



APUNTES DE HTML

ESTRUCTURA DOCUMENTO HTML

```
<!DOCTYPE html>
```

- Indica que el tipo de documento es HTML5

```
<html>
```

- Elemento principal de la página

```
<head>
```

- Contenedor de información

```
<meta charset="UTF-8">
```

- Indica la codificación de caracteres

```
<meta name="viewport"
content="width=device-width,initial-scale=1.0">
```

- Indica dimensiones y escala de zoom inicial

```
<title>Título</title>
```

- Declara el título de la página

```
</head>
```

```
<body>
```

- Contenedor de elementos visibles

```
<p>Estoy aprendiendo HTML!</p>
```

- Elemento que define un párrafo

```
</body>
```

```
</html>
```

ELEMENTO HTML

Etiqueta de apertura

`<h1 title="saludo">` Hola Mundo

Etiqueta de cierre

`</h1>`

Nombre del elemento

`<h1 title="saludo">` Hola Mundo `</h1>`

Nombre del atributo

`<h1 title="saludo">` Hola Mundo `</h1>`

Valor del atributo

`<h1 title="saludo">` Hola Mundo `</h1>`

Contenido del elemento

`<h1 title="saludo">` Hola Mundo `</h1>`

Elementos vacíos (no necesitan contenido ni cierre)

`
`, ``, `<meta>`, `<hr>`, `<input>`...

ELEMENTOS MÁS COMUNES EN BODY

Encabezados:

- Etiquetas: `<h1>` `<h2>` `<h3>` `<h4>` `<h5>` `<h6>`
- H1 indica lo más importante de la web (por ejemplo título de un post dentro de un blog)
- `<h1>`Hola soy un h1`</h1>`

Párrafo:

- Etiqueta: `<p>`
- Usado para incluir texto en párrafos, el grueso de la información (por ejemplo el contenido del post)
- `<p>`Hola soy un párrafo`</p>`

Link:

- Etiqueta: `<a>`
- Usado para enviar al usuario a una url específica
- Atributo más importante: ``
- `href` Usado para enviar al usuario a una url específica
- El contenido puede ser una imagen u otro elemento (no es obligatorio que sea texto únicamente)
- ``Web de Navis Code``

División:

- Etiqueta: `<div>`
- Usado como contenedor de otros elementos
- Se puede poner cualquier contenido en su interior
- `<div>` `<p>` Párrafo en un div `</p>` `</div>`

Imagen:

- Etiqueta: ``
- Incluye una imagen en una documento HTML
- Elemento vacío (solo contiene atributos, sin etiqueta de cierre)
- Atributos obligatorios ``
- `src` Ruta donde se encuentra la imagen (relativa o absoluta)
- `alt` Texto alternativo de una imagen (en caso de no poder ver la imagen o al usar un lector de página por accesibilidad)
- ``

Formulario:

- Etiqueta: `<form>`
- Usado para recoger datos que introduzca el usuario
- Es un contenedor para diferentes tipos de input element
- `<form>`
 - `<input type="text" name="apellido">`
 - `<input type="submit" value="Submit">`
- `</form>`

ELEMENTOS SEGÚN DISPLAY

Cada elemento HTML tiene por defecto un valor display. Block o Inline y lo usaremos para representar según queramos nuestra página web.

Block:

- Siempre empieza en una nueva línea y ocupa el ancho disponible
- Algunos elementos con display block por defecto:

`<div> <p> <form> <h1 - h6> <section> <header> <footer>`

Inline:

- Usa la línea actual y ocupa el ancho necesario
- Algunos elementos con display inline por defecto:

`<a> <button> <input> <small>`

**A tener en cuenta que estos son valores por defecto y se pueden modificar con CSS o JavaScript

CSS EN HTML (INLINE, INTERNO Y EXTERNO)

Inline:

- Da estilos a un único elemento de HTML
- Se usa gracias al atributo `style` de un elemento de HTML
- `<p style="color: green">Un párrafo de color verde</p>`

Interno:

- Da estilos a un único documento de HTML
- Se define como contenido de `<head>` dentro del elemento `<style>`
- `<head>`
 `<style>`
 `p {`
 `color: green;`
 `}`
 `</style>`
`</head>`

Externo:

- Da estilos a varios documento de HTML desde una hoja de estilos
- Se importa en un `<link>` dentro de `<head>` en cada documento
- `<head>`
 `<link rel="stylesheet" href="style.css">`
`</head>`

JAVASCRIPT EN HTML (INTERNO Y EXTERNO)

Interno:

- Ejecutará el código de JavaScript que exista como contenido
- Se define en `<script>` dentro de cada documento
- `<script> alert("Estás aprendiendo a tope!") </script>`

Externo:

- Usa un script de un archivo en el documento HTML
- Se importa en un `<script>` dentro de `<head>` en cada documento
- `<head>`
`<script src="main.js"></script>`
`</head>`

**Puede que veas una forma obsoleta de insertar un script dentro de un documento HTML y es incluirlo al final de la etiqueta `<body>`

**La forma moderna de insertarlo es usando los atributos de la etiqueta `<script>` como son `async` o `defer` quedando de la siguiente forma:

```
<head>  
  <script src="main.js" defer></script>  
</head>
```

**También es posible incluirlo indicando que son `modules`, quedando de la siguiente forma:

```
<head>  
  <script src="main.js" type="module"></script>  
</head>
```

RUTAS ABSOLUTAS Y RUTAS RELATIVAS

Ruta absoluta:

- Indica una dirección completa, se usa para apuntar fuera de la página web en la que nos encontramos.
- Ejemplo, usar una imagen directamente de la web Unsplash
- ``

** Ruta a ejemplos ilustrativos, no es correcta

Ruta relativa:

- Indica una dirección relativa a nuestra página web, usada para hacer referencia a algún elemento dentro de nuestro proyecto
- Ejemplo, usar una imagen del directorio img
- ``

ATRIBUTOS GLOBALES

Pueden ser usados en todos los elementos de HTML.
Entre los más usados están:

class:

- Permite dar uno o más nombre de clases para un elemento
- Se suele usar para dar estilos con CSS o modificar con JavaScript
- Diferentes elementos pueden compartir mismo nombre de clase
- `<p class="destacado">` Párrafo con clase `</p>`
- `<p class="destacado grande">` Párrafo con varias clases`</p>`

id:

- Da una identificación única a un elemento
- Se suele usar para dar estilos con CSS o modificar con JavaScript
- Id deben ser únicos (solo 1 elemento con mismo id por HTML)
- `<h1 id="miCabecera">` Cabecera principal `</h1>`

data-*:

- Permite guardar datos privados de la página
- Se pueden crear atributos data personalizados
- Estos datos pueden ser usados en JavaScript posteriormente
- `<h2 data-user-name="Juan Lopez Gonzalez">` Juanito `</h2>`

hidden:

- Usado para NO mostrar un elemento que contenga este atributo
- Es un atributo que no necesita que se indique un valor
- Se suele usar JavaScript para mostrar o esconder un elemento
- `<p hidden >` Este párrafo no se mostrará en el navegador `</p>`

style:

- Especifica un estilo inline para el elemento
- Este estilo sobrescribirá a cualquier estilo global
- Se suele usar JavaScript para mostrar o esconder un elemento
- `<p style="text-align: center;">` Párrafo alineado al centro `</p>`

title:

- Usado para dar información extra sobre el elemento
- Suele mostrar esta información como tooltip
- `<p title="Párrafo con un título">` Párrafo `</p>`

SEMÁNTICA EN HTML (No todo son divs)

Un elemento semántico nos indica su intención de forma directa.

Ejemplo elem. semánticos:

```
<header></header>
```

```
<nav></nav>
```

```
<section></section>
```

```
<section></section>
```

```
<footer></footer>
```

Ejemplo elem. no semánticos:

```
<div id="header"></div>
```

```
<div id="nav"></div>
```

```
<div class="section"></div>
```

```
<div class="section"></div>
```

```
<div id="footer"></div>
```

Beneficios:

- Ayuda a mantener el código y orientarnos por el
- Navegación fácil para usuario, especial si usa lectores de pantalla (screen readers) para accesibilidad
- Mejora el SEO de la página

GUÍAS DE ESTILO

- Usar minúsculas (lowercase), en nombre de elementos y atributos

`<p class="main">Hola</p>` ✓

`<P CLASS="main">Hola</P>` ✗

- Usar etiquetas de cierre siempre que sea posible

`<p>Hola</p>` ✓

`<p>Hola` ✗

- Usar comillas para los valores de los atributos

`<p class="main">Hola</p>` ✓

`<p class=main>Hola</p>` ✗ Funciona aunque no recomendado

`<p class=main par>Hola</p>` ✗ No funciona

- Definir el doctype como HTML5, incluye lenguaje, charset y viewport

`<!DOCTYPE html>` ✓

`<html lang="es">` ✓

`<meta charset="utf-8">` ✓

`<meta name="viewport" ...>` ✓

- No incluir espacios entre los signos de = al indicar valores

`<p class="main">Hola</p>` ✓

`<p class = "main">Hola</p>` ✗



APUNTES DE HTML

Creado por Navis Code como parte de
[Frontend Fastlane Plan](#)

Si estás aprendiendo me alegraría mucho saberlo
[@NavisCode](#)