**¿Qué es el DOM?**

Cuando creamos una página web nos limitamos a poner etiquetas HTML que deli- mitan cada uno de los elementos que tendrá la página: cabecera, cuerpo, párrafo, tabla, etc. Sin embargo, el navegador debe convertir ese código en elementos visuales para que un usuario pueda ver su contenido, interpretando cada etiqueta y creando a la vez una jerarquía de objetos que representan los elementos que la componen. Esta jerarquía es el Modelo de Objetos del Documento, más conocido como DOM (Document Object Model).

Esto nos será muy útil como programadores en JavaScript, ya que, gracias al DOM, podremos acceder a cada elemento de la página y obtener o modificar sus propiedades a nuestro antojo.

**Objeto Window:**

Es el más alto dentro de la jerarquía y el encargado de representar la ventana del navegador donde se está visualizando la página. La pestaña de un navegador tendrá su propio objeto window, ya que en definitiva se trata de páginas distintas.

**Colección de objetos:**

Puesto que el DOM está organizado como una jerarquía o árbol de objetos, veremos que algunos de ellos pueden contener a su vez otros objetos que pertenecen al DOM. Alguno de ellos son:

Frames, document, history, location, navigator, screen.

**Propiedades:**

**Name:** fija o devuelve el nombre de la ventana que nos servirá para referirnos a ella en el código.

Self: se refiere a la ventana actual (similar a la sentencia this de los objetos).

Parent: se refiere a la ventana que contiene el frame en el que estamos trabajando. En caso de no haber marcos en la página equivale a la propia ventana (self).

Top: devuelve la ventana que se encuentra en el nivel más alto por encima del frame actual. En caso de tener un único frame en una página coincidirá con la propiedad parent, pero, si hubiese varios frames anidados, el valor devuelto sería el de la ventana que contiene a todos.

Opener: hace referencia a la ventana que abrió la actual (ventana emergente). Si la ventana no fue abierta por otra, obtendremos un valor null.

outerHeight, outerWidth: Devuelve el tamaño en pixels, del espacio de toda la ventana en vertical y horizontal.

innerHeight, innerWidth: Obtiene la altura o anchura, en pixeles del espacio disponible, donde se visualiza propiamente la página. Este valor excluye todas las barras (menú, herramientas…)

screenX, sreenY: Nos revelan la posición de la ventana respecto a la esquina superior izquierda ( punto [0, 0].

scrollX, scrollY: Indican cuántos píxeles ha sido desplazada la página actual dentro de las barras de desplazamiento, en horizontal y vertical respectivamente.

Las siguientes propiedades son a su vez objetos, por lo que tienen algunos métodos disponibles.

localStorage, sessionStorage: Son objetos que nos permiten almacenar información que pueda ser de utilidad para la página durante la visita actual o en la posteriores. Son lo sustitutos naturales de las cookies. La diferencia entre ambas es que los datos de sessionStorage desaparecen al terminar la sesión de la página, mientras que los de localStorage no tiene una fecha de expiración.

* setItem (clase, valor): almacena un valor, como un string, asociándolo a una clave.
* getItem (clave): Permite recuperar un valor accediendo a su clave. Si no hay clave se obtiene null.
* removeItem (clave): elimina el valor asociado a la clave indicada.
* Clear ().

Console: nos da acceso al objeto console. Log, info, debug, warn, error, clear, time, timeEnd.

**Metodos objeto window:**

* Print(): manda imprimir la página que estemos viendo.
* Stop(): detiene la carga de la página.
* Close(): cierra la ventana. Si se utiliza junto con una instancia, cierra dicha ventana.
* Open(URL, nombre, configuración): crea una nueva ventana ( una pop-up normalmente) y devuelve una instancia de window (para poder hacer referencia a ella después) o un valor null.
* URL: la url de la pag que se cargará en la ventana.
* Nombre: asocia un nombre a la ventana para poder identificarla.
* Configuración: cadena formada por pares “propiedad = valor” separados por comas, pa definir aspectos de la ventana.
* moveBy (x, y): desplaza la ventana el numero de pixeles indicados horizontal y verticalmente.
* moveTo(x, y): mueve la ventana a una posición concreta.
* resizeBy(x,y): redimensiona la ventana tantos píxeles como indiquen los parámetros.
* resizeTo(x,y): establece una altura y anchura concreta a la ventana.
* scrolBy(x,y): realia un desplazamiento horizontal y/o vertical de tantos píxeles ( positivos o negativos) como marquen los parámetros.
* scrollTo(x,y): realiza un desplazamiento horizontal y/o vertical hasta una posición concreta.