

Actividad 7

Conversión De Un Documento Mediante Una Plantilla XSLT



Rubén Beltrán Muñoz

Lenguaje De Marcas

1ºDAM

Índice

- El primer nombre del cliente se debe imprimir en color rojo. 2
- En la tercera entrada del pedido no debe imprimirse el nombre, sino el texto siguiente: “###Este es el 3rd 'pedido'###”..... 3
- Todas las entradas 'PC' se presentarán en negrita. 3
- Utilizar el color amarillo para todos los dataset más allá de "proveedor" si el precio determinado dentro del elemento "precio" es 2000 o inferior..... 4
- Usar el color verde para todas las órdenes de cliente 'Reich'. 5
- Configurar el nombre del proveedor en cursiva si se pueden ordenar más de dos productos en ese lugar. 6

Una vez montado el código en mi caso por medio de SublimeText y su posterior comprobación en Internet Explorer (Funciona a la perfección tanto en Chrome, FireFox).

Procedo a la explicación de cada uno de los apartados

Cientes

Cliente	PC
Arm	pc400
Meier	pc500
###Este es el 3er pedido ###	pc600

Precio Menor 2000

Proveedor	Parte	Precio	Entrega
Vobis	pc400	1700	3
Dell	pc500	2000	6
Vobis	pc500	2000	3

Busqueda de Resultado Igual

Cientes	PC
Cesar	pc5145
Cesar	PC Cuantico

- El primer nombre del cliente se debe imprimir en color rojo.

En la siguiente parte del código declaramos una id a la línea correspondiente al nombre del primer cliente, que a su vez a sido declara en uno archivo .css

```
<tr id="Arm">
  <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[1]/Cliente"/></td>
  <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[1]/PC/text()"/></strong></td>
</tr>
```



```
1 *{
2   font-family: Helvetica, Arial, sans-serif;
3   font-size: 26px;
4 }
5
6 *{
7   background: #FFFFFF ;
8 }
9
10 #Arm{
11   color: red;
12 }
```



Cientes

Cliente	PC
Arm	pc400
Meier	pc500
###Este es el 3er pedido ###	pc600

Si quisiéramos ser más exactos respecto a las especificaciones de este punto y solo quisiéramos el nombre, establecemos en la primera columna de nuestra línea el Id="Arm" en vez del id="cliente" el cual no ha sido declarado en nuestro archivo .css

```
<tr>
  <td id="Arm"><xsl:value-of select= "/tablas/Orden/dataset[1]/Cliente"
  /></td>
  <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[1]/PC/text()"/></strong></td>
</tr>
```

Cientes

Cliente	PC
Arm	pc400
Meier	pc500
###Este es el 3er pedido ###	pc600

- En la tercera entrada del pedido no debe imprimirse el nombre, sino el texto siguiente: "###Este es el 3rd 'pedido'###".

En esta parte del código en el que estamos estableciendo las rutas a mostrar de nuestro código principal lo sustituiremos por el mensaje "###Este es el 3rd 'pedido'###".

➤ Mostrando el nombre

```
<tr id="Reich">
  <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[3]/Cliente"/></td>
  <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[3]/PC/text()"/></strong></td>
</tr>
```



Cientes

Cliente	PC
Arm	pc400
Meier	pc500
Reich	pc600

➤ Sustituyéndolo por el mensaje "###Este es el 3rd 'pedido'###"

```
<tr id="Reich">
  <td>###Este es el 3er pedido ###</td>
  <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[3]/PC/text()"/></strong></td>
</tr>
</table>
```



Cientes

Cliente	PC
Arm	pc400
Meier	pc500
###Este es el 3er pedido ###	pc600

- Todas las entradas 'PC' se presentarán en negrita.

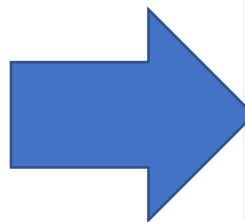
➤ Este punto lo realizaremos declarando la etiqueta en la línea correspondiente a la muestra de los datos de PC

```
<table border="1">
  <tr>
    <th> Cliente</th>
    <th> PC </th>
  </tr>

  <tr id="Arm">
    <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[1]/Cliente"/></td>
    <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[1]/PC/text()"/></strong></td>
  </tr>

  <tr id="Meier">
    <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[2]/Cliente"/></td>
    <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[2]/PC/text()"/></strong></td>
  </tr>

  <tr id="Reich">
    <td>###Este es el 3er pedido ###</td>
    <td id="cliente"><strong><xsl:value-of select = "/tablas/Orden/
dataset[3]/PC/text()"/></strong></td>
  </tr>
</table>
```



Cliente	PC
Arm	pc400
Meier	pc500
###Este es el 3er pedido ###	pc600



➤ Otra manera que tendríamos de realizarlos es establecerle un id a la línea que va mostrar el pc y establecerla en nuestro .css

```
<tr id="Arm">
  <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[1]/Cliente"/></td>
  <td id="pc"><xsl:value-of select = "/tablas/Orden/dataset[1]/PC/text()
"/></td>
</tr>

<tr id="Meier">
  <td id="cliente"><xsl:value-of select= "/tablas/Orden/
dataset[2]/Cliente"/></td>
  <td id="pc"><xsl:value-of select = "/tablas/Orden/dataset[2]/PC/text()
"/></td>
</tr>

<tr id="Reich">
  <td>###Este es el 3er pedido ###</td>
  <td id="pc"><xsl:value-of select = "/tablas/Orden/dataset[3]/PC/text()
"/></td>
```



```
background-color: #ffff00;
}

#Arm{
  color: red;
}

#pc{
  font-weight: bold;
}

.amarillo{
```

- Utilizar el color amarillo para todos los dataset más allá de "proveedor" si el precio determinado dentro del elemento "precio" es 2000 o inferior.

Para este punto vamos a establecer una tabla que nos muestre los datos de todos los proveedores con **precio menor de 2000**, mostrando en dicha tabla (Proveedor, Parte, Precio, Entrega)

En esta parte de nuestro **bucle for-each** declaramos que solo nos muestre mientras los recorre los que sean menor de 2000.

```
<h2> Precio Menor 2000 </h2>
<table border="1">
  <tr>
    <th> Proveedor </th>
    <th> Parte</th>
    <th> Precio </th>
    <th> Entrega </th>
  </tr>
  <xsl:for-each select="/tablas/Devoluciones/dataset[Precio < 2001]">
    <tr>
      <td class="amarillo"><xsl:value-of select = "Proveedor"/></td>
      <td class="amarillo"><xsl:value-of select = "Parte"/></td>
      <td class="amarillo"><xsl:value-of select = "Precio"/></td>
      <td class="amarillo"><xsl:value-of select = "Entrega"/></td>
    </tr>
  </xsl:for-each>
</table>
```

Además, le creamos una clase y las establecemos en nuestro .css, para los datos que nos los muestre en color amarillo

```
<h2> Precio Menor 2000 </h2>
<table border="1">
  <tr>
    <th> Proveedor </th>
    <th> Parte</th>
    <th> Precio </th>
    <th> Entrega </th>
  </tr>
  <xsl:for-each select="/tablas/Devoluciones/dataset[Precio < 2001]">
    <tr>
      <td class="amarillo"><xsl:value-of select = "Proveedor"/></td>
      <td class="amarillo"><xsl:value-of select = "Parte"/></td>
      <td class="amarillo"><xsl:value-of select = "Precio"/></td>
      <td class="amarillo"><xsl:value-of select = "Entrega"/></td>
    </tr>
  </xsl:for-each>
</table>
```

```
10 #Arm{
11   color: red;
12 }
13 .amarillo{
14   color: yellow;
15 }
16
17 .green{
18   color: green;
19 }
```

Precio Menor 2000

Proveedor	Parte	Precio	Entrega
Vobis	pc400	1700	3
Dell	pc500	2000	6
Vobis	pc500	2000	3

- Usar el color verde para todas las órdenes de cliente 'Reich'.

En esta parte realizaremos un **bucle for-each** que recorrerá nuestra dataset mostrándonos en verde todos los clientes que se llamen **“Reich”**

```
<h2> Busqueda de Resultado Igual </h2>
<table border="1">
  <tr>
    <th> Clientes </th>
    <th> PC </th>
  </tr>

  <xsl:for-each select="//Orden/dataset [Cliente = 'Reich']">
    <tr>
      <td class="green"><xsl:value-of select = "Cliente"/></td>
      <td class="green"><xsl:value-of select = "PC"/></td>
    </tr>
  </xsl:for-each>
</table>
```

Busqueda de Resultado Igual

Clientes	PC
Reich	pc600

Realizamos la prueba, pero esta vez introducimos dos clientes llamados **“Cesar”** en nuestro documento xml_pedidos.xml

```
</dataset>
<dataset>
  <Cliente>Cesar</Cliente>
  <PC>pc5145</PC>
</dataset>
<dataset>
  <Cliente>Cesar</Cliente>
  <PC>PC Cuantico</PC>
</dataset>
</Orden>
```

```
<h2> Busqueda de Resultado Igual </h2>
<table border="1">
  <tr>
    <th> Clientes </th>
    <th> PC </th>
  </tr>

  <xsl:for-each select="//Orden/dataset [Cliente = 'Cesar']">
    <tr>
      <td class="green"><xsl:value-of select = "Cliente"/></td>
      <td class="green"><xsl:value-of select = "PC"/></td>
    </tr>
  </xsl:for-each>
</table>
```

Busqueda de Resultado Igual

Clientes	PC
Cesar	pc5145
Cesar	PC Cuantico

- Configurar el nombre del proveedor en cursiva si se pueden ordenar más de dos productos en ese lugar.

Lo único que he podido realizar de este punto es realizar un **bucle for-each** y colocarlos por nombres con la instrucción **xsl:sort**, pero no he podido realizar comparaciones entre ellas. Se prueba con condiciones **xsl:if** pero no es posible. Se realiza también una búsqueda en internet sin encontrar la forma de la misma.

```
<h2> Tabla ordenada </h2>
<table border="1">
  <tr>
    <th> Proveedor </th>
    <th> Parte</th>
    <th> Precio </th>
    <th> Entrega </th>
  </tr>
  <xsl:for-each select="tablas/Devoluciones/dataset">
    <xsl:sort select="Proveedor"/>
    <tr>
      <td><xsl:value-of select = "Proveedor"/></td>
      <td><xsl:value-of select = "Parte"/></td>
      <td><xsl:value-of select = "Precio"/></td>
      <td><xsl:value-of select = "Entrega"/></td>
    </tr>
  </xsl:for-each>
</table>
```

Tabla ordenada

Proveedor	Parte	Precio	Entrega
Dell	pc500	2000	6
IBM	pc500	2500	6
IBM	pc600	3500	4
Vobis	pc400	1700	3
Vobis	pc500	2000	3
Vobis	pc600	2500	3