

Tema 5c.

Mapas

1. Descripción

- Cuando tenemos que trabajar con Arrays en JavaScript, hay varias opciones disponibles con los cuales podemos iterar, transformar o manipular nuestros arrays.
- Una opción es Map().
- El objeto Map almacena pares clave/valor. Cualquier valor (tanto objetos como valores primitivos) pueden ser usados como clave o valor.

1. Descripción

- Un objeto Map puede iterar sobre sus elementos en orden de inserción. Un bucle `for..of` devolverá un array de [clave, valor] en cada iteración.
- Cabe destacar que un Map el cual es un mapa de un objeto, especialmente un diccionario de diccionarios, solo se va a mapear en el orden de inserción del objeto — el cual es aleatorio y no ordenado.

2. Definición de map

- Podemos definir una variable de tipo mapa con new Map().
- Es posible inicializar el mapa en la declaración

```
var cp=new Map();  
var cp2=new Map([["30","Murcia"],["08","Madrid"]]);
```

3. Insertar tuplas en el mapa

- Se insertan con el método **.set** acompañado de la tupla

```
cp.set("30", "Murcia");  
cp.set("07", "Almeria");  
cp.set("08", "Madrid");
```

4. Obtener el valor de una clave

- El método **get** devuelve el valor asociado a la llave, o “undefined” si no tiene ninguno.

```
mapa.set("DIWEB", "Diseño de interfaces web");  
var nombre=mapa.get("DIWEB");  
document.write(nombre);
```

5. Mostrar las claves

- Para ello debemos utilizar el método `.keys()`

```
for (var clave of mapa.keys()) {  
    document.write(clave+", ");  
}
```

6. Mostrar los valores

- Para ello debemos utilizar el método `.values()`

```
for (var valor of mapa.values()) {  
    document.write(valor+", ");  
}
```


7. Mostrar la tupla

- Se realiza mediante for ..of del mapa (sin especificar .keys o .values)

```
for (var tupla of mapa) {  
    document.write(tupla+"<br>");  
}
```

8. Consultar si pertenece al mapa

- **has** comprueba que un valor, o el resultado de una expresión, se encuentra en el Map. Devuelve un valor true o false.

```
document.write(mapa.has("LM"));
```

9. Otros métodos

- **clear**: elimina todos los elementos de un conjunto.
- **entries**: devuelve un objeto de tipo iterable que contiene a su vez un array con los valores de cada elemento del Set.