

BÀI 5:

Lập trình giao diện JSP với EL&JSTL

Mục tiêu bài học

- ◆ Giới thiệu về Java Standard Tag Library (JSTL)
 - ◆ Core Tag Library
 - ◆ SQL Tag Library
 - ◆ FMT Tag Library
- ◆ Giới thiệu về Expression Language (EL)
 - ◆ Ý nghĩa
 - ◆ Các toán tử
- ◆ Kết hợp giữa EL và JSTL trong lập trình giao diện JSP

Giới thiệu về JSTL

- ◆ JSP được giới thiệu lúc ban đầu là cho phép tách biệt giữa mã nguồn liên quan đến giao diện (UI-related code) và các đoạn mã nguồn liên quan đến nghiệp vụ (business logic-related code) trong các ứng dụng Web.
- ◆ Tuy nhiên, việc sử dụng scriptlets trong các trang JSP đã không đáp ứng được yêu cầu này.
- ◆ Việc sử dụng scriptlets sẽ làm ảnh hưởng đến các vấn đề readability và maintenance của trang JSP.
- ◆ Vì vậy, các nhà phát triển JSP đã đưa ra thư viện JSTL để có thể giải quyết những vấn đề trên.

Giới thiệu về JSTL

- ◆ JSTL là thư viện cung cấp tập hợp các thẻ đã được định nghĩa sẵn để thực thi các tác vụ phổ biến trong các trang JSP, chẳng hạn như:
 - ◆ Sử dụng các vòng lặp để duyệt qua một danh sách.
 - ◆ Định dạng kết quả của trang, quốc tế hóa.
 - ◆ Sử dụng các cấu trúc điều kiện
 - ◆ Kết nối và thao tác với database
 - ◆ Thao tác với XML

Giới thiệu Core Tag Library

- ◆ JSTL bao gồm nhiều thẻ thuộc các thư viện con.
- ◆ Các thẻ ở trong JSTL được phân loại thành các thư viện con dựa trên chức năng như sau:
 - ◆ Core tags
 - ◆ Formatting tags
 - ◆ Database tags
 - ◆ XML tags
- ◆ Thư viện core tags bao gồm các thẻ được sử dụng cho các tác vụ sau:
 - ◆ Sử dụng các cấu trúc điều kiện
 - ◆ Sử dụng các vòng lặp
 - ◆ Cài đặt URL rewriting
 - ◆ Redirecting người dùng
 - ◆ Cho phép khai báo, gán giá trị hay hủy biến

Giới thiệu Core Tag Library

- Ý nghĩa một số thẻ trong core tags lib:

Tên thẻ	Ý nghĩa
<code><c:out></code>	In ra giá trị của biến
<code><c:set></code>	Khai báo biến
<code><c:if></code>	Thể hiện cấu trúc if
<code><c:choose></code>	Thể hiện cấu trúc switch
<code><c:when></code>	Thể hiện một nhánh case trong cấu trúc switch
<code><c:otherwise></code>	Thể hiện nhánh else trong cấu trúc switch
<code><c:forEach></code>	Thể hiện một cấu trúc lặp
<code><c:redirect></code>	Cho phép redirect
<code><c:url></code>	Khai báo một url

Giới thiệu Core Tag Library

- ◆ Đoạn code minh họa việc sử dụng core tag trong một trang JSP:

```
<%@taglib prefix="c"
    uri="http://java.sun.com/jsp/jstl/core" %>
<html>
<head>
<title>Core Tags</title>
</head>
<body>
    <c:out value="Message shown using the <out> tag"/>
</body>
</html>
```

Giới thiệu Core Tag Library

- `<c:if test = "testcondition" var = "varName" scope = "page | request | session | application">`
Body content
`</c:if>`
- `<c:choose>`
`<c:when test = "testCondition">`
Body Content
`</c:when>`
`<c:otherwise>`
Body Content

Giới thiệu Core Tag Library

- `<c:forEach var="varName" item="collection" begin="begin" end="end" step="step">`
Body Content
- `</c:forEach>`
- `<c:forTokens items="stringofToken" delims="delimiters" var="varName" >`
Body Content
- `</c:forTokens>`

Giới thiệu về Formatting Tag Lib

- ◆ Các thẻ định dạng (Formatting tags) được dùng để định dạng và hiển thị văn bản, số, ngày tháng hoặc thời gian theo một định dạng cụ thể.
- ◆ Ngoài ra các thẻ nằm trong thư viện FMT còn được dùng để cài đặt cơ chế quốc tế hóa (i18n) cho trang JSP.
- ◆ Ý nghĩa của một số thẻ định dạng:

Tên thẻ	Ý nghĩa
<code><parseNumber></code>	Định dạng giá trị số
<code><formatDate></code>	Định dạng giá trị ngày tháng
<code><timeZone></code>	Thiết lập time zone
<code><setLocale></code>	Thiết lập locale cho trang

Giới thiệu về Formating Tag Lib

- ◆ Đoạn mã minh họa việc sử dụng thẻ định dạng trong trang JSP:

```
<%@taglib prefix="fmt"  
    uri="http://java.sun.com/jsp/jstl/fmt" %>  
<html>  
<head>  
<title>Formatting Tags</title>  
</head>  
<body>  
<fmt:formatNumber value="123000.4509" type="currency"/>  
</body>  
</html>
```

Đoạn mã trên đã định dạng giá trị số 123000.4509, thành kết quả \$123,000.45.

JSTL

Core tags 10

<c: >



uri = "http://java.sun.com/jsp/jstl/core"

1. Set [var,value,scope]
2. Remove [var]
3. out [value]
4. if [test]
5. when [test]
6. choose [elseif Ladder]
7. otherwise [else]
8. forEach [var,begin,end,step]
9. forEachTokens [val,delims,items]
10. redirect [uri]

SQL tags 3

<sql: >

1. setDataSource [var,driver,url,user,password]
2. query [dataSource,var,sql]
3. update [dataSource,var,sql]

XML tags 6

<x: >

1. out
2. if
3. choose
4. when
5. otherwise
6. parse

Formatting tags 2

<fmt: >

1. formatNumber
value
type='number |currency|percent'
currencySymbol
maxIntegerDigits minIntegerDigits
maxFractionDigits minFractionDigits
2. formatDate
value
type = "date|time|both"
dateStyle="short/medium/long/full"
timeStyle="short/medium/long/full"
pattern="h:m:s, d/M"

Functions tags 7

<functions: >

1. toUpperCase
2. toLowerCase
3. length
4. startsWith
5. endsWith
6. contains
7. containsIgnoreCase

Giới thiệu về SQL Tags Lib

- SQL tags được dùng để kết nối và thao tác với các CSDL quan hệ, như Microsoft SQL Server hoặc Oracle.
- Ý nghĩa của một số thẻ thuộc SQL Tag Lib:

Tên thẻ	Ý nghĩa
<code><sql:setDataSource></code>	Cho phép thiết lập datasource để kết nối đến CSDL
<code><sql:query></code>	Dùng để thực thi câu lệnh SQL truy vấn
<code><sql:param></code>	Dùng để thiết lập tham số cho câu lệnh SQL
<code><sql:dateParam></code>	Dùng để thiết lập tham số thuộc kiểu datetime
<code><sql:transaction></code>	Cho phép thiết lập giao dịch được sử dụng khi thực thi các thao tác với CSDL
<code><sql:update></code>	Dùng để thực thi câu lệnh SQL cập nhật dữ liệu

Giới thiệu về SQL Tags Lib

- ◆ Đoạn mã minh họa việc sử dụng SQL tag trong trang JSP:

```
<%@taglib prefix="sql"
    uri="http://java.sun.com/jsp/jstl/sql" %>
<html>
<head>
<title>Core Tags</title>
</head>
<body>
<sql:setDataSource var="ds"
    driver="sun.jdbc.odbc.JdbcOdbcDriver"
    url="jdbc:derby://localhost:1527/ProductDetails"
    user="niit" password="pass123"/>
```

Đoạn mã này dùng để thiết lập kết nối đến CSDL.

Giới thiệu về SQL Tags Lib

```
<sql:query dataSource="${ds}" var="result">  
SELECT * from Products;  
</sql:query>  
</body>  
</html>
```

Đoạn mã này dùng để truy vấn dữ liệu trong database.

Sử dụng EL trong JSP

- ◆ EL là một ngôn ngữ kịch bản cho phép lập trình viên có thể sử dụng các biểu thức đơn giản để có thể tính toán, truy cập và thao tác với các thành phần sau:
 - ◆ Dữ liệu của ứng dụng được lưu trữ trong JavaBeans.
 - ◆ Các đối tượng ngầm định (implicit objects).
 - ◆ Các lớp Java.
 - ◆ Các tập hợp (collections).
 - ◆ Các biến trong trang hoặc biến theo phạm vi (scoped variables).
- ◆ EL:
 - ◆ Được sử dụng trong trang JSP để cho phép tính toán giá trị của các biểu thức, hay truy cập vào các biến, các tập hợp v.v.
 - ◆ Thường được sử dụng kết hợp với JSTL.
- ◆ Các biểu thức EL có thể được sử dụng theo 2 cách sau:
 - ◆ Sử dụng như giá trị của thuộc tính
 - ◆ Sử dụng như văn bản trong trang JSP

Sử dụng EL trong JSP

- ◆ EL cung cấp các đối tượng ngầm định (implicit objects), những đối tượng này có thể được sử dụng để truy cập đến các implicit objects và các biến phạm vi (scoped variables) của trang JSP.
- ◆ Dưới đây là một số đối tượng ngầm định của EL (EL implicit objects):
 - ◆ `pageContext`
 - ◆ `pageScope`
 - ◆ `requestScope`
 - ◆ `sessionScope`
 - ◆ `applicationScope`
 - ◆ `param`
 - ◆ `paramValues`
 - ◆ `header`
 - ◆ `headerValues`
 - ◆ `initParam`
 - ◆ `cookie`

Sử dụng EL trong JSP

- ◆ Đoạn code bên dưới minh họa việc lấy về giá trị của tham số khởi tạo bằng cách sử dụng một EL implicit object:

```
My Email is: ${initParam.myEmail}
```

- ◆ Đoạn code sau minh họa việc lấy thông tin từ một cookie có tên là `yourname` bằng cách sử dụng đối tượng ngầm định `cookie` của EL:

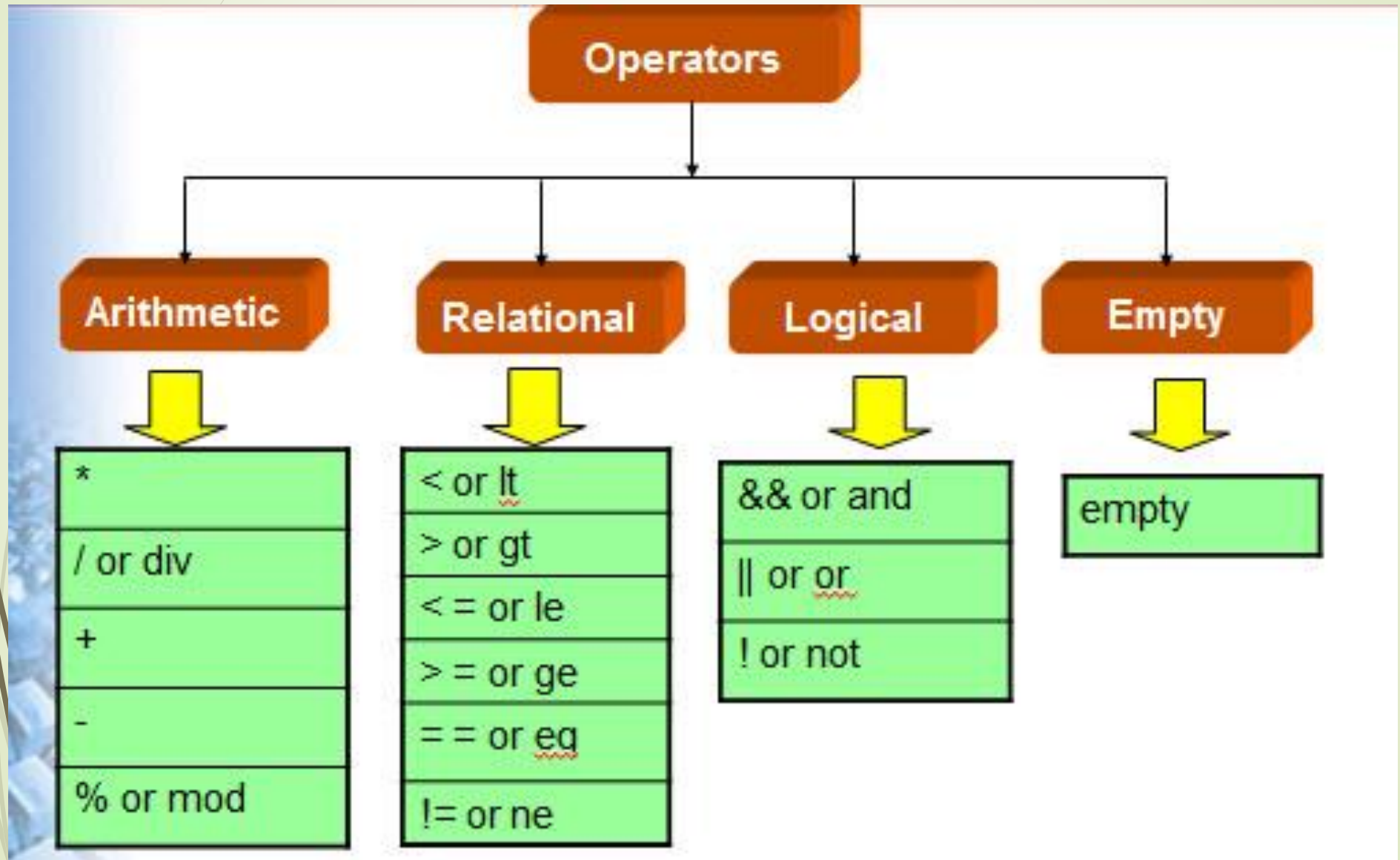
```
${cookie.yourname.value}
```

Sử dụng EL trong JSP

- ◆ EL cũng có thể được dùng để truy cập đến các JavaBeans.
- ◆ JavaBean:
 - ◆ Là một lớp Java đơn giản cung cấp các thuộc tính private cùng với các cặp phương thức getter/setter tương ứng.
 - ◆ Là một thành phần phần mềm cho phép tái sử dụng, có khai thác được điểm mạnh về bảo mật, cũng như platform-independent của Java.
 - ◆ Có thể được nhúng vào trang JSP để xử lý request của người dùng.
- ◆ Đoạn mã dưới đây minh họa việc thiết lập và lấy về giá trị của thuộc tính `firstName` từ một bean có tên là `customer`:

```
<jsp:setProperty name="customer" property="firstName" value="Sam" />
<h2>${customer.firstName}</h2>
```


Các toán tử trong EL



Minh họa

- ◆ Demo và thực hành EL&JSTL

- ◆ Giới thiệu về Java Standard Tag Library (JSTL)
 - ◆ Core Tag Library
 - ◆ SQL Tag Library
 - ◆ FMT Tag Library
- ◆ Giới thiệu về Expression Language (EL)
 - ◆ Ý nghĩa
 - ◆ Các toán tử
- ◆ Kết hợp giữa EL và JSTL trong lập trình giao diện JSP