JSP Standard Tag Library – JSTL

LEARNING OBJECTIVES

At the end of this lesson, you will be able to:

- Introduction to JSTL
- Core Tags

JSTL

- ➢ Provides standard library of tags
- ➢ Tags help in doing common tasks that need to be performed in a JSP
- ➢ Helps to reduce scripting from JSP pages
- ➢ Makes it easier to code and read, especially for web designers
- ➢ Before using the tags in a JSP, a taglib directive must be coded that identifies the JSTL Library and its prefix

Name	Prefix	URI	Description
Core	c	http://java.sun.com/jsp/jstl/core	Contains the core tags for common tasks such as looping and if/else statements.
Formatting	fmt	http://java.sun.com/jsp/jstl/fmt	Provides tags for formatting numbers, times, and dates so they work correctly with internationalization (i18n).
SQL	sql	http://java.sun.com/jsp/jstl/sql	Provides tags for working with SQL queries and data sources.
XML	x	http://java.sun.com/jsp/jstl/xml	Provides tags for manipulating XML documents.
Functions	fn	http://java.sun.com/jsp/jstl/functions	Provides functions that can be used to manipulate strings.

taglib directive and Core tags

➢ Provides the functionality to add a tag library

Syntax:
<%@taglib uri= "taglibURI " prefix= "tag prefix"%>

➢ Core Tags
- Most frequently used tags
- Helps in setting and removing variables from different scopes
- Helps in conditional branching and Looping
- Helps to rewrite URL and import resources from a URL
- Helps to redirect a request

The taglib directive that specifies the JSTL core library and its prefix
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

Using tags in the JSP
<p> <c:set var="salestax" scope="request" value="4"/>

Core Tags - Looping

forEach tag

- Used to loop through items of a Collection, including Arrays

  ```
  <c:forEach var="varName" items = "collection" />
  ```
- *items* attribute specifies the collection
- *var* attribute specifies the variable name used to access each item of collection

Servlet code

```
String[] countries = {"India", "Australia", "USA",
"France", "Oman"}
request.setAttribute("ctryList", countries);
```

JSP code

```
<c:forEach var="country" items="${requestScope.ctryList}">
  ...
  <tr>
   <td>  ${country} </td>
  </tr>
  ...
</c:forEach>
```

Core Tags - Looping

forEach Tag also has the following attributes
- begin – specifies the first index of the loop
- end   - specifies the last index of the loop
- step - specifies the amount to increment each time
- varStatus – variable which provides information about the status of the loop. It provides the first, last, count and index properties

```
<c:forEach var="country" items="${requestScope.ctryList}"
        begin="0" end= "3" step = "2"
        varStatus = "status">
 ...
  <tr>
  <td>

            Count: ${status.count} Index: ${status.index} :
${country}
   </td>
   </tr>
   ...
</c:forEach>
```

Core Tags – Conditional

➢ if tag
  - Performs conditional processing to change appearance of page depending on the test attribute values

      <c:if test = "*condition*" var = "varname" scope = "scope" >
          …body..
      </c:if>

      - If the condition in test is true the body is executed
      - *var* stores the result of condition
      - *scope* specifies the scope of *var* variable

    Consider a main page which has options to search and add products. All users can do search. If the User is admin he should be able to products

        <c:if test = "${user == 'admin'}">
             Add Product
        </c:if>
         Search Product

Core Tags – Conditional

➢ choose tag
- Performs conditional block execution based on the test result of the *when* subtags
- Executes the body of the first *when* tag whose test condition evaluates to true
- *otherwise* tag body is executed, If none of the test conditions of *when* tags evaluates to true

```
<c:choose>
    <c:when test="${user == 'admin'}">
            <a href="AddProduct.jsp">Add Product</a>
            <a href="EditProduct.jsp">Update Product</a>
            <a href="SearchProduct.jsp">Search Product</a>
    </c:when>
    <c:when test="${user == 'manager'}">
            <a href="EditProduct.jsp">Update Product</a>
            <a href="SearchProduct.jsp">Search Product</a>
    </c:when>
     <c:otherwise>
            <a href="SearchProduct.jsp">Search Product</a>
    </c:otherwise>
</c:choose>
```

8

Core Tags – URL related

➢ import tag:
  - Adds the content from specified *URL* attribute to the current page at *request* time

      <c:import url="header.jsp" >
              <c:param name="subTitle" value="Employee Search" />
      </c:import>

      <u>Equivalent standard action tag</u>
      <jsp:include page=" header.jsp " >
              <jsp:param name=" subTitle " value=" Employee Search "/>
      </jsp:forward>

  - More powerful than <jsp:include>
  - Can include content from outside the Container (other websites)

   <c:import url="http://victorshaw.files.wordpress.com/2010/05/sample.jpg" />

Core Tags – URL related

➢ url tag:
  - Used to rewrite URLs from a JSP page
  - Alternative method of rewriting URL using response.encodeURL() method
  - Can also be used to encode URL using c:param

```
JSP code with JSTL
<a href="<c:url value='/index.jsp' />">Continue Shopping</a>

Equivalent scripting
<a href="<%=response.encodeURL("index.jsp")%>">Continue Shopping</a>
```

```
<a href="<c:url value='/cart?productCode=${product.code}' />">
    Add To Cart
</a>
```

The same code with the JSTL param tag

```
<a href="
    <c:url value='/cart'>
        <c:param name='productCode' value='${product.code}' />
    </c:url>
">Add To Cart</a>
```

Core Tags – Variable Support

- ➢ set tag
  - Used to create and set values of scoped variables
  - Used to set Bean Properties and Map values
  - <jsp:setProperty> can only set Bean properties

  ```
  <c:set var="salestax" scope="request" value="4"/>
  <c:set var="userid" scope="session" value="${employee.userid}"/>
  <c:set target="${department}" property="deptName " >
                  ${param.deptName}
  </c:set >
  <c:set target="${capitalMap}" property="Karnataka" value="Bangalore" />
  ```

- ➢ remove tag
  - Used to remove variables from specified scope
    ```
    <c:remove var="salestax" scope="request" />
    ```

11

Core Tags

➤ out tag

   Displays the result of an expression

   Syntax:
```
<c:out value="value" [escapeXml="{true|false}"]
[default="defaultValue"] />
```

   Example

   <c:out value="${pageContext.session.id}" />

   <p>Employee Name <c:out value="${employee.name}" /> </p>

   <p>Hello <c:out value="${user}" default = "Guest" /> </p>

   <c:out value="<h1> This is header tag </h1>" />

• redirect tag

A redirect tag that redirects to another page

```
<c:if test="${e != null}">
    <c:redirect url="/error_java.jsp" />
</c:if>
```

SQL tags

➤ tags for interacting with relational databases

<sql:setDataSource> Creates a simple DataSource

<sql:query>   Execute SQL query

<sql:update>   Executes SQL DML Queries

<sql:param>   Sets a parameter in an SQL statement to the specified value.

```
<sql:setDataSource var="orclHr"
    driver="oracle.jdbc.OracleDriver"
    url="jdbc:oracle:thin:@localhost:1521:XE"
    user="HR" password="HR" scope="session" />
<sql:query dataSource="${orclHr}" var="results">
    SELECT * from Countries where region_id = ?
    <sql:param value="4" />
</sql:query>
<c:forEach var="ctry" items="${results.rows}">
        <c:out value="${ctry.COUNTRY_ID}" />
</c:forEach>
```

Formatting Tags

➢ used to format text, date, time and numbers

<fmt:formatNumber>   Formats numerical value with specific precision or format.

<fmt:parseNumber>   Parses the string representation of number, currency, or percentage.

<fmt:formatDate>   Formats a date and/or time using the supplied styles and pattern

<fmt:parseDate>   Parses the string representation of a date and/or time

<fmt:timeZone>   Specifies the time zone for any time formatting

<fmt:setLocale>   Stores the given locale in the locale configuration variable.

<fmt:formatNumber type="number" maxFractionDigits="2" value="${price}" />

<fmt:formatDate pattern="yyyy-MM-dd" value="${dob}" />

SUMMARY

JSP Standard Tag Library – JSTL

SUMMARY

In this lesson, you've learned to:

○ Introduction to JSTL
○ Core Tags