



HTML

Tags

apuntes

Introducción

Es un markup language:

- Sirve para formatear el texto

- Importa el orden

- Distinguible del texto

- No se muestra cuando se procesa

- Sistema para anotar documentos

Html → HyperText Markup Language

Hipertexto → texto con referencias a otro texto

HTML TAGS:

- Primera descripción pública de HTML

- Publicado en internet por Tim Berners-Lee, un científico del CERN, en 1991

- Como objeto facilitar la navegación entre diferentes documentos

Historia de HTML:

- 1990 → primer borrador Tim Berners-Lee

- 1995 → La primera versión de HTML que se convirtió en un estándar fue la 3.0, cuando el W3C (world wide web consortium) lo aceptó como uno de sus estándares

- 1999 → Llegó HTML 4

- 2008 → Se comienza a definir la versión 5 (Storage en la máquina, Tipo (fuentes ...), games, audio, 3d, video)

- Actualidad → se conoce solo como HTML y es un living standard (?)

Elemento

```
<!DOCTYPE html>
<!-- Declaración de tipo de documento -->
<!-- No es una etiqueta html, va al principio del documento -->
```

```
<html [lang="es"]> </html>
```

Cabecera:

```
<head>
  <title>Titulo único y obligatorio que sale en la barra</title>
  <meta name="descripción" content="esto es un metadato">
  <meta name="autor" content="Roberto">
  <!-- Favicon, puede ser
  .ico,
```

```
.png o
.svg -->
<link rel="icon" type="image/x-icon" href="/ruta/a/favicon.ico/">
</head>
```

Body:

```
<body>
  Todos los elementos visibles en la página web
</body>
```

Encabezado:

```
<h1>sdfbn</h1>
<h2></h2>
<h3></h3>
...
<h6></h6>
```

Párrafos:

```
<p></p>
```

Salto de línea:

```
<br><br><br> <!-- aqui van 3 -->
```

Estilos de texto

```
<b> negrita </b>
<i> cursiva </i>
<em>énfasis</em>
<strong>más énfasis</strong>
<small>Tamaño reducido</small>
<u>Subrayar</u>
```

Atributos:

```
<!--id -> tiene que ser único, solo uno por elemento-->
<p id="parrafo_loco">hola</p>
<!--clase -> puede tener varias un elemento y puede ser de varios elementos
las clases deben estar separadas por espacios dentro del mismo "-->
<p class="textos_feos textos_aburridos"></p>
```

Lista ordenada:

1. uno
2. dos
3. tres

```
<ol>
  <li>uno</li>
  <li>dos</li>
```

```
<li>tres</li>
</ol>
```

Lista sin ordenar:

- uno
- dos
- tres

```
<ul>
  <li>uno</li>
  <li>dos</li>
  <li>tres</li>
</ul>
```

Imagen → img:

Es un void element → no lo hay que cerrar, como


```

<!-- heigth y width en pixeles -->
```

Table

mes	ahorros
julio	100

```
<table>
  <tr>
    <th>mes</th>
    <th>ahorros</th>
  </tr>
  <tr>
    <th>julio</th>
    <th>100</th>
  </tr>
</table>
```

Tabla pro:

Phone numbers	
Name	Phone
John	577854 577855
Jack	577856 577857
Personal Office	

Link

Añadidos

- ```
 Pestaña en el ctxto actual (default)
 Nueva pestaña
<!--también existe _top-->
```

- **Download:**

```
ruta como descarga
```

Divisor:

```
<div>
 <h1>Título de la página</h1>
 <p>hola que tal xd</p>
</div>
```

Span:

Contenedor dentro de una línea → para resaltar una parte de un texto

```
<p>Hola que tal Esto es azul</p>
```

Formularios:

First name:

Second Name:  Fecha:

Chechbox ☐

```
<!-- en action ponemos la página a la que va destinada la informacion-->
<!-- get es el method por defecto, tm puede ser post-->
<form action="/action_page.php" method="get" id = "formulario_Ejemplo">
<label>First name: <input type="text" name="fname" placeholder="Mete el texto" required
<!--También hay maxlength-->
<label>Second Name: <input type="text" name="appell" placeholder="min size" minlength="1
<label for="fechita">Fecha: </label>
<!--Name sirve como clave para el valor que estamos introduciendomelo y pasamos al
archivo-->
<input type="datetime-local" id="fechita" name="clave para el servidor, fecha valor">

<!-- La caja y el texto están separados, no hay atr "for" ni anidado-->
<label>Chechbox</label><input type="checkbox">

<input type="submit" value="Submit">

<select name="campo_seleccionado" id="sel">
 <option value="volvito">volvito</option>
 <option value="nisan">nisancito</option>
</select>
</form>
<!--Otro tipo de botón, con form para ponerlo fuera-->

<button form="formulario_ejemplo">
 botoncito form
</button>
```

Estructura de contenido:

- `<article>` → Composición en un documento (post, comentario...)
- `<footer>` → Pie de página con información del autor, copyright y otros
- `<nav>` → Enlaces de navegación como menús o tablas de contenido
- `<header>` → Cabecera de la página
- `<section>` → Sección de un documento
- `<aside>` → Menú lateral



Canvas:

Contenedor de gráficos, usamos js



```
<canvas id="myCanvas">
 Your browser does not support the canvas tag.
</canvas>
<script>
 var canvas = document.getElementById("myCanvas");
 var ctx = canvas.getContext("2d");
 ctx.fillStyle = "#FF0000";
 ctx.fillRect(0, 0, 80, 80);
</script>
```

## IMPORTANTE

Consideraciones generales de estilo

- Los nombres de etiqueta y atributo siempre en minúscula
  - Aplicar etiquetas de cierre
  - Valores de atributos entre comillas
  - Especificar los atributos alt, width y height de las imágenes
- ```

```
- No usar espacios antes y después del signo igual
 - Evitar líneas de código demasiado largas
 - No añadir líneas en blanco ni aplicar sangrado sin necesidad

- Recomendable usar dos espacios para sangrar y no tabulador
- Usar siempre las etiquetas <html>, <head>, <body> y <title>

Carácteres especiales

| Secuencia de escape | Caracteres |
|---|------------------------|
| < | < |
| > | > |
| & | & |
| " | " |
| á é í
ó ú | á, é, í, ó, ú |
| ñ | ñ |
| ¿ | ¿ |
| | (Espacio no separador) |

robots.txt

- Archivo que puedes incluir en el directorio raíz
- Formato UTF-8
- Le indicas a los web crawlers qué URLs pueden acceder en tu sitio
- Sirve para evitar sobrecargar el sitio con peticiones
- No sirve para que un recurso no aparezca en los buscadores
- [Más info](#)

```
User-agent: Googlebot
Disallow: /nogoogobot/

User-agent: *
Allow: /

Sitemap: http://www.example.com/sitemap.xml
```

security.txt

- Archivo que puedes incluir en /.well-known/security.txt
- Indica a alguien que haya encontrado una vulnerabilidad cómo proceder
- Incluye información de contacto
- Puede incluir claves para cifrar la información
- [RFC 9116](#)

```
# Our security address
Contact: mailto:security@example.com

# Our OpenPGP key
Encryption: https://example.com/pgp-key.txt

# Our security policy
Policy: https://example.com/security-policy.html

# Our security acknowledgments page
Acknowledgments: https://example.com/hall-of-fame.html

Expires: 2021-12-31T18:37:07z
```



ai.txt

- Propuesta de estándar
- Archivo que puedes incluir en el directorio raíz
- Instrucciones sobre el uso de imágenes, media y código por parte de la AI
- [Más información](#)

```
# AI.TXT
# Spawning AI / Prevent datasets from using the following file types

User-Agent: *
Disallow: *.gif
Disallow: *.jpg
Disallow: *.png
...
Allow: *.txt
Allow: *.pdf
Allow: *.doc
...
Allow: *.css
Allow: *.php
Allow: *.sql
Allow: /
```



XHTML

HTML vs. XHTML Comparison

| HTML | XHTML |
|---|--|
| <pre><html> <header> <title>My Web Page </title> </header> <body> <p> Hello to my page!! </p> </body> </html></pre> | <pre><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml 1-transitional.dtd"> <html xmlns="http://www.w3.org/1999/xhtml"> <head> <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> <title>XHTML Sample Web Page</title> </head> <body> <p> Here is my content!! </p> </body> </html></pre> |

XHTML

| | HTML | XHTML |
|---------------------|---|---|
| Filename extension | .html, .htm | .xhtml, .xht, .xml, .html, .htm |
| Internet media type | text/html | application/xhtml+xml |
| Developed by | W3C & WHATWG | World Wide Web Consortium |
| Type of format | Document file format | Markup language |
| Extended from | SGML | XML, HTML |
| Stands for | HyperText Markup Language | Extensible HyperText Markup Language |
| Application | Application of Standard Generalized Markup Language (SGML). | Application of XML |
| Function | Web pages are written in HTML. | Extended version of HTML that is stricter and XML-based. |
| Nature | Flexible framework requiring lenient HTML specific parser. | Restrictive subset of XML and needs to be parsed with standard XML parsers. |
| Origin | Proposed by Tim Berners-Lee in 1987. | World Wide Web Consortium Recommendation in 2000. |
| Versions | HTML 2, HTML 3.2, HTML 4.0, HTML 5. | XHTML 1, XHTML 1.1, XHTML 2, XHTML 5. |

XHTML

| | HTML | XHTML |
|---|---|--|
| Tag nesting | Tags may be closed out of order | Tags must be closed in the order opened |
| Tag case | Tags may be written in upper or lower case | Tags must be written in lower case |
| Tag closure | Closing tags may be omitted for some elements | All opening tags must be matched with closing tags |
| Attribute name/value pairs | Value may occur without attribute name | Name/value pair required |
| Empty elements | Empty elements do not need to be closed | Empty elements must be closed |
| Script and Style elements | Elements may be placed within HTML document | Elements must be in separate documents and linked to XHTML documents |
| Attribute values | Values may be enclosed within quotes | Values must be enclosed within quotes |
| ID and Name attributes | Either attribute may be used | ID attribute may be used, Name attribute prohibited |
| Character codes for special characters | Codes recommended | Codes required |



CEU

Universidad
San Pablo

HTML 48