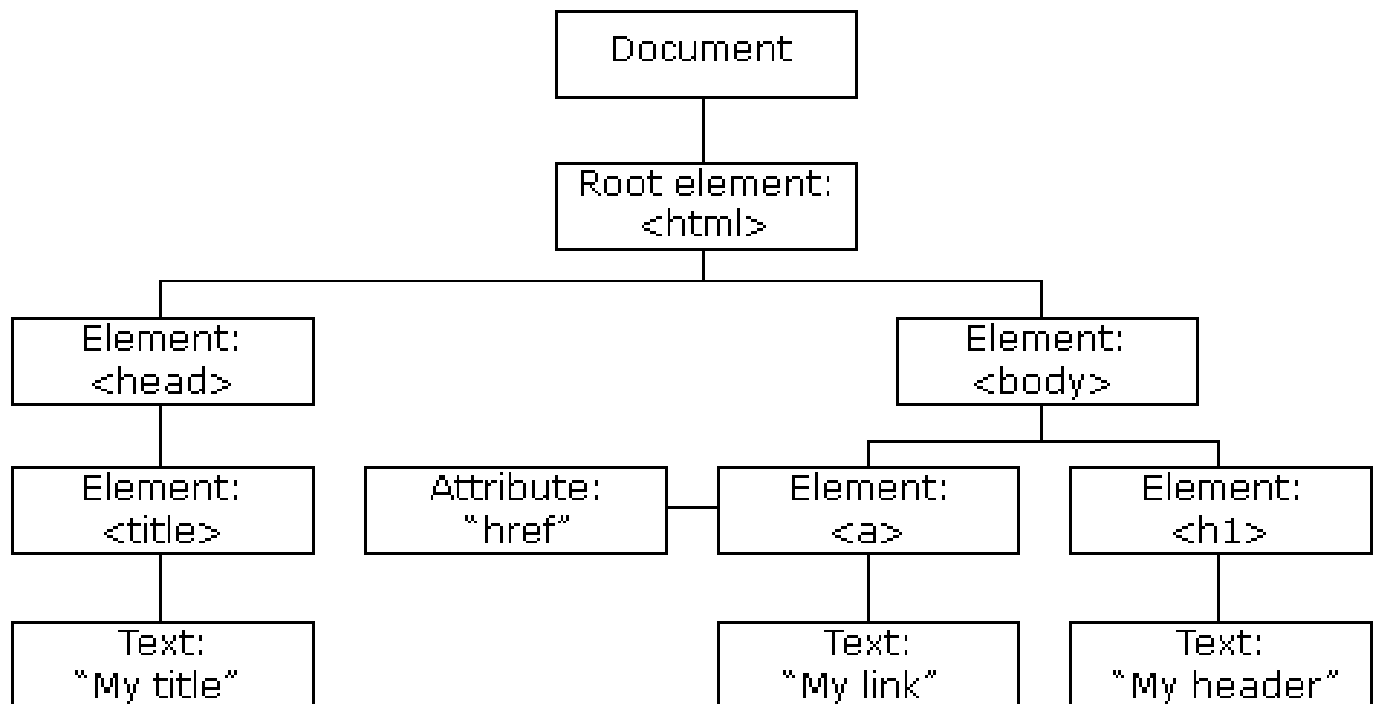


## Estructura del árbol DOM:



Para ordenar la estructura del árbol, existe una serie de reglas:

- En el árbol de nodos, al nodo superior (document), se le llama raíz.
- Cada nodo, exceptuando el nodo raíz, tiene un padre.
- Un nodo puede tener cualquier número de hijos.
- Una hoja es un nodo sin hijos.
- Los nodos que comparten el mismo padre, son hermanos.

## Tipos de nodos

La especificación completa de DOM define 12 tipos de nodos, aunque las páginas XHTML habituales se pueden manipular manejando solamente cuatro o cinco tipos de nodos:

- **Document**, nodo raíz del que derivan todos los demás nodos del árbol.
- **Element**, representa cada una de las etiquetas XHTML. Se trata del único nodo que puede contener atributos y el único del que pueden derivar otros nodos.
- **Attr**, se define un nodo de este tipo para representar cada uno de los atributos de las etiquetas XHTML, es decir, uno por cada par **atributo=valor**.
- **Text**, nodo que contiene el texto encerrado por una etiqueta XHTML.
- **Comment**, representa los comentarios incluidos en la página XHTML.

Los otros tipos de nodos existentes que no se van a considerar

son `DocumentType`, `CDataSection`, `DocumentFragment`, `Entity`, `EntityReference`, `ProcessingInstruction` y `Notation`.

## Constantes del objeto Node:

Tipo de nodo=Valor	
<code>Node.ELEMENT_NODE = 1</code>	<code>Node.PROCESSING_INSTRUCTION_NODE = 7</code>
<code>Node.ATTRIBUTE_NODE = 2</code>	<code>Node.COMMENT_NODE = 8</code>
<code>Node.TEXT_NODE = 3</code>	<code>Node.DOCUMENT_NODE = 9</code>
<code>Node.CDATA_SECTION_NODE = 4</code>	<code>Node.DOCUMENT_TYPE_NODE = 10</code>
<code>Node.ENTITY_REFERENCE_NODE = 5</code>	<code>Node.DOCUMENT_FRAGMENT_NODE = 11</code>
<code>Node.ENTITY_NODE = 6</code>	<code>Node.NOTATION_NODE = 12</code>

- `nodeType` y `nodeName`

`nodeType` y `nodeName` sirven respectivamente para comprobar el tipo de un nodo y el nombre de un nodo. `nodeType` devuelve un número, que corresponde a un tipo de nodo. La tabla siguiente muestra los tipos posibles, y su número:

Número	Tipo de nodo
1	Nodo elemento
2	Nodo atributo
3	Nodo texto
4	Nodo 4 para pasar CDATA (en relación con XML)
5	Nodo para referencia de la entidad
6	Nodo para entidad
7	Nodo para instrucción de tratamiento
8	Nodos para comentarios
9	Nodo documento
10	Nodo tipo de document
11	Nodo de fragmento de documento
12	Nodo para notación

`nodeName`, por su parte, devuelve el nombre del elemento en mayúsculas. Es aconsejable utilizar `toLowerCase()` (o `toUpperCase()`) para forzar un formato de letra y evitar sorpresas desagradables.

## Métodos y propiedades de Node:

Propiedad / Método	
nodeName	previousSibling
nodeValue	nextSibling
nodeType	hasChildNodes()
ownerDocument	attributes
firstChild	appendChild(nodo)
lastChild	removeChild(nodo)
childNodes	replaceChild(nuevoNodo, anteriorNodo)
parentNode	insertBefore(nuevoNodo, anteriorNodo)

Cada nodo tiene una serie de propiedades que reflejan el parentesco con otros, algunas de las cuales son:

- **childNodes**: array con los nodos hijos
- **firstChild**: primer nodo hijo, **lastChild**: último nodo hijo
- **parentNode**: nodo padre
- **nextSibling**: siguiente hermano (nodo al mismo nivel) **prevSibling**: hermano anterior.