

HTML
Revoluciones



Multimedia

El contenido **multimedia** es el complemento perfecto para que nuestros sitios puedan **transmitir** información, ideas, conceptos **más allá de los textos**.

“Una imagen, vale más que mil palabras” y en **HTML** podemos contar con distintos **tipos** de multimedia para llevar nuestros sitios a otro nivel.

Imágenes

La etiqueta utilizada para definir una imagen es ****.

```
<img src= “./img/remeraCali.jpg” alt= “Foto de remera con estampa” />
```



Source: el valor indica la ruta donde se encuentra la imagen



Descripción de la imagen para accesibilidad.

*Formatos de imagen soportados: **jpg, jpeg, png, svg, webp, gif**.*

Otras etiquetas para imágenes.

<picture>

Nos da flexibilidad a la hora de trabajar con imágenes de distintos tamaños.

https://www.w3schools.com/tags/tag_picture.asp

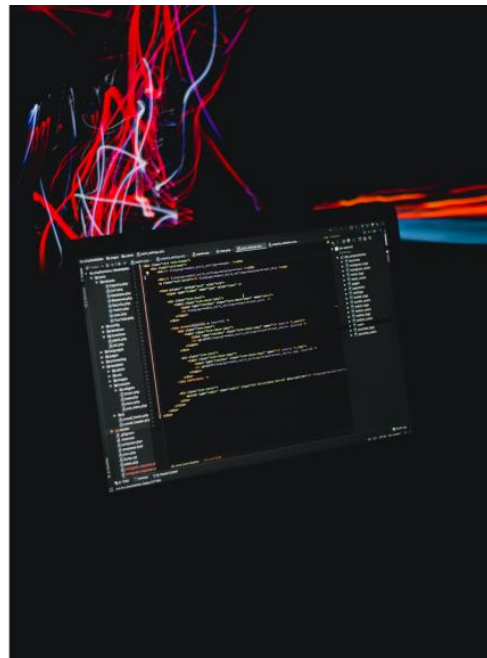
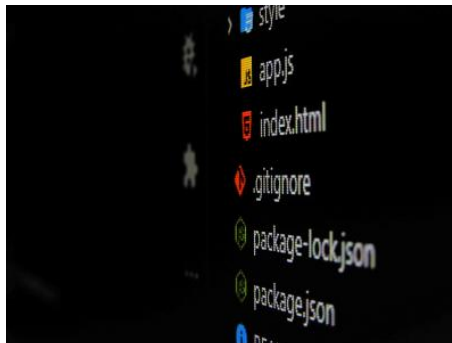
<figure>

Funciona como etiqueta contenedora de una imagen o foto.

https://www.w3schools.com/tags/tag_figure.asp

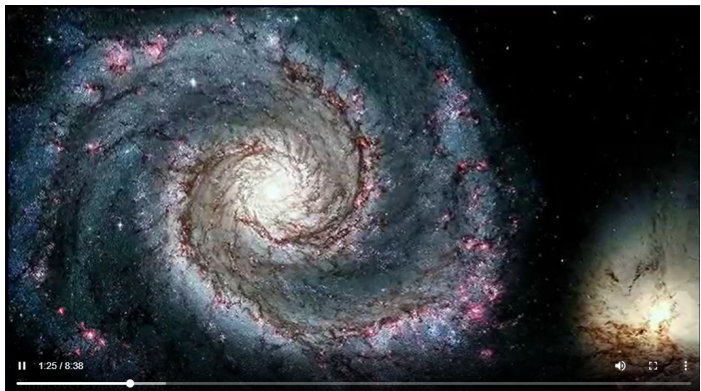
<figcaption>

Coloca un *caption* debajo de la imagen como información adicional.



Videos

```
<video controls autoplay>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
  El navegador no soporta el video  
</video>
```



Formatos de vídeo soportados: **mp4**, **ogg**, **webm**.

Atributos de la etiqueta video

controls

Atributo que habilita los controles de video.

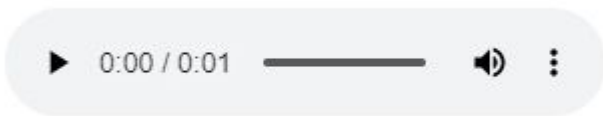
autoplay

El video se comenzará a reproducir solo (deshabilitado por algunos navegadores).

La etiqueta video también admite el atributo **src** en lugar de agregar una etiqueta **source**, como sucede en las imágenes.

Audio

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">
  El navegador no soporta el archivo.
</audio>
```



Formatos de vídeo soportados: **mp3**, **wav**, **ogg**.

Atributos de la etiqueta audio

controls (obligatoria)

Atributo que habilita los controles de audio.

autoplay

El audio se comenzará a reproducir solo (deshabilitado por algunos navegadores).

preload

Indica si el audio debe ser precargado o no al cargar la página.

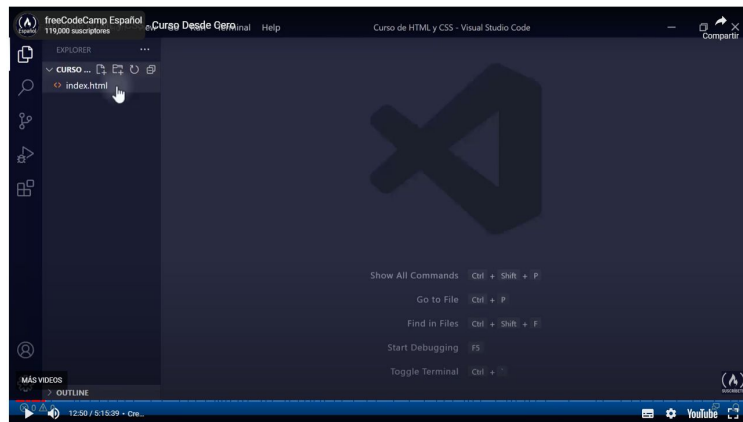
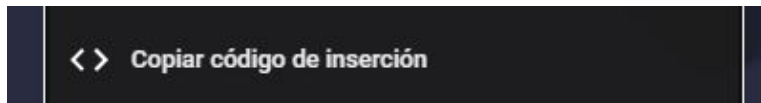
Iframe

El elemento HTML **<iframe>** (de inline frame) representa un contexto de navegación anidado, el cual permite incrustar otra página HTML en la página actual.

Por ejemplo *mapas*, *videos* o *porciones* de otros sitios web.

```
<iframe width="1280" height="720"
src="https://www.youtube.com/embed/XqFR2lqBYPs"
title="Aprende HTML y CSS - Curso Desde Cero"
frameborder="0" allow="accelerometer; autoplay;
clipboard-write; encrypted-media; gyroscope;
picture-in-picture" allowfullscreen></iframe>
```

Enlace al video de youtube: <https://www.youtube.com/watch?v=XqFR2lqBYPs>



Formularios

Tienen un papel **principal** en los sitios web ya que son capaces de **capturar** la entrada de **datos** del usuario, dando la posibilidad de **interacción** entre nuestra **aplicación** o página y una **persona**.

Gracias a ellos podemos **crear** sitios y **aplicaciones** web **dinámicas**, dependiendo del **usuario** y sus preferencias o comportamientos.

Etiqueta **<form>**

Esta etiqueta contiene todas las etiquetas internas de un formulario.

```
<form action="" method="" enctype="" >
  <!-- Entrada de datos a través de etiquetas -->
</form>
```

action

Indica el archivo que manejará la información enviada.

method

Puede ser **POST** o **GET** e indica si los datos se enviarán por **URL** u ocultos en la consulta.

enctype

Su valor contiene el tipo de información que se va a enviar:

text/plain: texto plano (no recomendado)

application/x-www-form-urlencoded: convierte caracteres especiales a ASCII para evitar inyecciones SQL.

multipart/form-data: necesario cuando además de texto enviamos archivos adjuntos.

Etiquetas de formulario

`<input>`

Es la más genérica de todas. Acepta un atributo **type** que indica el **tipo de dato** que recibirá esa etiqueta, un **name** donde se asigna el nombre que llevará ese **dato** al viajar y un **id** con el fin de **identificar** esa etiqueta de forma única.

`<label>`

Etiqueta que contiene un texto **asociado** a un **input** vinculado a través del atributo **for**.

```
<form action="index.js" method="GET">
  <label for="user">Usuario:</label>
  <input type="text" name="user" id="user">

  <label for="password">Contraseña:</label>
  <input type="password" name="password"
id="password">

  <input type="submit" value="Ingresar a mi
cuenta">
</form>
```

Usuario:

Contraseña:

Etiquetas de formulario

<input type="text">

Entrada de texto en una sola línea.

<input type="password">

Entrada de texto oculto.

<input type="email">

Entrada de correos electrónicos, al enviar solicita que su value contenga un @.

<input type="number">

Entrada de datos numéricos.

<input type="date">

Despliega un calendario para seleccionar una fecha.

Más info: https://www.w3schools.com/tags/tag_input.asp

<input type="file">

Permite cargar un archivo.

<input type="color">

Brinda color picker para seleccionar un color.

<textarea>

Representa un campo que nos permite introducir textos multilínea.

<button>

Es igual que los input de los tipos **submit**, **button** y **reset** solo que este posee etiqueta de apertura y cierre para colocarle contenido adicional que no sea texto.

Etiqueta `<input type="radio">`

Elementos de selección **simple** o **única**, por ejemplo cuando creamos encuestas con preguntas de una sola respuesta posible.

```
<h2>¿Qué lenguaje de programación se usa del lado Frontend? </h2>
<form action="index.js" method="GET">
  <input type="radio" name="preguntaUno" id="preguntaUno" value="python">
  <label for="">Python</label>
  <input type="radio" name="preguntaUno" id="preguntaUno" value="ruby">
  <label for="">Ruby</label>
  <input type="radio" name="preguntaUno" id="preguntaUno"
value="javascript">
  <label for="">Javascript</label>
  <input type="radio" name="preguntaUno" id="preguntaUno" value="java">
  <label for="">Java</label>
  <input type="radio" name="preguntaUno" id="preguntaUno" value="elixir">
  <label for="">Elixir</label>
</form>
```

¿Qué lenguaje de
programación se usa
del lado Frontend?

- ☐ Python
- ☐ Ruby
- ☒ Javascript
- ☐ Java
- ☐ Elixir

Etiqueta `<input type="checkbox">`

Elementos de selección **múltiple**, por ejemplo cuando una misma pregunta tiene **varias opciones** posibles de respuesta.

```
<h2>¿Qué lenguaje de programación se usa para el Backend? </h2>
<form action="index.js" method="GET">
  <input type="checkbox" name="preguntaUno" id="preguntaUno"
value="python">
  <label for="">Python</label>
  <input type="checkbox" name="preguntaUno" id="preguntaUno" value="ruby">
  <label for="">Ruby</label>
  <input type="checkbox" name="preguntaUno" id="preguntaUno"
value="csharp">
  <label for="">C#</label>
  <input type="checkbox" name="preguntaUno" id="preguntaUno" value="java">
  <label for="">Java</label>
  <input type="checkbox" name="preguntaUno" id="preguntaUno"
value="elixir">
  <label for="">
</label>
</form>
```

¿Qué lenguaje de programación se usa para el Backend?

- ☒ Python
- ☒ Ruby
- ☒ C#
- ☒ Java
- ☒ Elixir

Etiqueta <select>

Nos permite crear una lista desplegable de opciones, cada opción estará contenida como hija dentro de un elemento **<option>** donde vamos a encapsular cada opción de la lista.

```
<form action="index.js" method="GET">
  <select name="brands" id="brands">
    <option value="ferrari">Ferrari</option>
    <option value="dodge">Dodge</option>
    <option value="chevrolet">Chevrolet</option>
    <option value="volkswagen">Volkswagen</option>
    <option value="alfaRomeo">Alfa Romeo</option>
  </select>
</form>
```



Atributos de las etiquetas de formulario

En muchos de los elementos podemos añadir (opcionalmente) otros atributos como:

required: Valida que el campo esté completo antes de enviar la información.

placeholder: Texto soporte que aparece dentro de un input e indica un ejemplo de cómo llenar ese campo.

value: Para introducir un valor por defecto en el campo.

readonly: Si queremos que sea de sólo lectura.

Botones de formulario

Estos elementos son los que nos van a permitir enviar, limpiar o accionar sobre nuestros formularios.

Existen 3 tipos: **submit**, **reset** y **button** que pueden agregarse mediante una etiqueta **<input>** o una etiqueta **<button>**, la diferencia radica en que en la primera el texto se coloca en el atributo value mientras que en la segunda, el texto va dentro de la etiqueta.

submit: Ejecuta la acción de enviar los datos al archivo indicado en **action**

reset: Limpia todos los campos del formulario.

button: Carece de efecto, sirve para darle un comportamiento propio desde javascript.

```
<form action="index.js">
  <input type="submit" value="Enviar">
  <button type="submit">Enviar</button>
</form>
```