JSP - XML DATA

http://www.tutorialspoint.com/jsp/jsp_xml_data.htm

When you send XML data via HTTP, it makes sense to use JSP to handle incoming and outgoing XML documents for example RSS documents. As an XML document is merely a bunch of text, creating one through a JSP is no more difficult than creating an HTML document.

Sending XML from a JSP:

You can send XML content using JSPs the same way you send HTML. The only difference is that you must set the content type of your page to text/xml. To set the content type, use the <%@page%> tag, like this:

```
<%@ page contentType="text/xml" %>
```

Following is a simple example to send XML content to the browser:

```
<%@ page contentType="text/xml" %>

<books>
   <book>
      <name>Padam History</name>
      <author>ZARA</author>
      <price>100</price>
   </book>
</books>
```

Try to access above XML using different browsers to see the document tree presentation of the above XML.

Processing XML in JSP:

Before you proceed with XML processing using JSP, you would need to copy following two XML and XPath related libraries into your <Tomcat Installation Directory>\lib:

- **XercesImpl.jar:** Download it from http://www.apache.org/dist/xerces/j/
- **xalan.jar:** Download it from http://xml.apache.org/xalan-j/index.html

Let us put following content in books.xml file:

```
<books>
<book>
  <name>Padam History</name>
  <author>ZARA</author>
  <price>100</price>
</book>
<book>
  <name>Great Mistry</name>
  <author>NUHA</author>
  <price>2000</price>
</book>
</books>
```

Now try the following main.jsp, keeping in the same directory:

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>

<html>
<head>
  <title>JSTL x:parse Tags</title>
</head>
<body>
<h3>Books Info:</h3>
<c:import var="bookInfo" url="http://localhost:8080/books.xml"/>
```

```
<x:parse xml="${bookInfo}" var="output"/>
<b>The title of the first book is</b>:
<x:out select="$output/books/book[1]/name" />
<br>
<b>The price of the second book</b>:
<x:out select="$output/books/book[2]/price" />

</body>
</html>
```

Now try to access above JSP using http://localhost:8080/main.jsp, this would produce following result:

Books Info:

The title of the first book is:Padam History
The price of the second book: 2000

Formatting XML with JSP:

Consider the following XSLT stylesheet style.xsl:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl=
"http://www.w3.org/1999/XSL/Transform" version="1.0">

<xsl:output method="html" indent="yes"/>

<xsl:template match="/">
  <html>
  <body>
   <xsl:apply-templates/>
  </body>
  </html>
</xsl:template>

<xsl:template match="books">
  <table border="1" width="100%">
    <xsl:for-each select="book">
      <tr>
        <td>
          <i><xsl:value-of select="name"/></i>
        </td>
        <td>
          <xsl:value-of select="author"/>
        </td>
        <td>
          <xsl:value-of select="price"/>
        </td>
      </tr>
    </xsl:for-each>
  </table>
</xsl:template>
</xsl:stylesheet>
```

Now consider the following JSP file:

```
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>

<html>
<head>
  <title>JSTL x:transform Tags</title>
</head>
<body>
```

```
<h3>Books Info:</h3>
<c:set var="xmltext">
  <books>
    <book>
      <name>Padam History</name>
      <author>ZARA</author>
      <price>100</price>
    </book>
    <book>
      <name>Great Mistry</name>
      <author>NUHA</author>
      <price>2000</price>
    </book>
  </books>
</c:set>

<c:import url="http://localhost:8080/style.xsl" var="xslt"/>
<x:transform xml="${xmltext}" xslt="${xslt}"/>

</body>
</html>
```

This would produce following result:

Books Info:

Padam History	ZARA	100
Great Mistry	NUHA	2000

For more detail on XML processing using JSTL, you can check JSP Standard Tag Library.