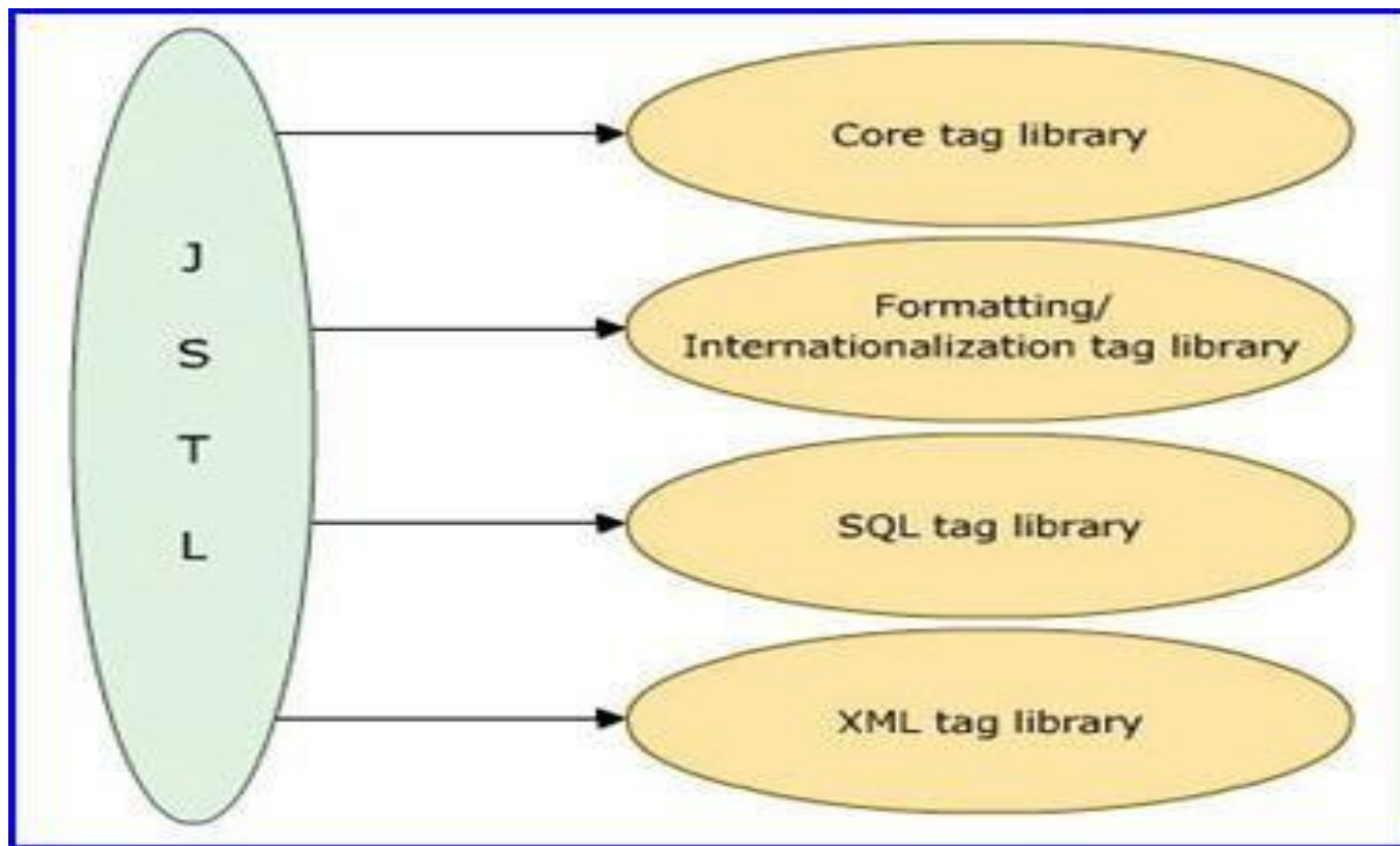




Bài 5 JSTL

- ❖ Tổng quan về JSTL
- ❖ Các thẻ chuẩn jstl core
- ❖ Các thẻ định dạng dữ liệu jstl formatting
- ❖ Các thẻ tương tác với CSDL jstl sql
- ❖ Các thẻ thao tác với kiểu dữ liệu xml jstl xml

- JSTL cung cấp nhiều các thư viện được sử dụng để thực thi nhiều chức năng trong các trang jsp



Tích hợp các mã xử lý Java vào trong các trang JSP mà không cần phải viết các đoạn scriptlet phức tạp

Bao gồm bộ các thẻ chuẩn cung cấp gần như đầy đủ các yêu cầu để xử lý các hành động trong các trang jsp

Các ứng dụng nhỏ gọn có thể chỉ cần sử dụng các thẻ có sẵn của JSP để xây dựng nên

Việc chuyển đổi ngôn ngữ và định dạng dữ liệu được thực hiện thông qua các thẻ của JSTL

Cung cấp các kỹ thuật cho phép các lập trình viên tự xây dựng bộ thẻ riêng của mình

❖ Thư viện thẻ Core

- ❑ Chức các thẻ lặp, tính toán biểu thức, và vào/ra cơ bản

- ❑ Được khai báo trong trang jsp bằng chỉ thị xử lý:

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

- ❑ Thư viện thẻ core bao gồm các thẻ:

- Thẻ mục đích chung
- Thẻ tạo quyết định
- Thẻ lặp

➤ Được sử dụng để thiết lập, loại bỏ và hiển thị giá trị các biến được tạo bên trong một trang jsp

❖ Gán giá trị cho 1 biến trong 1 phạm vi

```
<c:set var = "varName" value = "expression" scope =  
"page/request/ session/ application"/>
```

```
<c:set var = "sessionvariable" value ="${80+8}" scope = "session" />
```

❖ Hủy biến đã được gán bởi thẻ <c:set ...>

```
<c:remove var = "varName" scope =  
"page/request/session/application"/>
```

```
<c:remove var="simple" scope="page"/>
```

❖ Hiển thị giá trị của một biến

```
<c:out value = "value" escapeXml = "boolean"  
default = "defaultValue"/>
```

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

❖ Điều khiển chức năng ngoại lệ try..catch

```
<c:catch [var="varName"]>
    . . .
</c:catch>
```

Ví dụ:

```
<body>
<c:catch var="e">
    100 divided by 0 is
    <c:out value="${100/0}" /><br />
</c:catch>
```

❖ Thẻ tạo quyết định - if

```
<c:if test = "testCondition" var = "varName" scope =
"page/request/session/application">
    Body Content
</c:if>
```

❖ Ví dụ thẻ tạo quyết định - if

```
<!-- Test the username -->
<c:if test="{name=='admin'}">
    <h4> Valid User </h4>
</c:if>
```

❖ Thẻ tạo quyết định - choose

```
<c:choose>
    <c:when test = "testcondition"> Body
        Content
    </c:when>
        . . .
        . . .
    <c:otherwise>
        Body Content
    </c:otherwise>
</c:choose>
```


❖ Ví dụ thẻ tạo quyết định - choose

```
// Performs the checking of the condition
<c:choose>
  <c:when test="{name == 'admin'}">
    <h4> Valid User </h4>
  </c:when>
  <c:otherwise>
    <h4> Invalid User </h4>
  </c:otherwise>
</c:choose>
```

❖ Thẻ lặp - forEach

```
<c:forEach var = "varName" item = "collection" varStatus =
"varStatusName" begin = "begin" end = "end" step = "step">
  Body Content
</c:forEach>
```

❖ Ví dụ thẻ lặp - forEach

```
// Displays objects from collection  
"companies"  
<c:forEach var= "company" items = ${companies}>  
    ${company} <br>  
</c:forEach>  
  
// Displays values from 10 to 100  
<c:forEach var="i" begin="10" end="100">  
    ${i}  
</c:forEach>
```

❖ Thẻ lặp - forTokens

```
<c:forTokens items = "stringofToken" delims = "delimiters" var =  
"varName" varStatus = "varStatusName" begin = "begin" end = "end"  
step = "step">  
    Body Content  
</c:forTokens>
```

❖ Ví dụ thẻ lặp - forTokens

```
// Displays each token at a time separated by  
// \,' delimiter  
<c:forTokens var="token" items="Tom,Dick,Harry" delims=", ">  
  
    <c:out value="{token}"/>  
</c:forTokens>
```

❖ Thư viện thẻ định dạng dữ liệu, chuyển đổi ngôn ngữ

- ❑ Chứa các thẻ để định dạng dữ liệu như ngày tháng, số, tiền tệ, phần trăm.
- ❑ Được khai báo bằng chỉ thị xử lý:

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

❖ Định dạng ngày tháng

```
<fmt:formatDate type="..." dateStyle="..." value="..." ></c:forEach>
```

```
<fmt:formatDate type="date" value="${birthday}"/>
```

❖ Định dạng số, tiền tệ, phần trăm

```
<fmt:formatNumber type="..." dateStyle="..." value="..." ></c:forEach>
```

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

❖ Thư viện thẻ SQL

- ☐ Chứa các thẻ để truy xuất các CSDL SQL
- ☐ Cung cấp một giao diện để thực thi các truy vấn SQL trong CSDL qua trang JSP.
- ☐ Được khai báo bằng chỉ thị xử lý:

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

❖ Bao gồm các thẻ:

- ☐ `<sql:setDataSource>`
- ☐ `<sql:query>`
- ☐ `<sql:update>`
- ☐ `<sql:transaction>`
- ☐ `<sql:param>`

❖ Cú pháp:

```
<sql:setDataSource dataSource = "datasource" | url  
= "jdbcurl"  
driver = "driverclassdriver"  
user = "username" password = "userpwd" var = "varname"  
scope = "page/request/session/application"/>
```

```
// Sets the data source for the database
```

```
<sql:setDataSource
```

```
driver="com.microsoft.jdbc.sqlserver.SQLServerDriver"
```

```
url="jdbc:microsoft:sqlserver://10.1.3.27:1433;DataBaseName=pubs;" user="sa"
```

```
password="playware" var="conn"/>
```

❖ Cú pháp:

```
<sql:query sql = "sqlQuery" var = "varName" scope  
= "{page|request|session|application}" dataSource  
= "dataSource" maxRows = "maxRows" startRow = "startRow"/>
```

```
// Gets the records about the 'product' from datasource 'conn'  
<sql:query var="products" dataSource="${conn}"> select * from Products  
</sql:query>
```

❖ Cú pháp:

```
<sql:update dataSource = "dataSource" var = "varName" scope =  
"{page|request|session|application}">
```

SQL Statement

```
<sql:param value = "value"/>
```

```
</sql:update>
```

```
// Adds the product details in the Product table
```

```
<sql:update var="newrow" dataSource="${conn}"
```

```
INSERT INTO Products(ProductName, ProductType, Price, Brand,  
Description)
```

```
VALUES('Jeans', 'Clothes', '1000', 'Lee', 'Good Quality Jeans')
```

```
</sql:update>
```


❖ Cú pháp:

```
<sql:transaction dataSource = "dataSource" isolation = isolationLevel>  
    <sql:update> or <sql:query> statements  
</sql:transaction>
```

/ The code snippet performs transaction by first accessing a data source to create a table and then inserting a row. It then performs a SQL queries on the table **/**

```
<sql:transaction dataSource="${mydatasource}">  
    <sql:update var="newTable">  
        create table emp (  
            id int primary key,  
            name varchar(80)  
        )  
    </sql:update>  
    <sql:update var="updateCount">  
        INSERT INTO emp VALUES (1,'Jenny')  
    </sql:update>  
    <sql:update var="updateCount">  
        INSERT INTO emp VALUES (2,'Christina')  
    </sql:update>  
    ...  
    <sql:query var="empQuery"> SELECT * FROM emp  
    </sql:query>  
</sql:transaction>
```

❖ Cú pháp:

```
<sql:param value = "value"/>
```

```
//The code snippet sets the values for the  
parameters//  
<sql:update>  
    INSERT INTO Employee (DemoFirstName, DemoLastName, empDate)  
    VALUES (?, ?, ?)  
    <sql:param value="${param. DemofirstName}"/>  
    <sql:param value="${param. DemolastName}" />  
    <sql:dateParam value="${empDemoDate}" type="date" />  
</sql:update>
```

❖ Thư viện thẻ XML

- ❑ Chứa các thẻ để truy xuất các phần tử XML

- ❑ Được khai báo bằng chỉ thị xử lý

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

HỎI ĐÁP





TRẢI NGHIỆM THỰC HÀNH



TRÂN TRỌNG CẢM ƠN!



238 Hoàng Quốc Việt, Bắc Từ Liêm, Hà Nội



0968.27.6996



tuyensinh@bachkhoa-apttech.edu.vn



www.bachkhoa-apttech.edu.vn