

TEXT CLASS REVIEW

TEMAS QUE TRATAREMOS EN LA CUE

- Obtención y manipulación de valores y textos del DOM
- Selectores básicos
- Eventos básicos

Ya estuvimos trabajando con **JavaScript** previamente, pero aun nos falta ver una faceta fundamental de este interesante lenguaje de programación, nos referimos al manejo del **DOM**, incorporando eventos que den interactividad a los procesos que se realicen en una página web capturando distintos elementos con los selectores existentes para eso.

QUÉ ES EL DOM

El modelo **DOM** (Modelo de Objetos de Dominio o *Document Object Model*) es una **API** para **HTML** y **XML**, la cual entrega una representación estructural de los documentos en forma de árbol de nodos. Cada elemento corresponde a un nodo y cada texto a un nodo de texto.

A través del **DOM** y **JavaScript** se puede acceder a cualquiera de los elementos del árbol de nodos, alterar sus propiedades e invocar sus métodos. Así, es posible acceder a cualquier elemento de una página web, modificarlos, eliminarlos, añadir nuevos elementos, etc.

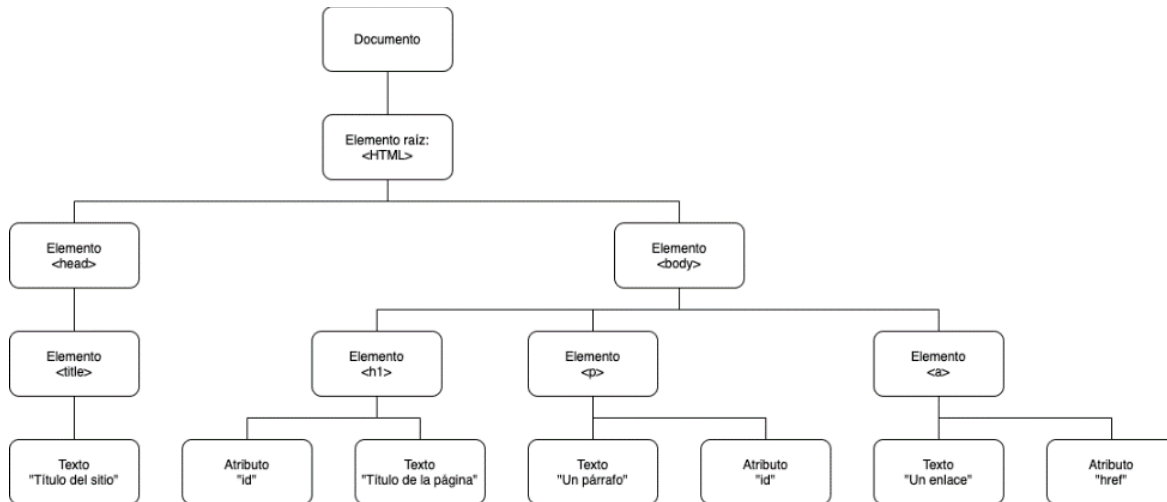


Ilustración 1 Árbol de objetos HTML DOM

Para leer los nodos, vamos a utilizar los elementos **Document** y **Element** en todas sus variantes, pero existen varios tipos de datos importantes que veremos a continuación:

<i>Tipo dato</i>	<i>Descripción</i>
<i>document</i>	Cuando un miembro devuelve un objeto del tipo document (por ejemplo, la propiedad ownerDocument de un elemento devuelve el documento "document" al cual pertenece), este objeto es la raíz del objeto documento en sí mismo.
<i>element</i>	Element se refiere a un elemento o a un nodo de tipo de elemento "element" devuelto por un miembro del API de DOM. Dicho de otra manera, por ejemplo, el método document.createElement() devuelve un objeto referido a un nodo, lo que significa que este método devuelve el elemento que acaba de ser creado en el DOM. Los objetos element ponen en funcionamiento a la interfaz Element del DOM y también a la interfaz de nodo "Node" más básica, las cuales son incluidas en esta referencia.
<i>nodeList</i>	Una "nodeList" es una serie de elementos, parecido a lo que devuelve el método document.getElementsByTagName(). Se accede a los elementos de la nodeList de cualquiera de las siguientes dos formas:

	<ul style="list-style-type: none"> • <code>list.item (1)</code> • <code>list [1]</code> <p>Ambas maneras son equivalentes. En la primera, item () es el método del objeto <code>nodeList</code>. En la última se utiliza la típica sintaxis de acceso a listas para llegar al segundo ítem de la lista.</p>
<i>attribute</i>	<p>Cuando un atributo ("attribute") es devuelto por un miembro (por ej., Por el método createAttribute ()), es una referencia a un objeto que expone una interfaz particular (aunque limitada) a los atributos. Los atributos son nodos en el DOM igual que los elementos.</p>
<i>NamedNodeMap</i>	<p>Un <code>namedNodeMap</code> es una serie, pero los ítems son accesibles tanto por el nombre o por un índice, este último caso es meramente una conveniencia para enumerar ya que no están en ningún orden en particular en la lista. Un <code>NamedNodeMap</code> es un método de ítem () por esa razón, permite poner o quitar ítems en un <code>NamedNodeMap</code></p>

Ilustración 2 Tabla descripción de tipos de datos

Comúnmente, las aplicaciones se ejecutan de manera lineal, esto quiere decir que siempre se ejecutan desde arriba hacia abajo, sin embargo, actualmente predomina el modelo de programación basada en eventos, en el cual los programas no realizan ninguna acción hasta que se produzca un evento y recién ahí ejecutan alguna tarea que se desencadena con aquel evento. Una vez terminado el evento siguen en espera a la ocurrencia de algún nuevo evento.

¿QUÉ ES UN EVENTO?

Un evento es alguna acción que realice el usuario que gatille algún proceso, por ejemplo, apretar un botón que muestre un texto antes oculto. Este se mantendrá oculto hasta que el usuario accione la visualización a través de un evento específico, por ejemplo, haciendo clic sobre el botón y no antes.

TIPOS DE EVENTOS

El nombre de los eventos se construye mediante el prefijo **on**, seguido del nombre en inglés de la acción que se va a realizar, por ejemplo, al hacer clic con el ratón, el evento se denomina **onclick()**.

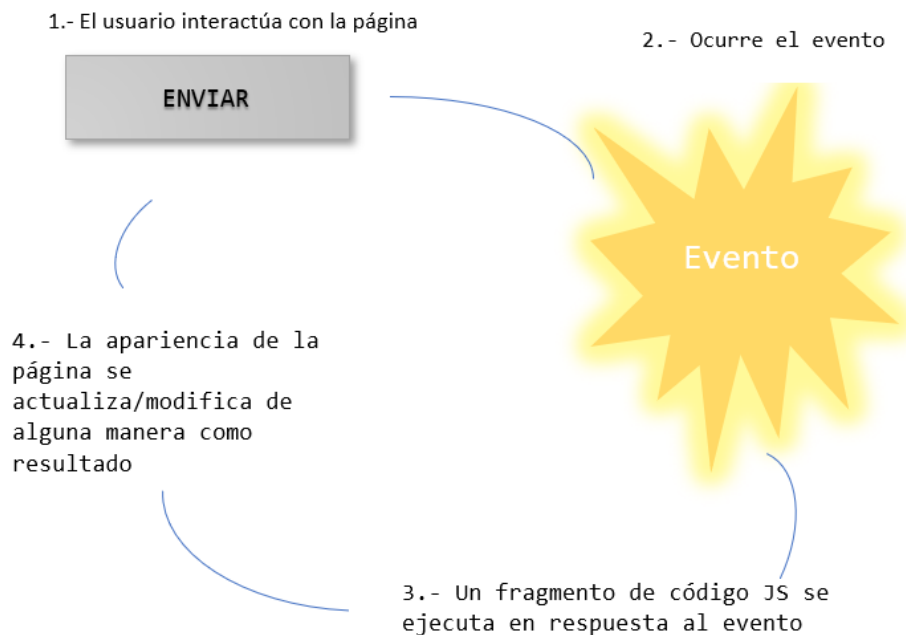


Ilustración 3 Interacción de un evento

Para que un evento realmente sea útil, debe ser asociado a una función o código **JavaScript**. De esta forma, cuando se produce un evento se ejecuta el bloque de instrucciones que hemos indicado, activando aquellos mensajes, cambios de estilo, envíos de formularios, entre otras acciones que se pueden gatillar.

Para leer más información sobre los temas a tratar el día de hoy, podemos ingresar a la documentación [¿Qué es el DOM?](#), [DOM HTML de JavaScript](#) y [Referencia de Eventos](#).