

Expression Language

Résumé

Variables et beans en attributs

- ▶ `${nom_Variable}`
 - ▶ Recherches successives dans les attributs de page, request, session, application.
 - ▶ Affiche la variable (types simples) si elle existe sinon, rien.
- ▶ `${bean.prop}` ou `${bean["prop"]}`
 - ▶ De même, recherche le bean en attribut et affiche la valeur de la propriété si un getter est bien défini pour elle et si le type le permet.
 - ▶ Ex: un bean `Personne` nommé `pers` placé en attribut de session
 - ▶ `${pers.nom}` affichera le nom.
 - ▶ `${pers.age}` pas de soucis de conversion.

Cas des tableaux, des listes ou des maps

- ▶ La notation par crochets permet d'accéder aux éléments de ces types.
 - ▶ Cas d'une variable simple
 - ▶ EX: Si l'attribut à récupérer est une `ArrayList<String>` nommée `listeStr`
 - ▶ `${listeStr[0]}` pour récupérer le premier élément par exemple.
 - ▶ Cas d'un bean dont une propriété est une liste
 - ▶ Ex: Bean `Personne` nommé `pers` dont la propriété `adresses` est une `ArrayListe` par exemple
 - ▶ `${pers.adresses[1]}` affichera l'adresse stockée à l'index(1) de la propriété `adresses` du bean `pers`.
- ▶ Remarque :
 - ▶ La syntaxe des crochets permet aussi d'avoir un index sous forme de variable : `${liste[i]}` si `i` est une variable définie dans la page.

Autres possibilités

► Evaluation :

- `${3 * (2+1) -4}` → 5
- `${5<4}` → false
- `${''2''+5}` → 7
- `${true && false}` → false : idem avec `||`

► Test conditionnel :

- `${test ? Express1 : express2}`
- `${true && true ? "hello" : "bye"}` → hello

JSTL

Résumé

Librairie de fonctionnalités

► Core :

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

-foreach :

Exemple 1 :

```
<core:foreach var=« nom » items=« collection »>
    ${nom}
</core:foreach>
```

Exemple 2 :

```
<core:foreach var=« i » begin=« 1 » end=« 10 » step=« 1 »>
    ${i}
</core:foreach>
```


Exemple 3 :

```
<core:foreach var=« i » begin=« 1 » end=« 10 » varStatus=« status »>
    ${status.first}
    ${status.last}
    ${status.index}
    ${status.begin}
    ${status.end}
    ${status.step}
</core:foreach>
```

<c:if />

Si la condition est réalisée, le code entre les balises est exécuté.

```
<c:forEach var="city" items="${cities}" varStatus="status">  
    ["${city.name}", "${city.shortTime}", "${city.population}"]  
    <c:if test="${!status.last}">,</c:if>  
</c:forEach>
```



<c:choose/>

► Équivalent du case

```
<UL>
<c:forEach var="i" begin="1" end="10">
  <LI>${i}
    <c:choose>
      <c:when test="${i < 4}">
        (small)
      </c:when>
      <c:when test="${i < 8}">
        (medium)
      </c:when>
      <c:otherwise>
        (large)
      </c:otherwise>
    </c:choose>
  </c:forEach>
</UL>
```

