

Propiedades básicas de los nodos del DOM

nodeType

- Un nodo del DOM es un objeto con propiedades como sí mismo y su contenido. Algunas de las propiedades son de solo lectura y algunas pueden ser actualizadas al vuelo
- Hay 12 tipos de nodos, descritos en el [DOM Level 1](#)
- El tipo 1 y el 3 son los más importantes

- Podemos listar todos los nodos que son de tipo ELEMENT_NODE (1), para ello iteramos con childNodes y usaremos `childNodes[i].nodeType != 1`

```
<body>
  <div>Allowed readers:</div>
  <ul> <li>John</li> <li>Bob</li> </ul>
  <script>
    var childNodes = document.body.childNodes
    for(var i=0; i<childNodes.length; i++) {
      if (childNodes[i].nodeType != 1) continue
      alert(childNodes[i])
    }
  </script>
</body>
```

- ¿Qué muestra este alert?

```
<!DOCTYPE HTML>
<html>
<body>
  <script>
    alert(document.body.lastChild.nodeType)
  </script>
</body>
</html>
```

- Explicación: en el momento

de la ejecución del script, el último nodo es el nodo script, el resultado es un 1

nodeName y tagName

- Ambas contienen el nombre del nodo
- Por ejemplo, ¿para el body?
- De hecho, la propiedad tagName está undefined en muchos tipos de nodo. Por tanto, tagName es menos informativa que nodeName

innerHTML

- The innerHTML es una parte del estándar HTML5
- Permite acceder a los contenidos del nodo, a su texto
- Es modificable al vuelo
- Funciona para cualquier elemento nodo, es muy útil

```
<body>
  <p>The paragraph</p>
  <div>And a div</div>
  <script>
    alert( document.body.innerHTML ) // leemos el contenido
    document.body.innerHTML = 'ies.pmo!' // reemplaza contenido
  </script>
</body>
```

Obstáculos con innerHTML

- No es tan simple como parece. Hay algunos obstáculos si no se han utilizado antes:
 - En IE, innerHTML es read-only para todos los tags relacionados con tablas excepto TD, por tanto (col, colgroup, framset, head, html, style, table, tbody, tfoot, thead, title y tr)
 - La propiedad innerHTML NO SE PUEDE AMPLIAR
- ```
elemento.innerHTML += "Hola";
elemento.innerHTML += "mundo";
```



# nodeValue

- innerHTML funciona solo para elementos de tipo nodo
- Para otros tipos de nodo, esta propiedad es la que contiene la información
- Por ejemplo, vamos a ver un ejemplo en el que lo usamos para nodos tipo texto y nodos tipo comentario

```
<body>
 The text
 <!-- A comment -->
 <script>
 for(var i=0; i<document.body.childNodes.length; i++) {
 alert(document.body.childNodes[i].nodeValue)
 }
 </script>
</body>
```

- En este ejemplo, aparecen pocas alerts vacías, para `<script>` es null, porque `<script>` es un elemento tipo nodo y para estos se usará `innerHTML`

```
for(var i=0; i<document.body.childNodes.length; i++) {
 alert("El elemento de tipo " +
 document.body.childNodes[i].nodeType +
 " y de nodeName " +
 document.body.childNodes[i].nodeName +
 " tiene el valor ->" +
 document.body.childNodes[i].nodeValue +
 "<-");
}
```

# Resumen

- `nodeType`: tipo del nodo, los típicos son 1 y 3
- `nodeName/tagName`: el nombre del tag
- `innerHTML`: contenido de un elemento del nodo. Writeable
- `nodeValue`: contenido de un nodo texto. También es Writeable

# Propiedades básicas de los nodos del DOM