

辅导：在Web服务客户端应用J PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/474/2021_2022__E8_BE_85_E5_AF_BC_EF_BC_9A_E5_c67_474985.htm 当服务器获得对JSP页面的请求后，就启动一系列的事件生成Java servlet来响应。通过这种过程，JSP代码转换成为servlet，这也是JSP被开发人员广泛应用的关键。页面的JSP代码包含了普通的HTML标记和特殊的“tag”。普通的HTML可以被看作被tag的动作所填充的模版。专门的JSP编译器把JSP代码转换成有效的Java servlet代码，再用普通的Java编译器转换成servlet类。服务器则把JSP页面的名称映射到生成的servlet上。因此，对于Web设计人员来说，就好象JSP代码被执行一样，不过背后却做了很多事情。简单的"scriptlet" tag允许你在HTML标签中间加入Java语句。你还可以声明完整的Java方法，使其成为servlet代码的一部分。用这些方法，你可以访问任何可扩展的Java库，因此你可以构建十分复杂的JSP，不过这不见得好罢了。JSP tag是对Java代码的额外扩展，这些代码一般是手工创建起来很麻烦的。这样做不好的地方在于JSP比普通Java代码更难调试。Tag扩展了基本的JSP语法，它可以被组织在称为“taglibs”的tag库中。在使用时，必须被显式加入页面中。JSP，“Model 1” and “Model 2” JSP技术刚被使用时，程序员很乐意把所有的决策制定等类似功能加入到JSP代码中。这样做很容易，而且你的代码都在一个文件中。但是，随着应用不断变大，这种方法变得很难调式和维护。实际上，它变得非常可怕。最近，这种在JSP中集中处理的方式被成为“Model 1”。它只用在那些基于很少的用户请求就能做决策的最简单的

应用中。而更好的架构被称为"Model 2"，它是用servlet来检查所有用户的请求，做运算并决定用哪个JSP来显示结果。这种设计模式被称为“Model-View-Controller (MVC)”，即用JSP根据请求来提供HTML视图，而用servlet来做控制器。注意：MVC中的M是指底层的数据，在Model1和Model2中都是如此。于是我们很快就能发现，可以用JSP页面来直接访问Web服务，解析结果并控制表示。但是，假如页面获得很多用户交互，那么就会由于重复获得同样内容而产生大量的低效复制工作。对于那些最简单的少交互页面。在从Web服务获得数据后绘制JSP时，应该间接地工作。JSP标准Tag库很多业界专家在Java Community Process (JSR 52) 之下标准化了一套基本的tag，称为JSP Standard Tag Library (JSTL)。可是并非像名字那样简单。实际上，由于Java Servlet和JSP API标准的不断发展，现在存在多种版本的JSTL。最早的JSTL1.0版对应Servlet2.3版及JSP1.2版。Tomcat 4是实现了这种JSP API的参考servlet引擎。JSTL1.1版对应Servlet2.4版及JSP2.0版。Tomcat 5是实现了这种JSP API的参考servlet引擎。最新的Java Enterprise Edition 5拥有JSTL1.2版。据我所知，这个版本中的更新是实现了JavaServer Faces应用框架于JSP的集成。Java EE 5已在两周前进入了最终发布状态。用标准Tag库访问Web服务为了用JSP进行展示，我从Apache Software Foundation网站下载了JSTL1.1包。这个包包含了很多很多JSTL之外的东西，因为该开源组织在为JSP创建定制tag库方面一直都很积极。我用Tomcat5.5.9做servlet容器。我们使用的Web服务是关联到新闻Web服务的TechTarget的RSS种子。原先的RSS代表了Rich Site Summary，但今天，RSS则表示Really Simple Syndication。不

管怎样，RSS使用到很简单的XML文件格式，很适合用JSTL来解析。我的JSP页面用两个tag开始，它们指定了被使用的taglib和在tag中出现的命名空间的前缀。此外，还有一些标准的HTML标签。Test RSS Feed from Techtarget 接着，我们用tag来定义如何获得源文件，并用保存在局部变量“document”中的引用解析成标准Java XML文档对象。XML解析工作如下：我已经添加了行数以方便解释。第一行用xml taglib的“set”动作把“channel”变量设为XML文档的根元素。第二行把“description”tag的内容加到HTML标签中。第三行是用于格式化未标号清单的HTML标签。1. 2. 3. 4. 5. >6. 7. 8.

RSS“channel”元素包含代表每个新闻标题的“item”元素。每个“item”元素有一个“title”，一个指向源网站的“link”，一大段描述以及一些其它元素。Web开发人员可以用很多方法在不同的JSP中使用这些数据。本例中，我只用了一个简单的表示层。接着，我们用forEach tag把所有的“item”元素输出到文档中。遵循XPath格式的“@select”属性为item元素创建一个odelist.在第五行，我把“link”元素的内容嵌入到HTML标签中作为超连接。而在第六行，我用“title”元素的内容创建被显示的文本。用浏览器打开Tomcat Web服务器中的JSP页面，将看到标题为“Test RSS Feed from Techtarget，”的页面，以及包含RSS描述的段落和一系列标题。每个标题都是可以点击的指向对应页面的链接。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