Tabla de Contenidos


- JavaScript Map ES6 y operaciones core
  - JavaScript Map y bucles
  - JavaScript Template String y mapas
- Otros artículos relacionados:



**Webinar JavaScript Console y sus trucos**

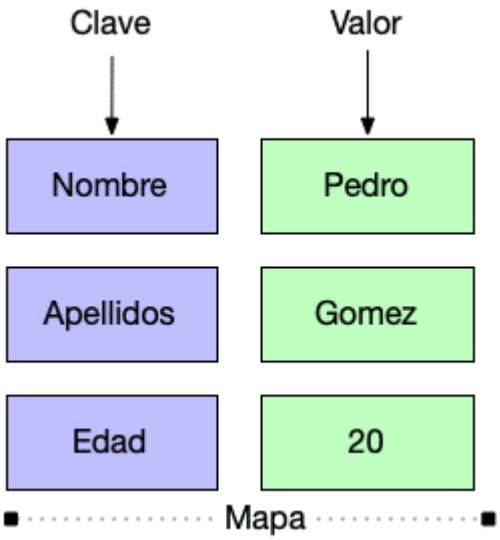
**GRATIS**

**APUNTATE!!**


El uso de JavaScript Map ES6 , es cada día más cotidiano cuando estamos trabajando con ES6. Los Diccionarios o Mapas son estructuras indispensables en todos los lenguajes de programación y hasta la llegada de ES6 Javascript no tenía soporte para ellos y había que crear estructuras de objetos para simularlos. Ese tiempo ya termino. Vamos a ver como usar los nuevos Mapas utilizando Node.js y construyendo varios ejemplos sencillos.

# JavaScript Map ES6 y operaciones core

Lo primero que tendremos que haces es crear un Mapa y añadirle una serie de elementos :




Esto es sumamente sencillo:

```
let mapa= new Map();
mapa.set("nombre","pedro");
mapa.set("apellidos","gomez");
mapa.set("edad",20);
console.log(mapa);
```

Podemos ver el resultado en la consola:

```
cecilioalvarezcaules@MacBook-Pro-de-cecilio-2 JavaScript % node mapas.js
Map { 'nombre' => 'pedro', 'apellidos' => 'gomez', 'edad' => 20 }
cecilioalvarezcaules@MacBook-Pro-de-cecilio-2 JavaScript %
```

Mapas y más operaciones

Otras de las operaciones más comunes son :

- get: Este método nos permitirá obtener un valor del mapa
- has: Comprueba si un valor este dentro del mapa
- clear : elimina todos los items del mapa

## JavaScript Map y bucles

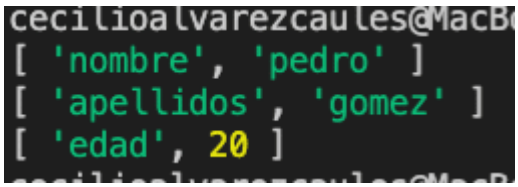
En muchas ocasiones nos encontramos con la necesidad de recorrer el mapa y acceder a todas sus claves:

```
let mapa= new Map();
mapa.set("nombre","pedro");
mapa.set("apellidos","gomez");
mapa.set("edad",20);

for (let clavevalor of mapa.entries()) {

    console.log(clavevalor);
}
```

El resultado le vemos en la consola:



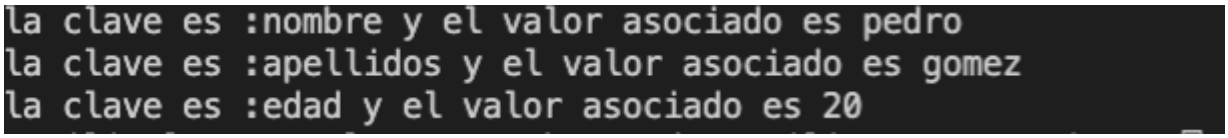

## JavaScript Template String y mapas

Muchas veces con esta opción no es suficiente y queremos acceder de una forma más flexible y poder imprimir cada dato de forma independiente . En ese caso podemos optar por un bucle forEach:

```
let mapa= new Map();
mapa.set("nombre","pedro");
mapa.set("apellidos","gomez");
mapa.set("edad",20);
mapa.forEach((valor,clave)=> {
```

```
    console.log(`la clave es :${clave} y el valor asociado es
${valor}`);
})
```

Nos hemos apoyado en JavaScript Template Strings para mostrar el resultado




De igual forma podríamos acceder unicamente a las claves a través del metodo keys o a los values a través el método values:

```
let mapa= new Map();
mapa.set("nombre","pedro");
mapa.set("apellidos","gomez");
mapa.set("edad",20);

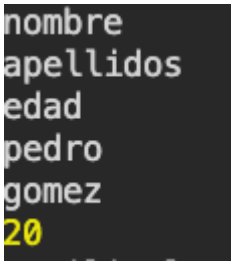
for (let clave of mapa.keys()) {

    console.log(clave);
}

for (let values of mapa.values()) {

    console.log(values);
}
```

El resultado es :




Acabamos de ver como utilizar un Javascript Map ES6 para simplificar el manejo de estructuras de clave valor.

## Otros artículos relacionados:

**CURSO Java Herencia**
**GRATIS**
**APUNTATE!!**

- El uso de JavaScript for in vs for of
- ¿Javascript sincrono o asincrono?
- Javascript filter map reduce y manejo de funciones
- JavaScript ES6