

1.2 Introducción al lenguaje de marcas XML

...

LENGUAJE DE MARCAS Y SISTEMAS DE GESTIÓN DE INFORMACIÓN
1º CFGS DAW/DAM/ASIR

XML: eXtensible Markup Language

- ¿Lenguaje de marcas o metalenguaje?
 - Lenguaje de marcas porque está definido por marcas o etiquetas.
 - Metalenguaje porque define las reglas a cumplir por un lenguaje de marcas.
- Creado por **W3C** a finales de los 90, con posterioridad al primer HTML.
- Define la estructura de un documento.
- Extensión del archivo: .xml
- Documento en texto plano.
- Objetivo: Estandarización del HTML -> XHTML ... sin éxito
- Uso actual:
 - Intercambio de info. entre aplicaciones, computación distribuida, documentación empresarial, etc.

XML: Características básicas

- Se concibió para trabajar en la Web, es compatible con protocolos como HTTP
- Todo documento que verifique las reglas de XML está conforme con SGML.
- No se requieren conocimientos de programación para realizar tareas sencillas en XML.
- Los documentos fáciles de crear.
- El marcado de XML puede ser legible para los humanos.
- El diseño XML es formal y conciso.
- XML es extensible y adaptable a variedad de situaciones.
- XML es orientado a objetos.
- Todo documento XML se compone exclusivamente de datos de marcado y datos carácter entremezclados.

XML: Algunos conceptos

- **Caracteres y codificación:**

Los documentos XML pueden estar codificados en distintos juegos de caracteres (ISO-8859-1, UTF-8, etc)

- **Procesador XML (XML parser):**

La aplicación recurre a un procesador XML para leer un documento XML. Este procesador es el que lee, analiza y le pasa la información en un formato estructurado a la aplicación.

- **Etiquetas (tags) o Marcas (mark-up):**

Las etiquetas son marcas que sirven para identificar un contenido concreto del resto del contenido del documento.

Existen tres tipos de etiquetas:

- **Etiquetas de apertura** (start-tag).

`<apartado>`

- **Etiquetas de cierre** (end-tag), que empiezan por "/".

`</apartado>`

- **Etiquetas vacías** (empty-tag), que terminan por "/".

`<salto-de-linea />`

XML: Algunos conceptos

- Los documentos XML pueden estar formados por dos partes:
 - **Prólogo** (opcional)
 - Declaración XML
 - Declaración del tipo de documento
 - **Ejemplar** (Contenido del documento)

XML: Prólogo

El prólogo debe preceder al ejemplar del documento.

Facilita el procesamiento de la información del ejemplar.

Contiene:

- Declaración XML
- Declaración del tipo de documento
- La declaración XML:

Es opcional, pero, si se escribe, el atributo **version** es obligatorio.

- Atrib. **version** `<?xml version="1.0" ?>`
- Atrib. **encoding** `<?xml version="1.0" encoding="iso-8859-1" ?>`
- Atrib. **standalone** `<?xml version="1.0" encoding="iso-8859-1" standalone="yes" ?>`

Los atributos **encoding** y **standalone** son opcionales .

El atributo **encoding** siempre después de **version** y el atributo **standalone** va en último lugar.

Standalone indica si el documento xml depende de un DTD externo. Sirve a nivel de validación.

XML: Prólogo

- La declaración define el tipo de documento.
- En ella se define el tipo de documento, y dónde encontrar la información sobre su Definición de Tipo de Documento, DTD (*Document Type Definition*) mediante un identificador público (PUBLIC) que hace referencia a dicha DTD, o mediante un *Identificador Universal de Recursos* (URI) precedido por la palabra SYSTEM.
- Este archivo DTD valida y define los datos que contiene nuestro documento XML.

`<!DOCTYPE nombre_tipo [PUBLIC | SYSTEM] archivo.dtd>`

DTD en archivo externo:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<!DOCTYPE Persona SYSTEM "persona.dtd">
```

DTD embebido en documento:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE persona [
    <!ELEMENT persona (nombre,apellido,ciudad)>
    <!ELEMENT nombre (#PCDATA)>
    <!ELEMENT apellido (#PCDATA)>
    <!ELEMENT ciudad (#PCDATA)>
]>
```

Contenido del
fichero DTD si
fuera externo.

```
<persona>
  <nombre>Carlos</nombre>
  <apellido>Carnero</apellido>
  <ciudad>Valencia</ciudad>
</persona>
```


XML: Ejemplar y elementos

El ejemplar contiene los datos reales del documento XML, **el contenido**.

Está formado por un elemento raíz que a su vez contendrá otros elementos anidados.

- **Elemento:** componente lógico de un documento XML.
 - Comienza por una etiqueta de apertura y termina con la etiqueta de cierre correspondiente, o bien consiste en una única etiqueta vacía.
 - El contenido de un elemento es todo lo que se encuentra entre las etiquetas de apertura y cierre.
 - También existen elementos vacíos, los cuales no tienen contenido.

- Elemento con texto como contenido:

```
<autor>EsCif</autor>
```

- Elementos anidados:

```
<falla>
```

```
  <autor>EsCif</autor>
```

```
</falla>
```

- Elemento vacío o sin contenido:

```
<ingreso />
```

o también puede escribirse así:

```
<ingreso></ingreso>
```

XML: Ejemplar y elementos

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE persona>
<persona>
  <nombre>Carlos</nombre>
  <apellido>Carnero</apellido>
  <nacimiento>
    <anyo>1956</anyo>
    <mes>Abril</mes>
    <dia>13</dia>
  </nacimiento>
  <ciudad>Alicante</ciudad>
</persona>
```

**Busquemos prólogo,
ejemplar, el elemento
raíz y resto de
elementos**

XML: Atributos

Un atributo es un componente de las etiquetas que consiste en una pareja nombre (name) / valor (value).

Se puede encontrar en las etiquetas de apertura o en las etiquetas vacías, pero no en las de cierre. En una misma etiqueta no pueden existir dos atributos con el mismo nombre.

La sintaxis es siempre: **nombreAtributo="valorAtributo"**

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<!DOCTYPE persona>
<persona nombre="Alfonso" apellidos="García Fernández">
  <nacimiento dia='13' mes='mar' anyo='2019'></nacimiento>
  <ciudad cp="46001">Valencia</ciudad>
</persona>
```

XML: Definir atributos en el DTD

Para definir atributos para un elemento, usaremos la etiqueta **!ATTLIST** indicando a continuación el nombre del elemento al que pertenece los atributos que se van a definir:

```
<!ELEMENT  persona (nombre,apellido,ciudad)>
<!ELEMENT  nombre (#PCDATA)>
<!ELEMENT  apellido (#PCDATA)>
<!ELEMENT  ciudad (#PCDATA)>
<!ATTLIST  ciudad cp CDATA #REQUIRED
           provincia CDATA "Valencia">
```

XML: Ejercicio. Encuentra los errores

```
<?XML version='1.0' encoding='UTF-8' standalone='yes'?>
<!DOCTYPE biblioteca SYSTEM "biblioteca.dtd">
<biblioteca>
  <ejemplar tipo_ejem='libro' editorial='Canalla'>
    <tipo>
      <libro isbn='978-2-7460-4958-1' edicion=paginas='347'>XML
Práctico</Libro>
    </tipo>
    <autor1 nombre=''>Sebastien Lecomte</autor1>
    <autor2 nombre=Thierry Boulanger />
    <Autor3 nombre= funcion='traductor'>Ángel Belinchon Calleja</autor3>
    <prestado lector='Pepito Grillo'>
      <fecha_pres dia='13' mes='mar' anyo='2019'></fecha_pres>
      <fecha_devol />
    </prestado>
  </biblioteca>
```

XML: Ejercicio. Encuentra los errores

```
<?XML version='1.0' encoding='UTF-8' standalone='yes'?>
<!DOCTYPE biblioteca SYSTEM 'biblioteca.dtd'>
<biblioteca>
  <ejemplar tipo_ejem='libro' editorial='Canalla'>
    <tipo>
      <libro isbn='978-2-4958-1' edicion=paginas='347'>Guía XML</Libro>
    </tipo>
    <autor1 nombre=''>Sebastien Lecomte</autor1>
    <autor2 nombre=Thierry Boulanger />
    <Autor3 nombre= funcion='traductor'>Ángel Belinchon Calleja</autor3>
    <prestado lector='Pepito Grillo'>
      <fecha_pres dia='13' mes='mar' anyo='2019'></fecha_pres>
      <fecha_devol />
    </prestado>
    ...
  </ejemplar>
</biblioteca>
```

XML: Ejercicio. Encuentra los errores

```
<?xml version='1.0' encoding='UTF-8' standalone='no'?>
<!DOCTYPE biblioteca SYSTEM 'biblioteca.dtd'>
<biblioteca>
  <ejemplar tipo_ejem='libro' editorial='Canalla'>
    <tipo>
      <libro isbn='978-2-4958-1' edicion="1" paginas='347'>Guía XML</Libro>
    </tipo>
    <autor1 nombre=-->Sebastien Lecomte</autor1>
    <autor2 nombre='Thierry Boulanger' />
    <autor3 nombre= funcion='traductor'>Ángel Belinchon Calleja</autor3>
    <prestado lector='Pepito Grillo'>
      <fecha_pres dia='13' mes='mar' anyo='2019' /></fecha_pres>
      <fecha_devol />
    </prestado>
  </ejemplar>
</biblioteca>
```

Tarea 1.1:

Creación de un documento XML.

Crea un documento XML con un DTD externo definido por tí.
(Ver enunciado en Aules)



That's all Folks!