### 以对象的方式处理表单数据

1. 属性驱动

Jsp页面

register.jsp

|  |
| --- |
| <%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>  <%  String path = request.getContextPath();  String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";  %>  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  <html>  <head>  <base href=*"*<%=basePath%>*"*>    <title>My JSP 'register.jsp' starting page</title>    <meta http-equiv=*"pragma"* content=*"no-cache"*>  <meta http-equiv=*"cache-control"* content=*"no-cache"*>  <meta http-equiv=*"expires"* content=*"0"*>  <meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>  <meta http-equiv=*"description"* content=*"This is my page"*>  <!--  <link rel="stylesheet" type="text/css" href="styles.css">  -->  </head>    <body>    <form action=*"register.action"* method=*"post"*>  用户名：<input type = *"text"* name = *"user.name"*/><br>  密码：<input type = *"password"* name = *"user.pwd"*/><br>  年龄：<input type = *"text"* name = *"user.age"*/><br>  邮箱：<input type = *"text"* name = *"user.email"*/><br>  <input type = *"submit"* value = *"提交"*>  </form>    </body>  </html> |

show.jsp

|  |
| --- |
| <%@ page language=*"java"* import=*"java.util.\*"* pageEncoding=*"utf-8"*%>  <%  String path = request.getContextPath();  String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";  %>  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  <html>  <head>  <base href=*"*<%=basePath%>*"*>    <title>My JSP 'show.jsp' starting page</title>    <meta http-equiv=*"pragma"* content=*"no-cache"*>  <meta http-equiv=*"cache-control"* content=*"no-cache"*>  <meta http-equiv=*"expires"* content=*"0"*>  <meta http-equiv=*"keywords"* content=*"keyword1,keyword2,keyword3"*>  <meta http-equiv=*"description"* content=*"This is my page"*>  <!--  <link rel="stylesheet" type="text/css" href="styles.css">  -->  </head>    <body>  姓名：${user}  </body>  </html> |

实体类User 代码：

|  |
| --- |
| **package** com.qimh.entity;  **public** **class** User {  **private** String *name*;  **private** String *pwd*;  **private** **int** *age*;  **public** **int** getAge() {  **return** age;  }  **public** **void** setAge(**int** age) {  **this**.age = age;  }  **public** String getEmail() {  **return** email;  }  **public** **void** setEmail(String email) {  **this**.email = email;  }  **private** String email;  **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  **public** String getPwd() {  **return** pwd;  }  **public** **void** setPwd(String pwd) {  **this**.pwd = pwd;  }  @Override  **public** String toString() {  **return** "User [name=" + name + ", pwd=" + pwd + ", age=" + age  + ", email=" + email + "]";  }          } |

Action类

|  |
| --- |
| **package** com.qimh.action;  **import** com.opensymphony.xwork2.Action;  **import** com.qimh.entity.User;  **public** **class** UserAction {  **private** User user;      //注册  **public** String register(){  System.*out*.println("user对象:"+user);  **return** Action.*SUCCESS*;  }      **public** User getUser() {  **return** user;  }  **public** **void** setUser(User user) {  **this**.user = user;  }  } |

struts.xml 配置文件

|  |
| --- |
| <?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>  <!-- 配置struts2属性 -->  <!-- 乱码解决 -->  <constant name = *"struts.i18n.encoding"* value = *"UTF-8"*/>  <!-- 自定义扩展名  <constant name = "struts.action.extension" value="action,,qimh"></constant>-->  <!-- 设置开发模式-->  <constant name = *"struts.devMode"* value=*"true"*></constant>  <!-- 自动加载-->  <constant name = *"struts.configuration.xml.reload"* value=*"true"*></constant>  <!-- extends 必须写，直接或者间接继承struts-default name自定义 -->  <package name=*"user"* extends=*"struts-default"*>  <!-- name是请求名称，不要写/ ;class对应action完全限定名=包名+类名 -->  <action name=*"register"* method =*"register"* class=*"com.qimh.action.UserAction"*>  <!-- result 是结果集 name值和对应的action中的方法的返回值匹配，默认是success-->  <result name = *"success"*>/show.jsp</result>  </action>  </package>  </struts> |

1. 模型驱动（modeldriven）

jsp页面

regModel.jsp

|  |
| --- |
| <%@ page language="java" import="java.util.\*" pageEncoding="utf-8"%>  <%  String path = request.getContextPath();  String basePath = request.getScheme()+"://"+request.getServerName()+":"+request.getServerPort()+path+"/";  %>  <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  <html>  <head>  <base href="<%=basePath%>">    <title>My JSP 'regModel.jsp' starting page</title>    <meta http-equiv="pragma" content="no-cache">  <meta http-equiv="cache-control" content="no-cache">  <meta http-equiv="expires" content="0">  <meta http-equiv="keywords" content="keyword1,keyword2,keyword3">  <meta http-equiv="description" content="This is my page">  <!--  <link rel="stylesheet" type="text/css" href="styles.css">  -->  </head>    <body>  <form action="regModel.action" method="post">  用户名：<input type = "text" name = "name"/><br>  密码：<input type = "password" name = "pwd"/><br>  年龄：<input type = "text" name = "age"/><br>  邮箱：<input type = "text" name = "email"/><br>  <input type = "submit" value = "提交">  </form>  </body>  </html> |

UserModelAction 类

|  |
| --- |
| package com.qimh.action;  import com.opensymphony.xwork2.Action;  import com.opensymphony.xwork2.ModelDriven;  import com.qimh.entity.User;  public class UserModelAction implements ModelDriven<User>{  private User user = new User();      //注册  public String register(){  System.out.println("表单数据："+user);  return Action.SUCCESS;  }    public User getModel() {  // TODO Auto-generated method stub  return user;  }  public User getUser() {  return user;  }  public void setUser(User user) {  this.user = user;  }      } |

实体类：

|  |
| --- |
| **package** com.qimh.entity;  **public** **class** User {  **private** String name;  **private** String pwd;  **private** **int** age;  **public** **int** getAge() {  **return** age;  }  **public** **void** setAge(**int** age) {  **this**.age = age;  }  **public** String getEmail() {  **return** email;  }  **public** **void** setEmail(String email) {  **this**.email = email;  }  **private** String email;  **public** String getName() {  **return** name;  }  **public** **void** setName(String name) {  **this**.name = name;  }  **public** String getPwd() {  **return** pwd;  }  **public** **void** setPwd(String pwd) {  **this**.pwd = pwd;  }  @Override  **public** String toString() {  **return** "User [name=" + name + ", pwd=" + pwd + ", age=" + age  + ", email=" + email + "]";  }          } |

Struts.xml

|  |
| --- |
| <?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>  <!-- 配置struts2属性 -->  <!-- 乱码解决 -->  <constant name = "struts.i18n.encoding" value = "UTF-8"/>  <!-- 自定义扩展名  <constant name = "struts.action.extension" value="action,,qimh"></constant>-->  <!-- 设置开发模式-->  <constant name = "struts.devMode" value="true"></constant>  <!-- 自动加载-->  <constant name = "struts.configuration.xml.reload" value="true"></constant>  <!-- extends 必须写，直接或者间接继承struts-default name自定义 -->  <package name="user" extends="struts-default">  <!-- name是请求名称，不要写/ ;class对应action完全限定名=包名+类名 -->  <action name="register" method ="register" class="com.qimh.action.UserAction">  <!-- result 是结果集 name值和对应的action中的方法的返回值匹配，默认是success-->  <result name = "success">/show.jsp</result>  </action>    <action name="regModel" method ="register" class="com.qimh.action.UserModelAction">  <result name = "success">/show.jsp</result>  </action>  </package>  </struts> |

属性驱动和模型驱动 自选。如果属性较多 使用属性模型驱动。