

HTML5



HTML | Intro

Después de esta lección podrás:

1. Entender el lenguaje de marcado.
2. Realizar tus propias páginas estáticas con HTML.

¿Qué es el HTML?

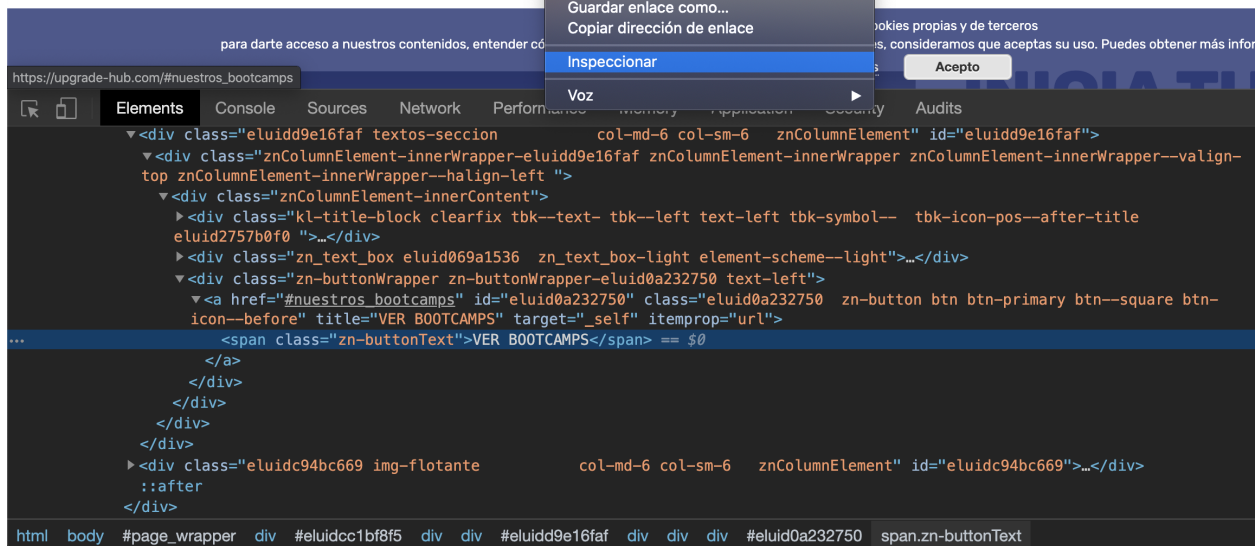
El HTML está presente en absolutamente TODAS las páginas web, es el lenguaje que se usa para indicar al navegador los elementos que tendrá nuestra página web. Si te pica la curiosidad, es muy fácil ver el HTML de una página haciendo click con el botón derecho de tu ratón sobre algún elemento de una web y después en inspeccionar.

REVOLUCIÓN DIGITAL

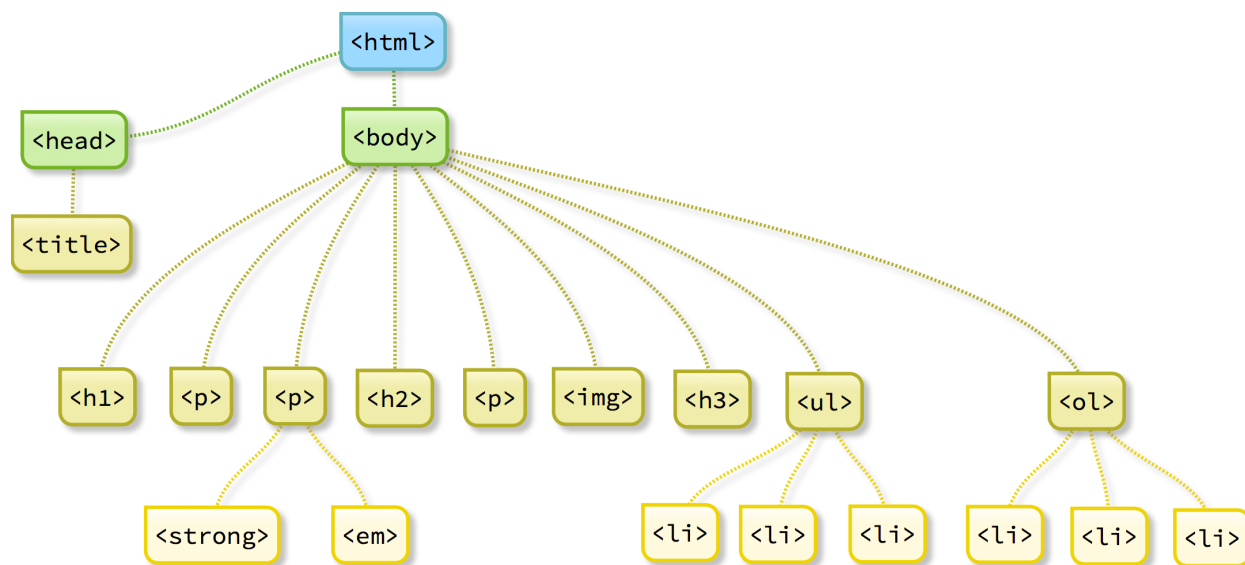
Cursos intensivos en Desarrollo Web y Diseño UX/UI y que las empresas te encuentren a ti

VER BOOTCAMPS

Abrir enlace en una pestaña nueva
Abrir enlace en una ventana nueva
Abrir enlace en una ventana de incógnito
Guardar enlace como...
Copiar dirección de enlace
Inspeccionar
Voz



El HTML es un documento de texto que tiene estructura de árbol. Es decir que hay un elemento raíz, y "colgando" de este elemento raíz, puede haber múltiples elementos anidados unos dentro de otros.



Cada uno de los elementos HTML tiene una etiqueta de apertura y una etiqueta de cierre. Ambas entre símbolos de mayor y menor, conteniendo la etiqueta de cierre además una barra, veamos un ejemplo:

`<p>`Esta es una etiqueta de tipo p (párrafo) en html**`</p>`**

Todas las etiquetas deben tener su apertura y su cierre, salvo en algunos casos especiales como por ejemplo las imágenes o los saltos de línea.

Estructura básica de un HTML

Ya hemos comentado que un HTML tiene estructura de árbol, vamos a ver los elementos mínimos de ese árbol:

- la raíz tiene que ser siempre la etiqueta html
- dentro de la raíz encontraremos head y body
- head contiene meta información de la página como por ejemplo el título de la página
- body contiene los elementos principales y visibles del documento HTML

Veamos un ejemplo:

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mi demo de HTML</title>
</head>
<body>
  <p>Soy un párrafo</p>
  <p>Aquí abajo viene una lista de frutas:</p>
  <ul>
    <li>Manzana</li>
    <li>Pera</li>
    <li>Plátano</li>
  </ul>
</body>
</html>

```

Como puedes observar dentro del `<body>` encontramos una serie de elementos (dos párrafos `<p>` y una lista ``) y de hecho podemos encontrar a su vez más nodos de HTML dentro de ellos, por ejemplo dentro de la lista `` encontramos los elementos ``

Elementos más comunes en HTML

Veamos una lista de los elementos más usados en HTML, todos ellos estarán explicados con un comentario HTML a su lado. Los comentarios son etiquetas que el usuario no verá, pero que nosotros como programadores podemos escribir para ayudarnos a entender el código.

Los comentarios en HTML se escriben empezando con `<!--` y terminando con `-->`.

¡Vamos al lío!

```

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mi demo de HTML</title>
</head>
<body>

  <!-- Aquí viene un título (Heading 1) -->
  <h1>¡Hola chicos, bienvenidos!</h1>

  <!-- La etiqueta div (divider) nos sirve para generar un contenedor -->
  <div>

```

```

<!-- La etiqueta p nos sirve para generar párrafos de texto -->
<p>Hola soy un párrafo</p>
<p>Soy otro párrafo, y ahora va a venir una lista.</p>

<!-- ul proviene de Unordered List (lista desordenada o sin numeración) -->
<ul>
  <!-- Cada li representa un elemento en la lista (List Item) -->
  <li>Primer elemento</li>
  <li>Segundo elemento</li>
</ul>
</div>

<!-- Veamos ahora una imagen -->


</body>
</html>

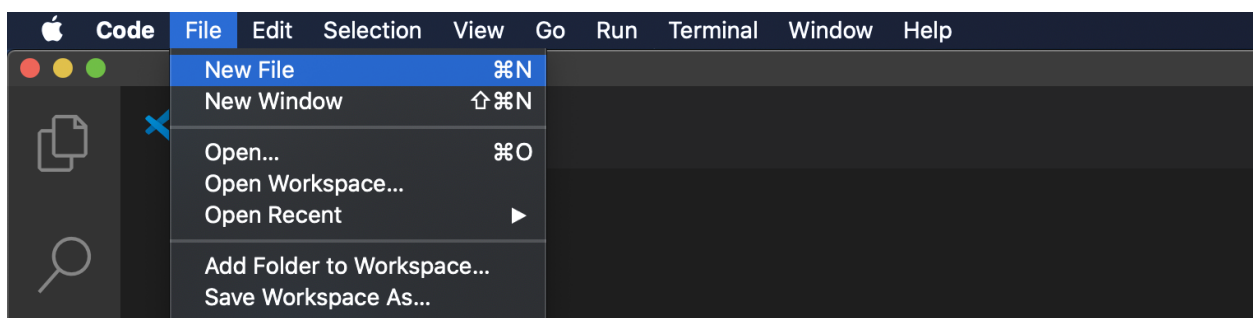
```

¿Cómo trabajamos con HTML?

Está muy bien tener algunas nociones teóricas sobre HTML... pero ahora bien ¿Eso cómo se aplica?

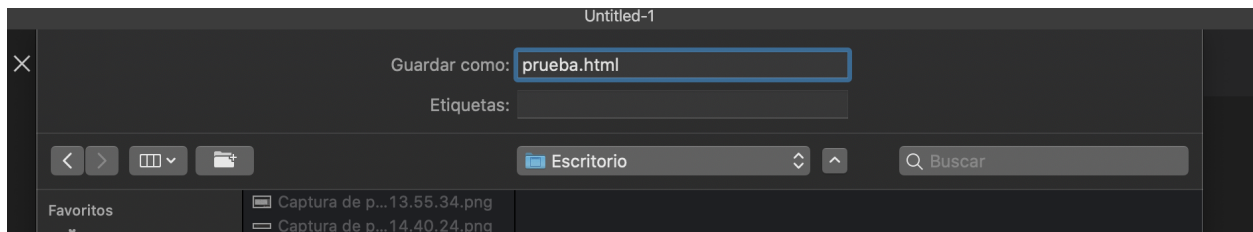
La manera más sencilla pasa por crear un documento HTML en tu ordenador y empezar a escribir directamente ahí tu HTML. Para editar el HTML podrías usar un simple editor de texto como por ejemplo el bloc de notas, pero te aconsejamos que uses algo más potente, un IDE de desarrollo te ayudará coloreando las etiquetas y dándote consejos o avisándote si algo está mal.

Nosotros usamos Visual Code Studio. Una vez lo hayas instalado debes crear un nuevo fichero:

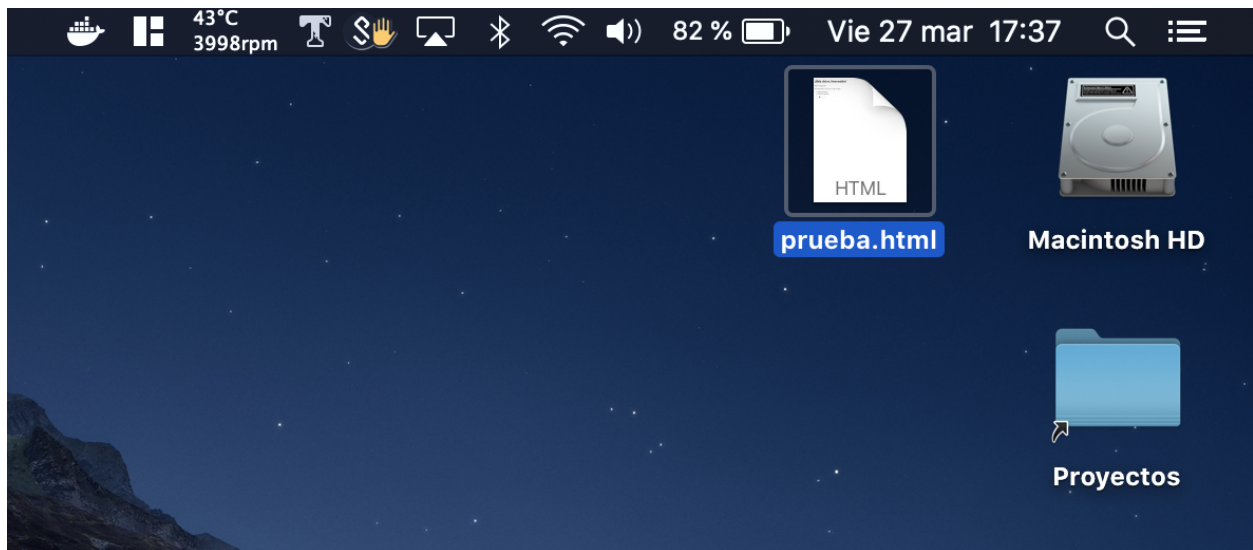


A partir de ahora puedes empezar a programar, te aconsejamos que empieces copiando el ejemplo HTML que te hemos dejado un poco más arriba con etiquetas. Después guárdalo,

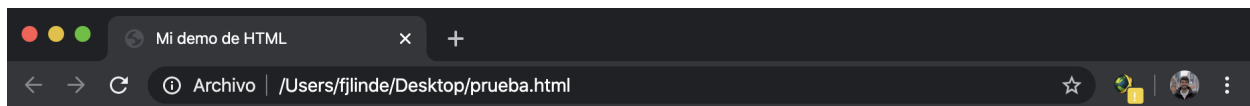
puedes llamarlo **prueba.html**, es importante que el nombre acabe en .html o sino luego tu navegador y tu editor no lo reconocerán como html:



Una vez lo hayas creado podrás ver el documento en tu escritorio:



Si haces doble click en el fichero, se abrirá tu navegador (te aconsejamos que uses Google Chrome) y verás tu primera página web:



¡Hola chicos, bienvenidos!

Hola soy un párrafo

Soy otro párrafo, y ahora va a venir una lista.

- Primer elemento
- Segundo elemento



Vídeo

Os dejamos un vídeo con lo visto hasta ahora:

<https://youtu.be/YT8lcCFEFOk>

Elementos en HTML

Ahora que ya sabemos lo básico para crear nuestro primer fichero HTML y hemos visualizado los primeros elementos, vamos a profundizar un poco en cómo deberíamos usar estas y otras etiquetas.

Como hemos visto, lo más común es estructurar nuestro HTML con contenedores `<div>`, pero existen más elementos:

- `<h2>`, `<h3>`, `<h4>`, `<h5>` y `<h6>` para declarar títulos y cabeceras de menor nivel
- `<button>` para declarar botones
- `<a>` para declarar enlaces a otras páginas
- `<input>` para declarar entradas de datos (como textos)

Vamos a verlos en detalle :)

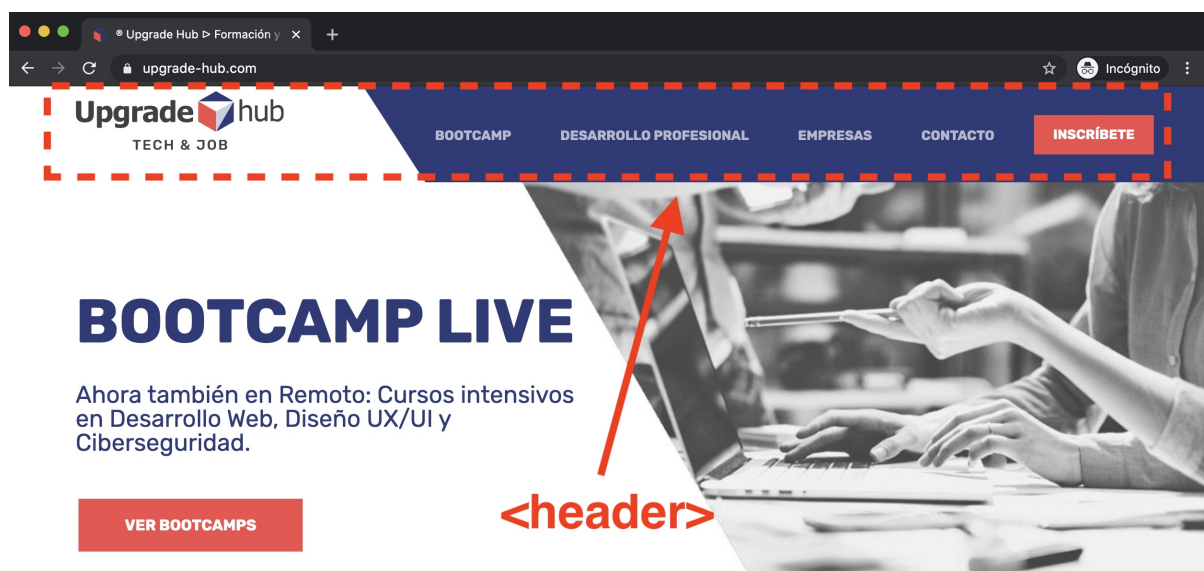
Etiquetas de estructura

Dentro de la página web encontramos ciertos elementos muy comunes, que nos sirven para dar estructura a nuestro documento, como por ejemplo la cabecera y el pie de página.

Veamos una lista de elementos y para qué podemos usarlos:

- **Header:** Se encuentra en la parte superior de la página y suele estar presente en todas las páginas de nuestro sitio web. Generalmente contiene los elementos principales de navegación y "marca", como son el menú, el acceso al login y el logotipo de la empresa. A nivel HTML está representado por la etiqueta `<header>`

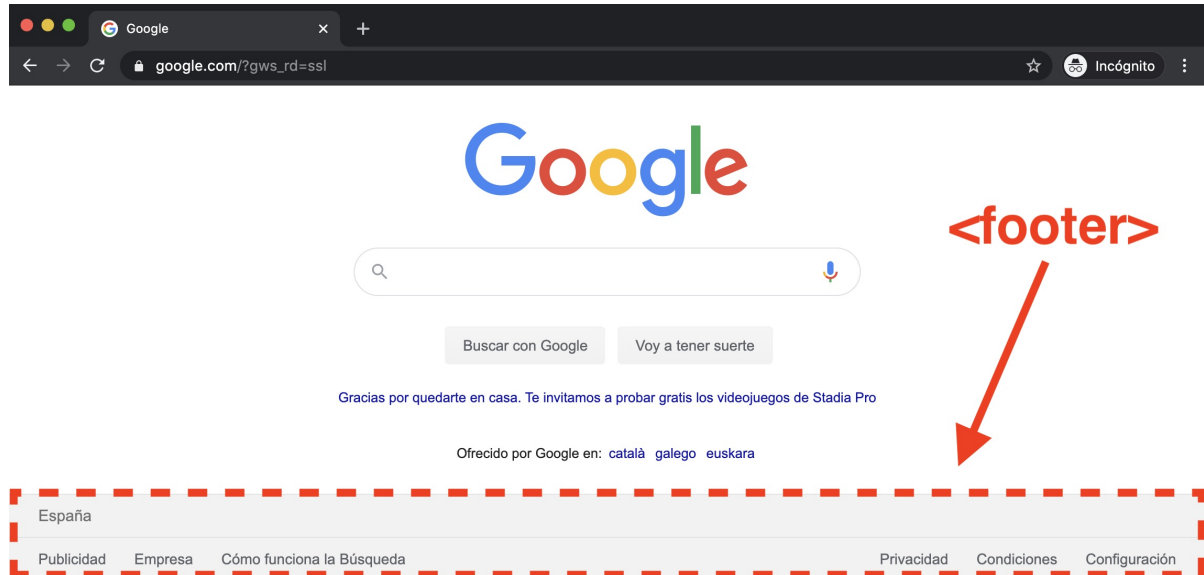
Veamos un ejemplo, el header de la página web de Upgrade Hub:



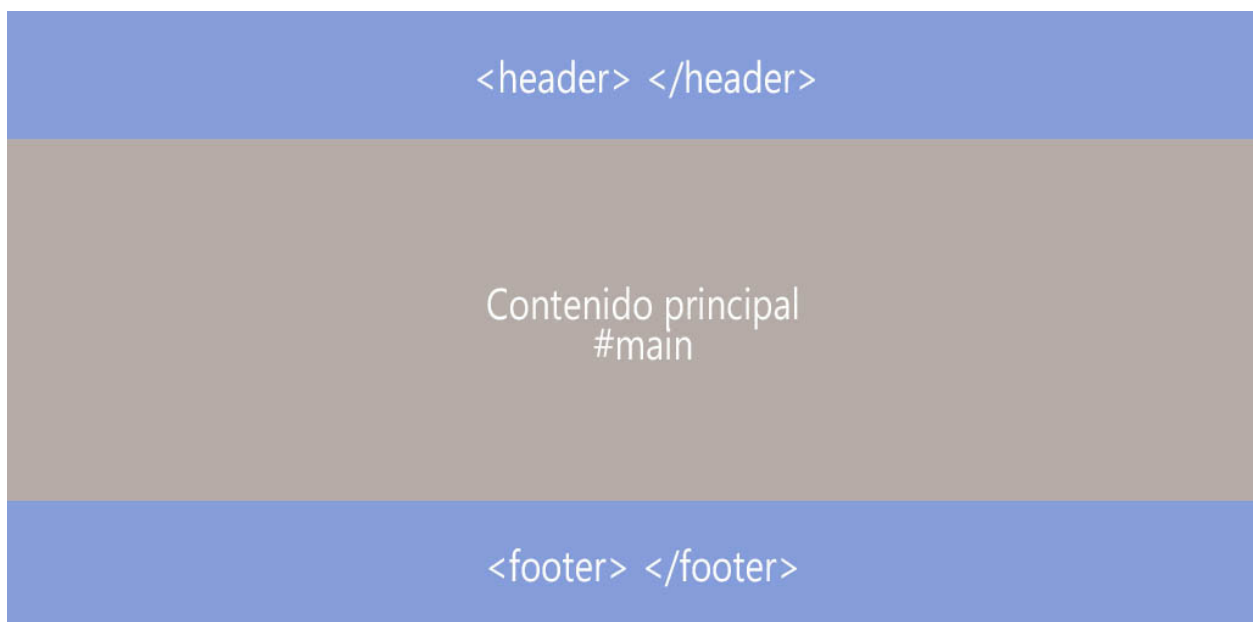
- **Footer:** Se encuentra en la parte inferior y al igual que el header, normalmente está presente en todas nuestras páginas. Es un elemento algo más secundario y normalmente se utiliza para dejar enlaces a redes sociales, secciones de ayuda, o el típico mensaje de copyright.

A nivel HTML está representado por la etiqueta `<footer>`

Veamos un ejemplo el footer la página web de Google:



Una estructura muy típica en web sería la siguiente:



Y a nivel de código, nuestro fichero **index.html** quedaría de la siguiente manera:

```
<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Web de prueba Footer y Header</title>
</head>

<body>
  <header>
    
    <a href="/">Ir a la home</a>
  </header>

  <div id="main">
    <!-- Aquí el contenido principal :) -->
    <h1>Hola!</h1>
    <p>Bienvenidos a nuestra web</p>
    <p>Eso es solo un ejemplo</p>
  </div>

  <footer>
    <p>Síguenos en nuestras redes sociales:</p>
    <a href="https://www.instagram.com/upgrade_hub">Instagram</a>
    <a href="https://www.facebook.com/UpgradeHub/">Facebook</a>
  </footer>
</body>

</html>
```

Etiquetas semánticas y de contenido

Hasta el momento hemos trabajado con el elemento **<div>** para envolver todos los elementos de nuestra web. Pero, ¿y si te dijésemos que hay más elementos similares?

Existen otros elementos que cumplen la misma función que los **<div>** pero de una manera más semántica, es decir, indica el objetivo de este elemento a otras personas que lean nuestro código y ayuda a mejorar nuestro posicionamiento web de cara al SEO.

Estos elementos son <article> y <section> ¡Vamos a explicarlos!

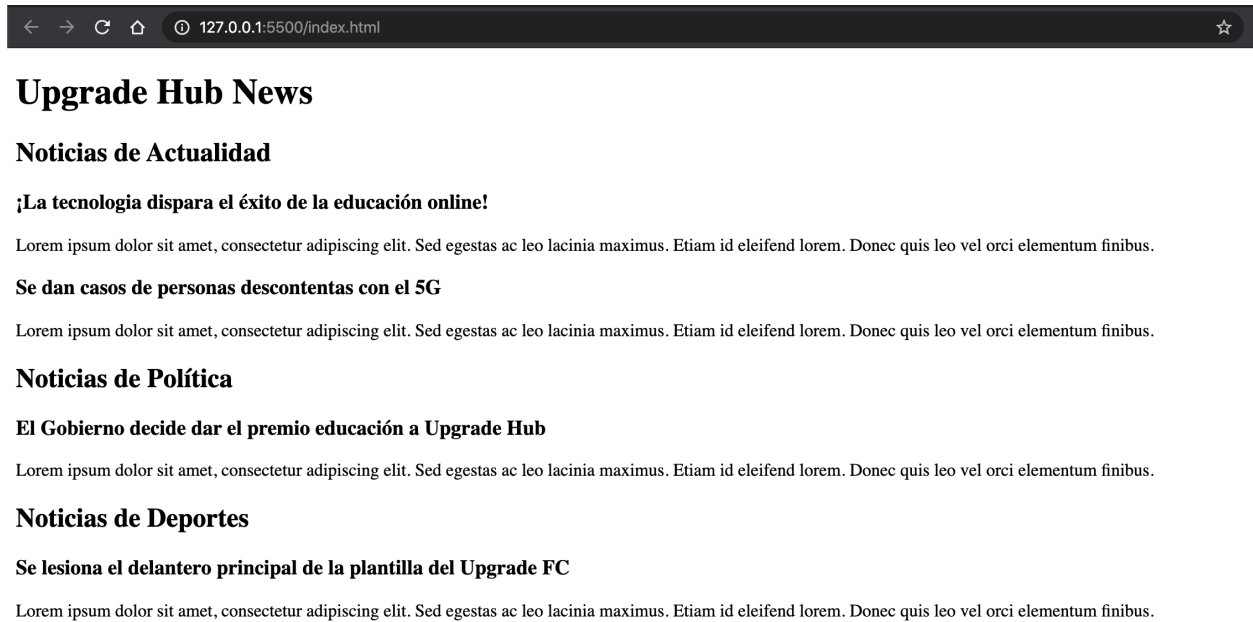
El elemento **<section>** lo usaremos para indicar que todo el contenido en su interior forma una sección importante de nuestra web. Como ejemplo, podríamos suponer una web de noticias donde mostramos noticias de Actualidad, Política y Deportes, cada elemento con noticias distintas podría ser una sección, tal y como vemos en el siguiente **<body>** de un archivo HTML:

```
<body>
  <h1>Upgrade Hub News</h1>
  <section>
    Noticias de Actualidad
  </section>
  <section>
    Noticias de Política
  </section>
  <section>
    Noticias de Deportes
  </section>
</body>
```

A este contenido le faltan las noticias, y para diferenciar cada noticia dentro de una sección. Para diferenciar conjuntos de elementos dentro de una sección de nuestra web usaremos el elemento **<article>**. Vamos a ver un ejemplo en el que añadiremos artículos a cada sección:

```
<section>
  <h2>Noticias de Actualidad</h2>
  <article>
    <h3>¡La tecnología dispara el éxito de la educación online!</h3>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed egestas
      ac leo lacinia maximus. Etiam id eleifend lorem. Donec quis leo vel
      orci elementum finibus.
    </p>
  </article>
  <article>
    <h3>Se dan casos de personas descontentas con el 5G</h3>
    <p>
      Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed egestas
      ac leo lacinia maximus. Etiam id eleifend lorem. Donec quis leo vel
      orci elementum finibus.
    </p>
  </article>
</section>
```

Si añadimos varios **<article>** más y abrimos en el navegador nuestro index.html obtendremos la siguiente vista web:



Con esto abremos aprendido dos nuevos elementos de estructura HTML, **<section>** y **<article>** 🚀

Formularios, inputs y botones

A lo largo de lo que llevamos visto de HTML te habrás preguntado, ¿cómo consigo que los usuarios me envíen información? Pues lo conseguiremos a través de formularios. ¡Vamos a verlos! 🎉

Un formulario puede enviar información a un servidor web a través de un método que conocemos como POST, y la dirección a la que enviaremos la información la introduciremos en el atributo **action**.

```
<form action="/"></form>
```

⬆️ Aquí tenemos un formulario con un atributo `action` que apunta a nuestra misma web, como no veremos servidores en el prework ya que tenemos que aprender primero más cosas en el curso, vamos a dejar por ahora el atributo **action** como acabamos de hacer en todos los ejercicios que completemos: `action="/"`

Pero, ¿cómo introducimos la información dentro del formulario? Usaremos elementos de tipo **<input>**, pero hay varios tipos de input que usaremos en HTML, vamos a explicar algunos de ellos. Pero antes, veremos un elemento más.

Elemento **<label>** para envolver a los inputs

Siempre que utilicemos un input, lo envolveremos en un elemento `<label>` para asignarle un título, aquí un ejemplo:

```
<label for="username">
  <p>Introduce tu nombre de usuario</p>
  <input name="username">
</label>
```

- El atributo **for="username"** lo utilizamos en la **<label>** para relacionar este **<label>** con el **<input>** cuyo atributo **name** sea igual a **username**

Esto lo veremos así en el navegador:

Introduce tu nombre de usuario

Como puedes observar, **<label>** actúa como una especie de **<div>**, pero su función es envolver a los elementos **<input>** y al texto que actúa como título para estos, que será un **<p>** en este ejemplo que acabamos de ver.

Elemento input de tipo texto **<input type="text">**

Los elementos input de tipo texto nos permiten escribir en ellos cadenas de texto. ¿Qué os parece crear un formulario que recoja nuestro nombre y apellidos?

```
<form action="/">
  <label for="firstName">
    <p>Introduce tu nombre de pila</p>
    <input name="firstName" type="text">
  </label>
  <label for="lastName">
    <p>Introduce tus apellidos</p>
    <input name="lastName" type="text">
  </label>
</form>
```

¿Te has fijado en el atributo **type="text"**? Este atributo nos indica el tipo de input que estamos utilizando, que será **text** en este caso.

Elemento input de tipo number **<input type="number">**

Para utilizar números, utilizaremos el input de tipo **number**. Vamos a mejorar el formulario anterior preguntado por la edad del usuario:

```
<form action="/">
  <label for="firstName">
    <p>Introduce tu nombre de pila</p>
    <input name="firstName" type="text">
  </label>
  <label for="lastName">
    <p>Introduce tus apellidos</p>
    <input name="lastName" type="text">
  </label>
  <label for="age">
    <p>Introduce tu edad</p>
    <input name="age" type="number">
  </label>
</form>
```

```
</label>  
</form>
```

Veamos en un GIF lo que hemos conseguido hasta ahora:

Introduce tu nombre de pila

Introduce tus apellidos

Introduce tu edad

¡Podemos aumentar o disminuir la cantidad usando las flechas del teclado!

Elemento input de tipo button <input type="submit">

Por último, entre los tipos de input que veremos en este prework, aprenderemos como enviar nuestro formulario. Esto lo haremos mediante un botón que actúe como **"submit"** y que conseguiremos mediante este nuevo tipo de input.

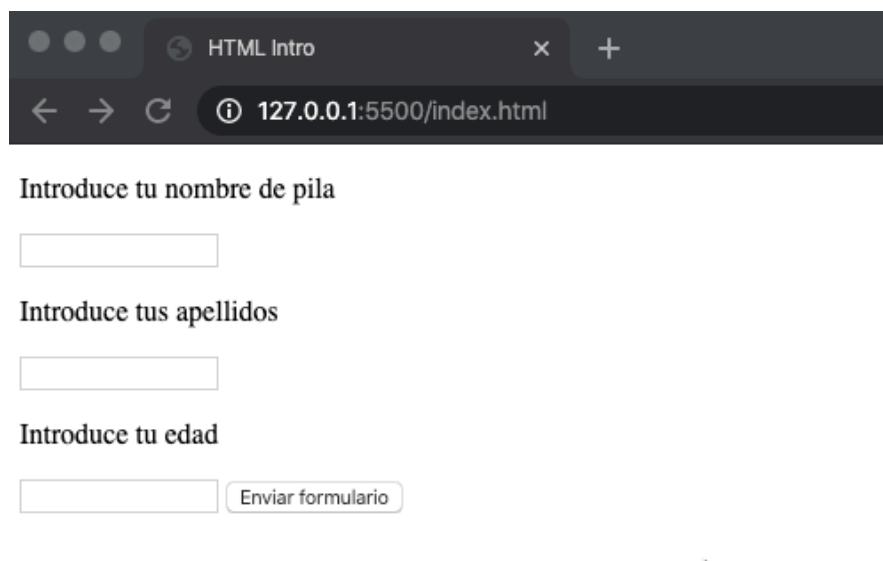
```
<form action="/">
  <label for="firstName">
    <p>Introduce tu nombre de pila</p>
    <input name="firstName" type="text">
  </label>
  <label for="lastName">
    <p>Introduce tus apellidos</p>
    <input name="lastName" type="text">
  </label>
  <label for="age">
    <p>Introduce tu edad</p>
    <input name="age" type="number">
  </label>

  <input type="submit" value="Enviar formulario"></input>
</form>
```

Este tipo de input no necesita **<label>** ni atributo **name=""**, y cuando hagamos click en el botón o pulsemos **Enter** cuando estemos escribiendo en el formulario, enviaremos la respuesta al servidor que elijamos.

También destacamos que hemos usado un nuevo atributo llamado **value**, que determinará el texto dentro del botón que aparezca como **<input type="submit">**. Siendo en este caso **"Enviar formulario"**.

Veamos un GIF para ver como funciona:



- ¡Cuidado! El formulario recargará la web cuando hagamos **Submit**.

¡Con esto hemos aprendido los elementos relacionados con formularios necesarios para empezar a trabajar! 🎉

¡A trabajar!

Ha llegado el momento de poner a prueba todo lo aprendido en este bloque, pasemos a realizar algún ejercicio 🧑💻

Recursos Extra

[W3SCHOOLS - HTML](#)