



Sesión 8: JSP



- **Introducción a JSP**
- **Relación entre servlets y páginas JSP**
- **Elementos de JSP**
 - **Código**
 - **Directivas**
 - **Acciones**



- **Introducción a JSP**
- **Relación entre servlets y páginas JSP**
- **Elementos de JSP**
 - **Código**
 - **Directivas**
 - **Acciones**

Páginas JSP



- Los servlets eran clases Java que generaban o mostraban contenido web
- Las páginas JSP son páginas HTML con código Java incrustado
- El *contenedor JSP* (una parte del servidor web) se encarga de sustituir el código Java por el resultado de su ejecución, y devolver la página HTML resultante al cliente

Ejemplo de JSP



```
<html>
  <head>
    <title>Mi primera página JSP</title>
  </head>
  <body>
    <h1> Hoy es:
      <%= new java.util.Date() %>
    </h1>
  </body>
</html>
```



- **Introducción a JSP**
- **Relación entre servlets y páginas JSP**
- **Elementos de JSP**
 - **Código**
 - **Directivas**
 - **Acciones**

Relación servlets - JSP



- **El servidor web traduce internamente cada página JSP en un servlet**
 - **En principio servlets y JSP ofrecen la misma funcionalidad**
- **Sin embargo:**
 - **Los JSP son mejores para presentar la información (generar las páginas de contenidos)**
 - **Los servlets son mejores para procesar la información, y en función de la misma, decidir qué página JSP mostrar**



- **Introducción a JSP**
- **Relación entre servlets y páginas JSP**
- **Elementos de JSP**
 - **Código**
 - **Directivas**
 - **Acciones**

Código: expresiones



- Van entre los marcadores `<%=` y `%>`. El contenedor JSP (servidor web) los traduce directamente por el resultado de la expresión que hayamos puesto entre esas marcas, y coloca dicha expresión directamente en la página servida

```
Hoy es <%= new java.util.Date() %>
```

Código: scriptlets



- Van entre los marcadores `<%` y `%>`. El contenedor JSP (servidor web) simplemente ejecuta el código, que puede o no producir una salida en la página. Los traduce directamente por el resultado de la expresión que hayamos puesto entre esas marcas, y coloca dicha expresión directamente en la página servida.

Código: scriptlets



```
<%  
    java.util.Calendar ahora =  
    java.util.Calendar.getInstance();  
    int hora =ahora.get(java.util.Calendar.HOUR_OF_DAY);  
%>  
<b> Hola mundo, <i>  
<% if ((hora>20)|| (hora<6)) { %>  
    buenas noches  
<% }  
    else if ((hora>=6)&&(hora<=12)) { %>  
        buenos días  
<% } else { %>  
        buenas tardes  
<% } %>
```

Código: declaraciones



- Van entre los marcadores `<%!` y `%>`. El contenedor JSP (servidor web) coloca este código en el servlet compilado como código global a todo el servlet. De esta forma, aquí podremos definir variables globales, y métodos propios que queramos tener dentro del servlet para poderlos llamar desde la página JSP en cualquier punto.

Código: declaraciones



```
<%!  
    private int accesos = 0;  
    public void imprimeAhora()  
    {  
        out.println("Hoy es " + new java.util.Date());  
    }  
%>  
...  
<b> Hola </b>  
<p> Esta página tiene <%=++accesos%> visitas</p>  
<p> Fecha y hora actuales: <% imprimeAhora(); %></p>
```

Objetos implícitos en JSP



Acceso al objeto "response":

```
<% reponse.sendRedirect("mipagina.jsp"); %>
```

Acceso al objeto "request":

```
<% String login = request.getParameter("login"); %>
```

Directivas



- **Influyen en la estructura que tendrá el servlet generado a partir de la página JSP. Hay tres tipos de directivas:**
 - **page:** importar clases de Java, fijar el tipo MIME de la respuesta, controlar el *buffer* de salida,...
 - **include:** sirve para incluir código en la página *antes de que se realice la compilación del JSP*.
 - **taglib:** se emplea cuando el JSP hace uso de etiquetas definidas por el usuario.

Directivas: ejemplos



```
<%@ directiva atributo="valor" %>
```

Importar clases Java para poderlas utilizar en el código:

```
<%@ page import="java.io.*" %>
```

Incluir en estático un fichero dentro de otro

```
<%@ include file="config.jsp" %>
```


Acciones



- permiten hacer mediante etiquetas parecidas a las de HTML una serie de operaciones, como por ejemplo, incluir el contenido de otra página en la actual

```
<jsp:include page=" otraPagina.jsp" />
```

NOTA: este `jsp:include` es DISTINTO a la directiva `include`

```
<jsp:forward page="error.html" />
```