

JSP EL & JSTL PART-1

JSP EL

JSP EL stands for Java Server Pages Expression Language.

JSP EL introduced with JSTL in JSP 2.0 version.

JSTL stands for JSP Standard Tag Library.

Both JSP EL & JSTL were introduced in JSP 2.0 to simplify the JSP.

By using JSP EL & JSTL, JSP becomes 100% tag based application.

By eliminating scripting elements from JSP, JSP becomes 100% tag based application.

Scriptlets & Declarations can be replaced with JSTL.

Expressions can be replaced with JSP EL.

The pattern that identifies JSP EL is `${}`

By default expression language is enabled in JSP.

To disable JSP EL, we use the following code in a JSP:

```
<%@ page isELIgnored="true" %>
```

Operator Meaning

+ Addition

- Subtraction

* Multiplication

/ (or) div Division

% (or) mod Modulo Division

< (or) lt Less than

> (or) gt Greater than

<= (or) le Less than or equals to

>= (or) ge Greater than or equals to

== (or) eq Equals to

!= (or) ne Not equals to

&& (or) and Logical AND

|| (or) or Logical OR

! (or) not Logical NOT

JSP EL implicit object references

1) pageScope

2) requestScope

3) sessionScope

4) applicationScope

5) param

6) initParam

7) cookie

1) pageScope:

It is used to retrieve page scope attributes.

2) requestScope:

It is used to retrieve request scope attributes.

3) sessionScope:

It is used to retrieve session scope attributes.

4) applicationScope:

It is used to retrieve application scope attributes.

Example

<html>

```
<body bgcolor=green text=yellow>

<h1>

<%

pageContext.setAttribute("book1", "Java2 Complete Reference",
PageContext.PAGE_SCOPE);

pageContext.setAttribute("book2", "Head First Java",
PageContext.REQUEST_SCOPE);

pageContext.setAttribute("book3", "Thinking In Java",
PageContext.SESSION_SCOPE);

pageContext.setAttribute("book4", "SCJP By Kathy Sierra",
PageContext.APPLICATION_SCOPE);

%>

${ pageScope.book1 }<br>
${ requestScope.book2 }<br>
${ sessionScope.book3 }<br>
${ applicationScope.book4 }

</h1>

</body>

</html>
```

5) param:

It is used to retrieve request parameters.

Example:

HTML Page

```
<input type=text name=username>

<input type=password name=pword>
```

JSP Page

`${ param.username }`

`${ param.pword }`

6) initParam:

It is used to retrieve initialization parameters.

Example:**web.xml**

`<init-param>`

`<param-name>message</param-name>`

`<param-value>Welcome</param-value>`

`</init-param>`

JSP Page

`${ initParam.message }`

7) cookie:

It is used to retrieve cookie values.

Example:**set.jsp**

`<% Cookie c=new Cookie("book", "Java2 Complete Reference");`

`response.addCookie(c); %>`

get.jsp

`${ cookie["book"].value }`

JSTL

JSTL stands for JSP Standard Tag Library.

JSTL introduced in JSP 2.0 version to simplify the JSP.

JSTL tags are divided into 5 categories:

- 1) Core Tags
- 2) SQL Tags
- 3) Formatting Tags
- 4) Function Tags
- 5) XML Tags

In order to use JSTL tags we must download the following JAR files from Internet and copy into tomcat lib folder:

- 1) jstl.jar
- 2) standard.jar

1) Core Tags:

- 1) <c:out>
- 2) <c:set>
- 3) <c:remove>
- 4) <c:if>
- 5) <c:choose>
- 6) <c:when>
- 7) <c:otherwise>
- 8) <c:forEach>
- 9) <c:forTokens>
- 10) <c:redirect>

In order to use the above tags we must include the following taglib directive in a JSP.

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

1) <c:out>:

It is used to display output messages.

2) <c:set>:

It is used to set the value in a variable.

3) <c:remove>:

It is used to remove the value from variable.

Example:

```
<html>
<body bgcolor=green text=yellow>
<h1>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<c:set var="a" value="10" />
<c:out value="{ a }" />
<c:remove var="a" />
<c:out value="{ a }" />
</h1>
</body>
</html>
```

4) <c:if>:

It is used to express the condition.

Example:

```
<html>
<body bgcolor=green text=yellow>
<h1>
```

```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<c:set var="a" value="10" />
<c:if test="${ a>0 }">
<c:out value="Positive Number" />
</c:if>
</h1>
</body>
</html>
```

5)<c:choose> 6)<c:when> & 7)<c:otherwise> tags are equivalent to if else if else statement or switch statement.

Example:

```
<html>
<body bgcolor=green text=yellow>
<h1>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<c:set var="a" value="-10" />
<c:choose>
<c:when test="${ a>0 }">
<c:out value="Positive Number" />
</c:when>
<c:when test="${ a<0 }">
<c:out value="Negative Number" />
</c:when>
<c:otherwise>
<c:out value="Zero" />
```

```
</c:otherwise>
```

```
</c:choose>
```

```
</h1>
```

```
</body>
```

```
</html>
```

8) <c:forEach>:

It is equivalent to for loop

Example:

```
<html>
```

```
<body bgcolor=green text=yellow>
```

```
<h1>
```

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

```
<c:forEach var="i" begin="1" end="10" step="1">
```

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

```
</c:forEach>
```

```
</h1>
```

```
</body>
```

```
</html>
```

9) <c:forTokens>:

It is equivalent to StringTokenizer class.

Example:

```
<html>
```

```
<body bgcolor=green text=yellow>
```

```
<h1>
```



```
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<c:forTokens var="s" items="Welcome:To:JSTL" delims=":">
<c:out value="{ s }" /><br>
</c:forTokens>
</h1>
</body>
</html>
```

10) <c:redirect>:

It is used to pass the control from one jsp to another jsp

Example:

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

SQL Tags:

- 1) <sql:setDataSource>
- 2) <sql:update>
- 3) <sql:query>

In order to use the above tags we must include the following taglib directive in a JSP.

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

1) <sql:setDataSource>:

It is used to establish the connection between JSP and database.

2) <sql:update>:

It is used to execute non select queries.

3) <sql:query>:

It is used to execute select queries.

Example:

```
<html>

<body bgcolor=red text=yellow>

<table border=20 width=500 height=300>

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

<sql:setDataSource var="con" driver="oracle.jdbc.driver.OracleDriver"
url="jdbc:oracle:thin:@localhost:1521:xe" user="system" password="manager" />

<sql:query var="rs" dataSource="${ con }" sql="select * from uinfo" />

<c:forEach var="record" items="${ rs.rows }">
<tr><td><c:out value="${ record.fname }" /></td>
<td><c:out value="${ record.lname }" /></td>
<td><c:out value="${ record.uname }" /></td>
<td><c:out value="${ record.pword }" /></td></tr>
</c:forEach>

</table>

</body>

</html>
```