

Bloque1: Reconocimiento de las caracter sticas de lenguajes de marcas



Visualizaci n de videos introductorios en el bloque1:

RECONOCIMIENTO DE LAS CARACTER STICAS DEL LENGUAJE

Links de inter s



Video 1. Etiquetando la Red



V deo 2. Ejemplos de uso de lenguajes basados en XML

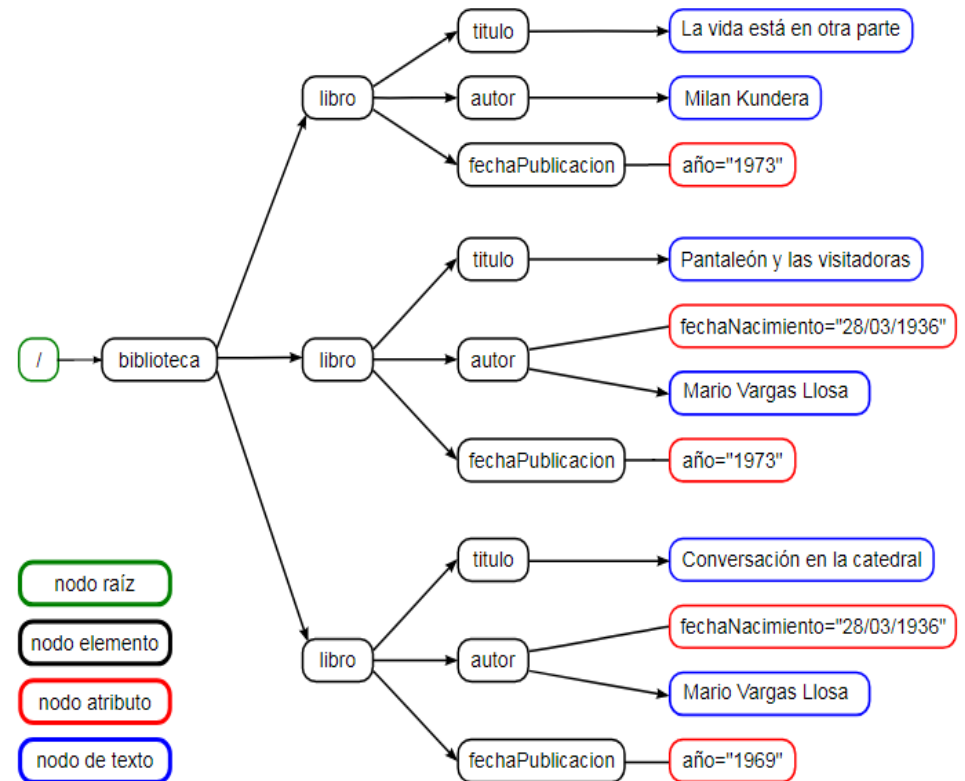
- **XML** (eXtensible Marckup Language): **Formato de almacenamiento de información** a base de etiquetas o marcas definidas por el usuario.
- Es un **metalenguaje de marcas** (no dispone de un conjunto fijo de etiquetas) que los que lo utilizan deben conocer.
- Debe cumplir una serie de reglas que harán que un **documento XML** esté **bien formado** (sintaxis correcta).
- Tiene mecanismos para **validar un documento XML** y por tanto crear un lenguaje de marcas para un uso específico. Utilizaremos **esquemas XSD**.
- Algunos lenguajes basados en XML: [SVG](#), [WML](#), [RSS](#), etc...

¿Qué NO es XML?

- **XML NO es un lenguaje de programación.** Es un METALENGUAJE (de marcado)!!!
- **XML NO es un protocolo de comunicación** (no envía datos a través de la red).
- **XML NO es un Sistema Gestor de Bases de Datos** (en sí mismo).
- **XML NO es propietario** (no pertenece a ninguna compañía como ocurre con otros formatos).

- La información **se estructura de forma jerárquica** con relaciones de padres, hijos, hermanos, ascendentes, descendentes, etc...

```
<?xml version="1.0" encoding="UTF-8"?>
<biblioteca>
  <libro>
    <titulo>La vida está en otra parte</titulo>
    <autor>Milan Kundera</autor>
    <fechaPublicacion año="1973"/>
  </libro>
  <libro>
    <titulo>Pantaleón y las visitadoras</titulo>
    <autor fechaNacimiento="28/03/1936">Mario Vargas Llosa</autor>
    <fechaPublicacion año="1973"/>
  </libro>
  <libro>
    <titulo>Conversación en la catedral</titulo>
    <autor fechaNacimiento="28/03/1936">Mario Vargas Llosa</autor>
    <fechaPublicacion año="1969"/>
  </libro>
</biblioteca>
```



- **No dispone de una visualización concreta** en un navegador (no refleja una apariencia, sino unos datos y su estructura).
- Maneras de representar visualmente los datos de un XML:
 - Mediante hojas de estilo CSS (indican al navegador cómo representar cada elemento del documento XML): <ejemplo>

```
<? xmlstylesheet type="text/css" href="estilos.css" ?>
```
 - Mediante uso de una hoja de transformaciones XSLT:

```
<? xmlstylesheet type="text/xsl" href="transforma.xsl" ?>
```
 - Mediante el uso de lenguajes de programación (Java, Javascript, C#, etc...) que procese el documento XML.

- Es un **procesador que lee documentos XML** y determina la estructura y propiedades de los datos que contiene.
- Un analizador estándar lee el documento XML y **genera el árbol jerárquico asociado**.
- Si el analizador comprueba las reglas de buena formación y además valida el documento contra un DTD o XSD (esquema) es un **analizador validador** (comprueban la semántica del documento e informan errores existentes).
- Existen validadores XML en línea. Por ejemplo:
 - XML Validation (<https://www.xmlvalidation.com/>)
 - Freeformatter (<https://www.freeformatter.com/xml-validator-xsd.html>)

- Existen **multitud de herramientas** relacionadas con el manejo de XML y sus tecnologías asociadas (DTD, XSD, XSLT, Xpath, XQuery, etc...).
- Algunas sencillas de código abierto:
- XMLSpy de Altova (<http://www.altova.com/es>).
- <oXygen/> de Sincro Soft (<http://www.oxygenxml.com>).
- XML Copy Editor (<http://xml-copy-editor.sourceforge.net/>).
- XMLPad Pro Edition de Wmhelp...
- Y muchas más....



Empezaremos utilizando
XML Copy Editor

- La especificación de XML (<https://www.w3.org/>) define la sintaxis que el lenguaje debe seguir:
 - Cómo se delimitan los elementos con etiquetas.
 - Qué formato puede tener una etiqueta.
 - Qué nombres son aceptables para los elementos.
 - Donde se colocan los atributos.
 - Etc...
- Un documento XML está **BIEN FORMADO** si cumple las **reglas establecidas por el W3C en las especificaciones para XML**.
- Las reglas son numerosas y en ocasiones complejas (las trabajaremos en **Actividad Práctica 1**)

- Un documento es **VÁLIDO** si **existen unas reglas de validación** (p.e. en formato DTD o XSD) asociadas a él y que deba cumplir.
- Estas reglas **especifican la estructura gramatical y sintaxis que debe tener el documento XML** (dicen qué elementos o atributos concretos pueden aparecer, en qué orden, de qué tipos, que valores pueden tomar, etc...)
- Todo documento válido está bien formado pero no a la inversa:

XML VÁLIDO



XML BIEN FORMADO

ACTIVIDAD PR CTICA 1 (AP1)

T tulo

Introducci n a XML (primeros pasos)

Objetivos

- Identificar las caracter sticas b sicas del lenguaje XML.
- Saber las normas de sintaxis b sicas de construcci n (elementos, atributos, comentarios, secciones CDATA, espacios en blanco, etc...)
- Saber declarar y crear un documento XML con una sintaxis correcta y una aplicaci n / herramienta inform tica sencilla.
- Conocer la funcionalidad de una instrucci n de procesamiento en XML.
- Conocer las caracter sticas de uso de un espacio de nombres en un documento XML.
- Realizar y construir documentos XML bien formados.
- Saber las caracter sticas de un documento v lido y la relaci n entre el concepto de documento XML bien formado y documento XML v lido.