**UNIDAD 1: ASPECTOS GENERALES DE LOS LENGUAJES DE MARCA. HTML y XML**

**Introducción**

A grandes rasgos podemos decir que un lenguaje de marcas es una forma de codificar un documento.

En el documento se guarda, además del texto, una serie de marcas o etiquetas que hacen referencia a la distribución y presentación del texto.

Con el lenguaje de marcas hacemos visible la estructura de documentos. El lenguaje de marcas define dicha estructura. No es un lenguaje de programación. No tiene condiciones, ni funciones aritméticas, ni siquiera variables.

**1.1. | Características y Elementos de un lenguaje de marcas |**

Sus características son:

• Uso de texto plano. Eso hace que cualquier plataforma pueda trabajar con los documentos directamente.

• Fáciles de interpretar y procesar

• Fáciles de crear y flexibles para representar datos muy diversos. Por ese motivo se puede combinar con diferentes lenguajes de programación como PHP y JavaScript.

• Permiten el uso de metadatos, como el autor del documento, la fecha de creación, etc.

En cuanto a los elementos de un lenguaje de marcas, hay que indicar que varían de un lenguaje a otro, pero existen unos elementos comunes que son:

**Etiquetas o tags**

Son las palabras que indican el inicio o final de un elemento en el documento. En HTML empiezan con un <etiqueta> y terminan con un </etiqueta>

**Atributos**

Características adicionales que se agregan a algún elemento para especificar su estilo. Con ello podemos, por ejemplo, indicar el color, la posición o la URL de un hipervínculo.

**Contenido**

Se trata de los datos que forman parte del documento propiamente dicho. Por ejemplo, el texto con el título entre dos etiquetas <h1> y </h1> en HTML.

**Comentarios**

Notas que se agregan al código para dar más información o explicaciones sobre lo que hacemos. En HTML los comentarios se escriben entre dos símbolos de etiqueta que son “<!-->” y “-->”.

**1.2. | Ventajas del uso de lenguajes de marcas en el tratamiento de la información |**

Las principales ventajas son:

• Separan el contenido de la presentación. Esto permite que podamos hacer cambios en la presentación sin alterar el contenido de forma totalmente independiente.

• Permiten describir el sentido o la semántica de los datos. Esto significa que los lenguajes de marca pueden expresar el significado y uso de los datos que se almacenan o se intercambian entre diferentes   
plataformas y/o aplicaciones.

• Siguen estándares internacionales por lo que facilitan el procesamiento e interoperabilidad entre diferentes plataformas y aplicaciones.

• Son abiertos, es decir, no son propiedad de ninguna entidad.

**1.3. | Clasificación de los lenguajes de marcas |**

Existen varios tipos de clasificaciones:

Basado en la sintaxis:

1.1. Basada en etiquetas (HTML, Markdown)

1.2. Basada en claves y valores (JSON, YAML)

Basado en la finalidad:

2.1. Creación de páginas web (HTML)   
2.2. Intercambio de datos entre aplicaciones (XML)   
2.3. Distribución de contenido de forma automatizada (RSS)   
2.4. Escritura de textos sencillos (Markdown)   
2.5. Serialización de datos estructurados (YAML)   
2.6. Creación de documentos estructurados y complejos (SGML)

Basado en el tipo de documento:

3.1. Creación de gráficos vectoriales escalables (SVG)   
3.2. Codificación de fórmulas matemáticas en documentos HTML (MathML)   
3.3. Páginas web (HTML)   
3.4. Gráficos vectoriales escalables. (SVG)   
3.5. Textos literarios y documentos históricos en formato XML (TEI)

Basado en la estructura:

4.1. Estructurados, descriptivo o semántico. Hacen referencia a las partes en las que se divide un documento, pero no entra en los formatos (XML)   
4.2. No estructurado o de presentación. Afectan únicamente al formato del texto (Markdown).   
4.3. Híbrido (HTML)

**1.4. | Ámbitos de aplicación |**

De todo lo expresado hasta ahora podemos inferir que existen númerosos ámbitos de aplicación de los lenguajes de marca.

- Documentación electrónica: se usan para dar formato a los documentos de texto, como RTF (Rich Text Format) o PDF (Portable Document Format).

- Estructura de datos: se usan para definir la estructura y el contenido semántico de los datos de un documento, como XML (Extensible Markup Language) o JSON (JavaScript Object Notation).

- Intercambio de ficheros: se usan para permitir el intercambio de ficheros entre diferentes aplicaciones y plataformas, como CSV (Comma Separated Values) o YAML (YAML Ain't Markup Language).

- Web: se usan para crear y presentar páginas web, como HTML (HyperText Markup Language) o CSS (Cascading Style Sheets).

- Matemática: se usan para expresar notación matemática, como MathML (Mathematical Markup Language) o OpenMath.

- Geomática: se usan para modelar, transportar y almacenar información geográfica, como Geography ML o KML (Keyhole Markup Language).

- Aeronáutica: se usan para describir las características y el comportamiento de las naves espaciales, como Spacecraft ML.

- Multimedia: se usan para integrar diferentes tipos de medios en una presentación sincronizada, como SMIL (Synchronized Multimedia Integration Language) o SVG (Scalable Vector Graphics).

Existen infinitos ámbitos de aplicación, esta es una lista no cerrada, dado que existen muchos campos de trabajo.

**1.5. | Necesidad de un lenguaje de marcas de propósito general |**

Un lenguaje de marcas de propósito general es un lenguaje que se puede utilizar en más de un propósito.

Además permite definir tus propias etiquetas o marcas para codificar un documento con información adicional sobre la estructura o la presentación del texto.

Un ejemplo de lenguaje de marcas de propósito general es XML (eXtensible Markup Language). Es una especificación del W3C que se usa para compartir datos a través de diferentes sistemas, como Internet.

XML tiene una estructura y unas reglas sintácticas que se deben seguir para crear documentos válidos.

**1.6. | Cuales son las características propias del lenguaje XML |**

• Lo primero es que es un lenguaje extensible, es decir, que permite crear etiquetas propias para organizar los documentos.

• Es un metalenguaje. Se basa en otro lenguaje, en este caso en SGML que ayuda a definir otros lenguajes de marcas.

• Es una especificación de W3C, el Consorcio del World Wide Web. Esta es una organización internacional que crea los estándares para la web.

• Es un lenguaje legible.

• Compatible con bases de datos.

**1.7. | Reglas sintácticas y estructura de un documento XML |**

La estructura de un documento XML está basada en un conjunto de reglas sintácticas que tenemos que cumplir si queremos que el documento sea válido.

Las reglas más importantes son:

• Los elementos de un documento XML han de seguir una estructura de árbol. Es por tanto una estructura jeráquica estricta.

• Los documentos XML deben tener un elemento raíz que contenga todos los demás elementos.

• Los elementos tienen que estar correctamente anidados.

• XML es case sensitive, es decir, distingue entre mayúsculas y minúsculas. Por lo tanto, no es lo mismo <Ordenador> que <ordenador>

• Todos los elementos han de abrirse y cerrarse con una etiqueta, o bien autocontenerse con el carácter / al final.

• Los atributos deben tener un valor entre comillas y no pueden repetirse en el mismo elemento.

**1.8. | La necesidad de crear documentos XML bien formados |**

El cumplimiento estricto de las normas que hemos visto antes y demás reglas sintácticas, es necesario para tener un documento XML bien formado y válido.

Que los documentos estén formados es condición necesaria para facilitar la transmisión de información a través de la web.

Además, si no hay errores de sintaxis y cumple con todas las reglas, se va a facilitar mucho su procesamiento por parte de las aplicaciones y lenguajes que lo requieran.

**1.9. | Los espacios de nombres en XML |**

Son una forma de evitar conflictos entre elementos o atributos que se llamen igual, pero pertenezcan a diferentes vocabularios XML. Definidos por una recomendación del W3C.

Cada espacio de nombres tiene una URI (Identificador uniforme de Recursos) específica que lo identifica y un prefijo opcional para calificar los elementos o atributos que le pertenecen.

En el ejemplo el prefijo xhtml se usa para indicar que los elementos html, head, title, body y p pertenecen al espacio de nombres xhtml.