



EL & JSTL

ThS. Nguyễn Nghiệm
0913.745.789 - NghiemN@fpt.edu.vn



MỤC TIÊU

- Sử dụng EL để truy xuất attribute trong từng scope
- Sử dụng EL để truy xuất bean, list, map, param, cookie, header
- Sử dụng JSTL để lập trình giao diện phía server





NỘI DUNG

- **EL: truy xuất các thành phần trong scope**
 - ✱ Attribute
 - ✱ Thuộc tính bean
 - ✱ Phần tử trong Collection
 - ✱ Phần tử trong Map
 - ✱ Tham số, cookie và header
- **JSTL: Lập trình giao diện phía server**
 - ✱ Thư viện thẻ core
 - `<c:set>`, `<c:remove>`
 - `<c:if>`, `<c:choose>`, `<c:forEach>`
 - ✱ Thư viện thẻ định dạng
 - `<fmt:formatNumber>`
 - `<fmt:formatDate>`
 - ✱ Thư viện hàm hỗ trợ EL





EL (EXPRESSION LANGUAGE)

- EL Được sử dụng để đơn giản hóa việc truy xuất attribute, param, cookie và header trong JSP
- Cú pháp:
 - ✱ `${biểu thức EL}`
- Ví dụ:
 - ✱ `${xyz}`: Xuất attribute trong scope
 - ✱ `${cookie['xyz'].value}`: Xuất giá trị cookie
 - ✱ `${param.xyz}`: Xuất giá trị tham số



TRUY XUẤT ATTRIBUTE

● Truy xuất attribute theo scope

- ✱ `${pageScope.xyz}` hoặc `${pageScope['xyz']}`
- ✱ `${requestScope.xyz}` hoặc `${requestScope['xyz']}`
- ✱ `${sessionScope.xyz}` hoặc `${sessionScope['xyz']}`
- ✱ `${applicationScope.xyz}` hoặc `${applicationScope['xyz']}`

● Tìm và truy xuất attribute

- ✱ `${xyz}`: Truy tìm attribute xyz theo thứ tự ưu tiên:
 - Page->Request->Session->Application



EL THƯỜNG GẶP

- Truy xuất thuộc tính bean
 - ✱ `${bean.property}`
- Truy xuất phần tử thứ i của list
 - ✱ `${list[i]}`
- Truy xuất phần tử của map
 - ✱ `${map[key]}` hoặc `${map.key}`
- Truy xuất tham số
 - ✱ `${param[name]}` hoặc `${param.name}`
- Truy xuất cookie
 - ✱ `${cookie[name].value}` hoặc `${cookie.name.value}`

Truy xuất attribute, param, cookie

ĐỀ MÔ



JSTL (JAVA STANDARD TAG LIBRARY)

- JSTL là các thư viện thẻ, hỗ trợ lập trình giao diện phía server.
- Các thư viện thẻ phổ biến
 - ✱ Thư viện cơ bản
`<%@taglib uri="http://java.sun.com/jstl/core_rt" prefix="c" %>`
 - ✱ Thư viện định dạng
`<%@taglib uri="http://java.sun.com/jstl/fmt_rt" prefix="fmt" %>`
 - ✱ Thư viện hàm
`<%@taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn"%>`
- Cần nhúng chỉ thị thẻ đầu trang jsp trước khi sử dụng



- **<c:if>**

- ✱ Tương tự lệnh if

- **<c:choose>**

- ✱ Tương tự if...else if...else

- **<c:forEach>**

- ✱ Tương tự for-each

- **<c:set>**

- ✱ Tương tự: scope.setAttribute()

- **<c:remove>**

- ✱ Tương tự scope.removeAttribute()



THẺ <C:IF>

```
<c:if test="${!empty user}">  
    Welcome ${user.fullname}  
</c:if>
```



THE <C:CHOOSE>

```
<ul>
  <c:choose>
    <c:when test="{empty user}">
      <li><a href="account/register.php">Register</a></li>
    </c:when>
    <c:otherwise>
      <li><a href="account/edit-profile.php">Edit Profile</a></li>
    </c:otherwise>
  </c:choose>
</ul>
```



<C:FOREACH>

```
<c:forEach var="p" items="{products}">
  <ul>
    <li>${p.name}</li>
    <li>${p.unitPrice}</li>
  </ul>
</c:forEach>
```



<C:FOREACH>

```
<c:forEach var="p" items="{products}"  
  begin="10" end="19" varStatus="st">  
  <ul>  
    <li>${st.index}</li>  
    <li>${p.name}</li>  
    <li>${p.unitPrice}</li>  
  </ul>  
</c:forEach>
```



<C:SET> & <C:REMOVE>

```
<c:set var="message" value="Hello Java"/>
```

```
<c:set var="message" value="Hello Java" scope="session"/>
```

```
<ul>
```

```
<li>${message}</li>
```

```
<li>${pageScope.message}</li>
```

```
<li>${sessionScope.message}</li>
```

```
</ul>
```

```
<c:remove var="message" scope="message"/>
```

Cơ bản

Liệt kê hàng hóa

ĐỀ MÔ



● Định dạng số

✱ `<fmt:formatNumber value="1000000" type="currency" />`

✱ `<fmt:formatNumber value="0.51" type="percent" />`

● Định dạng thời gian

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



JSTL FUNCTIONS

- `${fn:toUpperCase(s)}`
- `${fn:length(s)}`
- `${fn:substring(s, 10, 20)}`
- `${fn:substringAfter(s, 'VN')}`
- `<c:if test="${fn:contains(s, 'VN')}"></c:if>`
- `<c:if test="${fn:startsWith(s, 'VN')}"></c:if>`
- `<c:forEach var="item" items="${fn:split(s, '~')}">`



THƯ VIỆN HÀM

Tên hàm	Đối số	Trả về	Mô tả mục đích
fn:contains	String, String	boolean	Chuỗi (1) có chứa chuỗi (2) hay không
fn:containsIgnoreCase	String, String	boolean	Chuỗi (1) có chứa chuỗi (2) hay không (không phân biệt hoa thường)
fn:endsWith	String, String	boolean	Chuỗi (1) có kết thúc bởi (2) hay không
fn:escapeXML	String	String	Mã hóa thành thực thể các ký tự phạm cú pháp XML
fn:indexOf	String, String	int	Tìm vị trí xuất hiện đầu tiên của chuỗi (2) trong chuỗi (1)
fn:join	String[], String	String	Gia nhập các phần tử trong mảng (1) thành chuỗi sử dụng chuỗi(2) như là chuỗi phân cách.
fn:length	Map; array; Collection; Iterator; Enumeration; or String	int	Tìm độ dài của chuỗi hay số lượng các phần tử trong tập hợp.



THƯ VIỆN HÀM

Tên hàm	Đối số	Trả về	Mô tả mục đích
fn:replace	String, String, String	String	Thay thế chuỗi (1) bởi chuỗi (3) trong chuỗi (1)
fn:split	String, String	String[]	Tách chuỗi (1) thành mảng sử dụng chuỗi (2) như chuỗi phân cách
fn:startsWith	String, String	boolean	Chuỗi đối số thứ nhất có bắt đầu bởi chuỗi đối số thứ hai hay không
fn:substring	String, int, int	String	Lấy chuỗi trong chuỗi (1) tính từ vị trí (1) cho đến vị trí (3)
fn:substringAfter	String, String	String	Lấy chuỗi con trong chuỗi (1) đứng sau chuỗi (2)
fn:substringBefore	String, String	String	Lấy chuỗi con trong chuỗi (1) đứng trước chuỗi (2)
fn:toLowerCase	String	String	Đổi chuỗi sang chữ thường
fn:toUpperCase	String	String	Đổi chuỗi sang chữ HOA
fn:trim	String	String	Cắt bỏ khoảng trắng 2 đầu chuỗi

Định dạng giá, giảm giá, ngày nhập
Xử lý chuỗi, số lượng

ĐỀ MÔ



TÓM TẮT

● EL

- ✱ Truy xuất attribute
- ✱ Truy xuất thuộc tính bean
- ✱ Truy xuất Collection
- ✱ Truy xuất Map
- ✱ Truy xuất Parameter, cookie và header

● JSTL

- ✱ Thư viện thẻ core
 - <c:set>, <c:remove>
 - <c:if>, <c:choose>, <c:forEach>
- ✱ Thư viện thẻ định dạng
 - <fmt:formatNumber>
 - <fmt:formatDate>
- ✱ Thư viện hàm

