



¿Qué es JSON?

JSON (JavaScript Object Notation) es un formato de intercambio de datos que es fácilmente legible por humanos y fácilmente procesable por las máquinas. JavaScript proporciona métodos para trabajar con JSON. Aquí hay una explicación y ejemplos:

1. Convertir Objeto a JSON:

```
const persona = {  
  nombre: 'Juan',  
  edad: 30,  
  ciudad: 'Ejemploville',  
};
```

```
// Convertir el objeto a JSON
```

```
const personaJSON = JSON.stringify(persona);  
  
console.log(personaJSON);
```

En este ejemplo, `JSON.stringify` convierte el objeto `persona` a una cadena JSON.

2. Convertir JSON a Objeto:

```
const personaJSON = '{"nombre":"Juan","edad":30,"ciudad":"Ejemploville"}';  
  
// Convertir la cadena JSON a un objeto
```

```
const persona = JSON.parse(personaJSON);
```

```
console.log(persona);
```

JSON.parse convierte la cadena JSON de vuelta a un objeto JavaScript.

3. Trabajar con JSON anidado:

```
const libro = {  
  titulo: 'JavaScript para Principiantes',  
  autor: {  
    nombre: 'Maria',  
    edad: 35,  
  },  
  publicado: true,  
};
```

```
// Convertir el objeto a JSON con formato
```

```
const libroJSON = JSON.stringify(libro, null, 2);  
  
console.log(libroJSON);
```

El segundo argumento de JSON.stringify permite especificar una función de reemplazo o un array de propiedades a incluir. El tercer argumento (en este caso, 2) proporciona la indentación para hacer que la salida sea más legible.

4. Trabajar con JSON y Fetch:

```
// Haciendo una solicitud GET a una API que devuelve JSON
```

```
fetch('https://jsonplaceholder.typicode.com/todos/1')  
  .then(response => response.json())  
  .then(data => console.log(data))
```

```
.catch(error => console.error('Error:', error));
```

En este ejemplo, `response.json()` se utiliza para convertir la respuesta de la solicitud Fetch a un objeto JavaScript.