

## 5.1 - DOCUMENTOS XML

# DOCUMENTOS XML - INTRODUCCIÓN

- **XML**, siglas en inglés de *eXtensible Markup Language*, traducido como "Lenguaje de Marcado Extensible" o "Lenguaje de Marcas Extensible", es un metalenguaje que permite definir lenguajes de marcas desarrollado por el World Wide Web Consortium (W3C) utilizado para almacenar datos en forma legible.
- Proviene del lenguaje SGML y permite definir la gramática de lenguajes específicos (de la misma manera que HTML es a su vez un lenguaje definido por SGML) para estructurar documentos grandes.

# DOCUMENTOS

- A diferencia de otros lenguajes, XML da soporte a bases de datos, siendo útil cuando varias aplicaciones deben comunicarse entre sí o integrar información.
- XML es una tecnología sencilla que tiene a su alrededor otras que la complementan (DTD, Schemas, Xpath, Xquery, Namespaces, XSLT, DOM, etc...) y la hacen mucho más grande, con unas posibilidades mucho mayores.
- Tiene un papel muy importante en la actualidad ya que permite la compatibilidad entre sistemas **para compartir la información** de una manera segura, fiable y fácil.

# DOCUMENTOS XML - INTRODUCCIÓN

- **XML no es un lenguaje.** Es un estándar que permite crear lenguajes que cumplen una serie de reglas.( Reglas XML).
- XML describe una sintaxis que **permite crear** nuestros propios lenguajes.

- Ejemplo:

```
<note>  
  <to>Tove</to>  
  <from>Jani</from>  
  <heading>Reminder</heading>  
  <body>Don't forget me this weekend!</body>  
</note>
```

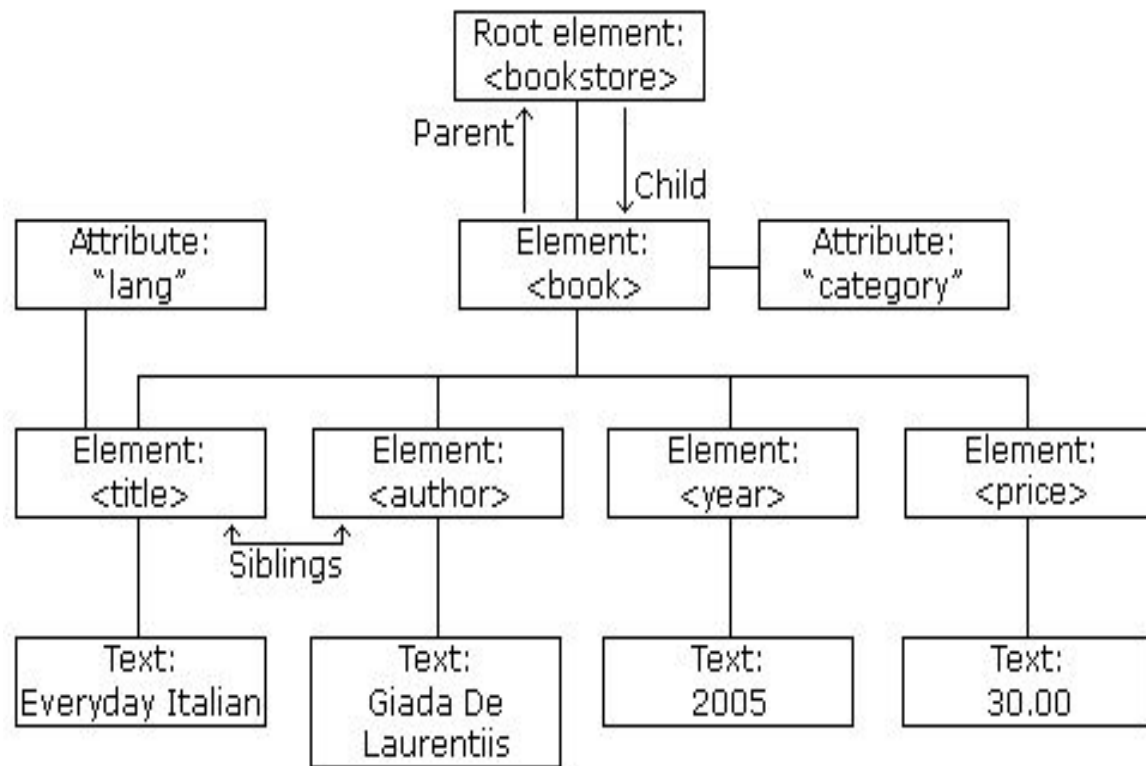
## DOCUMENTOS XML – XML vs HTML

XML y HTML se diseñaron con diferentes objetivos:

- XML se diseñó para **transportar datos**, centrándose en qué datos son.
- HTML fue diseñado para **mostrar datos**, centrándose en cómo se ven los datos.
- Las etiquetas XML no están predefinidas como las etiquetas HTML.

# DOCUMENTOS XML – ESTRUCTURA.

Los documentos XML forman una **estructura de árbol** que comienza en "la raíz" y se ramifica en "las hojas".



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

# DOCUMENTOS XML – SINTAXIS

Todo documento XML comienza con:

`<?xml version="1.0" encoding="UTF-8"?>`

o con

`<?xml version="1.0" encoding="ISO-8859-1"?>` para incluir los caracteres en castellano.

UTF-8 es la codificación de caracteres predeterminada para documentos XML.

# DOCUMENTOS XML – SINTAXIS

Los documentos XML deben tener un **elemento raíz** que sea el **padre** de todos los demás elementos:

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

Ejemplo elemento raíz:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<note>
```

```
  <to>Tove</to>
```

```
  <from>Jani</from>
```

```
  <heading>Reminder</heading>
```

```
  <body>Don't forget me this  
weekend!</body>
```

```
</note>
```



# DOCUMENTOS XML – ELEMENTOS

Según el contenido, podemos encontrar los siguientes tipos de elementos:

- Elementos vacíos: `<br/>` `<hr/>`
- Elementos con información: `<etiqueta>dato</etiqueta>`
- Elementos que contienen otros elementos anidados.
- Elementos que contienen información y a otros elementos anidados.

`<etiqueta>`

`dato`

`<otro_elemento>información </otro_elemento>`

`...`

`</etiqueta>`

# DOCUMENTOS XML – ELEMENTOS

Los elementos XML deben seguir estas reglas de nomenclatura:

- Los nombres de los elementos **distinguen** entre mayúsculas y minúsculas.
- Los nombres de los elementos **deben comenzar** con una letra o un guión bajo
- Los nombres de los elementos **no pueden comenzar con las letras xml** (o XML, o Xml, etc.)
- Los nombres de los elementos **pueden contener** letras, dígitos, guiones, guiones bajos y puntos.
- Los nombres de los elementos **no pueden contener espacios**.
- Se puede usar cualquier nombre, no se reservan palabras (excepto xml).
- Todos los elementos **deben estar correctamente anidados entre si**. El primer elemento en cerrar será el último que se abrió. `<a><b>información </b></a>`
- **IMPORTANTE: los documentos xml distinguen entre MAYÚSCULAS Y MINÚSCULAS en los nombre de elementos y atributos.**

# DOCUMENTOS XML – ELEMENTOS

Cuando se confecciona un documento, hay que empezar identificando cuáles son los elementos que aparecerán en el mismo.

Esta descomposición en elementos no es algo que se pueda hacer de forma automática, sin que tenga una única solución válida.

Una determinada solución puede ser más o menos adecuada dependiendo del uso que se vaya a hacer del documento.

```
<nombre>Juan Martín Fernández Moreno de la Vega </nombre>
```

```
<nombre>
```

```
  <nombre_pila>Juan Martín</nombre_pila>
```

```
  <apellidos>Fernández Moreno de la Vega</apellidos>
```

```
</nombre>
```

```
<nombre>
```

```
  <nombre_pila>Juan Martín</nombre_pila>
```

```
  <primer_apellido>Fernández</primer_apellido>
```

```
  <segundo_apellido>Moreno de la Vega</segundo_apellido>
```

```
</nombre>
```

# DOCUMENTOS XML – ATRIBUTOS

Los elementos XML pueden tener atributos, como en HTML.

Los atributos están diseñados para contener datos relacionados con un elemento específico. Es decir, aportan información complementaria al elemento.

Los valores de atributo siempre deben estar entrecomillados. Se pueden utilizar comillas simples o dobles.

```
<coche bastidor="78sdf8dsx">
```

```
<coche bastidor='78344dssx'>
```

Un elemento puede tener diferentes atributos separados por un espacio en blanco.

```
<producto codigo="G45">
```

```
  <nombre color="negro" precio="12.56">Gorro de lana</nombre>
```

```
</producto>
```

# DOCUMENTOS XML – ATRIBUTOS

## ¿Cuándo usar elementos y cuándo atributos?

No hay una regla fija, pero si algunas consideraciones:

- Si la información tiene una estructura interna debe ser un elemento.
- Si contiene una gran cantidad de información, parece más adecuado, un elemento.
- Podemos hacer el símil que un elemento es un sustantivo y un atributo un adjetivo.
- Los mecanismos de procesamiento y presentación de documentos, permiten tener mejor control sobre los elementos. Por tanto aquella información que tenga un procesamiento o presentación complejos debe ser un elemento.

# DOCUMENTOS XML – CARACTERES ESPECIALES

Algunos caracteres tienen un significado especial en XML.

Si coloca un carácter como "<" dentro de un elemento XML, **generará un error** porque el analizador lo interpreta como el inicio de un nuevo elemento.

```
<message>salary < 1000</message>
```

Para evitar este error, reemplace el carácter "<" con una **referencia de entidad** :

```
<message>salary &lt; 1000</message>
```

Hay 5 referencias de entidad predefinidas en XML:

&lt;	<	less than
&gt;	>	greater than
&amp;	&	ampersand
&apos;	'	apostrophe
&quot;	"	quotation mark

# DOCUMENTOS XML – CARACTERES ESPECIALES

El término **CDATA** significa, en los datos de caracteres. CDATA se definen como bloques de texto que no son analizadas por el analizador, pero son de otra forma reconocida como marcado.

`<![CDATA[ texto y caracteres que no serán procesados]]>`

Esta sección puede contener caracteres de markup ( <, >, y & ), pero son ignoradas por el procesador XML.

Ejemplo:

```
<lista>
  <nombre>John Doe</nombre>
  <correo><![CDATA[<jdoe@server.com>]]></correo>
</lista>
```

# DOCUMENTOS XML – CARACTERES ESPECIALES

## **Comentarios en XML**

La sintaxis para escribir comentarios en XML es similar a la de HTML:

`<!-- Este es el comentario -->`

No se permiten dos guiones en medio de un comentario:

`<!-- Este comentario -- no es correcto -->`



# DOCUMENTOS XML – DOCUMENTO BIEN FORMADO

Sin la presencia de un DTD, no se puede comprobar la validez de un documento, sólo se puede comprobar si está bien formado.

A continuación se enumeran algunas **reglas** que deben cumplir un documento para que se le considere **bien formado**:

- Los elementos deben estar anidados adecuadamente.
- Los valores de los atributos deben encerrarse entre comillas simples o dobles.
- Todo elemento debe tener una etiqueta de fin o utilizar la etiqueta de elemento vacío.
- El elemento debe tener un único elemento raíz.
- Todo texto debe estar incluido en un elemento.

# DOCUMENTOS XML – DOCUMENTO BIEN FORMADO

Carga en la aplicación xml-copy-editor el fichero doc-bien-formado.xml y comprueba si es un documento bien formado.

Comprobación de documento bien formado.

**Ejercicio:** Realiza algunas modificaciones sobre el contenido, elementos, etc... y comprueba los posibles errores que te muestra la aplicación.

```
1 <?xml:version="1.0" encoding="ISO-8859-1"?>
2 <oferta>
3   <vehiculo>
4     <marca>ford</marca>
5     <modelo-color="gris">focus</modelo>
6     <motor-combustible="gasolina">duratorc 1.4</motor>
7     <matricula>1234AAA</matricula>
8     <kilometros>12500</kilometros>
9     <precio_inicial>12000</precio_inicial>
10    <precio_oferta>10000</precio_oferta>
11    <extra-valor="250">pintura metalizada</extra>
12    <extra-valor="300">llantas</extra>
13    <foto>11325.jpg</foto>
14    <foto>11326.jpg</foto>
15  </vehiculo>
16  <vehiculo>
17    <marca>ford</marca>
```

# DOCUMENTOS XML – EJERCICIO LIBROS

## Ejercicio 1 – Libros

Crea un documento libros.xml en el que se contemple la información que puede disponer una librería sobre la relación de libros que tiene. Como mínimo sobre cada libro necesitaremos:

- Código ISBN
- Título
- El nivel de profundidad en el tratamiento de la materia (básico, intermedio y avanzado)
- Los autores
- La editorial
- Año de la publicación
- La página web del libro en caso de que tenga.
- El precio

Crea el documento xml con información de al menos 6 libros como ejemplo para mostrar el uso de elementos y atributos de tu modelo creado.

# DOCUMENTOS XML – EJERCICIO ALOJAMIENTOS

## **Ejercicio 2 – Alojamientos**

Visita y analiza varios portales web sobre alojamientos (hoteles, apartamentos, etc...)

Identifica la información que podría ayudarte a definir un alojamiento.

A partir de dicho análisis, crea una estructura y desarrolla un documento xml en el que puedas registrar información sobre alojamientos.

Crea el documento alojamientos.xml con información de al menos 6 alojamientos diferentes en los que muestres los diferentes valores que pueden tomar tus elementos y atributos.

# DOCUMENTOS XML CON CSS

Un documento XML puede enlazar a una hoja de estilo en cascada **CSS** mediante la instrucción de procesamiento `<?xml-stylesheet type="text/css" href="fichero.css"?>`

Ejemplo:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/css" href="ejemplo.css"?>
<libro>
  <titulo>La vida está en otra parte</titulo>
  <autor>Milan Kundera</autor>
  <fechaPublicacion año="1973"/>
</libro>
```

Si un documento XML enlaza a una hoja de estilo, los navegadores ya no muestran las etiquetas y aplican la hoja de estilo.

# DOCUMENTOS XML CON CSS

## **Ejercicio 3:**

Crea una CSS para visualizar el contenido del documento libros.xml creado en el ejercicio 1.

## **Ejercicio 4:**

Crea una CSS para visualizar el contenido del documento alojamientos.xml creado en el ejercicio 2.