

Le langage EL

Introduction

- EL à partir de JSP 2.0
- Caractéristiques
 - Syntaxe simplifiée
 - Permet l'affichage de variables , attributs d'objets
 - Opérateurs, objets implicites, appel de fonctions Java
 - Peut être utilisé dans les pages jsp scriptless
- EL unifié à partir de JSP 2.1
 - Évaluation différée des expressions
 - Support des expression qui permettent de modifier une valeur

Pages scriptless

- Une page jsp scriptless ne peut pas contenir des éléments de scripts jsp (déclarations, scriptlets, expression)
- Une page JSP peut être déclarée scriptless dans le DD web.xml (\geq JSP 2.0)
 - Dans web.xml

```
<jsp-config>
<jsp-property-group>
<url-pattern>*.jsp</url-pattern>
<scripting-invalid>>true</scripting-invalid>
</jsp-property-group>
</jsp-config>
```

Activation du langage EL

- A partir de la directive page

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

- Dans le DD de l'application web (web.xml)

```
<jsp-config>
```

```
<jsp-property-group>
```

```
<url-pattern>*.jsp</url-pattern>
```

```
<el-ignored>>true</el-ignored>
```

```
</jsp-property-group>
```

```
</jsp-config>
```

Syntaxe des instructions EL

- Syntaxe :
 - `${expr}` : évaluation immédiate
 - `#{expr}`: évaluation différée (JSP 2.1)
 - L'évaluation de l'expression n'est pas prise en charge par le moteur JSP, elle est réalisée au moment approprié par l'implémentation du langage EL unifié
 - Exemple:

```
<h:inputText id="nom" value="#{livre.titre}"/>
```
 - Une expression valide peut contenir des littérales, opérateurs, références d'objets, appel de fonctions
 - Des instructions ne peuvent pas être imbriquées.

Littérales

- true ou false .
- Une chaîne de caractère délimitée par “ ou ‘ (\ peut être utilisé comme séquence d'échappement).
- Integer
- float
- null

Exemple

– $\{2*4\}$

opérateurs

- +, -, *, /, div, %, mod
- == ou eq
- != ou ne
- < ou lt
- > ou gt
- <= ou le
- >= ou ge
- && ou and
- || ou or
- ! Ou not
- (), empty, [], . (point)

opérateurs

Exemples

- *`${empty var}`: retourne true si var référence un objet null ou si var est un objet vide (de type `String`, `List`, `Map`, `array`).*
- Soit un bean user
 - `${user.prenom}`
 - `${user[nom]}`

Objets implicites

- `pageContext` : cet objet permet d'accéder aux objets implicites d'une page
 - cet objet est de type `javax.servlet.jsp.PageContext`
 - Exemple : ``${pageContext.request}`
- `pageScope` : objet de type `Map` qui permet l'accès aux objet de portée page
 - ``${pageScope.obj}`
 - ``${pageScope.obj.attr}`
- `requestScope`
- `sessionScope`
- `applicationScope`

Objets implicites

- param: de type Map
 - Cet objet est obtenu par appel de la méthode `ServletRequest.getParameter(String nom)`.
 - `#{param.name}` : équivalente à `request.getParameter(nom)`.
- paramValues: de type Map, contient les valeurs d'un paramètre sous forme de `String[]`
 - Obtenu par appel de `ServletRequest.getParameterValues(String nom)`.
 - Exemple : `#{paramValues.name}` est équivalent à `request.getParameterValues(name)`.

Objets implicites

- `header`: type A Map
 - Obtenu par appel de la méthode `ServletRequest.getHeader(String name)`.
Requests always
- `headerValues`: Map contient les valeurs d'un élément header, équivalente à l'appel `ServletRequest.getHeaders(String)`
 - `headerValues.name` est équivalente à `request.getHeaderValues(name)`

Objets implicites

- `cookie`: de type `Map`
 - Exemple: `${cookie.c1.value}` retourne la première valeur du cookie qui porte le nom `c1`
 - Pour récupérer toutes les valeurs d'un cookie il faut utiliser `${headerValues.nom}`.
- `initParam`: de type `Map` retourne les valeurs d'un paramètre d'initialisation du contexte (application web)
 - Exemple: `${initParam.nom}`

Objets implicites

- Exemples

- Exemple 1: un bean de portée page

- Soit un bean questions `<jsp:useBean id="questions" scope="page" class="jsp.exemple3.Questions">`
`<jsp:setProperty name="questions" property="theme"/>`
`</jsp:useBean>`
 - Accès à partir d'une expression EL
``${pageScope.questions.theme}`.

- Exemple 2: une variable de portée session

- `<% session.put("adresse", "77 Lot iguider)%>`
 - Instruction EL: ``${sessionScope.adresse}`
 - Script expression: `<%= session.get("adresse")%>`

EL et attributs des actions standards

- Une expression EL peut être utilisée comme valeur de certains attributs dans les actions standards.
- Attributs dans les quelle une expression EL est valide:

Action	Attribute
useBean	beanName
setProperty	Value
include	page
forward	page
param	value

- Exemple:

```
<jsp:forward page="{param.pageSuivante}" />
```