

Tema 07: Modelo de Objetos del Documento (DOM)

**Lenguajes de Marcas y SGI
Administración de Sistemas Informáticos en Red
IES Los Manantiales**

1.- Introducción

El Document Object Model (DOM) transforma una página web en un árbol de nodos con el que el podemos manipular los diferentes elementos de una página web.

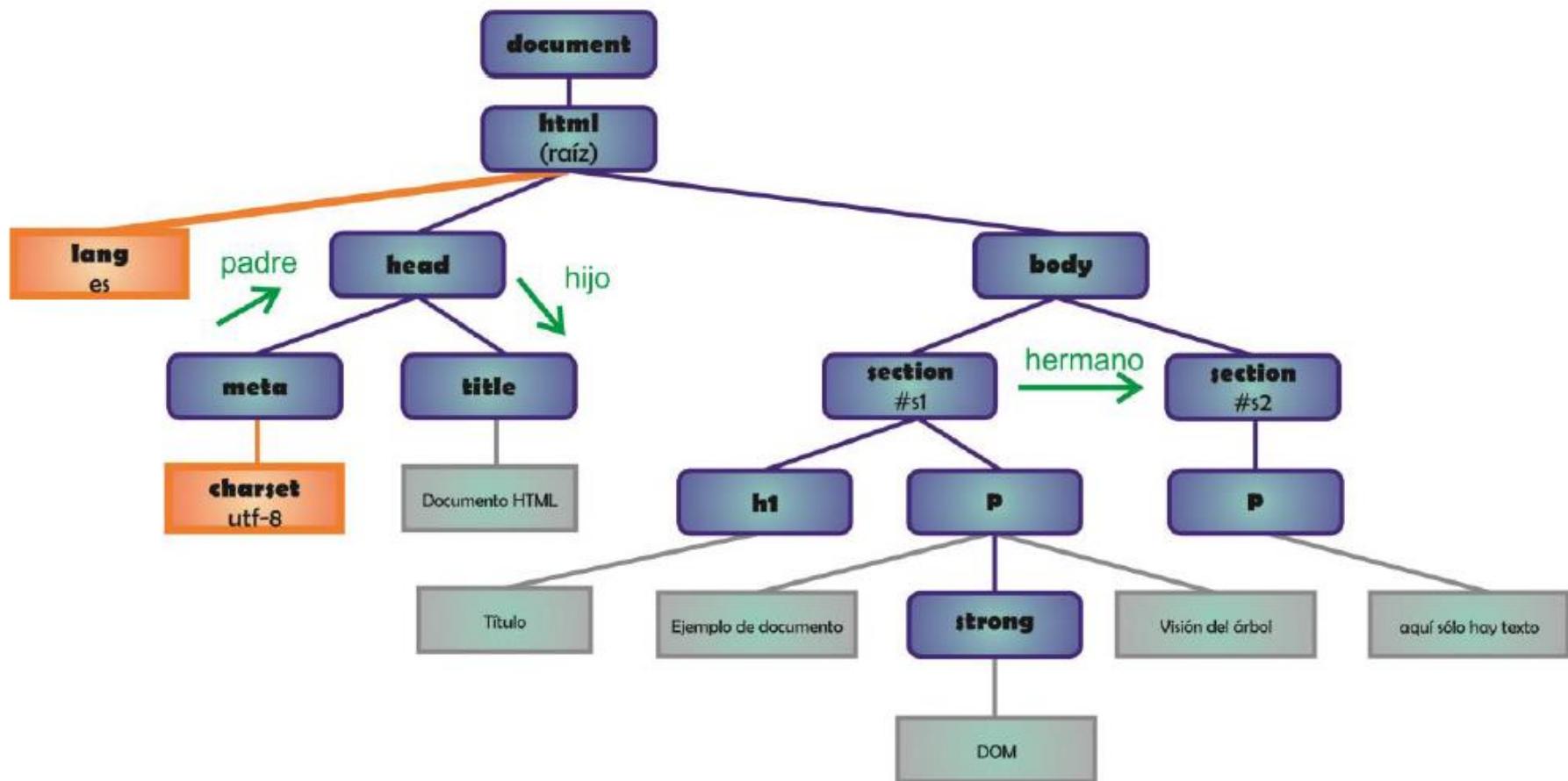
Cada nodo representa un elemento de la página. Podemos crear nodos nuevos, modificarlos o eliminarlos en cualquier momento.

Esta transformación de la página en árbol de nodos la realiza el navegador de forma automática.

2.- Árbol de nodos

```
<!doctype html>
<html lang="es">
<head>
  <meta charset="UTF-8">
  <title>Documento HTML</title>
</head>
<body>
  <section id="s1">
    <h1>Título</h1>
    <p>Ejemplo de documento <strong>DOM</strong> Visión en árbol</p>
  </section>
  <section id="s2">
    <p>Aquí sólo hay texto</p>
  </section>
</body>
</html>
```

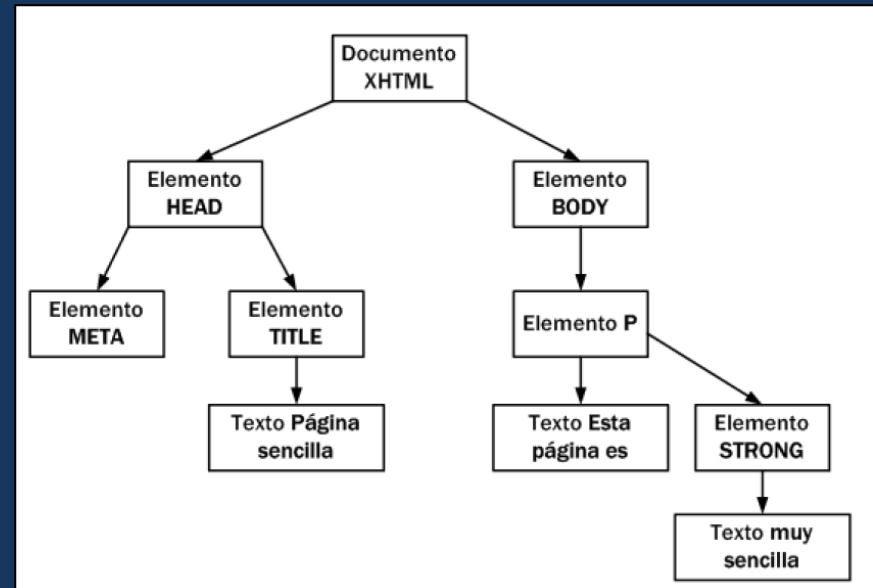
2.- Árbol de nodos



2.- Árbol de nodos

- Cada rectángulo es un nodo DOM.
- La raíz del árbol será siempre un nodo de tipo documento.
- Cada etiqueta de la página se transforma en un nodo de tipo Elemento mas otro nodo de tipo texto con el contenido de esa etiqueta.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html;
<title>Página sencilla</title>
</head>
<body>
<p>Esta página es <strong>muy sencilla</strong></p>
</body>
</html>
```



3.- Acceso a los nodos

Una vez construido el árbol de nodos, tenemos la posibilidad de acceder a cualquier nodo para alterar el contenido de la página web. Podemos crear nuevos elementos, eliminarlos, modificarlos, cambiar su ubicación, etc.

DOM permite acceder a los nodos de dos formas:

- **Jerárquica**: en la que accedemos a un nodo a través de sus nodos padre.
- **Directa**: en la que accedemos al nodo a través del nombre de su etiqueta, su nombre o Id.

4.- Acceso a los nodos. Buscar Elementos HTML

- Buscar elementos por su nombre etiqueta (<div>)
 - `getElementsByTagName`
- Buscar elementos por su nombre (`name = ...`)
 - `getElementsByName`
- Buscar elementos por su ID (`id = ...`)
 - `getElementById`
- Buscar elementos por su clase (`class = ...`)
 - `getElementsByClassName`

4.1.- Acceso a través de etiquetas

`getElementsByName()`

Obtiene todos los elementos de la página cuya etiqueta sea igual que el parámetro que se le pasa.

```
var parrafos = document.getElementsByTagName("p");
```

La función devuelve un array de nodos DOM con todos los elementos que pertenecen a la etiqueta <p>. Podemos recorrer el array para llegar a cada párrafo en particular.

```
for(var i=0; i<parrafos.length; i++) {  
    var parrafo = parrafos[i];  
}
```


4.1.- Acceso a través de etiquetas

Ejemplo de captura de párrafo

```
<p>Párrafo 1. Lorem Ipsum es simplemente el texto de relleno  
<p>Párrafo 2. Al contrario del pensamiento popular, el texto  
<p>Párrafo 3. Es un hecho establecido hace demasiado tiempo c  
<br><br>  
<h2>Acceder a un párrafo de la página con DOM</h2>  
<br>  
<form name="f1">  
    Introduce el párrafo a mostrar (1..3) .:  
    <input type="text" name="n1" value="" size="5"/><br>  
    <input type="submit" value="Obtener Párrafo" onclick="  
    var parrafos=document.getElementsByTagName('p') ;  
    miparrafo = parrafos[parseInt(f1.n1.value)-1] ;  
    alert(miparrafo.innerHTML) ;">  
</form>
```

Tras conseguir todos los párrafos, seleccionamos uno indicado por el usuario. Mostramos el párrafo seleccionado leyendo el valor con la propiedad innerHTML.

4.2.- Acceso a través de nombre

`getElementsByName()`

Obtiene todos los elementos de la página cuyo nombre sea igual que el parámetro que se le pasa.

```
var parrafoEspecial = document.getElementsByName("especial");
```

```
<p name="prueba">...</p>
```

```
<p name="especial">...</p>
```

```
<p>...</p>
```

En este caso sólo se pasaría un párrafo, ya que sólo uno tiene el name="especial".

4.3.- Acceso a través de Id

getElementById()

Obtiene el elemento de la página cuyo Id sea igual que el parámetro que se le pasa.

```
<p id="uno">Párrafo 1 (uno). Lorem Ipsum es simplemente el te  
<p id="dos">Párrafo 2 (dos). Al contrario del pensamiento pop  
<p id="tres">Párrafo 3 (tres). Es un hecho establecido hace d  
<br><br>  
<h2>Acceder a un párrafo de la página por Id con DOM</h2>  
<br>  
<form name="f1">  
    Introduce el párrafo a mostrar.:  
    <input type="text" name="n1" value="" size="5"/><br>  
    <input type="submit" value="Obtener Párrafo" onclick="  
    var parrafo=document.getElementById(f1.n1.value) ;  
    alert(parrafo.innerHTML) ;">
```

Ejemplo de captura de párrafo

4.3.- Acceso a través de Id. Cambiar hipervínculo

```
<head>
  <script>
    function cambiarEnlace() {
      var enlace=document.getElementById('nacional');
      enlace.innerHTML="El Pais";
      enlace.href="http://www.elpais.es";
      enlace.target="_blank";
      alert(enlace.href);      }
    </script>
  </head>
  <body>
    <a id="deportes" href="http://www.as.com">As.com</a> </br>
    <a id="nacional" href="http://www.elmundo.es">El Mundo</a> </br>
    <a id="local" href="http://www.diariosur.es">Diario Sur</a> </br>
    <a id="internacional" href="http://www.clarin.com">Clarín</a> </br>

    <input type="button" onclick="cambiarEnlace()" value="Cambia enlace">
  </body>
```

Ejemplo de cambio de hipervínculo

4.4.- Acceso a través de Clase

getElementsByClassName()

Obtiene todos los elementos de la página cuya clase sea igual que el parámetro que se le pasa.

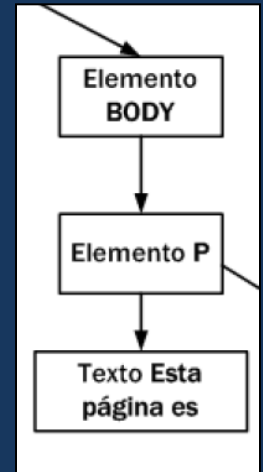
```
<p class="uno">Párrafo 1 (clase uno). Lorem Ipsum es simplemente el texto de r
<p class="uno">Párrafo 2 (clase uno). Al contrario del pensamiento popular, el
<p class="uno">Párrafo 3 (clase uno). Es un hecho establecido hace demasiado t
<br><br>
<h2>Acceder a los párrafos de la página de la clase 'uno' con DOM</h2>
<br>
<form name="f1">
    <input type="submit" value="Obtener Párrafos de la Clase 'uno'" onclick="
    var parrafo=document.getElementsByClassName( 'uno' ) ;
    alert(parrafo[0].innerHTML) ;
    alert(parrafo[1].innerHTML) ;
    alert(parrafo[2].innerHTML) ;
```

Ejemplo de captura por clase

5.- Creación de Nodos

Una de las propiedades más importantes del DOM es la posibilidad de añadir a nuestra web elementos en el momento que queramos.

Para crear un nuevo nodo es necesario realizar cuatro fases:



1. Creación de un nodo de tipo **Element** que represente al elemento.
2. Creación de un nodo de tipo **Text** que represente el contenido del elemento.
3. Añadir el nodo **Text** como nodo hijo del nodo **Element**.
4. Añadir el nodo **Element** a la página, en forma de nodo hijo del nodo correspondiente al lugar en el que se quiere insertar el elemento.

5.1.- Creación de Nodos de tipo Texto

```
// Crear nodo de tipo Element  
var parrafo = document.createElement("p");  
  
// Crear nodo de tipo Text  
var contenido = document.createTextNode("Hola Mundo!");  
  
// Añadir el nodo Text como hijo del nodo Element  
parrafo.appendChild(contenido);  
  
// Añadir el nodo Element como hijo de la pagina  
document.body.appendChild(parrafo);
```

5.1.- Creación de Nodos de tipo Texto

```
<head>
  <script>
    function crearnodo(parrafo) {
      var p = document.createElement('p');
      var texto = document.createTextNode(parrafo);
      p.appendChild(texto);
      document.body.appendChild(p); }
  </script>
</head>

<body>
  <p id="uno">Párrafo 1 (uno). Lorem Ipsum es simplemente el texto de relleno
  <p id="dos">Párrafo 2 (dos). Al contrario del pensamiento popular, el texto
  <p id="tres">Párrafo 3 (tres). Es un hecho establecido hace demasiado tiempo
  <h2>Insertar un párrafo en la página con DOM</h2> <br>
  <form name="f1">
    Introduce el párrafo a insertar.:
    <input type="text" name="n1" value="" size="100"/>
  </form>
  <a href="javascript:crearnodo(f1.n1.value)">Click para añadir párrafo</a>
</body>
```


5.2.- Creación de un Nodo hr controlando posición

```
<head>
  <script>
    function crearlinea(sitio){
      var linea = document.createElement('hr');
      document.getElementById(sitio).appendChild(linea);
    }
  </script>
</head>
<body>
  <p id="uno">Párrafo 1 (uno). Lorem Ipsum es simplemente el texto de relleno de las
  <p id="dos">Párrafo 2 (dos). Al contrario del pensamiento popular, el texto de Lore
  <p id="tres">Párrafo 3 (tres). Es un hecho establecido hace demasiado tiempo que un
  <h2>Insertar una línea en la página con DOM</h2>
  <form name="f1">
    Introduce dónde quieres insertar la línea.:
    <input type="text" name="n1" value="" size="5"/>
  </form>
  <div id="aqui">
    <a href="javascript:crearlinea(f1.n1.value)">Click para añadir una línea.</a>
  </div>
</body>
```

Ejemplo de inserción de línea especificando lugar

5.3.- Creación de un nodo Hipervínculo

```
<head>
  <script>
    function crearenlace() {
      var enlace=document.createElement('a');
      enlace.setAttribute('href','http://www.as.com');
      enlace.setAttribute('target','_blank');
      var texto=document.createTextNode('Pincha aquí para visitar as.com');
      enlace.appendChild(texto);
      document.getElementById('aqui').appendChild(enlace); }</script>
</head>
<body>
  <p id="uno">Párrafo 1 (uno). Lorem Ipsum es simplemente el texto de relleno de 1
  <p id="dos">Párrafo 2 (dos). Al contrario del pensamiento popular, el texto de I
  <p id="tres">Párrafo 3 (tres). Es un hecho establecido hace demasiado tiempo que
  <br><br>
  <h2>Insertar un enlace en la página con DOM</h2> <br>
  <div id="aqui">
    <p><a href="javascript:crearenlace()">Click para añadir un enlace</a>.</p>
  </div>
</body>
```

Ejemplo de inserción de enlace

6.- Eliminar Nodos

```
<head>
  <script>
    function eliminar(parrafo) {
      var nodo=document.getElementById(parrafo);
      nodo.parentNode.removeChild(nodo);
    }
  </script>
</head>

<body>
  <p id="uno">Párrafo 1 (uno). Lorem Ipsum es simplemente el texto de relleno
  <p id="dos">Párrafo 2 (dos). Al contrario del pensamiento popular, el texto
  <p id="tres">Párrafo 3 (tres). Es un hecho establecido hace demasiado tiempo
  <h2>Eliminar un párrafo en la página con DOM</h2> <br>
  <form name="f1">
    Introduce el ID del párrafo a eliminar.:
    <input type="text" name="n1" value="" size="10"/>
  </form>
  <a href="javascript:eliminar(f1.n1.value)">Click para eliminar párrafo</a>
</body>
```

Ejemplo de eliminación de nodo

7.- Acceso a los atributos HTML

El siguiente paso consiste en la modificación de cualquiera de los atributos de una etiqueta html. Así, podemos cambiar la dirección de destino de un enlace, la ruta a una imagen, etc.

```
<head>
  <script>
    function cambiaimagen() {
      var imagen=document.getElementById('foto');
      imagen.setAttribute('src','campeones.jpg'); }
  </script>
</head>

<body>
  <br><br>
  <h2>Del cielo al ...</h2>
  <br>
  <br>
  <div id="aqui">
    <br>
    <p><a href="javascript:cambiaimagen()">Click para cambiar atributos</a>.</p>
  </div>
</body>
```

8.- Acceso a los atributos CSS

También podemos acceder a los atributos del estilo, para así cambiar el aspecto de cualquier elemento.

```
var imagen = document.getElementById("imagen");  
alert(imagen.style.margin);
```

```

```

```
var parrafo = document.getElementById("parrafo");  
alert(parrafo.style.fontWeight); // muestra "bold"
```

```
<p id="parrafo" style="font-weight: bold;">...</p>
```

9.- Cambiar los atributos CSS

También podemos cambiar los atributos del estilo:

```
<head>
  <script>
    function cambiacolor(micolor) {
      var cuerpo=document.getElementById('texto');
      cuerpo.setAttribute("style", "color:" + micolor);
    }
  </script>
</head>

<body>
  <br><br>
  <h1 style="color:red" id="texto">Cambia de color ...</h1>
  <br>
  <p><a href="javascript:cambiacolor('red')">Rojo.</a></p>
  <p><a href="javascript:cambiacolor('blue')">Azul</a></p>
  <p><a href="javascript:cambiacolor('green')">Verde</a></p>
  <p><a href="javascript:cambiacolor('black')">Negro</a></p>
</body>
```