[HTML picture tag (w3schools.com)](https://www.w3schools.com/tags/tag_picture.asp) css

**Qué aprenderás sobre HTML y CSS**  1/55

[El camino para convertirte en Desarrollador Web este 2018 | by Martin Mucito | Code Army](https://codearmy.co/el-camino-del-desarrollador-web-en-2018-64efc5b68853)

* [HTML html tag (w3schools.com)](https://www.w3schools.com/tags/tag_html.asp) (etiquetas de ayuda)

En resumida cuenta según lo que he podido entender y explicándolo con analogías.  
.  
HTML sería la estructura, como si fuese los esqueletos de un ser humano o los cimientos de un edificio, de las páginas web sin colores, sin diseño.  
.  
CSS sería lo estético como la piel de ser humanos, la pintura y adornos de los edificios o la manera como es posible que la estructura del html se vea mejor para la vista de los usuarios.  
.  
JS seria los músculos lo que da esa interactividad a las personas para moverse y hacer acciones como correr.  
example: [css html js-c5cad547-d054-48d7-83be-bc2caee99577.jpg (472×485) (platzi.com)](https://static.platzi.com/media/user_upload/css%20html%20js-c5cad547-d054-48d7-83be-bc2caee99577.jpg)

<https://commondenominator.email/wp-content/uploads/2017/06/javascript-seo-making-your-bot-experience-as-good-as-your-user-experience.gif>

**¿Qué es Backend?**

3/55

**Back-end**  
Es lo opuesto a un **Front-end**, ya que el Back-end trabaja en el lado del servidor mientras el Front-end en el lado del cliente.  
Es el responsable de manejar toda la lógica que hay detrás de una petición dada por el navegador.  
Una característica que lo diferencia del **Front-end** es que no tiene estándares, ya que tiene varios lenguajes de programación **(Node.js, Python, PHP, Ruby, GO, Java, .NET entre otros)** . Los cuales estos lenguajes tienen sus propios frameworks como Django (Python), Lavarel (PHP), Rails (Ruby), Express (JS), Spring (Java).  
El **Back-end** tambien tiene que tener en cuenta la infraestructura, donde va a generar el deploy de su aplicación (esto tambien puede ser tarea de un DevOps, un perfil dedicado a la infraestructura). Con tecnologías como **Google Cloud, DigitalOcean, AWS y Heroku**.  
Y por último entramos en Bases de datos, teniendo las relacionales **(mongoDB)** y las no-relacionales. **(MySQL)**.  
Ahora que tienes más noción de que trata cada uno… **¿Cual te llama más la atención?** **¿Cual te gustaría aprender?** ¡En los comentarios podemos seguir esta charla! 😃😃

**HTML: anatomía de una página web**

6/55

[Web Anatomy-296b297a-6db0-4747-b96d-bcf4a7f5e569.jpg (1654×922) (platzi.com)](https://static.platzi.com/media/user_upload/Web%20Anatomy-296b297a-6db0-4747-b96d-bcf4a7f5e569.jpg)

**Index y su estructura básica: body**

8/55

**body** es la parte visible de nuestro website 😃. Es muy importante usar HTML semántico y no llenar todo de **<div>** para que nuestro sitio sea mejor interpretado por el navegador y por lo tanto más accesible.

<body>

<header> <!--Sección superior de nuestro website-->

<nav></nav> <!--Sección de navegación de nuestro website, siempre dentro del header-->

</header>

<main> <!--Main es el contenido central de nuestro website, "la parte del medio"-->

<section>

<!--Nuestro website puede estar divido por secciones, por ejemplo platzi tiene 3: El navegador de cursos y rutas, el feed y nuestras rutas de aprendizaje-->

<article>

<!--Contenido independiente de la página. Es reutilizable-->

</article>

</section>

<ul> <!--Lista desordenada: Sin numerar-->

<li><!--Item List. Elementos de la lista--></li>

</ul>

<ol></ol> <!--Lista ordenada: Numerada-->

</main>

<footer> <!--Sección final de nuestro website-->

</footer>

<p>Soy un texto</p> <!--Párrafo, texto-->

<h1>Soy un titulo</h1>

<!--Títulos, muestran el texto más grande y con negrilla. Existen desde el h1 al h6-->

<a href="#">Soy un link</a>

<!--Enlaces/links que nos permitirán movernos entre páginas.-->

</body>

**Optimización de imágenes**

12/55

Y de donde descargar imágenes ??? Pues te traigo un pack de páginas donde puedes descargar imágenes libre de copyright, en algunas puedes elegir el tamaño de la imagen 😉 🚀🚀🚀🚀🚀

[Pexels](https://www.pexels.com/es-es/)  
[Pixabay](https://pixabay.com/)  
[Freerange](https://www.freerangestock.com/)  
[Realistic Shots](https://realisticshots.com/)  
[PicJumbo](https://picjumbo.com/)  
[Magdeleine](https://magdeleine.co/)  
[Creativecommons](https://search.creativecommons.org/)  
[Freejpg](https://www.freejpg.com.ar/)

**Compresores de imágenes:**

[Optimizilla](https://imagecompressor.com/es/)  
[IloveIMG](https://www.iloveimg.com/es)

[TinyPNG – Compress WebP, PNG and JPEG images intelligently](https://tinypng.com/) Ayuda a modificar el tamaño de imágenes a uno mucho más optimo a tu proyecto

[Ver Exif online , quitar Exif online](https://www.verexif.com/) cada imagen que subas estas te sirvan para pesar menos El promedio **optimo** del **peso** de las **imágenes** debe ser de **70 KB.** Elimina los metadatos que puedan traer las imágenes (como fecha y ubicación) para reducir su tamaño

**Etiqueta img**

13/55

[Nature | Unsplash](https://unsplash.com/t/nature) <https://unsplash.com/t/nature> (para descargar imagenes de dif calidad libre de derechos de autor

[Fotos de stock gratis, imágenes libres de regalías y sin derech ttps://www.pexels.com/es-es/hos de autor · Pexels](https://www.pexels.com/es-es/)

  <body>

    <main>

      <section>

        <figure>

          <img

            src="./tiburonn.jpg"

            alt="Es una imagen de un tiburon" alt es para que aparezca descripción si la img no apareze

          />

          <figcaption>tiburonn</figcaption>

        </figure>

      </section>

    </main>

  </body>

**Etiqueta figure**

14/55

Hay una etiqueta muy cool que descubrí hace poco llamada <picture>, es ideal para usarla cuando quieres tener más flexibilidad al momento de especificar distintas imágenes, si quieres tener un diseño responsive en tu web.  
Dentro de ella van dos etiquetas:  
<source> (y sus atributos principales serían media: para especificar el tamaño y srcset: para indicar la url de la imagen  
<img> (con los atributos que ya explicó el profe Diego. Y esta sería la imagen principal)

En esta página, por ejemplo, se puede apreciar cuando paso el cursor sobre la url de cada imagen

**Etiqueta video**

15/55

La etiqueta <video>, tiene algunos atributos como:

controls: agrega al video los controles necesarios para reproducir, pausar y adelantar.

preload = auto: hace que el navegador descargue el video, en el momento en el que se acceda a la página.

La etiqueta <source>, se puede colocar dentro de una etiqueta <video> varias veces, para especificar diferentes rutas. Esto La etiqueta **<video>**, tiene algunos atributos como:

1. **controls:** agrega al video los controles necesarios para reproducir, pausar y adelantar.
2. **preload = auto:** hace que el navegador descargue el video, en el momento en el que se acceda a la página.  
   La etiqueta **<source>,** se puede colocar dentro de una etiqueta **<video>** varias veces, para especificar diferentes rutas. Esto para asegurar que cualquier navegador pueda mostrar el video

para asegurar que cualquier navegador pueda mostrar el video

**Controles y semántica básica**

* La semántica y estructura de un video es la siguiente:

<video src="rutaVideo"></video>

* + En este caso podemos ver la previsualización del video, pero si queremos agregarle controles al mismo, debemos agregarle el atributo **controls,** de esta forma:

<video src="rutaVideo" controls></video>

* Por otro lado tenemos la etiqueta prelaod, la cual nos va a funcionar a que nuestro video se renderice mientras la página se cree en el navegador. 📉

<video src="rutaVideo" controls preload="auto"></video>

* Además tenemos una opción muy interesante a la hora de cargar un video, si quisieramos tener un minuto con segundo específico de inicio y uno del final lo podemos hacer de esta forma:

<video src="rutaVideo#t=1.06, 1.40" controls preload="auto"></video>

* Si queremos colocar distintos formatos del video para tener una mayor compatibilidad con los distintos navegadores, podemos usar la etiqueta source de esta forma.

<video controls preload="auto">

<source src="rutaVideo#t=1.06, 1.40"/>

<source src="rutaVideo2#t=1.06, 1.40"/>

<source src="rutaVideo3#t=1.06, 1.40"/>

</video>

SHIFT + ALT + “Direccional Abajo”

 (hacienda click desde aca)<title>video tik tok</title>

 <title>video tik tok</title>

 <title>video tik tok</title>

<body> <main> <section> <!-- Control= "nos da los botones" \*\*\*\* preload ="permite <body>

<main>

<section>

<!-- Control= "nos da los botones" \*\*\*\* preload ="permite que el video se

auto carge antes de para una mejor experiencia del lado cliente" -->

<video controls preload="auto">

<!-- Para no tener problemas en los navegadores

defino las extenciones -->

<source src="./miVideo.mp4#t=10,60" />

<source src="./miVideo.mpv#t=10,60" />

<!-- Para darle un tiempo inicio a fin lo hacemos con el # despues t="como quiero

que inicie" lo separo con , despues le doy el otro tiempo donde quiero que

finalice-->

</video>

</section>

</main>

</body>```que el video se auto carge antes de para una mejor experiencia del lado cliente" --> <video controls preload="auto"> <!-- Para no tener problemas en los navegadores defino las extenciones --> <source src="./miVideo.mp4#t=10,60" /> <source src="./miVideo.mpv#t=10,60" /> <!-- Para darle un tiempo inicio a fin lo hacemos con el # despues t="como quiero que inicie" lo separo con , despues le doy el otro tiempo donde quiero que finalice--> </video> </section> </main> </body>```