|  |  |
| --- | --- |
|  | **2016** |
|  | DAW2  Ángel Pellejero Gomes |

|  |
| --- |
| **[Servlets, JSP y EJB]** |
|  |

**Servlets:**

Es una clase del lenguaje de programación de Java del estándar Java EE, su código suele llevar HTML embebido, son utilizados comúnmente para extender las aplicaciones alojadas por servidores web, el contenido que generan los Servlets, a parte del HTML, también puede generar otro tipo de lenguaje como XML. Se caracterizan por ser eficientes, utilizar menos recursos, sólo existe una copia cargada en la máquina virtual y por cada petición se inicia un hilo, lo cual reduce el uso de memoria del servidor y el tiempo de respuesta y además tienen persistencia, por lo que siguen "vivos" una vez terminada la petición. Los Servlets son utilizados en páginas web dinámicas que utilizan formularios para obtener la información que marca el usuario y proporcionar páginas HTML que responden a esa entrada. Ejemplos de esto incluyen la utilización de carritos de compras en línea, servicios financieros, y contenido personalizado.

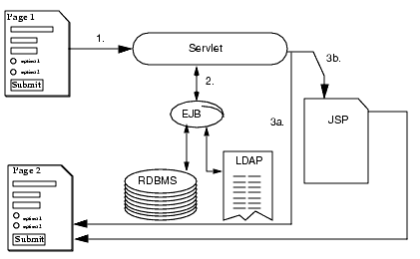
Ejemplo de código:



*Flujo de datos en un Servlet:*

Cuando un usuario hace clic en un botón submin, la información ingresada en una página de visualización se envía a un servlet. El servlet procesa los datos entrantes y genera una respuesta al contenido, normalmente en la capa lógica de negocio “EJB”. Una vez que se genera el contenido, el servlet crea una página de respuesta, esta respuesta puede ser enviada al usuario directamente pero por lo general envía el contenido a un JSP. La respuesta se devuelve al cliente y el servlet permanece en la memoria, disponible para procesar otra solicitud.

Esquema de flujo



**JSP(Java Server Pages):**

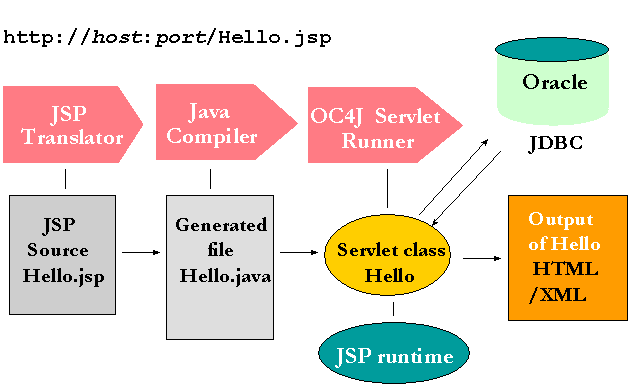
Es una tecnología para crear páginas web dinámicas que ayuda a los desarrolladores de software a crear páginas web dinámicas basadas en HTML, XML… (Similar a PHP) pero usa el lenguaje de programación Java, para su uso es necesario un servidor de Servlets como Apache Tomcat. Su función es principal es que el servidor de aplicaciones web interpreta el código JSP contenido en él para posteriormente construir el código Java del Servlet a crear, dicho Servlet es el encargado de generar el documento HTML que vera el usuario en su pantalla en el navegador. Su principal ventaja es que es Java es decir de propósito general y su portabilidad a otras plataformas, ideal para crear las clases que manejan la lógica de negocio y la de acceso a datos.

Estos JSPs serían una abstracción de los Servlets ya que el código JSP es traducido a código Servlet que será guardado en memoria caché y rehusado hasta que el JSP original sea modificado.



*Flujo de traducción y ejecución de JSP*

Para empezar se invoca el traductor JSP, traduciendo Hello.jsp y produciendo el archivo Hello.java. , a continuación se invoca el compilador Java, creando Hello.class., en este punto el archivo se ejecuta como un servlet, utilizando la biblioteca de tiempo de ejecución JSP. Para terminar la clase Hello accede a la base de datos a través de JDBC o SQLJ, según corresponda, y envía su salida al navegador.

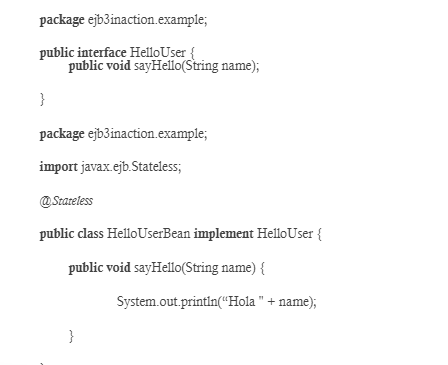


Puntos principales a modo resumen entre los Servlets y JSP:

1. JSP es un lenguaje de scripting web que pueden generar contenido dinámico mientras servlets son programas Java que ya están compilados que también crea contenido web dinámico.
2. Los servlets se ejecutan más rápido en comparación con JSP.
3. JSP puede ser compilado en Java Servlets.
4. Es más fácil el código JSP que en Java.
5. JSP y Java Servlets se utilizan generalmente en conjunto hoy en día.

**EJB (Enterprise JavaBeans):**

Es un API que forma parte del estándar de construcción de aplicaciones empresariales J2EE (ahora JEE). El objetivo de 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. EJB es el más indicado para desarrollar aplicaciones de tipo industrial/empresarial (Por ejemplo e-business). EJB a diferencia de JSP y los Servlets es un estándar que enumera una serie de especificaciones que están disponibles para dar solución a los problemas típicos de aplicaciones de este tipo, J2EE lo único que hace es juntarlas todas en un grupo para que se vea la solución completa, y asegurar que un servidor que dice ser compatible con J2EE implementa todas las del API y proporciona los servicios especificados. Algunas de sus características son su simplicidad, portabilidad, reusabilidad de componentes, poder separar la capa lógica de la de negocio, despliegue en diferentes entornos operativos, seguridad, etc.



Volviendo al JSP y Servlet como se dijo son de carácter genérico lo cual valen para cualquier tipo de aplicaciones web, pero a la hora de usar uno u otro yo creo que lo mejor sería utilizar los JSP ya que me parece un código más sencillo de programar.

Fuentes: Wikipedia y Oracle.