

Lección 17

# DOM

# ¿Qué es DOM?



## ¿Qué es DOM?

---

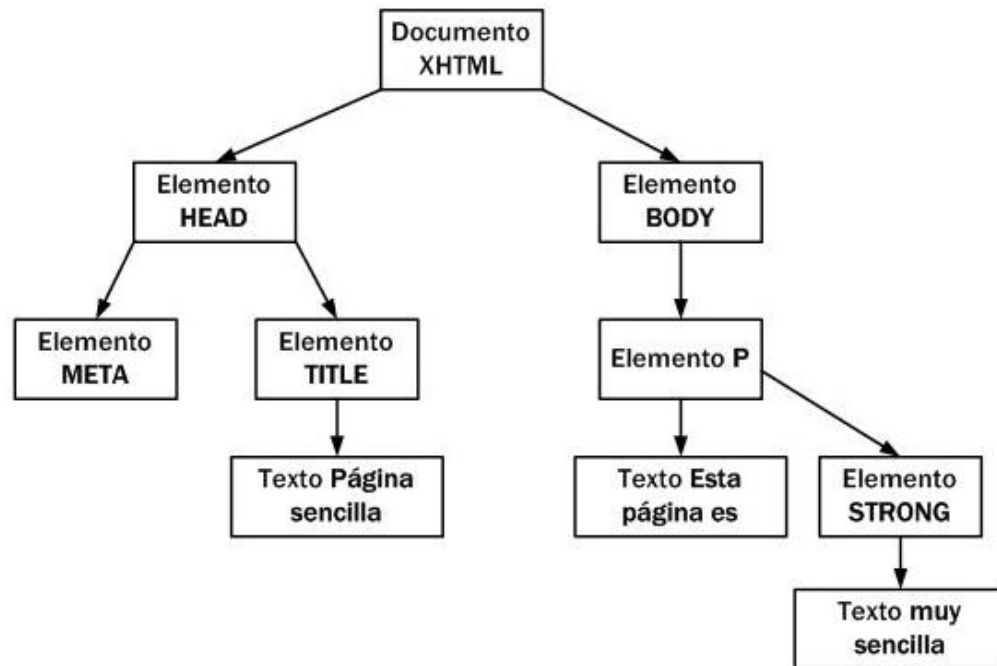
DOM: Modelo de Objetos del Documento es una interfaz de programación que puede **acceder y modificar el contenido, estructura y estilo** de los documentos HTML y XML.



## ¿Qué es DOM?

---

DOM transforma los documentos en un conjunto de elementos llamados **nodos** los cuales están interconectados, representando los contenidos de las páginas web y las relaciones entre ellos.



## ¿Qué es DOM?

---

DOM permite acceso a la estructura de una página HTML mediante el mapeo de los elementos de esta página en un árbol de nodos. Cada elemento se convierte en un **nodo** y cada porción de texto en un **nodo de texto**.

Es importante comprender la diferencia entre elementos y nodos de textos.

Todas las etiquetas son elementos, tales como <p>, <div>, <img> por lo que tienen **atributos** y contienen **nodos hijos**, pero los nodos de texto no poseen atributos ni hijos.

¿Qué es BOM?



## ¿Qué es BOM?

---

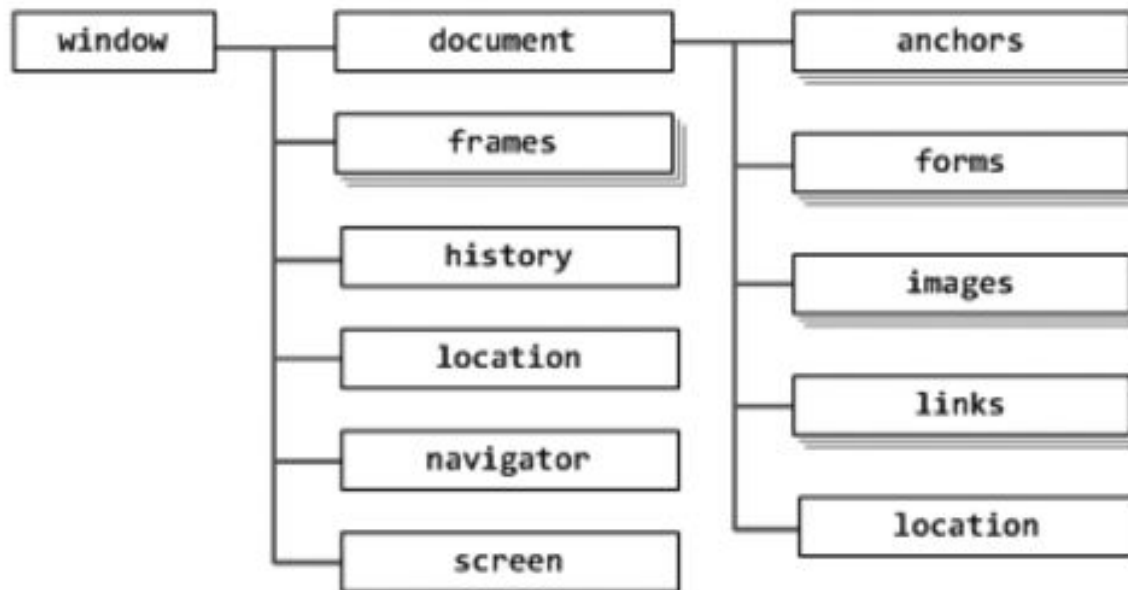
BOM: Browser Object Model permite acceder y modificar las propiedades de las **ventanas** del propio navegador. Con él se puede redimensionar y mover el navegador, modificar textos de la barra de estado y realizar manipulaciones no relacionadas con el HTML.

El mayor inconveniente de BOM es que, al contrario de lo que sucede con DOM, ninguna entidad se encarga de su estandarización o de definir parámetros mínimos de interoperabilidad entre navegadores.

## ¿Qué es BOM?

---

BOM está compuesto por varios objetos relacionados entre sí, tal como lo podemos ver en la siguiente imagen





Modificando  
contenido de los  
elementos



## Modificando contenido de los elementos

---

Para modificar el contenido de los elementos de nuestro DOM podemos hacerlo de varias maneras:

- **.innerHTML**: nos permite acceder al código HTML comprendido entre las etiquetas de apertura y cierre que definen el nodo y modificarlo si lo deseamos. Con esta se puede modificar completamente el HTML, no sólo el texto.

## Modificando contenido de los elementos

---

- **.textContent**: Sirve para extraer (o modificar) el texto completo contenido en todos los nodos de un nodo padre. Se rescatan todos los elementos anidados.
- **.innerText**: Su comportamiento es muy similar al de `textContent`, aunque no exactamente igual. `innerText` no es reconocido por todos los navegadores y no se considera un estándar válido

## Modificando contenido de los elementos

---

- **.value**: establece o retorna el atributo **value** de un campo de texto, pudiendo ser el valor por defecto o el texto que ingresa el usuario.
  - Valor por defecto: ""

Correo

El atributo **value** del input es:

- Valor ingresado por el usuario:  
"sumate@laboratoria.la"

Correo

El atributo **value** del input es: sumate@laboratoria.la

# Obtener elementos selectores



## Obtener elementos selectores

---

**.getElementsByTagName()**: Es una función que obtiene todos los elementos de la página XHTML cuya **etiqueta** sea igual que el parámetro que se le pasa.

```
//selecciona los elementos indicados por parámetro  
document.getElementsByTagName('li');
```

## Obtener elementos selectores

---

**.getElementById()**: Es una función que devuelve el elemento XHTML cuyo atributo id coincide con el parámetro indicado en la función.

Las **id** son únicas, por lo tanto la función devuelve únicamente el elemento deseado.

```
//selecciona el elemento que contiene el id indicado por parámetro  
document.getElementById('nombre');
```

## Obtener elementos selectores

---

**.getElementsByClassName()**: Es una función que devuelve todos los elementos de la página XHTML cuya **clase** sea igual que el parámetro que se le pasa a la función.

Puede ser más de una clase indicada por parámetro.

```
//selecciona los elementos que contengan la clases indicada por parámetro  
document.getElementsByClassName('edad');  
  
document.getElementsByClassName('edad estatura');
```



## Obtener elementos selectores

---

**.querySelector():** Acepta como parámetro un selector que identifica el elemento (o elementos) a seleccionar. La función devuelve solamente el primero de estos que cumpla la condición. Si no existe el elemento, el valor retornado es null.

```
//selecciona el primer selector del grupo con el parámetro indicado  
document.querySelector('.mascotas');
```

## Obtener elementos selectores

---

**.querySelectorAll():** Esta función toma todos los elementos con el selector asignado y devuelve un objeto de tipo NodeList con los elementos que coincidan.

La diferencia entre usar `document.querySelector` y `document.getElementById` es que el primero es compatible desde IE8+ mientras que el segundo es compatible desde IE5

```
//selecciona la colección completa de selectores del  
//grupo con el parámetro indicado  
document.querySelectorAll('.mascotas');
```