

Introducción a Lenguaje de Marcas

1. Conceptos y ventajas.

¿Qué es una marca en un texto?

Una **marca** es una señal que se coloca dentro de un texto para delimitar una parte específica de este. En algunos casos, esa marca también aplica un formato particular a la sección que delimita. Sin embargo, las marcas también pueden tener otros propósitos, no solo el formato.

Estructura de las marcas

Las marcas más comunes están compuestas por una palabra que describe su función, y están delimitadas por los símbolos de menor (<) y mayor (>). Un ejemplo es: `<html>`.

Normalmente, estas marcas aparecen en pares:

- **Marca de inicio** (por ejemplo, `<h1>`)
- **Marca de cierre** (por ejemplo, `</h1>`).

Ejemplo de marcas:

```
html
```

```
<h1> Primera prueba </h1>
```

```
<h2> Segunda prueba </h2>
```

¿Qué son los lenguajes de marcas?

Un **lenguaje de marcas** es un tipo de lenguaje que combina información textual con marcas o anotaciones sobre la estructura o presentación de los datos en un documento. Estos lenguajes no son solo para formato, sino que también estructuran cómo se organizan los datos.

Características de los lenguajes de marcas

El **lenguaje de marcas** define las **etiquetas** (o marcas) que se pueden usar, dónde colocarlas, y qué significado tiene cada una. Normalmente, las marcas no se muestran al usuario final, ya que el usuario está más interesado en el contenido visible.

Diferencia con lenguajes de programación

A pesar de llamarse también lenguajes, los **lenguajes de marcas** no son lenguajes de programación. No tienen elementos como variables, arrays, sentencias de control, o funciones, que son comunes en los lenguajes de programación como **Python** o **PHP**.

Sin embargo, **los lenguajes de marcas** pueden combinarse con lenguajes de programación dentro de un mismo documento, como ocurre en las páginas web, para añadir funcionalidad y dinamismo.

¿Cómo se interpretan los lenguajes de marcas?

Los **lenguajes de marcas** pueden ser interpretados por diferentes tipos de "clientes", no solo navegadores. Por ejemplo, hay aplicaciones móviles híbridas que también utilizan estos lenguajes. Por eso, es más adecuado hablar de **agentes de usuario** (user agents) en lugar de solo "navegadores".

Independencia del lenguaje de marcas respecto al destinatario final

El **lenguaje de marcas** debe ser **independiente del destinatario final**, es decir, no está diseñado para adaptarse a un tipo específico de usuario o dispositivo. En su lugar, el **intérprete del lenguaje** (como un navegador web) es el encargado de representar las marcas de manera adecuada.

Discrepancias entre navegadores

Debido a esto, pueden existir discrepancias en la manera en que diferentes navegadores representan las mismas etiquetas. Por ejemplo, algunos navegadores pueden soportar ciertos elementos como `<svg>` (para gráficos vectoriales) o ciertos tipos de campos, como los campos de fecha, de maneras distintas.

¿Qué es SVG?

SVG significa **Scalable Vector Graphics**, un lenguaje basado en **XML** utilizado para construir gráficos vectoriales 2D. Es flexible y soporta una gran variedad de efectos y características avanzadas. Al ser una **recomendación del W3C** (World Wide Web Consortium), su formato está basado en XML, lo que lo hace abierto y accesible, a diferencia de otros formatos binarios cerrados.

Independencia entre contenido y presentación: CSS

Para separar claramente el **contenido** de una página web de su **presentación visual**, se creó el lenguaje **CSS (Cascading Style Sheets)**. Con **CSS**, se puede controlar cómo se presenta la información en distintos medios (monitores, dispositivos móviles, papel, etc.) y para diferentes **intérpretes** (como navegadores).

Este enfoque es especialmente útil con el auge de los dispositivos móviles, ya que muchas páginas web ofrecen versiones adaptadas según el dispositivo que esté utilizando el usuario.

Ejemplos de lenguajes y su propósito

1. Lenguajes que representan datos:

- **XML, XHTML, HTML5**: Se encargan de estructurar y representar los datos.
- Lenguajes basados en XML como:
 - **MathML**: Para representar lenguaje matemático.
 - **NewsML**: Para representar noticias electrónicas.
 - **CML**: Para la notación química.

2. Lenguajes que definen la presentación de datos:

- **CSS**: Define la presentación de documentos HTML.

- **XSL**: Define la presentación de documentos XML.

3. Lenguajes que definen gráficos o multimedia:

- **SVG**: Define la representación de gráficos vectoriales dentro de cualquier documento.
- **SMIL**: Permite integrar audio, video, imágenes y otros contenidos multimedia.

4. Otros lenguajes especializados:

- **XAML**: Define interfaces gráficas para aplicaciones en el sistema operativo Windows.
- **LaTeX**: Es un sistema de composición de textos de alta calidad, muy utilizado en ambientes científicos para documentos complejos como artículos de investigación.

2. Evolución de los lenguajes de marcas

El problema de compatibilidad en los años 60

En los años 60, las empresas que trabajaban con **publicaciones y documentos electrónicos** se enfrentaban a un gran problema: la **falta de compatibilidad** entre las diferentes aplicaciones que utilizaban. Cada aplicación tenía sus **propias marcas** para describir los distintos elementos de un documento, lo que hacía muy difícil o imposible **intercambiar documentos** entre plataformas o sistemas diferentes.

Además, había otro problema importante: la **falta de separación entre la estructura** del documento (su contenido) y su **aspecto visual** (cómo se veía).

Intentos de solución: GML y SGML

Para abordar estos problemas, IBM intentó resolver la situación con un lenguaje llamado **GML** (Generalized Markup Language). Sin embargo, no fue hasta que la **ISO** (International Organization for Standardization) intervino que se dio un paso importante. La ISO creó **SGML** (**Standard Generalized Markup Language**), que es un **metalinguaje**.

¿Qué es SGML?

Un **metalinguaje** es un conjunto de reglas o normas que permite **crear otros lenguajes de marcas**. SGML proporcionó una base sólida y estandarizada para que las empresas y desarrolladores pudieran diseñar **lenguajes compatibles entre sí**.

Nacimiento de HTML

A partir de SGML, se creó **HTML** (**HyperText Markup Language**), que es el lenguaje de marcas que se utiliza ampliamente para la creación de páginas web.

3. características

Texto plano

1. Características:

- Los **archivos de texto plano** están formados únicamente por **caracteres de texto**, a diferencia de los **archivos binarios**, que pueden contener imágenes, sonidos, archivos comprimidos, etc.
- Los caracteres de texto pueden codificarse con diferentes **códigos**, como **ASCII** o **UTF-8**.
- Los archivos de texto plano pueden ser **abiertos directamente** por cualquier editor de texto, mientras que los archivos binarios necesitan **software especializado** para abrirse.
- Al ser archivos de texto, son **independientes del sistema operativo**, lo que facilita la **interoperabilidad**, especialmente en la comunicación a través de Internet.

Compacidad

- En los lenguajes de marcas, las **instrucciones de marcado** se mezclan con el propio contenido del documento. Algunos ejemplos:
 - **HTML**: `<h2> Contenido </h2>`
 - **LaTeX**: `\section{Título}`
 - **Wikitexto**: `== Título ==`

Independencia del agente de usuario

- Un **mismo documento** puede ser interpretado de manera diferente dependiendo del **dispositivo o navegador** (agente de usuario) en el que se visualice. Esto permite flexibilidad en cómo se presenta el contenido.

Especialización

- Los lenguajes de marcas se usan en muchas áreas, como:
 - **Gráficos vectoriales** (SVG)
 - **Sindicaciones de contenido** (RSS, Atom)
 - **Notación científica** (MathML)
 - **Interfaz de usuario** (XAML)
 - **Síntesis de voz** y otras aplicaciones especializadas.

Flexibilidad

- Los lenguajes de marcas son muy **flexibles**, ya que se pueden combinar con otros lenguajes dentro del mismo archivo. Por ejemplo, en una página HTML puedes incluir:
 - **Python**
 - **JavaScript**

- **PHP**

Esto permite que las páginas web sean dinámicas y respondan a la interacción del usuario.

4. Clasificación de tipos de marcas

1. Tipos de marcas según su propósito:

1.1. De procedimiento y presentación:

- Estas marcas indican el **formato de texto** o **tipografía** sin especificar la estructura del contenido.
- **Ejemplos:**
 - **nroff**
 - **troff**
 - **RTF**
 - **TeX**
 - **DocBook**
 - **MS Word**

1.2. Descriptivas, estructurales o semánticas:

- Indican las diferentes partes en las que se estructura un documento, pero no especifican cómo deben representarse ni en qué orden.
- **Ejemplos:**
 - **XML** y sus derivados:
 - **EBML**
 - **RDF**
 - **XFML**
 - **OWL**
 - **XTM**
 - **YAML**

1.3. Híbridas:

- Son una combinación de los tipos anteriores, que incluyen tanto aspectos de presentación como de estructura.
- **Ejemplos:**
 - **HTML**
 - **XHTML**
 - **WML**

4. Clasificación según la funcionalidad

Para crear documentación electrónica:

- **RTF**
- **TeX**
- **troff**
- **nroff**
- **ASN.1**
- **EBML**
- **YAML**
- **Wikitext**
- **DocBook**
- **LinuxDoc**

Tecnologías de internet:

- **HTML**
- **XHTML**
- **WML** (web)
- **GladeXML**
- **XForms**
- **XAML** (formularios/interfaz de usuario)
- **RSS**
- **Atom** (sindicador de contenido)
- **WSDL**
- **SOAP**
- **UDDI** (servicios web)
- **XMPP** (mensajería instantánea)

De propósito específico:

- **MathML** (matemáticas)
- **CML** (química)
- **SSML**
- **SRGS**
- **VoiceXML** (sintetización de voz)
- **MusicXML** (partituras de música)
- **SVG**
- **VML**
- **X3D** (gráficos vectoriales)
- **SMIL** (archivos multimedia)
- **XLL** (enlaces)
- **XLINK** (asociación de recursos)
- **XML Base** (URI básico)

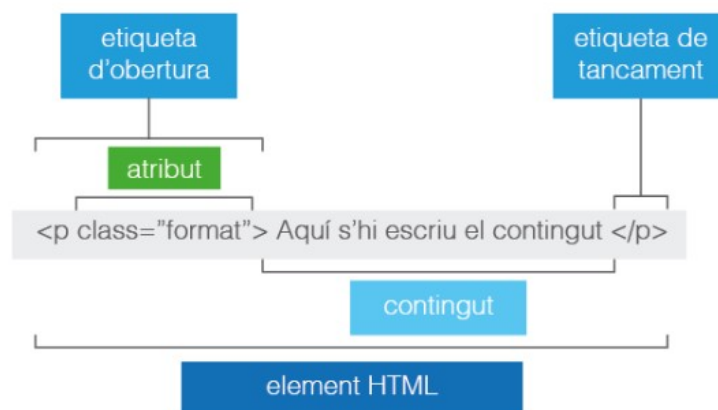
- **XPOINTER** (localización de recursos)
- **XSLT** (transformación de documentos)
- **XTM** (mapas de temas)
- **RDF**
- **XFML**
- **OWL**
- **XMP** (catalogación y clasificación de documentos)
- **GML** (información geográfica)
- **OFX** (intercambio de información financiera)
- **ebXML** (comercio electrónico)
- **XML Dsig**
- **XML Enc**
- **SAML**
- **XACML**
- **XKMS**
- **XrML** (seguridad)
- **XInclude** (inclusión de archivos)

5. Organizaciones y estándares

- **Estandarización:**
 - Se puede definir como el **proceso de especificación de normas** que garantizan el correcto funcionamiento de elementos contruidos de forma independiente.
- **Organismos que definen estas normas:**
 - Existen **organismos internacionales, nacionales** y también **organizaciones privadas** que se encargan de establecer estos estándares.
 - Las más importantes son:
 - **W3C** (World Wide Web Consortium)
 - **ISO** (International Organization for Standardization)
 - **Open Source**
- **W3C:**
 - El **World Wide Web Consortium (W3C)** es una comunidad internacional que desarrolla estándares para asegurar el crecimiento sostenible de la web a largo plazo.
 - Es imprescindible alcanzar **consenso** con las empresas involucradas, como **navegadores, buscadores, desarrolladores web y fabricantes de dispositivos móviles** (por ejemplo, **Microsoft, Apple, Google**, etc.).

6. Lenguaje de Marcado: Etiquetas, elementos y atributos

- **Etiqueta (tag):**
 - Una **etiqueta** es un texto que se encuentra entre los símbolos menor que y mayor que: < >.
 - Existen **etiquetas de apertura** y **etiquetas de cierre**.
 - Ejemplo: <Nombre> </Nombre>
- **Elementos:**
 - Los **elementos** representan estructuras que organizan el contenido del documento o desencadenan acciones cuando el navegador interpreta el documento.
 - Un elemento consta de:
 - **Etiqueta de apertura**
 - **Etiqueta de cierre**
 - **Contenido** que se encuentra entre ambas etiquetas.
 - Algunos elementos no tienen contenido y se denominan **elementos vacíos**. Estos no requieren etiqueta de cierre, es decir, son **etiquetas únicas**.
- **Atributo:**
 - Un **atributo** es un par **nombre-valor** que se encuentra dentro de la etiqueta de apertura de un elemento.
 - Los atributos indican las **propiedades** que pueden estar asociadas a los elementos.



```
<domicili>
...
  <propietari>
    <tractament>Mrs.</tractament>
    <nom>Mary</nom>
    <cognoms>McGoon</cognoms>
  </propietari>
  <carrer>1401 Main Street</carrer>
  <ciutat estat="NC">Anytown</ciutat>
  <codi_postal>34829</codi_postal>
</domicili>
```


7. Principales lenguajes de marcas

HTML (HyperText Markup Language)

- **Definición:** Es un lenguaje diseñado para la creación de páginas web, que combina texto y contenido multimedia.
- **Enlaces:** Permite crear hipervínculos para conectar con otras páginas relacionadas.
- **Historia:** Creado a principios de los años 90 para un uso divulgativo, ha evolucionado de manera anárquica.
- **Desafíos:** Diferentes navegadores implementan el estándar de manera independiente, provocando discrepancias en la representación del contenido.
- **Herramientas:** Aunque se pueden crear documentos HTML con editores de texto, también existen aplicaciones específicas que facilitan la creación de código.

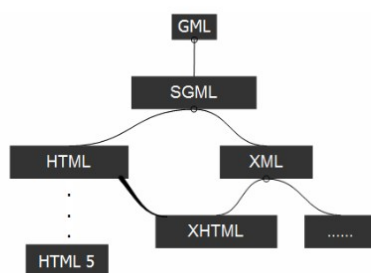
7.1 Evolución del lenguaje HTML

- **HTML1:** La primera versión, creada por Tim Berners-Lee en 1991.
- **HTML2:** Aparece entre 1994 y 1996, sin grandes cambios significativos.
- **HTML3.0:** Establece las bases para las versiones futuras, añadiendo nuevas funcionalidades como tablas y scripts (1996).
- **HTML4:** La versión más común (HTML4.01), lanzada en 1998, que introduce marcos y compatibilidad con CSS.
- **HTML5:** La última versión (completada en 2014), que mejora la inclusión de vídeos y nuevas funcionalidades para formularios.

8. Principales lenguajes de marcas. XML

- **Definición:** Simplificación del SGML que permite definir lenguajes específicos (XHTML, SVG, MathML, RSS, entre otros).
- **Extensibilidad:** Permite la creación de nuevas etiquetas.
- **Versatilidad:** Separa contenido, estructura y presentación.
- **Estructuración:** Permite modelar datos a cualquier nivel de complejidad.
- **Validación:** Los documentos se pueden validar frente a un DTD/Schema.
- **Apertura:** Independiente de empresas, sistemas operativos o lenguajes de programación.
- **Sencillez:** Fácil de aprender y utilizar.

Usos: XML se utiliza no solo en Internet, sino como un estándar para el intercambio de información estructurada entre plataformas, bases de datos ligeras, editores de texto y transacciones comerciales.



9. Principales lenguajes de marcas. XHTML

- **Definición:** XHTML (eXtensible HyperText Markup Language) es la versión en formato XML del lenguaje HTML, recomendada por W3C en 2003.
- **Características:** Es más estricta que HTML; todas las etiquetas deben cerrarse. Por ejemplo, `
` se convierte en `
`.
- **Procesamiento:** Con XHTML, es posible crear páginas web con la misma funcionalidad que HTML, pero garantizando una estructura y sintaxis correcta.
- **Extensión:** Normalmente utiliza la extensión `.xhtml`.

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Ejemplo</title>
</head>
```

10. Principales lenguajes de marcas. JSON

- **Definición:** JSON (JavaScript Object Notation) es un formato que utiliza JavaScript para representar sus arrays.
- **Uso:** Se ha convertido en una alternativa universal a XML para el almacenamiento e intercambio de datos.
- **Compatibilidad:** Casi todos los lenguajes de programación cuentan con analizadores para JSON.
- **Características:**
 - Más ligero que XML.
 - Tiene una sintaxis más sencilla, aunque menos rica en comparación con XML.
- **Extensión:** Los archivos suelen tener la extensión `.json`.

11. Principales lenguajes de marcas. RSS y Atom

- **Definición:** RSS (Really Simple Syndication) y Atom son dialectos de XML.
- **Función de RSS:** Se utilizan para la distribución de contenidos a los que los usuarios se han suscrito.
- **Alternativa:** Atom se presenta como una alternativa a RSS.
- **Ejemplo:** Documentos en formato RSS de AEMET (Agencia Estatal de Meteorología).

12. Evolución del lenguaje CSS

Aunque CSS (Cascading Style Sheets) no se considera un lenguaje de marcas, sino un lenguaje de diseño, es importante introducirlo para dar forma a nuestras prácticas. Las versiones existentes son las siguientes:

- **CSS1:** Apareció en 1996. Establece las bases que permiten mejoras de presentación en las páginas web, como colores, márgenes, fuentes, entre otros.
- **CSS2:** Lanzado en 1999 y completado con CSS 2.1, esta versión agrega numerosas opciones. Permite técnicas de posicionamiento muy precisas para colocar elementos en el lugar deseado en la página.
- **CSS3:** Es la última versión y agrega características muy esperadas, como bordes redondeados, degradados, sombras, entre otros. Después de comprender HTML, nos centraremos en los estilos con CSS3.

13. Herramientas de edición

A continuación, se presentan herramientas para la edición de páginas web y creación de documentos XML. Al igual que para todo tipo de software, hay herramientas en línea y fuera de línea que deben descargarse e instalarse.

- **Bloc de notas:** Editor de texto que permite crear documentos de texto plano. Usa la extensión `.txt` y permite codificación ASCII, Unicode y UTF-8.
- **Gedit:** Editor de texto para el entorno GNOME. Se utiliza comúnmente como bloc de notas y entorno de desarrollo.
- **Notepad++:** Editor de texto potente con licencia GPL. Está escrito en C++. Permite seleccionar múltiples lenguajes de programación y ofrece resaltado de sintaxis y numeración de líneas.
- **XML Copy Editor:** Editor de XML que permite crear y comprobar de manera sencilla si el documento está bien formado.
- **Atom:** Editor de texto de código abierto para macOS, Linux y Windows, con soporte para múltiples complementos escritos en Node.js (que amplían sus capacidades) y control de versiones Git integrado, que puede utilizarse como IDE para una gran variedad de lenguajes de programación.