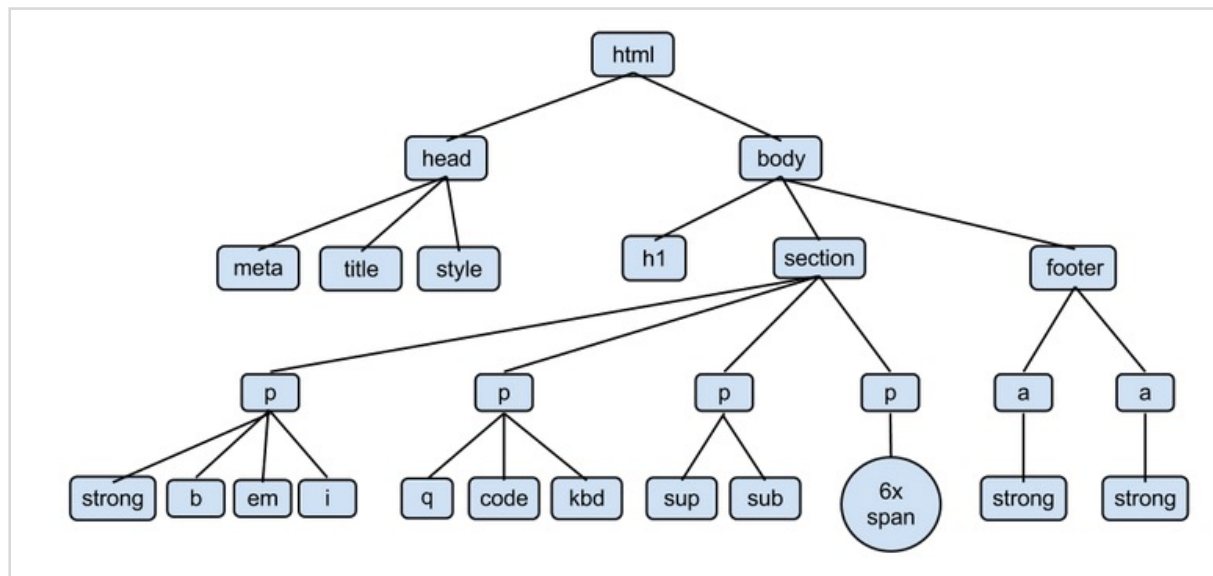


DOM introducción y selectores

#salesians/js



Paso a paso

1. Crea un archivo html
2. Añade algo de código:

```
<!DOCTYPE>
<html>
<head>
  <title>Introduction to the DOM</title>
</head>
```

```
<body>

</body>
</html>
```

3. Añade algún div dentro del body

```
<h1>Dog</h1>
<div id="cat"></div>
<div class="mouse"></div>
<div class="mouse"></div>
<div class="mouse"></div>
```

4. Ahora abre el archivo html en el navegador. Comprueba el DOM en el inspector de elementos

Accediendo al dom con javascript

Podemos acceder al DOM de toda web a través del objeto document. Como podemos ver, document devuelve todo el código de nuestra web.

Buscando elementos por su id

```
var element = document.getElementById('id')
```

Añadiendo contenido a un elemento html

Una vez tenemos al elemento seleccionado, podemos alterar sus propiedades. Cualquier cambio que realizamos con javascript, se verá reflejado inmediatamente en nuestro html.

- innerHTML

Para cambiar el contenido html de un elemento, podemos usar esta propiedad:

```
var element = document.getElementById("cat");
=> undefined

element.innerHTML = "I'm a cat";
```

¿Y si queremos poner el mismo contenido en otro elemento?

- style

Otro método bastante útil es el de style. Con él, podemos establecer el estilo css que queramos.

```
var element = document.getElementById("cat");  
=> undefined  
  
element.style.backgroundColor = "red";  
element.style.border = "2px green solid";
```

Fijaos que la propiedad no se escribe exactamente igual que en css

Ejercicio

Vamos a intentar generar algo con sentido:

1. Comenta lo realizado hasta ahora.
2. Crea un div y dale un id
3. Dentro un h1 con otro id y escribe algún titular
4. Al body dale una imagen de fondo, y una altura y anchura de todo el viewport
5. Con css también, haz que el div ocupe el 100% en ancho y alto, alinea el texto en el centro y dale un color al texto.
6. Sobre el h1, coloca el texto en el centro verticalmente, dale un font-size bastante grande y elige una fuente.
7. Ahora nos vamos a javascript
8. Crea una función que genere un color aleatorio y lo devuelva en hexadecimal y con la almohadilla:

```
'#'+Math.floor(Math.random()*16777215).toString(16);
```
9. Crea una función que cambie el color de fondo del div. Tendrás que seleccionar el elemento que necesites, y cambiarle el background-color usando la función creada en el punto 8.
10. Crea una función que cambie el contenido del h1. Selecciona el elemento necesario, y con el método innerHTML, genera 3 o 4 casos distintos de contenido.
11. Crea una función que llame a las dos anteriores.
12. Ejecuta a la función del punto 11 con un setInterval.

Atributos y contenido

Obteniendo y añadiendo clases

```
// devuelve la clase de un elemento
var cName = element.className;

// pone la clase obtenida a otro elemento
otherElement.className = cName;
```

Obteniendo y añadiendo ids

```
// devuelve el id
var idStr = element.id;

// añade el id a otro elemento
otherElement.id = "some-value";
```

Seleccionando elementos

Ya hemos visto como seleccionar elementos por id, pero también podemos hacerlo por clase:

```
var elements = document.getElementsByClassName(names);

var mice = document.getElementsByClassName('mouse');
```

- elements es una variable que guarda todos los elementos html que encajen con la selección.
- names es la clase por la que estamos buscando
- getElementByClassName es un método que puede ser llamado sobre cualquier elemento.

Lo que devuelve este método no es en sí un array, por lo que no podremos usar los métodos específicos de un array: forEach, map, push, etc. Para iterar sobre ellos, debemos usar un for.

Otra opción es conseguir organizar lo que nos devuelve este método en un array con el siguiente código: `var array = [].slice.call(htmlCollection)`

Ejercicio

1. Genera en html, unos cuantos divs (12), y todos con la misma clase.
2. Dale estas propiedades css a los divs para que se organicen:

```
div {  
  display: inline-block;  
  box-sizing: border-box;  
  margin: 1vh;  
  width: 22vw;  
  height: 30vh;  
  background-color: #fff;  
}
```

4. Genera un función que devuelva un color aleatorio en hexadecimal.
5. Genera una función que seleccione a todos los divs, itere sobre ellos y a cada uno le añada un color distinto.
6. Llama a la función del punto 5 cada 500ms.

Accediendo por tag

```
var divs = document.getElementsByTagName('div');
```

Accediendo al primer elemento encontrado

```
var firstMouse = document.querySelector('.mouse');
```

Accediendo a todos los encontrados

```
var matches = document.querySelectorAll(".mouse, #cat");
```

Recursos

[Element - Web APIs | MDN](#)

[Element - Web APIs | MDN](#)

