

# Introducción JavaScript

1. Crea una tabla describiendo el propósito de las funciones javascript incluidas en la última transparencia del documento de introducción a javascript
2. Realiza un esquema del árbol DOM que genera el [código HTML en el fichero enlazado](#)
3. Define una función en javascript en la que se pueda calcular el cambio entre diferentes monedas (al menos 3 diferentes).
4. Define una función javascript que calcule el importe de la compra de productos, con el IGIC aplicado. (Los productos estarán almacenados en un json)
5. Crear un script que muestre un informe de la cantidad de párrafos, enlaces y elementos ul en tu página de inicio.

NOTAS: La 1º y la 2º pregunta no hay que escribir ningún código es sólo explicarlo en un documento de texto. En cambio las 3º, 4º y 5º es con javascript

## 1ª pregunta

URLs:

[https://www.w3schools.com/jsref/jsref\\_reference.asp](https://www.w3schools.com/jsref/jsref_reference.asp) → Referencia javascript  
[https://www.w3schools.com/jsref/dom\\_obj\\_document.asp](https://www.w3schools.com/jsref/dom_obj_document.asp) → “document”  
[https://www.w3schools.com/jsref/dom\\_obj\\_all.asp](https://www.w3schools.com/jsref/dom_obj_all.asp) → “element”  
[https://www.w3schools.com/jsref/prop\\_node\\_attributes.asp](https://www.w3schools.com/jsref/prop_node_attributes.asp) → attributes del elemento  
[https://www.w3schools.com/jsref/prop\\_html\\_style.asp](https://www.w3schools.com/jsref/prop_html_style.asp) → style  
[https://www.w3schools.com/jsref/met\\_element\\_setattribute.asp](https://www.w3schools.com/jsref/met_element_setattribute.asp) → setAttribute  
[https://www.w3schools.com/jsref/met\\_document\\_createelement.asp](https://www.w3schools.com/jsref/met_document_createelement.asp) → createElement  
[https://www.w3schools.com/jsref/met\\_node\\_removechild.asp](https://www.w3schools.com/jsref/met_node_removechild.asp) → removeChild  
[https://www.w3schools.com/jsref/met\\_node\\_appendchild.asp](https://www.w3schools.com/jsref/met_node_appendchild.asp) → appendChild  
[https://www.w3schools.com/jsref/met\\_node\\_replacechild.asp](https://www.w3schools.com/jsref/met_node_replacechild.asp) → replaceChild  
[https://www.w3schools.com/jsref/met\\_doc\\_write.asp](https://www.w3schools.com/jsref/met_doc_write.asp) → write  
[https://www.w3schools.com/jsref/met\\_document\\_getelementbyid.asp](https://www.w3schools.com/jsref/met_document_getelementbyid.asp) → getElementById

Se trata de una tabla con los métodos del “objeto” document, cuando se carga una página web HTML en el navegador, se convierte en un objeto de tipo “document” (documento). Algunos métodos son:

Métodos Javascript para el árbol DOM	Descripción
document.getElementById(id)	Devuelve el elemento (por ejemplo una etiqueta párrafo “p”) al que se le haya asignado como atributo el identificador “id”, luego podremos por ejemplo guardarlo en una variable para realizarle cambios
document.getElementsByTagName(name)	Devuelve una lista de nodos (NodeList) conteniendo todos los elementos que coincidan con el nombre de la etiqueta HTML pasada por argumento entre paréntesis

<code>document.getElementsByClassName(name)</code>	Devuelve una <code>NodeList</code> con todos los elementos con el nombre de etiqueta entre paréntesis, es decir, todas las etiquetas HTML cuyo atributo “class” tengan el mismo nombre que el que hay dentro del paréntesis
<code>element.innerHTML = new html content</code>	Añade al elemento “element” de la página HTML ( <code>innerHTML</code> ) un contenido nuevo, puede ser un elemento HTML obtenido con un método de tipo “ <code>getElement...</code> ”
<code>element.attribute = new value</code>	Cambia el valor del atributo del elemento actual (Elemento/Nodo del árbol DOM) a uno nuevo
<code>element.style.property = new style</code>	Da un nuevo valor ( <code>new style</code> ) a la propiedad ( <code>property</code> ) CSS del estilo del elemento a modificar. ‘property’ va desde el color de fondo del estilo del elemento ( <code>backgroundColor</code> ), al alineado, etc. son propiedades de la hoja de estilos CSS
<code>element.setAttribute(attribute, value)</code>	Establece para el elemento HTML ‘element’ un atributo nuevo de estilo ( <code>attribute</code> ) con el valor ( <code>value</code> ) que puede ser desde un identificador de un estilo concreto hecho por nosotros, a un valor para cambiar el elemento de un formulario de un tipo a otro (de una caja de texto a un botón por ejemplo)
<code>document.createElement(element)</code>	Crea un elemento y un nodo de dicho elemento o nombre del nodo en el árbol DOM HTML. El elemento a añadir ( <code>element</code> ) suele ser un valor de etiqueta HTML (‘p’ para añadir un párrafo, ‘button’ para añadir un botón, etc.). NOTA: Así tal cual no se añadirá el elemento al documento sino que necesitamos el método “ <code>appendChild</code> ” para que se haga efectivo
<code>document.removeChild(element)</code> NOTA: El método está en realidad en ‘ <code>element.removeChild(element)</code> ’	Elimina el elemento especificado como nodo en el árbol DOM HTML y del documento. Al usar éste método se devuelve como argumento el elemento eliminado si existía previamente o ‘NULL’ si no existía.
<code>document.appendChild(element)</code> NOTA: El método está en realidad en ‘ <code>element.appendChild(element)</code> ’	Se usa sobre todo para añadir texto a un elemento añadido con ‘ <code>document.createElement(element)</code> ’, como por ejemplo el texto que aparecerá dentro de un botón o el texto que debe aparecer en un elemento nuevo de lista añadido con el ‘ <code>createElement</code> ’
<code>document.replaceChild(new, old)</code> NOTA: Este método está en ‘ <code>element.replaceChild(newnode, oldnode)</code> ’	Reemplaza el nodo hijo de un nodo del árbol DOM HTML, el cual es el primer argumento, por un nuevo nodo hijo exactamente en la misma posición del viejo en el árbol DOM HTML, el cual es el segundo argumento.

<code>document.write(text)</code>	<p>Escribe el texto pasado como argumento (entre comillas dobles) en el documento en la parte en la que esté escrito este método javascript. Si incluimos entre las comillas dobles etiquetas como las de párrafo o demás se añadirán su función al documento también</p>
<code>document.getElementById(id).onclick = function(){code}</code>	<p>Al clicar sobre el elemento (un botón, por ejemplo) se obtiene el identificador de dicho elemento y se ejecuta el código de la función</p>

## 2º Pregunta

URLs:

[https://developer.mozilla.org/es/docs/Referencia\\_DOM\\_de\\_Gecko/Introducci%C3%B3n](https://developer.mozilla.org/es/docs/Referencia_DOM_de_Gecko/Introducci%C3%B3n)

<https://www.w3.org/DOM/>

