



## **LẬP TRÌNH JAVA 4**

### **BÀI 5: EXPRESSION LANGUAGE (EL), JAVA STANDARD TAG LIBRARY (JSTL)**

#### **PHẦN 1**

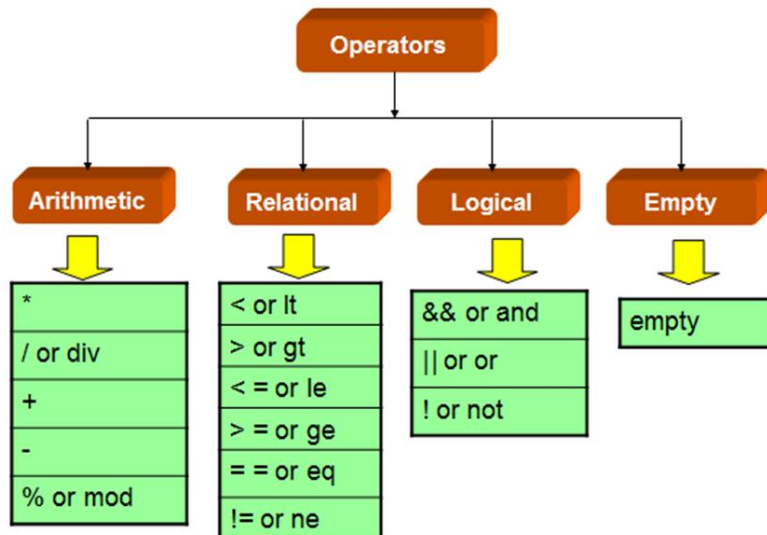
- ◎ Kết thúc bài học này bạn có khả năng
  - ❖ Expression Language (EL)
  - ❖ Java Standard Tag Library (JSTL)



- ❑ Giới thiệu:
  - ❖ EL là một tính năng mới trong JSP 2.0
  - ❖ Dễ dàng viết biểu thức trên trang JSP ngắn hơn, dễ sử dụng hơn
  - ❖ Cho phép nhà phát triển JSP có thể truy cập các đối tượng Java dễ dàng và nhanh chóng.
- ❑ Cú pháp của EL: **`${expression}`**
  - ❖ Hai biểu thức thông dụng nhất của EL là : Static text. (Text tĩnh) và Standard and Custom tags.
- ❑ Có thể truy cập đến giá trị của biến bằng dùng dấu "." (chấm) hoặc dấu [].
  - ❖ Ví dụ:  
**`${sessionScope.color} = ${sessionScope["color"]}`**

## ❑ Tính năng:

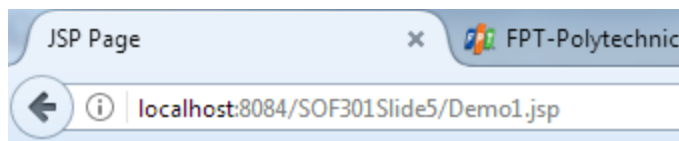
- ❖ Để hiển thị thông tin, dữ liệu và thực thi các thao tác một cách đơn giản mà ẩn đi được những dòng mã Java
- ❖ Hỗ trợ sử dụng các đối tượng ẩn và mảng
- ❖ Và đặc biệt có thể hỗ trợ các phép toán.
- ❖ EL sẽ được tính toán trong thời gian chạy (Runtime).



Implicit Object	Purpose
pageContext	Specifies the PageContext object.
servletContext	The context for the JSP page's servlet and any web components contained in the same application.
session	The session object for the client.
request	The request triggering the execution of the JSP page.
response	The response returned by the JSP page.
pageScope	Provides access to all page-scoped objects.
requestScope	Provides access to all request-scoped variables.
sessionScope	Provides access to all session-scoped variables.
applicationScope	Provides access to all application-scoped variables.
param	Provides access to all single-value components that were posted from a form.
paramValues	Provides access to all array value components that were posted from a form.
header	Provides access to all single-value HTTP headers.
headerValues	Provides access to all array components that were sent as HTTP headers.
cookie	Provides access to the cookies.
initParam	Used to access XPath initialization parameters.

## ❑ Ví dụ 1:

```
<body>
  <h1>EL Expression</h1>
  <table border="1">
    <tr>
      <td>Concept</td>
      <td>Expression</td>
      <td>Result</td>
    </tr>
    <tr>
      <td>Cộng</td>
      <td>${"${"}5+5}</td>
      <td>${5+5}</td>
    </tr>
    <tr>
      <td>Trừ</td>
      <td>${"${"}5-5}</td>
      <td>${5-5}</td>
    </tr>
    <tr>
      <td>Nhân</td>
      <td>${"${"}5*5}</td>
      <td>${5*5}</td>
    </tr>
    <tr>
      <td>Chia</td>
      <td>${"${"}5/5}</td>
      <td>${5/5}</td>
    </tr>
  </table>
</body>
```

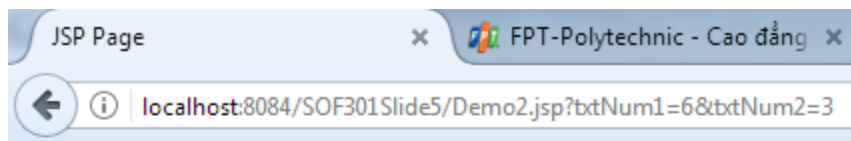


## EL Expression

Concept	Expression	Result
Cộng	$\${5+5}$	10
Trừ	$\${5-5}$	0
Nhân	$\${5*5}$	25
Chia	$\${5/5}$	1.0

## □ Ví dụ 2:

```
<body>
  <h1>Calculator</h1>
  <form>
    So1: <input name="txtNum1" value="${param.txtNum1}" /><br/>
    So2: <input name="txtNum2" value="${param.txtNum2}" /><br/>
    <input type="submit" value="Cong" />
    <h1>Tổng 2 số: ${param.txtNum1 + param.txtNum2}</h1>
  </form>
</body>
```



## Calculator

So1:

So2:

**Tổng 2 số: 9**



# DEMO

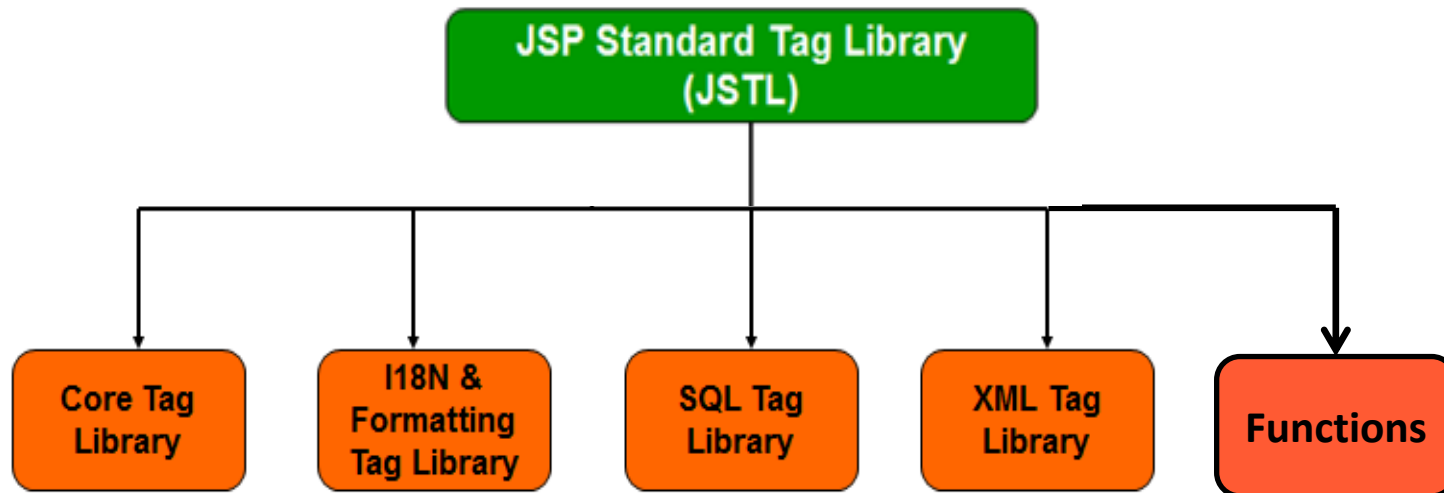
Chạy và giải thích



- ❑ JSTL là bộ thư viện thẻ chuẩn được bổ sung với mục đích tối ưu lập trình trong JSP
- ❑ Các thư viện cần thiết cho JSTL gồm
  - ❖ jstl-api.jar
  - ❖ jstl-impl.jar
- ❑ Trong JSTL có rất nhiều bộ thẻ để xử lý các vấn đề khác nhau
  - ❖ Core: chứa các thẻ lệnh điều khiển cơ bản
  - ❖ Format: chứa các thẻ định dạng và đa ngôn ngữ
  - ❖ Xml: chứa các thẻ xử lý xml
  - ❖ Sql: chứa các thẻ làm việc với CSDL
  - ❖ Function: cung cấp các hàm hỗ trợ cho EL



# JAVA STANDARD TAG LIBRARY (JSTL)



- **Core Tags:** Nhóm thẻ cơ bản
- **Formatting tags:** Nhóm thẻ định dạng
- **SQL tags:** Nhóm thẻ SQL
- **XML tags:** Nhóm thẻ XML
- **JSTL Functions:** Nhóm hàm JSTL

## ❑ Tổng quan về các chức năng của JSTL

Chức năng	Mô tả / Khai báo
Các thẻ cơ bản (Core Tags)	Các thẻ cơ bản cung cấp hỗ trợ cho bộ lặp (iteration), các điều kiện logic, if, forEach...
	<code>&lt;%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %&gt;</code>
Các thẻ định dạng (Formatting and Localization Tags)	Những thẻ này cung cấp định dạng và quốc tế hóa
	<code>&lt;%@ taglib uri="http://java.sun.com/jsp/jstl/fmt" prefix="fmt" %&gt;</code>
Các thẻ SQL (SQL Tags)	Các thẻ JSTL SQL cung cấp các hỗ trợ cho việc tương tác với cơ sở dữ liệu quan hệ như Oracle, MySql v.v.
	<code>&lt;%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql" %&gt;</code>
Các thẻ XML (XML Tags)	Thẻ XML được sử dụng để làm việc với các tài liệu XML như phân tích và đọc dữ liệu file XML
	<code>&lt;%@ taglib uri="http://java.sun.com/jsp/jstl/xml" prefix="x" %&gt;</code>
Các thẻ hàm JSTL (JSTL Functions Tags)	Cung cấp một số chức năng mà chúng ta có thể sử dụng để thực hiện với các toán tử thông dụng, hỗ trợ các hàm với kiểu String như nối chuỗi, phân tách chuỗi, ...
	<code>&lt;%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %&gt;</code>

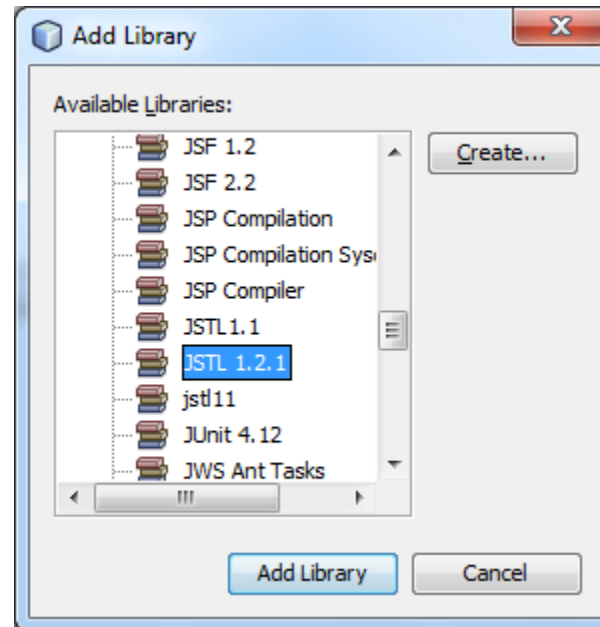
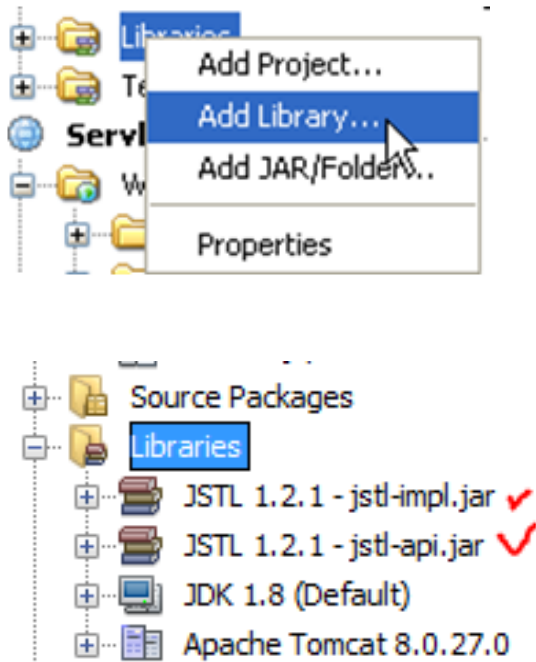
Trong phạm vi môn học này, các bạn sẽ sử dụng bộ thẻ Core Tag và SQL Tag

## ❑ Các thẻ JSTL cơ bản (JSTL Core Tags)

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

Thẻ	Ý nghĩa
<c:catch>	Catch những exception xảy ra trong code body
<c:choose>	Chọn 1 trong nhiều đoạn mã để thi hành khi thỏa mãn điều kiện nào đó
<c:forEach>	Vòng lặp quét hết tập hợp nào đó
<c:forToken>	Vòng lặp quét hết 1 chuỗi nào đó theo chuỗi phân cách
<c:if>	Thi hành đoạn code khi thỏa mãn điều kiện
<c:import>	Import một URL nào đó
<c:otherwise>	Xác định giá trị mặc định cho <c:choose>
<c:out>	Gửi output cho JspWriter hiện hành
<c:param>	Xác định 1 tham số URL cho <c:import> hay <c:url>
<c:redirect>	Điều hướng request sang URL mới
<c:remove>	Xóa 1 đối tượng trong 1 scope nào đó
<c:set>	Tạo 1 đối tượng trong 1 scope nào đó
<c:url>	Tạo 1 URL
<c:when>	Khai báo điều kiện trong <c:choose>

## ❑ Add thư viện JSTL vào project

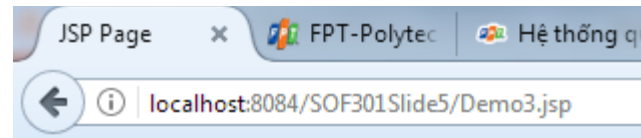


## ❑ Ví dụ 1: Core Tag

```

6  <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
7  <%@page contentType="text/html" pageEncoding="UTF-8"%>
8  <!DOCTYPE html>
9  <html>
10 <head>
11     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
12     <title>JSP Page</title>
13 </head>
14 <body>
15     <h1>Core Tag</h1>
16     <c:set var="num1" value="9" scope="page"/>
17     Giá trị num1: <c:out value="${num1}"/><br>
18     Remove giá trị num1: <c:remove var="num1" scope="page"/><br>
19     Giá trị num1: <c:out value="${num1}"/>
20 </body>
21 </html>

```



## Core Tag

Giá trị num1: 9

Remove giá trị num1:

Giá trị num1:




# DEMO

Chạy các ví dụ và giải thích



## ❑ Ví dụ 2: Core Tag



```

5  -->
6  <@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
7  <@page contentType="text/html" pageEncoding="UTF-8"%>
8  <!DOCTYPE html>
9  <html>
10  <...4 lines />
14  <body>
15      <h1>Core Tag</h1>
16      <c:set var="num1" value="20" scope="page"/>
17      <c:set var="num2" value="30" scope="page"/>
18      <c:if test="${num1 < num2}">
19          num1 < num2
20      </c:if>
21  </body>
  </html>

```



## Core Tag

num1 < num2



# DEMO

Chạy các ví dụ và giải thích





## ❑ Ví dụ 3: Core Tag

```

6  <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
7  <%@page contentType="text/html" pageEncoding="UTF-8"%>
8  <html>
9      <...4 lines />
13 <body>
14     <h1>Core Tag</h1>
15     <form>
16         So 1: <input type="text" name="num1" value="${param.num1}" /><br/>
17         So 2: <input type="text" name="num2" value="${param.num2}" /><br/>
18         <c:catch var="er">
19             <c:if test="${not empty param.num1 and not empty param.num2}">
20                 <c:set var="kq" value="${param.num1 / param.num2}" />
21                 Kết quả: <c:out value="${kq}" /><br/>
22             </c:if>
23         </c:catch>
24         <input type="submit" value="Chia" />
25         <c:if test="${not empty er}">
26             Error:<c:out value="${er}" />
27         </c:if>
28     </form>
29 </body>

```

### Core Tag

So 1:

So 2:

Kết quả: 3.0

### Core Tag

So 1:

So 2:

Error:java.lang.NumberFormatException: For input string: "a"



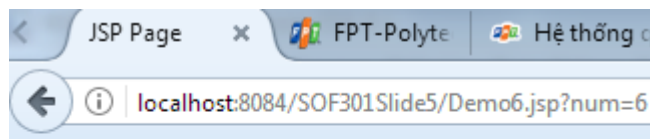
# DEMO

Chạy các ví dụ và giải thích



## ❑ Ví dụ 4: Core Tag

```
<h1>Choose ... when</h1>
<form>
Number: <input type="text" name="num" value="${param.num}"/><br/>
<c:choose>
    <c:when test="${empty param.num}">
        ...
    </c:when>
    <c:when test="${param.num %2 == 0}">
        ${param.num} là số chẵn
    </c:when>
    <c:when test="${param.num %2 != 0}">
        ${param.num} là số lẻ
    </c:when>
</c:choose>
<input type="submit" value="Submit" />
</form>
```



### Choose ... when

Number:   
 6 là số chẵn



### Choose ... when

Number:   
 7 là số lẻ

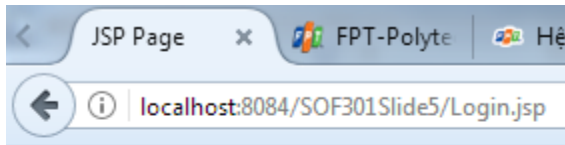


# DEMO

Chạy các ví dụ và giải thích



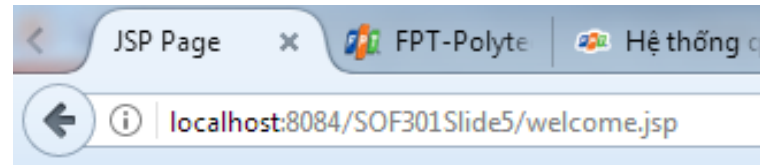
## ❑ Ví dụ 5: Core Tag



### Login

username:

password:



### Welcome page

Welcome Nguyen Van Teo



### Page

Invalid username or password!

## ❑ Ví dụ 5: Core Tag

```
<h1>Login</h1>
<form action="process.jsp" method="POST">
    username: <input type="text" name="username" value="" /><br>
    password: <input type="password" name="password" value="" /><br>
    <input type="submit" value="Login" name="action" />
</form>
```

```
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
```

```
<html>
```

```
<...4 lines />
```

```
<body>
```

```
<h1>Page</h1>
```

```
<jsp:useBean id="login" class="poly.bean.LoginBean" scope="session" />
```

```
<jsp:setProperty name="login" property="*" />
```

```
<c:if test="${login.isCheckLogin()}">
```

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

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

```
<c:if test="${not login.isCheckLogin()}">
```

```
Invalid username or password!
```

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

```
</body>
```

```
</html>
```

```
<body>
```

```
<h1>Welcome page</h1>
```

```
<jsp:useBean id="login" class="poly.bean.LoginBean" scope="session"/>
```

```
<font color="red">Welcome <jsp:getProperty name="login" property="lastname" />
```

```
</font><br/>
```

```
</body>
```

## ❑ Ví dụ 5: Core Tag

```
public class LoginBean {
    private String username;
    private String password;
    private String lastname;
    public boolean isCheckLogin()
    {
        try{
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            String url = "jdbc:sqlserver://localhost:1433; databaseName=AssignSOF301";
            Connection con = DriverManager.getConnection(url, "sa", "");
            String sql = "select * from Customers where Username=? and Password=?";
            PreparedStatement stm = con.prepareStatement(sql);
            stm.setString(1, this.username);
            stm.setString(2, this.password);
            ResultSet rs = stm.executeQuery();
            if(rs.next()){
                lastname = rs.getString("Hoten");
                System.out.println(lastname);
                rs.close();
                stm.close();
                con.close();
                return true;
            }
        }catch(Exception e){
            e.printStackTrace();
        }
        return false;
    }
    //...
```



# DEMO

Chạy các ví dụ và giải thích





## ❑ Ví dụ 6: Core Tag

```

1  <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
2
3  <%@page contentType="text/html" pageEncoding="UTF-8"%>
4  <!DOCTYPE html>
5  <html>
6  <...4 lines />
7  <body>
8      <h1>forEach - forTokens</h1>
9      <c:set var="lang" value="Core Java:Servlet;JSP:Spring;Struts:EJB" scope="page"/>
10     <%! String[] names={"Fpoly","HCM","Ha Noi","Da Nang","Tay Nguyen"}; %>
11     <b>forEach:</b><br/>
12     <c:forEach var="item" items="<%=names%>">
13         <c:out value="\${item}"/><br/>
14     </c:forEach>
15     <br><b>forTokens:</b><br/>
16     <c:forTokens items="\${lang}" delims=";" var="java">
17         <c:out value="\${java}"/><br/>
18     </c:forTokens>
19     <br><b>Counter:</b>
20     <c:forEach begin="1" end="10" step="1" var="counter">
21         <c:out value="\${counter}"/>
22     </c:forEach>
23 </body>
24 </html>
  
```

forEach - forTokens

Hệ thống quản trị

localhost:8084/SOF301Slide5/Demo7.jsp

## forEach - forTokens

### forEach:

Fpoly  
HCM  
Ha Noi  
Da Nang  
Tay Nguyen

### forTokens:

Core Java  
Servlet  
JSP  
Spring  
Struts  
EJB

Counter: 1 2 3 4 5 6 7 8 9 10



# DEMO

Chạy các ví dụ và giải thích





## LẬP TRÌNH JAVA 4

**BÀI 5: EXPRESSION LANGUAGE (EL),  
JAVA STANDARD TAG LIBRARY (JSTL)**

**PHẦN 2**

## ❑ Các thẻ SQL (SQL Tags)

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

## ❑ Tag <sql:setDataSource>: Được dùng để thiết lập thông tin data source cho database

❖ Cú pháp:

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

## ❑ Tag <sql:query>: Dùng để select data từ các tables, kết quả trả về chứa các dòng các cột dữ liệu

❖ Cú pháp:

```
<sql:query var="varName" dataSource="datasource" scope="page |  
request | session | application">  
    SQL Statement  
    <sql:param value="value" />  
</sql:query>
```

## ❑ Các thẻ SQL (SQL Tags)

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

## ❑ Tag <sql:update>: Dùng để Insert, Update, Delete

❖ Cú pháp:

```
<sql:update var="varName" dataSource="datasource" scope="page |  
    request | session | application">  
    SQL Statement  
    <sql:param value="value" />  
</sql:update>
```

## ❑ Ví dụ 1: SQL tags

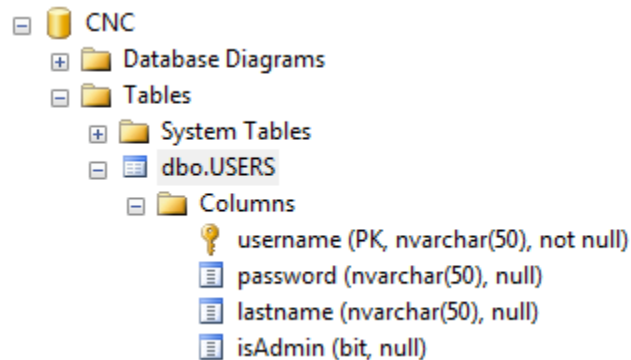
```

Start Page  welcome.jsp  Demo8.jsp
Source  History  [Icons]
11  <%@taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql" %>
12  <%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
13  <html>
14  <...4 lines />
17  <body>
18  <h1>Load data</h1>
19  <sql:setDataSource driver="com.microsoft.sqlserver.jdbc.SQLServerDriver"
20  url="jdbc:sqlserver://localhost:1433;databaseName=CNC"
21  user="sa" password=""
22  var="con"/>
23  <sql:query dataSource="${con}" var="rs">
24  select * from USERS
25  </sql:query>
26  <table border="1">
27  <tr>
28  <td>No.</td><td>Username</td><td>Password</td><td>Lastname</td>
29  </tr>
30  <c:set var="count" value="0"/>
31  <c:forEach var="dong" items="${rs.rows}">
32  <tr>
33  <c:set var="count" value="${count+1}"/>
34  <td>${count}</td>
35  <td>${dong.username}</td>
36  <td>${dong.password}</td>
37  <td>${dong.lastname}</td>
38  </tr>
39  </c:forEach>
40  </table>
41  </body>
42  </html>

```

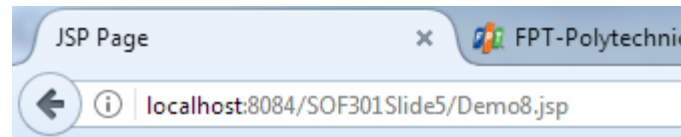
## ❑ Ví dụ 1: SQL tags

### ❖ Database



SCD050718.CNC - dbo.USERS SQLQuery1.sql - SC...50718.AP (sa (51))				
	username	password	lastname	isAdmin
	phung	123	le van	True
	trung	123	phan van	True
	kien	321	hieu	False
	an	abc	le vy	False
▶*	NULL	NULL	NULL	NULL

### ❖ Kết quả:



## Load data

No.	Username	Password	Lastname
1	an	abc	le vy
2	kien	321	hieu
3	phung	123	le van
4	trung	123	phan van



# DEMO

Chạy và giải thích

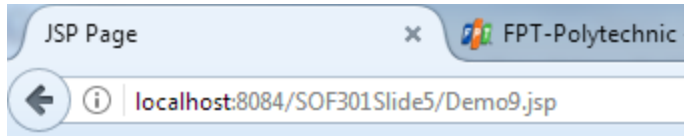




## ❑ Ví dụ 2: SQL tags

```
<body>
  <h1>Load data</h1>
  <form>
    Lastname: <input type="text" name="txtLastname" value="" /><br>
    <input type="submit" value="Search" name="action" />
  </form>
  <sql:setDataSource driver="com.microsoft.sqlserver.jdbc.SQLServerDriver"
    url="jdbc:sqlserver://localhost:1433;databaseName=CNC"
    user="sa" password=""
    var="con"/>
  <sql:query dataSource="${con}" var="rs">
    select * from USERS where lastname like ?
    <sql:param value="%${param.txtLastname}%" />
  </sql:query>
  <table border="1">
    <tr>
      <td>No.</td><td>Username</td><td>Password</td><td>Lastname</td>
    </tr>
    <c:set var="count" value="0"/>
    <c:forEach var="dong" items="${rs.rows}">
      <tr>
        <c:set var="count" value="${count+1}"/>
        <td>${count}</td>
        <td>${dong.username}</td>
        <td>${dong.password}</td>
        <td>${dong.lastname}</td>
      </tr>
    </c:forEach>
  </table>
</body>
```

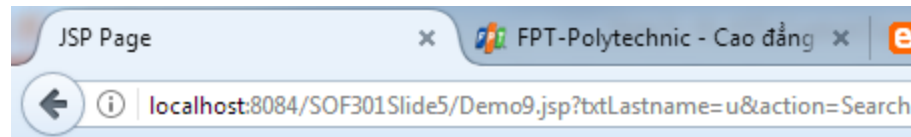
## ❑ Ví dụ 2: SQL tags



### Load data

Lastname:

No.	Username	Password	Lastname
1	an	abc	le vy
2	kien	321	hieu
3	phung	123	le van
4	trung	123	phan van



### Load data

Lastname:

No.	Username	Password	Lastname
1	kien	321	hieu



# DEMO

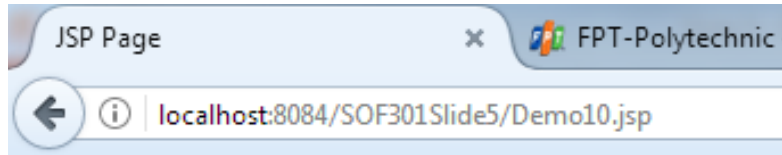
Chạy và giải thích



## ❑ Ví dụ 3: SQL tags

```
<body>
  <h1>Load data</h1>
  <form>
    Username: <input type="text" name="txtUsername" value="" /><br>
    Password: <input type="password" name="txtPassword" value="" /><br>
    Lastname: <input type="text" name="txtLastname" value="" /><br>
    <input type="submit" value="Insert" name="action" />
  </form>
  <sql:setDataSource driver="com.microsoft.sqlserver.jdbc.SQLServerDriver"
    url="jdbc:sqlserver://localhost:1433;databaseName=CNC"
    user="sa" password=""
    var="con"/>
  <c:if test="${not empty param.txtUsername}">
    <sql:update dataSource="${con}">
      Insert into USERS (username,password,lastname) values (?, ?, ?)
      <sql:param value="${param.txtUsername}" />
      <sql:param value="${param.txtPassword}" />
      <sql:param value="${param.txtLastname}" />
    </sql:update>
  </c:if>
  <sql:query dataSource="${con}" var="rs">
    select * from USERS
  </sql:query>
  <...15 lines />
</body>
```

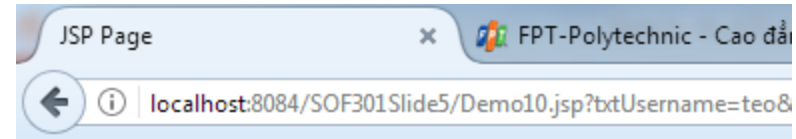
## ❑ Ví dụ 3: SQL tags



### Load data

Username:   
 Password:   
 Lastname:

No.	Username	Password	Lastname
1	an	abc	le vy
2	kien	321	hieu
3	phung	123	le van
4	trung	123	phan van



### Load data

Username:   
 Password:   
 Lastname:

No.	Username	Password	Lastname
1	an	abc	le vy
2	kien	321	hieu
3	phung	123	le van
4	teo	123	nguyen van
5	trung	123	phan van



# DEMO

Chạy và giải thích



- ❖ Expression Language (EL)
- ❖ Java Standard Tag Library (JSTL)





**Cảm ơn**