

25/7/2020

## JSTL Formatting tags - `<fmt:formatDate>`

The output of the previous class example is as follows: -

### Different formats of the Date

Formatted time : 11:20:15 AM

Formatted Date : - Jul 24, 2020

Formatted Date and Time : - Jul 24, 2020 11:20:15 AM

Formatted Date and Time in short style : 7/24/20 11:20 AM

Formatted Date and time in medium style : - Jul 24, 2020 11:20:15 AM

Jul 24, 2020 11:20:15 AM IST

JSTL XML tags - These tags are used for creating xml documents. These tags provide flow control, transformation etc.

The url for the xml tag is -

<http://java.sun.com/jsp/jstl/xml>

The prefix is - **x**

The JSTL XML tag library has custom tags used for interacting with XML data. The syntax used for including JSTL XML tags library in our JSP is -

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

Please Note:- Before doing any example we have to download and copy Xalan.jar file from <http://xml.apache.org/xalan-j/index.html>

and XercesImpl.jar ~~and~~ from <http://www.apache.org/dist/xerces/j/>

Paste these files in lib folder of Tomcat installation directory.

These above jar files are needed to download and paste in the lib directory of our Apache Tomcat Server's installation directory.

Then we can able to run xml tags in our examples.

Now, The XML tags and their descriptions are given as :-

Tags

Descriptions

- x:out ——— Same as  $\langle \% \Rightarrow \dots \rangle$  tag, but used for XPath expressions.
- x:parse ——— It is used for parse the XML data specified either in the tag body or an attribute.
- x:set ——— It is used to sets a variable to the value of an XPath expression.
- x:choose ——— It is conditional tag that establish a context for mutually exclusive conditional operations.

x:when ————— It is a subtag of that- will include its body if the condition evaluated be 'true'.

x:otherwise ————— It is subtag of that- follows tags and runs only if all the prior conditions evaluated be 'false'.

x:if ————— It is used for evaluating the test XPath expression and if it is true, it will processes its body content.

x:transform — It is used in a XML document for providing the XSL transformation. (Extensible stylesheet Language)

x:param — It is used along with the transform tag for setting the parameter in the XSLT style sheet.

Now let us consider examples of every tags one by one :- JSP XML <x:out> tag :-

The <x:out> tag is used for displaying the result of an xml path expression and writes the result to JSP writer object. It is same as <%= --- %> tag used in JSP. Syntax :- <x:out attributes />

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> XML tag example </title>
```

```
</head> <body> <h2> Vegetable Information: </h2>
```

```
<c:set var="veg">
```

```
<Vegetables>
```

```
<vegetable> <v-eg>
```

```
<name> Potato </name>
```

```
<price> 35 per kg </price>
```

```
</v-eg>
```

```
<veg> <name> Onion </name>
```

```
<price> 20 per kg </price>
```

```
</veg>
```

```
<veg> <name> Tomato </name>
```

```
<price> 60 per kg </price>
```

```
</veg>
```

```
</Vegetables>
```

```
</c:set>
```

```
<x:parse xml="{ $ {veg} " var="output"/>
```

```
<b> Name of the vegetable is: </b>
```

```
<x:out select=" $output/vegetables/veg[1]/name"/>
```

```
<br> <b> Price of the Tomato is: </b>
```

```
<x:out select=" $output/vegetables/veg[3]/price"/>
```

```
</body> </html>
```

output is :-  
Name of the vegetable is: Potato  
Price of the Tomato is :- 60 per kg