

JSTL XML <X:CHOOSE>, <X:WHEN>, <X:OTHERWISE> TAGS

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

Copyright © tutorialspoint.com

The <x:choose> works like a Java **switch** statement in that it lets you choose between a number of alternatives. Where the **switch** statement has **case** statements, the <x:choose> tag has <x:when> tags. A switch statement has **default** clause to specify a default action and similar way <x:choose> has <x:otherwise> as default clause.

Attribute:

- The <x:choose> tag does not have any attribute.
- The <x:when> tag has one attributes which is listed below.
- The <x:otherwise> tag does not have any attribute.

The <x:when> tag has following attributes:

Attribute	Description	Required	Default
select	Condition to evaluate	Yes	None

Example:

```
<%@ 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:choose 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>

<x:parse xml="{xmltext}" var="output"/>
<x:choose>
  <x:when select="$output//book/author = 'ZARA'">
    Book is written by ZARA
  </x:when>
  <x:when select="$output//book/author = 'NUHA'">
    Book is written by NUHA
  </x:when>
  <x:otherwise>
    Unknown author.
  </x:otherwise>
</x:choose>

</body>
```

```
</html>
```

This would produce following result:

Books Info:

Book is written by ZARA