



# Curso de Java Standard



Ing. Octavio Robleto



octavio.robleto@gmail.com



<https://octaviorobleto.com>



# Java Server Pages (JSP)

Es la solución de Java para el desarrollo Web dinámico.

Son documentos que poseen elementos (tags) HTML y además posee la capacidad de incluir fragmentos de código Java a través de tags propios de esta tecnología, estos fragmentos de códigos también son llamados Scriptlets.

Un JSP se convierte en una clase Servlet y se ejecuta en el servidor, el código de Java incluido en el documento JSP pasa a formar parte de esta nueva clase generando una respuesta HTML del lado del cliente.

Recordemos que los clientes Web (Navegadores) no ejecutan otro tipo de código que no sea HTML, CSS y JavaScript



# Objetos Implícitos en JSP

JSP nos proporciona objetos implícitos en cada documento para poder ser utilizada de la manera que necesitemos, adicionalmente podemos sin ningún problema crear nuestros propios objetos o variables de forma explicita.

Declaración	Descripción
request	Objeto de la clase HttpServletRequest
response	Objeto de la clase HttpServletResponse
session	Objeto de la clase HttpSession que esta asociada al request, si es que trabajamos con sesiones
out	Objeto de la clase PrintWriter usada para imprimir directamente en la pagina resultante

# Elementos JSP

Para poder incorporar código Java en un documento HTML, JSP y /o XML se debe encerrar entre llaves angulares y los símbolos de porcentaje “<% %>”.

**Comentarios:** Nos ayudan a saber sobre el código.

```
<!-- Comentario de una linea en JSP --%>
<%
    // Comentarios de una linea en Java

    /*
    Comentarios de varias lineas en Java
    */
%>
```

# Elementos JSP

**Expresiones:** Se usan para mandar valores directamente a la pagina web resultante del JSP.

```
<!-- Podemos utilizar explicitas o implicitas --%>  
<%=request.getParameter("nombre") %>  
  
<%=i %>
```

# Elementos JSP

**Scriptlets:** Se utilizan para utilizar cualquier bloque o sentencia de Java.

```
<%  
|   for (int i = 0; i < 10; i++){  
|       out.print(i);  
|   }  
|  
%>  
  
<!-- Podemos intercalar codigo Java y HTML -->  
  
<%  
|   for (int i = 0; i < 10; i++){  
|  
%>  
|   <h1><%=i %></h1>  
|  
|   }  
|  
%>
```

# Elementos JSP

**Declaraciones:** Se utilizan para definir variables y métodos que se utilizarán en scripts y expresiones.

```
<%!  
    private int contadorIngresos=0;  
  
    public String breakHTML(){  
        return "<br>";  
    }  
%>
```

# Elementos JSP

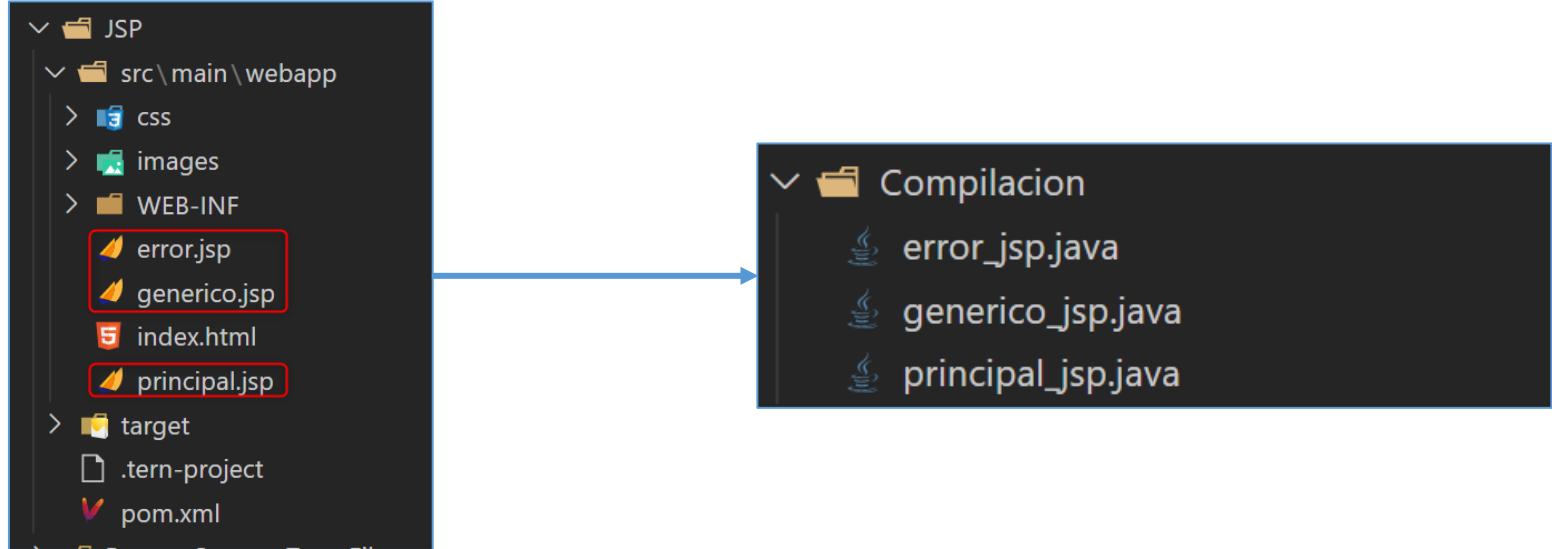
**Directivas:** Son etiquetas a partir de las cuales se genera información que puede ser utilizada por el motor JSP. No producen una salida visible al usuario sino que configuran cómo se ejecutará la página JSP.

Tipo	Método	Descripción
include	<code>&lt;%@include file="cabecera.html" %&gt;</code>	Incluye el contenido de un fichero en la página mediante el atributo file.
taglib	<code>&lt;%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%&gt;</code>	importa bibliotecas de etiquetas (Tag Libraries)
page import	<code>&lt;%@page import="java.util.Map"%&gt;</code>	Importa clases y paquetes Java para ser utilizadas dentro del fichero JSP.
page session	<code>&lt;%@page session="true"%&gt;</code>	Especifica si utiliza los datos contenidos en sesión; por defecto "true".
page errorPage	<code>&lt;%@page errorPage="error.jsp"%&gt;</code>	Especifica la ruta de la página de error que será invocada en caso de producirse una excepción durante la ejecución de este fichero JSP.



# JSP → Servlets

Como habíamos comentado al principio todos los documentos JSP al ser invocados el servidor o contenedor WEB creara un Servlet por cada documento que generara la lógica en Java que terminara mostrando el resultado en HTML



# RequestDispatcher

Esta interfaz proporciona la posibilidad de enviar o incluir objetos a otro recurso (html, servlet o jsp).

Tipo	Método	Descripción
void	<code>forward(ServletRequest request, ServletResponse response)</code>	Reenvía una solicitud de un servlet a otro recurso
void	<code>include(ServletRequest request, ServletResponse response)</code>	Incluye el contenido de un recurso

```
request.setAttribute("nombreObjeto", objeto);

RequestDispatcher dispatcher = request.getRequestDispatcher("documento.ext");
dispatcher.forward(request, response);
```