

«Talento Tech»

# Desarrollo Web 1

Clase 02





## Clase N° 2 | Conceptos básicos

### Temario:

- HTML semántico
- Etiquetas de Bloque y Etiquetas de Línea
- Creación de enlaces

## Etiquetas semánticas de HTML

Usar las etiquetas semánticas de **HTML** nos permite asignar significado a cada parte de una página web. Más que sólo presentar visualmente el contenido, podemos darle sentido a la estructura de nuestro sitio, lo cual va a facilitar su lectura tanto para los desarrolladores como para las máquinas.

### Veamos algunos ejemplos:

**<main>**: Indica el contenido principal de la página, excluyendo encabezados, pies y barras laterales. Es crucial para estructurar el contenido central de la página.

**<header>**: Utilizado para la cabecera de una página o sección, contiene información introductoria o de navegación.

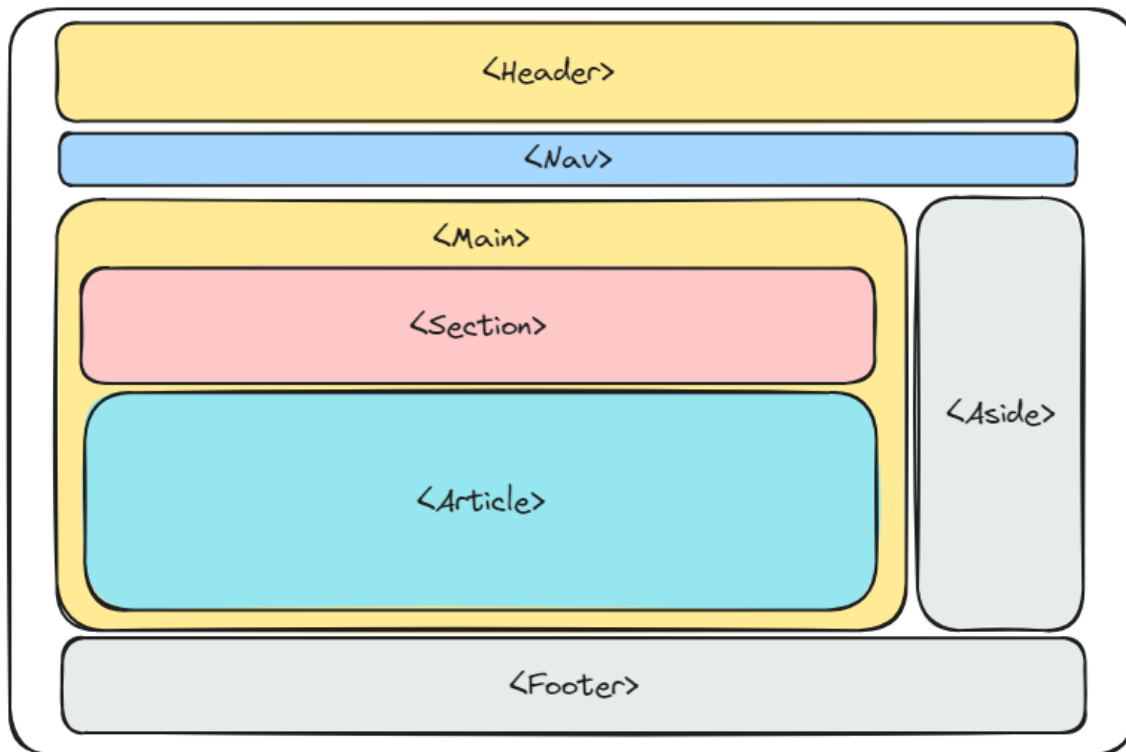
**<nav>**: Define una sección para la barra de navegación con enlaces a otras partes del sitio.

**<article>**: Destinado a contenido informativo, como noticias o blogs.

**<section>**: Nos sirve para dividir nuestra página en secciones. pueden contener varios elementos.

**<aside>**: Representa una sección de una página con contenido que está indirectamente relacionado con el contenido principal del documento. Estas secciones son a menudo representadas como barras laterales.

**<footer>**: Se utiliza para el pie de un documento o sección, con espacio para información de derechos de autor o enlaces relacionados.



## ¿Por qué es importante utilizarlas en nuestro sitio?

### Mejoran la Accesibilidad:

Ayudan a personas con discapacidades visuales o que utilizan tecnologías de asistencia a que puedan navegar y comprender nuestras páginas.

Beneficios para el **SEO** (Search Engine Optimization, Optimización para motores de búsqueda):

El uso correcto de etiquetas semánticas puede mejorar la visibilidad en los motores de búsqueda. Haciendo que sea más fácil encontrar nuestro sitio para las personas que están interesadas en nuestro contenido.

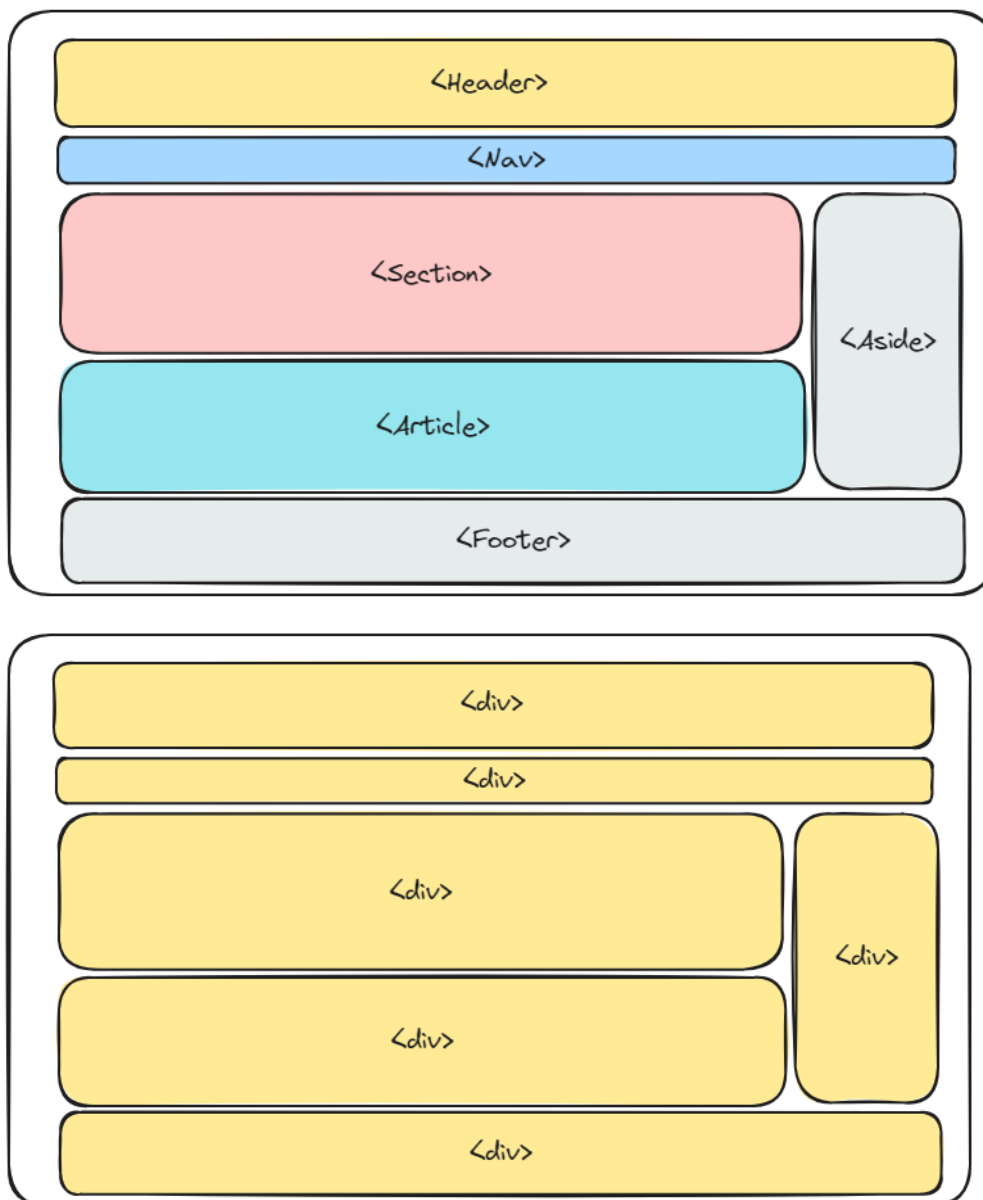
Los motores de búsqueda, como Google, utilizan algoritmos complejos para analizar y comprender el contenido de las páginas web. El HTML semántico proporciona una estructura clara y significativa, lo que facilita a los motores de búsqueda entender la relación y la importancia de cada elemento en la página.

El uso de etiquetas semánticas, como **<header>**, **<section>**, y **<article>**, ayuda a establecer una jerarquía en el contenido. Los motores de búsqueda van a interpretar esta jerarquía para determinar qué partes de la página son más relevantes. Por ejemplo, el contenido dentro de un elemento **<article>** se considera más relevante que el contenido en una sección de navegación **<nav>**.

### **Mantenimiento y Claridad del Código:**

Facilita la lectura y el mantenimiento del código, permitiendo una colaboración más eficiente en proyectos. La misma estructura podría lograrse utilizando solamente la etiqueta **<div>**. Pero no tendríamos la misma claridad de qué es exactamente el rol que cumple cada parte de la página.

### **Una comparación simple:**



Podemos ver que el uso de **HTML** semántico hace que sea mucho más fácil entender qué rol cumple cada elemento de nuestra página y como programadores va a ser más sencillo saber en qué elemento tenemos que trabajar cuando queramos hacer alguna modificación.

## Conclusión

El **HTML** semántico no sólo mejora la forma en que escribimos código, sino que también tiene un impacto positivo en la accesibilidad y la visibilidad en motores de búsqueda (como google). A lo largo del curso seguiremos trabajando con estas etiquetas y veremos cómo trabajan juntas para construir una estructura web significativa.

## Etiquetas de Bloque y de Línea

Profundizaremos en la diferencia entre etiquetas de bloque (block) y etiquetas de línea (inline) en HTML. Comprender estas diferencias es fundamental para la estructuración adecuada de nuestro contenido.

### Etiquetas de Bloque:

Las etiquetas de bloque definen bloques completos de contenido y generalmente inician en una nueva línea y ocupan todo el ancho disponible en su elemento padre (contenedor).

**<div>**: Utilizado como contenedor genérico para agrupar otros elementos y aplicar estilos.

**<p>**: Define un párrafo y se presenta en una nueva línea, con un espacio antes y después.

**<h1>**, **<h2>** ... **<h6>**: Encabezados que van desde el más importante **<h1>** hasta el menos importante **<h6>**. Estos también inician en una nueva línea.

**<ul>**, **<ol>**, **<li>**: Elementos de lista que organizan la información en bloques distintos.

## Etiquetas en Línea:

Estas etiquetas, por otro lado, no inician en una nueva línea y solo ocupan el ancho que necesitan. Son ideales para elementos que deben aparecer en la misma línea. Algunos ejemplos son:

**<span>**: similar a **<div>**, pero para contenidos en línea. Se utiliza para aplicar estilos o scripts a partes específicas de un texto.

**<a>**: Define un hipervínculo y puede contener texto e imágenes.

**<strong>** y **<em>**: Resalta el texto en negrita y cursiva respectivamente, sin iniciar una nueva línea.

## ¿Para qué vamos a utilizarlas?

Comprender la distinción entre etiquetas de bloque y etiquetas de línea es crucial para el diseño y la maquetación efectiva de una página web. Las etiquetas de bloque son esenciales para estructurar grandes secciones, mientras que las etiquetas de línea son útiles cuando necesitamos elementos en la misma línea, como parte de una oración o frase.

En general, los elementos en bloque pueden contener elementos en línea y otros elementos en bloque. La idea de que los elementos en bloque crean estructuras "más grandes" que los elementos en línea. Podríamos pensar cada elemento como una "caja" para nuestro contenido.

**Veamos el siguiente ejemplo:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```



```

<meta name="viewport" content="width=device-width,
initial-scale=1.0">
<title>Document</title>
</head>
<body>
  <article>
    <h2> Las etiquetas <em>h1, h2, h3, etc</em> <u> son elementos
  </u></h2>
    <p>La etiqueta <em>p</em> tambien es un elemento de <strong>
bloque </strong>, asi que tanto el titulo como el parrafo <u> van a
ocupar todo el ancho del elemento contenedor</u>. En este caso el
contenedor es la etiqueta <em>article</em> que tambine es un elemento
de bloque es decir que ocupa todo el ancho de su propio elemento
contenedor, <spam> en este caso es el</spam> elemento body</p>
  </article>
</body>
</html>
    
```



Ejemplo

127.0.0.1:5500

## Las etiquetas *h1*, *h2*, *h3*, *etc* son elementos de bloque

la etiqueta *p* tambien es un elemento de **bloque**. asi que tanto el titulo como el parrafo van a ocupar todo el ancho del elemento contenedor. En este caso el contenedor es la etiqueta *article* que también es un elemento de bloque es decir que ocupa todo el ancho de su propio elemento contenedor, en este caso es el elemento *body*

¿Qué conclusiones podemos sacar de este ejemplo?

**<body> (Elemento de Bloque):** Esta etiqueta contiene todo el contenido de la página. Ocupa todo el ancho disponible en la ventana del navegador.

**<article>** es otro elemento de bloque y se utiliza para agrupar contenido relacionado. En este caso, contiene un encabezado (**<h2>**) y un párrafo (**<p>**).

**<h2>** es un elemento de bloque que define un título. ocupa todo el ancho de **<article>** contiene elementos **<em>** y **<u>**

**<em>** y **<u>** son elementos de línea y se utilizan para resaltar texto en cursiva y subrayado respectivamente. Aunque estas etiquetas de línea están dentro de un bloque (**<h2>**), no inician una nueva línea por sí mismas.

**<p>** elemento de bloque que define un párrafo de texto. Ocupa todo el ancho de su contenedor **<article>**. Contiene varias etiquetas dentro, cada una utilizada para modificar la presentación del texto.

**<span>** también es un elemento de línea, pero en este caso, vemos que no modifica la apariencia del texto. la utilizaremos para asignarle estilos mediante sus atributos (class, id, style, etc). En la próxima clase veremos cómo asignarle estilos mediante CSS

### Conclusión:

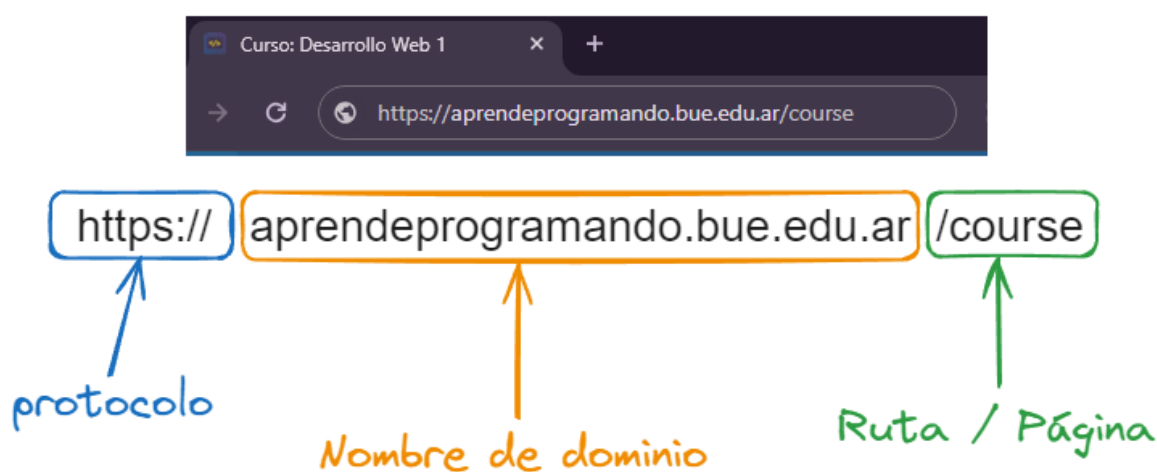
En resumen, el uso adecuado de etiquetas de bloque y etiquetas de línea es esencial para lograr una maquetación efectiva y estructurada en HTML. La comprensión de cuándo y cómo emplear estas etiquetas contribuye a la presentación visual y la legibilidad del contenido.

## Creación de enlaces

¿Alguna vez dijiste “pasame el link” y te mandaron algo escrito muy largo, de color azul y casi ilegible?.

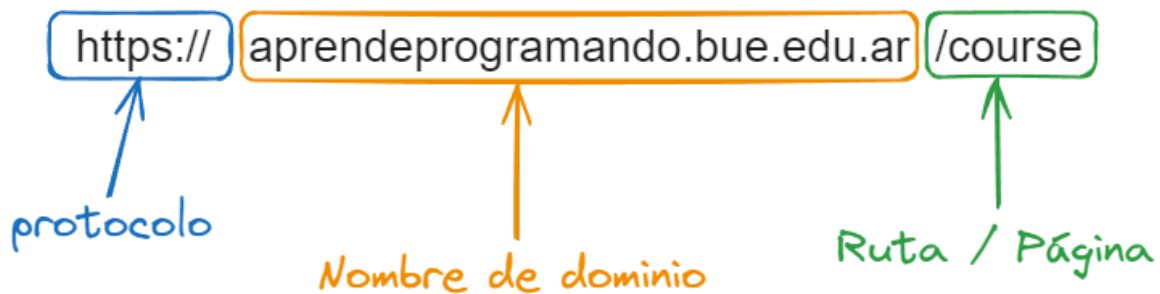
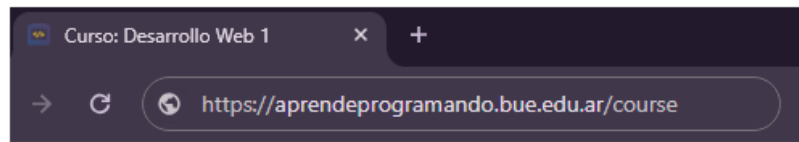


Ese link se lo conoce como **URL** (Uniform Resource Locator, que en español significa Localizador Uniforme de Recursos) y es la ubicación exacta de dónde se encuentra cualquier elemento que esté en internet, a esto lo llamamos ruta.



## ¿Cuál es la estructura de una URL?

Una URL por sus siglas en inglés es un localizador uniforme de recursos (URL). Para simplificar, podemos decir que es una dirección que apunta al servidor donde están alojados dichos recursos. Es decir: la ubicación exacta donde un elemento(puede ser una página de internet, una imagen, un video, etc. Para poder entenderlo mejor veremos la estructura de una URL:



Como pueden ver, una URL consta de varias partes:

#### Protocolo:

Suele ser http o https, se refiere a los protocolos de comunicación entre los servidores web y los navegadores.

La diferencia entre HTTP y HTTPS es que este último cifra el proceso de transmisión de datos. Este protocolo de seguridad protege mejor un sitio web y es un factor esencial para mejorar el posicionamiento en los resultados de búsqueda.

#### Nombre de dominio:

Es lo que escribimos en la barra de nuestro navegador cuando queremos ingresar a un sitio web determinado. Está compuesto por el nombre de la página y una extensión. (por ejemplo: aprendeprogramando es el nombre y .bue.edu.ar es la extensión).

#### Ruta del recurso:

Se encuentra a la derecha de la extensión. Se la conoce también como la estructura de carpetas del sitio web. Comienza con una barra diagonal '/'.

### Etiqueta <a>:

En HTML disponemos de la etiqueta **<a>**. Es una de las etiquetas más importantes ya que nos permite crear un enlace a otras páginas de internet, archivos o ubicaciones dentro de la misma página, direcciones de correo, o cualquier otra URL. Es fundamental para que los usuarios de nuestra página puedan moverse a distintas secciones de la misma o para que podamos redirigirlos a otras páginas.



## Atributos href y target

Las etiquetas **<a>** cuentan con los atributos **href** y **target**, que son esenciales para definir la dirección del enlace y cómo se abrirá el destino.

En el **atributo href** especificaremos la URL a la que se dirige el enlace. Puede apuntar a una sección específica de nuestra página, a una página interna (`segunda_pagina.html`), una página externa (`https://www.google.com`), o incluso a una dirección de correo electrónico (`mailto:aprendeprogramando@ejemplo.com`).

En el **atributo target** especificaremos donde queremos que se abra el nuevo enlace. (en una nueva pestaña, en la misma, o en una nueva ventana.)

## Rutas absolutas y relativas

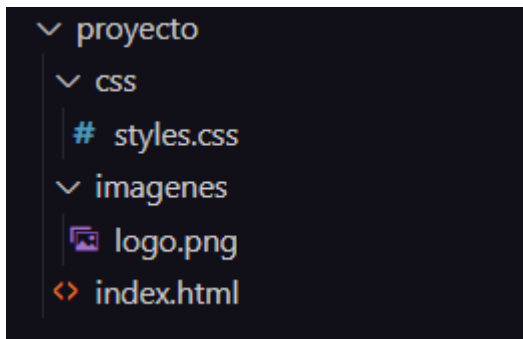
Las rutas absolutas especifican la ubicación completa de un recurso. Es decir, es la dirección web completa o la ruta completa desde la raíz del servidor. por ejemplo:

### Rutas relativas

Existen dos tipos principales de rutas relativas: las rutas relativas a directorios y las rutas relativas a archivos.

#### 1. Rutas Relativas a Carpetas:

Supongamos que tienes la siguiente estructura de carpetas:



Para acceder al archivo **"styles.css"** desde **"index.html"** debemos usar la ruta relativa en el atributo **href** de la etiqueta link

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

La ruta **"css/estilo.css"** indica que el archivo **"styles.css"** se encuentra dentro de la carpeta **"css"**, que está en el mismo nivel que el archivo **"index.html"**.

#### Rutas Relativas a Archivos:

Ahora imaginemos que queremos acceder a la imagen **"logo.png"** desde **"styles.css"**.



Deberemos hacerlo de la siguiente manera.

```
.fondo{
  background-image: url("../imagenes/banner.png");
}
```

En este caso, `../` indica que debes retroceder un nivel en la jerarquía de carpetas. Es decir una carpeta más "arriba". Luego, `imagenes/logo.png` especifica la ruta desde ese nivel hacia la carpeta "imagenes" y el archivo "logo.png".

## Inclusión de Imágenes en HTML

Ahora, pasemos a otro elemento crucial en la construcción de páginas web: las imágenes. En **HTML**, utilizamos la etiqueta `<img>` para insertar imágenes en nuestras páginas.

### ¿Cómo se utiliza la etiqueta `<img>`?

La etiqueta `<img>` no tiene una etiqueta de cierre, ya que es una etiqueta abierta. Aquí hay un ejemplo básico de cómo se utiliza:

```

```

### Analicemos parte por parte:

**src (source):** Especifica la ruta o URL de la imagen. Puede ser una ruta relativa al archivo de la imagen en tu proyecto o una URL completa si la imagen está en línea.

**alt (alternativo):** Proporciona un texto descriptivo de la imagen. Este texto se mostrará si la imagen no puede cargarse y es esencial para la accesibilidad, ya que las tecnologías de asistencia pueden leerlo a los usuarios con discapacidad visual.

**width y height:** Estos atributos permiten especificar el ancho y el alto de la imagen en píxeles. Ayudan a controlar el diseño de la página mientras la imagen se carga.



## Pregunta de reflexión

- ¿Por qué es importante considerar la accesibilidad al desarrollar una página web y cómo el HTML semántico nos ayuda en este aspecto?
- ¿Cuál es la función principal de las etiquetas de bloque en HTML? Proporciona un ejemplo de una etiqueta de línea y explica su propósito.
- ¿Por qué es importante distinguir entre etiquetas de bloque y etiquetas de línea al diseñar una página web?





## Desafío N° 1:

### 1. Estructura semántica:

- Utiliza las etiquetas semánticas para estructurar tu página. Incluye al menos `<header>`, `<nav>`, `<main>`, `<section>`, `<article>`, y `<footer>`.
- Define claramente el contenido principal usando la etiqueta `<main>`.

### 2. Enlaces y navegación:

- Crea una barra de navegación en la etiqueta `<nav>`, con una lista desordenada `<ul>`, elementos de la lista `<li>` y en su interior los enlaces a otras páginas de tu interés.

### 3. Imágenes:

- Incorpora imágenes en tu sitio web utilizando la etiqueta `<img>`

### 4. Entrega:

- Comprime la carpeta de tu proyecto en formato zip.
- Subilo a la plataforma





**Buenos Aires**  
*aprende*  
Agencia de Actividades para el Futuro

**BA** Buenos  
Aires  
Ciudad