

CHEAT SHEET SERVLET/JSP

Servlet / JSP

Servlet

généralités

Hérite de `HttpServlet`,

Annotation `@WebServlet("/cheminweb")` sur la classe

Exemple :

```
@WebServlet("/affichelalist")
public class AfficheList extends HttpServlet {
```

Surcharge de `doGet` : Méthode qui s'exécute lors d'une requête client en GET

Surcharge de `doPost` : Méthode qui s'exécute lors d'une requête client en POST

(Formulaires)

Ces deux méthodes prennent deux paramètres :

`HttpServletRequest` qui correspond à la requête en entrée.

`HttpServletResponse` qui correspond à la réponse vers le client.

Forward vers un jsp, l'URL reste la même que celle de la servlet

```
request.getRequestDispatcher("showList.jsp").forward(request, response);
```

Redirect vers une servlet ou une jsp, l'URL change.

```
response.sendRedirect("accueil");
```

JSP

C'est ici que l'on code du HTML dans notre cas, on peut déclarer des taglibs de la forme suivante (exemple avec "core")

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

On peut coder du Java en ouvrant les balises suivantes:

`<% %>` mais on préférera toujours JSTL

Méthodes usuelles et équivalence JSTL & EL (Expression Languages (\${}))

Description	Java	Jstl
Récupère un paramètre d'url ou de formulaire	<code>request.getParameter("action");</code>	<code>\${param.action}</code>
Passe un paramètre dans le scope Request (pour transmettre la valeur d'une servlet à une jsp par exemple)	<code>request.setAttribute("variable", "valeur");</code>	<pre><c:set var="variable" scope= "request" value="valeur"/></pre>

Récupère un paramètre dans le scope Request	<code>request.getAttribute("variable");</code>	<code>\${requestScope.variable}</code> si pas de concurrence <code>\${variable}</code>
Passe un paramètre dans le scope Session	<code>request.getSession().setAttribute("variable", "valeur");</code>	<code><c:set var="variable" scope="session" value="valeur"/></code>
Récupère un paramètre dans le scope Session	<code>request.getSession().getAttribute("variable");</code>	<code>\${sessionScope.variable}</code>

On peut itérer sur une liste de la manière suivante :

Si j'ai dans servlet :

```
request.setAttribute("listeProduits", Application.getInstance().getProduitDao().findAll());
```

J'ai accès à la variable listeProduits dans ma JSP. Je peux la parcourir de la manière suivante.

```
<table>
  <c:forEach items="${listeProduits}" var="prod">
    <tr>
      <td>
        ${prod.nom}
      </td>
      <td>
        <a href="produit?action=del&id=${prod.id}">supprimer</a>
      </td>
    </tr>
  </c:forEach>
</table>
```

Ici, on accède automatiquement au getter de nom et id

Il existe de nombreuses autres méthodes de taglibs, une liste ici :

<https://www.journaldev.com/2064/jsp-expression-language-el-example-tutorial>