

Ejercicios de Lenguaje de Marcas

Nombre: Francisco Jesús García – Uceda Díaz – Albo.

Curso ASIR 1º.

a) Identifica las características generales de los lenguajes de marcas.

- **Texto Plano:** Son archivos que están compuestos únicamente por texto sin formato, sólo caracteres. Estos pueden ser codificados de distintas maneras, los más usados son ASCII, ISO-8859-1 o Latín-1, Unicode...

Un extra de ventaja es que como los lenguajes de marcas son en texto plano se pueden interpretar directamente.

- **Compacidad (o compactación):** Las instrucciones de marcado se entremezclan con el propio contenido en un único archivo o flujo de datos. Este es un ejemplo en diferentes lenguajes de marcas:

El código entre corchetes con o códigos \section, son instrucciones de marcado que también son llamado etiquetas. Las etiquetas, en concreto, son descriptivas de la estructura del documento pudiendo ser su presentación visual de distintas formas.

- **Facilidad de procesamiento:** Las organizaciones de estándares han venido desarrollando lenguajes especializados para los tipos de documentos de comunidades o industrias concretas. Esto ha conducido que sus manuales se editen únicamente en versiones electrónica y después se obtenga a partir de las versiones impresas.
- **Flexibilidad:** Aunque los lenguajes de marcas se idearon para ser documentos de texto, se han empezado a utilizar en otra área como servicios web o gráficos vectoriales. Esto ha ocurrido por la sencillez y potencia de los lenguajes de marca que ha permitido combinar varios lenguajes de marcas en un mismo archivo para sacarle mucho más partido.

b) Enumera las ventajas que proporcionan en el tratamiento de la información.

Son diversas las ventajas que nos ofrece el tratamiento de la información:

Ventajas:

- Reutilización de los datos.
- Integridad y mayor control sobre los datos.
- Portable.
- Flexible, multipropósito.
- Perdurabilidad de la información.

Inconvenientes:

- Alta complejidad.

c) Clasifica los lenguajes de marcas e identificado los más relevantes.

- **De presentación:** Indica el formato del texto (información para el maquetado). Este tipo de marcado es útil para maquetar la presentación de un documento para su lectura, pero, por el contrario, resulta insuficiente para el procesamiento de la información. Ejemplo de marcado de presentación son: Por ejemplo: Rich Text Format (RTF), S 1000D, TeX, troff, HTML...
- **De procedimiento:** Orientado también a la presentación, pero, en este caso, se indican los procedimientos que deberá realizar el software de representación. El programa representara el documento que debe interpretar el código en el mismo orden que aparece.

Algunos ejemplos de marcado de procedimientos son nroff, troff, TeX. Este tipo de marcado se ha usado extensamente en aplicaciones de edición profesional, manipulados por tipógrafos calificados, ya que puede llegar a ser extremadamente complejo.

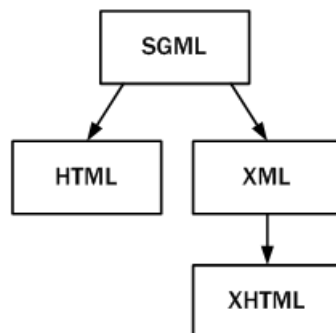
- **Descriptivo o semántico:** Describen las diferentes partes en la que se estructura el documento, pero sin especificar cómo deben representarse. Utiliza etiquetas para describir los fragmentos de texto, pero sin especificar cómo deben ser representados o en que orden. Los lenguajes expresamente diseñados para generar marcado descriptivo son el SGML y el XML.

d) Clasifica los lenguajes de marcas según los ámbitos de aplicación.

- **De presentación:** Su ámbito es maquetar la presentación.
- **De procedimiento:** Su ámbito está enfocado a la presentación del texto.
- **Descriptivo o semántico:** Su ámbito está enfocado a describir los fragmentos del texto, pero si especificar cómo deben ser representados o en qué orden.

e) Indica la necesidad y los ámbitos específicos de aplicación de un lenguaje de marcas de propósito general.

XHTML es muy similar al lenguaje HTML. De hecho, XHTML no es más que una adaptación de HTML al lenguaje XML. Técnicamente, HTML es descendiente directo del lenguaje SGML, mientras que XHTML lo es del XML (que, a su vez, también es descendiente de SGML).



En resumen, XHTML es casi idéntico al HTML 4.01 con sólo algunas diferencias. Esta es una versión más limpia y más estricta de HTML 4.01. XHTML fue desarrollado por el W3C para ayudar a los desarrolladores web a hacer la transición de HTML en XML.

f) Analizado las características propias del lenguaje XML.

- Es una simplificación y adaptación del SGML. Por lo tanto, XML no es realmente un lenguaje en particular, sino una manera de definir lenguajes para diferentes necesidades. Algunos de estos lenguajes que usan XML para su definición son XHTML, SVG, MathML.
- Se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

- XML es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande y 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.

g) Explica la estructura de un documento XML y sus reglas sintácticas.

XML puede parecer a simple vista a HTML, pero hay una diferencia principal. Un documento XML contiene datos que se autodefinen, exclusivamente. Un documento HTML contiene datos mal definidos, mezclados con elementos de formato. En XML se separa el contenido de la presentación de forma total.

- HTML no indica lo que está representando
- XML describe el contenido de lo que etiqueta

Un Ejemplo de HTML:

```
1<b>Elizabeth Castro</b>
2<b><i>XML Guía de Aprendizaje</i></b>
3<b>precio: 30 €</b>
```

El mismo Ejemplo en XML:

```
1<libro>
2 <autor>Elizabeth Castro</autor>
3 <titulo>XML Guía de Aprendizaje</titulo>
4 <precio moneda="euros">30</precio>
5</libro>
```

⇒ Observamos como HTML se centra en colocar etiquetas para presentar la información.

⇒ XML utiliza etiquetas para definir el contenido y el significado de la información.

Las reglas importantes a tener en cuenta en XML son:

- Todo documento XML debe tener un elemento raíz que no puede duplicarse, y que es el primero que se abre y el último que se cierra. En nuestro ejemplo es empleados.
- Los atributos siempre deben ir entrecomillados.
- Para los atributos que sean booleanos, su valor debe explicitarse de manera redundante.
- Los espacios en blanco se preservan.
- Los comentarios se marcan como en HTML.

- Los nombres de los elementos pueden contener letras, números y tres signos de puntuación: guión (-), guión bajo (_) y punto (.).
- Los nombres de los elementos no pueden empezar por un número o un signo de puntuación (excepto el guión bajo).
- Los nombres de los elementos no pueden empezar por las letras «xml» (ni cualquier combinación de caso, como XML, Xml, etc.).
- Los nombres de los elementos no pueden contener espacios.

Siguiendo estas reglas, nos aseguramos que la sintaxis está bien formada y que un analizador sintáctico no generará ningún error.

h) Explica que es un documento XML bien formados.

Un documento XML se dice "bien formado" cuando cumple una serie de reglas descritas en la especificación oficial de XML (ver reglas arribas puestas en g).

Para comprobar que un archivo xml está bien formado es tratar de abrirlo con un navegador (que contiene un analizador XML interno) en caso de que el archivo esté bien formado, lo mostrará cómo se ve en la Ilustración 1. Si se produce algún error, se mostrará el error que se muestra en la Ilustración 2.

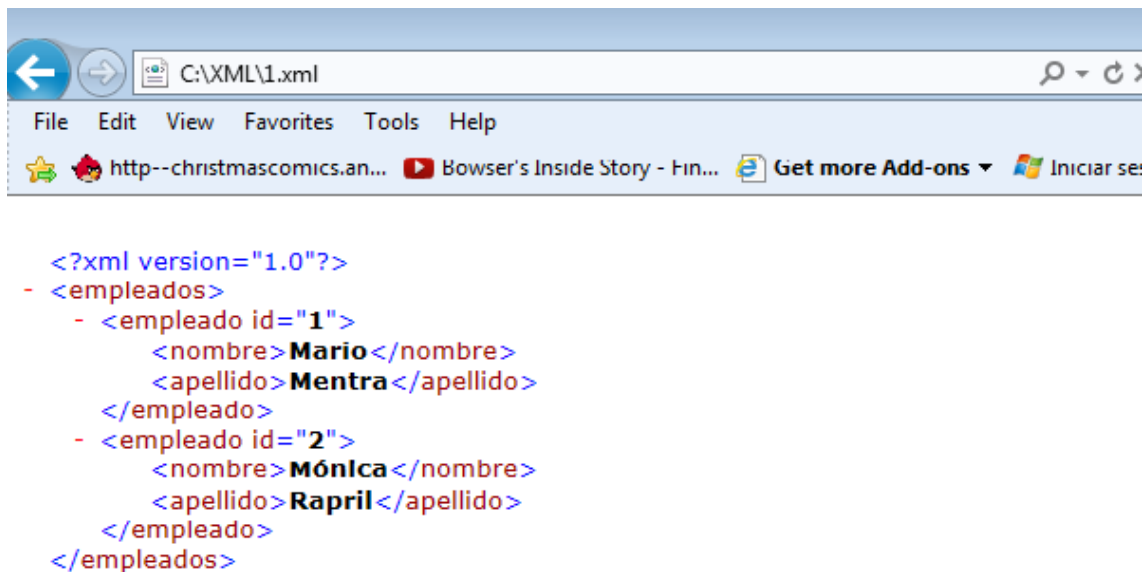


Ilustración 1

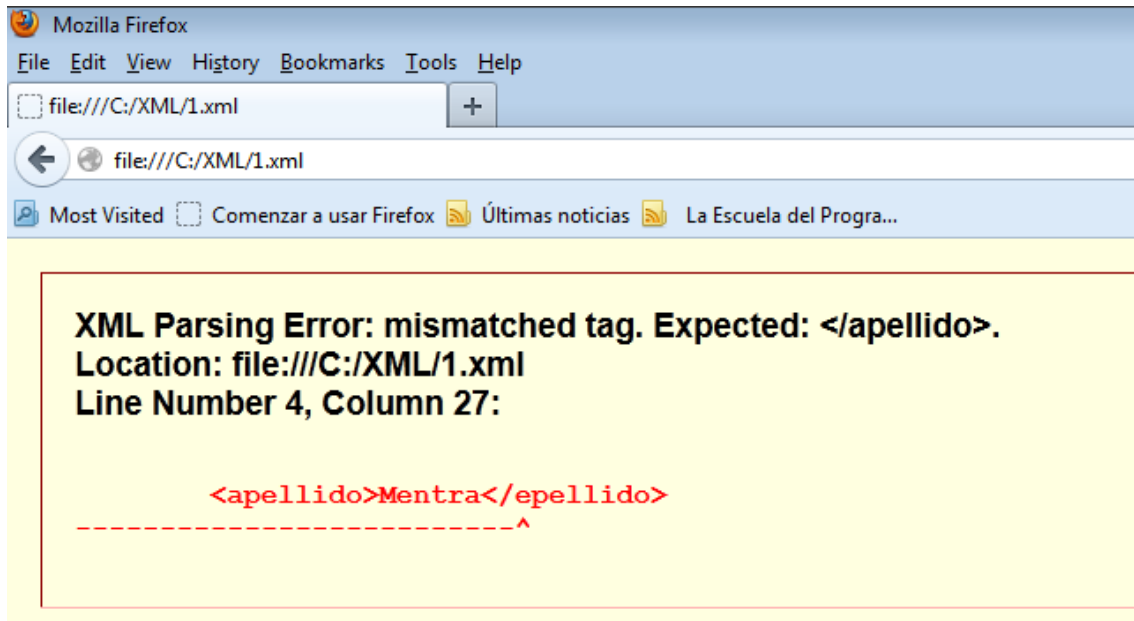


Ilustración 2