

# Programación en Internet

## Práctica 1: Desarrollo de páginas web con XML/XSLT (3ª parte)

### Índice

<a href="#">3.1 DESCRIPCIÓN GENERAL.....</a>	<a href="#">1</a>
<a href="#">3.2 ESPECIFICACIONES DE DISEÑO.....</a>	<a href="#">2</a>
<a href="#">2.2.1 DTD.....</a>	<a href="#">2</a>
<a href="#">2.2.2 XML.....</a>	<a href="#">3</a>
<a href="#">2.2.3 XSLT.....</a>	<a href="#">3</a>
<a href="#">2.2.4 Notas adicionales.....</a>	<a href="#">4</a>

### 3.1 Descripción General

En las dos primeras partes de esta práctica desarrollamos una página HTML con CSS que contenía la información relativa a las distintas tecnologías web de la asignatura.

Como ha quedado de manifiesto en el enunciado de la segunda parte de esta práctica, el alumno deberá tendrá preparado un fichero *recursos.css* con los estilos a aplicar a la página *recursos.html* que contiene una lista de enlaces a los recursos de interés de cada una de las unidades de la asignatura.

Con frecuencia se suelen presentar casos en los que haga falta añadir algún nuevo recurso de interés, modificar otro ya existente o incluso eliminarlo. Así, si se hubiese creado el fichero *recursos.html* de forma previa, cualquier cambio de los comentados llevaría implícito la edición del fichero *recursos.html*, cosa que no es deseable. Aunque la frecuencia de dichos cambios no es muy alta, no queremos tener que editar la página cada vez que sucedan.

En un futuro, está previsto que esta información se almacene en una base de datos, de la que se extraerán en formato XML. Mientras se decide que sistema de gestión de bases de datos utilizar y se implanta, se deberá diseñar un documento XML para representar sucintamente la información necesaria.

Además, debemos crear una hoja de estilos XSLT que nos permita transformar el documento XML en un documento HTML que tenga un formato parecido al de la figura 1 y que cumpla una serie de características que se explicarán en el siguiente punto.

Dispondremos, en principio, de un DTD externo (*recursos.dtd*) que describirá la aplicación XML que utilizaremos, un documento XML (*recursos.xml*) y una hoja de estilos de transformación XSLT (*recursos.xsl*). Por último, deberá crear un shellscript para validar y transformar automáticamente *recursos.xml* en *recursos.html* cuando sea necesario. Todos estos ficheros deberán guardarse en un directorio adicional llamado *recursos* que se añadirá a la estructura de directorios del sitio web ya creada anteriormente (estilos, imagenes, html), excepto el generado automáticamente que se almacenara por supuesto en el directorio html.



Figura 1.- Página recursos

## 3.2 Especificaciones de Diseño

### 2.2.1 DTD

El elemento raíz se llamará *recursos*. Este elemento puede contener uno o varios elementos recurso. Cada recurso constara, por orden, de los siguientes elementos:

- La *descripción* del recurso.
- El *tipo* de recurso también siendo posible el tipo *externo* (que hace referencia a un enlace a una web externa) o el tipo *local* (que hace referencia a un fichero de recurso como por ejemplo un pdf, un html, etc pero local)
- El *idioma* en el que está escrito su contenido.
- El *contenido* de que trata ese recurso que abarcará únicamente los siguientes: General, HTML, CSS, XML, JavaScript, DOM ó PHP.
- La *versión* de la tecnología que aparezca en al elemento anterior en el caso que corresponda añadirla.
- La *unidad* didáctica con la que se relaciona ese contenido. El texto que debe aparecer en el fichero XML para este elemento será: unidad 1, unidad 2, etc. Este elemento será obligatorio. La unidad didáctica para los recursos de contenido general pertenecerá a la unidad 0.

y de los siguientes atributos:

- La *url* del recurso en caso de ser de tipo *externo*.
- El *fichero* con el nombre del fichero al que hace referencia el recurso si es de tipo *local*. Todos los ficheros a los que haga referencia el documento XML deberán guardarse en un subdirectorio de *recursos* llamado *locales*.

### 2.2.2 XML

El documento XML deberá ser válido respecto del DTD especificado y estar codificado en UTF-8.

Un ejemplo de algunos recursos podría ser:

descripcion	tipo	idioma	contenido	version	Unidad	url	fichero
Consortio W3C	Externo	Inglés	General	No lleva este elemento	Unidad 2	http://...	No lleva elto
Especificación HTML	Externo	Castellano	HTML	4.01	Unidad 2	http://	No lleva elto
Enunciado práctica 3	Interno	Castellano	PHP	No lleva este elemento	Unidad 4		Practica3_200708.pdf
W3C DOM	Externo	Inglés	DOM	Level 3	Unidad 3	http://	No lleva elto

La información relativa a los diferentes recursos se puede obtener del propio campus virtual de la asignatura. Es muy recomendable que se cree el documento XML con una amplia variedad de recursos de diferentes tipos, con diferentes idiomas, de diferentes unidades, etc pero sin seguir un orden determinado.

### 2.2.3 XSLT

En esta última parte se van a generar dos transformaciones a partir del documento xml original. La primera transformación mediante la hoja de estilos XSLT *ordena\_recursos.xsl* se realizará al documento *recursos.xml* para obtener otro documento nuevo llamado *recursos\_ordenados.xml*. Este nuevo documento xml será exactamente el mismo que el anterior pero con los siguientes criterios de ordenación por orden de preferencia:

- Ordenación ascendente por unidad.
- Ordenación ascendente por contenido.
- Ordenación descendente por tipo.

La segunda transformación mediante la hoja de estilos de XSLT *recursos.xsl* se le aplicará al nuevo documento obtenido *recursos\_ordenados.xml* para obtener el documento html llamado *recursos.html*, un documento HTML 4.01 estricto, manteniendo la codificación.

Las características de esta segunda transformación como se puede observar en la figura 1 son:

- Se mostrará un único encabezado para cada grupo de recursos que pertenezca a la misma unidad didáctica. Pista: intentar acceder al hermano anterior del árbol de elementos del documento XML. Todos los encabezados tendrán el mismo formato aplicado mediante *recursos.css*.
- Cada recurso será un hipervínculo a una web externa o a un fichero local. Al igual que en el caso anterior, todos los recursos tendrán el mismo formato aplicado mediante *recursos.css*.
- Cada recurso externo debe mostrarse en una nueva ventana.
- Cada recurso presentará una numeración previa con el formato: nº.- descripción
- Todo recurso de tipo local deberá mostrar a su derecha una imagen de una carpeta cuyo texto alternativo será: “recurso local”.
- Todo recurso de tipo externo deberá mostrar a su derecha una imagen de un mapa del mundo cuyo texto alternativo será: “enlace externo”.
- Para recalcar los recursos en castellano, se añadirá una imagen con una bandera española junto a cada enlace cuyo texto alternativo será: “recurso en castellano”.

## 2.2.4 Notas adicionales

Durante esta práctica quizás necesite emplear algunas cosas que no conozca. En particular, deberá averiguar cómo:

- Realizar una ordenación múltiple, por varias claves. Pista: pueden especificarse varios elementos `xsl:sort`.
- Ordenar numéricamente. Pista: atributo `data-type` de `xsl:sort`.
- Crear nuevos atributos para un elemento. Pista: elemento `xsl:attribute`.
- Crear nuevos elementos. Pista: elemento `xsl:element`.
- Insertar texto. Pista: elemento `xsl:text`.
- Acceder a elementos anteriores en el árbol de documento XML. Pista: `preceding::`.

Observando el punto anterior, debe de haber quedado claro que se van a crear dos hojas de estilo de transformación, una para ordenar el xml y otra para diseñar el aspecto de visualización de los recursos. Pues bien, se valorará positivamente al alumno que desarrolle las dos transformaciones en un único fichero XSL. Pista: al acceder a los elementos precedentes, éstos siguen el mismo orden que tenían en el documento original incluso aun habiendo sido ordenados. Puede que haga falta el uso de los elementos `xsl:variable`, `xsl:copy-of` y `node-set()` que pertenece a una extensión de XSL (EXSL) externa a W3C. `node-set()` si es reconocida por la herramienta `xsltproc`.

La fuente de información más precisa sobre cualquiera de las tecnologías implicadas está representada por los documentos correspondientes del W3C cuyos enlaces se encuentran en el curso virtual de la asignatura (XML, XPath, XSLT).