

Transformación de documentos XML

Introducción a XSLT



Jorge Castellanos Vega

Introducción

- **XSL** → Lenguaje extensible de hojas de estilo
 - Familia de lenguajes basados en XML que describen la forma de transformar la información contenida en un documento para que sea representada en un medio
 - Parecido a lo que hace un archivo CSS pero aplicado a XML
 - Puesto que XML no tiene etiquetas predefinidas los navegadores no saben cómo mostrarlo
 - La etiqueta <table> en XML no tiene porque ser el inicio de una declaración de tabla HTML
 - XSL describe cómo deben mostrarse los documentos XML

Introducción

- **Lenguajes XSL**
 - XSLT – Lenguaje de transformación de documentos XML
 - XPath – Lenguaje para navegar en documentos XML
 - XSL-FO – Lenguaje para formatear documentos XML (desaparecido en 2013)
 - XQuery – Lenguaje de consultas en documentos XML

XSLT

- **XSL Transformations**

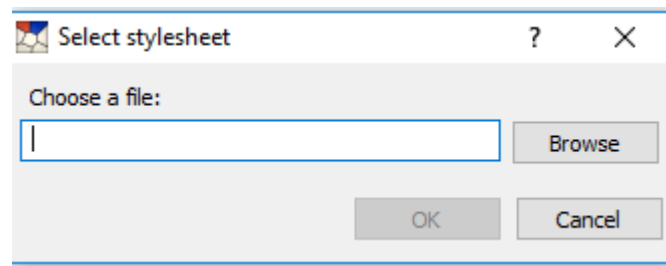
- Parte más importante de XSL
- Permite transformar un documento XML en:
 - Otro documento XML
 - Otro tipo de documento reconocible por un navegador
- Permite añadir o eliminar elementos y atributos en el archivo de salida. También cambiar el orden o decidir qué elementos mostrar y cuáles no.



XSLT

- **Xsl Transformations**

- Utiliza **Xpath** para navegar en documentos XML
- Durante el proceso de transformación XSLT utiliza **Xpath** para definir partes del documento origen que coincidirán con una o más plantillas determinadas.
- Cuando se encuentra una coincidencia XSLT transformará la parte coincidente del documento origen en el documento de destino
- XSLT funciona en la mayoría de navegadores
- En **XML Copy Editor** podemos aplicar una transformación sobre un documento XML pulsando F8 y seleccionando el archivo XSL a aplicar



Ejemplo

- Archivo XML → Canciones.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <MisCancionesPreferidas>
3   <archivo>
4     <canCIÓN>Hangar 18</canCIÓN>
5     <artista>Megadeth</artista>
6     <disco>Rust in Peace</disco>
7   </archivo>
8   <archivo>
9     <canCIÓN>Peace Sells</canCIÓN>
10    <artista>Megadeth</artista>
11    <disco>Peace Sells...But Who's Buying</disco>
12  </archivo>
13  <archivo>
14    <canCIÓN>Master of Puppets</canCIÓN>
15    <artista>Metallica</artista>
16    <disco>Master of Puppets</disco>
17  </archivo>
18  <archivo>
19    <canCIÓN>Among The Living</canCIÓN>
20    <artista>Anthrax</artista>
```

```
21    <disco>Among The Living</disco>
22  </archivo>
23  <archivo>
24    <canCIÓN>For Whom The Bell Tolls</canCIÓN>
25    <artista>Metallica</artista>
26    <disco>Ride The Lightning</disco>
27  </archivo>
28 </MisCancionesPreferidas>
```

Ejemplo

- Archivo XSL → Canciones.xsl

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:template match="/">
4 <html>
5 <body>
6 | <h1>Mis Canciones favoritas</h1>
7 <table style="border: 1px solid blue;">
8 <tr style="background-color:lightblue;">
9 | <th style="text-align:left">Título</th>
10 | <th style="text-align:left">Artista</th>
11 | <th style="text-align:left">Disco</th>
12 </tr>
13 <xsl:for-each select="MisCancionesPreferidas/archivo">
14 <tr>
15 | <td><xsl:value-of select="canción"/></td>
16 | <td><xsl:value-of select="artista"/></td>
17 | <td><xsl:value-of select="disco"/></td>
18 </tr>
19 </xsl:for-each>
20 </table>
21 </body>
22 </html>
23 </xsl:template>
24 </xsl:stylesheet>
25
```

Ejemplo

- Resultado de la transformación

Mis Canciones favoritas

Titulo	Artista	Disco
Hangar 18	Megadeth	Rust in Peace
Peace Sells	Megadeth	Peace Sells...But Who's Buying
Master of Puppets	Metallica	Master of Puppets
Among The Living	Anthrax	Among The Living
For Whom The Bell Tolls	Metallica	Ride The Lightning



[Enlace: XSLT Tryit Editor](#)

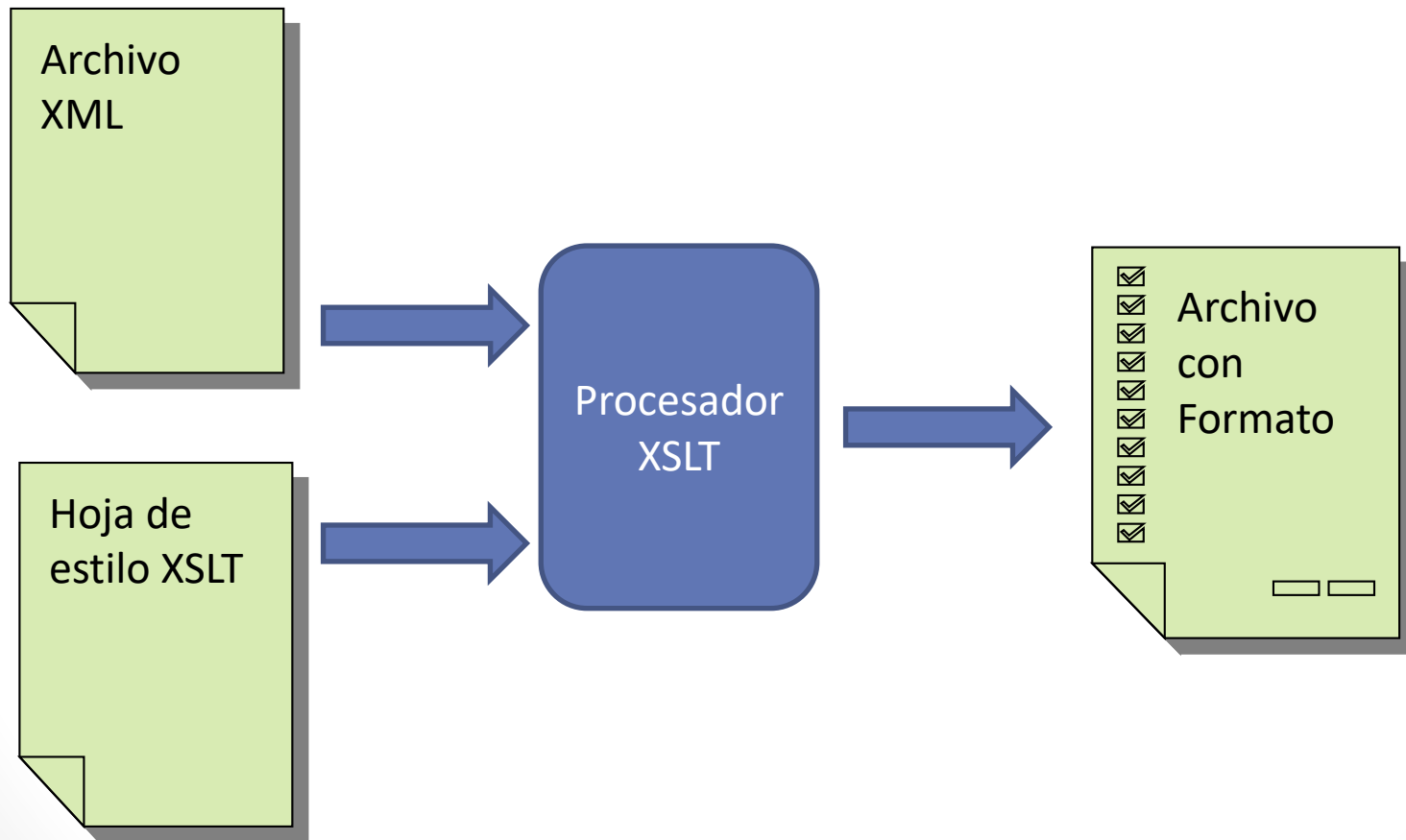
Ejemplo

- Resultado de la transformación

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <html>
3 <body>
4   <h1>Mis Canciones favoritas</h1>
5   <table style="border: 1px solid blue;">
6     <tr style="background-color:lightblue;">
7       <th style="text-align:left">Título</th>
8       <th style="text-align:left">Artista</th>
9       <th style="text-align:left">Disco</th>
10    </tr>
11    <tr>
12      <td>Hangar 18</td>
13      <td>Megadeth</td>
14      <td>Rust in Peace</td>
15    </tr>
16    <tr>
17      <td>Peace Sells</td>
18      <td>Megadeth</td>
19      <td>Peace Sells...But Who's Buying</td>
20    </tr>
```

```
21 <tr>
22   <td>Master of Puppets</td>
23   <td>Metallica</td>
24   <td>Master of Puppets</td>
25 </tr>
26 <tr>
27   <td>Among The Living</td>
28   <td>Anthrax</td>
29   <td>Among The Living</td>
30 </tr>
31 <tr>
32   <td>For Whom The Bell Tolls</td>
33   <td>Metallica</td>
34   <td>Ride The Lightning</td>
35 </tr>
36 </table>
37 </body>
38 </html>
```

Esquema de funcionamiento



Declaración

- Podemos declarar el documento de cualquiera de las dos maneras siguientes:

```
1 <?xml version="1.0" encoding="UTF-8"?>  
2 <xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
```

```
1 <?xml version="1.0" encoding="UTF-8"?>  
2 <xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
```

XSLT - Elementos

- Elemento **plantilla**

- Definido como **<xsl:template match="ExpresiónXPath">**
- El atributo match sirve para indicar dónde aplicar la plantilla. Esta se aplicará a los elementos hijos del nodo que coincida con la expresión del atributo
 - P.e: match="/" define el documento entero

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <xsl:stylesheet version="1.0"
3  xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4  <xsl:template match="/">
5  <html>
6  <body>
7  |  <h1>Mis Canciones favoritas</h1>
8  |  <table style="border: 1px solid blue;">
9  |  |  <tr style="background-color: lightblue;">
10 |  |  |  <th style="text-align: left">Título</th>
11 |  |  |  <th style="text-align: left">Artista</th>
12 |  |  |  <th style="text-align: left">Disco</th>
```

XSLT - Elementos

- Elemento **valor de**
 - Formato **<xsl:value-of select="ExpresiónXPath">**
 - Permite extraer el valor de un elemento XML y añadirlo en el archivo de salida

```
<tr>
  <td><xsl:value-of select="MisCancionesPreferidas/archivo/canción"/></td>
  <td><xsl:value-of select="MisCancionesPreferidas/archivo/artista"/></td>
  <td><xsl:value-of select="MisCancionesPreferidas/archivo/disco"/></td>
</tr>
```

- Es posible utilizar expresiones de **Xpath** más complejas para seleccionar elementos

XSLT - Elementos

- Elemento **for each**
 - Definido como **<xsl:for each select="expresión-XPath">**
 - Permite recorrer un conjunto de nodos XML

```
<xsl:for-each select="MisCancionesPreferidas/archivo">
  <tr>
    <td><xsl:value-of select="canción"/></td>
    <td><xsl:value-of select="artista"/></td>
    <td><xsl:value-of select="disco"/></td>
  </tr>
</xsl:for-each>
```

- En el ejemplo obtenemos los valores de los nodos **canción**, **artista** y **disco** en la ruta XPath **MisCancionesPreferidas/archivo**

XSLT - Elementos

- Elemento **for each**
 - Podemos modificar la expresión Xpath para seleccionar ciertos valores:
 - Los operadores de filtrado son:
 - Igual =
 - Distinto !=
 - < menor que
 - > mayor que

```
<xsl:for-each select="MisCancionesPreferidas/archivo[artista='Megadeth']">
  <tr>
    <td><xsl:value-of select="canción"/></td>
    <td><xsl:value-of select="artista"/></td>
    <td><xsl:value-of select="disco"/></td>
  </tr>
</xsl:for-each>
```

- En el ejemplo obtenemos los valores de los nodos **canción**, **artista** y **disco** para el artista indicado

XSLT - Elementos

- Elemento **ordenar**

- Permite ordenar la salida

- Por defecto:

- Tipo texto
- Ascendente
- Upper-first

```
<xsl:sort
  select = string-expression
  lang = { nmtoken }
  data-type = { "text" | "number" | QName }
  order = { "ascending" | "descending" }
  case-order = { "upper-first" | "lower-first" } >
</xsl:sort>
```

- El atributo select indica el elemento XML sobre el que ordenar
- Simplemente hay que añadir un elemento **<xsl:sort>** dentro de **<xsl:for-each>**

```
<xsl:for-each select="MisCancionesPreferidas/archivo[artista='Megadeth']">
  <xsl:sort select="canción"/>
  <tr>
    <td><xsl:value-of select="canción"/></td>
    <td><xsl:value-of select="artista"/></td>
    <td><xsl:value-of select="disco"/></td>
  </tr>
</xsl:for-each>
```


XSLT - Elementos

- Elemento **alternativa simple**
 - Definido como **<xsl:if test="expresión">**
 - Se añade dentro del elemento **<xsl:for-each>**

```
<xsl:for-each select="MisCancionesPreferidas/archivo">
  <xsl:if test="artista='Megadeth'">....
    <tr>
      <td><xsl:value-of select="canción"/></td>
      <td><xsl:value-of select="artista"/></td>
      <td><xsl:value-of select="disco"/></td>
    </tr>
  </xsl:if>....
</xsl:for-each>
```

XSLT - Elementos

- Elemento **Alternativa Múltiple**
 - Definido como `<xsl:choose>`

Sintaxis

```
<xsl:choose>
  <xsl:when test="condición1">
    || salida para condición 1
  </xsl:when>
  <xsl:when test="condición2">
    || salida para condición 2
  </xsl:when>
  .....
  <xsl:otherwise>
    ..... salida para cualquier otro caso
  </xsl:otherwise>
</xsl:choose>
```

Ejemplo

```
<xsl:for-each select="MisCancionesPreferidas/archivo">
  <xsl:choose>
    <xsl:when test="artista='Megadeth'">
      <tr>
        <td>MeGaDeTH</td>
        <td><xsl:value-of select="canción"/></td>
        <td><xsl:value-of select="artista"/></td>
        <td><xsl:value-of select="disco"/></td>
      </tr>
    </xsl:when>
    <xsl:when test="artista='Metallica'">
      <tr>
        <td>MeTaLLiCa</td>
        <td><xsl:value-of select="canción"/></td>
        <td><xsl:value-of select="artista"/></td>
        <td><xsl:value-of select="disco"/></td>
      </tr>
    </xsl:when>
  </xsl:choose>
</xsl:for-each>
```

XSLT - Elementos

- Elemento **aplicar plantilla**
 - Formato **<xsl:apply-templates select="AtributoSelección">**
 - Selecciona todos los nodos coincidentes con los nodos indicados en el atributo **select** y los itera aplicando las plantillas que estén definidas para ellos.

XSLT - Elementos

- Elemento **aplicar plantilla** – ejemplo (1/2)

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0"
3   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4   <xsl:template match="/"><!-- plantilla principal -->
5     <html>
6     <body>
7       <h1>Mis Canciones</h1>
8       <xsl:apply-templates/><!-- llamada para aplicar el resto de plantillas -->
9     </body>
10    </html>
11  </xsl:template>
12
13  <xsl:template match="archivo"><!-- plantilla para el nodo archivo -->
14    <p>
15      <xsl:apply-templates select="canción"/><!-- aplica la plantilla definida para el nodo canción -->
16      <xsl:apply-templates select="artista"/><!-- aplica la plantilla definida para el nodo artista -->
17      <xsl:apply-templates select="disco"/><!-- aplica la plantilla definida para el nodo disco -->
18    </p>
19  </xsl:template>
20
```

XSLT - Elementos

- Elemento **aplicar plantilla** – ejemplo (2/2)

```
21 <xsl:template match="canción"> <!-- plantilla definida para el nodo canción -->
22 Canción <span style="color:darkblue">
23   <xsl:value-of select="."/></span>
24   <br />
25 </xsl:template>
26
27 <xsl:template match="artista"> <!-- plantilla definida para el nodo artista -->
28 Artista <span style="color:blue">
29   <xsl:value-of select="."/></span>
30   <br />
31 </xsl:template>
32
33 <xsl:template match="disco"> <!-- plantilla definida para el nodo disco -->
34 Disco <span style="color:lightblue">
35   <xsl:value-of select="."/></span>
36   <br />
37 </xsl:template>
38 </xsl:stylesheet>
39
```

XSLT - Elementos

- Elemento **aplicar plantilla** - resultado

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <html>
3 <body><h1>Mis Canciones</h1>
4 <p>
5   Canción <span style="color:darkblue">Hangar 18</span><br/>
6   Artista <span style="color:blue">Megadeth</span><br/>
7   Disco <span style="color:lightblue">Rust in Peace</span><br/></p>
8 <p>
9   Canción <span style="color:darkblue">Peace Sells</span><br/>
10  Artista <span style="color:blue">Megadeth</span><br/>
11  Disco <span style="color:lightblue">Peace Sells...But Who's Buying</span><br/></p>
12 <p>
13  Canción <span style="color:darkblue">Master of Puppets</span><br/>
14  Artista <span style="color:blue">Metallica</span><br/>
15  Disco <span style="color:lightblue">Master of Puppets</span><br/></p>
16 <p>
17  Canción <span style="color:darkblue">Among The Living</span><br/>
18  Artista <span style="color:blue">Anthrax</span><br/>
19  Disco <span style="color:lightblue">Among The Living</span><br/></p>
20 <p>
21  Canción <span style="color:darkblue">For Whom The Bell Tolls</span><br/>
22  Artista <span style="color:blue">Metallica</span><br/>
23  Disco <span style="color:lightblue">Ride The Lightning</span><br/></p>
24 </body>
25 </html>
26
```

Algunos ejemplos

- Utilizaremos un documento como el siguiente

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <MiBibliotecaMP3>
3   <archivo>
4     <canCIÓN>Hangar 18</canCIÓN>
5     <artista>Megadeth</artista>
6     <disco>Rust in Peace</disco>
7     <puntuacion>9</puntuacion>
8   </archivo>
9   <archivo>
10    <canCIÓN>Peace Sells</canCIÓN>
11    <artista>Megadeth</artista>
12    <disco>Peace Sells...But Who's Buying</disco>
13    <puntuacion>9</puntuacion>
14  </archivo>
15  <archivo>
16    <canCIÓN>Master of Puppets</canCIÓN>
17    <artista>Metallica</artista>
18    <disco>Master of Puppets</disco>
19    <puntuacion>10</puntuacion>
20  </archivo>
```

```
21 </archivo>
22   <canCIÓN>Among The Living</canCIÓN>
23   <artista>Anthrax</artista>
24   <disco>Among The Living</disco>
25   <puntuacion>8</puntuacion>
26 </archivo>
27 <archivo>
28   <canCIÓN>For Whom The Bell Tolls</canCIÓN>
29   <artista>Metallica</artista>
30   <disco>Ride The Lightning</disco>
31   <puntuacion>8</puntuacion>
32 </archivo>
33 </MiBibliotecaMP3>
```

Algunos ejemplos

- Ejemplo: plantilla principal y bucle

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0"
3   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4   <xsl:template match="/"><!-- plantilla principal -->
5     <html>
6     <body>
7       <h1>Mis Canciones</h1>
8       <table style="border: 1px solid blue;">
9         <tr style="background-color:lightblue;">
10           <th style="text-align:left">Título</th>
11           <th style="text-align:left">Artista</th>
12           <th style="text-align:left">Disco</th>
13           <th style="text-align:left">Puntuación</th>
14         </tr>
15         <xsl:for-each select="MiBibliotecaMP3/archivo"><!-- inicio bucle -->
16           <tr>
17             <td><xsl:value-of select="canción"/></td>
18             <td><xsl:value-of select="artista"/></td>
19             <td><xsl:value-of select="disco"/></td>
20             <td><xsl:value-of select="puntuación"/></td>
21           </tr>
22         </xsl:for-each><!-- final bucle -->
23       </table>
24     </body>
25   </html>
26 </xsl:template>
27 </xsl:stylesheet>
```

Mis Canciones

Titulo	Artista	Disco	Puntuación
Hangar 18	Megadeth	Rust in Peace	9
Peace Sells	Megadeth	Peace Sells...But Who's Buying	9
Master of Puppets	Metallica	Master of Puppets	10
Among The Living	Anthrax	Among The Living	8
For Whom The Bell Tolls	Metallica	Ride The Lightning	8

Algunos ejemplos

- Ejemplo: Ordenando por puntuación

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0"
3   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4   <xsl:template match="/"><!-- plantilla principal -->
5     <html>
6       <body>
7         <h1>Mis Canciones</h1>
8         <table style="border: 1px solid blue;"
9           <tr style="background-color:lightblue;"
10            <th style="text-align:left">Título</th>
11            <th style="text-align:left">Artista</th>
12            <th style="text-align:left">Disco</th>
13            <th style="text-align:left">Puntuación</th>
14          </tr>
15          <xsl:for-each select="MiBibliotecaMP3/archivo"><!-- inicio bucle -->
16            <xsl:sort select="puntuacion"/><!-- ordenado por puntuación -->
17            <tr>
18              <td><xsl:value-of select="canción"/></td>
19              <td><xsl:value-of select="artista"/></td>
20              <td><xsl:value-of select="disco"/></td>
21              <td><xsl:value-of select="puntuacion"/></td>
22            </tr>
23          </xsl:for-each><!-- final bucle -->
24        </table>
25      </body>
26    </html>
27  </xsl:template>
28 </xsl:stylesheet>
```

Mis Canciones

Título	Artista	Disco	Puntuación
Master of Puppets	Metallica	Master of Puppets	10
Among The Living	Anthrax	Among The Living	8
For Whom The Bell Tolls	Metallica	Ride The Lightning	8
Hangar 18	Megadeth	Rust in Peace	9
Peace Sells	Megadeth	Peace Sells...But Who's Buying	9

Error! Por defecto considera ordenación ascendente y tipo texto por lo que la ordenación no es todo lo correcta que nos gustaría....

Algunos ejemplos

- Ejemplo: Ordenando especificando tipo de datos

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0"
3 xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
4 <xsl:template match="/"> <!-- plantilla principal -->
5 <html>
6 <body>
7 |   <h1>Mis Canciones</h1>
8 |   <table style="border: 1px solid blue;">
9 |     <tr style="background-color:lightblue;">
10 |       <th style="text-align:left">Titulo</th>
11 |       <th style="text-align:left">Artista</th>
12 |       <th style="text-align:left">Disco</th>
13 |       <th style="text-align:left">Puntuación</th>
14 |     </tr>
15 |     <xsl:for-each select="MiBibliotecaMP3/archivo"> <!--
16 |       <xsl:sort select="puntuacion" data-type="number"/>
17 |       <tr>
18 |         <td><xsl:value-of select="canción"/></td>
19 |         <td><xsl:value-of select="artista"/></td>
20 |         <td><xsl:value-of select="disco"/></td>
21 |         <td><xsl:value-of select="puntuacion"/></td>
22 |       </tr>
23 |     </xsl:for-each> <!-- final bucle -->
24 |   </table>
25 | </body>
26 | </html>
27 </xsl:template>
28 </xsl:stylesheet>
```

Mis Canciones

Titulo	Artista	Disco	Puntuación
Among The Living	Anthrax	Among The Living	8
For Whom The Bell Tolls	Metallica	Ride The Lightning	8
Hangar 18	Megadeth	Rust in Peace	9
Peace Sells	Megadeth	Peace Sells...But Who's Buying	9
Master of Puppets	Metallica	Master of Puppets	10

Al indicar tipo de datos
numérico la ordenación
ahora sí es correcta

Algunos ejemplos

- Ejemplo: Mostramos solo los puntuados por encima de 9

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:template match="/"><!-- plantilla principal -->
4 <html>
5 <body>
6 |   <h1>Mis Canciones</h1>
7 |   <table style="border: 1px solid blue;">
8 |     <tr style="background-color: lightblue;">
9 |       <th style="text-align: left;">Título</th>
10 |      <th style="text-align: left;">Artista</th>
11 |      <th style="text-align: left;">Disco</th>
12 |      <th style="text-align: left;">Puntuación</th>
13 |    </tr>
14 |    <xsl:for-each select="MiBibliotecaMP3/archivo"><!-- inicio bucle -->
15 |      <xsl:if test="puntuacion > 9"><!-- solo muestra puntuados > 9 -->
16 |        <tr>
17 |          <td><xsl:value-of select="canción"/></td>
18 |          <td><xsl:value-of select="artista"/></td>
19 |          <td><xsl:value-of select="disco"/></td>
20 |          <td><xsl:value-of select="puntuacion"/></td>
21 |        </tr>
22 |      </xsl:if>
23 |    </xsl:for-each><!-- final bucle -->
24 |  </table>
25 | </body>
26 </html>
27 </xsl:template>
28 </xsl:stylesheet>
```

Mis Canciones

Título	Artista	Disco	Puntuación
Master of Puppets	Metallica	Master of Puppets	10

Algunos ejemplos

- Alternativa múltiple

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3 <xsl:template match="/"><!-- plantilla principal -->
4 <html>
5 <body>
6 <h1>Mis Canciones</h1>
7 <table style="border: 1px solid blue;">
8 <tr style="background-color:lightblue;">
9 <th style="text-align:left">Título</th>
10 <th style="text-align:left">Artista</th>
11 <th style="text-align:left">Disco</th>
12 <th style="text-align:left">Puntuación</th>
13 </tr>
14 <xsl:for-each select="MiBibliotecaMP3/archivo"><!-- inicio bucle -->
15 <xsl:choose>
16 <xsl:when test="puntuacion = 8"><!-- puntuados con 8, fondo amarillo -->
17 <tr>
18 <td style="background-color:yellow"><xsl:value-of select="canción"/></td>
19 <td style="background-color:yellow"><xsl:value-of select="artista"/></td>
20 <td style="background-color:yellow"><xsl:value-of select="disco"/></td>
21 <td style="background-color:yellow"><xsl:value-of select="puntuacion"/></td>
22 </tr>
23 </xsl:when>
```

```
24 <xsl:when test="puntuacion = 9"><!-- puntuados con 9, fondo naranja -->
25 <tr>
26 <td style="background-color:orange"><xsl:value-of select="canción"/></td>
27 <td style="background-color:orange"><xsl:value-of select="artista"/></td>
28 <td style="background-color:orange"><xsl:value-of select="disco"/></td>
29 <td style="background-color:orange"><xsl:value-of select="puntuacion"/></td>
30 </tr>
31 </xsl:when>
32 <xsl:otherwise><!-- el resto, fondo rojo -->
33 <tr>
34 <td style="background-color:red"><xsl:value-of select="canción"/></td>
35 <td style="background-color:red"><xsl:value-of select="artista"/></td>
36 <td style="background-color:red"><xsl:value-of select="disco"/></td>
37 <td style="background-color:red"><xsl:value-of select="puntuacion"/></td>
38 </tr>
39 </xsl:otherwise>
40 </xsl:choose>
41 </xsl:for-each><!-- final bucle -->
42 </table>
43 </body>
44 </html>
45 </xsl:template>
46 </xsl:stylesheet>
```

Mis Canciones

Título	Artista	Disco	Puntuación
Hangar 18	Megadeth	Rust in Peace	9
Peace Sells	Megadeth	Peace Sells...But Who's Buying	9
Master of Puppets	Metallica	Master of Puppets	10
Among The Living	Anthrax	Among The Living	8
For Whom The Bell Tolls	Metallica	Ride The Lightning	8

Referencias

- W3School - <https://www.w3schools.com>