La interfaz Document representa cualquier página web cargada en el navegador y sirve como punto de entrada al contenido de la página web, que es el árbol DOM. El árbol DOM incluye elementos como <body> y <table>, entre muchos otros. Proporciona funcionalidad globalmente al documento, como obtener la URL de la página y crear nuevos elementos en el documento.

Existen varios métodos de Document que servirán para acceder a los elementos del DOM:

1. querySelector: Es un método de Document que regresa el primer elemento dentro del documento que coincide con el selector o grupo de selectores especificado. Si no se encuentran coincidencias, regresa null.
2. getElementsByClassName: regresa un objeto similar a una matriz con todos los elementos secundarios que tienen todos los nombres de clase dados.
3. getElementById: Devuelve una referencia al elemento por su id.
4. document.createElement: crea el elemento HTML especificado por la etiqueta, o un HTMLUnknownElement si no se reconoce la etiqueta.

Y algunos métodos de la interfaz Element:

1. elemento.innerHTML: La propiedad Element.innerHTML devuelve o establece la sintaxis HTML describiendo los descendientes del elemento. Al establecerse se reemplaza la sintaxis HTML del elemento por la nueva. (Permite modificar el contenido html del elemento, lo que puede ser peligroso, porque literalmente, se está modificando el HTML)
2. elemento.innerText: representa el contenido de texto renderizado de un nodo y sus descendientes. Como getter, se aproxima al texto que el usuario obtendría si resaltara el contenido del elemento con el cursor y luego lo copiara en el portapapeles. Como setter, esto reemplazará a los hijos del elemento con el valor dado, convirtiendo cualquier salto de línea en elementos <br>.
3. elemento.getAttribute: devuelve el valor del atributo especificado en el elemento. Si el atributo especificado no existe, el valor retornado puede ser tanto null como “”
4. elemento.setAttribute: Establece el valor de un atributo en el elemento indicado. Si el atributo ya existe, el valor es actualizado, en caso contrario, el nuevo atributo es añadido con el nombre y valor indicado.
5. elemento.rreplaceChldren: eemplaza los elementos secundarios existentes de un nodo con un nuevo conjunto específico de elementos secundarios. Estos pueden ser objetos de cadena o de nodo.
6. elemento.append: inserta un conjunto de objetos de nodo u objetos de cadena después del último elemento secundario del elemento. Los objetos de cadena se insertan como nodos de texto equivalentes.

Son los métodos de las interfaces mencionadas hasta el momento. Dale corazón para que este aporte suba y les ayude a más personas.

Para realizar una conversión de string a int (coerción explicita) se pueden usar los métodos number(), parseInt(), pero también sirve si le agregas un + a cada variable para que la conversión se realice automáticamente.

**addEventListener:**

Este metodo sirve para escuchar cualquier tipo de evento que ocurra dentro de un objeto, estos objetos pueden ser un elemento HTML, una ventana, el mismo documento, un XMLhttpRequest.

Los eventos pueden ser:

* **blur:** Cuando el elemento pierde el foco.
* **click:** El usuario hace clic sobre el elemento.
* **dblclick:** El usuario hace doble clic sobre el elemento.
* **focus:** El elemento gana el foco.
* **keydown:** El usuario presiona una tecla.
* **keypress:** El usuario presiona una tecla y la mantiene pulsada.
* **keyup:** El usuario libera la tecla.
* **load:** El documento termina su carga.
* **mousedown:** El usuario presiona el botón del ratón en un elemento.
* **mousemove:** El usuario mueve el puntero del ratón sobre un elemento.
* **mouseout:** El usuario mueve el puntero fuera de un elemento.
* **mouseover:** El usuario mantiene el puntero sobre un elemento.
* **mouseup:** El usuario libera el botón pulsado del ratón sobre un elemento.
* **unload:** El documento se descarga, bien porque se cierra la ventana, bien porque se navega a otra página.

**Syntaxis**

element.addEventListener(evento, funcion);

onclick: Pinchar y soltar el ratón => Todos los elementos  
ondblclick: Pinchar dos veces seguidas con el ratón => Todos los elementos  
onfocus: Seleccionar un elemento => <button>, <input>, <label>,<select>, <textarea>, <body>  
onkeydown: Pulsar una tecla y no soltarla => Elementos de formulario y <body>  
onkeypress: Pulsar una tecla => => Elementos de formulario y <body>  
onkeyup: Soltar una tecla pulsada => Elementos de formulario y <body>  
onload: Página cargada completamente => <body>  
onmousedown: Pulsar un botón del ratón y no soltarlo => Todos los elementos  
onmousemove: Mover el ratón => Todos los elementos  
onmouseout: El ratón “sale” del elemento => Todos los elementos  
onmouseover:El ratón “entra” en el elemento => Todos los elementos  
onmouseup: Soltar el botón del ratón => Todos los elementos  
onreset :Inicializar el formulario => <form>  
onresize :Modificar el tamaño de la ventana => <body>  
onselect: Seleccionar un texto =><input>, <textarea>  
onsubmit :Enviar el formulario  
onunload: Se abandona la página, por ejemplo al cerrar el navegador =>

Evento : | Descripción | Elementos para los que está definido  
onblur: Deseleccionar el elemento => <button>, <input>, <label>,<select>, <textarea>, <body>  
onchange: Deseleccionar un elemento que se ha modificado => <input>, <select>, <textarea>