

DOM

Tema 6-Continuación

DOM

Atributos

- La propiedad `attributes` permite acceder a los atributos de un elemento.

`elemento.attributes`

- Devuelve un dato de tipo `namedNodeMap`.
- Un dato de tipo `namedNodeMap` es una colección a la que se puede acceder tanto por el índice de cada elemento, como por el nombre del atributo.
- DOM no determina el orden de los elementos.
- Cada elemento de la colección es un nodo de tipo `Attr`.
- La propiedad `name` del nodo devuelve el nombre del atributo.
- La propiedad `value` del nodo devuelve el valor del atributo
- [ejemplo](#)

DOM

Atributos

- Los elementos html de DOM, tienen propiedades para cada uno de los atributos definidos en los elementos html.
- Se accede mediante *elemento.nombreAtributo*.
- Los nombre de los atributos son los mismos que en html, con la excepción del atributo class que es className.
- Devuelve el valor del atributo *nombreAtributo de elemento*.
- Se puede utilizar para establecer el valor de un atributo.
- Para modificar el valor de un atributo, no se debe utilizar la colección attributes.

Crear y eliminar atributos

El método `setAttribute` permite crear un nuevo nodo de atributo en un elemento DOM.

`nodo.setAttribute(nombreAtributo, valorAtributo)`

- Crea o establece el valor de un atributo del elemento *nodo*.
 - ✓ Si *nombreAtributo* ya existe modifica su valor.
 - ✓ Si *nombreAtributo* no existe, crea un nuevo atributo.
- No devuelve nada.

Crear y eliminar atributos

- ❑ El método `removeAttribute` elimina un atributo de un elemento.
`nodo.removeAttribute(nombreAtributo)`
 - Elimina `nombreAtributo` de `nodo`.
 - Si el atributo no existe genera una excepción.
 - ✓ Se puede utilizar el método `nodo.hasAttribute(nombreAtributo)` para determinar si un elemento tiene un atributo.

Crear y eliminar atributos

- ❑ El método `getAttribute` permite obtener el valor de un atributo.

`nodo.getAttribute(nombreAtributo)`

- Devuelve una cadena con el valor del atributo.
- Si el atributo no existe, devuelve una cadena nula o el valor nulo.

Estilos

- La propiedad style de los elementos permite acceder a los distintos estilos **en línea de un nodo**.
- Devuelve un objeto de tipo style que representa el conjunto de las propiedades de estilos en línea establecidas para el elemento mediante el atributo html style.
- Como los estilos en línea tienen mayor prioridad que el resto dentro de CSS, permite cambiar el estilo de los elementos

Estilos

- El objeto style devuelto permite acceder a todos los atributos de estilo definidos en CSS.

nodo.style.nombreAtributoCSS

El nombre de los atributos varía ligeramente respecto a CSS.

- En los atributos CSS con guiones, los guiones desaparecen.
- Cuando hay palabras compuestas, la primera va en minúsculas y las siguientes en mayúsculas.
- Por ejemplo, la propiedad color de CSS llamaría igual que la propiedad color del objeto style de DOM, la propiedad background-color de CSS se llamaría backgroundColor en DOM o la propiedad border-top-color de CSS se llamaría borderTopColor en DOM.
- La especificación de DOM del W3C da una correspondencia de todas las propiedades CSS con DOM en www.w3.org/TR/DOMLevel2/css CSS2Properties

Estilos

- El objeto style es de sólo lectura por lo que no se puede modificar directamente.
- Para modificar un estilo en línea hay que utilizar las propiedades del objeto style (los nombre de los atributos del estilo) que son de lectura escritura.

```
Var nodo=document.getElementById("ultimoParrafo");  
nodo.style.color = "blue"
```

Estilos

- Si se desea cambiar varios atributos de estilo...
- Crear una regla para una clase con todos los atributos que se desea modificar.
- Cambiar el atributo class del elemento mediante *nodo.className*.
- `<style type="text/css">`
- `.recuadro {padding: 0.5em; background-color:#EBEBEB; border: #d6d6d6`
- `solid 1px;}`
- `</style>`
- `function enmarcarPárrafo(){`
- `var nodo = document.getElementById("primerparrafo");`
- `if(nodo.className=="normal"){`
- `nodo.className="recuadro"; }`
- `else{`
- `nodo.className="normal"; }`