

第5章 JSP 与 JavaBean

本章主要介绍在 JSP 程序中如何使用 JavaBean 组件。通过本章的学习,读者应熟练掌握<jsp: useBean>、<jsp: setProperty>、<jsp: getProperty>等 JSP 的操作指令。

- JavaBean 的含义
- JSP 中使用 JavaBean

一个 JSP 页面通过使用 HTML 标记为用户显示数据(静态部分),页面中变量的声明、程序片以及表达式为动态部分,对数据进行处理。如果 Java 程序片和 HTML 标记大量掺杂在一起使用,就不利于 JSP 页面的扩展和维护。JSP 和 JavaBean 技术的结合不仅可以实现数据的表示和处理分离,而且可以提高 JSP 程序代码重用的程度,是 JSP 编程中常用的技术。

本章涉及的 Java 源文件保存在工程 ch5 的 src 中,涉及的 JSP 页面保存在工程 ch5 的 WebContent 中。

5.1 编写 JavaBean

JavaBean 是一个可重复使用的软件组件,是遵循一定标准、用 Java 语言编写的一个类,该类的一个实例称为一个 JavaBean,简称 bean。JavaBean 具有可重用、升级方便、不依赖于平台等特点。JavaBean 又可分为业务 bean 和数据 bean。业务 bean 用于封装业务逻辑、数据库操作等;数据 bean 用于封装数据。

编写一个 JavaBean 就是编写一个 Java 类(该类必须带有包名),这个类创建的一个对象称为一个 bean,为了让 JSP 引擎(比如 Tomcat)知道这个 bean 的属性和方法,JavaBean 类必须遵守以下规则:

(1) 如果类的成员变量的名字是 name,那么为了获取或更改成员变量的值,类中必须提供以下两个方法:

getName(),用来获取属性 name。

setName(),用来修改属性 name。

即方法的名字用 get 或 set 为前缀,后缀是首字母大写的成员变量的名字。

(2) 对于 boolean 类型的成员变量,允许使用“is”代替上面的“get”和“set”。

(3) 类中方法的访问权限必须是 public。

(4) 构造方法必须无参数。

【例 5-1】 创建 bean 的源文件 Rectangle.java(在包 com. bean 中)。该 bean 的作用是封装矩形的长和宽,并计算矩形的面积和周长。

例 5-1 Rectangle.java 的代码如下:

```
package com.bean;
public class Rectangle {
private double length;
```

```

private double width;
public Rectangle() {
    length = 20;
    width = 10;
}
public double getLength() {
    return length;
}
public void setLength(double length) {
    this.length = length;
}
public double getWidth() {
    return width;
}
public void setWidth(double width) {
    this.width = width;
}
public double computerArea() {
    return length * width;
}
public double computerLength() {
    return (length + width) * 2;
}
}

```

JavaBean 可以在任何 Java 程序编写环境下完成编写,再通过编译成为一个字节码文件,为了让 JSP 引擎(比如 Tomcat)找到这个字节码,必须把字节码文件放在特定的位置。本书使用 Eclipse 集成环境开发 JSP 程序,Java 类的字节码文件由 Eclipse 自动保存到 Web 工程的 build\classes 中。例如,例 5-1 中的 Rectangle.class 文件保存在 ch5\build\classes\com\bean 目录中。

5.2 JSP 中使用 JavaBean

在 JSP 页面中使用 bean 时,首先使用 page 指令的 import 属性导入创建 bean 的类,例如:

```
<%@page import="com.bean.*"%>
```

5.2.1 动作标记 useBean

useBean 动作标记是用来查找或者实例化一个 JavaBean。useBean 标记的格式为:

```
<jsp:useBean id="bean 的名字" class="创建 bean 的类" scope="bean 的有效范围" />
```

或

```
<jsp:useBean id="bean 的名字" type="创建 bean 的类" scope="bean 的有效范围" />
```

例如：

```
<jsp:useBean id="rectangle" class="com.bean.Rectangle" scope="page"/>
```

useBean 标记中各属性的含义如表 5.1 所示。

表 5.1 useBean 标记中各属性的含义

属性名	描 述
id	指定该 JavaBean 实例的变量名,通过 id 可以访问这个实例
class	指定 JavaBean 的类名。如果需要创建一个新的实例,Web 容器会使用 class 指定的类,并调用无参数的构造方法来完成实例化
scope	指定 JavaBean 的作用范围,包括 page、request、session 和 application。缺省值为 page,表明此 JavaBean 只能应用于当前页;值为 request 表明此 JavaBean 只能应用于当前的请求;值为 session 表明此 JavaBean 能应用于当前会话;值为 application 则表明此 JavaBean 能应用于整个应用程序
type	指定 JavaBean 对象的类型,通常在查找已存在的 JavaBean 时使用,这时使用 type 将不会产生新的对象。如果是查找已存在的 JavaBean 对象,type 属性的值可以是此对象的准确类名、其父类或者其实现的接口;如果是新建实例,则只能是准确类名或者父类。另外,如果能够确定此 JavaBean 的对象肯定存在,则指定 type 属性后可以省略 class 属性。

当含有 useBean 动作标记的 JSP 页面被 Web 容器加载执行时,Web 容器首先根据 id 的名字,在 pageContent 内置对象中查看是否含有名字为 id 和作用域为 scope 的对象;如果该对象存在,Web 容器就将这个对象的副本(bean)分配给 JSP 页面使用;如果没有找到,就根据 class 指定的类创建一个名字是 id 的 bean,并添加到 pageContent 对象中,同时将这个 bean 分配给 JSP 页面使用。useBean 动作标记执行流程如图 5.1 所示。

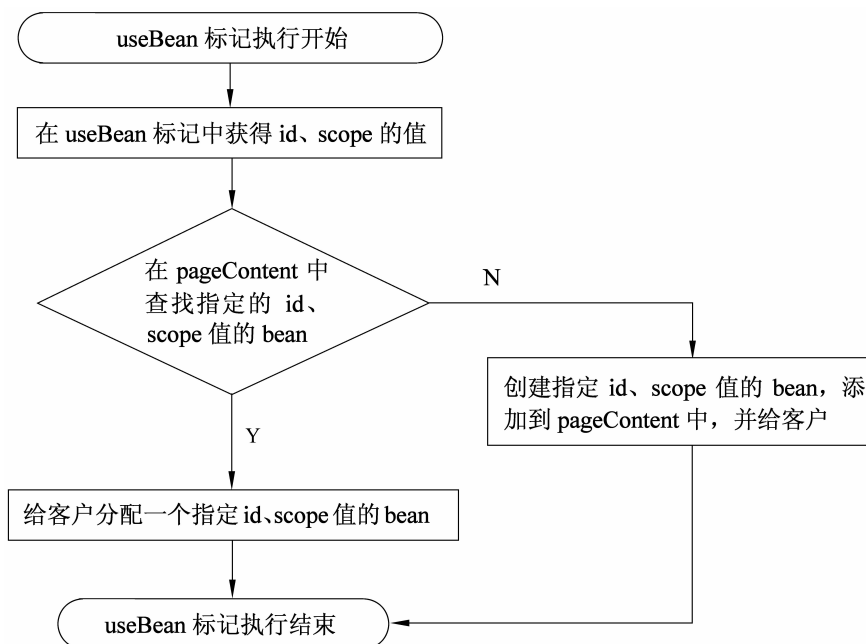


图 5.1 useBean 标记执行流程图

【例 5-2】 编写一个 JSP 页面 example5_2.jsp,在 JSP 页面中使用 useBean 标记获得一个 bean,负责创建 bean 的类是例 5-1 中的 Rectangle 类,bean 的名字是 rectangle,rectangle 的 scope 取值为 page。JSP 页面的运行效果如图 5.2 所示。

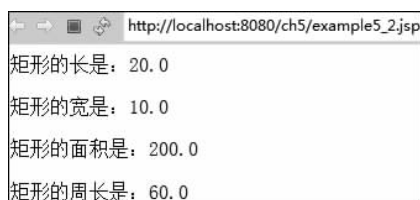


图 5.2 使用 bean 的 JSP 页面

例 5-2 页面文件 example5_2.jsp 的代码如下:

```
<%@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">
<%@page import="com.bean.Rectangle"%>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>example5_2.jsp</title>
</head>
<body>
    <jsp:useBean id="rectangle" class="com.bean.Rectangle" scope="page"/>
    <p>矩形的长是:<%=rectangle.getLength()%>
    <p>矩形的宽是:<%=rectangle.getWidth()%>
    <p>矩形的面积是:<%=rectangle.computerArea()%>
    <p>矩形的周长是:<%=rectangle.computerLength()%>
</body>
</html>
```

5.2.2 动作标记 getProperty

JavaBean 的实质是遵守一定规范的类所创建的对象,可以通过两种方式获取 bean 的属性:①在 Java 程序片或表达式中,使用 bean 对象调用 getXxx()方法获取 bean 的属性值,比如,例 5-2 中的语句: <%=rectangle.getLength()%>; ②先通过<jsp: useBean>标记获得一个 bean,再通过<jsp: getProperty>标记获取 bean 的属性值。

使用 getProperty 动作标记可以获得 bean 的属性值。使用该动作标记之前,必须事先使用 useBean 动作标记获得一个相应的 bean。getProperty 动作标记语法格式如下:

```
<jsp:getProperty name="bean 的名字" property="bean 的属性" />
```

或

```
<jsp:getProperty name="bean 的名字" property="bean 的属性"/></jsp:getProperty>
```

其中,name取值是bean的名字,与useBean标记中的id对应;property取值是bean的一个属性名字,与创建该bean的类的成员变量名对应。这条指令相当于在Java表达式中使用bean的名字调用getXxx()方法。

【例 5-3】 创建bean的源文件NewRectangle.java,该bean的作用是计算矩形的面积和周长。编写一个JSP页面useGetProperty.jsp,在该JSP页面中使用useBean标记创建一个名字是pig的bean,并使用getProperty动作标记获得pig的每个属性值。负责创建pig的类是NewRectangle类。JSP页面运行效果如图5.3所示。



图 5.3 使用getProperty标记获得bean的属性值

NewRectangle.java的代码如下:

```
package com.bean;

public class NewRectangle {
    double length;
    double width;
    double rectangleArea;
    double rectangleLength;
    public NewRectangle() {
        length = 20;
        width = 10;
    }
    public double getLength() {
        return length;
    }
    public void setLength(double length) {
        this.length = length;
    }
    public double getWidth() {
        return width;
    }
    public void setWidth(double width) {
        this.width = width;
    }
    public double getRectangleArea() {
        return length * width;
    }
    public double getRectangleLength() {
        return 2 * (width + length);
    }
}
```

```
}  
}
```

例 5-3 页面文件 useGetProperty.jsp 的代码如下：

```
<%@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">  
<%@page import="com.bean.NewRectangle"%>  
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
<title>useGetProperty.jsp</title>  
</head>  
<body>  
    <jsp:useBean id="pig" class="com.bean.NewRectangle" scope="page"/>  
    <%pig.setLength(30);%>  
    <%pig.setWidth(20);%>  
    <p>矩形的长是:<jsp:getProperty property="length" name="pig"/>  
    <p>矩形的宽是:<jsp:getProperty property="width" name="pig"/>  
    <p>矩形的面积是:<jsp:getProperty property="rectangleArea" name="pig"/>  
    <p>矩形的周长是:<jsp:getProperty property="rectangleLength" name="pig"/>  
</body>  
</html>
```

5.2.3 动作标记 setProperty

除了在 Java 程序片中使用 bean 对象调用 setXxx() 方法修改 bean 的属性值外,我们还可以使用 setProperty 动作标记修改 bean 的属性值。使用该动作标记之前,必须事先使用 useBean 动作标记获得一个相应的 bean。使用 setProperty 动作标记进行 bean 属性值的设置有以下三种方式。

1. 用表达式或字符串设置 bean 的属性

1) 用表达式设置 bean 的属性

```
<jsp:setProperty name="bean 的名字" property="bean 的属性" value="<%=  
expression%>" />
```

2) 用字符串设置 bean 的属性

```
<jsp:setProperty name="bean 的名字" property="bean 的属性" value="字符串" />
```

用表达式修改 bean 属性值时,表达式值的类型必须与 bean 的属性类型一致。用字符串修改 bean 属性值时,字符串会自动转化为 bean 的属性类型,不能转化成功的可能会抛出 NumberFormatException 异常。

2. 通过 HTTP 表单参数值设置 bean 的属性

```
<jsp:setProperty name="bean 的名字" property="*" />
```

通过 HTTP 表单参数值设置 bean 的属性时,表单参数的名字必须与 bean 属性的名字相同,服务器会根据名字自动匹配,类型自动转换。

3. 任意指定请求参数设置 bean 的属性

```
<jsp:setProperty name="bean 的名字" property="属性名" param="参数名"/>
```

可以根据自己的需要,任意选择传递的参数,请求参数名无须与 bean 属性名相同。

【例 5-4】 用表达式或字符串修改 bean 的属性。具体要求如下:

(1) 创建 bean 的源文件 Car.java,该 bean 的作用是描述小汽车的一些属性。

(2) 编写一个 JSP 页面 car.jsp,在该 JSP 页面中使用 useBean 标记创建一个名字是 smallCar 的 bean,其有效范围是 page,并使用动作标记修改、获取该 bean 的属性值。负责创建 smallCar 的类是 Car。JSP 页面运行效果如图 5.4 所示。



图 5.4 使用字符串或表达式修改 bean 的属性

Car.java 的代码如下:

```
package com.bean;

public class Car {
    String tradeMark;
    String number;
    public String getTradeMark() {
        return tradeMark;
    }
    public void setTradeMark(String tradeMark) {
        this.tradeMark = tradeMark;
    }
    public String getNumber() {
        return number;
    }
    public void setNumber(String number) {
        this.number = number;
    }
}
```

例 5-4 页面文件 car.jsp 的代码如下:

```
<%@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">
<%@page import="com.bean.Car"%>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>car.jsp</title>
</head>
```

```

<body>
    <jsp:useBean id="smallCar" class="com.bean.Car" scope="page"/>
    <%
        String carNo="京 A8888";
    %>
    <!--使用 setProperty 标记设置 smallCar 的属性 tradeMark 值为"宝马 X6"-->
    <jsp:setProperty property="tradeMark" name="smallCar" value="宝马 X6"/>
    <!--使用 setProperty 标记设置 smallCar 的属性 number 值为 carNo-->
    <jsp:setProperty property="number" name="smallCar" value="<%=carNo %>"/>
    汽车的品牌是:<jsp:getProperty property="tradeMark" name="smallCar"/>
    <br>汽车的牌号是:<jsp:getProperty property="number" name="smallCar"/>
</body>
</html>

```

【例 5-5】 通过 HTTP 表单参数值设置 bean 的属性。具体要求如下：

- (1) 编写 JSP 页面 inputCar.jsp 和 showCar.jsp。
- (2) 在 inputCar.jsp 页面中输入信息后提交给 showCar.jsp 页面显示信息。
- (3) JSP 页面中用到的 bean 是例 5-4 中 Car 类创建的。JSP 页面运行效果如图 5.5 所示。

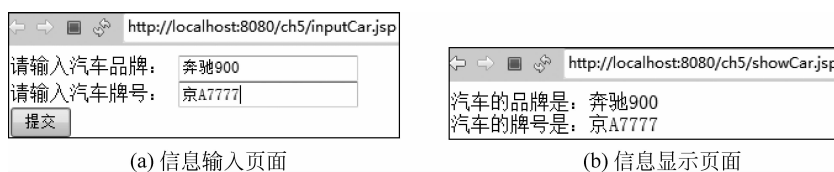


图 5.5 JSP 页面运行效果

例 5-5 页面文件 inputCar.jsp 的代码如下：

```

<%@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>inputCar.jsp</title>
</head>
<body>
    <form action="showCar.jsp" method="post">
        请输入汽车品牌:
        <input type="text" name="tradeMark"/>
        <br>
        请输入汽车牌号:
        <input type="text" name="number"/>
        <br>
    </form>

```



```

        <input type="submit" value="提交"/>
    </form>
</body>
</html>

```

例 5-5 页面文件 showCar.jsp 的代码如下：

```

<%@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">
<%@page import="com.bean.Car"%>
<%
request.setCharacterEncoding("utf-8");
%>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>showCar.jsp</title>
</head>
<body>
    <jsp:useBean id="smallCar" class="com.bean.Car" scope="page"/>
    <!--通过 HTTP 表单的参数的值设置 bean 的属性(表单参数与属性自动匹配)-->
    <jsp:setProperty property="*" name="smallCar"/>
    汽车的品牌是:<jsp:getProperty property="tradeMark" name="smallCar"/>
    <br>汽车的牌号是:<jsp:getProperty property="number" name="smallCar"/>
</body>
</html>

```

5.2.4 实践环节——使用 JavaBean 计算三角形的面积和周长

编写两个 JSP 页面：inputTriangle.jsp 与 showTriangle.jsp。inputTriangle.jsp 提供一个表单,用户可以通过表单输入三角形的三条边提交给 showTriangle.jsp。用户提交表单后,JSP 页面将计算三角形面积和周长的任务交给一个 bean 去完成,创建 bean 的源文件是 Triangle.java。页面运行效果如图 5.6 所示。

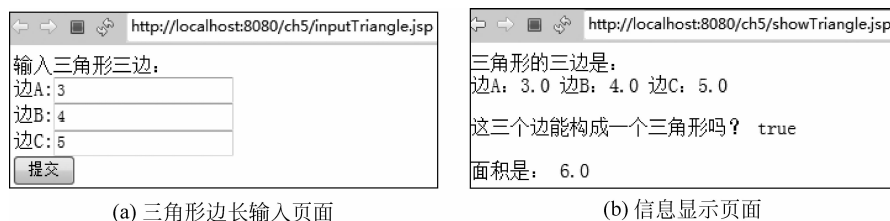


图 5.6 页面运行效果

5.3 本章小结

本章重点介绍了在 JSP 中如何使用动作标记 `useBean`、`getProperty` 和 `setProperty`。

习 题 5

- 下面哪一个是正确使用 JavaBean 的方式? ()
A. `<jsp: useBean id="address" class="tom. AddressBean" scope="page"/>`
B. `<jsp: useBean name="address" class="tom. AddressBean" scope="page"/>`
C. `<jsp: useBean bean="address" class="tom. AddressBean" scope="page" />`
D. `<jsp: useBean beanName="address" class="AddressBean" scope="page" />`
- JavaBean 中方法的访问属性必须是()。
A. private B. public C. protected D. friendly
- 在 JSP 中调用 JavaBean 时不会用到的标记是()。
A. `<javabeen>` B. `<jsp: useBean>`
C. `<jsp: setProperty>` D. `<jsp: getProperty>`
- JavaBean 的作用域可以是()、page、session 和 application。
A. request B. response C. out D. 以上都不对
- 在 test.jsp 文件中有如下一行代码: `<jsp: useBean class="tom. jiafei. Test" id="user" scope="___" />`要使 user 对象一直存在于会话中,直至终止或删除为止,下画线处应填入()。
A. page B. request C. session D. application
- 在 JSP 中,使用`<jsp: useBean>`动作可以将 javaBean 引入 JSP 页面,对 JavaBean 的访问范围不能是()。
A. page B. request
C. response D. application
- 下面语句与`<jsp: getProperty name="aBean" property="jsp"/>`等价的是()。
A. `<%=jsp()%>` B. `<%out. print(aBean. getJsp());%>`
C. `<%=aBean. setJsp()%>` D. `<%aBean. setJsp();%>`
- 以下是有关 `jsp: setProperty` 和 `jsp: getProperty` 标记的描述,正确的是()。
A. `<jsp: setProperty>`和`<jsp: getProperty>`标记都必须在`<jsp: useBean>`的开始标记和结束标记之间
B. 这两个标记的 name 属性值必须与`<jsp: useBean>`标记的 id 属性值相对应
C. 这两个标记的 name 属性值可以与`<jsp: userBean>`标记的 id 属性值不同
D. 以上均不对
- 在 JSP 中使用`<jsp: getProperty>`标记时,不会出现的属性是()。
A. name B. property C. value D. 以上皆不会出现