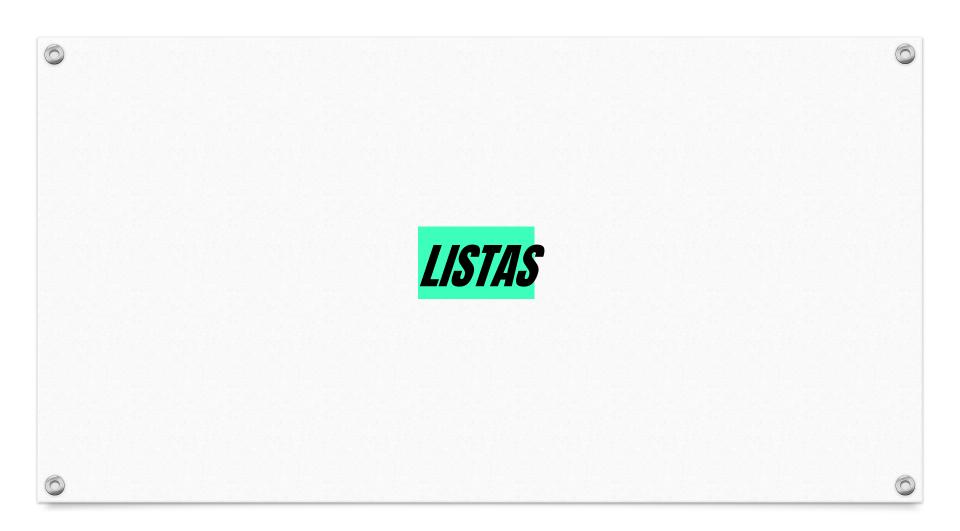




Dominar el concepto de web semántica y las etiquetas HTML5 de estructura







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 es denominado como listas.









## TIPOS DE LISTAS



Listas no ordenadas



Listas ordenadas



Listas de definición







## **Ejemplos de Listas**

- Paso a paso de un procedimiento, ejemplo: una receta de cocina.
- Características de una persona.
- Galería de imágenes.
- Menú de una página web.





## ¿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).
  - ul>: Define una lista de artículos sin orden (viñetas).
  - : Define un artículo de una lista.









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

Ejemplo de servicios de una empresa (Lista de viñetas/sin orden)

```
        Empresa
        Producto
        Servicios
        Contacto
```

- Empresa
- Producto
- Servicios
- Contacto







## ANIDAR LISTAS

Es probable que nos veamos en la necesidad de crear una estructura de sub-listas como la siguiente:

- Computadoras portátiles
  - Procesador I4
  - Procesador I5
- Computadoras de escritorio
  - Procesador Pentium
  - Procesador Celeron









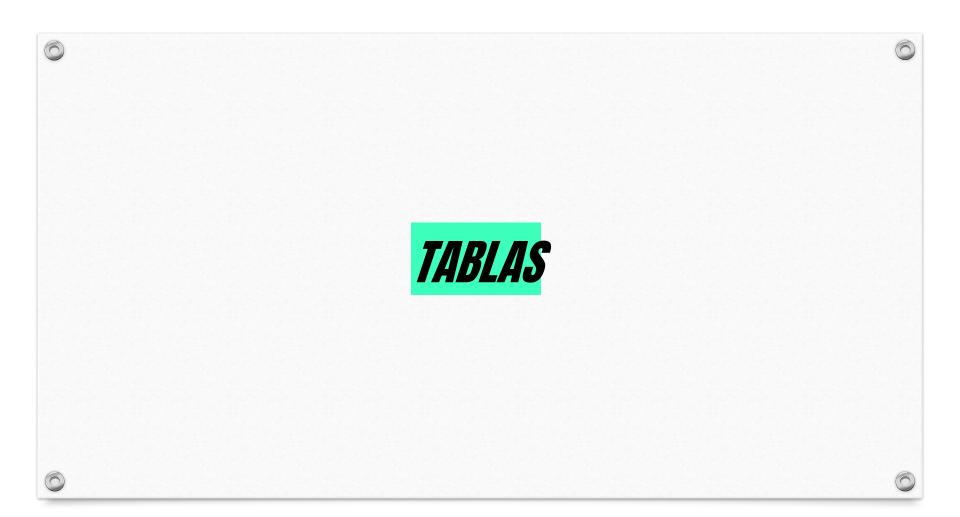
(Duración: 15 min)

Crear dos listas usando las etiquetas de HTML:

- Una lista con viñetas que contenga cinco nombres.
- Una lista ordenada con 5 pasos para preparar un mate.











Es un **conjunto de celdas organizadas** dentro de las cuales podemos 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











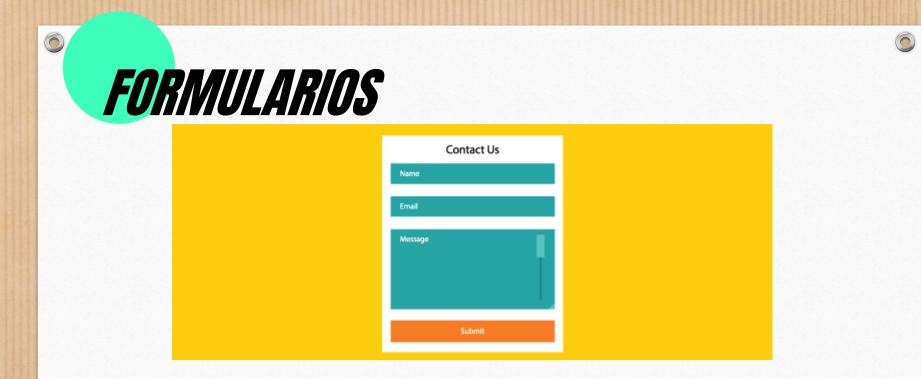
```
<!-- inicio de fila-->
 Fila 1 - Columna 1
 Fila 1 - Columna 2
 Fila 1 - Columna 3
<!-- cierre de fila -->
<!-- inicio de otra fila-->
 Fila 2 - Columna 1
 Fila 2 - Columna 2
 Fila 2 - Columna 3
<!-- cierre de la segunda fila -->
```

La etiqueta acepta 3 atributos de "diseño":

- Border: Border de la tabla
- Cellpadding: Especifica el espacio, en píxeles, entre la pared de la celda y su contenido.
- Cellspacing: Especificar la distancia entre las celdas y el margen exterior de la tabla







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 envio:
   GET y POST.
- Enctype: Cómo se codificarán los contenidos.







En este atributo indicaremos **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, por defecto el **Action** es el mismo archivo donde está el formulario.

Recordemos que: HTML no es un lenguaje de programación.







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").

Por defecto (si no se indica el Method) es GET.









Cuando el valor del atributo **method** es **post**, este atributo es el <u>tipo MIME</u> del contenido que es usado para enviar el formulario al servidor. Los posibles valores son:

- application/x-www-form-urlencoded: 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







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>







Los botones disparan las acciones del formulario.

Hay 3 tipos de botones:

- 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" y está 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.





## CONTROLES DE SELECCIÓN

En estos casos el usuario no puede ingresar libremente un texto, sino que el programador le da una lista predefinida. En todos los casos, el dato que nos 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 elegir una, todas o ninguna opción.
- Menú desplegable: Sólo se puede elegir una opción.





### Atributo "value"



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

### Botones de radio

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

### Casillas de chequeo







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

El principal atributo de la etiqueta label 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>
```







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>
```







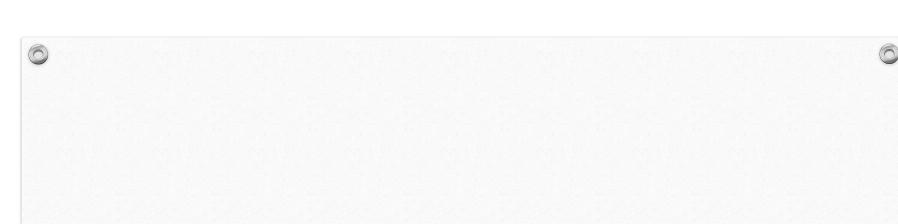
Las etiquetas **<fieldset>** y **<legend>** 

La etiqueta fieldset tiene como objetivo crear grupos de elementos del formulario que poseen un mismo propósito. La etiqueta legend, define formalmente el propósito del elemento fieldset. Se estructura de la siguiente manera:

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



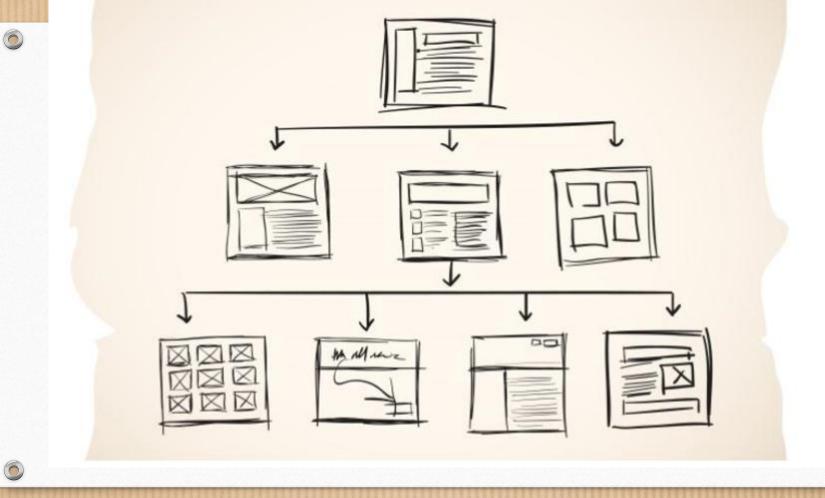


















## ARQUITECTURA DE UN SITIO

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

Para crear un enlace es necesario utilizar la etiqueta de ancla <a> con el atributo "href", que establecerá el destino al que apunta

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





# 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.

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

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

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









### **ENLACES ABSOLUTOS**

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

<a href="http://www.desarrollo.com/frontend">Curso de Frontend</a>









## **ENLACES INTERNOS**

Los enlaces internos permiten referenciar secciones de nuestra página, para hacer esto se utiliza el id:

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









También podemos 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 podemos agregar enlaces a texto, también podemos hacerlo con otros elementos. Por lo general se usan texto o imágenes. Ejemplo de enlaces con una imagen:

```
<a href="http://www.desarrollo.com/cursos.html#frontend">
    <img src="img/logo_desarrollo.png" alt="desarrollo"/>
    </a>
```



