

## 14.4 Example: StringBean

[Listing 14.1](#) presents a simple class called `StringBean` that is in the `coreservlets` package. Because the class has no public instance variables (fields) and has a zero-argument constructor since it doesn't declare any explicit constructors, it satisfies the basic criteria for being a bean. Since `StringBean` has a method called `getMessage` that returns a `String` and another method called `setMessage` that takes a `String` as an argument, in beans terminology the class is said to have a `String` property called `message`.

[Listing 14.2](#) shows a JSP file that uses the `StringBean` class. First, an instance of `StringBean` is created with the `jsp:useBean` action as follows.

```
<jsp:useBean id="stringBean" class="coreservlets.StringBean" />
```

After this, the `message` property can be inserted into the page in either of the following two ways.

```
<jsp:getProperty name="stringBean" property="message" />
<%= stringBean.getMessage() %>
```

The `message` property can be modified in either of the following two ways.

```
<jsp:setProperty name="stringBean"
                 property="message"
                 value="some message" />
<% stringBean.setMessage("some message"); %>
```

Please note that we do not recommend that you really mix the explicit Java syntax and the XML syntax in the same page; this example is just meant to illustrate the equivalent results of the two forms.

### Core Approach



*Whenever possible, avoid mixing the XML-compatible `jsp:useBean` tags with JSP scripting elements containing explicit Java code.*

[Figure 14-2](#) shows the result.

### Listing 14.1 StringBean.java

```
package coreservlets;

/** A simple bean that has a single String property
 *  called message.
 */

public class StringBean {
    private String message = "No message specified";

    public String getMessage() {
        return(message);
    }
}
```

```

    public void setMessage(String message) {
        this.message = message;
    }
}

```

## Listing 14.2 StringBean.jsp

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE>Using JavaBeans with JSP</TITLE>
<LINK REL=STYLESHEET
      HREF="JSP-Styles.css"
      TYPE="text/css">
</HEAD>
<BODY>
<TABLE BORDER=5 ALIGN="CENTER">
  <TR><TH CLASS="TITLE">
    Using JavaBeans with JSP</TH>
</TR>
<TR>
  <TD>
    <jsp:useBean id="stringBean" class="coreservlets.StringBean" />
    <OL>
      <LI>Initial value (from jsp:getProperty):
        <I><jsp:getProperty name="stringBean"
                          property="message" /></I>
      <LI>Initial value (from JSP expression):
        <I><%= stringBean.getMessage() %></I>
      <LI><jsp:setProperty name="stringBean"
                        property="message"
                        value="Best string bean: Fortex" />
        Value after setting property with jsp:setProperty:
        <I><jsp:getProperty name="stringBean"
                          property="message" /></I>
      <LI><%= stringBean.setMessage("My favorite: Kentucky Wonder"); %>
        Value after setting property with scriptlet:
        <I><%= stringBean.getMessage() %></I>
    </OL>
  </TD>
</TR>
</TABLE>
</BODY></HTML>

```

**Figure 14-2. Result of StringBean.jsp.**



[ [Team LiB](#) ]

◀ PREVIOUS    NEXT ▶