

# Esta clase va a ser

- grabada

Certificados oficialmente por



**CODERHOUSE**

Clase 02. DESARROLLO WEB

# Primeros pasos con HTML

Certificados oficialmente por



**CODERHOUSE**

# Objetivos de la clase



**Comprender** la arquitectura básica de un sitio web.



**Identificar y utilizar** elementos constitutivos de una página web moderna.



**Conocer y aplicar** elementos HTML para interacción con el usuario.

# Glosario

**Sketch:** es un dibujo rápido o bosquejo guía, que reproduce de manera muy sencilla un concepto, una idea o generalidad de un proyecto.

**Wireframe:** es la representación estática, en baja calidad, de un diseño.

**Mockup:** es la representación estática de un diseño, en calidad media o alta.

**Prototipo:** es la representación navegable del producto final.

**HTML:** es un "lenguaje" de marcado de etiquetas, que permite crear documentos para web.

**Doctype:** cuando escribes tu documento HTML, lo primero que debes hacer es escribir el DOCTYPE, el cual declara el tipo de documento. Es decir, sirve para indicar que tu documento está escrito siguiendo la estructura determinada por un DTD concreto. Un DTD es la definición del tipo de documento.

# Temario

01

## Prototipado y conceptos básicos de HTML

- ✓ Prototipado desde papel
- ✓ HTML
- ✓ Etiquetas

02

## Primeros pasos con HTML

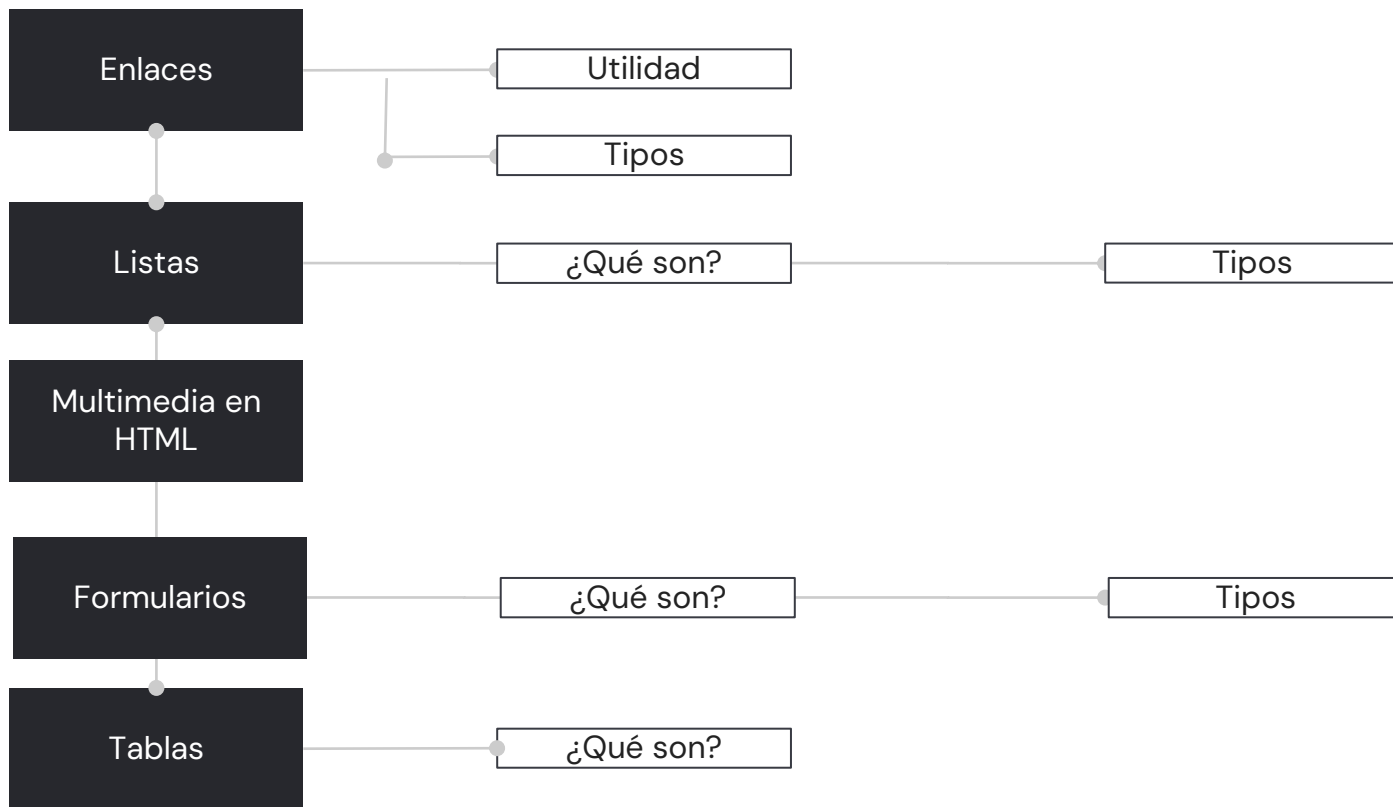
- ✓ [Enlaces](#)
- ✓ [Multimedia con HTML](#)
- ✓ [Listas](#)
- ✓ [Formularios](#)

03

## Incluyendo CSS a nuestro proyecto

- ✓ Bases de CSS
- ✓ Sintaxis de CSS
- ✓ Herencia y cascada

## MAPA DE CONCEPTOS



# Enlaces

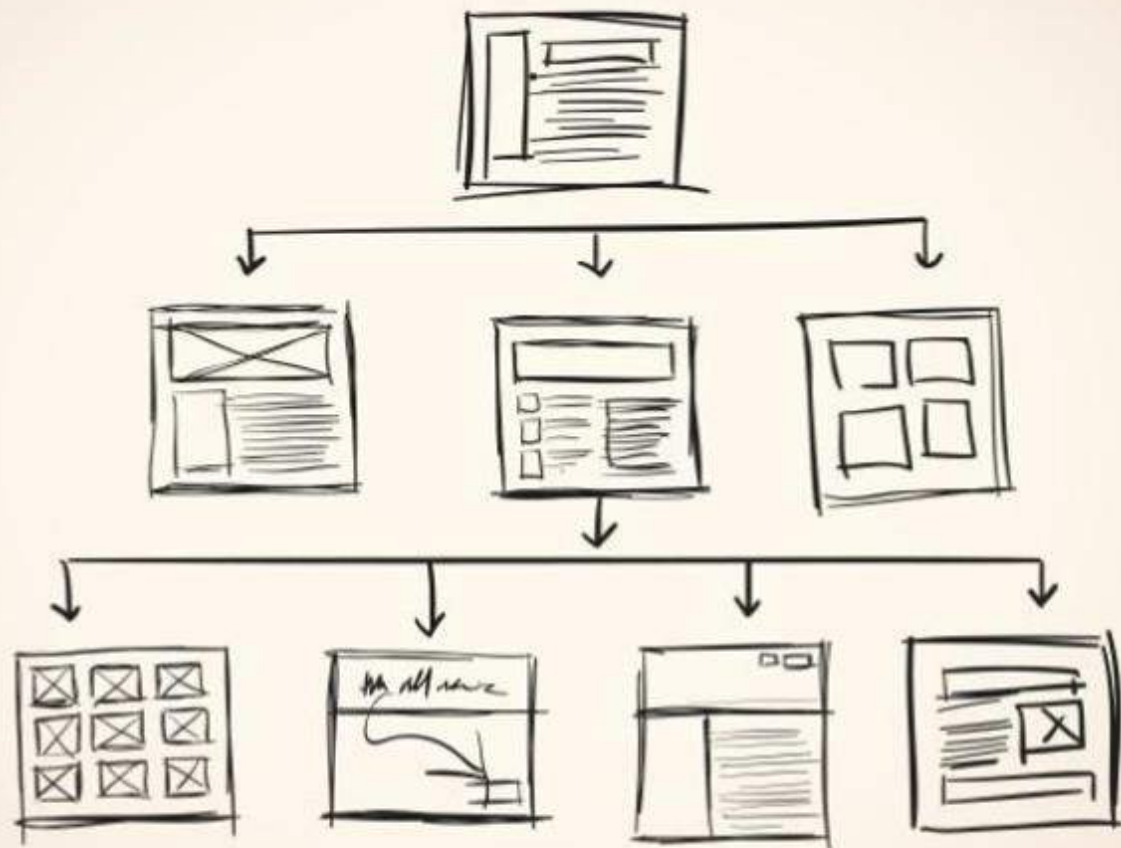
# Arquitectura de un sitio

Los enlaces, también conocidos como links o anchors, se utilizan para relacionar partes del mismo documento. Por defecto, se visualizan azules y subrayados.

Para crear uno, es necesario utilizar la etiqueta de ancla <a> con el atributo "href", que establecerá el destino al que apunta. Por ejemplo:

```
<a href="productos.html">Productos</a>
```





# Enlaces absolutos, relativos e internos



# Enlaces absolutos

Los enlaces absolutos son aquellos cuyo destino apunta a un documento que está fuera del sitio, y debe ser especificado utilizando la URL completa:

```
<a href="https://www.coderhouse.com/online/desarrollo-web-online">Curso de  
Desarrollo Web</a>
```



# Enlaces relativos

Los **enlaces relativos** son aquellos que apuntan a páginas ubicadas dentro del mismo proyecto. Si la página referenciada se encuentra en el mismo directorio, alcanza con mencionar el nombre de la misma para generar el enlace.

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

En caso de que el archivo se encuentre en un directorio (carpeta) específico, el mismo deberá ser mencionado.

```
<a href="imagenes/mapa.jpg">ver mapa</a>
```



# Enlaces relativos

También puedes usar como destino una sección específica una página distinta:

```
<a href="contacto.html#formulario">Formulario de contacto</a>
```

En el ejemplo anterior, el enlace apunta a la sección que tiene el id “formulario” dentro de la página “contacto.html”. No sólo es posible utilizar etiquetas <a> con texto, también puedes hacerlo con otros elementos. Veamos un ejemplo de enlace con una imagen:

```
<a href="https://www.coderhouse.com">  
    
</a>
```



# Enlaces internos

Los enlaces internos permiten referenciar secciones de la misma página, para lo cual se utiliza el id:

```
<a href="#pie">Ir al pie de página</a>  
...  
<footer id="pie"></footer>
```

El **id** es un **atributo** de los elementos html, que nos permite identificar de manera inequívoca a dicho elemento entre todos los demás que conforman el documento. Por lo tanto, el **valor** de dicho atributo debe ser **único para cada elemento**, y no se debe repetir a lo largo del documento.

# Multimedia en HTML

# Imágenes

Enriquecen el HTML: las imágenes son elementos que, bien utilizados, mejoran la experiencia de los usuarios.

The screenshot displays the Coderhouse live class interface. On the left, a dark sidebar contains the text 'Unite a la comunidad de aprendizaje online en vivo más grande de Latinoamérica' in white and yellow, followed by a description of the live classes and a yellow button labeled 'Ver todos los cursos'. The main area shows a video feed of a female instructor at a laptop, with three smaller video thumbnails above it. A right-hand sidebar lists participant avatars and names. The top of the interface features the Coderhouse logo and a 'Difundir en vivo' button.

**Unite a la comunidad de aprendizaje online en vivo más grande de Latinoamérica**

Clases online en vivo dictadas por expertos de la industria, enfoque 100% práctico, mentorías personalizadas y acceso a una comunidad de +50.000 estudiantes.

[Ver todos los cursos](#)

Calidad 100% asegurada

**CODERHOUSE**

Difundir en vivo

Código en vivo: **Diseño UX/UI**

Participantes:

- Sandra
- Francisco
- Mica
- Paul
- Pablo



# Insertar imágenes

✂ Se insertan con la etiqueta `<img/>`, que pertenece al grupo de las etiquetas abiertas, autocontenidas o que “se cierran a sí mismas” (con la barra al final).

📍 Para funcionar requiere, como mínimo, indicar en dónde está el archivo a mostrar. Eso se hace con el atributo “src” (el source o fuente), que respeta todas las reglas de ruteo vistas en los links.

👥 Se comportan como elementos de línea, esto significa que se verán una al lado de la otra.

# Atributo “alt”



El “alt” es un texto que debe representar la foto que se está visualizando. Tiene que ser conciso y breve, pero dejar en claro de qué se trata la imagen.

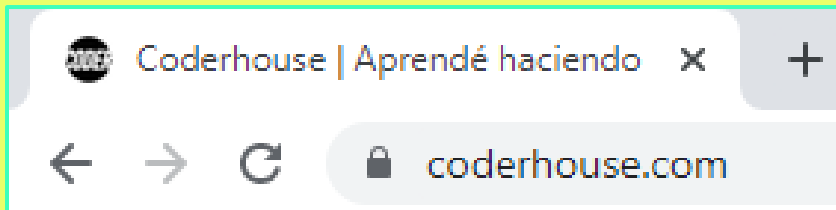
```

```

# Favicon



El favicon es un pequeño ícono que identifica un sitio web cuando está abierto en una pestaña o cuando es guardado como favorito.  
HTML recomendado para agregar dentro de tu etiqueta <head>:



```
<link rel="shortcut icon" href="favicon.png">
```



# Etiqueta iframe

Es un elemento HTML que permite insertar o incrustar un documento HTML dentro de un documento HTML principal.

```
<iframe src="pagina_fuente.html"
width=290 height=250>Texto para
cuando el navegador no conoce la
etiqueta iframe</iframe>
```

# Etiqueta iframe



Insertar vídeo

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed/
P03IJBtrFyc" title="YouTube video
player" frameborder="0"
allow="accelerometer; autoplay;
clipboard-write; encrypted-media;
gyroscope; picture-in-picture"
allowfullscreen></iframe>
```

☐ Empezar en 0:12

COPIAR



## Ejemplo en vivo

Página con contenido multimedia

- ✓ Crearemos un archivo HTML con contenido multimedia

Duración: **15 minutos**

# Listas

# Listas

HTML permite agrupar elementos que tienen más significado de forma conjunta. El menú de navegación de un sitio web, por ejemplo, está formado por un grupo de palabras.

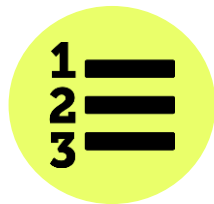
Aunque cada palabra por separado tiene sentido, de forma conjunta constituyen el menú de navegación de la página, por lo que su significado conjunto es mayor que por separado. Esto se denomina listas.



# Tipos de listas



**Listas no ordenadas**



**Listas ordenadas**



**Listas de definición**

Ejemplos: paso a paso de un procedimiento (como una receta de cocina), características de una persona, galería de imágenes, el menú de una página web, entre otros.

# ¿Viñetas o números?

- ✓ Las listas numéricas establecen un orden en la lectura de sus ítems.
- ✓ Las listas de viñetas no representan ningún orden o importancia entre sus ítems. Son elementos compuestos.



`<ol>`

Crea una lista ordenada de artículos (numéricas).



`<ul>`

Crea una lista de artículos sin orden (viñetas).



`<dl>`

Crea una “lista de definiciones” (por ejemplo: un diccionario).

# Veamos un ejemplo



Ambas listas se deben insertar mediante la etiqueta `<li></li>` (list-item). Ejemplo de servicios de una empresa (lista de viñetas/sin orden):

```
<ul>  
  <li>Empresa</li>  
  <li>Producto</li>  
  <li>Servicios</li>  
  <li>Contacto</li>  
</ul>
```

- ✓ Empresa
- ✓ Producto
- ✓ Servicios
- ✓ Contacto



# Anidar listas

Es probable que te veas en la necesidad de crear una estructura de sublistas como la siguiente:

- ✓ Computadoras portátiles:
  - Procesador I4.
  - Procesador I5.
- ✓ Computadoras de escritorio:
  - Procesador Pentium.
  - Procesador Celeron.



## Ejemplo en vivo

### Listas

- ✓ Crearemos dos listas usando las etiquetas de HTML.

Duración: **10 minutos**

# ¿Preguntas?

**#CoderTip:** Ingresa al [siguiente link](#) y revisa el material interactivo que preparamos sobre Preguntas Frecuentes, estamos seguros de que allí encontrarás algunas respuestas.



## Para pensar

HTML no es un lenguaje de programación

**¿Verdadero o falso?**

Contesta mediante el chat de Zoom



# Break

¡10 minutos y volvemos!



# Formularios

# Definición

Son etiquetas donde el usuario ingresará o seleccionará valores, que serán enviados a un archivo encargado de procesar la información.



The image shows a 'Contact Us' form with a yellow border. It contains three teal input fields: 'Name', 'Email', and 'Message'. The 'Message' field is a larger text area with a vertical scrollbar. Below these fields is an orange 'Submit' button.

# Etiqueta <form>

Para insertar un formulario se usa la etiqueta <form>, que dentro lleva todos los controles que vayan al mismo destino. Un formulario requiere 3 atributos para funcionar:

- ✓ **Action:** documento que se encarga de recibir los datos y procesarlos.
- ✓ **Method:** la forma en que será enviada la información. Existen dos métodos de envío, que son GET y POST.
- ✓ **Enctype:** cómo se codificarán los contenidos.

Por ejemplo:

```
<form action="procesar_form.php" method="post"
enctype="text/plain">
    <!-- Campos del form -->
</form>
```

# Atributos del form

# Action

En este atributo se indicará cuál es el archivo que recibe y procesa los datos. Debe ser de un lenguaje de los llamados “del lado del servidor” (PHP / ASP / JSP). Si no se indica un valor, el Action será por defecto el mismo archivo donde está el formulario.

# Method

Forma en la que se recopilan y envían los datos. Existen dos métodos comunes en el HTML:

- ✓ **GET:** la información viajará por la barra de direcciones a continuación del nombre del archivo.
- ✓ **POST:** la información viajará junto a los encabezados del HTML (será "invisible").

Si el method no se indica, por defecto será GET.

# Enctype

Cuando el valor del atributo method es post, el mismo es el tipo MIME del contenido, que es usado para enviar el formulario al servidor. Los posibles valores son:

- ✓ application/x-www-form-urlencoded: será el valor por defecto si un atributo no está especificado.
- ✓ multipart/form-data: usar este valor si se está usando el elemento input con el atributo type ajustado a "file".
- ✓ text/plain (HTML5)

Normalmente, se utiliza para permitir el envío de archivos a través de un formulario.

# Controles del form



# Ingreso de texto

Existen tres controles generales para el ingreso de texto:

- ✓ Cajas de texto de una sola línea (no acepta el uso de la tecla Enter).
- ✓ Cajas para el ingreso de contraseñas (el contenido no será visible).
- ✓ Cajas para contenido multilínea. Puede ser una o muchas líneas de texto.

Atributo "name"

Control de formulario: `<input>`: Text, Email, Password.

Control de formulario: `<textarea></textarea>`

# Botones

Los botones disparan las acciones del formulario. Hay 3 tipos:

- ✓ El que envía los datos al archivo indicado como Action.
- ✓ El que vacía todo lo ingresado y resetea los campos.
- ✓ El que “no hace nada”, pensado para usarse con Javascript.

Todos los botones son etiquetas `<input>`, con distintos tipos de “Type”. El botón debe de estar dentro del `<form>` que afectará.



# Atributo "value"

Representa la etiqueta del botón, la cual es normalmente mostrada por los navegadores dentro de éste.

```
<form>
  <input type="submit"
value="Enviar
formulario"/>
  <input type="reset"
value="Limpiar
formulario"/>
  <input type="button"
value="Sin acciones"/>
</form>
```

- ✓ Input de tipo "submit": envía el formulario.
- ✓ Input de tipo "reset": resetea el formulario.
- ✓ Input de tipo "button": no tiene acciones por defecto.

Profundiza y conoce atajos en  
nuestra [Cheat Sheet](#)



# Controles de selección

En estos casos, el usuario no puede ingresar libremente un texto, sino que el programador le da una lista predefinida. El dato que llega al elegir una opción se define desde el atributo "value". Existen 3 grupos de controles de selección:

- ✓ Botones de radio: sólo se puede elegir una opción.
- ✓ Casillas de chequeo: de toda la lista de opciones, el usuario puede optar por una, todas o ninguna opción.
- ✓ Menú desplegable: sólo es posible seleccionar una opción.



# Atributo "value"

En este caso es el valor que se enviará al enviarse el formulario.

✓ Botones de radio:

```
<form>  
  <div>Turno diurno</div>  
  <input type="radio" name="turno" value="diurno" />  
  <div>Turno nocturno</div>  
  <input type="radio" name="turno" value="nocturno" />  
</form>
```



# Etiqueta <label>

Define formalmente a cada elemento de un formulario. Esta etiqueta es de mucha ayuda para generar un formulario accesible.

**Su principal atributo es "for", que indica a cuál elemento del formulario corresponde la <label>.** El valor del atributo "for" debe ser igual al valor del atributo "id" o "name" del elemento.

```
<form>  
  <label for="nombre">Nombre:</label>  
  <input type="text" name="nombre" />  
</form>
```



# Menú desplegable

Es el llamado combo-box, selector o menú. De toda la lista, se puede elegir una opción (aunque tiene un atributo que permite cambiarlo). Lo ideal es que sean al menos dos elementos distintos para observar el select:

```
<form>  
  <select name="talles">  
    <option value="L">Large</option>  
    <option value="M">Medium</option>  
    <option value="S">Small</option>  
  </select>  
</form>
```



# Conjunto de campo



Las etiquetas `<fieldset>` y `<legend>` se utilizan en conjunto. La primera, tiene como objetivo crear grupos de elementos del formulario que posean un mismo propósito; mientras que la segunda, define formalmente el propósito del elemento `fieldset`. Se estructuran de la siguiente manera:

```
<form>
  <fieldset>
    <legend>Talle de remeras</legend>
    <!-- Aquí irán los elementos de formulario -->
  </fieldset>
</form>
```



## Ejemplo en vivo

Formularios

✓ Crearemos un formulario de contacto

Duración: **15 minutos**

# Tablas

# Tablas

Son un conjunto de celdas organizadas, dentro de las cuales es posible alojar distintos contenidos. HTML dispone de una gran variedad de etiquetas y atributos para crear tablas. Sirven para representar información tabulada, en filas y columnas.

En HTML4 las tablas se usaban para maquetar. Cuando CSS creció y se hizo más fuerte, nacieron los detractores de las tablas.

# Etiquetas básicas para tablas en HTML

Fila 1 - Columna 1	Fila 1 - Columna 2	Fila 1 - Columna 3	Fila 1 - Columna 4	Fila 1 - Columna 5
Fila 2 - Columna 1	Fila 2 - Columna 2	Fila 2 - Columna 3	Fila 2 - Columna 4	Fila 2 - Columna 5
Fila 3 - Columna 1	Fila 3 - Columna 2	Fila 3 - Columna 3	Fila 3 - Columna 4	Fila 3 - Columna 5

Diagram illustrating the basic HTML tags for tables:

- `<table>` (blue box) points to the entire table structure.
- `<tr>` (pink boxes) points to each row of the table.
- `<th></th>` (yellow box) points to the header row.
- `<td></td>` (pink box) points to the data cells within the rows.



# Etiquetas básicas para tablas en HTML

```
<table>
  <tr><!-- inicio de fila-->
    <td>Fila 1 - Columna 1</td>
    <td>Fila 1 - Columna 2</td>
    <td>Fila 1 - Columna 3</td>
  </tr><!-- cierre de fila -->
  <tr><!-- inicio de otra fila-->
    <td>Fila 2 - Columna 1</td>
    <td>Fila 2 - Columna 2</td>
    <td>Fila 2 - Columna 3</td>
  </tr><!-- cierre de la segunda fila -->
</table>
```



## #Coderalert

Ingresa al manual de prácticas y realiza la primera actividad "Estructura del proyecto". Ten en cuenta que el desarrollo de la misma será importante para la resolución del Proyecto Final.



## Actividad N° 2

# Estructura del proyecto

### Descripción de la actividad.

- ✓ Crea **5 archivos HTML** para cada sección del sitio web, aplica entre estos los enlaces relativos y escribe con etiquetas semánticas. Deberás incluir:
  - Una lista no ordenada con el menú principal del sitio web con los nombres de cada una de las secciones y replicarlo en las mismas.
  - Un formulario en la página de contacto.
  - Un footer y replicarlo en todos tus HTML.
  - Etiquetas multimedia como img e iframes.

Recuerda respetar el diseño que generaste en el Wireframe de la actividad anterior. Podrás encontrar un ejemplo en la carpeta de clase.



¿Preguntas?



# Encuesta

Por encuestas de Zoom

¡Terminamos el **módulo 1: HTML!**

Cuéntanos qué temas te resultaron más complejos de entender. **Puedes elegir más de uno.** Vamos a retomar aquellos temas que resultaron de mayor dificultad en el próximo AfterClass.

# Resumen

## de la clase hoy

- ✓ Arquitectura básica de un sitio web.
- ✓ Elementos constitutivos de una página web moderna.
- ✓ Elementos HTML para interacción con el usuario.

**Opina y valora**  
**esta clase**

**Muchas gracias.**

**#DemocratizandoLaEducación**