

Parsear JSON

Antonio Espín Herranz

eval()

- Esta función nos permite convertir una cadena JSON en objetos de JS.
- *No es una buena opción, es vulnerable, mejor JSON.parse(), siempre y cuando el navegador sea compatible con JSON.parse()*
- Los () extras son obligatorios, si no, no funciona:
 - `cad = '{"a":1,"b":2,"c":3,"d":4}';`
 - `obj = eval("(" + cad + ")");`
 - `console.log(obj.a, obj.b);`

JSON.parse

- `cad = '{"a":1,"b":2,"c":3,"d":4}';`
- `obj2 = JSON.parse(cad);`
- `console.log(obj.a, obj.b);`
- Convierte una cadena en json en un objeto.

JSON.stringify

- Es la operación contraria. Partimos de un objeto en JSON y lo convierte a una cadena.
 - `let cad = JSON.stringify(obj_json)`

Contexto

- Estas funciones JSON.parse y JSON.stringify nos sirven para almacenar datos en el navegador.
- Los guardamos en una cadena de Json → JSON.stringify y cuando los restauramos los convertimos a objetos de javascript para facilitar el acceso a las propiedades → JSON.parse.

Ejemplo

```
// Definir el objeto en JSON:
```

```
let obj = {nombre:"Ana",edad:48}
```

```
console.log('Nombre: '+obj.nombre+, edad: "+obj.edad+"
```

```
Tipo:"+typeof obj)
```

```
// Convertir el objeto JSON a String:
```

```
let cad = JSON.stringify(obj)
```

```
console.log('Cad: '+cad + " tipo:"+typeof cad)
```

```
// Convertir el String a un objeto JSON:
```

```
let obj2 = JSON.parse(cad);
```

```
console.log('Obj: '+obj2+ " tipo:"+typeof obj2)
```

Importar fichero json

- Podemos tener dos situaciones:
 - Importar el fichero json:
 - Desde un fichero html que haga referencia a un script de JS.
 - En la etiqueta a <script> de html tendremos que utilizar el atributo **type="module"** y la referencia al fichero de JS.
 - Desde un fichero JS (Node.js):
 - Definir un fichero **package.json** y con la propiedad **type = "module"**

```
{  
  "type": "module"  
}
```

Desde HTML

- Asegúrese de que el atributo type está la etiqueta del script esté establecido en module.
 - <script type="module" src="fichero.js" />
- Utilice una aserción de importación para importar el archivo JSON.
- Por ejemplo, import myJson from './example.json' assert {type: 'json'}.
- Esta opción puede dar problemas por las políticas de seguridad de los navegadores. No permiten acceder a ficheros locales.

```
① Access to script at 'file:///D:/antonio/TRABAJO/CURSOS/JavaScript%20ES6/ejemplos/import_json_html/json_import_file.js' from origin 'null' has been blocked by CORS policy: Cross origin requests are only supported for protocol schemes: http, data, isolated-app, chrome-extension, chrome, https, chrome-untrusted. index.html:1
① Failed to load resource: net::ERR_FAILED json_import_file.js:1 ⓘ
```

Desde Node.js

- `import myJson from './ejemplo_json.json' assert {type: 'json'};`
 - En la sentencia import se da un nombre al objeto que representará el objeto JSON.
 - Path al fichero
 - Y nos aseguramos el tipo de los datos en formato json.
 - `console.log(myJson.person);`
 - `console.log(myJson.person.name);`
 - `console.log(myJson.person.country);`

Enlaces

- https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON/parse
- Formateador online:
 - <https://jsonformatter.org/>

Enlaces

- Fetch
 - https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch
- Conversor XML to JSON online:
 - <https://codebeautify.org/xmltojson>
- Comparativa:
 - <https://stackoverflow.com/questions/35549547/fetch-api-vs-xmlhttprequest>