

## Boletín 2 XQuery

Dado el siguiente documento XML, *bib.xml*,

```
<?xml version="1.0" encoding="UTF-8"?>
<bib>
  <libro anyo="1994">
    <titulo>TCP/IP Illustrated</titulo>
    <autor>
      <apellido>Stevens</apellido>
      <nombre>W.</nombre>
    </autor>
    <editorial>Addison-Wesley</editorial>
    <precio>65.95</precio>
  </libro>
  <libro anyo="1992">
    <titulo>Advanced Programming for Unix environment</titulo>
    <autor>
      <apellido>Stevens</apellido>
      <nombre>W.</nombre>
    </autor>
    <editorial>Addison-Wesley</editorial>
    <precio>65.95</precio>
  </libro>
  <libro anyo="2000">
    <titulo>Data on the Web</titulo>
    <autor>
      <apellido>Abiteboul</apellido>
      <nombre>Serge</nombre>
    </autor>
    <autor>
      <apellido>Buneman</apellido>
      <nombre>Peter</nombre>
    </autor>
    <autor>
      <apellido>Suciu</apellido>
      <nombre>Dan</nombre>
    </autor>
    <editorial>Morgan Kaufmann editorials</editorial>
    <precio>39.95</precio>
  </libro>
  <libro anyo="1999">
    <titulo>Economics of Technology for Digital TV</titulo>
    <editor>
      <apellido>Gerbarg</apellido>
      <nombre>Darcy</nombre>
      <afiliacion>CITI</afiliacion>
    </editor>
    <editorial>Kluwer Academic editorials</editorial>
    <precio>129.95</precio>
  </libro>
</bib>
```

Realiza las siguientes consultas con XQuery (consulta las funciones XPath en [https://www.w3schools.com/xml/xsl\\_functions.asp](https://www.w3schools.com/xml/xsl_functions.asp)):

1. Listar el título de todos los libros.

Resultado:

```
<titulo>TCP/IP Illustrated</titulo>
<titulo>Advanced Programming for Unix environment</titulo>
<titulo>Data on the Web</titulo>
<titulo>Economics of Technology for Digital TV</titulo>
```

2. Listar año y título de todos los libros, ordenados por el año ascendentemente. Después, ordenar la información descendentemente

Resultado:

```
Advanced Programming for Unix environment-->1992
TCP/IP Illustrated-->1994
Economics of Technology for Digital TV-->1999
Data on the Web-->2000
```

3. Listar los libros cuyo precio sea 65.95.

Resultado:

```
<titulo>TCP/IP Illustrated</titulo>
<titulo>Advanced Programming for Unix environment</titulo>
```

4. Listar los libros publicados antes del año 2000 (no incluido).

Resultado:

```
<titulo>TCP/IP Illustrated</titulo>
<titulo>Advanced Programming for Unix environment</titulo>
<titulo>Economics of Technology for Digital TV</titulo>
```

5. Listar año y título de los libros publicados por Addison-Wesley después del año 1992 (no incluido).

Resultado:

```
<titulo>TCP/IP Illustrated</titulo>
```

6. Listar año y título de los libros que tienen más de un autor.

Resultado:

```
Data on the Web-2000
```

7. Listar año y título de los libros que tienen o no tienen autor. Dar una versión con la función EXISTS y otra con la función EMPTY.

Resultado:

```
Economics of Technology for Digital TV-1999
```

8. Mostrar los apellidos de los autores que aparecen en el documento, sin repeticiones, ordenados alfabéticamente.

Resultado:

```
Abiteboul  
Buneman  
Stevens  
Suciu
```

9. Por cada libro, listar agrupado en un elemento <result> su título y autores.

Resultado:

```
<result>TCP/IP Illustrated</result>  
<result>Advanced Programming for Unix environment</result>  
<result>Data on the Web</result>  
<result>Economics of Technology for Digital TV</result>
```

10. Por cada libro, obtener su título y el número de autores, agrupados en un elemento <libro-autores>.

Resultado:

```
<libro-autores>TCP/IP Illustrated(1)</libro-autores>  
<libro-autores>Advanced Programming for Unix environment(1)</libro-autores>  
<libro-autores>Data on the Web(3)</libro-autores>  
<libro-autores>Economics of Technology for Digital TV(0)</libro-autores>
```

11. Generar un documento html con un encabezado que diga "Listado de libros" seguido de una tabla de 3 columnas, en la que se muestre el título, editorial y precio de cada libro añadiendo el símbolo del € después de la cantidad. La tabla deberá tener bordes y la primera fila tendrá un formato especial con los nombres de las columnas.

Resultado:

## Listado de libros

Título	Nombre autor	Apellido autor	Precio
TCP/IP Illustrated	W.	Stevens	65.95 €
Advanced Programming for Unix environment	W.	Stevens	65.95 €
Data on the Web	Serge Peter Dan	Abiteboul Buneman Suciu	39.95 €
Economics of Technology for Digital TV			129.95 €

12. Generar el siguiente documento html con la información de los libros del autor Stevens. El valor precio total es un campo calculado utilizando una consulta con una cláusula let.

Resultado:

<b>Listado de libros</b>	
<b>Libros del autor Stevens</b>	
TCP/IP Illustrated	65.95
Advanced Programming for Unix environment	65.95
Total:	131.9
<b>Libros del autor Abiteboul</b>	
Data on the Web	39.95
Total:	39.95
<b>Libros del autor Buneman</b>	
Data on the Web	39.95
Total:	39.95
<b>Libros del autor Suciu</b>	
Data on the Web	39.95
Total:	39.95

## SOLUCIONES

Todas las soluciones suponen que el fichero *bib.xml* está como contexto.

1.

Solución:

```
for $libro in /bib/libro
return $libro/titulo
```

```
for $titulo in /bib/libro/titulo
return $titulo
```

2.

Solución:

La ordenación ascendente es la de por defecto, por lo que no hace falta indicar nada. Si se quisiera indicar se indicaría ascending.

```
for $libro in /bib/libro
order by $libro/@anyo
return string-join(($libro/titulo,$libro/@anyo), '-->')
```

Para ordenarlo descendentemente:

```
for $libro in /bib/libro
order by $libro/@anyo descending
return string-join(($libro/titulo,$libro/@anyo), '-->')
```

3.

Solución:

```
for $libro in /bib/libro
where $libro/precio=65.95
return $libro/titulo
```

```
for $libro in /bib/libro[precio=65.95]
return $libro/titulo
```

4.

Solución:

```
for $libro in /bib/libro[@anyo<2000]
return $libro/titulo
```

```
for $libro in /bib/libro
where $libro/@anyo<2000
return $libro/titulo
```

5.

Solución:

```
for $libro in /bib/libro
```

```
where $libro/editorial="Addison-Wesley" and $libro/@anyo>1992
return $libro/titulo
```

```
for $libro in /bib/libro[editorial="Addison-Wesley"]
where $libro/@anyo>1992
return $libro/titulo
```

```
for $libro in /bib/libro[@anyo>1992]
where $libro/editorial="Addison-Wesley"
return $libro/titulo
```

6.

Solución:

```
for $libro in /bib/libro[count(autor)>2]
return string-join(($libro/titulo, $libro/@anyo),'-')
```

```
for $libro in /bib/libro
where $libro/count(autor)>2
return string-join(($libro/titulo, $libro/@anyo),'-')
```

7.

Solución:

```
for $libro in /bib/libro
where $libro/count(autor)=0
return string-join(($libro/titulo, $libro/@anyo),'-')
```

```
for $libro in /bib/libro
where $libro/not(exists(autor))
return string-join(($libro/titulo, $libro/@anyo),'-')
```

```
for $libro in /bib/libro
where $libro/empty(autor)
return string-join(($libro/titulo, $libro/@anyo),'-')
```

8.

Solución:

```
for $apellido in distinct-values(/bib/libro/autor/apellido)
order by $apellido
return $apellido
```

9.

Solución:

Cuando tengo que devolver una etiqueta HTML, necesito la función data().

```
for $libro in /bib/libro
return <result>
  {data($libro/titulo)},
  {data($libro/autor/apellido), data($libro/autor/nombre)}
</result>
```

10.

Solución:

```
for $libro in /bib/libro
```

```
return
```

```
<libro-autores>{data($libro/titulo)}({data($libro/count(autor))})</libro-autores>
```

11.

Solución:

```
<html>
```

```
<title>
```

```
Listado de libros
```

```
</title>
```

```
<body>
```

```
<h1>Listado de libros</h1>
```

```
<table border="1">
```

```
<tr>
```

```
<th>Título</th>
```

```
<th>Nombre autor</th>
```

```
<th>Apellido autor</th>
```

```
<th>Precio</th>
```

```
</tr>
```

```
{
```

```
for $x in /bib/libro
```

```
return
```

```
<tr>
```

```
<td>{data($x/titulo)}</td>
```

```
<td>{data($x/autor/nombre)}</td>
```

```
<td>{data($x/autor/apellido)}</td>
```

```
<td>{concat(data($x/precio),' €')}</td>
```

```
</tr>
```

```
}
```

```
</table>
```

```
</body>
```

```
</html>
```

12.

Solución:

```
<html>
```

```
<title>
```

```
Listado de libros
```

```
</title>
```

```
<body>
```

```
<h1>Listado de libros</h1>
```

```
{
```

```
for $x in /bib/distinct-values(libro/autor/apellido)
```

```
return
```

```
<p>
```

```
<h1>Libros del autor {data($x)}</h1>
```

```
<table border="1px solid black">
```

```

    {
      for $y in /bib/libro/autor[apellido=$x]
      return <tr>
        <td>{data($y/../../titulo)}</td>
        <td>{data($y/../../precio)}</td>
      </tr>
    }
    {
      let $y := sum(/bib/libro/autor[apellido=$x]/../../precio)
      return <tr>
        <td>Total:</td>
        <td>{data($y)}</td>
      </tr>
    }
  </table>
</p>
}
</body>
</html>

```