

Selección de objetos del DOM

Podemos acceder en una primera instancia al DOM mediante los selectores ordinarios:

<code>document.getElementById();</code>	Devuelve un único elemento del DOM que tenga como valor del ID el string indicado como argumento, null si no existe.
<code>document.getElementsByClassName();</code>	Devuelve una colección (Array) de elementos del DOM que tengan alguna de las clases indicadas como argumento, null si no existe. Cada clase se separa por un espacio. El orden en el que son almacenados es el orden en el que aparecen en el DOM.
<code>document.getElementsByTagName();</code>	Devuelve una colección (Array) de elementos del DOM que tengan la etiqueta de HTML indicada como argumento, null si no existe. El orden en el que son almacenados es el orden en el que aparecen en el DOM. Previo a realizar la búsqueda en el DOM, convierte el valor del argumento a lowercase.
<code>document.querySelector();</code>	Devuelve el primer objeto del selector indicado como argumento, null si no existe. Los selectores avanzados CSS3 son válidos. Los pseudo selectores no son válidos, devolviendo un NodeList vacío. Este método es menos eficiente que los tres primeros.
<code>document.querySelectorAll();</code>	Devuelve una objeto NodeList del selector o selectores indicados como argumento, null si no existe. Los selectores avanzados CSS3 son válidos, y se separan entre comas. Los pseudo selectores no son válidos, devolviendo un NodeList vacío. Este método es menos eficiente que los tres primeros.

Propiedades de selección relacional de objetos del DOM

Una vez hemos seleccionado un elemento, podemos acceder a sus hijos de forma relacional mediante la propiedad `children`:

<code>element.children;</code>	Retorna un array con los hijos directos del elemento seleccionado, permite modificaciones haciendo uso del índice
<code>element.firstElementChild;</code>	Retorna el objeto que sea primer hijo directo del elemento seleccionado
<code>element.lastElementChild;</code>	Retorna el objeto que sea último hijo directo del elemento seleccionado

Método de creación e inserción de nodos (etiquetas)

Podemos crear elementos (etiquetas) en el documento:

<code>document.createElement('etiqueta');</code>	Crea un nodo de etiqueta en el documento con un nombre de etiqueta similar al presente como string de su argumento.
<code>document.createTextNode('texto');</code>	Crea un nodo de texto en el documento que puede ser inyectado en el documento o en una etiqueta

Para más adelante inyectarlas de cara al usuario:

<code>element.appendChild(node);</code>	Inserta un elemento en otro elemento como último hijo del mismo.
<code>element.insertBefore(newElement, referenceElement);</code>	Inserta el nuevo elemento (<code>newElement</code>) antes del elemento de referencia (<code>referenceElement</code>) como hijo del nodo actual (<code>element</code>).

Métodos de manipulación de atributos

<code>element.setAttribute('name', 'value');</code>	Agrega un nuevo atributo o cambia el valor de un atributo en un elemento especificado.
<code>element.getAttribute('attrName');</code>	Retorna el valor del atributo pasado como argumento.
<code>element.removeAttribute('attrName');</code>	Elimina el atributo de un elemento especificado.
<code>element.hasAttribute('attName');</code>	Devuelve un valor booleano, siendo true en caso de que el elemento disponga del atributo mencionado, y false en caso de que no exista. Omite en todo caso el valor del atributo.
<code>element.dataset.nombreAtrData;</code>	Retorna o modifica el valor del atributo 'data-' del elemento, undefined si no dispone del atributo solicitado

Propiedades predefinidas del objetos del DOM

<code>element.className</code>	Devuelve el string del valor del atributo class del elemento o lo modifica
<code>element.id</code>	Devuelve el string del valor del atributo id del elemento o lo modifica

Manipulación de clases a través de classList

<code>element.classList.remove("class");</code>	Elimina la clase indicada de las clases existentes en el elemento seleccionado. Acepta indefinidos argumentos separados por comas.
<code>element.classList.add("cl1", "cl2");</code>	Añade la clase indicada a las clases existentes en el elemento seleccionado. Acepta indefinidos argumentos separados por comas.
<code>element.classList.toggle("class");</code>	En caso de que el elemento seleccionado disponga de la clase indicada, la elimina retornando false. De lo contrario, la añade retornando true.
<code>element.classList.contains("class");</code>	Devuelve un valor booleano, siendo true si existe esa clase como uno de los valores del atributo class del elemento seleccionado.

Métodos y propiedades de estilizado del DOM

<code>element.style</code>	<p>Devuelve un objeto de estilo, o actualiza una de las propiedades de estilo del elemento (referenciadas mediante camelCase - la propiedad font-size del element sería element.fontSize).</p> <pre>element.style.marginTop= '20px'; element.style.cssText = 'margin-top: 20px'; element.setAttribute('style', 'margin-top: 20px');</pre>
<code>element.style.cssText</code>	<p>Retorna el texto contenido en el atributo style del elemento seleccionado (estilos en línea)</p>
<code>window.getComputedStyle(elm, pElm);</code>	<p>Devuelve el objeto de estilo actual de un elemento que puede recorrerse a través de un bucle mediante su propiedad item[indice], pudiendo opcionalmente indicar un pseudoelemento del mismo (pElm). Para una propiedad específica, se utiliza en combinación con getPropertyValue() indicando qué propiedad se desea consultar el valor actual.</p> <pre>window.getComputedStyle(elm,null).getPropertyValue ("color"); // es igual que elm.style.color;</pre>
<code>document.styleSheets;</code>	<p>Devuelve un array con todas las hojas de estilo enlazadas a un documento. Es posible seleccionar cualquiera de ellas a través de un índice.</p>
<code>styleSheet.insertRule('regla', index);</code>	<p>Inserta dentro de una hoja de estilos previamente seleccionada una regla como string, en el índice indicado.</p> <pre>sheet.insertRule("header { float: left;", 1);</pre>
<code>styleSheet.deleteRule(index);</code>	<p>Elimina la regla correspondiente al índice indicado de la hoja de estilos previamente seleccionada.</p>