

## Listas HTML



Las listas en [HTML](#) nos permite crear conjuntos de elementos en forma de lista dentro de una página, todos los cuales irán precedidos, generalmente, por un guión o número.

Los tipos de listas en [HTML](#) son los siguientes:

- Listas ordenadas
- Listas desordenadas
- Listas de definiciones

### 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>
```

Atributos start/type en HTML 4.01

Aunque en [HTML 5](#) han vuelto a la especificación los [atributos type](#) y start hay que tener cuidado con ellos, ya que en ciertas versiones como HTML 4.01 fueron declarados obsoletos, por lo cual si usamos tipos de documentos HTML 4.01 strict podríamos tener un problema en su definición.

Para estos casos (y para todos en general) podemos utilizar las hojas de estilo [CSS](#) de cara a dar estilo a las listas en [HTML](#).

De esta forma en [CSS](#) podemos escribir lo siguiente:

```
<style type="text/css">
```

```
ol {
```

```
list-style-type: lower-roman;
```

```
}
```

```
</style>
```

Lo cual hará que las listas en [HTML](#) ordenadas se muestren en números romanos y en minúsculas.

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:

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

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>
```

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>
```

Lo cual nos acabará mostrando por pantalla lo siguiente:

**Pizpireta**

Dicho de una mujer: Viva, pronta y aguda.

**Pulular**

Dicho de las personas, animales o cosas: Abundar y bullir en un lugar.

### **Concupiscencia**

En la moral católica, deseo de bienes terrenos y, en especial, apetito desordenado de placeres deshonestos.

Por defecto los navegadores dejan el término y en la siguiente línea, junto con un tabulador, la definición.

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 puede 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

Rutas de enlaces:

Se llama ruta al valor que se escribe dentro del atributo href="" y se pueden clasificar en dos tipos:

Rutas absolutas:

Las rutas absolutas son las que definen la ubicación completa de un archivo en la web y empiezan desde `https://nombredelsitio.com/archivo.html` (el archivo de destino puede ser de cualquier extensión). Por ejemplo si queremos vincular nuestro sitio con Facebook, usaríamos una ruta absoluta y el código sería:

```
<a href="https://www.facebook.com/pagina">Estamos en Facebook</a>
```

Rutas relativas:

Las rutas relativas en cambio, definen la ubicación del archivo de destino en relación a la ubicación del archivo que lo está llamando. Ejemplo: si el archivo `index.html` vincula con `quien-soy.html` y están en la misma carpeta, el código sería así:

```
<a href="quien-soy.html">Quién soy</a>.
```

Tipos de enlaces

Existen diferentes tipos de enlaces que detallo a continuación:

Enlaces internos entre archivos .html

Son los enlaces principales de un sitio. Si en el archivo `index.html`, queremos agregar un vínculo a `contacto.html` el código sería así:

```
<a href="contacto.html">Contáctenos</a>.
```

El texto que queda encerrado entre las etiquetas `<a>` es lo que ve el usuario en color azul y subrayado.

## Enlaces externos

Son los enlaces entre diferentes sitios web. Si queremos vincular un sitio web con otro, tenemos que indicar la ruta absoluta del sitio escribiendo la url completa que conviene copiarla del navegador. En este caso, es común usar el atributo `target=""` que nos permite abrir el enlace en una pestaña nueva. Por ejemplo, si quisiera poner un vínculo a Google y que éste abra en pestaña nueva, sería así:

```
<a href="https://google.com" target="_blank">Ir a Google</a>
```

## Enlaces internos: anclas

Las anclas son enlaces hacia un punto determinado dentro de un html. Se usa en varios casos, por ejemplo en una página de preguntas frecuentes donde las respuestas están desarrolladas debajo de las preguntas, o en Wikipedia, cuando el texto es extenso el menú tiene vínculos con anclas. También en los textos largos, al finalizar muchas veces se coloca un botón para subir. En los sitios de una sola página donde los botones en realidad hacen scroll, esas son anclas.

El punto de destino tiene que estar marcado con el atributo `id="algo"` y en el enlace se coloca un # (numeral) seguido del nombre. Por ejemplo:

```
<a href="#respuesta4">Pregunta 4</a>
```

"salta" a este párrafo:

```
<p id="respuesta4">acá está escrita la respuesta</p>
```

## Enlaces para ampliar una imagen

Es común armar una galería de fotos con imágenes miniaturas que vinculan a imágenes ampliadas. Si combinamos este recurso con código javascript, logramos galerías de fotos muy atractivas. El código HTML es así:

```
<a href="imagenes/foto1-ampliada.jpg"></a>
```

El navegador le muestra al usuario la foto miniatura y si cliquea en ella, le muestra la foto ampliada.

## Enlaces para descargar archivos

En realidad, dentro del atributo `href=""` podemos colocar la ruta hacia cualquier tipo de archivo. Si el navegador reconoce la extensión, lo abre por ejemplo: html, jpg, png, gif, svg, pdf, etc. Pero si no lo reconoce o es un archivo comprimido (.rar, .zip), el navegador le ofrece al usuario descargarlo. Ejemplo:

`<a href="fotos.rar">Descargá todas las fotos</a>`

Enlaces a una dirección de correo

Se puede vincular una dirección de correo para que abra en el programa de correo predeterminado. El código se escribe así:

`<a href="mailto:info@dominio.com">Consúltenos a info@dominio.com</a>`

## Referencias

Listas de HTML. Recuperando de <http://www.manualweb.net/html/listas-html/>

Enlaces HTML. Recuperando de <https://www.laurachuburu.com.ar/tutoriales/enlaces-html.php>