

JSP

Première page JSP

```
<html>
  <head>
    <title>Ma première page JSP</title>
  </head>
  <body>
    <%
      System.out.println("Hello World");
      out.println("Hello World");
    %>
  </body>
</html>
```

Première page JSP

- Placer le code précédant dans un fichier nommé helloworld.jsp à la racine de votre application web (myapp).
- Avec votre navigateur allez à l'adresse :
<http://localhost:8080/myapp/helloworld.jsp>

Les tags JSP

- Declaration tag
- Expression tag
- Scriptlet tag
- Directive Tag
- Action tag

Declaration tag (<%! %>)

- Exemple :

```
<%!  
private int counter = 0 ;  
private int getCounter() {  
    counter +=1;  
    return counter;  
}  
%>
```

- Déclaration de variables et de méthode.

Expression tag (<%= %>)

- Exemple

Date : <%= new java.util.Date() %>

- Une instruction par expression tag ne comporte pas de ; à la fin de la ligne

Scriptlet tag (<% ... %>)

```
<%  
if(this.counter > 0){  
    out.print(" Conteur = " + this.counter);  
}  
else out.print(" Le conteur est à zéro ");  
%>
```

- Code java quelconque.

Outils pour le développement

- IDE :
 - netBeans
 - eclipse
- SGBDR :
 - toad
 - SQLNavigator
 - Tora (open source)
- Plus aide mémoire :
 - SQL
 - API java
 - HTML

Directive tag (<%@ directive ... %>)

- 1) page – information pour le traitement de la page
- 2) Include – fichier à inclure
- 3) Tag library – librairie de tag à utiliser dans cette page

Page directive

JSP Syntax

<%@ page

```
[ language="java" ]  
[ extends="package.class" ]  
[ import="{package.class | package.*}, ..." ]  
[ session="true|false" ]  
[ buffer="none|8kb|sizekb" ]  
[ autoFlush="true|false" ]  
[ isThreadSafe="true|false" ]  
[ info="text" ]  
[ errorPage="relativeURL" ]  
[ contentType="mimeType [ ; charset=characterSet ]" |  
  "text/html ; charset=ISO-8859-1" ]  
[ isErrorPage="true|false" ]  
[ pageEncoding="characterSet | ISO-8859-1" ]
```

%>

cf page 15 du JavaServer Pages™ v1.2 Syntax Reference • June 2002

Page directive

- <%@ page import= "java.util.Collection,
java.sql.* " %>
- <% @ page errorPage= "
/error/404_error.html " %>

Include Tag

- <%@ include file=
"/header/headerBegin.jsp " %>

Tag library

- `<%@ taglib uri="URIForLibrary" prefix="tagPrefix" %>`
- `<%@ taglib uri="/WEB-INF/c.tld" prefix=" c" %>`
`<c:loop>`
`...`
`</c:loop>`

Action tag

- `<jsp:forward>` (`response.sendRedirect(url)`)
- `<jsp:include>`
- `<jsp:useBean>`
- `<jsp:setProperty>`
- `<jsp:getProperty>`

Object implicites à une page JSP

- `request` (`javax.servlet.HttpServletRequest`)
- `response` (`javax.servlet.ServletResponse`)
- `pageContext` (`javax.servlet.jsp.PageContext`)
- `session` (`javax.servlet.http.HttpSession`)
- `application` (`javax.servlet.ServletContext`)
- `out` (`javax.servlet.jsp.JspWriter`)
- ...

JavaBeans

- Classes normales répondant à quelques contraintes.
- Chaque variable devant être accéssée doit avoir des getter et setter.
- les getter setter doivent avoir la syntaxe suivante :
 - `nomVariable`
 - `getNomVariable()`
 - `setNomVariable(...)`

Exemple JavaBeans

```
public class Person {
    private String lastname="";
    private String firstname="";
    public Person() {}
    public java.lang.String getLastName() {
        return lastname;
    }
    public void setLastName(java.lang.String lastname) {
        this.lastname = lastname;
    }
    public java.lang.String getFirstname() {
        return firstname;
    }
    public void setFirstname(java.lang.String firstname) {
        this.firstname = firstname;
    }
}
```

Référence

- Tutorial JSP
 - <http://java.sun.com/products/jsp/docs.html>

HTML et JSP

- Transmission des paramètres :
 - GET
 - POST

GET

- les paramètres sont passés via l'url.
 - permet de conserver les paramètres dans l'url.
 - taille des valeurs limitée
- Exemple :
- <http://localhost:8080/bda/info.jsp?nom=clau de&prenom=jean-marc>

POST

- Propre au formulaire
 - les paramètres ne sont pas passés par l'url.
 - la gestion des paramètres se fait coté serveur.

GET POST

- Pour les deux GET et POST les valeurs peuvent être récupérer via le request :
- exemple :
 - `request.getParameter(« prenom »);`
- Retourne soit le champ de name = « prenom » d'un formulaire (GET ou POST)
- soit le paramètre de name = « prenom » d'un d'une Url (GET)