

Francesco Fevoli Chacón – 1DAM

XPath

- elemento: Elemento de nombre elemento
- /elemento: Elemento de nombre elemento ubicado en la raíz del documento
- e1/e2: Elemento e2 hijo directo del elemento e1
- e1//e2: Elemento e2 descendiente (hijo, nieto, bisnieto...) del elemento e1
- //elemento: Elemento de nombre elemento ubicado en cualquier nivel por debajo de la raíz del documento
- @atributo: Atributo de nombre atributo
- *: Cualquier elemento (todos los elementos)
- @*: Cualquier atributo (todos los atributos)
- . : Nodo actual
- ..: Nodo padre
- espNom:* : Todos los elementos en el espacio de nombres de prefijo espNom
- @espNom:* : Todos los atributos en el espacio de nombres de prefijo espNom

Funciones:

- round, ceiling, floor, abs
- substring, starts-with, contains, normalize-space, translate, string-length
- name, root, node, comment
- count, avg, min, max, sum

Acceso a elementos mediante ejes:

- self::* nodo de contexto. Equivalente a .
- child::* nodos “hijo” del nodo de contexto.
- parent::* nodo padre del nodo contexto. Equivale a ..
- ancestor::* “antepasados” del nodo de contexto
- ancestor-or-self::* “antepasados” del nodo de contexto además del nodo de contexto
- descendant::* nodos “descendientes” (hijo, nieto...) del nodo de contexto
- descendant-or-self::* nodos “descendientes” (hijo, nieto...) del nodo de contexto además del propio nodo de contexto. Equivalente a //
- following::* nodos después del nodo de contexto, excluyendo nodos descendientes, atributos y nodos del espacio de nombres
- preceding::* nodos antes del nodo de contexto, excluyendo nodos ascendientes, atributos y los nodos de espacio de nombres.
- preceding-sibling::* “hermanos mayores” del nodo de contexto.
- following-sibling::* “hermanos menores” del nodo de contexto.
- attribute::* Atributos del nodo de contexto. Equivale a @

XQuery

FLWOR:

- for (at)
- let (\$var :=)
- where (para elementos seleccionados por for)

- order by
- return

Otras funciones:

- declare function
- if, else
- current-date, current-time, day-from-date, hours-from-time

Ejemplo:

```
for $agua in doc('etc/factbook.xml')/mondial/sea | /mondial/lake
let $nombre := normalize-space($agua/@name), $tipo := $agua/name()
order by $nombre
return <agua tipo='{ $tipo }'> { $nombre } </agua>
```

XSLT

Enlace de documentos XML con su hoja de estilo: <?xml-stylesheet ?>

Elemento raíz: <xsl:stylesheet|xsl:transform version="1.0"

xmlns:xsl="http://www.w3.org/1999/XSL/Transform"> (version, xmlns, exclude-result-prefixes, href) </xsl:stylesheet|xsl:transform>

- xsl:output (method, version, encoding, indent, omit-xml-declaration, standalone)
- xsl:template (name, match, priority, mode), xsl:apply-templates (select, mode)

Ejemplo:

```
<xsl:template match="/">
  <xsl:apply-templates select="contratoAlquiler/arrendatario" />
</xsl:template>
<xsl:template match="/contratoAlquiler/arrendatario">
```

Texto de ejemplo

```
</xsl:template>
```

Resolución de conflictos: (prioridades de 9 a -9)

- “*” , “@*” , “text ()” y otros patrones para los que no se indica un nombre concreto -0.5
- “Prefijo:*” , “@prefijo:*” y otros patrones para los que se indica el espacio de nombres, pero no un nombre concreto -0.25
- “producto” , “@unidades” y otros patrones donde se indiquen nombres de elementos y atributos concretos 0
- “proveedor/representante” , “productos/producto/precio” ,
- “precio/@unidades” y otros patrones para los que se indica una jerarquía (además del nombre del elemento y/o atributo)
- xsl:call-template (name)
- xsl:value-of (select, disable-output-escaping), xsl:text
- xsl:for-each (select)
- xsl:sort (select, lang, data-type, order, case-order)
- xsl:if (test)
- xsl:choose, xsl:when, xsl:otherwise
- xsl:element (name, namespace, use-attribute-sets), xsl:attribute (name, namespace), xsl:comment, xsl:processing-instruction, xsl:copy-of (select), xsl:copy