

剖析JSP与XML的交互编程技巧Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/561/2021_2022__E5_89_96_E6_9E_90JSP_E4_c104_561663.htm

JavaServer Pages (JSP)和XML是Sun的J2EE的两个至关重要的组件。 JSP是用于创建应用程序服务器端程序的有效工具，而客户可以是一个浏览器，一个设备或其它的应用程序。 你可以使用XML描述数据并在联系服务器与系统的其余部分之间传递。 如果你仔细考虑Web服务的抽象概念的话，JSP可以被认为是实现技术而XML则是数据封装和消息传送技术。 JSP页面可以通过三种方式使用XML：直接使用XML文件，使用JavaBeans来执行XML处理或者通过标记库使用XML。 一、直接使用XML 我们可以在JSP页面中直接使用XML，这要分为三类： 1. JSP可以读取XML文件并且基于这些数据执行动作。例如：一应用程序可以读取具有某些特定结构的数据的XML文件。 2. JSP可以创建XML文件以发送数据到客户程序或其它的应用程序。 JSP可以转换XML文件,这个变换可以是交给XSLT处理，由JSP作为控制器，或者通过非XSLT解决方案来完成。在这两种情况下，JSP的作用都是读取XML文件，转换它并生成一个输出。 因为JSP包含内嵌的Java程序，所以它可以直接地调用一个分析程序来读/写xml数据。 这是非常不合理的方法，因为数据和代码逻辑并不能很好的被分离开。 另外，这样的程序也很难读得懂。 所以，下面我将介绍第二方法：使用JavaBeans。 二、使用JavaBeans JSP可以通过gt.标记与JavaBeans紧密的整合起来。 下面的程序片段示范了如何使用在JSP页面中的一个JavaBean来设置并获取属性。 gt. gt. gt. First Name is: gt. gt. gt.

gt. Last Name is: .gt. gt. JSP与JavaBeans整合的特征就是可以自动把超文本标志语言的表单元素翻译成JavaBean属性。如果有一个HTML表单并且想让它向JavaBean提交表单内容，你可以写下面的代码：gt. name属性包含JSP页面已经引用Bean的值。前面gt.标记设置名称为"cb"。与设置单独的Bean属性不同，你可以使用星号来标志"全部的"属性。JSP页面自动地映射HTML表单值为同名的Bean属性。如果你读取每个HTML表单元素，然后调用相应属性的Bean设置方法，那么结果也将是相同的。可以看到，类XML的标记允许JSP页面访问JavaBeans。我们通过尽可能多的把封装的代码转变成可重用组件(JavaBeans)，就可以把JSP页面中代码优化到最小的程度。你可以使用通用的语法分析程序，像Xerces或者JAXPI，在单独的JavaBeans里与XML文件交互作用--并且你可以在不改动JSP页面的情况下改变分析程序。此外，Beans还可以使用XSLT来执行XML文件的转换。使用JSP和JavaBeans来完成的这些抽象的动作比直接在JSP页面中插入原始的Java程序要好得多，但是你仍然需要熟悉Java程序，以便随时改变JSP页面。应用程序的一致性与条理性依靠JavaBeans合作创建一个统一的输出结果的好坏程度。例如，Bean中的缺陷可能会造成整个XML输出无效。依靠Beans指定资源的方法，可能同时带来执行性能的问题。

三、通过标记库JSP与XML交互 这也就是在前文中重点提到的，但是因为它实在太重要了，所以在本文中我必须也要提到。标记库可以定义出现在JSP页面中的作为类XML元素的自定义标签，可以把特定的Java代码与每个标记都关联起来。例如，假定你可以访问一个天气情况数据库，而且你需要输出现在的天气状况。那么，你可以

在JSP中插入JDBC程序代码直接查询数据库(虽然这并不是一个好的选择)，把这些代码封装到一个JavaBean内，或者把它包装成一个标记库。使用最后一个选择，你的JSP页面内的程序代码看上去就像：`<jt:Current weather is jt: />`。注意在上述程序代码中看不到任何Java代码的踪迹。作为一个页面设计人员，你使用一种熟悉的语法就像`<jt: />`，这些与它们的任何标记看起来非常相似。在页面中包含现在的天气状况的HTML字符串的地方插入它。标记库有一个关联的XML格式的描述符文件，名叫Tag Library Descriptor(标记库描述符，TLD)。在TLD文件中的每个标记都有一个条目，包括它的名称，版本和它的任选信息。在JSP页面内，你可以用`<jt: />`指令指定TLD文件。属性`prefix`用来指定一个前缀，用来在JSP页面内使用特定的库来引用任何标记。那为什么我们要使用标记`<jt: />`而不仅仅是`<jt: />`。TLD文件的精确位置依赖于正在使用的应用程序服务器。一个标记库标记可以代替为了完成这段程序逻辑的相应的Java程序代码。每个标记都相当于一个相同名称的Java类。这个类必须实现TagSupport接口，包含捕获事件触发器方法作为处理这个页面的JSP引擎。当它第一次遇到这个标记的时候，引擎就会调用`doStartTag()`方法。可以使这个方法为空或者当需要的时候才执行应用程序逻辑。当该方法返回SKIP_BODY时，那么引擎跳过这个标记体。当它返回EVAL_BODY_INCLUDE时，引擎将会处理这个标记以及它的子标记。类似地，JSP引擎在分析了结束标记之后就会调用`doEndTag()`方法。`doAfterBody()`方法让你可以在引擎处理元素体之后执行动作，但是必须在`doEndTag()`方法作用之前。下面就是实现天气状况的Weather类的一个样本程序代码：

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com