

BIBLIOGRAFÍA

García-Miguel, D. (2019). Diseño de Interfaces Web (1.a ed.). Madrid, España: Síntesis.

1. Lenguaje de marcas HTML

Los sitios web están compuestos por páginas que están escritas en algún lenguaje. En la actualidad, el sector profesional relacionado con el diseño web tiene muy buenas perspectivas y para ello es necesario comprender, conocer y manejar correctamente el lenguaje de marcas.

HTML es el Lenguaje de Marcado de Hipertexto utilizado en las páginas web. Este tipo de texto presenta una forma estructurada y agradable, con hyperlinks que conducen a otros documentos y con inserciones multimedia (sonido, imágenes, vídeos...). La base de este lenguaje es especificar en el texto la estructura lógica del contenido (título, párrafos de texto normal, enumeraciones, definiciones, citas...). Las características principales de este lenguaje son:

- Es muy sencillo
- No hay variables
- No se compila
- Es un lenguaje interpretado por el propio navegador
- A las instrucciones se les llama etiquetas
- Permite escribir hipertexto

1. Estructura

Un documento en HTML debe estar delimitado en sus extremos por las llamadas etiquetas, estas consisten en palabras reservadas que dividen el documento en secciones y les dan formato. Para crear un documento HTML, debemos delimitar todo el archivo con la etiqueta `<html>`, dentro de estas se distinguen dos partes claramente diferenciadas, la cabeza y el cuerpo, esto es, head y body. El siguiente diagrama muestra las etiquetas que se han de utilizar para definir un documento HTML y las secciones a las que hacen referencia en su representación web.

```
<html>...  
  <head>  
    Cabecera  
  </head>  
  <body>  
    Contenido  
  </body>  
</html>
```

La sintaxis de las etiquetas es:

`<identificador_de_la_etiqueta> </identificador_de_la_etiqueta>`

2. Texto, párrafos, estilos y formatos

El texto que se desea mostrar en las páginas web se escribe entre un conjunto de etiquetas que aportan estilo y formato. Una de las más conocidas son las etiquetas `<p>` y `</p>` las cuales definen un párrafo e introducen un salto y dejan una línea en blanco antes de continuar con el resto del documento.

El código HTML:

```
<p> <b>Hola, estamos escribiendo en un párrafo </b></p>
```

```
<p> Ahora ya hemos cambiado de párrafo </p>
```

se verá:

Hola, estamos escribiendo en un primer párrafo

Ahora ya hemos cambiado de párrafo

Existen multitud de etiquetas, tales como `` que ponen el texto en negrita, `<i>` el texto en cursiva. En la siguiente tabla se muestran alguna de las más utilizadas.

ETIQUETA	
<code>...</code>	Negrita
<code><i>...</i></code>	Cursiva
<code><u>...</u></code>	Subrayado
<code><sub>...</sub></code>	Pone el texto en subíndice
<code><h1>...</h1></code>	
<code><h2>...</h2></code>	
<code><h3>...</h3></code>	
<code><h4>...</h4></code>	Etiquetas de encabezado. Uno de los elementos principales para marcar el nivel de encabezado.

Cuadro 3.2 Tabla de etiquetas más actualizadas de formato en HTML

3. Enlaces

Para la creación de enlaces se utiliza la etiqueta `<a>`. Esta etiqueta tiene dos formas de uso, en primer lugar, indicar una url hacia el recurso donde se desea enlazar, para ello se utiliza el atributo “href” seguido de la url.

La sintaxis de los enlaces:

```
<a href="..."> ..... </a>
```

El código HTML:

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>PRUEBA ENLACES</title>
</head>

<body>
  <p><a href="https://www.sintesis.com">EDITORIAL SÍNTESIS</a></p>
</body>
</html>
```

se verá : (se podrá pinchar sobre el enlace y llevará el sitio web indicado)

[EDITORIAL SÍNTESIS](https://www.sintesis.com)

Finalmente, es interesante recordar otra de las principales características del lenguaje web, la creación de hipervínculos que permiten al usuario navegar por diferentes páginas, de esta forma puede dirigir el sentido de su navegación hacia el objetivo deseado.

La creación de enlaces se realiza a través de la etiqueta ****, indicando la ruta, relativa o absoluta, hacia la que apunta el enlace.

- **Ruta relativa.** Se indica solo el nombre del fichero hacia el que apunta el enlace. En este caso implica que el fichero destino se encuentra en la misma carpeta que el fichero origen desde el que se está invocando.

```
<a href="ficheroDestino.html">...</a>
```

- **Ruta absoluta.** En este caso se indica la ruta completa del fichero o sitio web destino.

```
<a href="https://www.google.es">...</a>
```

4. Imágenes

Cuando se diseña un sitio web, incorporar imágenes le aporta más atractivo para el usuario, que si por el contrario solo incluimos texto, ahora bien, es aconsejable que la calidad de las imágenes sea óptima.

Además, cuando se utilizan imágenes estas debemos cumplir los derechos de publicación de las mismas, en el caso de no ser realizadas expresamente para el sitio web donde van a ser usadas.

La sintaxis de la inserción de imágenes:

(sin atributos)

```

```

(con atributos)

```

```

La inclusión de imágenes se trata de otro de los elementos esenciales en el diseño de interfaces web. La etiqueta utilizada es ****.

En su forma más sencilla simplemente hay que indicar la ruta en la que se encuentra la imagen a insertar, pero también es posible incluir otros atributos que definen la apariencia y posición de la imagen.

```

```

La función para la inserción de imágenes permite la gestión de diversos atributos asociados a la imagen insertada, desde al tamaño de la misma, hasta el título.

5. Formularios. El elemento input

Los formularios son utilizados para recoger información en un sitio web, esta información es enviada al servidor donde es procesada. Cada formulario contiene uno o varios tipos de controles, estos permiten recolectar la información de varias formas diferentes, desde el ingreso de una palabra en un cuadro de texto, hasta la subida de un archivo.

La sintaxis de la estructura de un formulario es:

`<form>`

....

`</form>`

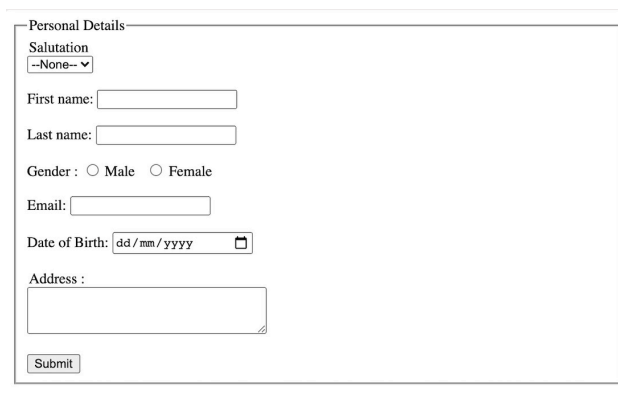
Los formularios son elementos que permiten introducir diferente tipo de información para ser tratada por la aplicación.

Habitualmente se modela en entornos cliente-servidor, los cuáles ante las entradas completadas por el usuario realiza una determinada acción y a continuación, muestra el resultado de esta a través de la interfaz desarrollada.

Para la creación de formularios la etiqueta principal utilizada es `<form>` así como un conjunto de opciones asociadas.

`<form action="" method="get"> ...</form>`

Uno de las características más importantes en los formularios es la definida mediante la etiqueta **input**, la cual a través del atributo **type**, permite definir el tipo de información sobre la que se va a recoger los datos completados por el usuario.



The image shows a web form titled "Personal Details". It contains several input fields: a dropdown menu for "Salutation" with "--None--" selected, text boxes for "First name:" and "Last name:", radio buttons for "Gender : Male" and "Female", a text box for "Email:", a date picker for "Date of Birth:" with the format "dd/mm/yyyy", and a large text area for "Address :". At the bottom is a "Submit" button.

En los siguientes apartados se recogen algunos de los valores más utilizados para la etiqueta input, así como un ejemplo de su sintaxis.

Uno de los controles más importante y utilizado es el elemento `<input>`, se utiliza para crear controles interactivos en formularios. La forma en que `<input>` funciona viene determinado por el valor del atributo **type**. El valor predeterminado del control **input** es **text**, este permite el ingreso de una única línea de texto. En la siguiente tabla se recogen los valores más utilizados para el atributo **type**.

5.1. TEXT

1. **text.** Se trata de un cuadro de texto vacío donde se puede escribir cualquier texto. Modela una caja de texto donde el usuario puede escribir cualquier texto, el cual podrá ser validado a través de la programación oportuna. Algunas de las opciones más comunes de este tipo de input son:

- **name:** el nombre identificador de la caja de texto
- **value:** contenido de la caja de texto por defecto, si no se indica nada aparece vacía.

```
<input name ="" value="" type ="text" >...</input>
```

El código HTML:

```
<html>
<head>
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    <br> Nombre <br/>
    <input type="text" name="nombre" value="" />
  </form>
</body>
</html>
```

Nombre




5.2. BUTTON

2. **button**. Se debe diferenciar entre la acción de enviar que realizan los botones creados mediante el tipo “submit”, y otros botones que se deseen crear para el resto de operaciones, estas deben ser implementadas mediante lenguaje Javascript, se pueden programar para que realicen cualquier tarea, por muy compleja que sea.

El código HTML:

```
<html>
<head>
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    <input type="button" name="guardar" value="Guardar" />
  </form>
</body>
</html>
```

se verá :



Este valor se utiliza para la creación de un botón, normalmente es empleado en el diseño de botones sencillos.

Existe otro modelado de botones con el tipo **submit**, utilizado principalmente para el envío de valores a través de un formulario. En HTML5 este tipo incluido en la etiqueta **input**, ha sido sustituido por la instrucción directa **<button>**.

- **name**: el nombre identificador del botón
- **value**: texto que aparece en el botón diseñado

Sintaxis HTML:

```
<input name = "botonAceptar" value="Aceptar" type ="button" >...</input>
```

Sintaxis HTML5:

```
<button type ="button" value="Aceptar" >
... </button>
```

5.3. CHECKBOX

3. **checkbox**. Casilla de selección, se usa junto al atributo value para definir el valor que se enviará. El atributo checked para indicar si el elemento está seleccionado.

El código HTML:

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    <br> Elige una o varias opciones <br>
    <br> <input name="op1" type="checkbox" value="op1"/> Opción 1 <br>
    <br> <input name="op2" type="checkbox" value="op2"/> Opción 2 <br>
    <br> <input name="op3" type="checkbox" value="op3"/> Opción 3 <br>
  </form>
</body>
</html>
```

Elige una o varias opciones

☐ Opción 1

☐ Opción 2

☐ Opción 3

Este elemento permite la inclusión de un conjunto de casillas de selección. Permite incluir tantas opciones para ser seleccionadas como el diseño requiera. Por ello basta con indicar un nombre identificador de cada opción, de esta forma, el valor o valores seleccionados podrán ser evaluados.

- **name**: el nombre identificador de la opción de selección
- **checked**: si aparece esta opción a true, por defecto aparece marcada la opción donde se indica este valor. Si no aparece se supone a false, es decir, las opciones se mostrarán sin marcar. Para restringir que solo un número determinado de opciones se puedan marcar, se utilizarán funciones en Javascript.


```
<br><input name="op1" type="checkbox" value="op1" checked=true/> Opción 1 </br>
<br><input name="op2" type="checkbox" value="op2" checked=true/> Opción 2 </br>
<br><input name="op3" type="checkbox" value="op3" /> Opción 3 </br>
```

Código 3. Etiquetas básicas para la creación de un fichero HTML

☒ Opción 1

☒ Opción 2

☐ Opción 3

5.4. RADIO

4. **radio**. Este tipo de controles son similares a los checkbox, pero estos son mutuamente excluyentes. Se utilizan cuando solo puede escogerse una de las opciones. Cada vez que se selecciona una opción, se deselecta cualquier otra que estuviera seleccionada.

1. El parámetro **name** se utiliza para indicar que dos, o más radiobutton están relacionados y por tanto, no deben estar seleccionados al mismo tiempo.

El código HTML:

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    <br> Elige una opción <br/>
    <br> <input type="radio" name="Tipo" value="tipo1" /> Tipo 1 </br>
    <br> <input type="radio" name="Tipo" value="tipo2" /> Tipo 2 </br>
  </form>
</body>
</html>
```

Elige una opción

☒ Tipo 1

☐ Tipo 2

2. Si el parámetro **name** no tiene el mismo nombre se podrán seleccionar varias opciones.

El código HTML:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>...</title>
</head>
<body>
  <form action="" method="get">
    <br> Elige una opción <br/>
    <br> <input type="radio" name="Tipo" value="tipo1" /> Tipo 1 </br>
    <br> <input type="radio" name="Tipo" value="tipo2" /> Tipo 2 </br>
  </form>
</body>
</html>
```

Elige una opción

☐ Tipo 1

☐ Tipo 2

3. Atributo **checked**: para indicar si se debe marcar algunos de los elementos por defecto.

El código HTML:

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>...</title>
</head>
<body>
  <form action="" method="get">
    <br> Elige una opción <br/>
    <br> <input type="radio" name="Tipo" value="tipo1" checked/> Tipo 1 </br>
    <br> <input type="radio" name="Tipo" value="tipo2" /> Tipo 2 </br>
  </form>
</body>
</html>
```

4. Atributo **required**: para hacer obligatoria completar ese campo y enviar el formulario

5.5. FILE

5. **file**. Control que permite al usuario seleccionar un archivo. Se puede usar el atributo `accept` para definir los tipos de archivo que el control podrá seleccionar. Al pulsar sobre el botón, se despliega una ventana que permite navegar por el sistema y seleccionar el fichero deseado,

El código HTML:

```
<html>
<head>
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    Seleccione un fichero
    <br> <input type="file" name="adjunto" /> </br>
  </form>
</body>
</html>
```

5.6. SUBMIT

6. **submit**. Botón que envía el formulario con los datos introducidos por el usuario al servidor, se puede establecer el texto que va a aparecer en el botón, si no se hace, se muestra el valor predeterminado “Enviar consulta”.

El código HTML:

```
<html>
<head>
  <title>...</title>
</head>
<body>
  <form action="" method="get">
    <input type="submit" name="buscar" value="Buscar" />
  </form>
</body>
</html>
```

5.7. SELECT

7. **select**. Se utiliza para definir listas desplegables, dentro de esta etiqueta se puede definir el número de elementos que se deseen mediante la etiqueta `<option>`.

El código HTML:

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
  <title>...</title>
</head>
<body>
  <label for="so">Lista desplegable</label> <br/>
  <select id="lista" name="lista">
    <option value="" selected="selected">- selecciona una opción -</option>
    <option value="opcion1">Opción 1</option>
    <option value="opcion2">Opción 2</option>
    <option value="opcion3">Opción 3</option>
    <option value="opcion4">Opción 4</option>
  </select>
</body>
</html>
```

Para incorporar listas desplegables con varios valores se utiliza el elemento **select**. En este caso hay que tener en cuenta la creación del elemento select, así como de tantas opciones bajo la etiqueta **option** como sean necesarias.

- **name**: el nombre identificador de la lista desplegable
- **selected**: si no aparece esta opción, la que aparecerá seleccionada en el desplegable por defecto será la primera opción etiquetada. Si se indica en una de las opciones, esa será la mostrada y seleccionada por defecto. En el siguiente ejemplo, aparecería la última opción seleccionada antes que el usuario escogiera una nueva.

```
<select name="asignaturas">
  <option>Diseño de Interfaces Web</option>
  <option>Programación</option>
  <option>Bases de datos</option>
  <option>Lenguaje de marcas</option>
  <option selected="selected">-</option>
</select>
```

Código 2. Ejemplo de lista desplegable con SELECT

ACTIVIDAD PROPUESTA 2.4. Crea un formulario una página HTML que contenga al menos cinco tipos de elementos posibles de formulario.

ACTIVIDAD PROPUESTA 2.5. Realiza una página web en la que se incorporen cada uno de los puntos del apartado 2.

6. CAPAS

Las capas son uno de los elementos más utilizados en el diseño de páginas web, se trata de bloques de contenido que se sitúan dentro de una misma página web que permiten definir su comportamiento de forma dinámica. Este tipo de elementos definen su formato a través del lenguaje de hojas de estilo CSS.

La creación de las capas se realiza a través de la etiqueta **<div>**, como se puede observar en el ejemplo, la creación de cada capa implica la creación de bloques diferenciados, por esa razón los elementos contenidos en cada bloque aparecen en líneas diferentes, por el contrario las dos palabras que aparecen en la misma capa, aún estando en líneas diferentes, en el resultado final se muestran en una única línea.

```
<html>
  <head><meta charset=utf-8 /></head>
  <body>
    <div>
      Hola
      Hola
    </div>
    <div>
      Adiós
    </div>
  </body>
</html>
```

Código 6. Creación de capas en HTML

3.5. HTML5

HTML5, consiste en la última versión del lenguaje para la programación de páginas web, HTML. Los sitios implementados con este lenguaje solo pueden visualizarse correctamente en los navegadores más actuales, puesto que versiones anteriores de estos no son capaces de interpretar algunas de las nuevas etiquetas que se incluyen y que veremos en este apartado.

Una de las características principales de HTML5 son los elementos semánticos, también llamado **marcado semántico**, estos generan divisiones dentro del documento, pero a diferencia de la habitual etiqueta `<div>`, la cual también define una división, sí describen al navegador claramente su propósito. Los nuevos elementos semánticos en HTML5 se describen a continuación:

- **<article>**. La etiqueta `article` se utiliza para describir las unidades de contenido. Se pueden definir tantos elementos de este tipo como información vaya a contener la página web. Cuando dos o más de estos elementos están anidados, se supone que el elemento `article` interior guarda relación con el elemento `article` que actúa como padre. Este podría ser el caso de una entrada de blog (siendo en sí misma un `article`), donde cada comentario puede estar encerrado por un elemento `article`.

```
1. <article>
2.   <header>
3.     <h1>La cocina china</h1>
4.     <p>Publicado <time pubdate datetime="2014-03-28T20:00-04:00">2 meses atrás</time></p>
5.   </header>
6.   <p>La cocina china incluye estilos originarios de diversas regiones de China, así como de personas chinas en
   otras partes del mundo. La historia de la cocina en China se remonta a cientos de años atrás y ha cambiado de
   período a período y en cada región de acuerdo al clima, las modas imperiales, y la preferencia local.</p>
7.   <p>...</p>
8.   <footer>
9.     <p>&copy; Todos los derechos reservados</p>
10.  </footer>
11. </article>
```

- **<footer>**. Equivale al pie de página, se utiliza para situar los créditos del sitio web, los datos de contacto, el copyright, formularios de contacto, etc. Puede utilizarse de forma general para todo el sitio web, pero también para definir el pie de un elemento tipo `<article>`.
- **<header>**. Cabecera del sitio, define la parte superior de la página web, suele mostrar algún elemento representativo que caracterice a la marca del sitio web. En la mayor parte de los casos se sitúa el menú de navegación `<nav>`.
- **<main>**. Representa el contenido principal del `<body>` de un documento. El área principal consiste en el contenido que está directamente relacionado con el tema central de un documento. Este contenido debe ser único al documento. Es importar tener en cuenta que solo debe haber un elemento `<main>` en un documento, y no debe ser descendiente de un elemento `<article>`, `<aside>`, `<footer>`, `<header>`, o `<nav>`.

```
<main>
  <nav>...</nav>
  <p> ... </p>
</main>
```

- **<nav>**. La etiqueta `<nav>` se crea un menú de navegación, que enlaza con las diferentes partes del sitio web. Normalmente se sitúa justo debajo del encabezado, o a la izquierda de la página.
- **<section>**. Se podría decir que esta etiqueta es la sustituta de `<div>`, creando diferentes secciones dentro de una misma página web. Estas secciones pueden contener a su vez varios elementos de tipo `<article>`.
- **<time>**. Representa la hora y la fecha en una página web.
- **<email>**. En cuanto a los formularios, HTML5 incorpora nuevos elementos, tales como `<email>`, este crea un campo para introducir el email y realiza una validación automática, de esta forma no será necesario implementar código que realice esta función en JavaScript.

Otros para formularios:

- **<url>**. El elemento `input`, teniendo el valor "url" en su atributo `type`, representa un campo para un solo [URL absoluto](#). El control asociado a este campo es una caja de texto que permite a los usuarios editar una sola línea de texto regular.

```
<p>
  Sitio web: <input type="url" name="sitioweb" placeholder="¿Tienes un sitio web?">
  <input type="submit" value="Enviar datos">
</p>
```

- **<date>**. El elemento `input`, teniendo el valor "date" en su atributo `type`, representa un campo para la entrada de una fecha. .

```
2. <p>Fecha esperada: <input type="date" name="fechaesperada"> <input type="submit" value="Enviar datos"></p>
3. </form>
```

Fecha esperada:

- **<month>**. El elemento `input`, teniendo el valor "month" en su atributo `type`, representa un campo para la entrada de un mes.

```
2. <p>Mes de nacimiento: <input type="month" name="mesnacimiento"> <input type="submit" value="Enviar datos"></p>
3. </form>
```

Mes de nacimiento:

- **<number>**. El elemento `input`, teniendo el valor "number" en su atributo `type`, representa un campo para la entrada de un número. Los atributos `min` y `max` pueden ser usados en este elemento para establecer un rango de números válidos que el usuario tendrá permitido enviar.

- `min`
- `max`
- `Step`

```

2. <p>Tu edad: <input type="number" name="edad"> <input type="submit" value="Enviar datos"></p>
3. </form>

```

Tu edad:

```

2. <p>
3.   Limones usados para el jugo: <input type="number" name="limonesusados" min="1" max="5" step="0.5">
4.   <input type="submit" value="Enviar datos">
5. </p>
6. </form>

```

Limones usados para el jugo:

- **<color>**. El elemento `input`, teniendo el valor "color" en su atributo `type`, representa un campo para la entrada de un color.

```

2. <p>Color principal del perfil: <input type="color" name="colorprincipal"> <input type="submit" value="Enviar
   datos"></p>
3. </form>

```

Color principal del perfil:

- **<mark>**. Esta etiqueta se utiliza para representar un texto resaltado, es decir, subrayado, de esta forma queda marcada su relevancia en el contexto.

```

<main>
  <h2>Bienvenido a <mark>HTML5</mark></h2>
</main>

```

- **<figure>**. El elemento `<figure>` representa contenido de carácter visual, como: imágenes, ilustraciones, diagrama, fragmentos de código, etc. Este elemento puede ser movido a otra página o a un apéndice sin que afecte al flujo principal.
- **<figcaption>**. Añade título a una figura. Se ubica anidado del elemento de imagen, `<figure>` al que va asociado.

En HTML:

```
<html>
<head>
  <title>Título de la WEB</title>
  <meta charset="UTF-8">
</head>
<body>
  <header>
    <h1><b>Imagen en HTML5</b></h1>
  </header>
  <section>
    <article>
      <h2>Bloque de contenido 1. Article</h2>
      <figure>
        
        <figcaption> Ejemplo de leyenda</figcaption>
      </figure>
    </article>
  </section>
</body>
</html>
```

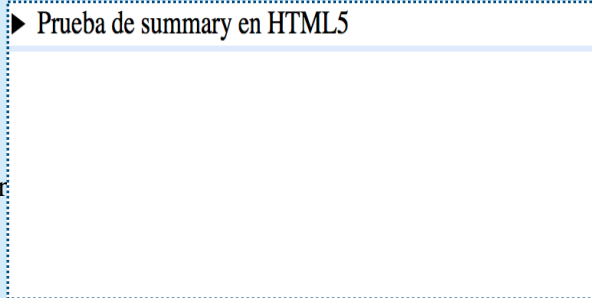
- **<details>**. Esta etiqueta tiene como finalidad mostrar información adicional a algún elemento de la página web. El funcionamiento de esta etiqueta consiste en un widget que oculta y muestra información, similar a un menú efecto acordeón.
- **<summary>**. Esta nueva etiqueta incorpora un sumario a la etiqueta details, es decir, añade información complementaria.

En HTML:

```
<html>
<head>
  <title>Título de la WEB</title>
  <meta charset="UTF-8">
</head>
<body>
  <details open>
    <summary>Prueba de summary en HTML5</summary>
    <p>TEXTO QUE SE MUESTRA AL INCORPORAR UN
ELEMENTO DETAILS EN HTML5 Y DESPLEGARLO</p>
  </details>
</body>
</html>
```

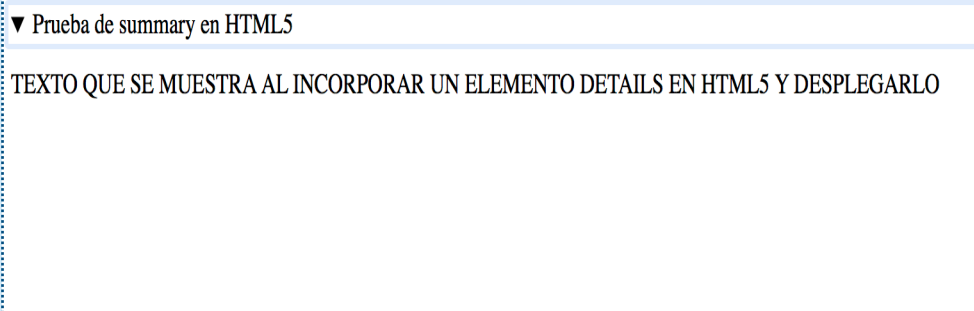
Se verá:

- Antes de desplegar el menú:



► Prueba de summary en HTML5

- Después de desplegar



▼ Prueba de summary en HTML5

TEXTO QUE SE MUESTRA AL INCORPORAR UN ELEMENTO DETAILS EN HTML5 Y DESPLEGARLO