Lenguajes de marcas y sistemas de gestión de información

ÍNDICE

XSL (eXtensible Style-sheet Language) es una familia de lenguajes estandarizada por W3C. Incluye:

- XSL-FO (XSL Formatting Objetcs): permite formatear documentos XML (vocabulario XML que especifica el formato a utilizar para visualizar XML en un medio como papel o pantalla). Obsoleto.
- XPath (XSL Path): permite navegar dentro de un documento XML.
- XSLT (XSL Transformations): permite transformar documentos XML en otros documentos.
- XQuery: permite realizar consultas (queries) a un documento XML.

1. Introducción XQuery

XQuery es un lenguaje de consulta utilizado para manipular datos XML. La información se encuentra repartida entre elementos y atributos XML.

Está basado en XPath e incluye la sintaxis **FLWOR** (For-Let-Where-Order by-Return). Es semánticamente similar a SQL, aunque incluye algunas capacidades de programación.

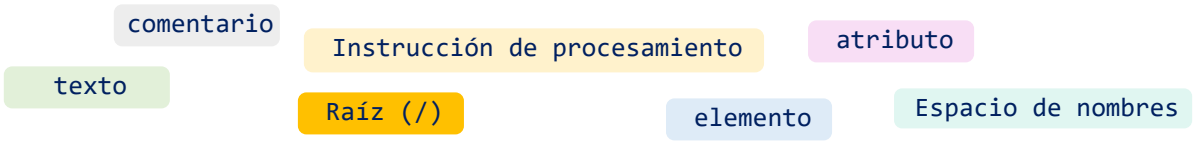
XQL (eXtensible Query Language) es la extensión de los ficheros XQuery.

Es un **estándar W3C**.

W3C XML Query (XQuery)

XQuery Tutorial (w3schools.com)

El lenguaje se basa en el modelo en árbol de la información contenida en el documento XML, que consiste en siete tipos distintos de nodo: elementos, atributos, nodos de texto, comentarios, instrucciones de procesamiento, espacios de nombres y nodos de documentos:



Utilidades XQuery:

- Procesar ficheros XML
- Manipular o extraer información almacenada en una base de datos XML
- Realizar búsquedas textuales en la web y compilar los resultados de la misma.
- Seleccionar y transformar datos de XML a otro tipo de documento, por ejemplo, XHTML.
- Obtener datos desde diferentes fuentes con vistas a ser integradas por una aplicación o servicio web.
- Dividir un documento XML que representa una serie de múltiples transacciones en varios documentos XML, uno por cada transacción.

2. Datos XML

Existen 3 formas de almacenar información en formato XML:

- Ficheros XML. No es la mejor opción, puesto que no puede garantizarse la concurrencia, integridad de atomicidad, escalabilidad o alto nivel de seguridad que ofrecen otros sistemas como las bases de datos.

- Bases de datos XML-enabled (habilitadas). Son bases de datos relacionales que convierten los documentos XML en un esquema relacional. Tienen como entrada un XML y pueden generar un XML como salida, aunque puede haber problemas de conversión por la diferente filosofía entre un modelo relacional y un modelo jerárquico. No se almacenan los documentos XML como tal, por lo que puede no sr posible recuperar un documento original una vez transformado.

- Bases de datos nativas XML. Actualmente hay una tendencia a implantar bases de datos Nosql, ya que ofrecen una serie de características que no tienen las bases de datos relaciones.

  Ejemplos:

  Características:

  - La unidad de almacenamiento es el documento XML (en una base de datos relacional sería un registro o fila)
  - Utilizan XPath y XQuery
  - Los datos tienen estructura jerárquica
  - Los datos se presentan en orden

Instala la base de datos XML-nativa eXist

3. FLWOR

Sintaxis XQuery:

- Son validas las expresiones y funciones XPath
- Para hacer referencia a una colección o a un fichero se emplea, respectivamente:

```
collection("/db/peliculas")
```

```
doc("/db/peliculas/LaJungla.xml")
```

- Es sensible a mayúsculas y minúsculas
- Las variables se definen con $
- Los comentarios se delimitan con (: :)

```
Ejemplo: (:comentario:)
```

Se tiene el fichero de ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="pelicula.xsl"?>
<película xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="pelicula.xsd"
estreno="2017">
<título>Jungle</título>
<dirección>Greg McLean</dirección>
<reparto principal="true">Daniel Radcliffe</reparto>
<reparto>Thomas Kretschmann</reparto>
<reparto>Alex Russell</reparto>
<reparto>Joel Jackson</reparto>
<sinopsis>Basada en hechos reales. A principios de la década de 1980, el joven aventurero israelí Yossi Ghinsberg viaja a Bolivia con la intención de viajar
↳ al corazón de la selva amazónica. Una vez allí, conoce a Marcus Stamm, un maestro de escuela suizo, y a su amigo Kevin Gale, un excursionista
↳ estadounidense y gran fotógrafo. Estando los tres alojados en un hostal de La Paz, Yossi conoce a un austriaco, Karl Ruprechter, que firma conocer la
↳ existencia de una tribu indígena en la jungla que los tres deberían ir a ver. Karl dice que conoce la jungla y que les puede acompañar. Yossi,
↳ emocionado por la perspectiva de conocer la jungla inexplorada y conocer tribus desconocidas, decide creerle. Regresa al hostal para convencer a
↳ Marcus y Kevin de que lo acompañen.</sinopsis>
<cartel>jungla.jpg</cartel>
</película>
```

- **For**

Iteración que se repite según un contador.

```
for $x in (1 to 5)
return <numero>{$x}</numero>
```

```
<numero>1</numero>
<numero>2</numero>
<numero>3</numero>
<numero>4</numero>
<numero>5</numero>
```

En cada iteración $i tomará el valor de cada elemento <reparto>:

1ª iteración: <reparto principal="true">Daniel Radcliffe</reparto>
2ª iteración: <reparto>Thomas Kretschmann</reparto>
Etc

```
for $i in collection("/db/peliculas")//reparto
return <actor>{data($i)}</actor>
```

```
<actor>Daniel Radcliffe</actor>
<actor>Thomas Kretschmann</actor>
<actor>Alex Russell</actor>
<actor>Joel Jackson</actor>
```

data($i) es el contenido, por ejemplo si:

$i = <reparto principal="true">Daniel Radcliffe</reparto>
data($i)= Daniel Radcliffe

- **Let**

  permite asignar valor a variables.

```
let $var:=5
```

- **Where**

  Condición.

```
for $i in collection("/db/peliculas")//reparto
where $i/@principal='true'
return <actor>{data($i)}</actor>
```

```
<actor>Daniel Radcliffe</actor>
```

- **Order by**

  Ordena el resultado.

```
for $i in collection("/db/peliculas")//reparto
order by $i
return <actor>{data($i)}</actor>return data($i)
```

```
<actor>Alex Russell</actor>
<actor>Daniel Radcliffe</actor>
<actor>Joel Jackson</actor>
<actor>Thomas Kretschmann</actor>
```

- **Return**

  Devuelve un valor.

Actividad 1. Crear XQuery

1.1 Crear una colección de datos
1.2 Realiza las siguientes Xqueries