

Introducción a HTML

Contenidos de la Clase

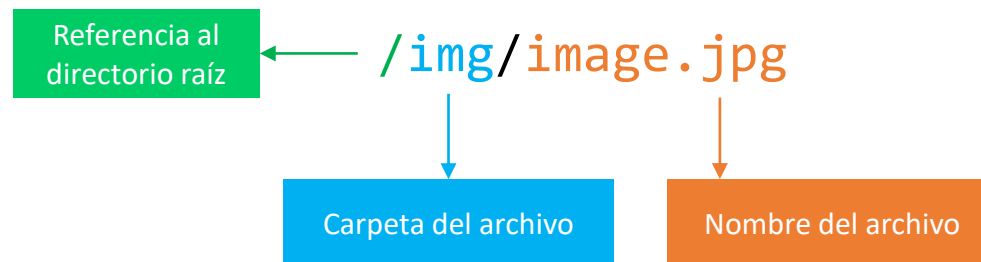
- Enlaces absolutos y relativos
- Multimedia
- Listas
- Formularios
- ¿Qué es CSS?
- Vincular CSS con HTML

iComencemos!

Enlaces

Los enlaces están formados por una serie de rutas, donde se le indica al navegador la dirección a la que tiene que ir cuando pulsamos sobre ese link. Tenemos dos tipos de ruta:

Ruta absoluta: incluye el nombre del dominio.



Ruta relativa: indica el nombre del directorio.

Enlaces

Los enlaces internos permiten referenciar secciones de una misma página. Utilizaremos la etiqueta `<a>`, junto con el atributo “`href`”.

Antes en el valor de “`href`” poníamos un enlace web o de otro archivo. Para indicarle al navegador que se trata de un lugar dentro de la misma página utilizaremos los marcadores, que inician con un `#`.

El nombre que tendrán estos marcadores dependerá del `id` del elemento o sección a la que haré referencia.

La sintaxis quedará así:

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

Listas en HTML



Listas de definición

Listas en las que tenemos “valor” y “definición”. Suelen utilizarse para definir términos.

```
<dl>
  <dt>Término1</dt>
  <dd>Definición 1</dd>
  <dt>Término 2</dt>
  <dd>Definición 2</dd>
  ...
  <dt>Término N</dt>
  <dd>Definición N</dd>
</dl>
```



Listas ordenadas

Cada uno de los elementos irá precedido de un número o letra que establezca su orden.

```
<ol>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ol>
```



Listas no ordenadas

Los elementos estarán precedidos por una viñeta.

```
<ul>
  <li>Elemento 1</li>
  <li>Elemento 2</li>
  ...
  <li>Elemento N</li>
</ul>
```

Formularios en HTML

Los formularios permiten a los usuarios la introducción de datos, que generalmente se envían a un servidor web o a un archivo para su procesamiento y almacenamiento.

<form></form>: elemento contenedor que define formalmente un formulario. Todos sus atributos son opcionales, pero es una práctica estándar establecer siempre al menos los siguientes atributos:

- **“action”**: define la ubicación (URL) donde se envían los datos que el formulario ha recopilado cuando se validan.
- **“method”**: define con qué método HTTP (o a dónde) se envían los datos. Puede ser:
 - **GET**: la información viajará por la barra de direcciones a continuación del nombre del archivo. Si no se indica en “method”, será GET por defecto.
 - **POST**: la información viajará junto a los encabezados del HTML (será “invisible”). No se obtienen los datos adjuntos en la dirección URL, sino que son incluidos en el cuerpo de la petición.

Ejemplo

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

  <label for="area">Texto largo: </label>
  <textarea name="area" id="area" cols="30" rows="10"></textarea>
  <br>

  <label for="telefono">Teléfono: </label>
  <input type="number" name="telefono" id="telefono"/>
  <br>

  <label for="email">Correo Electrónico: </label>
  <input type="email" name="email" id="email"/>
  <br>

  <label for="comentarios">Comentarios: </label>
  <input type="text" name="comentarios" id="comentarios"/>
  <br>

  <input type="submit"/>
  <input type="reset" value="Borrar Campos"/>
</form>
```

Nombre:

Texto largo:

Teléfono:

Correo Electrónico:

Comentarios:

Formularios - Atributos

Algunos controles generales para el ingreso de texto:

- `<input type="text">`: Cajas de texto de una sola línea (no acepta el uso de la tecla Enter).
- `<input type="password">`: Cajas para el ingreso de contraseñas (el contenido no será visible).
- `<textarea>`: Cajas para contenido multilínea. Puede ser una o muchas líneas de texto.

Más información sobre atributos en [<input>](#)

`<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 debe ser igual al valor de `id` o `name`.

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

Formularios - Botones

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

- “**submit**”: envía los datos al archivo indicado como ***Action***.
- “**reset**”: vacía todo lo ingresado y resetea los campos.
- “**button**”: lo utilizaremos con JavaScript más adelante.

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

Formularios - Controles

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”.

- Botones de radio: sólo se puede elegir una opción.

```
<div>Opción 1</div>
<input type="radio" name="pregunta 1" value="opciónuno"/>
<div>Opción 2</div>
<input type="radio" name="pregunta 1" value="opción dos"/>
```

Opción 1
☐
Opción 2
☐

- Casillas de chequeo: de toda la lista de opciones, el usuario puede optar por una, todas o ninguna opción.

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

Acepta términos y condiciones
☐

- Menú desplegable: sólo es posible seleccionar una opción.

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

Large
Medium
Small
Large ▼

Fieldset y Legend

`<fieldset>` y `<legend>` se utilizan en conjunto y sirven respectivamente para definir y etiquetar grupos lógicos de elementos de formularios.

Al formar varios grupos de elementos se puede crear una estructura más fácil de asimilar por el usuario, especialmente para formularios con muchos elementos.

`<fieldset>`: agrupa los elementos. Se utiliza con su respectiva etiqueta de cierre y lo que hace es crear un recuadro que rodea a los elementos de formulario colocados dentro de ella.

`<legend>` sirve para nombrar o etiquetar un grupo creado con FIELDSET. Añade una nota de aclaración sobre qué tipo de información se está agrupando en el recuadro. Puede ayudar al usuario y mejorar la interfaz y la claridad de los formularios.

Fieldset y Legend

Solo <fieldset> :

```
<form action="">
  <fieldset>
    <!--Elemento de formulario-->
    <input type="text" name="elemento1">
    <br>
    <!--Otro elemento-->
    <input type="text" name="otro">
  </fieldset>
</form>
```

<fieldset> con <legend> :

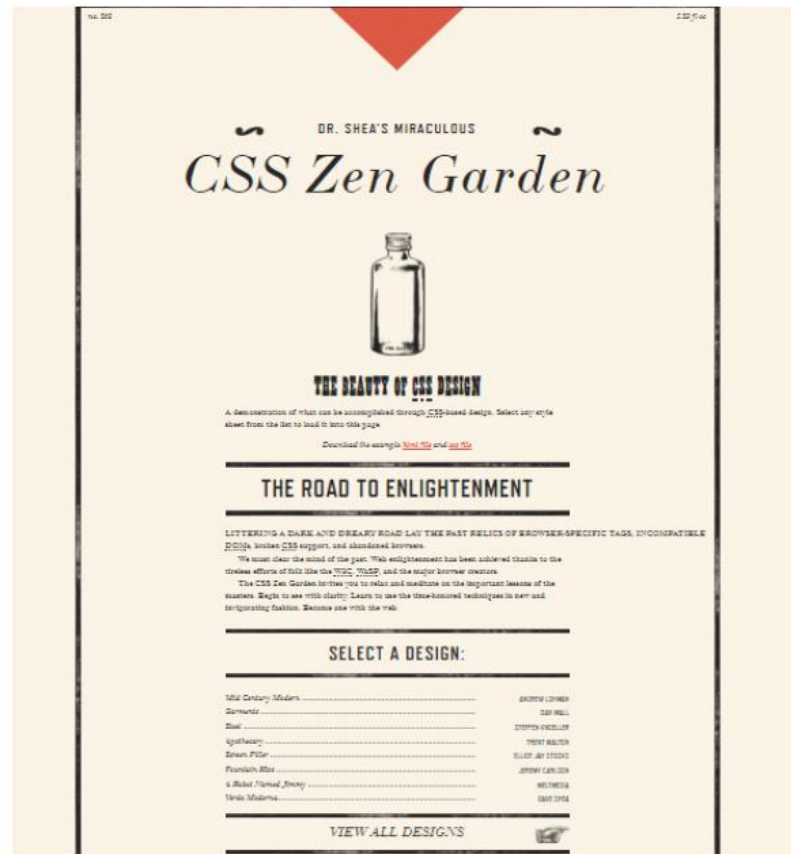
```
<form>
  <fieldset>
    <legend>Datos personales</legend>
    Nombre: <input type="text" name="nombre">
    <br>
    Edad: <input type="text" name="edad" size="2">
    <br>
    Dirección: <input type="text" name="direccion">
  </fieldset>
</form>
```

Practiquemos en Visual Studio Code...

Una breve presentación a lo que veremos la próxima clase

¿Qué es CSS?

CSS (Cascading Style Sheets) es un lenguaje web para aplicar formato visual al HTML. Con CSS se puede cambiar por completo el aspecto de cualquier etiqueta HTML, sin modificarla.



Vincular con HTML

- ❑ Forma EXTERNA: dentro de <head> llamamos al archivo CSS (recordar rutas relativas y absolutas).

```
<link rel="stylesheet" href="pruebas.css"/>
```

- ❑ Forma INTERNA 1: recomendable utilizarla dentro de <head> (se puede usar en <body> pero es más desprolijo).

```
<style>
  /*Código CSS*/
</style>
```

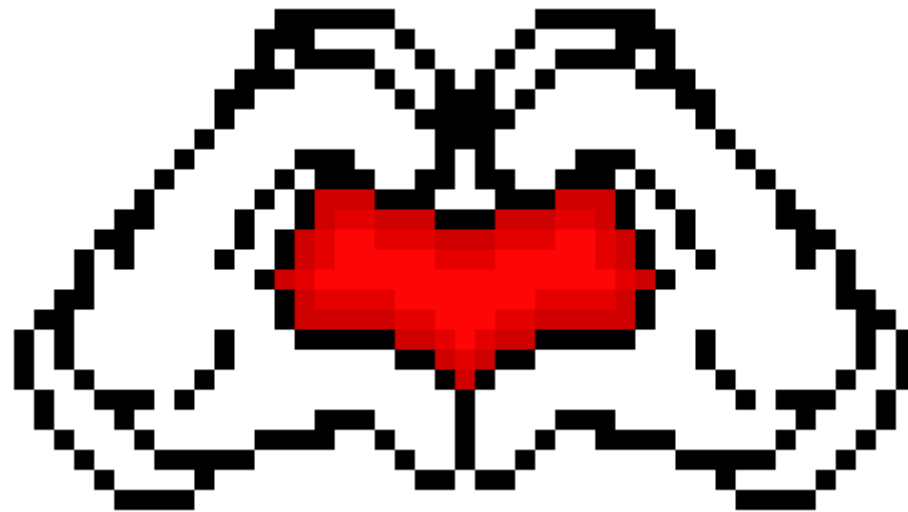
- ❑ Forma INTERNA 2: no es recomendable ya que es difícil de mantener. Se utiliza para parches o pruebas.

```
<section style="background-color: aqua;">
```

¿Preguntas?



¡Nos vemos la próxima clase!



BA MULTIPLICA 2.0

jóvenes  jóvenes



UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

