

- **LISTAS HTML**

Las listas en HTML nos permiten crear conjuntos de elementos en forma de lista dentro de una página, todos los cuales irán precedidos, generalmente, por un guion o número.

Los tipos de listas en HTML son los siguientes:

- ✓ Listas ordenadas
- ✓ Listas desordenadas
- ✓ Listas de definiciones

## **1. Listas Ordenadas**

Las listas en HTML ordenadas son aquellas que nos muestran los elementos de la lista en orden. Para representar el orden tendremos los elementos numerados. Es decir, cada uno de los elementos irá precedido de un número o letra que establezca su orden.

Las listas en HTML ordenadas se representan mediante el elemento OL.

---

```
<ol> ... </ol>
```

Cada uno de los elementos de la lista ordenada se representará mediante el elemento LI.

---

```
<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ol>
```

Un ejemplo de lista ordenada sería el siguiente:

```
<ol>
  <li>Julio</li>
  <li>Carmen</li>
  <li>Ignacio</li>
  <li>Elena</li>
</ol>
```

Que produciría el siguiente efecto:

1. Julio
2. Carmen
3. Ignacio
4. Elena

**Número de inicio de la lista: start**

El atributo start nos permite indicar el número por el cual queremos que empiece la lista, ya que por defecto las listas ordenadas en HTML empiezan por el número 1.

```
<ol start="numero"> ... </ol>
```

De esta forma, si queremos que la lista empiece por el número 5, escribiremos el siguiente código:

```
<ol start="5">
  <li>Julio</li>
  <li>Carmen</li>
  <li>Ignacio</li>
  <li>Elena</li>
</ol>
```

Que produciría el siguiente efecto:

5. Julio
6. Carmen
7. Ignacio
8. Elena

### Tipo de lista ordenada: type

De igual manera podemos indicar el tipo de lista ordenada en HTML que queremos representar.

Entre los tipos de listas que podemos representar tenemos:

- 1 - Listas decimales
- a - Listas alfabéticas en minúsculas
- A - Listas alfabéticas en mayúsculas
- i - Listas de números romanos en minúsculas
- I - Listas de números romanos en mayúsculas

Los valores indicados al principio son los que le podemos asignar al atributo type de la lista ordenada en HTML.

```
<ol type="tipo"> ... </ol>
```

Si queremos que nuestra lista aparezca ordenada mediante letras en mayúsculas escribimos lo siguiente:

```
<ol type="A">
  <li>Julio</li>
  <li>Carmen</li>
  <li>Ignacio</li>
  <li>Elena</li>
</ol>
```

En este caso la lista se representará de la siguiente forma:

```
<ol type="A">
```

- Julio
- Carmen
- Ignacio
- Elena

```
</ol>
```

## Listas en orden inverso: reversed

En HTML 5 aparece el atributo reversed para las listas ordenadas. El atributo reversed nos permite invertir el orden de la lista. Es decir, realiza la numeración de la lista de forma inversa.

Para utilizarlo simplemente indicamos el atributo reversed sobre el elemento OL.

```
<ol reversed> ... </ol>
```

En este caso, si escribimos la siguiente lista:

```
<ol reversed>
<li>Julio</li>
<li>Carmen</li>
<li>Ignacio</li>
<li>Elena</li>
</ol>
```

Lo que obtendremos por pantalla será lo siguiente:

4. Julio
3. Carmen
2. Ignacio
1. Elena

## 2. Listas Desordenadas

Las listas desordenadas en HTML nos sirven para mostrar los elementos sin ningún tipo de orden, simplemente precedidos por una viñeta que puede ser un punto, un cuadrado.

Para definir una lista desordenada en HTML utilizamos el elemento ul.

```
<ul> ... </ul>
```

Para representar los elementos de la lista desordenada utilizamos el mismo elemento que con las listas ordenadas, es decir, el elemento li.

```
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ul>
```

De esa forma podríamos tener el siguiente código HTML:

```
<ul>
  <li>FC. Barcelona</li>
  <li>Real Madrid</li>
  <li>Real Betis</li>
  <li>Atlético de Madrid</li>
</ul>
```

Lo que nos mostrará por pantalla:

- FC. Barcelona
- Real Madrid
- Real Betis
- Atlético de Madrid

### Tipos de lista desordenada

En el caso de las listas en HTML desordenadas en HTML no podemos indicarle el tipo de lista mediante HTML. En este caso tenemos que recurrir a CSS para poder indicar el tipo de lista utilizando el atributo List-style-type

```
<style type="text/css">
ul {
  list-style-type: circle;
}
</style>
```

## 3. Listas de Definiciones

Las listas en HTML de definiciones en HTML nos sirven para montar listas en las que tenemos la estructura valor y definición. Suelen ser listas para definir términos, como si fuese un diccionario, si bien pueden ser cualquier par valor-definición.

Las listas en HTML de definiciones en HTML se construyen mediante el elemento dl.

```
<dl> ... </dl>
```

En este caso, dentro de las listas en HTML de definiciones tenemos dos elementos anidados, el que representa al valor dt y el que representa a la definición dd. De esta forma la estructura de las listas en HTML de definiciones es la siguiente:

```
<dl>
  <dt>Término1</dt>
  <dd>Definición 1</dd>
  <dt>Término 2</dt>
  <dd>Definición 2</dd>
  ...
  <dt>Término N</dt>
  <dd>Definición N</dd>
</dl>
```

Si queremos crear una lista en HTML con definiciones de palabras, podemos escribir lo siguiente:

```

<dl>
  <dt>Pizpireta</dt>
  <dd>Dicho de una mujer: Viva, pronta y aguda.</dd>
  <dt>Pulular</dt>
  <dd>Dicho de las personas, animales o cosas: Abundar y bullir en un lugar.</dd>
  <dt>Concupiscencia</dt>
  <dd>En la moral católica, deseo de bienes terrenos y, en especial, apetito desordenado de placeres deshonestos.</dd>
</dl>

```

## Listas anidadas

Cuando estemos manejando listas podemos anidar unas en otras independientemente del tipo de lista que estemos anidando.

Para crear listas anidadas en HTML simplemente tenemos que hacer que el elemento de una de las listas sea a su vez una lista. Es decir, el esquema de listas sería parecido al siguiente:

```

<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  <li>
    <ol>
      <li>Elemento 2.1</li>
      <li>Elemento 2.2</li>
      ...
      <li>Elemento 2.N</li>
    </ol>
  </li>
  <li>Elemento 3</li>
  ...
  <li>Elemento N</li>
</ul>

```

En este caso hemos anidado una lista ordenada dentro del tercer elemento li de una lista desordenada.

Hay que tener cuidado de poner el elemento li dentro de la lista anidada, ya que si no lo ponemos estaremos generando un código incorrecto. Las anidaciones de listas pueden ser de cualquier tipo de lista y sin límite de anidación.

Un ejemplo de listas anidadas sería una clasificación de animales como la siguiente:

```

<ul>
  <li>Carnívoro
    <ul>
      <li>León</li>
      <li>Buitre</li>
      <li>Hiena</li>
    </ul>
  </li>
  <li>Herbívoro
    <ul>
      <li>Cabra</li>
      <li>Vaca</li>
    </ul>
  </li>
  <li>Omnívoro
    <ul>
      <li>Oso</li>
      <li>Urraca</li>
    </ul>
  </li>
</ul>

```

El efecto que obtendríamos sería el siguiente:

- Carnívoro
  - León
  - Buitre
  - Hiena
- Herbívoro
  - Cabra
  - Vaca
- Omnívoro
  - Oso
  - Urraca

## • ENLACES HTML

### ¿Qué son los enlaces en HTML?

Lo más importante de los documentos HTML son los enlaces. Ya que mediante los enlaces en HTML podemos comunicar una página con otra. De esta forma, enlazando documentos HTML podemos acabar tejiendo lo que es Internet.

Para crear un enlace en HTML utilizamos el elemento A con la siguiente sintaxis:

```
<a>Contenido del enlace</a>
```

Pero solo con esto el enlace no tendrá mucha utilidad ya que el principal objetivo del enlace es enlazar a un destino. Para poder indicar el destino de un enlace utilizamos el atributo href. En valor del atributo href puede ser cualquier URI que represente un recurso. Es decir, una página, una parte de una página, una URL genérica, un archivo. De esta forma el enlace en HTML lo crearemos con la sintaxis:

```
<a href="URI">Contenido del enlace</a>
```

A modo de ejemplo podríamos tener los siguientes enlaces:

```
<a href="documento.html">Enlace a un documento</a>
<a href="documento.html#resumen">Enlace a una parte de un documento</a>
<a href="http://www.manualweb.net">Enlace a una web</a>
<a href="http://www.manualweb.net/tutorial-html/">Enlace a un directorio</a>
<a href="miimagen.png">Enlace a una imagen</a>
<a href="mimusica.mp3">Enlace a un archivo de sonido</a>
```

---

### Destino del enlace

Pero ¿dónde se abre el enlace? Pues por defecto y si no hemos configurado nada en el navegador web que estemos utilizando el enlace se abre en la misma ventana en la que tengamos el enlace.

Si bien, en el enlace, podemos indicar el destino que queremos darle a dicho enlace. Eso lo podemos hacer mediante el atributo target. Los posibles valores que admite el atributo target son:

- `_blank`, el agente de usuario intentará abrir el enlace en una nueva ventana. La nueva ventana no tendrá nombre.

- `_self`, el agente de usuario intentará abrir el enlace en el mismo marco donde está en código actual.
- `_parent`, el agente de usuario intentará abrir el enlace en el frameset inmediatamente superior al que se encuentra la página. Esto suele suceder cuando tenemos el enlace en un área de frames.
- `_top`, el agente de usuario intentará abrir el enlace en la ventana padre. En el caso de que exista un frameset lo eliminará y se hará con toda la ventana.
- `nombre_marco`, se podrá indicar el nombre de un frame. En este caso el agente de usuario intentará abrir el enlace en el frame que coincida con el nombre. En el caso de no existir un frame con ese nombre lo abrirá en una nueva ventana, asignándole dicho nombre

Así podremos tener el siguiente código:

```
<a href="enlace.html" target="_blank">Abrir enlace en una nueva ventana</a>
<a href="enlace.html" target="_top">Abrir enlace en la ventana superior</a>
```

## Título de los enlaces

El enlace en HTML, tal y como lo hemos visto hasta ahora, sirve para enlazar contra un recurso de la web: servidor, directorio, dominio y lo que en mayor o menor medida describe lo que enlazamos es el contenido que encontramos entre las etiquetas A.

Si bien el elemento A nos ofrece un atributo llamado title. En el atributo title podemos describir de una forma textual el destino del enlace. Esto servirá no solo al usuario para que pueda obtener más información de dónde va, si no a las máquinas a la hora de asignar un nombre a la URI sobre la que estamos enlazando.

## Un ejemplo sería:

```
<a href="http://www.manual.net" title="Web de Manuales y Tutoriales de Programación">Ir a Manual Web</a>
```

- Enlaces en HTML a una parte del documento

Hasta el momento lo que hemos visto es como montar enlaces en HTML a documentos. Ya sea porque enlazamos directamente al documento o bien porque enlazamos a un servidor o directorio que nos dará un documento.

Pero otra capacidad que tenemos en HTML es la de enlazar a una parte concreta del documento. Imagina que en un documento tenemos un título y varios capítulos. Y lo que queremos hacer desde otro documento HTML o bien desde el mismo documento es enlazar directamente al inicio de un capítulo.

- Marcando una parte del documento

Para poder enlazar a una parte concreta de un documento lo primero que tenemos que hacer es marcar esa parte del documento. Para ello contamos con el atributo name. Si creamos un enlace con el atributo name, este enlace se definirá como posición y no como enlace de navegación.

La sintaxis será:

```
<a name="partedocumento">Contenido</a>
```

En este caso el contenido puede ser vacío ya que no se ofrecerá nada para navegar. Es por ello que podemos ponerlo antes de nuestro capítulo.

```
<p>Parrafo</p>  
<a name="cap2"></a><h2>Nuevo Capítulo</2>
```

Es muy importante el contenido que le demos al atributo name, ya que dicho contenido vamos a utilizarlo para acceder desde un enlace.

- Enlazando a una parte del documento

Una vez que hemos creado el marcaje del enlace en HTML en un documento es hora de acceder a esa parte del documento. Para ello solo tenemos que poner el nombre que le hayamos dado al atributo name precedido de una almohadilla.

La sintaxis será:

```
<a href="#name">Enlace a parte del documento</a>
```

Es decir, en el caso de que la parte marcada en el documento la hayamos marcado con “cap2” el enlace que tenemos que conformar será:

```
<a href="#cap2">Enlace al capítulo 2</a>
```

La parte del documento al que accedemos no tiene por qué estar en el mismo documento del enlace, puede estar en otro documento o servidor. De esta forma podemos tener enlaces a partes de otros documentos de la siguiente forma:

```
<a href="documento2.html#cap2">Enlace al capítulo 2 del documento2</a>  
<a href="http://servidor.com/#cap2">Enlace al capítulo 2 del servidor</a>
```

El utilizar la almohadilla como valor del href nos puede servir para acceder a la parte superior del documento. Por lo que esto es utilizado como enlace en las partes inferiores de los documentos HTML para subir a la parte de arriba.

```
<a href="#">Ir Arriba</a>
```

**Consejo:** para profundizar mejor sobre enlaces en HTML con imágenes, para descargar fichero, protocolos en la URL del enlace, etc. Consulta la bibliografía a continuación.



## **REFERENCIAS BIBLIOGRAFICAS**

Manual Web. (2021). Listas HTML. Recuperado de Manual Web:

<http://www.manualweb.net/html/listas-html/>

Manual Web. (2021). Enlaces HTML. Recuperado de Manual Web:

<http://www.manualweb.net/html/enlaces-html/>