

# Bootcamp Java Developer

Fase 2 - Java Web Developer  
Módulo 18



# HTML DOM

# Introducción

¿Qué es HTML? **El lenguaje más usado para la construcción de páginas web.** La sigla HTML significa *Hyper Text Markup Language* (lenguaje de marcas de hipertexto). No es un lenguaje de programación, sino de marcas con sentido a las que llamamos: ***etiquetas***.

**Las etiquetas le indican al navegador qué es lo que debe mostrar en pantalla.**



## Analizar código HTML

Una gran forma de aprender este lenguaje es leer código escrito por otros. **El código HTML de un sitio web es público** y se puede acceder a él de esta manera:

- Hacer clic derecho sobre una página web de tu interés.
- Seleccionar la opción **«Ver código fuente»** en el navegador. Ahí se puede ver realmente la información que está recibiendo el navegador web y cómo la está interpretando.

Otras formas de acceder al código fuente son:

- **en Internet Explorer**, simplemente hay que desplegar el menú **«Ver»** y luego elegir **«Código fuente»**;
- **en Chrome y Firefox** presionar **Ctrl+U**.

De esta forma, se abrirá una pestaña con el código fuente de la página que se esté viendo en ese momento en el navegador.

## HTML DOM: document.getElementById( )

Para poder hacer más interesante una aplicación es necesario **acceder a elementos del HTML** y mostrar la información.

Se puede hacer con JavaScript. **Los elementos del HTML se definen como objetos**, y cada uno **posee métodos y propiedades**.

En este caso accedemos al elemento a través del **id**, que es un atributo que trabajamos desde el HTML.

Por ejemplo, generamos un **enunciado** en el HTML:

```
<h1> Mi enunciado </h1>
```

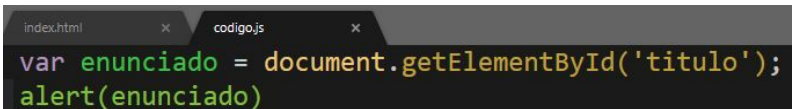
Luego, para acceder a éste, se debe trabajar con un **id**:

```
index.html x
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Desarrollo</title>
</head>
<body>
<h1 id="titulo"> Mi enunciado </h1>

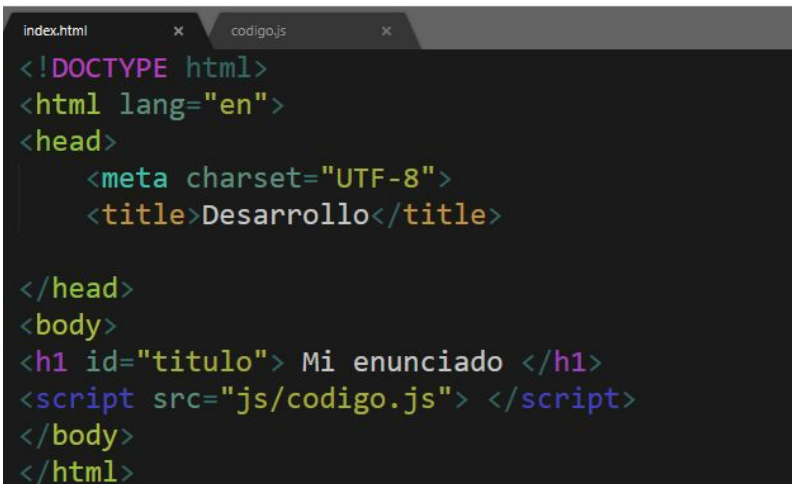
</body>
</html>
```

Este código mostrará el texto en una **alerta**:

Para que esto funcione correctamente, **el código debe estar vinculado debajo del enunciado**, ya que es necesario que éste se cargue para luego acceder a él. De otra forma, se accede a un elemento que no está en nuestro HTML. En la imagen de la derecha se puede observar correctamente:



```
index.html x codigo.js x
var enunciado = document.getElementById('titulo');
alert(enunciado)
```



```
index.html x codigo.js x
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Desarrollo</title>
</head>
<body>
  <h1 id="titulo"> Mi enunciado </h1>
  <script src="js/codigo.js"> </script>
</body>
</html>
```

## HTML DOM: innerHTML()

Para avanzar en output, el método `innerHTML()` provee la posibilidad de **acceder a la información de un elemento o modificar su contenido**. Si continuamos el caso anterior:

```
1 var enunciado = document.getElementById('titulo');  
2 console.log(enunciado.innerHTML);
```

El resultado será:

Esta página dice

Mi enunciado

Aceptar



Para **modificar el valor** de texto:

```
index.html x código.js  
var enunciado = document.getElementById('titulo');  
enunciado.innerHTML = "Aca hay un nuevo contenido"
```

De esta forma podemos acceder a un elemento en el HTML DOM, modificar su contenido u obtenerlo. Otra manera de hacerlo, también muy simple, es a través de **document.querySelector()**, que **permite acceder tanto a un elemento por su tipo como por su class o id**.

```
index.html x código.js  
var enunciado = document.querySelector('h1');  
enunciado.innerHTML = "Aca hay un nuevo contenido"
```

En el caso anterior, tenemos estas tres posibilidades ciertas:

```
index.html x  código.js x
var enunciado = document.querySelector('h1');
var enunciado = document.querySelector('#titulo');
var enunciado = document.getElementById('titulo');
```

# Revisión

- Repasar el concepto de **objeto** en JavaScript.
- Implementar **document.getElementById()** para acceder al DOM.
- **Avanzar con el output** a través de **innerHTML()**.
- Trabajar en el **Proyecto Integrador**.
- Realizar las preguntas necesarias al/la docente antes de continuar.



**¡Sigamos  
trabajando!**

