

TEMA 4

Alex Pérez

LENGUAJE DE MARCAS 1º daw

1. Cuenta el número de elementos del siguiente código XML:

```
<item>
  <title>Agregadores</title>
  <link>http://www.librolm.com/html</link>
  <description>Apuntes agregadores</description>
  <guid>B4BA124D-D535-4830-B28C-E0F0BD49D80E</guid>
  <pubDate>Tue, 26 Dec 2017 17:02:39 GMT</pubDate>
</item>
```

Solución 

Elemento raíz/padre: item

Elementos hijos: Title, link, description, guid, pubDate.

En este código hay 6 elementos.

2. Contesta a las siguientes preguntas:

- a) ¿Está bien formado el siguiente XML teniendo en cuenta su esquema?
- b) ¿Es válido el siguiente XML teniendo en cuenta su esquema?

XML:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<bdproductos xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="zax.xsd">
  <producto>
    <nombre>teclado</nombre>
```

[.../...]

```
    <código>C001</código>
    <iva>21</iva>
    <categoria>A</categoria>
    <precio>15</precio>
    <descuento>10</descuento>
  </producto>
  <producto>
    <nombre>monitor</nombre>
    <código>C002</código>
    <categoria>B</categoria>
    <iva>21</iva>
    <precio>147.2</precio>
    <descuento>9</descuento>
  </producto>
</bdproductos>
```

Esquema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDe-
  fault="qualified">
  <xs:element name="bdproductos">
    <xs:complexType>
      <xs:sequence>
        <xs:element maxOccurs="unbounded" ref="producto"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="producto">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="nombre"/>
        <xs:element ref="código"/>
        <xs:element ref="categoría"/>
        <xs:element ref="iva"/>
        <xs:element ref="precio"/>
        <xs:element ref="descuento"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="nombre" type="xs:NCName"/>
  <xs:element name="código" type="Tlongitudentre"/>
  <xs:element name="categoría" type="Tcat1"/>
  <xs:element name="iva" type="xs:integer"/>
  <xs:element name="precio" type="Tdecimal"/>
  <xs:element name="descuento" type="Tdescuento"/>
  <xs:simpleType name="Tdescuento">
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1"/>
      <xs:maxInclusive value="10"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:simpleType name="Tdecimal">
```

[.../...]

```

        <xs:restriction base="xs:decimal">
            <xs:totalDigits value="4"/>
            <xs:fractionDigits value="2"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Tcat">
        <xs:restriction base="xs:string">
            <xs:enumeration value="A"/>
            <xs:enumeration value="B"/>
            <xs:enumeration value="C"/>
            <xs:enumeration value="D"/>
            <xs:enumeration value="E"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Tcat1">
        <xs:restriction base="xs:string">
            <xs:pattern value="[A-E]"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Tlongitud">
        <xs:restriction base="xs:string">
            <xs:length value="4"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="Tlongitudentre">
        <xs:restriction base="xs:string">
            <xs:minLength value="1"/>
            <xs:maxLength value="4"/>
        </xs:restriction>
    </xs:simpleType>
</xs:schema>

```

En principio no se encuentra ningún error en el código, elemento `<xs:element maxOccurs="unbounded" ref="producto"/>` está correcto y permite diversos elementos de `<producto>`

La referencia de elementos “ref” se refiere a elementos como nombre, código, categoría, iva etc.

Los tipos personalizados como TDECIMAL restringe a 4 dígitos con dos decimales, lo que es correcto. En cuanto a TDESCUENTO restringe los valores entre 1 y 10, el tipo Tcat y Tcat1 definen las categorías con restricciones.

Las restricciones en Tlongitud y Tlongitudentre, son válidas y están bien definidas.

En conclusión, el Código tiene una declaración XML en la parte superior, cada elemento de apertura tiene su elemento de cierre.

¿Es válido el siguiente XML teniendo en cuenta su esquema?

Está bien escrito según lo que he visto de cada atributo y cada declaración y las etiquetas de apertura y cierre.

3. Dado el siguiente documento XML, donde se almacenan datos relacionados de una librería con libros de informática, crea el documento DTD asociado.

XML:

```
<?xml version="1.0" encoding="iso-8859-1"?>
<librería>
  <libro>
    <título>Lenguaje de marcas</título>
    <autor>Autor2</autor>
    <autor>Autor1</autor>
    <ISBN></ISBN>
    <páginas>158</páginas>
    <precio>15</precio>
  </libro>
  <libro>
    <título>XML</título>
    <autor>Autor1</autor>
    <ISBN></ISBN>
    <páginas>278</páginas>
    <precio>30</precio>
  </libro>
</librería>
```

¡El DTD se guarda en un archivo separado, y se enlaza desde el XML con
<!DOCTYPE>

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<librería>
  ▼<libro>
    <título>Lenguaje de marcas</título>
    <autor>Autor2</autor>
    <autor>Autor1</autor>
    <ISBN>1234567890</ISBN>
    <paginas>158</paginas>
    <precio>15</precio>
  </libro>
  ▼<libro>
    <título>XML</título>
    <autor>Autor1</autor>
    <ISBN>0987654321</ISBN>
    <paginas>278</paginas>
    <precio>30</precio>
  </libro>
</librería>
```

```

G: > libro tema4 > libreria.xml
1  <?xml version="1.0" encoding="iso-8859-1"?>
2  <!DOCTYPE libreria SYSTEM "libreria.dtd">
3  <libreria>
4      <libro>
5          <titulo>Lenguaje de marcas</titulo>
6          <autor>Autor2</autor>
7          <autor>Autor1</autor>
8          <ISBN>1234567890</ISBN>
9          <paginas>158</paginas>
10         <precio>15</precio>
11     </libro>
12     <libro>
13         <titulo>XML</titulo>
14         <autor>Autor1</autor>
15         <ISBN>0987654321</ISBN>
16         <paginas>278</paginas>
17         <precio>30</precio>
18     </libro>
19 </libreria>
20

```

<libreria> contiene uno o mas elementos <libros>, por eso se usa el símbolo+, además cada libro debe tener un título, uno o mas autores, un ISBN, paginas y precio.

<titulo>, <autor>, <ISBN>, <paginas>, <precio> son elementos que contienen datos de texto, por lo que se define con #PCDATA.

<autor+> se utiliza el símbolo + para que haya mas de un autor

G: > libro tema4 > libreria.dtd

```
1  <!ELEMENT libreria (libro+)>
2  <!ELEMENT libro (titulo, autor+, ISBN, paginas, precio)>
3  <!ELEMENT titulo (#PCDATA)>
4  <!ELEMENT autor (#PCDATA)>
5  <!ELEMENT ISBN (#PCDATA)>
6  <!ELEMENT paginas (#PCDATA)>
7  <!ELEMENT precio (#PCDATA)>
8
```

<ELEMENT librería (libro+)> : El elemento padre es librería y debe contener uno o mas elementos libro.

<!ELEMENT libro (titulo, autor+,ISBN, paginas, precio)>: Cada libro tiene que contener un título, uno o más autores, un ISBN, el número de paginas y un precio.

<#PCDATA: Indica que los elementos contienen textos

¿Qué modificación harías en el documento DTD para que puedan añadirse varios precios, teniendo en cuenta que puede haber libros en papel y online?

```
1  <?xml version="1.0" encoding="iso-8859-1"?>
2  <!DOCTYPE libreria SYSTEM "libreria.dtd">
3  <libreria>
4      <libro>
5          <titulo>Lenguaje de marcas</titulo>
6          <autor>Autor2</autor>
7          <autor>Autor1</autor>
8          <ISBN>ISBN</ISBN>
9          <paginas>158</paginas>
10         <precio tipo="papel">15</precio>
11         <precio tipo="online">10</precio>
12     </libro>
13     <libro>
14         <titulo>Alex</titulo>
15         <autor>Autor1</autor>
16         <ISBN>ISBN</ISBN>
17         <paginas>278</paginas>
18         <precio tipo="papel">30</precio>
19         <precio tipo="online">20</precio>
20     </libro>
21 </libreria>
22
```

Para permitir múltiples precios dependiendo del formato del libro, se puede definir el elemento `<precio>` con un atributo que especifique el tipo (papel u online)

```
G: > libro tema4 > libreria.xml
1  <!ELEMENT libreria (libro+)>
2  <!ELEMENT libro (titulo, autor+, ISBN, paginas, precio+)>
3  <!ELEMENT titulo (#PCDATA)>
4  <!ELEMENT autor (#PCDATA)>
5  <!ELEMENT ISBN (#PCDATA)>
6  <!ELEMENT paginas (#PCDATA)>
7  <!ELEMENT precio (#PCDATA)>
8  <!ATTLIST precio tipo (papel | online) #REQUIRED>
9
```

Precio+: indica que puede haber uno o más elementos `<precio>`

<!ATTLIST precio tipo (papel | online) #REQUIRED> Define un atributo tipo para indicar si el precio es de un libro en papel o online, y es obligatorio.

¿Qué modificación harías en el documento DTD para que puedan añadirse varios precios, teniendo en cuenta que puede haber libros en papel y *online*?

Solución 

4. Pon un ejemplo de los siguientes elementos:

- Elementos sin atributos con datos.
- Elementos con atributos y datos.
- Elementos con atributos sin datos.

Solución 

Elementos sin atributos con datos:

```
<titulo>Lenguaje de marcas</titulo>
```

El titulo es un elemento que no tiene atributos, pero si contiene datos.

Elementos con atributos y datos:

```
<precio tipo="papel">15</precio>
```

En este caso, el precio tiene el atributo tipo y contiene dato (15)

Elementos con atributos sin datos:

```
<disponible estado="si" />
```