

## 6.5.- Ejemplos XQuery.

Vamos a ver algunos ejemplos de XQuery utilizando el fichero [libros.xml](#). En este fichero:

- ✓ El elemento raíz es **biblioteca**. Contiene un elemento **libros**.
- ✓ Dentro de **libros**, hay varios elementos **libro**.
- ✓ Los elementos **libro** tienen atributos **publicacion** y **edicion** (opcional). También tienen elementos **titulo**, **autor** (puede haber más de uno), **editorial**, **paginas** y un elemento opcional para indicar si hay edición electrónica, **edicionElectronica**.

### Ejercicio Resuelto 1

1.- Título y editorial de todos los libros Para devolver varios campos, los envolvemos en un elemento.

Mostrar retroalimentación

```
1 for $x in doc("libros.xml")/biblioteca/libros/libro
2 return <libro>{$x/titulo, $x/editorial}</libro>
```

```
1 <libro>
2   <titulo>Learning XML</titulo>
3   <editorial>O'Reilly</editorial>
4 </libro>
5 <libro>
6   <titulo>XML Imprescindible</titulo>
7   <editorial>O'Reilly</editorial>
8 </libro>
9 <libro>
10  <titulo>XML Schema</titulo>
11  <editorial>O'Reilly</editorial>
12 </libro>
13 <libro>
14  <titulo>XPath Essentials</titulo>
15  <editorial>Wiley</editorial>
16 </libro>
17 <libro>
18  <titulo>Beginning XSLT 2.0: From Novice to Professional</titulo>
19  <editorial>Apress</editorial>
20
```

```
</libro>
<libro>
  <titulo>XQuery</titulo>
  <editorial>O'Reilly</editorial>
</libro>
```

## Ejercicio Resuelto 2

2.- El título (sin etiquetas) de todos los libros de menos de 100 páginas. Para hacer comparaciones con números, lo mejor es convertir los datos con la función **number** para evitar problemas de tipo de dato o que los compare como cadenas.

Mostrar retroalimentación

```
1 | for $x in doc("libros.xml")/biblioteca/libros/libro
2 | where number($x/paginas) < 100
3 | return data($x/titulo)
```

```
1 | Learning XML
```

## Ejercicio Resuelto 3

3.- El número de libros de menos de 100 páginas. Utilizamos la función **count()**.

Mostrar retroalimentación

```
1 | for $x in doc("libros.xml")/biblioteca/libros
2 | let $y := $x/libro[number(paginas) < 100]
3 | return count($y)
```

1

## Ejercicio Resuelto 4

4.- Una lista HTML con el título de los libros de la editorial "O'Reilly" ordenados por título. Podemos mezclar etiquetas HTML y XQuery y obtener HTML como resultado de una consulta.

Mostrar retroalimentación

```
1 | <ul>
2 | {
3 |   for $x in doc("libros.xml")/biblioteca/libros/libro
4 |   where $x/editorial = "O'Reilly"
5 |   order by $x/titulo
6 |   return <li>{data($x/titulo)}</li>
7 | }
8 | </ul>
```

```
1 | <ul>
2 |   <li>Learning XML</li>
3 |   <li>XML Imprescindible</li>
4 |   <li>XML Schema</li>
5 |   <li>XQuery</li>
6 | </ul>
```

## Ejercicio Resuelto 5

5.- Título y editorial de los libros de 2002.

Mostrar retroalimentación

```
1 | for $x in doc("libros.xml")/biblioteca/libros/libro
2 | where $x[@publicacion=2002]
3 | return <libro>{$x/titulo, $x/editorial}</libro>
```

```
1 | <libro>
2 |   <titulo>XML Schema</titulo>
3 |   <editorial>O'Reilly</editorial>
4 | </libro>
5 | <libro>
6 |   <titulo>XPath Essentials</titulo>
7 |   <editorial>Wiley</editorial>
8 | </libro>
```

## Ejercicio Resuelto 6

6.- Título y editorial de los libros con más de un autor.

Mostrar retroalimentación

```
1 | for $x in doc("libros.xml")/biblioteca/libros/libro
2 | where count($x/autor)>1
3 | return <libro>{$x/titulo, $x/editorial}</libro>
```

```
1 <libro>
2   <titulo>XML Imprescindible</titulo>
3   <editorial>O'Reilly</editorial>
4 </libro>
```

## Ejercicio Resuelto 7

7.- Título y editorial de los libros que tienen versión electrónica.

Mostrar retroalimentación

```
1 for $x in doc("libros.xml")/biblioteca/libros/libro
2 where $x/versionElectronica
3 return <libro>{$x/titulo, $x/editorial}</libro>
```

```
1 <libro>
2   <titulo>Learning XML</titulo>
3   <editorial>O'Reilly</editorial>
4 </libro>
5 <libro>
6   <titulo>XPath Essentials</titulo>
7   <editorial>Wiley</editorial>
8 </libro>
```

## Ejercicio Resuelto 8

8.- Título de los libros que no tienen versión electrónica.

## Mostrar retroalimentación

```
1 | for $x in doc("libros.xml")/biblioteca/libros/libro
2 | where not($x/versionElectronica)
3 | return$x/titulo
```

```
1 | <titulo>XML Imprescindible</titulo>
2 | <titulo>XML Schema</titulo>
3 | <titulo>Beginning XSLT 2.0: Form Novice to Professional</titulo>
4 | <titulo>XQuery</titulo>
```