



Universidad Tecnológica de Tijuana

Lenguaje de Mercado Y Presentación

Materia: Desarrollo de Aplicaciones Web.

Alumno: Martin Rojas Carlos Aiel.

Grupo: 3ro C.

Maestro: Ray Brunette Parra Galaviz.

Materia :Base de Datos para Aplicaciones Web.

Fecha de Entrega:29 de Agosto del 2024.

Lenguajes de Mercado	3
Introducción	3
1. Definición y Propósito	4
2. Tipos de Lenguajes de Mercado	4
a. Lenguajes de Mercado de Presentación	4
b. Lenguajes de Mercado Semántico	4
c. Lenguajes de Mercado de Transformación	5
3. Características Comunes	5
4. Aplicaciones Comunes	5
5. Ventajas de los Lenguajes de Mercado	6
6. Desafíos y Consideraciones	6
7. Futuro de los Lenguajes de Mercado	6
Lenguajes de Presentacion	6
1. Definición y Propósito	6
2. Principales Lenguajes de Presentación	6
a. CSS (Cascading Style Sheets)	7
b. XSL-FO (Extensible Stylesheet Language Formatting Objects)	7
c. SVG (Scalable Vector Graphics)	8
d. PDF (Portable Document Format)	8
e. PostScript	8
3. Características Comunes	9
4. Aplicaciones Comunes	9
5. Ventajas de los Lenguajes de Presentación	9
6. Desafíos y Consideraciones	9

Lenguajes de Marcado

Introducción

Los lenguajes de marcado, también denominados lenguajes de marcas, provienen y toman su nombre de la práctica tradicional e historia de marcar los manuscritos con instrucciones de impresión en los márgenes. Esta área, tan habitual en la industria editorial desde la aparición de la imprenta, fue dando lugar a un grupo de marcas estandarizadas, cuya esencia ha sido trasladada hoy al mundo de la informática.

Actualmente se distinguen dos tipos fundamentales de lenguajes de marcado en la práctica pueden combinarse.

- LENGUAJES DE MARCADO DE PROCEDIMIENTO, orientados hacia la presentación del texto. Sus símbolos o marcas indican la clase de operaciones tipográficas que deben ser aplicadas a cada uno de los elementos del documento electrónico, para dar formato al texto. Su misión es, pues, configurar la apariencia física de éste (fuente, estilo, tamaño de letra, etc.), tanto en pantalla como impreso.
- LENGUAJES DE MARCADO DESCRIPTIVO, diseñados para identificar las piezas o bloques estructurales que componen el texto. Sus marcas determinan la estructura lógica del documento electrónico y/o la descripción de su contenido, no ya su tipografía ni el formato que presentará cada fragmento en su posterior visualización.

Así pues, en el contexto automatizado actual, los lenguajes de marcado permiten codificar documentos intercalando, junto al texto, etiquetas o marcas provistas de información adicional sobre su estructura o presentación. En realidad, más que de lenguajes, podría hablarse de metalenguajes o conjuntos de reglas que se encargan de establecer y definir la forma digital de los documentos, bien para controlar su procesamiento, bien para representar su significado.

1. Definición y Propósito

Los lenguajes de marcado utilizan etiquetas o "marcas" que se insertan en el texto para indicar cómo debe ser formateado, estructurado o presentado. Estas etiquetas son elementos que rodean el contenido y proporcionan información adicional que no es visible para el usuario final, pero que los navegadores web, editores de texto, o aplicaciones pueden interpretar para renderizar el contenido de manera específica.

2. Tipos de Lenguajes de Marcado

a. Lenguajes de Marcado de Presentación

Estos lenguajes se centran en cómo se presenta el contenido, es decir, su apariencia visual. Los más conocidos son:

- **HTML (Hypertext Markup Language):** Es el lenguaje de marcado estándar para crear páginas web. Utiliza etiquetas para estructurar contenido como encabezados, párrafos, enlaces, imágenes, y otros elementos multimedia. HTML define la estructura básica de una página web y trabaja en conjunto con CSS y JavaScript para crear páginas web interactivas y visualmente atractivas.
- **XML (eXtensible Markup Language):** Es un lenguaje de marcado que define reglas para codificar documentos en un formato que es legible tanto para humanos como para máquinas. A diferencia de HTML, XML no tiene etiquetas predefinidas; en cambio, permite que los usuarios creen sus propias etiquetas, lo que lo hace extensible y flexible para diferentes tipos de datos.
- **XHTML (eXtensible Hypertext Markup Language):** Es una versión más estricta de HTML que sigue las reglas de XML. Aunque similar a HTML, XHTML requiere que todas las etiquetas sean cerradas y que el código esté bien formado.

b. Lenguajes de Marcado Semántico

Estos lenguajes no solo definen la presentación del contenido, sino que también proporcionan información sobre su significado.

- **Markdown:** Es un lenguaje de marcado ligero que se utiliza para formatear texto simple en un formato más legible. Es popular en plataformas como GitHub y se utiliza para escribir documentación, artículos, y otros textos que necesitan ser convertidos a HTML u otros formatos.
- **YAML (YAML Ain't Markup Language):** Aunque no es estrictamente un lenguaje de marcado, es un formato de serialización de datos legible por humanos, comúnmente utilizado en archivos de configuración. Es más fácil de leer y escribir que JSON o XML.
- **LaTeX:** Es un lenguaje de marcado utilizado para la composición de textos científicos y matemáticos. Es muy potente para manejar ecuaciones complejas, referencias cruzadas, y bibliografías.

c. Lenguajes de Marcado de Transformación

Estos lenguajes se utilizan para transformar un documento de un formato a otro.

- **XSLT (eXtensible Stylesheet Language Transformations):** Es un lenguaje basado en XML utilizado para transformar documentos XML en otros formatos, como HTML, texto plano o incluso otros documentos XML.

3. Características Comunes

- **Etiquetas:** Los lenguajes de marcado utilizan etiquetas que indican cómo debe interpretarse el texto entre ellas. Las etiquetas suelen ser pares de una etiqueta de apertura y una de cierre, aunque algunos lenguajes también tienen etiquetas de auto-cierre.
- **Atributos:** Muchas etiquetas pueden incluir atributos, que proporcionan información adicional sobre el contenido de la etiqueta. Por ejemplo, en HTML, una etiqueta de imagen (``) puede tener un atributo `src` que indica la fuente de la imagen.
- **Anidación:** Las etiquetas en los lenguajes de marcado suelen poder anidarse, es decir, colocarse unas dentro de otras para definir estructuras complejas.
- **Extensibilidad:** Algunos lenguajes de marcado, como XML, permiten a los usuarios definir sus propias etiquetas, lo que les da una gran flexibilidad para adaptarse a diferentes necesidades y aplicaciones.

4. Aplicaciones Comunes

- **Web:** HTML es la base de la web, proporcionando la estructura y contenido de todas las páginas web. CSS se utiliza para definir la presentación de esas páginas, mientras que JavaScript añade interactividad.
- **Documentación Técnica:** Lenguajes como Markdown y LaTeX son ampliamente utilizados en la creación de documentación técnica, artículos académicos, y libros debido a su capacidad para manejar contenido técnico de manera eficiente.
- **Configuración de Software:** YAML y JSON son comunes en archivos de configuración para software, debido a su simplicidad y legibilidad.

- **Publicación y Tipografía:** LaTeX es el estándar en la publicación académica, especialmente en campos que requieren fórmulas matemáticas complejas.

5. Ventajas de los Lenguajes de Marcado

- **Flexibilidad:** Permiten la creación de documentos que pueden ser fácilmente interpretados y procesados por máquinas.
- **Compatibilidad:** Los documentos creados con lenguajes de marcado pueden ser utilizados en diferentes plataformas sin necesidad de cambios significativos.
- **Simplicidad:** Algunos lenguajes, como Markdown, son fáciles de aprender y utilizar, lo que los hace accesibles incluso para usuarios sin conocimientos técnicos avanzados.

6. Desafíos y Consideraciones

- **Curva de Aprendizaje:** Algunos lenguajes de marcado, como LaTeX, tienen una curva de aprendizaje más pronunciada debido a su complejidad.
- **Compatibilidad:** La compatibilidad entre diferentes navegadores y dispositivos puede ser un desafío, especialmente en el caso de HTML y CSS.
- **Mantenimiento:** A medida que los documentos se vuelven más complejos, mantener un código de marcado limpio y bien estructurado puede requerir esfuerzo adicional.

7. Futuro de los Lenguajes de Marcado

El uso de lenguajes de marcado sigue evolucionando, especialmente con el crecimiento de tecnologías web y la necesidad de manejar datos estructurados. Los lenguajes de marcado como HTML y XML seguirán siendo fundamentales en la tecnología, mientras que lenguajes más nuevos y específicos continuarán emergiendo para satisfacer las necesidades cambiantes de la industria.

En resumen, los lenguajes de marcado son herramientas esenciales para la estructura, presentación y transformación de información, con aplicaciones que abarcan desde la web hasta la publicación académica y la configuración de software.

Lenguajes de Presentación

Los lenguajes de presentación son un subconjunto de los lenguajes de marcado que se centran en la apariencia visual y la disposición del contenido en diferentes medios, como páginas web, documentos impresos, y más. A continuación, te ofrezco un resumen extenso sobre ellos:

1. Definición y Propósito

Los lenguajes de presentación están diseñados para controlar cómo se muestra el contenido a los usuarios finales. A diferencia de otros lenguajes de marcado que pueden centrarse en la estructura o el significado semántico del contenido, los lenguajes de presentación se preocupan principalmente por

aspectos visuales, como la disposición de los elementos en una página, el estilo del texto, los colores, y otros aspectos estéticos.

2. Principales Lenguajes de Presentación

a. CSS (Cascading Style Sheets)

- **Descripción:** CSS es el lenguaje de estilo más utilizado en la web. Su propósito es describir cómo debe ser presentado el contenido de un documento HTML o XML. CSS permite a los diseñadores y desarrolladores web controlar la disposición, colores, fuentes, márgenes, bordes, espaciado, y otros aspectos visuales de los elementos en una página web.
- **Características:**
 - **Cascada:** Las reglas CSS siguen un orden de prioridad que permite aplicar estilos desde múltiples fuentes. Las hojas de estilo externas, internas y en línea pueden combinarse y superponerse para determinar la apariencia final.
 - **Selectores:** CSS utiliza selectores para aplicar estilos a elementos específicos o grupos de elementos en un documento.
 - **Diseño Responsivo:** CSS es fundamental para crear sitios web que se adapten a diferentes tamaños de pantalla, desde dispositivos móviles hasta monitores de escritorio.

Ejemplo:

css

Copiar código

```
body {  
    font-family: Arial, sans-serif;  
    color: #333;  
    margin: 0;  
    padding: 0;  
}  
  
h1 {  
    color: #4CAF50;  
    text-align: center;  
}
```

b. XSL-FO (Extensible Stylesheet Language Formatting Objects)

- **Descripción:** XSL-FO es un lenguaje de presentación utilizado principalmente para formatear documentos XML en formatos de salida como PDF o impresos. Es parte de la familia de lenguajes XSL y se utiliza para especificar la disposición de texto, imágenes y otros contenidos en una página.
- **Características:**
 - **Diseño Preciso:** XSL-FO permite un control detallado sobre la presentación, como la especificación de márgenes, columnas, encabezados, pies de página y más.

- **Transformación de Documentos:** Se suele utilizar junto con XSLT para transformar documentos XML en documentos que pueden ser formateados con XSL-FO.

Ejemplo:

xml

Copiar código

```
<fo:block font-family="Arial" font-size="12pt" color="#333">
  This is a block of text formatted with XSL-FO.
</fo:block>
```

•

c. SVG (Scalable Vector Graphics)

Descripción: SVG es un lenguaje de marcado basado en XML que describe gráficos vectoriales bidimensionales. A diferencia de las imágenes rasterizadas, los gráficos SVG pueden escalarse sin perder calidad, lo que los hace ideales para la web.

Características:

- **Independencia de Resolución:** Los gráficos SVG se pueden escalar a cualquier tamaño sin pérdida de calidad, lo que es útil para pantallas de alta resolución.
- **Interactividad:** SVG permite incorporar interactividad, como enlaces y animaciones, directamente en los gráficos.
- **Compatibilidad con CSS y JavaScript:** Los elementos SVG pueden ser estilizados con CSS y manipulados dinámicamente con JavaScript.

Ejemplo:

```
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40" stroke="black" stroke-width="3"
  fill="red" />
</svg>
```

d. PDF (Portable Document Format)

Descripción: Aunque no es un lenguaje de marcado en sí, el formato PDF es un formato de presentación que conserva el diseño y formato de un documento tal como fue creado, independientemente del dispositivo o software utilizado para verlo. PDF combina texto, gráficos, y otros elementos visuales en un formato de archivo fijo que es ideal para documentos que necesitan ser impresos o compartidos sin cambios.

Características:

- **Consistencia en la Presentación:** Los documentos PDF se ven igual en cualquier dispositivo o impresora, lo que asegura que el contenido y la disposición no cambien.
- **Incorporación de Medios:** Los PDFs pueden incluir enlaces, botones, formularios interactivos, imágenes, y multimedia como videos.

e. PostScript

Descripción: PostScript es un lenguaje de programación orientado a la descripción de páginas, utilizado principalmente en impresión. Fue uno de los primeros lenguajes en ofrecer la capacidad de describir con precisión la disposición de texto e imágenes en una página impresa.

Características:

- **Control Preciso de la Presentación:** PostScript permite especificar con precisión la colocación de cada elemento en la página.
- **Escalabilidad:** Como los gráficos en PostScript son vectoriales, pueden ser escalados sin perder calidad.
- **Independencia de Dispositivo:** PostScript fue diseñado para ser independiente de cualquier dispositivo específico, lo que significa que los documentos se pueden imprimir en cualquier impresora compatible sin necesidad de ajustes.

3. Características Comunes

- **Control Visual:** Los lenguajes de presentación permiten a los diseñadores y desarrolladores especificar cómo se debe mostrar el contenido en diferentes medios, ya sea en pantalla o impreso.
- **Independencia de Contenido:** A menudo, el contenido (como el texto o datos) está separado de las reglas de presentación, permitiendo una mayor flexibilidad y reutilización del contenido en diferentes contextos.
- **Escalabilidad y Responsividad:** Muchos lenguajes de presentación permiten crear diseños que se adaptan a diferentes tamaños de pantalla y resoluciones, asegurando una experiencia de usuario consistente en dispositivos variados.

4. Aplicaciones Comunes

- **Diseño Web:** CSS es fundamental en el desarrollo web para estilizar y dar forma a las páginas HTML. Es esencial para la creación de sitios web modernos y adaptables.
- **Publicación Digital:** Formatos como PDF y XSL-FO se utilizan ampliamente en la publicación de documentos electrónicos que necesitan mantener su formato original, como libros, informes, y formularios.
- **Impresión Profesional:** PostScript y PDF son estándares en la industria de la impresión, permitiendo la creación de documentos que se pueden reproducir con alta fidelidad en cualquier impresora.

5. Ventajas de los Lenguajes de Presentación

- **Separación de Contenido y Estilo:** Al separar la estructura del contenido de su presentación, se facilita la gestión y actualización del diseño sin necesidad de modificar el contenido.
- **Precisión y Consistencia:** Los lenguajes de presentación permiten un control detallado sobre el aspecto final de un documento o página, asegurando consistencia en diferentes plataformas y dispositivos.

- **Flexibilidad y Reutilización:** Permiten reutilizar estilos en diferentes documentos o páginas, lo que reduce el esfuerzo y tiempo de desarrollo.

6. Desafíos y Consideraciones

- **Compatibilidad de Navegadores:** En el caso de CSS, las diferencias en la implementación entre distintos navegadores pueden crear desafíos para garantizar una presentación consistente.
- **Complejidad en Diseño:** A medida que los diseños se vuelven más complejos, la gestión de las reglas de presentación puede ser complicada y propensa a errores.
- **Rendimiento:** El uso intensivo de estilos o gráficos complejos puede afectar el rendimiento, especialmente en dispositivos más antiguos o con recursos limitados.