

CS 416

Web Programming

Core JSTL tag library

Dr. Williams
Central Connecticut State University

JSTL tag library

- Collection of JSP tags that perform common tasks preventing the need to create custom tags
- Examples
 - Conditional logic and iteration
 - Formatting
 - SQL for interacting with the database
 - XML tags for XML processing
- Used in the same way as custom tags

Including core JSTL

- Just like any other tag library the core JSTL tags must have an include statement as well
- For core tags reference java site as uri

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

Variables

- For accessing tags in the core JSTL library the standard tag prefix is “c”
- With core library you can define variable to be used within the page using:

```
<c:set var="myName" scope="session" value="${param.name}" />
```

- Similar to being able to define the scope of bean (page,request,session,application) can define scope of variable

Output

- The way variable should be output to the page is using:

`<c:out value="${myName}" />`

- Why this rather than just `${myName}`?
- By using `<c:out../>` the JSP will escape any special characters such as HTML characters

Redirection

- Similar to redirect within a servlet a JSP can redirect the browser using:

```
<c:redirect url="mypage.jsp"/>
```

If statements with JSTL

```
<c:if test="\${param.guess!='5'}">  
    You did not guess my number!
```

```
</c:if>
```

But there is no elseif or else using <c:if...

Logical expressions

- Logical expressions for `if` and `when` is similar to logical expression in Java `if` statement
 - “!” is not
 - “==” is equivalent
 - “&&” “||” and/or
- There are a couple of additional ones as well
 - “`empty myString`” checks the string and returns true if the string is either null or has a length of 0
 - “`E1?E2:E3`” check condition in `E1`, if true evaluates `E2` else evaluates `E3`

If else statements with JSTL

- The syntax for if statements within JSTL is created using 3 tags
 - `<c:choose>`
 - `<c:when test="logical expression">`
 - `<c:when test="logical expression">`
 - `<c:otherwise>`
 - `</c:choose>`
- `<c:choose>` indicates there will be a set of choices coming up defined by the when and otherwise tags
- `<c:when test="logical expression">` is the if statement with the test being the condition to be checked
- `<c:otherwise>` is equivalent to else

Choose syntax

```
<c:choose>
  <c:when test="\${empty name}">
    No name specified
  </c:when>
  <c:when test="\${name == 'John'}">
    John rocks!
  </c:when>
  <c:otherwise>
    <c:out value="Hi \${name}"/>
  </c:otherwise>
</c:choose>
```

Choose example

- EnterData.jsp
- MyExample.jsp (c:out,if, choose, redirect)

Loops

- Core JSTL provides functionality for performing loops using:

`<c:forEach var="item" items="${myCollection}">`

- This can be used for iterating through any `java.util.Collection`, array or `java.util.Map`
- Within the `forEach` tags the variable `item` can then be used `${item}`
- Note for Maps the `item` will be the Map Entry, you would then access the key using `${item.key}` and the value using `${item.value}`

Tokenizing

- To tokenize a string and loop through the values

```
<c:set var="myString" scope="request" value="Here is my delimited string" />
<c:forEach items="${myString}" delims=" ,|" var="myToken">
  <c:out value="Next token: ${myToken}" /><br/>
</c:forEach>
```

- var is the items is the String to be tokenized
- delims is a string containing each character that should be a delimiter
- var is the current token

MVC Example

- Example3Entry.jsp
- Example3Servlet (enhance and forward to JSP)
- Example3Results (tokenizing, for loops)

Catching exceptions

- Back at the beginning we talked about using `errorPage="error.jsp"` to handle exceptions, exceptions can also be handled within the page

```
<c:catch var="e">
```

```
    Some good content then create an  
exception<br/>
```

```
    <Some bad code>
```

```
    This is never reached because of the  
exception
```

```
</c:catch>
```

```
Content after catch
```

```
<c:out value="exception ${e}"/>
```

Formatting library

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

Provides functions:

- Internationalization
- Formatting output (Dates, numbers)
- Parsing text
- Internationalization functions allow you to set page content to a properties files and based on the user's locale retrieve content based on properties names

Internationalization

```
<fmt:setLocale value="en_US"/>  
  <fmt:message key='greeting'/><br/>  
  <fmt:message key='message'/><br/>  
<fmt:setLocale value="es_ES"/>  
  <fmt:message key="greeting"/><br/>  
  <fmt:message key="message"/><br/>
```

Hello

How are you?

Hola

¿Cómo estás?

Formatting/parsing

```
<fmt:parseDate value="04/02/1976" pattern="MM/dd/yyyy" var="myDate"/>
```

```
<fmt:parseNumber value="40,000.00" var="myNumber"/>
```

```
<fmt:formatDate value="${myDate}" pattern="M/d/yy"/><br/>
```

```
<fmt:formatNumber value="${myNumber}"/>
```

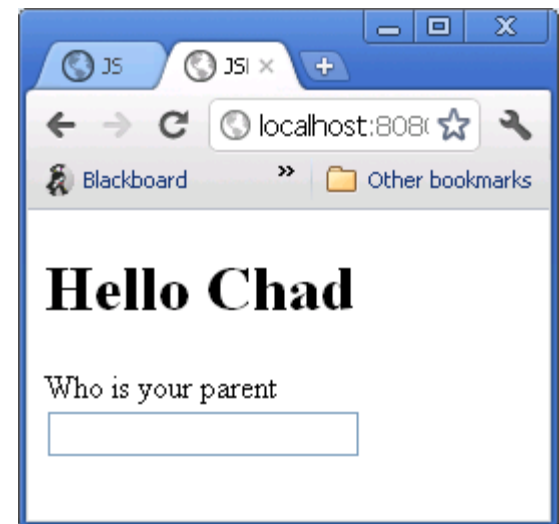
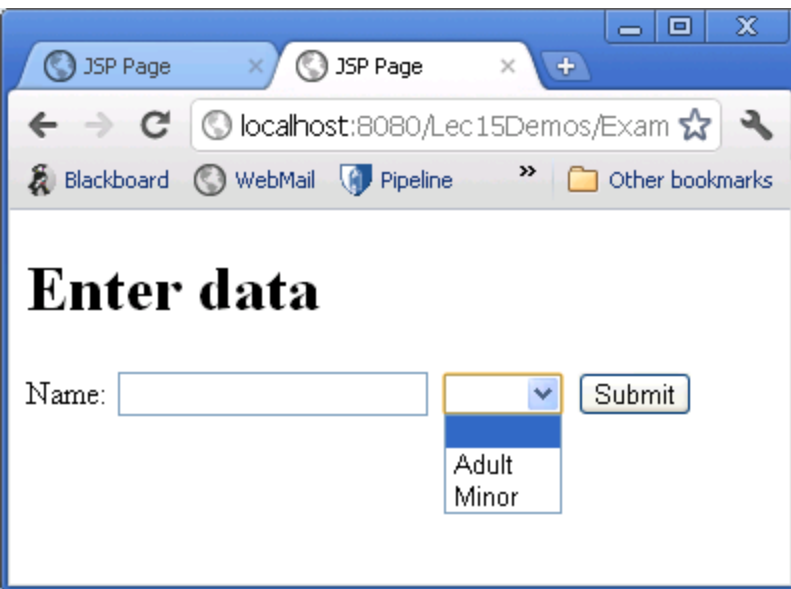
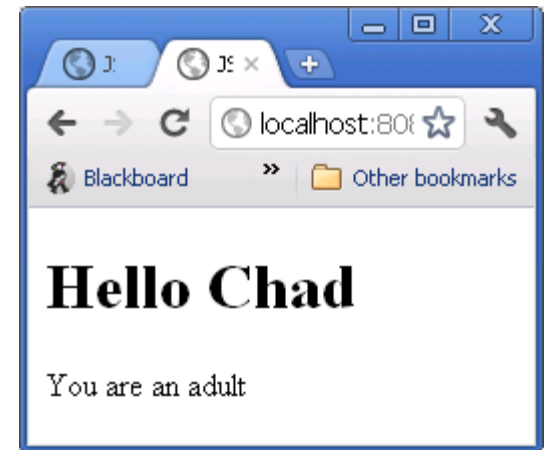
Example

- index.jsp – all elements

Example

Conditional display

- Use entry template Example2Entry.jsp
- Create Example2Results.jsp
- Use c:choose



Otherwise redirect to entry page

Free form - Halloween

- Create a page that submits 3 entries:
 - text field - person's name
 - Dropdown list – (2 options of costume)
 - Check box (3 options candy preference)
- Have the target page use when to display different content for entries of each option
 - For text field – if the entry matches your name display a special greeting (if statement)
 - Drop down list – have 2 options if option 1 is picked display 1 image if option 2 is picked display a different image (when statement)
 - Display each of the options picked in a table (for loop)

Design the app

- Bob's Moving Company wants you to design a web application that doesn't require refresh
 - As detailed estimates of belongings are provided complex calculations are performed to estimate num. boxes and moving van size
 - Based on van size needed, it looks up available dates and prices for those dates
 - Returns HTML rich site specific display of results
- **JUSTIFY YOUR DESIGN DECISIONS**

Revise the app

- CCSUMoveMe.com a moving company consolidator wants to be able to do the same as Kayak.com for moving quotes. They say Bob's can be added if they provide their same functionality through their site, but they want to control the display.
- How (if at all) would you change your architecture to meet this new need?