

Atributos comunes HTML

HTML está formado por etiquetas. Cada etiqueta tiene una misión y tarea y contendrá cierto contenido relacionado con su misión. Y además tenemos los **atributos**, que son palabras clave de texto que modifican ligeramente el comportamiento de la etiqueta que lo contiene.

Atributo id

El identificador **id** se suele usar cuando queremos localizar un elemento dentro de nuestro documento y sabemos que dicho elemento o etiqueta no se va a repetir dentro de nuestro documento.

Este es un identificador **ÚNICO**, por lo que varios elementos no podrán tener el mismo id. También lo utiliza JavaScript para acceder y manipular el elemento con la identificación específica.

Para usarlo simplemente añadimos **id** y añadimos el nombre como valor del atributo. Este nombre no deberá comenzar por números y puedes mezclar mayúsculas, minúsculas u otros caracteres.

```
<div id="personal_info">  
  <p>Alberto Rivera</p>  
  <p>Desarrollador Full Javascript</p>  
</div>
```

Atributo class

La idea de las clases es asociar características comunes a diferentes etiquetas, es decir si tenemos varios **<p>** que queremos que visualmente sean iguales podemos asignarles la misma clase y así su visualización en el navegador será similar.

Por lo tanto, esta será la principal diferencia entre `id` y `class`, la posibilidad de asociar estilos comunes a varios elementos.

```
<div class="personal_container" id="personal_info">
  <p class="personal_text">Alberto Rivera</p>
  <p class="personal_text">Desarrollador FullStack</p>
</div>
```

Al tener varios `<p>` la clase `personal_text` podemos dotarlas de contenido sin necesidad de hacerlo por separado. Lo único complejo es que una etiqueta puede tener varias clases.

```
<div class="personal_container" id="personal_info">
  <p class="personal_text name">Alberto Rivera</p>
  <p class="personal_text job">Desarrollador Full Javascript</p>
</div>
```

Debemos tener en cuenta que cuando queramos añadir varias clases se hace dentro de `class` y envolvemos con comillas dobles `""` pero dejamos un espacio de separación entre nombre de clase y nombre de clase.

Atributo idioma

Utilizar el atributo `lang` es muy sencillo. Como cualquier otro atributo, se incluye en la etiqueta del elemento y se utiliza `=` para asignarle un valor.

Siempre debe incluir el `lang` atributo dentro de la `<html>` etiqueta, para declarar el idioma de la página web. Esto se usa para ayudar a los motores de búsqueda y navegadores.

```
<p lang="es">Soy un texto.</p>
```

El atributo `lang` se puede **utilizar en elementos anidados** para especificar un idioma diferente al idioma del elemento padre.

```
<!doctype html>
<html lang="es">
....
<body>
  <p>Texto en español</p>
  <p lang="en">Text in english.</p>
  <p>saludo, en inglés <span lang="en">hello</span>, que significa hola.</p>
</body>
</html>
```

Atributo no traducible

También podemos utilizar el atributo `translate` para indicar si el contenido de un texto es o no traducible.

```
<p>
  hoy he comido un <span translate="no">Sandwich</span>.
</p>
```

Atributo title

El atributo `title` define alguna información adicional sobre un elemento.

El valor del atributo de título se mostrará cuando el usuario detenga el ratón sobre el elemento un instante.

```
<div>
  
</div>
```

Atributo dirección

El atributo `dir` nos ayuda a marcar la dirección de nuestras cadenas de texto que se visualizan dentro de las aplicaciones web. Esto es perfecto cuando tenemos que trabajar con multi-idioma, ya que algunos idiomas como el japonés su lectura es de izquierda a derecha.

```
<p>Dirección de izquierda a derecha.</p>
<p dir="rtl">Dirección de derecha a izquierda.</p>
```

Por defecto el valor es `ltr` por ello solamente tendremos que indicar los campos `rtl`.

Atributo dataset

La mayoría de los metadatos que se incluyen en una web están insertados dentro del `<head>`. Sin embargo, no es la única manera que tenemos de insertarlos, también podemos hacerlo a través de las etiquetas que contengan el atributo con prefijo `data-` (`data-share` por ejemplo).

Normalmente esto está orientado a usarse junto a js:

```
<div id="compartir">
  <a href="https://twitter.com/mitweet"
    data-share="190">Twitter</a>
  <a href="https://www.facebook.com/mipost"
    data-share="340">Facebook</a>
</div>
```

En este ejemplo de código, el atributo `data-share` representará el número de veces que ha sido compartido un enlace de la página.

Atributo accesskey

Este atributo indica un atajo de teclado o shortcut. Vamos a poner un ejemplo con dos campos de texto, un enlace y un botón. Cada uno tendrá un accesskey.

IMPORTANTE: estos shortcuts variarán dependiendo del sistema operativo y navegador que uséis.

```
<form>
  <input accesskey="N" placeholder="Nombre (ALT+N)" /> <!-- Campo de datos -->
  <input accesskey="A" placeholder="Apellidos (ALT+A)" /> <!-- Campo de datos -->
  <a accesskey="L" href="#">Enlace (ALT+L)</a> <!-- Enlace -->
  <button accesskey="B">Botón (ALT+B)</button> <!-- Botón -->
</form>
```

De esta forma, si pulsamos ALT + N se hará foco automáticamente en el input referido al nombre, ALT + L será como si hubiésemos clicado el enlace y ALT + B pulsará el botón.

Una de las mayores limitaciones de este atributo es la falta de cohesión entre navegadores y sistemas operativos. Esto quiere decir que, mientras en Windows el comando anterior era ALT + N en el navegador Chrome, en Firefox para el mismo sistema operativo será ALT + SHIFT + A. Por este motivo es que este atributo es poco práctico en un trabajo profesional.

Atributo style

El atributo **style** utiliza las etiquetas de HTML y sirve para incrustar código CSS directamente en la etiqueta. También se le llama css en línea.

Normalmente la mejor forma de insertar este código css es a través de un documento con la extensión `.css` aparte. Esto nos aporta mejor visualización, ordenación y facilidad. Sin embargo, hay situaciones en las que puede que nos veamos obligados a insertar este css en línea.

Os vamos a poner un ejemplo de código de este atributo **style**:

```
<p style="background: blue; color: whitesmoke;">Usando el atributo style</p>
```

En definitiva estamos usando código css en el propio documento HTML, pero como os podéis imaginar hacer esto en una aplicación grande puede llenar la pantalla de líneas de código innecesarias que podríamos tener en un archivo `.css` aparte.

¿Qué hemos aprendido?

1. Los atributos más comunes de HTML con su significado y uso.
2. Dar estilos en línea a nuestro HTML.