

RELACIÓN 3

CHRISTIAN MILLÁN SORIA

1. Dado el siguiente documento XML, realiza las siguientes consultas con XQUERY:

```
<?xml version="1.0" encoding="UTF-8"?>
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday Italian</title>
    <author>Giada De Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>

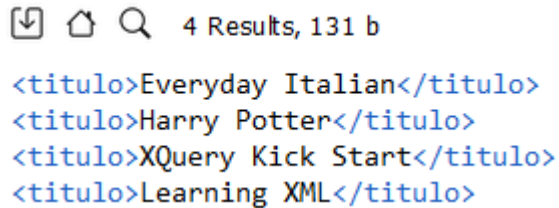
  <book category="CHILDREN">
    <title lang="en">Harry Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>

  <book category="WEB">
    <title lang="en">XQuery Kick Start</title>
    <author>James McGovern</author>
    <author>Per Bothner</author>
    <author>Kurt Cagle</author>
    <author>James Linn</author>
    <author>Vaidyanathan Nagarajan</author>
    <year>2003</year>
    <price>49.99</price>
  </book>

  <book category="WEB">
    <title lang="en">Learning XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

a. Mostrar los títulos de los libros con la etiqueta "titulo".

```
for $titulo in /bookstore/book/title/text()
return <titulo>{$titulo}</titulo>
```



```

<titulo>Everyday Italian</titulo>
<titulo>Harry Potter</titulo>
<titulo>XQuery Kick Start</titulo>
<titulo>Learning XML</titulo>

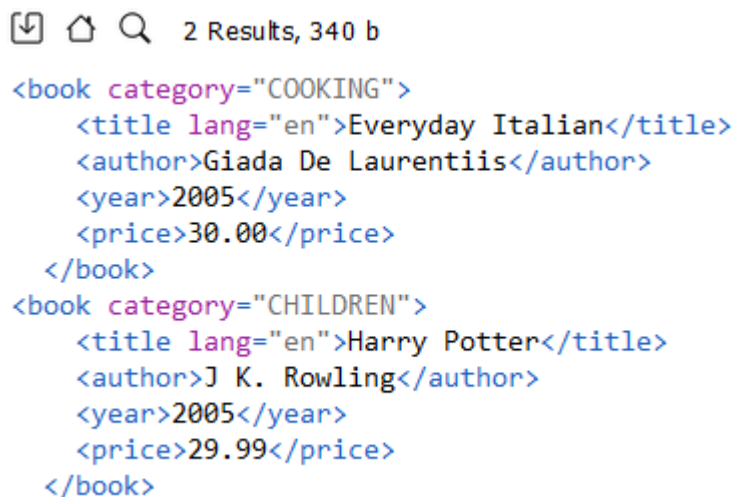
```

b. Mostrar los libros cuyo precio sea menor o igual a 30. Primero incluyendo la condición en la cláusula "where" y luego en la ruta del XPATH.

```

for $libro in /bookstore/book
where $libro/price<=30
return $libro

```



```

<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>
<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

```

```

for $libro in /bookstore/book[price<=30]
return $libro

```



```

<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>
<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>

```

c. Mostrar sólo el título de los libros cuyo precio sea menor o igual a 30.

```
for $book in //book[price<=30]
return $book/title
```



2 Results, 80 b

```
<title lang="en">Everyday Italian</title>
<title lang="en">Harry Potter</title>
```

d. Mostrar sólo el título sin atributos de los libros cuyo precio sea menor o igual a 30.

```
for $book in //book[price<=30]
return <title>{data($book/title)}</title>
```



2 Results, 60 b

```
<title>Everyday Italian</title>
<title>Harry Potter</title>
```

e. Mostrar el título y el autor de los libros del año 2005 y etiquetar cada uno de ellos con "lib2005".

```
for $book in //book[year=2005]
return <lib2005>
      {$book/title}
      {$book/author}
    </lib2005>
```






2 Results, 183 b

```
<lib2005><title lang="en">Everyday Italian</title><author>Giada De Laurentiis</author></lib2005>
<lib2005><title lang="en">Harry Potter</title><author>J K. Rowling</author></lib2005>
```




f. Mostrar los años de publicación, primero con "for" y luego con "let" para comprobar la diferencia entre ellos. Etiquetar la salida con "publicacion".

```
<publicacion>
{
  for $book in //book
  return <year>{$book/year}</year>
}
</publicacion>
```

   1 Result, 147 b

```
<publicacion><year><year>2005</year></year><year><year>2005</year></year><year><year>2003</year></year><year><year>2003</year></year></publicacion>
```

```
<publicacion>
{
  let $years:=//book/year
  for $year in $years
  return <year>{$year}</year>
}
</publicacion>
```

   1 Result, 147 b

```
<publicacion><year><year>2005</year></year><year><year>2005</year></year><year><year>2003</year></year><year><year>2003</year></year></publicacion>
```

g. Mostrar los libros ordenados primero por "category" y luego por "title" en una sola consulta.

```
for $book in bookstore/book
order by $book/@category, $book/title
return $book
```

4 Results, 813 b

```
<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>
<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>
<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>
```

h. Mostrar cuántos libros hay y etiquetarlo con "total".

```
<total>{count(//book)}</total>
```

1 Result, 16 b

```
<total>4</total>
```

i. Mostrar los títulos de los libros y al final una etiqueta con el número total de libros.

```
<resultado>
{
  for $book in //book
  return <titulo>{$book/title/text()}</titulo>
}
<total_libros>{count(//book)}</total_libros>
</resultado>
```

```

1 Result, 178 b
<resultado><titulo>Everyday Italian</titulo><titulo>Harry Potter</titulo><titulo>XQuery Kick Start</titulo><titulo>Learning XML</titulo><total_libros>4</total_libros></resultado>

```

j. Mostrar el precio mínimo y máximo de los libros.

```

<resultado>
  <max>{max(//price)}</max>
  <min>{min(//price)}</min>
</resultado>

```

```

1 Result, 55 b
<resultado><max>49.99</max><min>29.99</min></resultado>

```

k. Mostrar el título del libro, su precio y su precio con el IVA incluido, cada uno con su propia etiqueta. Ordénalos por precio con IVA.

```

<libros>
{
  for $book in //book
  let $price:=$book/price
  let $price_iva:=$price*1.21
  order by $price_iva ascending
  return <libro>
    <titulo>{$book/title/text()}</titulo>
    <precio>{$price} €</precio>
    <precio_iva>{$price_iva} €</precio_iva>
  </libro>
}
</libros>

```

```

1 Result, 507 b
<libros><libro><titulo>Harry Potter</titulo><precio><price>29.99</price> €</precio><precio_iva>36.2879 €</precio_iva></libro><libro><titulo>Everyday Italian</titulo><precio><price>30.00</price> €</precio><precio_iva>36.3 €</precio_iva></libro><libro><titulo>Learning XML</titulo><precio><price>39.95</price> €</precio><precio_iva>48.3395 €</precio_iva></libro><libro><titulo>XQuery Kick Start</titulo><precio><price>49.99</price> €</precio><precio_iva>60.4879 €</precio_iva></libro></libros>

```

l. Mostrar la suma total de los precios de los libros con la etiqueta "total".

```

<total>{sum(/bookstore/book/price)}</total>

```

```

1 Result, 21 b
<total>149.93</total>

```

m. Mostrar cada uno de los precios de los libros, y al final una nueva etiqueta con la suma de los precios.

```
<libros>
{
  for $p in //book/price
  return <price>{string($p)}</price>
}
<total>{sum(//book/price)}</total>
</libros>
```

1 Result, 118 b

```
<libros><price>30.00</price><price>29.99</price><price>49.99</price><price>39.95</price><total>149.93</total></libros>
```

n. Mostrar el título y el número de autores que tiene cada título en etiquetas diferentes.

```
for $book in //book
return
  <libro>
    <title>{$book/title}</title>
    <autores>{count($book/author)}</autores>
  </libro>
```

4 Results, 363 b

```
<libro><title><title lang="en">Everyday Italian</title></title><autores>1</autores></libro>
<libro><title><title lang="en">Harry Potter</title></title><autores>1</autores></libro>
<libro><title><title lang="en">XQuery Kick Start</title></title><autores>5</autores></libro>
<libro><title><title lang="en">Learning XML</title></title><autores>1</autores></libro>
```

ñ. Mostrar en la misma etiqueta el título y entre paréntesis el número de autores que tiene ese título.

```
for $book in //book
return <libro>{$book/title/text()}({count($book/author)})</libro>
```

4 Results, 135 b

```
<libro>Everyday Italian(1)</libro>
<libro>Harry Potter(1)</libro>
<libro>XQuery Kick Start(5)</libro>
<libro>Learning XML(1)</libro>
```

o. Mostrar los libros escritos en años que terminen en "3".

```
for $book in //book
where substring($book/year, string-length($book/year))='3'
return $book
```

📄 🏠 🔍 2 Results, 471 b

```
<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>
<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>
```

p. Mostrar los libros cuya categoría empiece por "C".

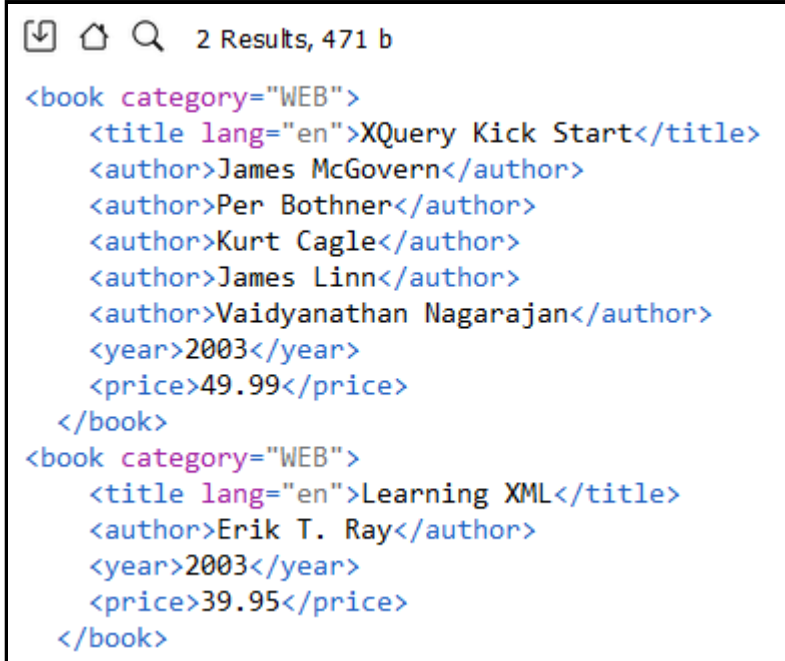
```
for $book in //book[starts-with(@category, 'C')]
return $book
```

📄 🏠 🔍 2 Results, 340 b

```
<book category="COOKING">
  <title lang="en">Everyday Italian</title>
  <author>Giada De Laurentiis</author>
  <year>2005</year>
  <price>30.00</price>
</book>
<book category="CHILDREN">
  <title lang="en">Harry Potter</title>
  <author>J K. Rowling</author>
  <year>2005</year>
  <price>29.99</price>
</book>
```

q. Mostrar los libros que tengan una "X" mayúscula o minúscula en el título ordenados de manera descendente.

```
for $book in bookstore/book[contains(title, 'X') or contains(title, 'x')]
order by $book/title descending
return $book
```

```

<book category="WEB">
  <title lang="en">XQuery Kick Start</title>
  <author>James McGovern</author>
  <author>Per Bothner</author>
  <author>Kurt Cagle</author>
  <author>James Linn</author>
  <author>Vaidyanathan Nagarajan</author>
  <year>2003</year>
  <price>49.99</price>
</book>
<book category="WEB">
  <title lang="en">Learning XML</title>
  <author>Erik T. Ray</author>
  <year>2003</year>
  <price>39.95</price>
</book>

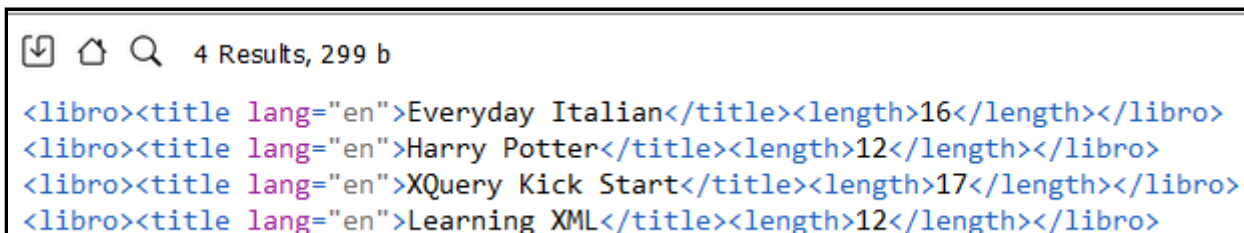
```

r. Mostrar el título y el número de caracteres que tiene cada título, cada uno con su propia etiqueta.

```

for $book in //book
return
<libro>
  <title lang="en">{data($book/title)}</title>
  <length>{string-length(data($book/title))}</length>
</libro>

```




```

<libro><title lang="en">Everyday Italian</title><length>16</length></libro>
<libro><title lang="en">Harry Potter</title><length>12</length></libro>
<libro><title lang="en">XQuery Kick Start</title><length>17</length></libro>
<libro><title lang="en">Learning XML</title><length>12</length></libro>

```

s. Mostrar todos los años en los que se ha publicado un libro eliminando los repetidos. Etiquétalos con "año".

```
distinct-values(//book/year)!<año>{.}</año>
```



```

<año>2005</año>
<año>2003</año>

```

t. Mostrar todos los autores eliminando los que se repiten y ordenados por el número de caracteres que tiene cada autor.

```
for $author in distinct-values(//book/author)
order by string-length($author)
return <autor>{$author}</autor>
```

8 Results, 243 b

```
<autor>Kurt Cagle</autor>
<autor>James Linn</autor>
<autor>Per Bothner</autor>
<autor>Erik T. Ray</autor>
<autor>J K. Rowling</autor>
<autor>James McGovern</autor>
<autor>Giada De Laurentiis</autor>
<autor>Vaidyanathan Nagarajan</autor>
```

u. Mostrar los títulos en una tabla de HTML.

```
<table>
{
for $book in //book
return <tr><td>{data($book/title)}</td></tr>
}
</table>
```

1 Result, 144 b

```
<table><tr><td>Everyday Italian</td></tr><tr><td>Harry Potter</td></tr><tr><td>XQuery Kick Start</td></tr><tr><td>Learning XML</td></tr></table>
```