Java Server Pages Standard Tag Library (JSTL)

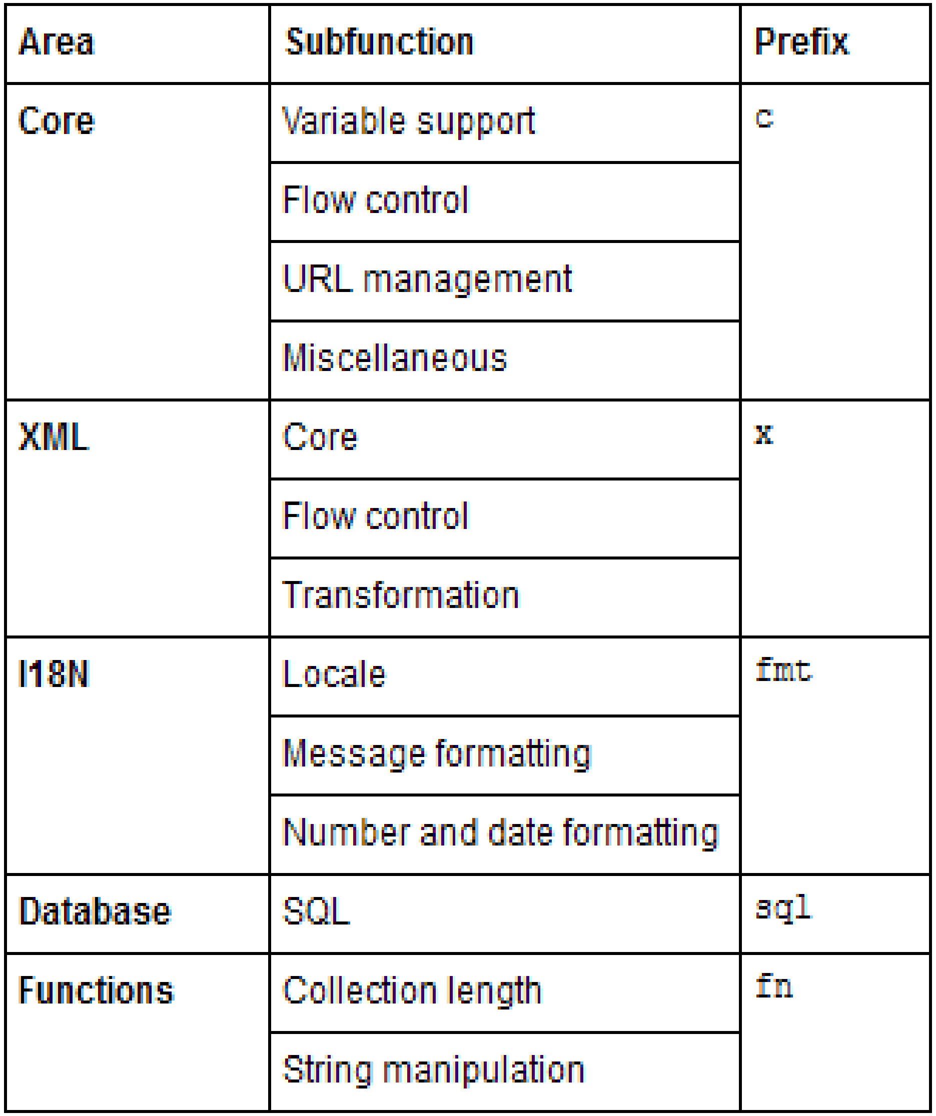
- JSTL is a collection of useful JSP tags which encapsulates core functionality common to many JSP applications.

- JSTL has support for common, structural tasks such as iteration and conditionals, tags for manipulating XML documents, internationalization tags, and SQL tags.

- It also provides a framework for integrating existing custom tags with JSTL tags

JSTL Tags

- The following JSTL tag library groups that can be used when creating a JSP page:
  - **Core Tags** : http://java.sun.com/jsp/jstl/core
  - **Formatting tags** : http://java.sun.com/jsp/jstl/fmt
  - **SQL tags :** http://java.sun.com/jsp/jstl/sql
  - **XML tags** : http://java.sun.com/jsp/jstl/xml
  - **JSTL Functions :** http://java.sun.com/jsp/jstl/functions



Area	Subfunction	Prefix
Core	Variable support	c
	Flow control	
	URL management	
	Miscellaneous	
XML	Core	x
	Flow control	
	Transformation	
I18N	Locale	fmt
	Message formatting	
	Number and date formatting	
Database	SQL	sql
Functions	Collection length	fn
	String manipulation	

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

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

```
<%@ taglib prefix="fmt"
uri="http://java.sun.com/jsp/jstl/fmt" %>
```

```
<%@ taglib prefix="sql"
uri="http://java.sun.com/jsp/jstl/sql" %>
```

```
<%@ taglib prefix="fn"
uri="http://java.sun.com/jsp/jstl/functions
" %>
```

JSTL Core Tags

- Core Tag provides support for iteration, conditional processing, and expression language for the tags in JSP pages.

Syntax of Core Tag :

- <%@ taglib uri="http://java.sun.com/jstl/core" prefix="c" %>

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Core Tag Example</title>
</head>
<body>
<c:forEach var="j" begin="1" end="3">
   Item <c:out value="${j}"/><p>
</c:forEach>
</body>
</html>
```

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
<head>
<title>Core Tag Example</title>
</head>
<body>
<c:set var="income" scope="session" value="${4000*4}"/>
<p>Your income is : <c:out value="${income}"/></p>
<c:choose>
    <c:when test="${income <= 1000}">
       Income is not good.
    </c:when>
    <c:when test="${income > 10000}">
        Income is very good.
    </c:when>
    <c:otherwise>
       Income is undetermined...
    </c:otherwise>
</c:choose>
</body>
</html>
```

JSTL XML Tags

- It is used to work with XML data such as parsing, selecting and transforming XML data in JSP page.

Syntax of **XML** Tag :

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

JSTL Formatting Tags

It is used for formatting the data used in JSP page. The data can be dates, numbers and time specifications for internationalized web sites.

Syntax of Format Tag Library :

- <%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %>

JSTL SQL Tags

- It provides SQL support. It is used to access relational data bases such as Oracle, MySql etc used in a JSP page.

Syntax of SQL Tag Library :

- <%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql" %>

JSTL Function Tags

- JSTL function tag library contains set of string manipulation functions that can be used with the JSP Expression Language.

Syntax of Function Tag Library :

- <%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>

Advantages of JSTL

- Easy to read and maintain.
- Reusability of the code.
- JSTL reduces use of scriptlets in a JSP page.
- JSTL allows using predefined tags instead of writing the java code.
- We can easily access and manipulate the application data without using the scriptlets.
- JSTL is easy to learn and understand without knowledge of Java programming.
- It easier to read because JSTL based on XML which is similar to HTML.

Disadvantages of JSTL

- Large complex JSP pages make use of JSTL tag libraries which exceeds class size limitation. Because size of java class is limited.

- JSTL is not so flexible and not as extensive as JSP Scriptlet. The JSTL is not nearly as evolved as Java itself.

- It is easier to programmer who knows HTML skill set. If JSTL get translated into Java code, experienced programmer may wonder why he does not do that himself and save the time. So it may difficult to carry out for experienced programmers.

Exception Handling

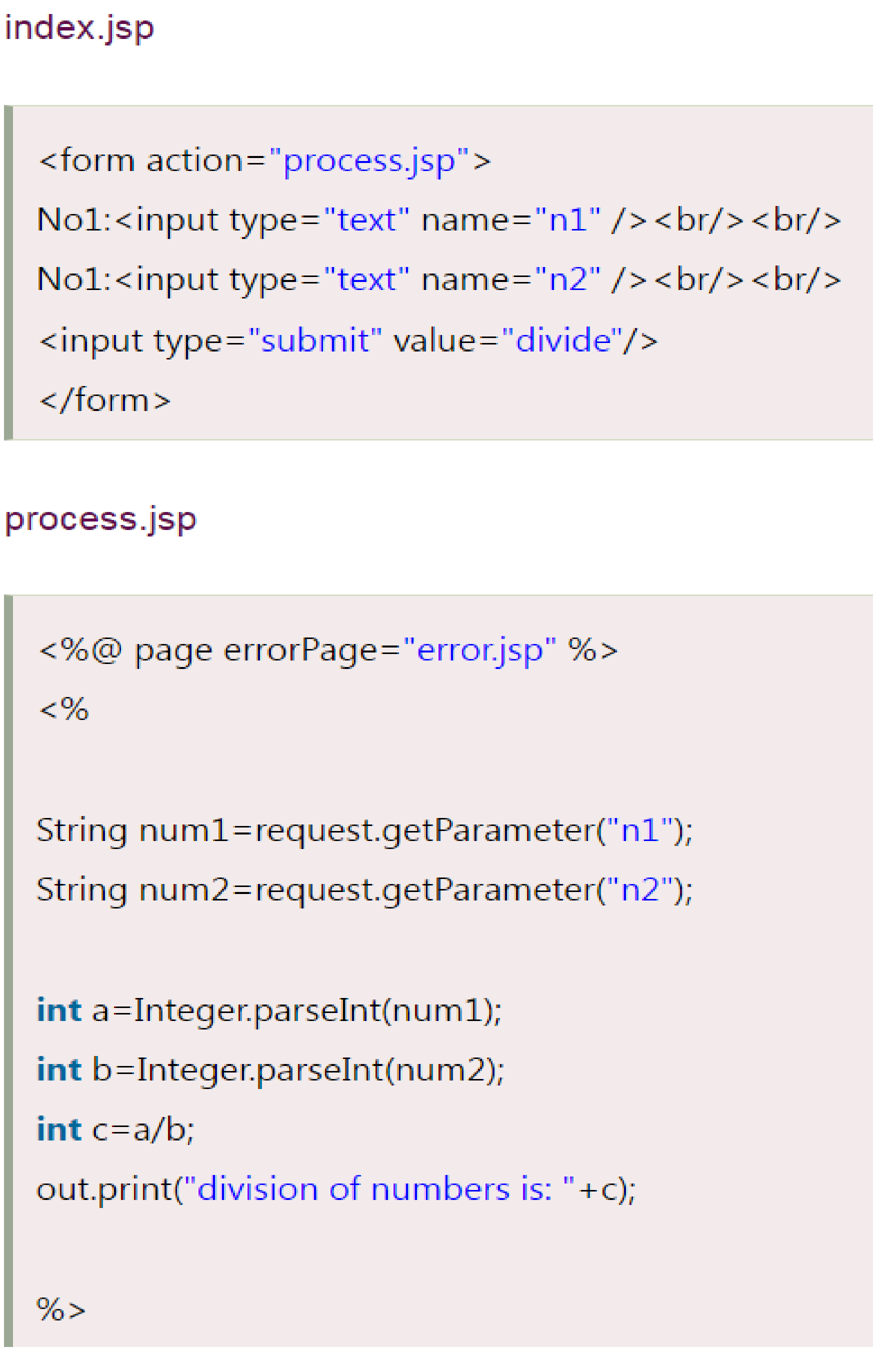
- The exception is normally an object that is thrown at runtime. Exception Handling is the process to handle the runtime errors.
- There may occur exception any time in your web application. So handling exceptions is a safer side for the web developer.
- In JSP, there are two ways to perform exception handling:
  - By **errorPage** and **isErrorPage** attributes of page directive
  - By **<error-page>** element in web.xml file

Page that may raise exception

- <%@ page errorPage="error.jsp" %>

- error.jsp

<%@ page isErrorPage="true" %>



index.jsp

```
<form action="process.jsp">
No1:<input type="text" name="n1" /><br/><br/>
No1:<input type="text" name="n2" /><br/><br/>
<input type="submit" value="divide"/>
</form>
```

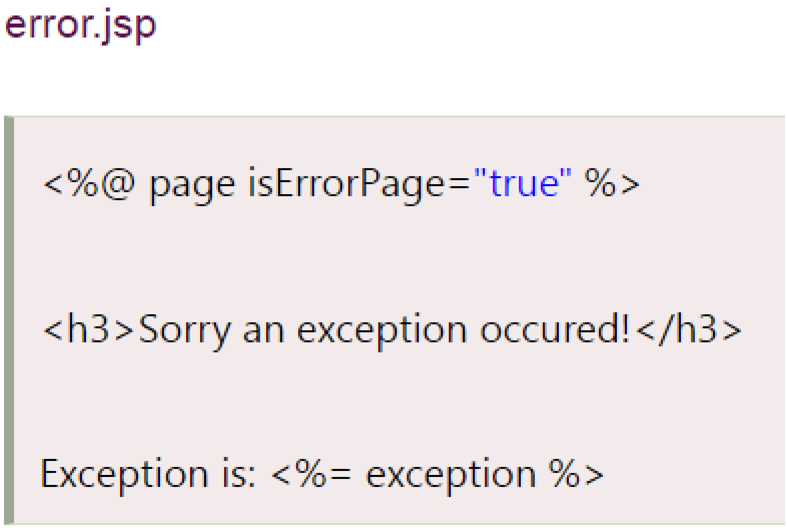
process.jsp

```
<%@ page errorPage="error.jsp" %>
<%

String num1=request.getParameter("n1");
String num2=request.getParameter("n2");

int a=Integer.parseInt(num1);
int b=Integer.parseInt(num2);
int c=a/b;
out.print("division of numbers is: "+c);

%>
```



error.jsp

```
<%@ page isErrorPage="true" %>

<h3>Sorry an exception occured!</h3>

Exception is: <%= exception %>
```

Exception handling through web.xml file

```
<web-app>

 <error-page>
  <exception-type>java.lang.Exception</exception-type>
  /* <error-code>500</error-code>  */
  <location>/error.jsp</location>
  </error-page>

</web-app>
```