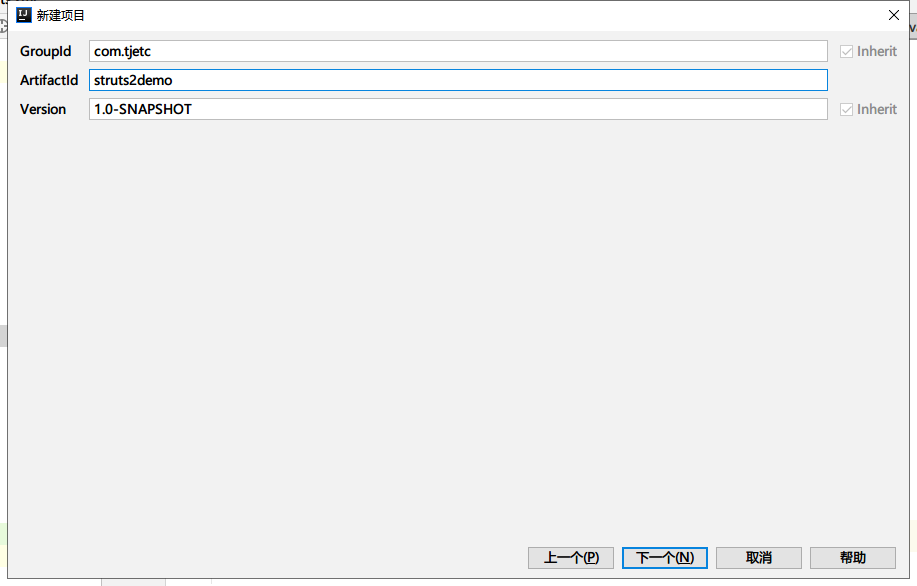
http://struts.apache.org/

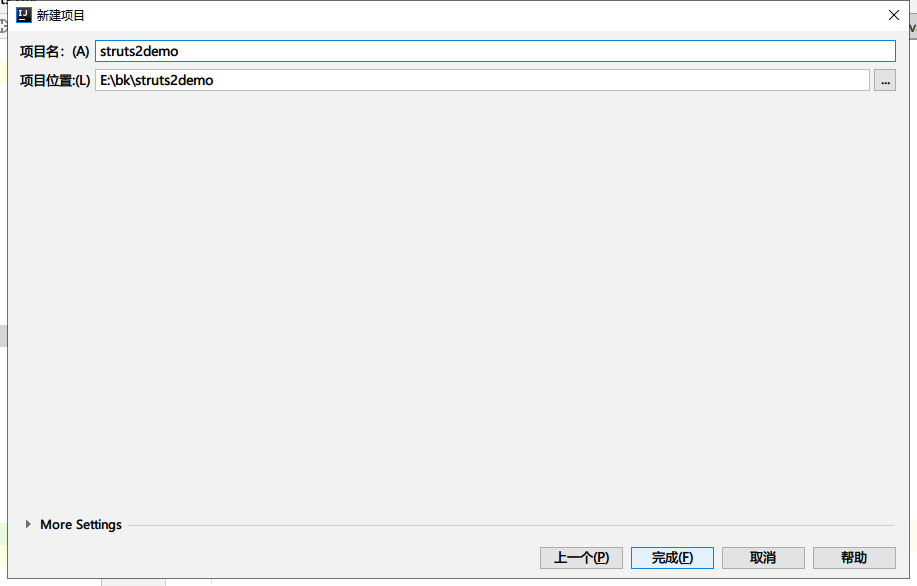
# 项目创建及配置

1. struts2项目创建

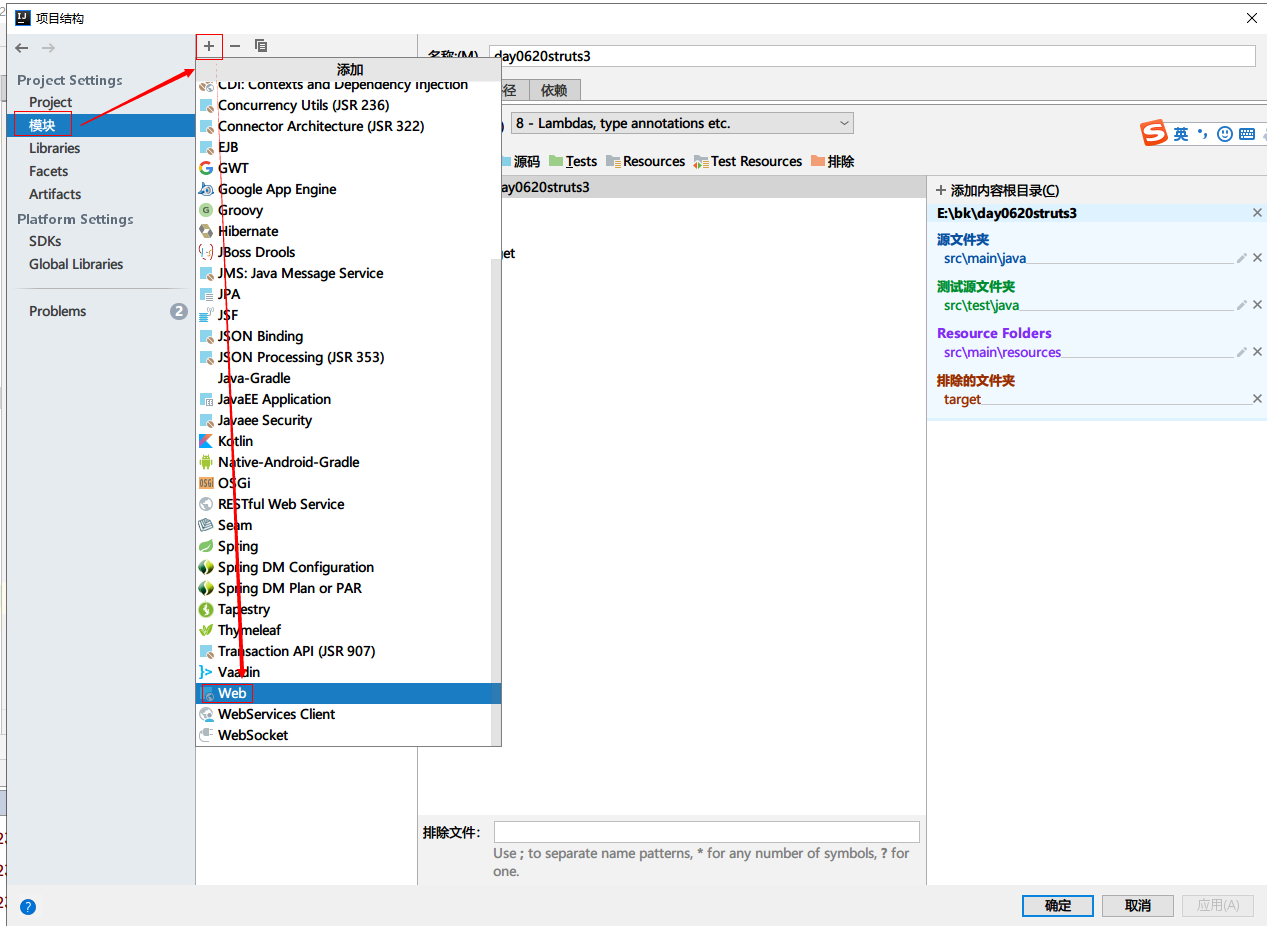
1) 创建Maven Project

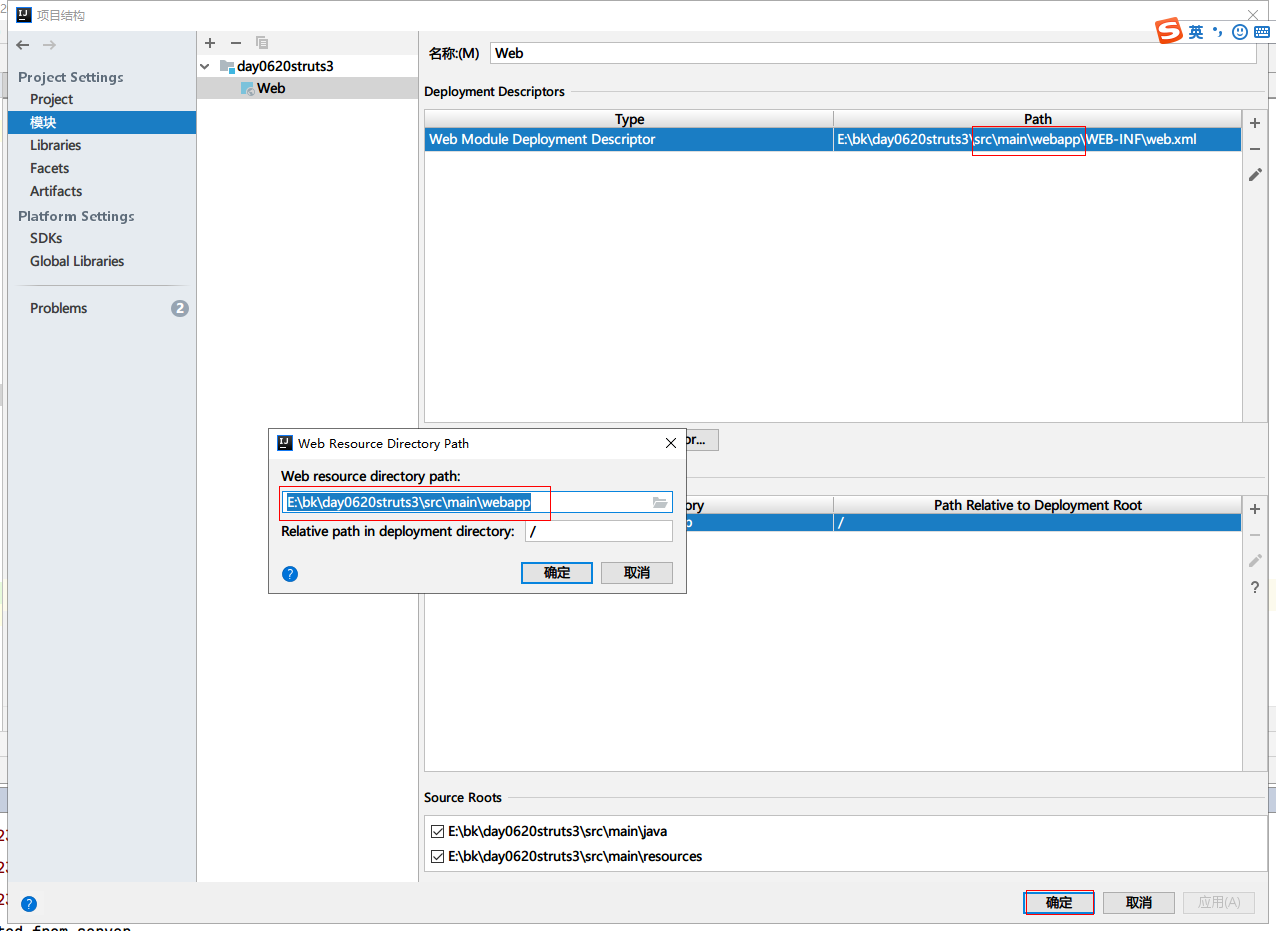






1. 配置Web





――――――――――――――基础配置完毕――――――――――――

maven项目中tomcat插件的使用

tomcat插件可以极大的简化开发人员的操作

|  |
| --- |
| <build>  <plugins>  <plugin>  <groupId>org.apache.tomcat.maven</groupId>  <artifactId>tomcat7-maven-plugin</artifactId>  <version>2.2</version>  <configuration>  <port>8080</port>  <path>/</path>  <uriEncoding>UTF-8</uriEncoding>  </configuration>  </plugin>  </plugins>  </build> |

path 是访问应用的路径

port 是tomcat 的端口号

uriEncoding URL按UTF-8进行编码，这样就解决了中文参数乱码。

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.tjetc</groupId>  <artifactId>day0620struts3</artifactId>  <version>1.0-SNAPSHOT</version>  <packaging>war</packaging>  <dependencies>  <dependency>  <groupId>org.apache.struts</groupId>  <artifactId>struts2-core</artifactId>  <version>2.5.2</version>  </dependency>  </dependencies>  <build>  <plugins>  <plugin>  <groupId>org.apache.tomcat.maven</groupId>  <artifactId>tomcat7-maven-plugin</artifactId>  <version>2.2</version>  <configuration>  <port>8080</port>  <path>/</path>  <uriEncoding>UTF-8</uriEncoding>  </configuration>  </plugin>  </plugins>  </build>  </project> |

web.xml

|  |
| --- |
| <**web-app**>  <**display-name**>Archetype Created Web Application</**display-name**>  <**filter**>  <**filter-name**>struts2</**filter-name**> <**filter-class**>org.apache.struts2.dispatcher.filter.StrutsPrepareAndExecuteFilter</**filter-class**>  </**filter**>  <**filter-mapping**>  <**filter-name**>struts2</**filter-name**>  <**url-pattern**>/\*</**url-pattern**>  </**filter-mapping**>  </**web-app**> |

在src/main/java中创建一个包：com.tjetc.action

创建一个类HelloAction

|  |
| --- |
| **package** com.tjetc.action;  **public** **class** HelloAction {  **public** String execute() {  System.***out***.println("hello()...");  **return** "hello";  }  } |

src/main/resources/struts.xml ，注册Action

|  |
| --- |
| **<?xml version="1.0" encoding="UTF-8"?>**  **<!DOCTYPE struts PUBLIC**  **"-//Apache Software Foundation//DTD Struts Configuration 2.5//EN"**  **"http://struts.apache.org/dtds/struts-2.5.dtd">**  **<struts>**  **<!--package表示包，name表示包的名字，便于其他的包引用，因为其他的包可以继承这个包-->**  **<!--指定该包的命名空间，默认的是“/”,一个配置文件中可能有相同名字的action，**  **如果配置不同的namespace，那么就可以配置相同的action的名字-->**  **<!-- extends表示继承其他的包，但是我们一般都会继承struts-default-->**  **<package name="hello" namespace="/" extends="struts-default">**  **<!--action表示处理的请求配置，name表示请求的url，class表示处理请求的类-->**  **<action name="hello" class="com.tjetc.action.HelloAction">**  **<!--result 表示结果，例如结果返回 /hello.jsp资源-->**  **<result name="hello">/hello.jsp</result>**  **</action>**  **</package>**  **</struts>** |

Src\main\webapp 创建hello.jsp

|  |
| --- |
| <%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*  pageEncoding=*"UTF-8"*%>  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  <html>  <head>  <meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>  <title>Insert title here</title>  </head>  <body>  hello.jsp  </body>  </html> |

启动Tomcat

访问http://localhost:8080/hello 测试是否成功

hello.jsp

# 用户登录案例

任务名称 基于Struts2实现用户登录功能

开发环境 Intellij IDEA Chrome

完成功能 使用Struts2实现用户登录功能

1.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.tjetc</groupId>  <artifactId>day0622struts2</artifactId>  <version>1.0-SNAPSHOT</version>  <packaging>war</packaging>  <dependencies>  <dependency>  <groupId>org.apache.struts</groupId>  <artifactId>struts2-core</artifactId>  <version>2.5.2</version>  </dependency>  <dependency>  <groupId>javax.servlet</groupId>  <artifactId>javax.servlet-api</artifactId>  <version>4.0.1</version>  <scope>provided</scope>  </dependency>  </dependencies>  <build>  <plugins>  <plugin>  <groupId>org.apache.tomcat.maven</groupId>  <artifactId>tomcat7-maven-plugin</artifactId>  <version>2.2</version>  <configuration>  <port>8080</port>  <path>/</path>  <uriEncoding>UTF-8</uriEncoding>  </configuration>  </plugin>  </plugins>  </build>  </project> |

2.在src/main/webapp/ 创建login.jsp

|  |
| --- |
| <body>  <form action="user/login" method="post">  用户名:<input type="text" name="username" /><br>  密码:<input type="password" name="password" /><br>  <input type="submit" value="登录" /><br>  </form>  </body> |

LoginAction

|  |
| --- |
| package com.tjetc.action;  import com.opensymphony.xwork2.ModelDriven;  import com.tjetc.domain.User;  public class LoginAction implements ModelDriven<User> {  private User user=new User();  @Override  public User getModel() {  // TODO Auto-generated method stub  return user;  }  public String execute() {  System.out.println(user);  if (user.getUsername().equals("admin") && user.getPassword().equals("123456")){  ServletActionContext.getRequest().getSession().setAttribute("user", user);  return "success";  }  return "fail";  }    } |

在src/main/java 下创建 创建一个实体类：User

|  |
| --- |
| package com.tjetc.domain;  public class User {  private String username;  private String password;  public String getUsername() {  return username;  }  public void setUsername(String username) {  this.username = username;  }  public String getPassword() {  return password;  }  public void setPassword(String password) {  this.password = password;  }  @Override  public String toString() {  return "User [username=" + username + ", password=" + password + "]";  }    } |

1. struts.xml

|  |
| --- |
| <package name="login" namespace="/user" extends="struts-default">  <action name="login" class="com.tjetc.action.LoginAction">  <result name="success">/success.jsp</result>  <result name="fail">/fail.jsp</result>  </action>  </package> |

1. success.jsp

|  |
| --- |
| <%@ page contentType="text/html;charset=UTF-8" language="java" %>  <html>  <head>  <title>Title</title>  <base href="${pageContext.request.contextPath}/"/>  </head>  <body>  登录成功!!!  </body>  </html> |

6.fail.jsp

|  |
| --- |
| <%@ page contentType="text/html;charset=UTF-8" language="java" %>  <html>  <head>  <title>Title</title>  <base href="${pageContext.request.contextPath}/"/>  </head>  <body>  登录失败,请 <a href="login.jsp">重新登录</a>  </body>  </html> |

5.运行login.jsp，并测试是否可以访问到loginAction

