



**Certified Tech
Developer**

The Ultimate Degree

Ejemplos para la clase

JSON + NODE sin módulo

La idea es mostrar todo el camino requerido para utilizar archivos JSON como fuentes de datos.

Tomando el [siguiente archivo de base](#), se puede producir el siguiente código:

```
// Requerimos el módulo File System
const fs = require('fs')

// Utilizamos el método readFileSync para obtener el string
let equipoJson = fs.readFileSync(__dirname + '/equipo.json',
'utf8');

// Convertimos el string a un formato que podamos usar en JS
let equipo = JSON.parse(equipoJson);

// Verificamos los contenidos de la variable
console.log(equipo);
```

JSON + NODE con módulo

Otra posibilidad sería crear un módulo con una función leer() de manera que importen y usen la funcionalidad.

```
const fs = require('fs');

const jsonHelper = {
  // Este método se encarga de leer el archivo y retornar un
  // formato de JS
  // El método asume que el archivo a leer está en el mismo
  // directorio que este módulo
  leer: function (archivo) {
    // Leemos el archivo con la ayuda del módulo fs
    let contenido = fs.readFileSync(__dirname + '/' + archivo,
    'utf8');
    // Transformamos el contenido de string a formato de JS
    return JSON.parse(contenido);
  }
}

module.exports = jsonHelper;
```

Y así podríamos usarlo en nuestro archivo principal.

```
// Ejemplo con el módulo jsonHelper
const jsonHelper = require('./jsonHelper');
let equipoBis = jsonHelper.leer('autos.json');

// Verificamos los contenidos de la variable
console.log('**** equipo con jsonHelper ****');
console.log(equipoBis);
console.log();
```

array.sort() - valores numéricos

La idea es mostrar el funcionamiento básico de sort cuando lo que queremos hacer es ordenar un valor numérico.

```
// Ordenamos por un valor numérico, en este caso edad.  
let ordenadoPorEdad = equipo.sort(function(a, b) {  
    return a.edad - b.edad;  
})  
  
// Verificamos los contenidos de la variable  
console.log('*** equipo ordenado ***');  
console.log(ordenadoPorEdad);  
console.log();
```

array.sort() - valores de texto

Si llegamos cómodos al primer ejemplo y queremos mostrar algo más completo, podemos explicar cómo funciona sort cuando queremos especificar nosotros el criterio de ordenamiento.

En este caso utilizamos comparaciones de strings para ordenar alfabéticamente por apellido.

```
// Ordenamos por un valor de string, en este caso el apellido.
let ordenadoPorApellido = equipo.sort(function(a, b) {
    // Si a es mayor que b, osea que viene después por abecedario
    // retornamos 1 para que quede después en el orden
    if (a.apellido > b.apellido) {
        return 1
    }
    // Si a es menor que b, osea que viene después por
    abecedario
    // retornamos -1 para que quede antes en el orden
    if (a.apellido < b.apellido) {
        return -1
    }
    // Solo nos queda el caso en que a y b son iguales
    // Entonces retornamos 0 para que queden como estaban.
    return 0
})

// Verificamos los contenidos de la variable
console.log('**** equipo ordenado ****');
console.log(ordenadoPorApellido);
console.log();
```