

0^a
Emisión

DIPLOMADO Desarrollo de Sistemas con Tecnología Java

Módulo 8 Desarrollo de aplicaciones Web con Spring Web MVC

Ing. Jorge Alberto Montalvo Olvera



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Dirección General de Cómputo y de Tecnologías de información y Comunicación

Dirección de Docencia en TIC



Educación
Continua
1971 - 2021

TEMARIO

- Introducción
 - Protocolo HTTP
 - XHTML y HTML5
 - Aplicación web
 - Servidor web
 - Servlets
 - Java Server Pages
 - Lenguaje de expresión
 - Sesiones
 - JSTL

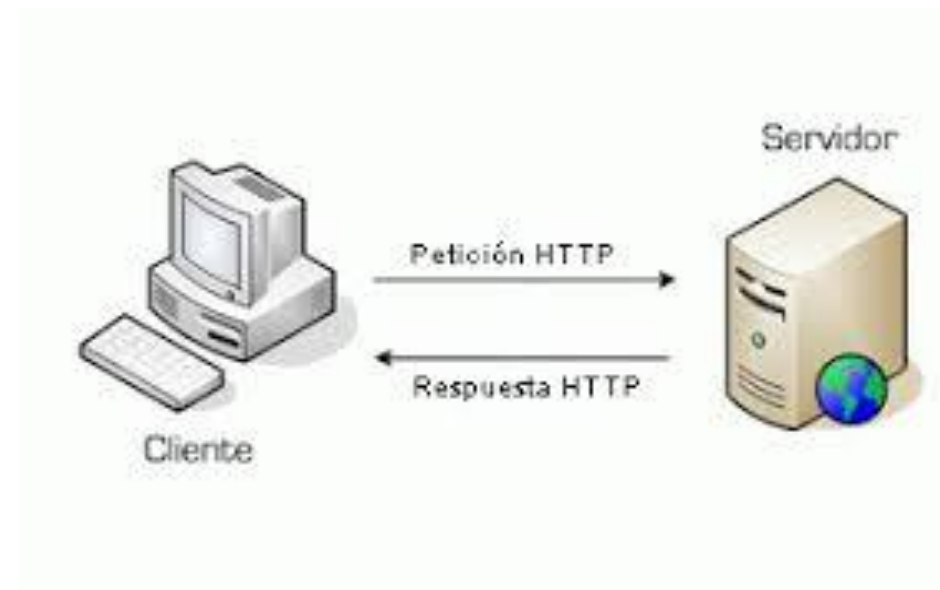
Protocolo HTTP

- El protocolo de transferencia de hipertexto (HTTP, HyperText Transfer Protocol) es el protocolo usado en cada transacción de la Web (www).
- El hipertexto es el contenido de las páginas web, y el protocolo de transferencia es el sistema mediante el cuál se envían las peticiones de acceso a una página y la respuesta con el contenido.



Protocolo HTTP

- El protocolo HTTP se basa en un paradigma de peticiones y respuestas.
- Una petición de un cliente a un servidor ha de incluir el método que se aplica al recurso, el identificador del recurso y la versión del protocolo que usa para realizar la petición.



Protocolo HTTP



Protocolo HTTP

- Existen distintos métodos de petición:
 - GET. El método GET requiere la devolución de información al cliente identificada por la URI.
 - HEAD. El método HEAD es igual que el método GET, salvo que el servidor no tiene que devolver el contenido, sólo las cabeceras.
 - POST. El método POST se usa para hacer peticiones en las que el servidor destino acepta el contenido de la petición como un nuevo subordinado del recurso pedido.
 - PUT. El método PUT permite guardar el contenido de la petición en el servidor bajo la URI de la petición.

Protocolo HTTP

- En la petición y respuesta se encuentran cabeceras que aportan información como codificación, longitud, localización de los mensajes:
 - Date
 - Content-Encoding
 - Content-Type

```
HTTP/1.1 200 OK
    Date: Sat, 09 Oct 2010 14:28:02 GMT
    Server: Apache
    Last-Modified: Tue, 01 Dec 2009 20:18:22 GMT
    ETag: "51142bc1-7449-479b075b2891b"
    Accept-Ranges: bytes
    Content-Length: 29769
    Content-Type: text/html

<!DOCTYPE html... (here comes the 29769 bytes of the requested web page)
```


XHTML

- XHTML (eXtensible Hypertext Markup Language), es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML.



XHTML

- Por tanto, un documento debe seguir todos los criterios siguientes:
 - El documento debe ser conforme con las restricciones expresadas en las DTD's.
 - El elemento raíz del documento debe ser <html>.
 - El nombre del espacio de nombres por defecto del elemento raíz debe ser el nombre del espacio de nombre XHTML, <http://www.w3.org/1999/xhtml>.
 - Debe existir una declaración DOCTYPE en el documento anterior al elemento raíz.

XHTML

```
1 <?xml version='1.0' encoding='UTF-8' ?>
2 <!DOCTYPE composition PUBLIC "-//W3C//DTD XHTML 1.0 Transiti
3 <ui:composition xmlns:ui="http://java.sun.com/jsf/facelets"
4     template="./template.xhtml">
5
6     <ui:define name="top">
7         top
8     </ui:define>
9
10    <ui:define name="body">
11        body
12    </ui:define>
13
14 </ui:composition>
```

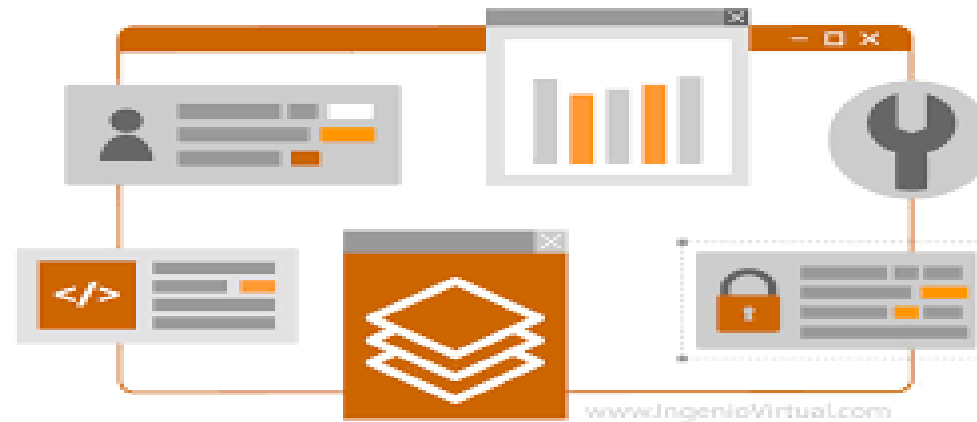
HTML5

- HTML5 es la nueva versión de HTML, tiene más etiquetas y por lo tanto más opciones de diseño.



Aplicación Web

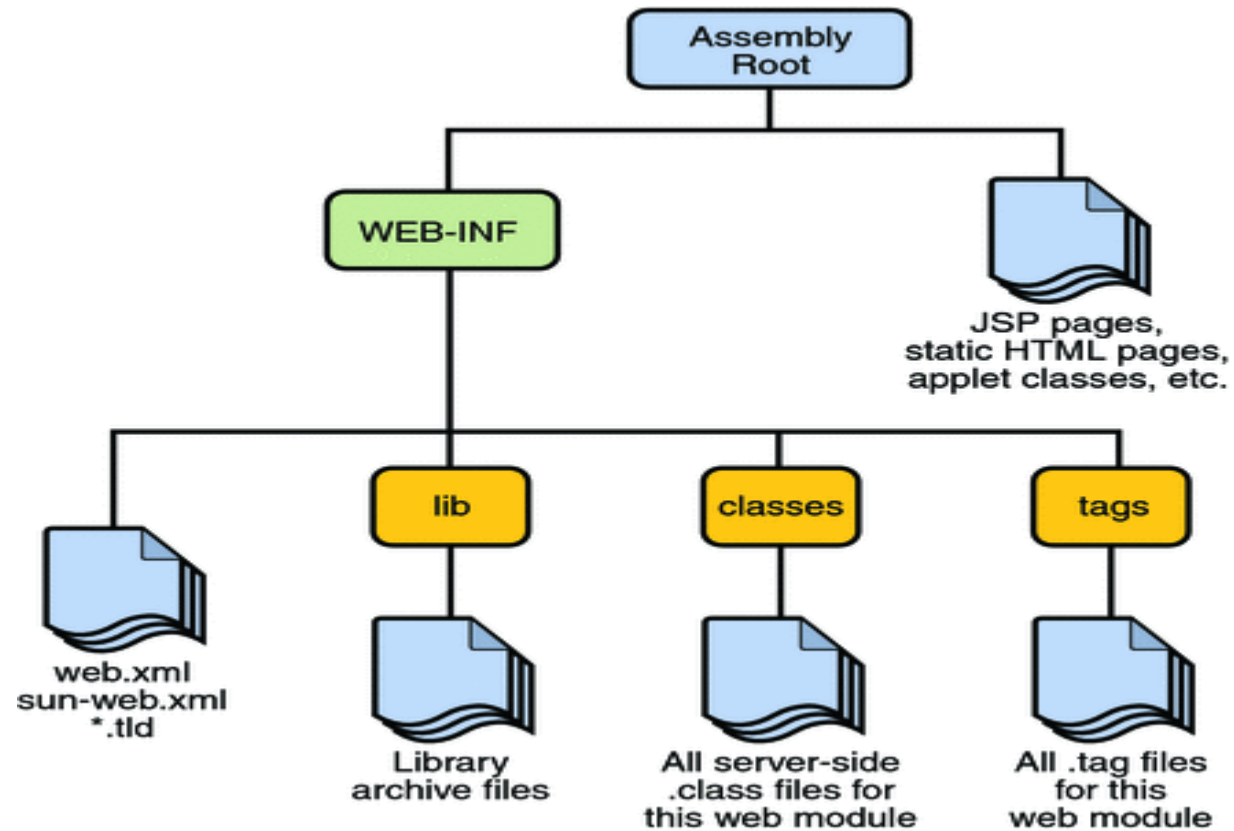
- Una aplicación web es una aplicación informática que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet. Aplicaciones como los webmails, wikis, weblogs, las tiendas en línea son ejemplos bien conocidos de aplicaciones web.



Aplicación Web

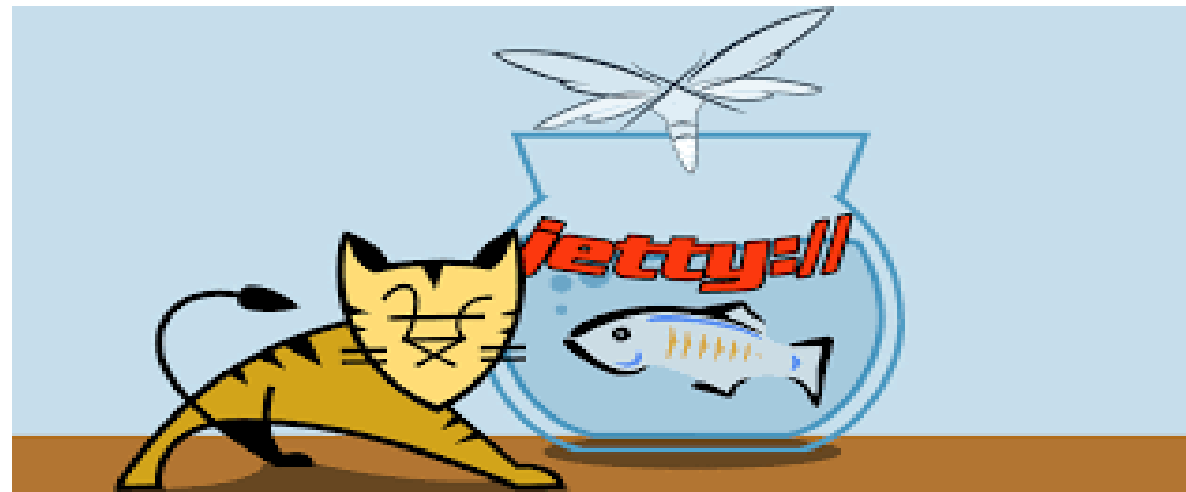
- En tecnología Java una aplicación web puede existir de dos modos:
 - Archivo war (Web Application Resource, a veces también se le suele llamar Web ARchive) que engloba a todo su contenido. Se crea del mismo modo que un fichero jar.
 - Mediante una estructura de directorios basada en la especificación definida para los Servlets

Aplicación Web



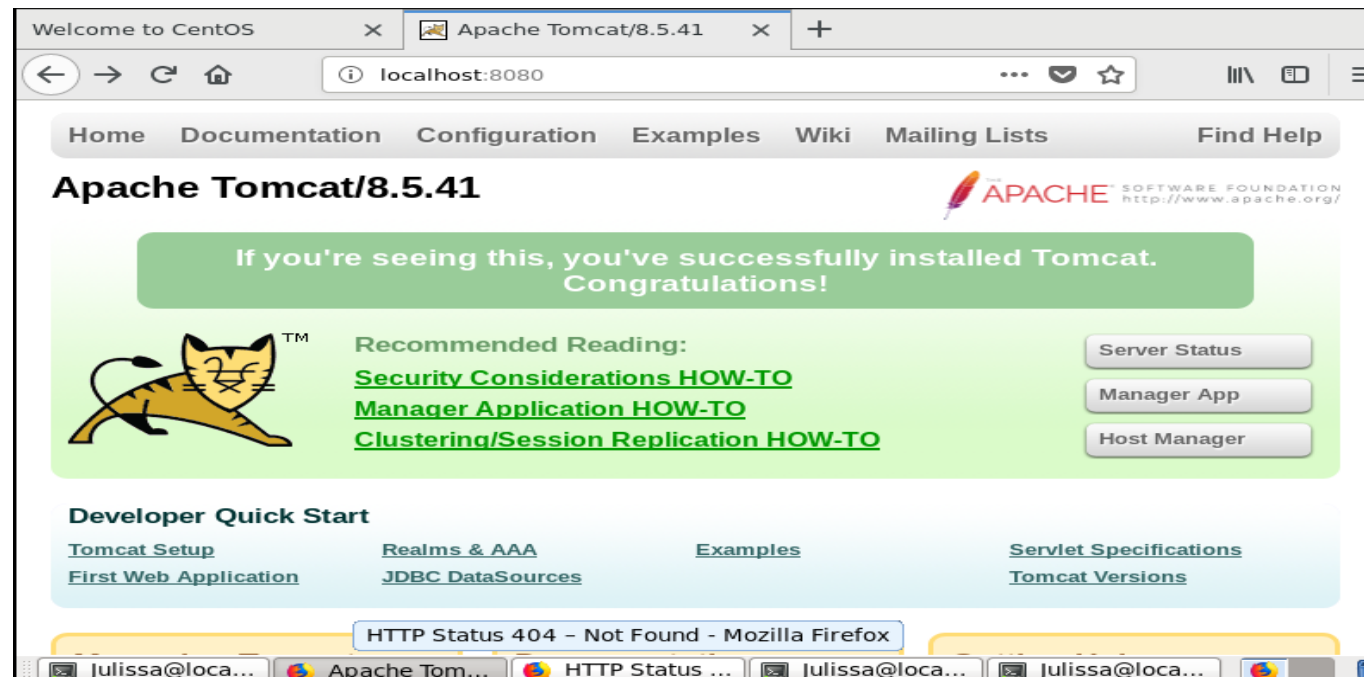
Servidor Web

- Es la implementación que hace cumplimiento del contrato de componentes web de la arquitectura J2EE. Este contrato especifica un entorno de ejecución para componentes web que incluye seguridad, concurrencia, gestión del ciclo de vida, procesamiento de transacciones, despliegue y otros servicios.



Contenedor Web

- Un contenedor web se suministra incluido en un servidor web o J2EE. (Tomcat para Java Web)
- Contenedor web = contenedor servlet + contenedor jsp

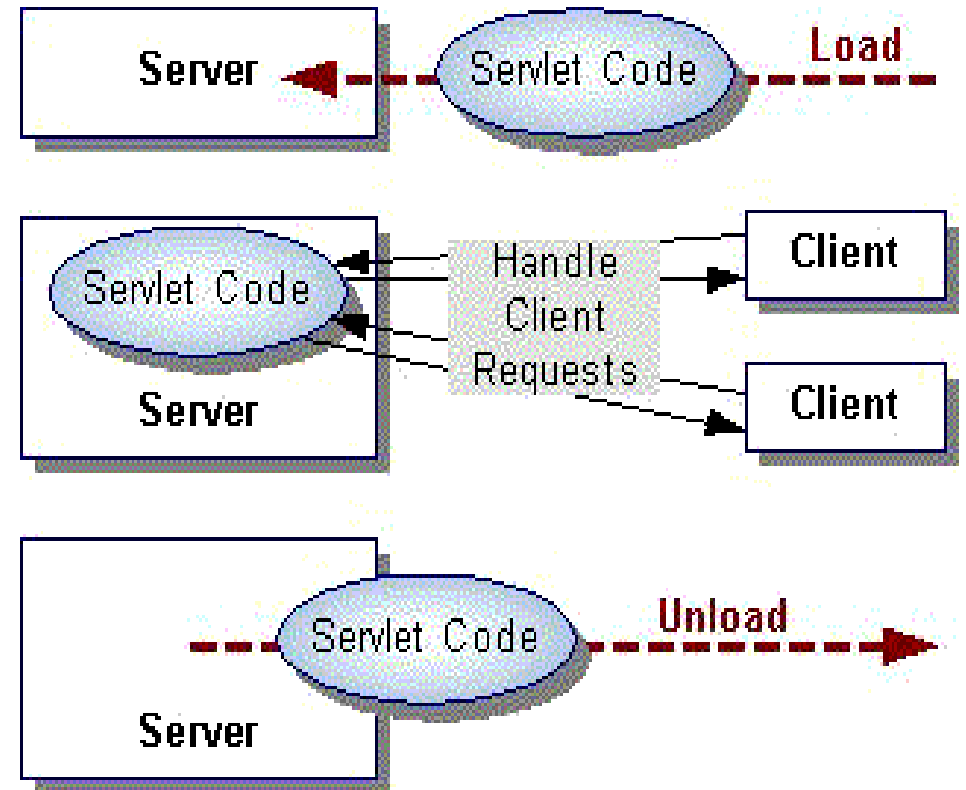


Servlets

- Un servlet es un programa Java que se guarda y ejecuta en un servidor J2EE compatible. Su función primordial es interaccionar dinámica con el cliente que solicita sus servicios.
- La interacción entre cliente y servlet se traduce, en la mayoría de los casos, en la generación y posterior envío de una respuesta codificada en html al cliente, pero no tiene por qué ser así: la respuesta podría realizarse en otro tipo de formatos tales como pdf, xml, wml, zip, jar, gif, doc, xls, etc.

Servlets

- El ciclo de vida de un servlet es el siguiente:
 - Un servidor carga e inicializa el servlet.
 - El servlet maneja cero o más peticiones de cliente.
 - El servidor elimina el servlet.



Servlets

- La clase Servlet tiene métodos para cada uno de los métodos de respuesta HTTP.
 - doGet (HttpServletRequest, HttpServletResponse)
 - doPost (HttpServletRequest, HttpServletResponse)
 - doPut (HttpServletRequest, HttpServletResponse)

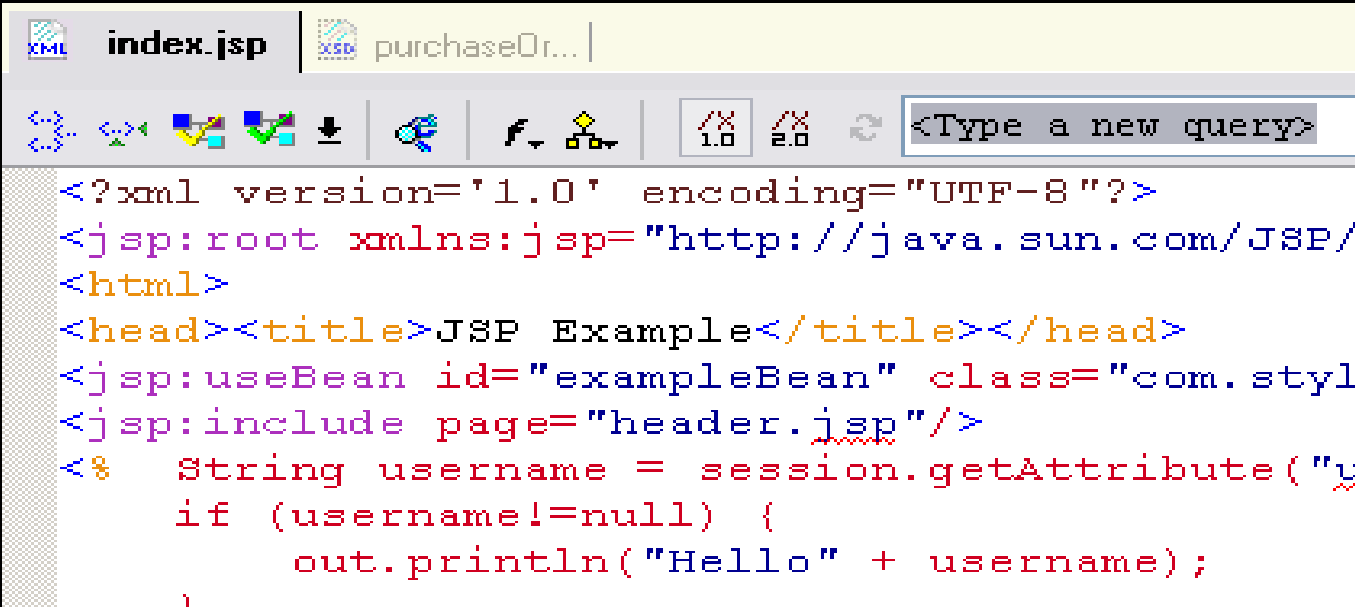
Java Server Pages

- Java Server Pages (JSP) es una tecnología Java J2EE que permite a los programadores generar contenido dinámico para web, en forma de documentos HTML, XML, o de otro tipo. Las JSP's permite al código Java y a algunas acciones predefinidas ser incrustadas en el contenido estático del documento web.



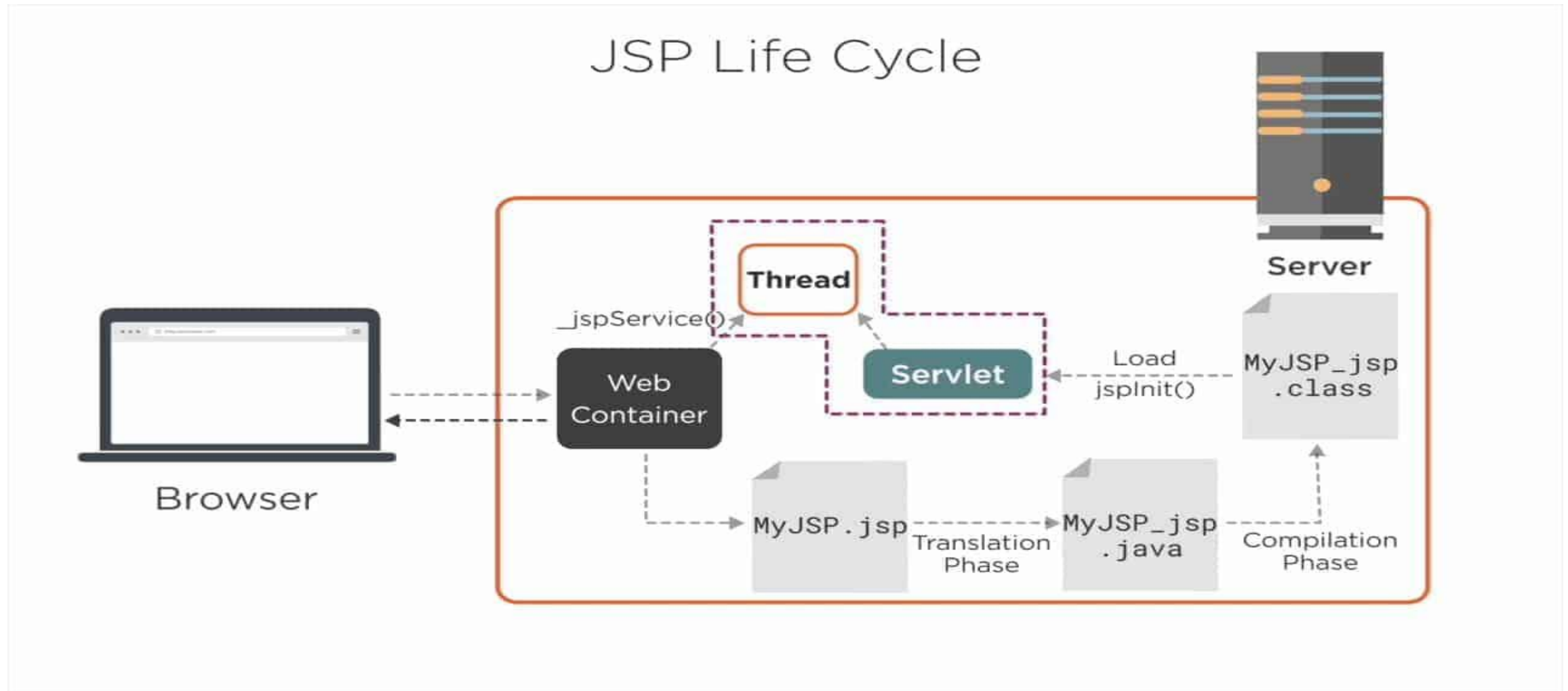
Java Server Pages

- En realidad, una página JSP es un tipo especial de servlet orientado a generar el texto de la interfaz gráfica.

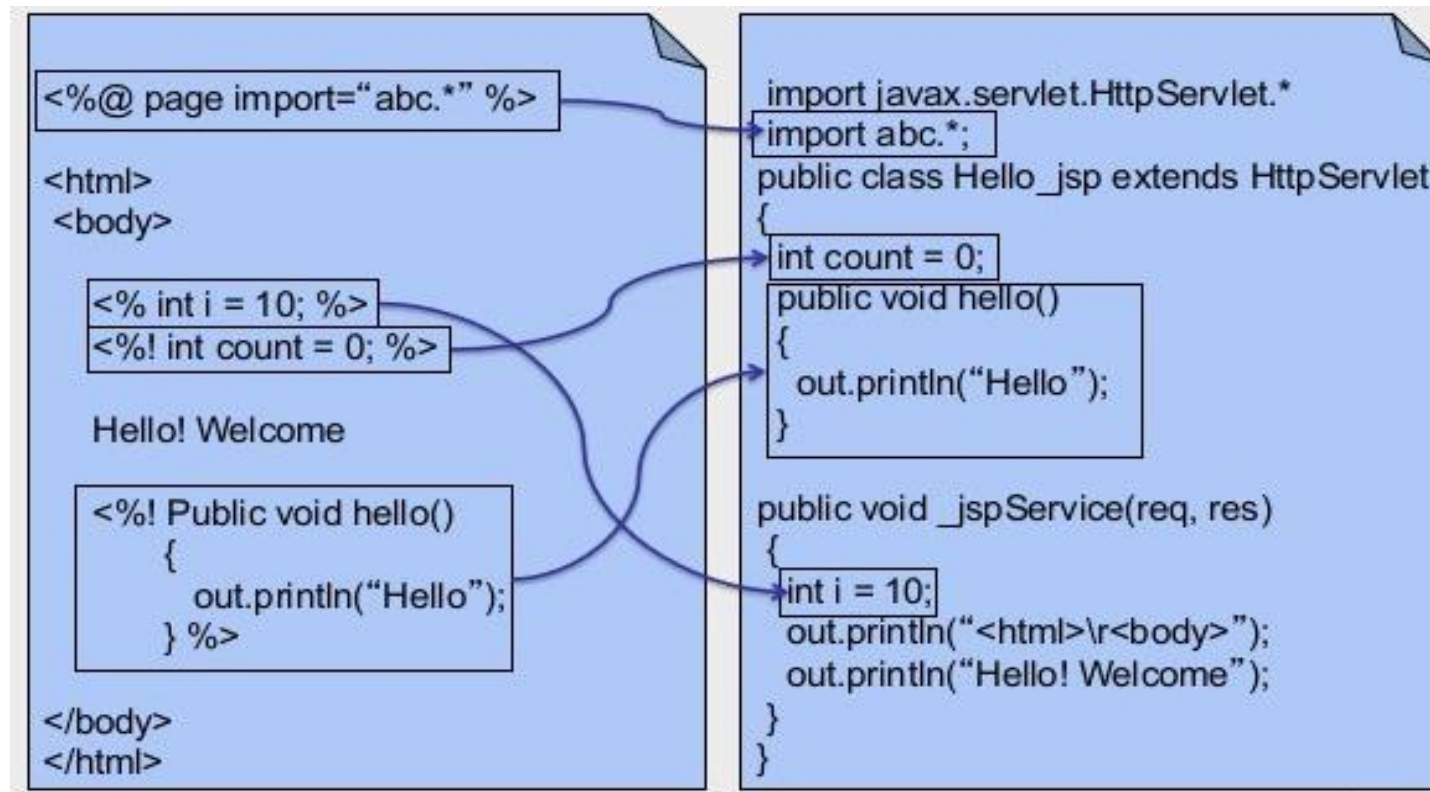


```
<?xml version='1.0' encoding="UTF-8"?>
<jsp:root xmlns:jsp="http://java.sun.com/JSP/"
<html>
<head><title>JSP Example</title></head>
<jsp:useBean id="exampleBean" class="com.styl
<jsp:include page="header.jsp"/>
<%
    String username = session.getAttribute("u
        if (username!=null) {
            out.println("Hello" + username);
        }
    %>
```

Java Server Pages



Java Server Pages



Java Server Pages

- Los componentes JSP pueden incluir los siguientes elementos:
 - Scriptlets
 - Directivas
 - Código HTML/XHTML
- Existen 8 objetos explícitos con las siguientes variables disponibles: request, response, out, session, application, config, pageContext, page y exception.

Lenguaje de Expresión

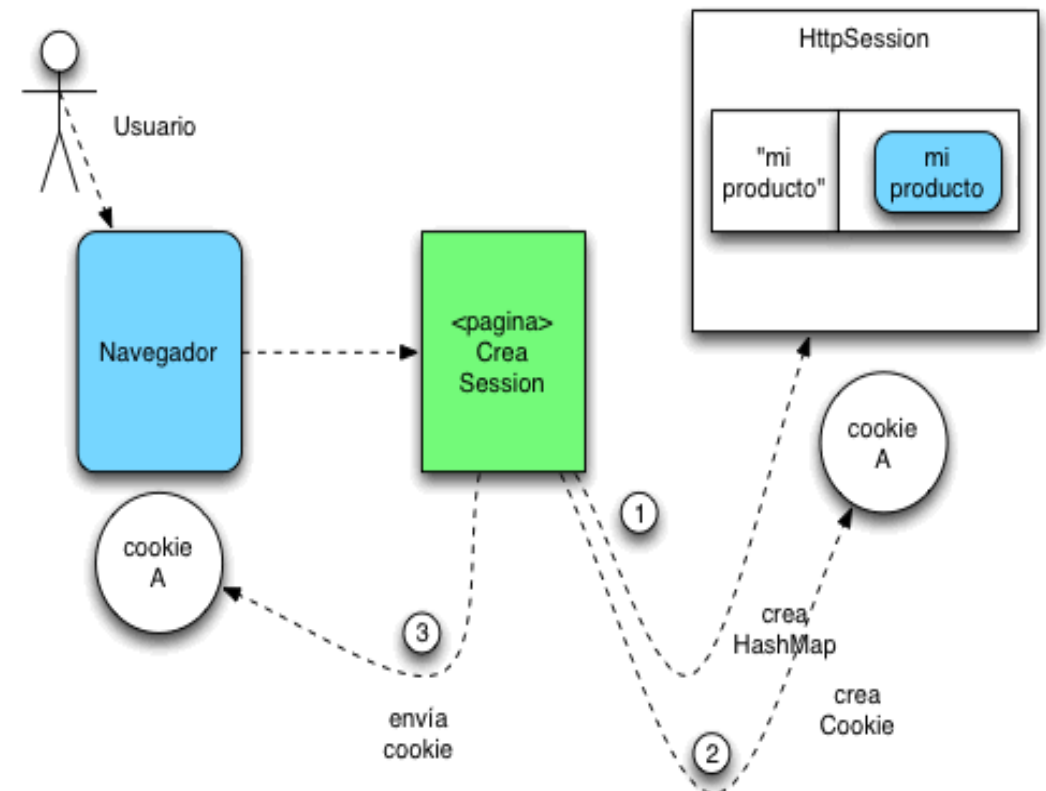
- EL (Expression Language) es un lenguaje utilizado en los componentes JSP para referenciar objetos y propiedades, sin importar su ámbito.
- Se basa en las siguientes normas:
 - Las expresiones comienzan con \${ y terminan con }. Lo que hay en medio se trata como una expresión.
 - Las expresiones se componen de:
 - Identificadores
 - Literales
 - Operadores y operadores de acceso

Lenguaje de Expresión

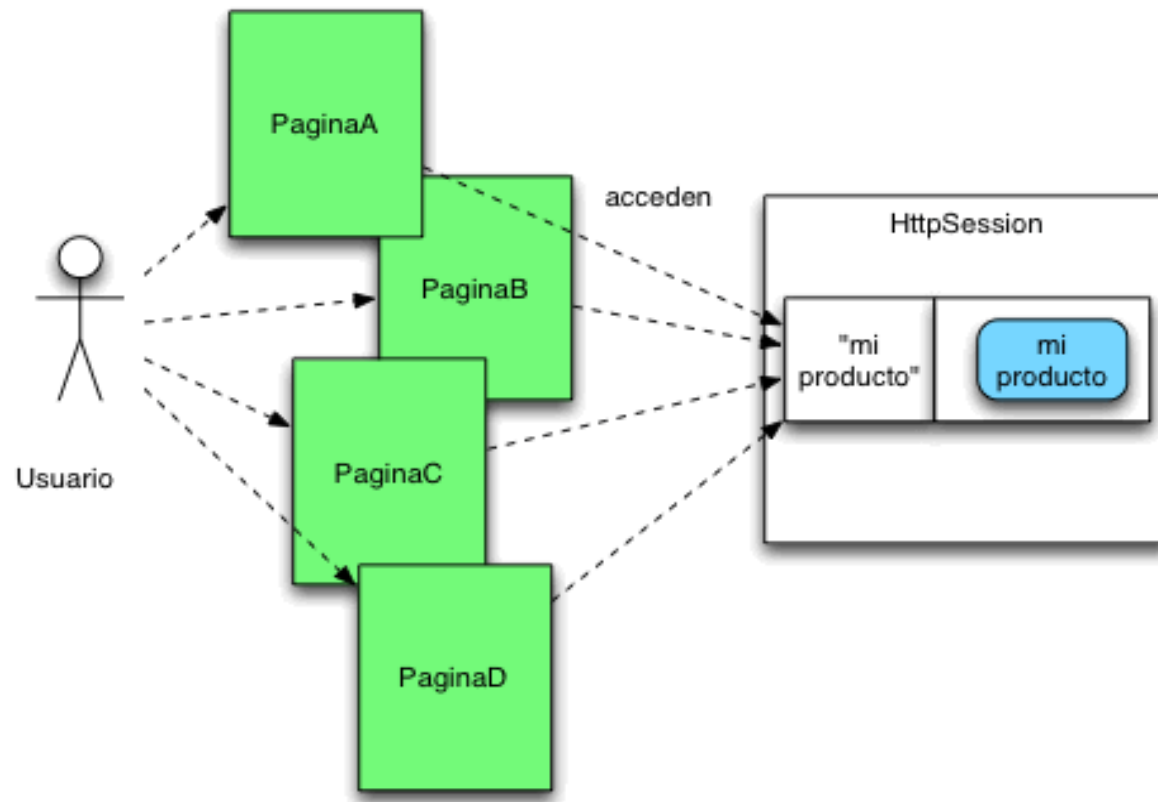
- Ejemplos de EL
 - `${variable1}`
 - `${object.propiedad}`
 - `${mapa.elemento}`
 - `${array['key']}`
 - `<c:out value="${2+2}"/>`
 - `<c:if test="${tabla.índice % 2 == 0}">es par</c:if>`
 - `<input type="text" name="usuario" value="${requestScope.usuario.nombre}"/>`

Sesiones

- Una sesión refiere a todas las conexiones que un solo cliente pudo hacer a un servidor en el curso de acceder cualquier página asociada a un uso dado. Las sesiones son específicas al usuario individual y al uso. Como resultado, cada usuario de una aplicación tiene una sesión separada y un acceso al conjunto de variables de sesión.



Sesiones



Sesiones

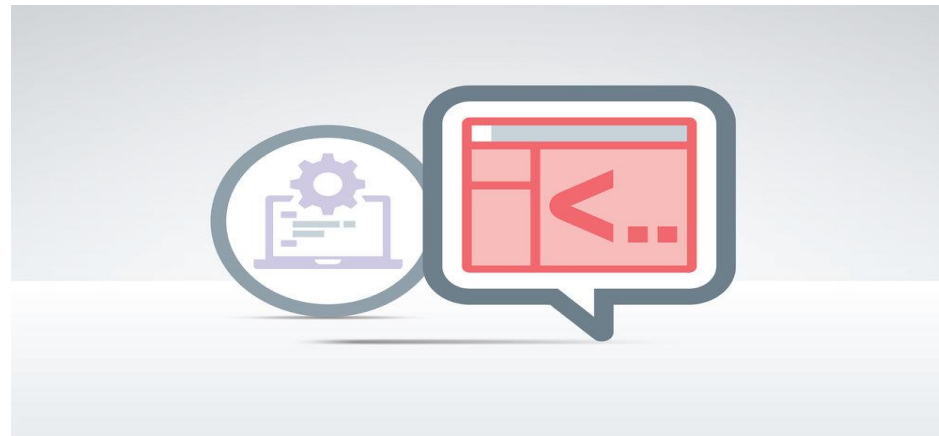
- El seguimiento de sesión es un mecanismo que los servlets utilizan para mantener el estado sobre la serie de peticiones desde un mismo usuario durante un periodo de tiempo.
- Las sesiones son compartidas por los servlets a los que accede el cliente. Esto es conveniente para aplicaciones compuestas por varios servlets.

Sesiones

- Para dar seguimiento a una sesión debemos:
 - Obtener una sesión (HttpSession) para un usuario
 - `HttpSession sesión = request.getSession();`
 - Almacenar u obtener datos desde objeto HttpSession
 - `request.setAttribute("datos", obj); //almacenar`
 - `request.getAttribute("datos"); //obtener`
 - Invalidar la sesión (opcional)
 - `request.getSession().invalidate();`

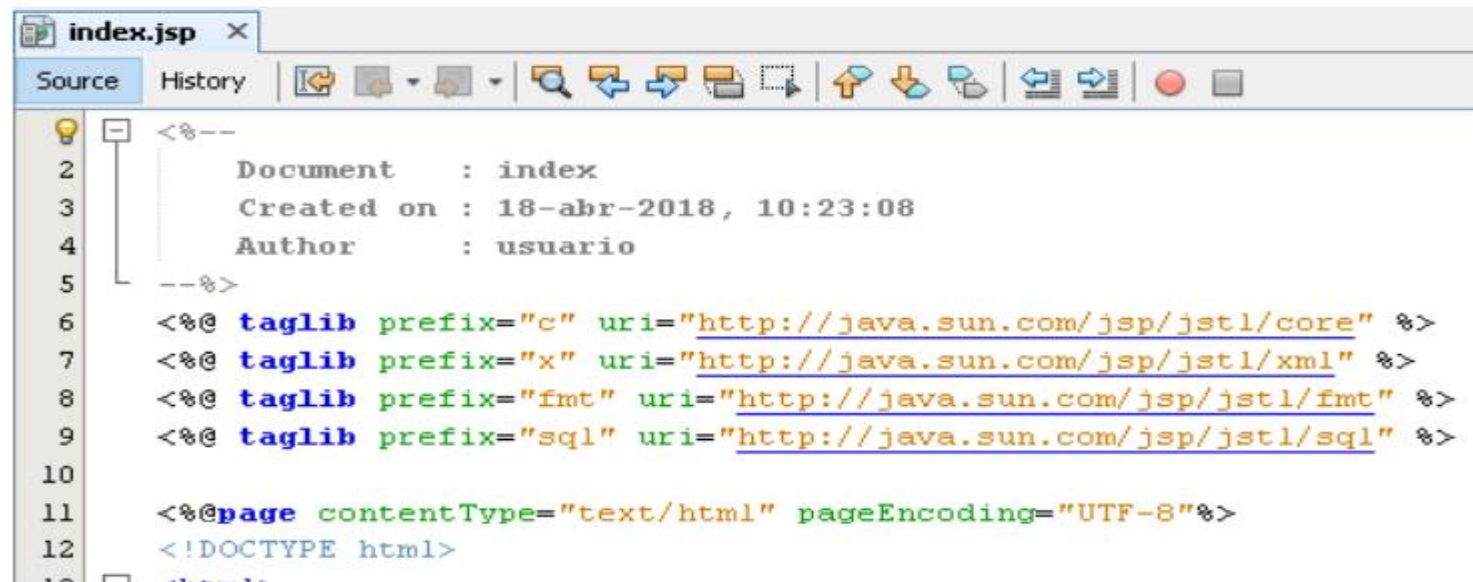
JSTL

- Java Standard Template Library (biblioteca de etiquetas estándar para Java).
- La librería JSTL es un componente dentro de la especificación del Java Enterprise Edition (JEE), es un conjunto de librerías de etiquetas simples y estándares que encapsulan la funcionalidad principal que es usada comúnmente para escribir páginas JSP.



JSTL

- Etiquetas core: Comprende las funciones script básicas como loops, condicionales, y entrada/salida.
- Etiquetas fmt: Comprende la internacionalización y formato de valores como de moneda y fechas.



```
index.jsp
Source History
Document : index
Created on : 18-abr-2018, 10:23:08
Author : usuario
-->
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="sql" uri="http://java.sun.com/jsp/jstl/sql" %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
```

Contacto

Jorge Alberto Montalvo Olvera
Ingeniero en Computación

jorge.Montalvo@gm3s.com.mx