

Programador Web Inicial

Clase 02. HTML y CSS

Objetivos

- Conocer conceptos básicos del HTML y sus etiquetas.
- Armar la estructura básica de un documento HTML y conocer su sintaxis

LENGUAJES PARA EL DESARROLLO WEB

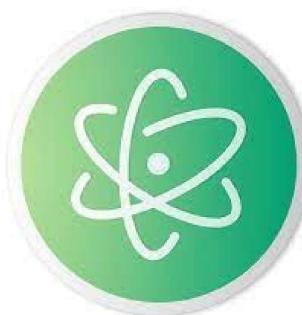


Para el diseño y desarrollo web existen diferentes lenguajes, que nos permiten llevar el diseño en papel (sketch) a una estructura que pueda interpretar un computador. En este curso se verán dos lenguajes bases: HTML y CSS

Editores de código



Brackets



Es un "lenguaje" de marcado de etiquetas, que permite crear documentos para web.

Durante el curso estaremos viendo HTML, incluyendo toda su estructura y etiquetas. Los siguientes términos serán de uso frecuente:

- Etiqueta.
- Atributo.
- Estructura.



HTML

El CSS permite controlar la apariencia de una página web.

El CSS, en español «hojas de estilo en cascada», es un **lenguaje de diseño gráfico**, utilizado para definir y crear la presentación de un documento estructurado, escrito en un lenguaje de marcado.

Algunos términos que utilizaremos serán:

- Estilo.
- Reglas.
- Medidas.
- Fuente.



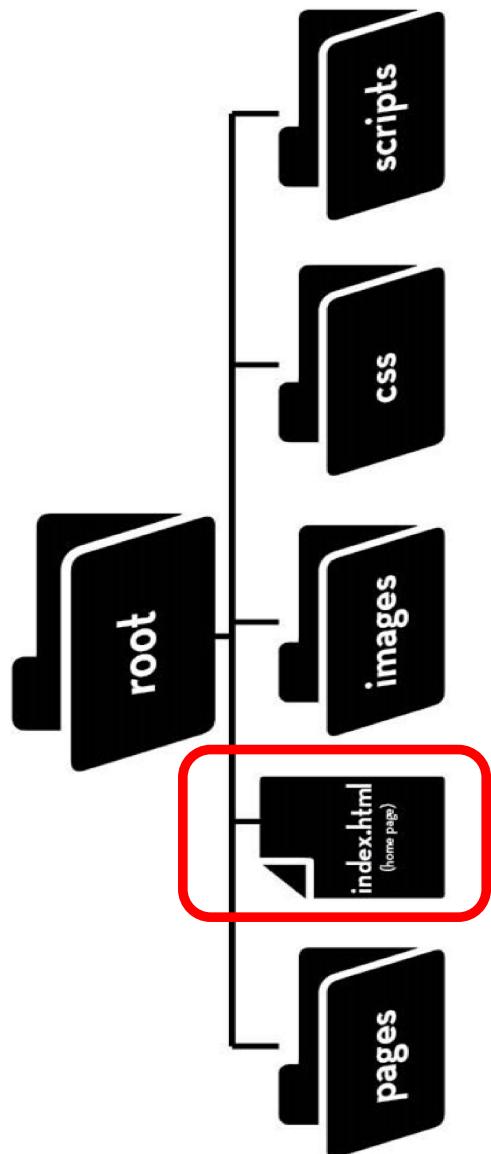
CSS

CONVENIO DE ARCHIVOS

- No deben tener espacios, acentos, eñes, ni símbolos o si son varias palabras usar guiones “ - ” O “ _ ” ej: “mi -pagina-web”
- Tienen que estar escritos en minúsculas.
- Deben tener la extensión “.html” (es la forma en que el servidor web sabe que se trata un documento web).

Atención: al guardar archivos en Windows, verificar que no se guarden con la doble extensión de archivos (**archivo.html.txt**).

— DOCUMENTO PREDETERMINADO



Es el archivo que se carga cuando todavía no has hecho en ningún link.

Debe llamarse **index.html** , ya que es el nombre estandarizado.

ETIQUETAS HTML

Las etiquetas HTML están delimitadas por un inicio y un final de cada elemento. Lo que se encuentra dentro de la etiqueta (el contenido) es lo que estás formateando.

Toda etiqueta es un juego de pares: una etiqueta abre, otra cierra.

```
<etiqueta>  
Contenido  
</etiqueta>
```

— ATRIBUTOS DE LAS ETIQUETAS

Todas las etiquetas aceptan atributos, esto es, cualquier característica que pueda ser diferente entre una etiqueta y la otra. El valor que tendrá va entre comillas. Cada una puede tener más de un atributo, separados entre sí por espacios. Los mismos sólo van en la de apertura.

```
<etiqueta atributo="valor">  
Contenido  
</etiqueta>
```

ETIQUETAS CERRADAS Y ABIERTAS

Las cerradas encierran un contenido, por lo general texto, mientras que las abiertas sirven, entre otras cosas, para incluir elementos como imágenes, líneas, entre otros.

En el ejemplo tenemos una etiqueta cerrada llamada Párrafo, que engloba un texto y una etiqueta abierta para incluir una línea horizontal. El signo “/” se utiliza para las etiquetas de

```
<p>Este es un párrafo con texto en su interior - Etiqueta Cerrada </p>
```

```
<hr/> <!-- Esto es una línea horizontal - Etiqueta abierta -->
```



ETIQUETAS ABIERTAS EN HTML5

Ya no es una obligación poner el signo “/” . Por ejemplo, funcionará correctamente, y lo mismo sucederá con
, <hr>, los meta tag.

ESTRUCTURA BÁSICA



<head> : es la parte privada del documento, que se utiliza como un espacio de comunicación entre el sitio web y el navegador. Esta etiqueta envuelve otras importantes como **<title>** , las etiquetas **<meta>** y aquellas relacionadas con la importación de documentos CSS y JS.

<body> : encierra el contenido propiamente dicho del sitio.

Ambos deben estar dentro de un elemento principal: la etiqueta **<html>** .

ESTRUCTURA BÁSICA

- **<html>** : etiqueta inicial, que define que el documento está bajo el estándar de HTML. Abre y cierra, por lo tanto es fundamental no olvidar la etiqueta **</html>** al finalizar el documento, pues sino no cargará correctamente el contenido de mi sitio.
- **<title>**: la etiqueta title define el título de la página, el cual será visualizado en la solapa del navegador.
- **<meta>**: se utiliza para añadir información sobre la página (ya sean palabras clave, el autor, la descripción del sitio, etcétera), la cual pueden valerse los buscadores. También puede definir el idioma y la codificación en la cual está escrita la página.

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.

```
<!DOCTYPE html>
```

Tips: <https://www.w3.org/QA/Tips/Doctype.html.es>

```
1 <!DOCTYPE html> <!-- Tipo de Documento-->
2 <!--
3 <html lang="en"> <!-- Etiqueta inicial-->
4 <!--
5   <!-- Cabecera, recursos y
6   datos importantes sobre nuestra web-->
7 <head>
8   <meta charset="UTF-8">
9   <title>...</title>
10 </head>
11 <!--
12 <body>
13 </body>
14 <!--
15 </html> <!-- Fin de nuestra web-->
```

TIPOS DE ETIQUETAS: GRUPO GENERAL

Todas las etiquetas que van dentro del <body></body> se dividen en dos grupos:

- Elementos de bloque: son aquellos que, sin ser modificados por CSS, ocupan el 100% del ancho del contenedor, y se mostrarán uno abajo del otro.
- Elementos de línea: sólo ocupan el ancho que diga el contenido, y se verán uno al lado del otro.

```
<h1>  
Elemento de Bloque  
</h1>  
<h2>  
Elemento de Bloque  
</h2>
```

<h1> a <h6> (de bloque): un encabezado es, semánticamente hablando, el texto que encabeza o titula el contenido que sigue. Se puede tratar de un artículo, un texto o una sección del documento que estamos viendo.

Existen 6 niveles: del **<h1></h1>** al **<h6></h6>**

Esta jerarquía se debe respetar en cada documento HTML que forme parte del sitio web.

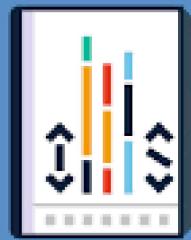
<h1> REINO ANIMAL </h1>
<h2> Vertebrados </h2>
<h3> Mamíferos </h3>
<h4> Bipedos </h4>
<h4> Cuadrúpedos </h4>
<h3> Aves </h3>
<h4> Voladuras </h4>
<h4> No voladoras </h4>
<h3> Reptiles </h3>
<h3> Peces </h3>
<h3> Anfibios </h3>
<h2> Invertebrados </h2>
<h3> Insectos </h3>
<h4> Voladores </h4>
<h4> No Voladores </h4>

ETIQUETAS

HTML:

<p>Etiquetas

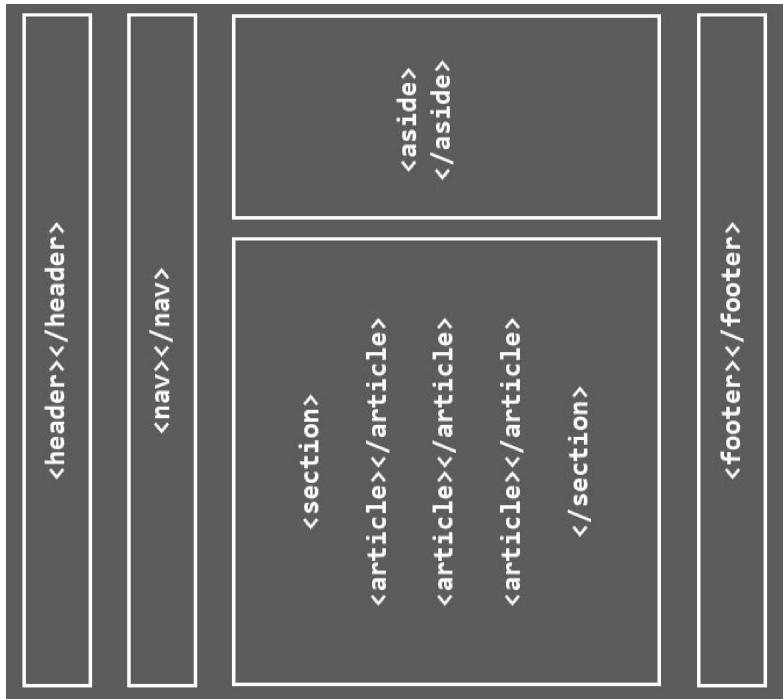
HTML</p>



- <p></p>
-
 (debloque)
- (delínea)
- (delínea)
- <div>(debloque)
- (delínea)

Utilizaremos todas estas etiquetas en nuestro primer ejemplo de estructura

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Mi primer sitio web</title>
  </head>
  <body>
    <h1>Encabezado nivel 1</h1>
    <h2>Mi nombre es:<h2> Tu Nombre y Apellido
    <h2>Objetivos del curso:<h2> Lorem ipsum
    <h3>Mini bio:<h3> Lorem ipsum
  </body>
</html>
```



HTML5 incorpora **etiquetas semánticas** que no sólo generan estructura, sino que también definen su contenido.

Etiquetas HTML5

<section>: Define una sección en un documento.

<nav>: Define un bloque que contiene enlaces de navegación, como por ejemplo el menú.

<article>: Define contenido autónomo que podría existir independientemente del resto del contenido.

<aside>: Define contenidos vagamente relacionados con el resto del contenido de la página.

<main>: Define el contenido principal o importante en el documento. Solamente existe un elemento **<main>** en el documento.

<header>: Define la cabecera de una página o sección.

<footer>: Define el pie de una página o sección.

¡A PRACTICAR!

Crea un documento nuevo llamado “index.html”. Escribe con etiquetas HTML:

- Título de la página
- Encabezado y subtítulo
- Incluí tu nombre y una mini bio de presentación.
- Incluí **etiquetas semánticas**

Guarda, abre el archivo en el browser de tu preferencia, y observa el resultado.

Multimedia en Html



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.

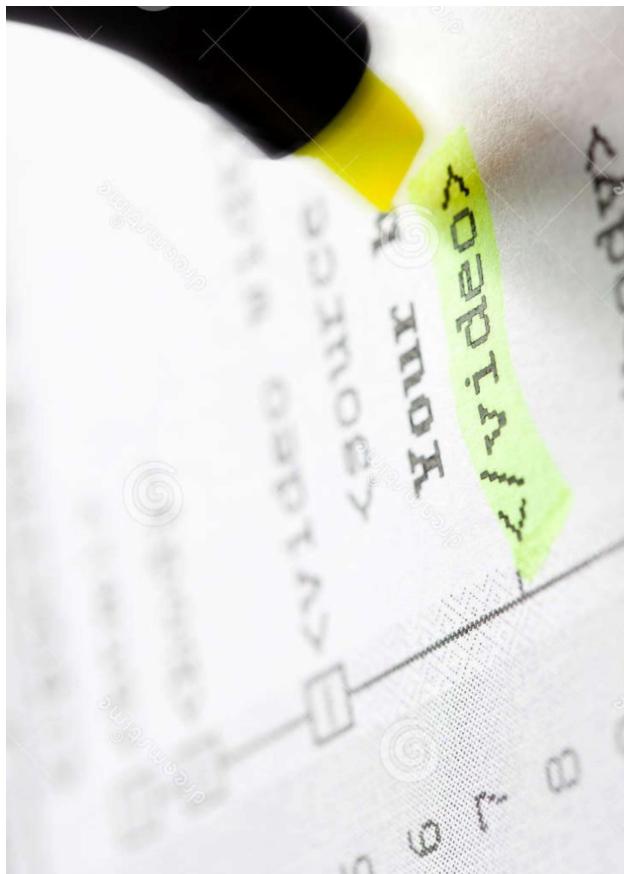
```

```

ETIQUETA VIDEO

```
<video src="mivideo.mp4"  
controls>
```

Tu navegador no implementa el
elemento <code><video></code>.
</video>



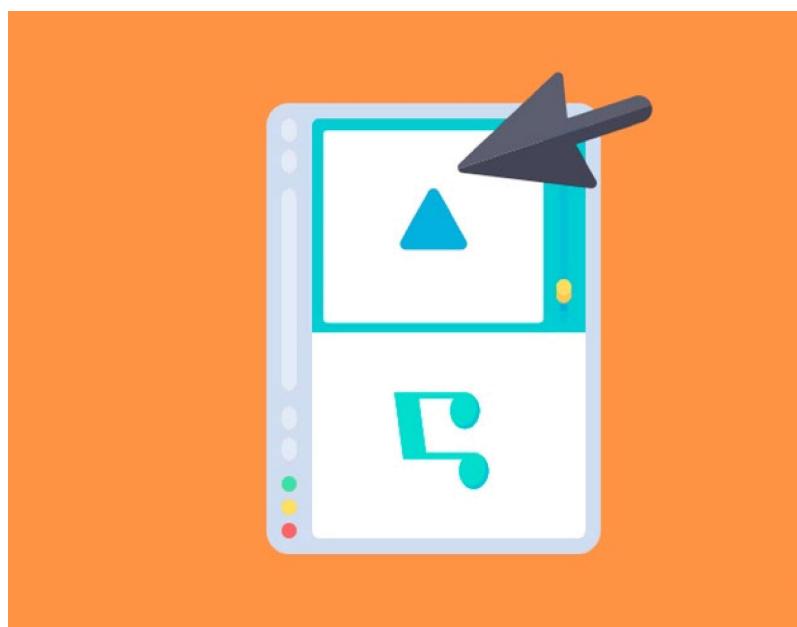
ETIQUETA AUDIO

```
<audio src="perfect.mp3" preload="auto"  
controls></audio>
```



ATRIBUTOS DE LA ETIQUETA <AUDI

- **Controls:** controles para manejar el audio.
- **Autoplay:** reproducción automática.
- **Loop:** repetición automática.
- **Preload:** almacenamiento temporal del audio.
 - "none" : no almacena el archivo.
 - "auto" : almacena temporal del archivo.
 - "metadata" : almacena temporalmente sólo los metadatos del archivo.



Se pueden especificar múltiples fuentes de archivos usando el elemento **<source>**, con el fin de proporcionar vídeo o audio codificados en formatos diferentes, para diversos navegadores.

```
<video controls>
  <source src="foo.ogv" type="video/ogg">
  <source src="foo.mp4" type="video/mp4">
Tu navegador no implementa el elemento
<code>video</code>.
</video>
```

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 IFRA ME

Insertar video

X

```
<iframe width="560" height="315"
src="https://www.youtube.com/embed
/5Ujb8B7JHv4" frameborder="0"
allow="accelerometer; autoplay;
encrypted-media; gyroscope;
picture-in-picture"
allowfullscreen></iframe>
```

Empezar en 0:01

COPiar

¡A PRACTICAR!



Crea un archivo HTML que contenga: una imagen y un iframe que muestre un video de youtube.

HTML: Listas y formularios



OBJETIVOS

- Comprender la sintaxis de HTML.
- Conocer las diferentes etiquetas y el uso de cada una.
- Dominar el concepto de web semántica, y las etiquetas HTML5 de estructura.



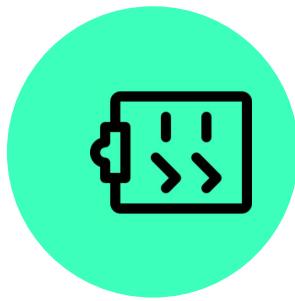
HTML: es un "lenguaje" de marcado de etiquetas, que permite crear documentos para web.

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.

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.
 - : define una lista ordenada de artículos (numéricas).
 - : define una lista de artículos sin orden (viñetas).
 - : define un artículo de una lista.

EJEMPLO

Ambas listas se deben insertar mediante la etiqueta `` (list-item).

Ejemplo de servicios de una empresa (lista de viñetas/sin orden):

- Empresa
 - Producto
 - Servicios
 - Contac to
- ```

Empresa
Producto
Servicios
Contac to

```

# *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.



# LISTAS

1. Crea dos listas usando las etiquetas de HTML: una con viñetas que contenga cinco nombres, y otra ordenada con 5 pasos para preparar un mate.

# FORMULARIOS



# FORMULARIOS

Contact Us

Name

Email

Message

Submit

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

# **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.

# 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.

# 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.

# 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.

- Input de tipo “*submit*”: envía el formulario.
- Input de tipo “*reset*”: resetea el formulario.
- Input de tipo “*button*”: no tiene acciones por defecto.

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

# **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.

# ATTRIBUTO “VALUE”

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

```
<form>
 <div>hombre</div>
 <input type="radio" name="genero" value="hombre"/>
 <div>mujer</div>
 <input type="radio" name="genero" value="mujer"/>
 <div>otro</div>
 <input type="radio" name="genero" value="otro"/>
</form>
```

Botones de radio:

```
<form>
 <div>Acepta términos y condiciones</div>
 <input type="checkbox" name="acepta" value="1"/>
</form>
```

Casillas de chequeo:

# 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 va a referenciar a “label” con su elemento del formulario.**

El valor del atributo “for” debe ser igual al valor del atributo “id” o “name” del elemento.

```
<form>
 <label for="nombre_apellido">Nombre:</label>
 <input type="text" name="nombre_apellido"/>
</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="tales">
<option value="L">Large</option>
<option value="M">Medium</option>
<option value="S">Small</option>
</select>
</form>
```

# CONJUNTO DE CAMPO

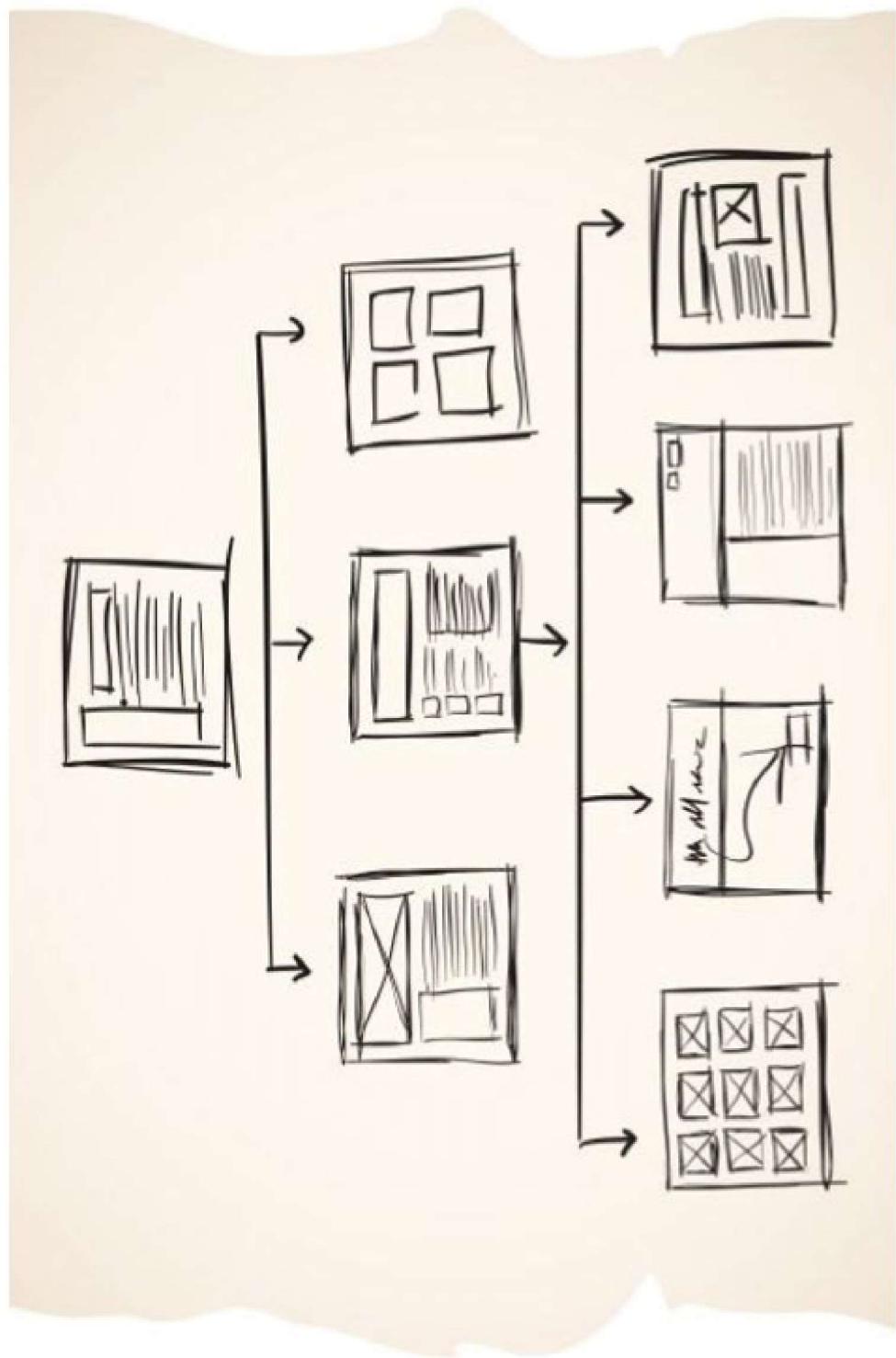
---

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

# 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:

```
Productos
```

# **ENLACES RELATIVOS, ABSOLUTOS E INTERNOS**



# *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.

```
Contacto
```

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

```
ver mapa
```

# ***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:

```
Desarrollo Web
```

# *ENLACES INTERNOS*

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

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



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

```
Formulario de conta cto
```

# ¡A PRACTICAR!

Crea un formulario de contacto como indica la imagen a continuación, usando los códigos vistos anteriormente. Tienes 15 minutos para realizar la actividad.

Información Personal:

Completa el siguiente formulario.

Nombre:

Apellido:

Talle:  ▾