

# CURSO: Desarrollo de Entornos Web

## Unidad 2: El Modelo de Objeto de Documento (DOM)

### Tema 2: **Modelo DOM**



# Índice

2.1.1 Definición

2.1.2 Árbol de nodos

2.1.3 Tipos de nodos

2.1.4 Acceso directo a los nodos: getElementById, getElementsByTagName, getElementsByClassName

2.1.5 Acceso directo a los atributos.



# Capacidades

Diseña una página web utilizando el modelo de objetos.



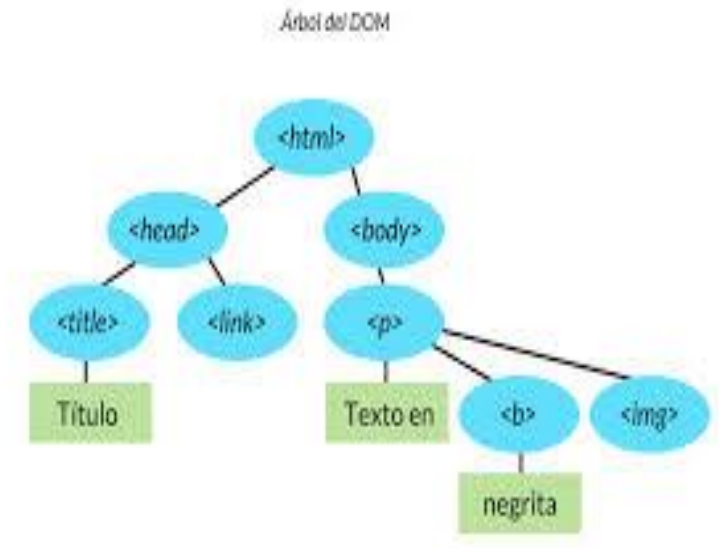
## DOM: Definición

- *Document Object Model* o DOM es una interfaz de programación de aplicaciones (API) que representa a los documentos HTML y XML.
- Define la estructura lógica de los documentos HTML y el modo en que se accede a los elementos de éste por medio de Javascript, para modificar sus propiedades o invocar a sus métodos.



## DOM: Definición

- **DOM** se representa como un árbol donde los elementos de la página web están organizados de manera jerárquica, con nodos superiores (nodos padre o parent) y nodos que derivan de los nodos padre (nodos hijo o child)

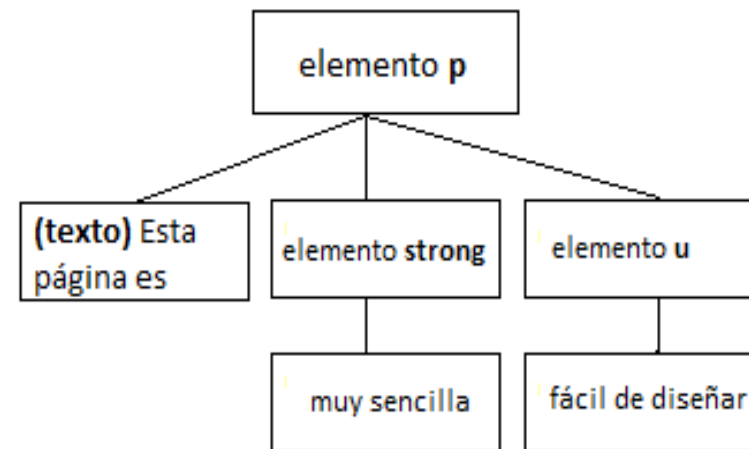


## DOM: Definición

La siguiente etiqueta HTML:

```
<p>Esta página es <strong>muy sencilla </strong> y  
<u>fácil de diseñar</u></p>
```

Genera los siguientes nodos:



## Accediendo a los Elementos

### Tipos de nodos

- ***Document***, nodo raíz del que se derivan los demás nodos.
- ***Element***, representa cada una de las etiquetas. Nodo que contiene atributos y se derivan otros nodos.
- ***Attr***, se define un nodo de este tipo para representar cada uno de los atributos de las etiquetas HTML.
- ***Text***, nodo que contiene el texto encerrado por una etiqueta HTML.



# Accediendo a los Elementos

## Acceso a los elementos

Javascript permite acceder a los elementos de una página utilizando algunos métodos y propiedades.

- ***getElementById()***: accede a un elemento del documento a través de su ID asignado.
- ***getElementsByTagName()***: accede a los elementos del documento a través de su nombre de etiqueta
- ***getElementsByClassName()***: accede a los elementos del documento a través de su nombre de clase.
- ***getElementsByName()***: accede a los elementos del documento a través de su nombre





## Manipulando el texto

- Acceder al texto de un elemento

- Accede al texto del elemento p1

```
var texto = document.getElementById("p1").innerText;  
document.write(texto);
```

- Modifica el contenido del elemento div1

```
document.getElementById("div1").innerHTML =  
"<p style='color:yellow;'>Introduzca los datos</p>";
```



## Manipulando los Estilos

Se puede manipular el estilo al documento o a un elemento definido, a través de la propiedad **style**

```
document.body.style.fontSize="24px"
```

```
document.getElementById("parrafo").style.fontSize="24px"
```

o tambien de esta forma:

```
var obj = document.getElementById("parrafo");
```

```
obj.style.color = "red";
```

```
obj.style.fontSize = "25px";
```



## Manipulando los Estilos

El método **getElementsByClassName** se genera un arreglo de elementos de la clase. Para manipular sus estilos, hay que recorrer el arreglo con una estructura de repetición.

```
for(i=0; i<document.getElementsByClassName("p").length; i++)  
{  
    document.getElementsByClassName("p")[i].style.color="red";  
    document.getElementsByClassName("p")[i].style.fontSize="14px";  
}
```

El mismo procedimiento se aplica a los métodos **getElementsByName** y **getElementsByTagName**.



# Conclusiones

- Document Object Model o DOM es una interfaz de programación de aplicaciones (API) que representa a los documentos HTML y XML.
- Define la estructura lógica de los documentos HTML y el modo en que se accede a los elementos de éste por medio de Javascript.
- DOM se representa como un árbol organizado de manera jerárquica, con nodos superiores (nodos padre o parent) y nodos que derivan de los nodos padre (nodos hijo o child)
- Javascript permite acceder a los elementos de una página utilizando algunos métodos y propiedades: `getElementById()`, `getElementsByTagName()`, `getElementsByClassName()`, `getElementsByName()`
- Se puede manipular el estilo al documento o a un elemento definido, a través de la propiedad `style`.



GRACIAS

