

Elementos de Programación y Lógica.
Unidad 1 - Clase 3.

Lenguajes de Mercado.

HTML.

.

Sección 1. HTML.

Parte 1. El lenguaje.

HTML.

HyperText Markup Language (HTML) es un lenguaje de marcado que sirve para describir páginas web.

Los archivos HTML son archivos de texto, que pueden editarse con cualquier editor de texto, y visualizarse con un **Browser**.

Browser.

El **Browser** o **Navegador web** permite visualizar archivos HTML que se encuentran en el equipo, así como también visualizar archivos HTML que se encuentran en otros equipos (Internet).

Al solicitar un sitio web en el browser, este realiza distintos mecanismos para determinar en que equipo se encuentra el archivo HTML a visualizar, lo descarga, y se lo presenta al usuario.

Hay muchos browsers (algunos más populares que otros) de distintas compañías. Internet Explorer o Edge de Microsoft, Chrome de Google, Safari de Apple, Firefox de la Fundación Mozilla, Opera de Opera Software, son algunos de los más conocidos.

Descripción de imagen: Logos de los navegadores más populares

Etiquetas HTML.

Una etiqueta HTML tiene un nombre que va dentro de los signos de etiqueta. La mayoría de las etiquetas deben abrir y cerrar, las etiquetas de apertura llevan los signos de “<” (menor) y “>” (mayor) para indicar el nombre, mientras que las de cierre llevan “</” (menor seguido de barra) y “>” (mayor).

Ejemplo apertura:

```
1 <nombre_de_etiqueta>
```

Ejemplo cierre

```
1 </nombre_de_etiqueta>
```

Contenido HTML.

El contenido de una etiqueta se coloca entre la etiqueta de apertura y la de cierre.

Ejemplo:

```
1 <nombre_etiqueta> Contenido </nombre_etiqueta>
```

Etiquetas anidadas en HTML.

Una etiqueta puede contener otras etiquetas dentro, como contenido.
Ejemplo:

```
1 <etiqueta1>
2   <etiqueta2>
3     Contenido de la etiqueta
4   </etiqueta2>
5 </etiqueta1>
```

Atributos.

Algunas etiquetas llevan atributos, valores que indican cosas acerca de dicha etiqueta.

Un atributo se indica dentro de las marcas de etiquetas, luego del nombre de la etiqueta, indicando el nombre del atributo, el signo igual y el valor asociado de dicho atributo entre comillas.

Ejemplo:

```
1 <nombre_de_etiqueta atributo="valor">
```

Atributos Cont..

Los atributos se separan con espacios, y puede haber tantos como se quiera.
Ejemplo:

```
1 <nombre attr1="val1" attr2="val2">
```

Parte 2. Comunicación del código.

Comunicación e Indentación.

El código

```
1 <etiqueta1><etiqueta2>Contenido de la etiqueta
2 </etiqueta2></etiqueta1>
```

Es equivalente a

```
1 <etiqueta1>
2   <etiqueta2>
3     Contenido de la etiqueta
4   </etiqueta2>
5 </etiqueta1>
```

Indentación.

Sin embargo, el segundo se lee mejor.
Esto no es casual, vamos a elegir siempre indentar correctamente el código.
La indentación es el tabulado, o sangría que se coloca en el código, de forma tal de hacerlo más legible. También implica los saltos de línea de forma correcta.

```
1 <etiqueta1>
2   <etiqueta2>
3     Contenido de la etiqueta
4   </etiqueta2>
5 </etiqueta1>
```

Comentarios.

Los comentarios permiten agregar información que ayudan a entender el código del sitio web, pero que es completamente ignorado por el navegador.
Un comentario es similar a una etiqueta, pero comienza con “<!--” y termina con “-->”.

```
1 <!-- Esto es un comentario,
2     puede ponerse lo que se desee aquí -->
```

Parte 3. Etiquetas del HTML

Etiquetas básicas.

Veamos algunos ejemplos básicos de etiquetas:

- **p**: Delimita un párrafo
- **strong**: Indica que una palabra o frase debe estar remarcada, en negrita.
- **em** Indica que una palabra o frase debe estar enfatizada, en itálica.
- **section**: Delimita un sección, probablemente conteniendo varios párrafos. Cumple solo una función organizacional, pero no visual.

Etiquetas de títulos.

También podemos generar títulos:

- **h1**: Título
- **h2**: Subtitulo
- **h3**: Título de 3er nivel
- **h4**: Título de 4to nivel

- **h5**: Título de 5to nivel
- **h6**: Título de 6to nivel

Etiquetas de listas.

Se pueden crear listas:

- **ul**: Lista sin orden (con viñetas). Requiere ítems adentro.
- **ol**: Lista ordenada (numerada). Requiere ítems adentro.
- **li**: Item de una lista

Etiquetas de enlace e imagen.

O enlaces e imágenes:

- **a**: Un enlace. Requiere un atributo obligatorio:
 - **href**: Indica a donde se esa enlazando.
- **img**: Una imagen. Requiere un atributo obligatorio:
 - **src**: Indica donde se ubica la imagen que se desea mostrar

Y otros opcionales:

- **width**: Indica el ancho de la imagen
- **height**: Indica el alto de la imagen

Etiquetas de tablas.

Se pueden crear tablas de datos (datos de dos entradas)

- **table**: Una tabla, requiere que se declare dentro el contenido.
- **tr**: Genera una fila en la tabla. Requiere celdas dentro.
- **td**: Genera una celda en la tabla, requiere contenido dentro.

Parte 4. Estructura de un documento

Estructura básica de documento.

Todo documento HTML comienza siempre con una pseudo-etiqueta que tiene por finalidad indicarle al navegador que se trata de un documento HTML (Otros lenguajes usan etiquetas similares para describir cosas distintas a páginas web). La pseudo-etiqueta debe ser en todos los casos la siguiente:

1 `<!DOCTYPE html>`

Estructura básica de documento - Cont..

Luego de la pseudo-etiqueta debe haber una única etiqueta que englobe todos los elementos siguientes.

Esta etiqueta tiene por nombre **html** y debe ser la única etiqueta externa.

La etiqueta html debe contener a su vez dos sub-etiquetas, **head** y **body**.

Estructura básica de documento - Cont..

Dentro de la etiqueta head se agregará meta-información que el browser puede utilizar para saber como mostrar el contenido, como el título que debe poner en la ventana, o el ícono a mostrar en la pestaña, o el lenguaje en el que se espera se encuentre la página.

La etiqueta body en cambio engloba a todos los elementos que representa contenido que va a ser mostrado en la página web. Si es algo visible en el sitio va dentro del body.

Estructura básica de documento - Cont..

Como vamos a escribir en español nos va a interesar agregar una de las etiquetas de metadatos dentro de head que va a indicarle que el archivo contiene caracteres en español. La etiqueta **meta** con el atributo **charset**.

Así que la estructura general del documento nos va a quedar así:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <!-- Otros metadatos relevantes -->
6   </head>
7   <body>
8     <!-- Aquí va el contenido del sitio -->
9   </body>
10 </html>
```

Parte 5. Ejemplos

Ejemplos de HTML - 1.

```
1 <!DOCTYPE html>
2 <html>
3   <head> <meta charset="utf-8"> </head>
4   <body>
5     <h1>Titulo principal</h1>
6     <p>
7       Este es el primer párrafo. En el se
8       puede agregar <strong>negritas</strong>
9       o <em>itálicas</em>.
```

```
10     </p>
11     <p>
12         Podemos agregar otros párrafos
13     </p>
14 </body>
15 </html>
```

Ejemplos de HTML - 1 Cont..

Descripción de imagen: Resultado del ejemplo anterior

Ejemplos de HTML - 2.

```
1 <!DOCTYPE html>
2 <html>
3   <head> <meta charset="utf-8"> </head>
4   <body>
5     <h1>Enlaces y subtítulos</h1>
6     <p>
7       También podemos agregar enlaces como:
8       <a href="http://google.com">Ir a Google</a>
9     </p>
10    <h2>0 subtítulos</h2>
11    <p>
12      Y luego comenzar en nuevo párrafo.
13    </p>
14  </body>
15 </html>
```

Ejemplos de HTML - 2 Cont..

Descripción de imagen: Resultado del ejemplo anterior

Ejemplos de HTML - 3.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5   </head>
```

```

6   <body>
7     <h1>Listas</h1>
8     <h3>Una lista numerada:</h3>
9     <ol>
10      <li>El primer elemento</li>
11      <li>El segundo elemento</li>
12      <li>El tercero</li>
13    </ol>

```

Ejemplos de HTML - 3 - Cont..

```

1     <h3>Una lista con viñetas:</h3>
2     <ul>
3       <li>El primer elemento</li>
4       <li>El segundo elemento</li>
5       <li>El tercero</li>
6     </ul>
7   </body>
8 </html>

```

Ejemplos de HTML - 3 Cont..

Descripción de imagen: Resultado del ejemplo anterior

Ejemplos de HTML - 4.

```

1 <!DOCTYPE html>
2 <html>
3   <head> <meta charset="utf-8"> </head>
4   <body>
5     <h1>Una tabla de ejemplo</h1>
6     <table>
7       <tr>
8         <th>Nombre</th>
9         <th>Apellido</th>
10        <th>Edad</th>
11      </tr> <tr>
12        <td>Juan</td>
13        <td>Perez</td>
14        <td>25</td>

```


Ejemplos de HTML - 4 Cont..

```
1      </tr> <tr>
2          <td>Cosme</td>
3          <td>Fulanito</td>
4          <td>23</td>
5      </tr> <tr>
6          <td>Susana</td>
7          <td>Mengano</td>
8          <td>28</td>
9      </tr>
10 </table>
11 </body>
12 </html>
```

Ejemplos de HTML - 4 Cont..

Descripción de imagen: Resultado del ejemplo anterior

Parte 6. Etiquetas de estructura

Etiquetas de estructura.

Unas de las etiquetas más importantes son las etiquetas que indican estructura.

Estas etiquetas no cumplen una función visual, sino que simplemente indican secciones importantes del sitio, agrupando los contenidos de forma semántica. El poder agrupar semánticamente los contenidos permite no solo comprender mejor el código, sino también que servicios como Google u otros comprendan mejor el contenido del sitio e indexarlo correctamente.

Etiquetas de estructura.

- **header**: Indica la cabecera del sitio (logo de la página con el nombre, otros elementos que suelen ser constante en los diseños de los sitios). Pueden haber headers dentro de una sección o artículo, en casos muy raros. NO DEBE SER CONFUNDIDO CON HEAD.
- **nav**: Indica una barra de navegación que contiene links a distintas secciones del sitio. Suele estar dentro del header.
- **footer**: El footer es el pie de página, suele indicar cosas como el nombre del autor, información de contacto, información de copyright, etc.

- **article**: Se puede usar en sitios de noticias y blogs para englobar la nota y otros contenidos.
- **section**: Indica una sección del sitio, por ejemplo, dentro de una nota, puede haber varias secciones.

Etiquetas de estructura.

- **section**: Indica una sección del sitio, por ejemplo, dentro de una nota, puede haber varias secciones.
- **aside**: Suele usarse para contenidos secundarios, que no son parte prioritaria del contenido del sitio. Por ejemplo, cosas que van en las barras laterales de un sitio, incluyendo publicidades, calendarios, etc.
- **div**: Aquellas agrupaciones que no caen en ninguna de las otras categorías o que se usan por cuestiones semánticas o de estilo.

Etiquetas de estructura - Ejemplo.

Ej. La información en un sitio web de noticias:

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5 </head>
6 <body>
7 <section>
8   <h2>Noticias destacadas del día</h2>
9   <article>
10    <h4>Los liberales se levantan en armas</h4>

```

Etiquetas de estructura - Ejemplo.

```

1   <p>
2     El coronel <strong>Aureliano
3     Buendía</strong> habría marchado
4     a la guerra al grito de <em>
5     Viva el partido liberal</em>.
6     Se esperan fuertes repercusiones
7     en todo Macondo.
8   </p>
9 </article>

```

Etiquetas de estructura - Ejemplo Cont..

```
1  <article>
2    <h4>Se desata la guerra contra Eurasia</h4>
3    <p>
4      La guerra contra <em>Eurasia</em> ha
5      comenzado y soldados de toda Oceanía
6      marchan hacia la frontera para defender
7      la nación.
8    </p>
9  </article>
10 </section>
11 </body>
12 </html>
```

Etiquetas de estructura - Ejemplo Cont..

Descripción de imagen: Resultado del ejemplo anterior

Sección 2. Uso de URIs

URI: Referencias absolutas.

Supongamos el siguiente ejemplo:

```

```

Si enviamos nuestro archivo HTML a otra persona, y este intenta abrir el archivo, no verá la imagen, pues no la tiene en su equipo.

Peor aún, la ruta completa probablemente no exista en la máquina de nadie más que Juancito.

El problema es que se están usando URIs absolutas que referencian a un archivo local.

URI: Referencias web.

Ahora el siguiente ejemplo:

```

```

Si enviamos nuestro archivo HTML a otra persona, este verá la imagen, siempre y cuando tenga conexión a internet (http y https se usan para recursos web).

No está bueno si no sabemos con seguridad que la otra persona tiene conexión.

URI: Rutas locales.

Un sitio web de verdad está compuesto, no solo de su HTML, sino de todos los archivos que lo acompañan, por ejemplo, imágenes, videos, etc.

Generalmente uno recurre a colocar todo dentro de una carpeta, con subcarpetas de ser necesario. Una vez hecho esto, uno puede referenciar a los archivos de esa carpeta con URIs relativas.

URI: Rutas locales.

Consideremos la siguiente estructura:

Descripción de imagen: Estructuras de carpetas del proyecto HTML

Desde el archivo sitio.html ahora podemos hacer referencia a las imágenes del sitio refiriéndonos siempre al camino que hay que seguir desde la carpeta principal del proyecto.

```

```

```

```

TIP Entregas.

Cuando debamos enviar un sitio web que contenga recursos adicionales al HTML, siempre debemos utilizar rutas relativas y enviar toda la carpeta completa.

Cuando no se tengan archivos adicionales, basta con el archivo HTML.

Sección 3. Relación con Markdown

Relación con Markdown.

Para ser visualizado, los visualizadores de Markdown transforman el archivo en un archivo HTML, para luego mostrar el mismo.

Así se puede transformar un código de markdown a un código en HTML.

Relación con Markdown - Ejemplo.

```
1 # Título
2 ## Subtitulo
3
4 Esto es texto simple
5 y aquí hay un [enlace](http://google.com)
6
7 * Elemento 1
8 * Elemento 2
9 * Elemento 3
```

Relación con Markdown - Ejemplo Cont.

```
1 <html>
2   <head> <meta charset="utf-8"> </head>
3   <body>
4     <h1>Titulo</h1>   <h2>Subtitulo</h2>
5     <p> Esto es texto <strong>simple</strong>
6       y aquí hay un <a href="http://google.com">
7         enlace</a></p>
8     <ul>
9       <li>Elemento 1</li>
10      <li>Elemento 2</li>
11      <li>Elemento 3</li>
12    </ul>
13  </body>
14</html>
```

Sección 4. Estándar

W3C.

La w3c (**World Wide Web Consortium**) es un consorcio internacional creado por **Tim Berners-Lee**, creador de la URL, el HTTP y el HTML.

El consorcio comienza con la participación del MIT (Massachusetts Institute of Technology) al que luego se unió el INRIA (Institut National de Recherche en Informatique et en Automatique) de Francia, el ERCIM (European Research Consortium for Informatics and Mathematics) y la Universidad de Keiō en Japón.

La organización cuenta con miembros especialistas e investigadores en ciencias de la computación. Está encargada de definir los estándares de diferentes lenguajes y tecnologías: HTML, XML, SVG, SOAP, RDP, entre otras.

Estándar HTML.

El lenguaje HTML está en su versión 5 (HTML5), pero existen versiones anteriores.

Cada nueva versión es compatible con las anteriores en mayor parte. Algunas etiquetas desaparecen, porque se vuelven obsoletas, y otras nuevas que resultan necesarias aparecen.

El estándar HTML5 es un estándar vivo, es decir que no está en su versión final, sino que constantemente se está cambiando.

Estándar HTML - Cont..

Un archivo HTML correcto debería cumplir con todos los estándares postulados por la w3c.

A veces, los navegadores web no cumplen los estándares y por tanto toman determinación de mostrar el HTML incluso si este no es correcto.

Para validar que un HTML es correcto podemos usar el validador que proporciona la w3c en: <https://validator.w3.org/>

Archivos que no cumplen el estándar pueden parecer verse bien en un navegador, pero puede verse mal en otro.

Resultados de validación.

El validador da dos tipos de mensajes en caso de que haya cosas incorrectas en el sitio:

- **Errores:** cuando se rompe una regla HTML estricta
- **advertencias:** Cuando no se siguen buenas prácticas que son recomendables, o se tiene elementos innecesarios en el sitio web.

Ej. error: la etiqueta “hgroup” está en un lugar no permitido:

Descripción de imagen: Resultado del validador de la W3C con errores

Ej. advertencia: la etiqueta “h1” debería ser utilizado solo como encabezado principal del sitio:

Descripción de imagen: Resultado del validador de la W3C con advertencias

Sección 5. Estilos e interactividad

Estilos.

Las páginas web que podemos crear con HTML, se ven bastante planas. Sin embargo, los sitios que vemos en internet se ven muy distintos.

La pregunta sería entonces, ¿Cómo hacer para tener un sitio como alguno de los siguientes?

Descripción de imagen: Muestra de CSS

Cascade Style Sheets.

CSS es un lenguaje adicional que permite describir “cómo” se va a ver un determinado contenido en una página web.

También es desarrollado y mantenido por la W3C.

Permite definir, por ejemplo, tipos y tamaños de fuentes, colores de texto y fondo, alineados de texto, sombreado de elementos, transparencias, posicionado de elementos complejos, etc.

Puede agregarse al HTML mediante una etiqueta “style”, o mejor aún, en un archivo separado utilizando la etiqueta “link”

Aprender este lenguaje no es pertinente a esta asignatura.

Interactividad.

JavaScript es un lenguaje de programación que ejecuta directamente en el navegador.

También desarrollado y mantenido por la W3C.

Permite realizar interactividad con el usuario de formas avanzadas, mediante el agregado o quitado de etiquetas y contenido al HTML tras una determinada acción del usuario.

Aprender este lenguaje no es pertinente a esta asignatura.

Fin de la lección.