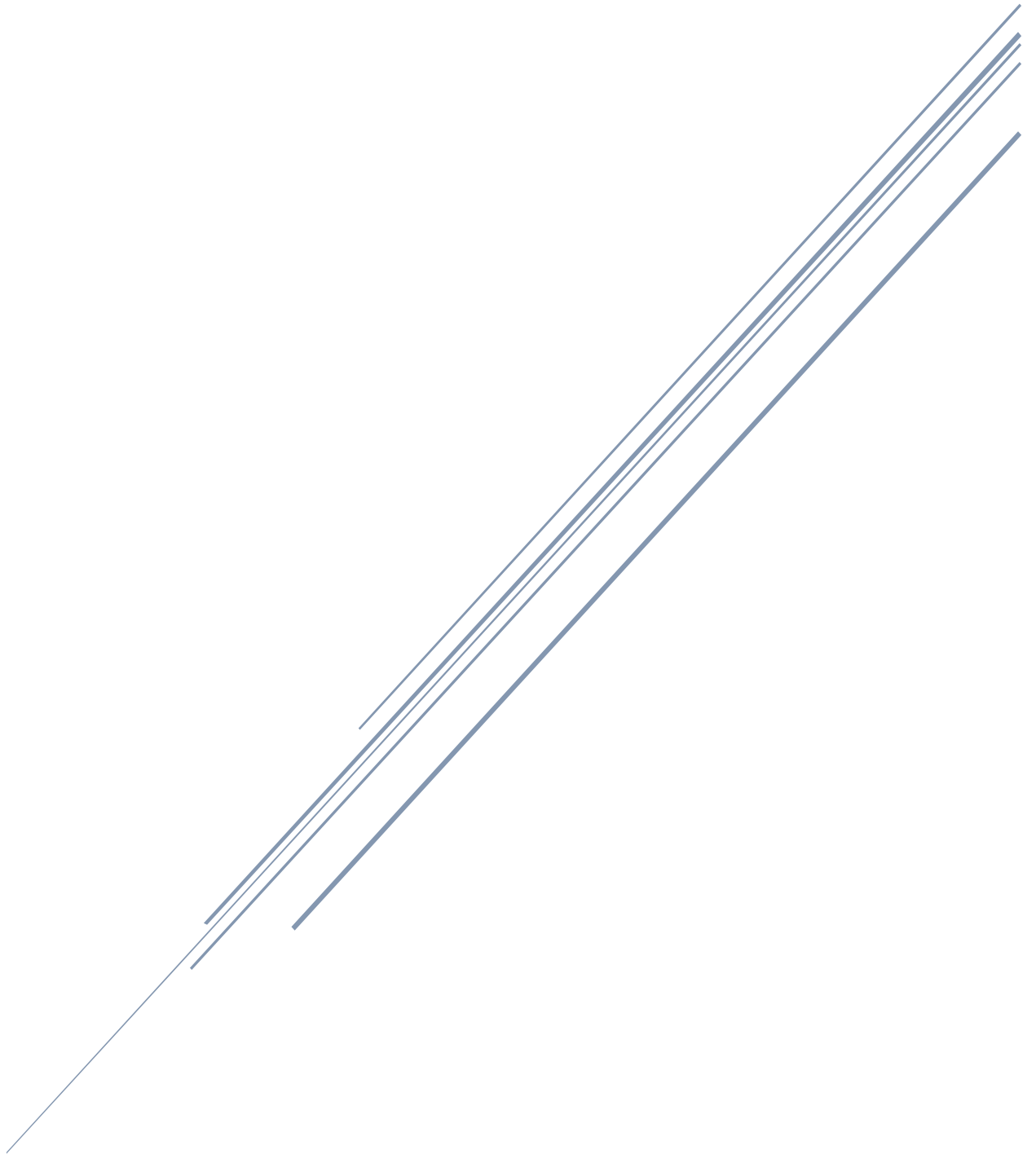


SERVLETS JSP Y EJB

Aplicaciones Web con JAVA



Enrique Hormilla Aragón

Contenido

Servlet	1
¿Qué es, para qué se utiliza?	2
Ventajas	2
¿Qué característica diferenciadora tiene?	3
JSP	4
¿Qué es, para qué se utiliza?	4
Características	4
Similitud con Servlets	4
¿Qué característica diferenciadora tiene?	5
EJB.....	6
¿Qué es, para qué se utiliza?	6
Características	6
Existen tres tipos de EJB	6
Según su funcionalidad trabajan en:	7
¿Qué característica diferenciadora tiene?	8

Servlet

¿Qué es, para qué se utiliza?

Es una clase en el lenguaje de programación Java, la cual amplía las capacidades de un servidor.

Son applets de Java que se ejecutan en servidores en vez de navegadores web.

Estos applets son un componente de una aplicación que se ejecuta en el contexto de otro programa, no puede ejecutarse de manera independiente, ofrece información gráfica y a veces interactúa con el usuario, típicamente carece de sesión y tiene privilegios de seguridad restringidos.

Se utilizan como contrapartida por parte de Java hacia otras tecnologías de contenido dinámico Web, como PHP y ASP.NET.

Características

Ventajas

1. Multiplataforma,
2. Poder trabajar en cualquier versión inferior a la que se programó
3. Puede alcanzar velocidades muy parecidas a lenguajes compilados
4. Puede ser almacenado en la memoria caché de la mayoría de los navegadores web, de modo que se cargará rápidamente cuando se vuelva a cargar la página web, aunque puede quedar atascado en la caché, causando problemas cuando se publican nuevas versiones.
5. Puede tener acceso completo a la máquina en la que se está ejecutando, si el usuario lo permite.
6. Puede trasladar el trabajo del servidor al cliente, haciendo una solución web más escalable tomando en cuenta el número de usuarios o clientes.

¿Qué característica diferenciadora tiene?

Servlet ayudaría a mejorar el uso de aplicaciones web donde la carga de procesamiento sea mayor que el ancho de banda y por lo tanto no permite tanta comunicación simultánea con el servidor, también es eficaz para aplicaciones con muchos usuarios.

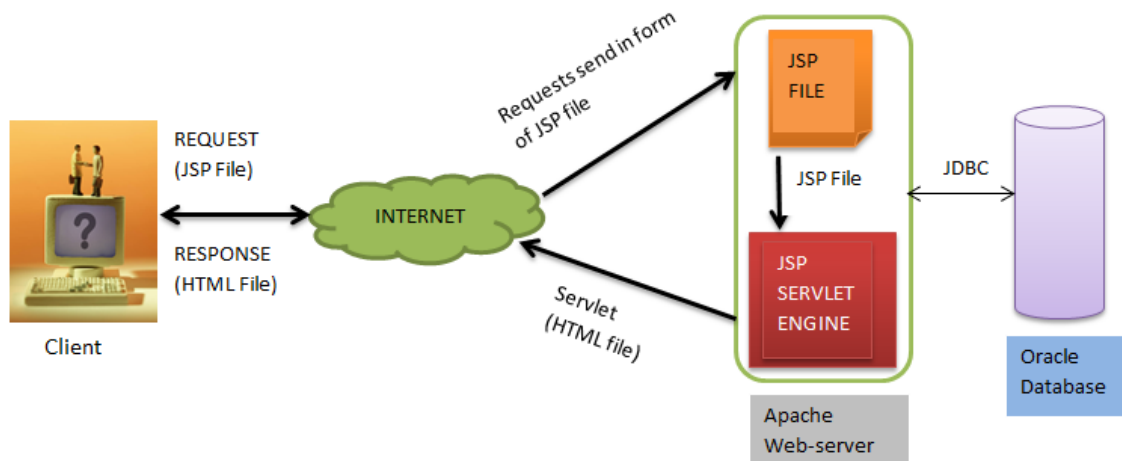


JSP

¿Qué es, para qué se utiliza?

JavaServer Pages (JSP) es una tecnología que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML, XML, entre otros tipos de documentos. JSP es similar a PHP, pero usa el lenguaje de programación Java.

JSP puede ser visto como una abstracción de alto nivel de los servlets Java. Las JavaServer Pages son traducidas a servlets en tiempo real, cada servlet es guardado en caché y reusado hasta que la JSP original es modificada.



Características

1. El rendimiento de una página JSP es el mismo que tendría el servlet equivalente. Esto hace que JSP tenga un buen desempeño y sea más eficiente que otras tecnologías web que ejecutan el código de una manera puramente interpretada.
2. El lenguaje Java es un lenguaje de propósito general

Similitud con Servlets

Los servlets y Java Server Pages (JSPs) son dos métodos de creación de páginas web dinámicas en servidor usando el lenguaje Java.

Las JSPs son en realidad una forma alternativa de crear servlets ya que el código JSP se traduce a código de servlet Java la primera vez que se le

invoca y en adelante es el código del nuevo servlet el que se ejecuta produciendo como salida el código HTML que compone la página web de respuesta.



¿Qué característica diferenciadora tiene?

Como se ha mencionado antes JSP (Java Server Pages) es una evolución de Servlets y por lo tanto es una adaptación más actual del mismo a las necesidades actuales tanto es así que actualmente los navegadores más usados están dejando de dar soporte a Servlets.



EJB

¿Qué es, para qué se utiliza?

Las **Enterprise JavaBeans** (también conocidas por sus siglas EJB) son una de las interfaces de programación de aplicaciones (API) que forman parte del estándar de construcción de aplicaciones empresariales J2EE (ahora JEE) de Oracle Corporation.

Su utilidad detalla sería ser como los servidores de aplicaciones, proveen objetos desde el lado del servidor, que son precisamente los EJB:

- Comunicación remota utilizando CORBA.
- Transacciones.
- Control de la concurrencia.
- Eventos utilizando JMS (Java Messaging Service).
- Servicios de nombres y de directorio.
- Seguridad.
- Ubicación de componentes en un servidor de aplicaciones.

Características

Los EJB proporcionan un modelo de componentes distribuido estándar del lado del servidor. El objetivo de los EJB es dotar al programador de un modelo que le permita abstraerse de los problemas generales de una aplicación empresarial (concurrencia, transacciones, persistencia, seguridad, etc.) para centrarse en el desarrollo de la lógica de negocio en sí. El hecho de estar basado en componentes permite que éstos sean flexibles y sobre todo reutilizables.

Existen tres tipos de EJB

1. EJB de Entidad: su objetivo es encapsular los objetos del lado del servidor que almacena los datos.
 - a. Persistencia gestionada por el contenedor (CMP): el contenedor se encarga de almacenar y recuperar los datos

- b. Persistencia gestionada por el bean (BMP): el propio objeto entidad se encarga, mediante una base de datos u otro mecanismo, de almacenar y recuperar los datos
- 2. EJB de Sesión: gestionan el flujo de la información en el servidor.
 - a. Con estado: almacenan datos específicos obtenidos durante la conexión con el cliente.
 - b. Sin estado: objetos distribuidos que carecen de estado asociado permitiendo por tanto que se los acceda concurrentemente.
- 3. EJB Dirigidos por Mensajes: Funcionamiento asíncrono, asociados a un tema y se activan al recibir un mensaje dirigido a dicho tema.
 - a. Funcionamiento de un Enterprise JavaBean

Los EJB se disponen en un contenedor EJB dentro del servidor de aplicaciones.

Según su funcionalidad trabajan en:

- 1. Interfaz "Home": La interfaz "Home" permite al código cliente manipular métodos de clase del EJB que no están asociados a ninguna instancia particular. La Interfaz "Home" permite crear las instancias de EJB de entidad o sesión a través del método create que puede ser sobrecargado.
- 2. Interfaz remota: La interfaz remota especifica los métodos de instancia públicos encargados de realizar las operaciones. Una sesión bean puede implementar 1 sola interfaz, con la interfaz apuntada por un tipo de cliente diferente. La interfaz local es para aquellos clientes que corren en la misma máquina virtual que el contenedor EJB. La interfaz remota es para clientes fuera del contenedor EJB.
- 3. Clase de implementación EJB: Las clases de implementación EJB las suministran los desarrolladores de aplicaciones, que facilitan la lógica de negocio ("business logic") o mantienen los datos ("business data") de la interfaz de objeto.

¿Qué característica diferenciadora tiene?

El desarrollo basado en componentes promete un paso más en el camino de la programación orientada a objetos. Con la programación orientada a objetos puedes reutilizar clases, pero con componentes es posible reutilizar un mayor nivel de funcionalidades e incluso es posible modificar estas funcionalidades y adaptarlas a cada entorno de trabajo particular sin tocar el código del componente desarrollado, esto es un gran rasgo diferenciador de los EJB.



Fuentes

Servlet
https://es.wikipedia.org/wiki/Java_Servlet
https://users.dcc.uchile.cl/~jbarrios/servlets/general.html
https://www.tutorialspoint.com/servlets/
http://www.javatpoint.com/servlet-tutorial
https://es.wikipedia.org/wiki/JavaServer_Pages
JSP
www.jsp.es/
http://www.oracle.com/technetwork/java/javaee/jsp/index.html
http://desarrolloweb.com/articulos/831.php
www.tutorialspoint.com/jsp/
EJB
https://es.wikipedia.org/wiki/Enterprise_JavaBeans
www.tutorialspoint.com/ejb/
http://www.oracle.com/technetwork/java/javaee/ejb/index.html
http://www.davidmarco.es/articulo/introduccion-a-ejb-3-1-i
http://chuwiki.chuidiang.org/index.php?title=Ejemplo_Sencillo_con_EJB