

Alumno/a: Rafael Miguel Cruz Álvarez

Francisco Javier González Sabariego

Tarea 3.1. Presentación Objetos predefinidos en JavaScript: String, Regexp & Json

¿Qué es JSON?

JSON es una estructura de datos basada en la notación de un objeto literal de JavaScript.

A diferencia de un objeto literal de JavaScript, JSON no soporta ni métodos ni funciones por tanto que es únicamente una estructura de datos basada en clave → valor.

```
a = {
  id: 433,
  name: "TV Sony",
  model: "Bravia",
  price: 1856.95,
  desc: "asdasfsa",
  properties: [],
  stock: true,
  opinions: [
    {
      user: "Perico el de los Palotes",
      opinion: "afds",
      value: 5,
    }
  ]
}
```

Tipos de datos

Métodos

JSON.stringify(json): El método stringify recibe como parámetro un objeto tipo JSON y lo convierte en una cadena de caracteres con el formato del objeto JSON.

```
JSON.stringify(a)
{"id":433,"name":"TV Sony","model":"Bravia","price":1856.95,"desc":"asdasfsa","properties":[],"stock":true,"opinions":[{"user":"Perico el de los Palotes","opinion":"afsds","value":5}]}
```

Métodos

JSON.parse(string): El método parse recibe como parámetro una cadena con formato JSON y lo convierte en un objeto JSON.

```
JSON.parse('{"name":"Ejemplo","price":30}')
```

▼ {name: "Ejemplo", price: 30} ⓘ

name: "Ejemplo"

price: 30

► __proto__: Object

Clonando un JSON: Spread Operator (1ª parte)

A la hora de copiar un JSON es recomendable hacer uso del **spread operator**, sin embargo, hay que tener presente que aquellos atributos que posean un objeto mutable (array, JSON) también deberán ser copiados con un spread operator:

```
a = {
  name: "Ana",
  age: 27,
  attributes: [
    "Smart",
    "Creative",
    "Curious"
  ]
}
► {name: "Ana", age: 27, attributes: Array(3)}
b = {...a}
► {name: "Ana", age: 27, attributes: Array(3)}
b.age = 22
22
b.attributes.push("Beautiful")
4
b
▼ {name: "Ana", age: 22, attributes: Array(4)} ⓘ
  age: 22
  attributes: (4) ["Smart", "Creative", "Curious", "Beautiful"]
  name: "Ana"
  ► __proto__: Object
a
▼ {name: "Ana", age: 27, attributes: Array(4)} ⓘ
  age: 27
  attributes: (4) ["Smart", "Creative", "Curious", "Beautiful"]
  name: "Ana"
  ► __proto__: Object
```

```
a = {
  name: "Ana",
  age: 27,
  attributes: [
    "Smart",
    "Creative",
    "Curious"
  ]
}
► {name: "Ana", age: 27, attributes: Array(3)}
b = { ...a, attributes: [...a.attributes] }
► {name: "Ana", age: 27, attributes: Array(3)}
b.age = 22
22
b.attributes.push( "Beautiful" )
4
b
▼ {name: "Ana", age: 22, attributes: Array(4)} ⓘ
  age: 22
  attributes: (4) ["Smart", "Creative", "Curious", "Beautiful"]
  name: "Ana"
  ► __proto__: Object
a
▼ {name: "Ana", age: 27, attributes: Array(3)} ⓘ
  age: 27
  attributes: (3) ["Smart", "Creative", "Curious"]
  name: "Ana"
  ► __proto__: Object
```

Clonando un JSON: Spread Operator (2ª parte)

Pero ¿qué pasaría si quisieramos clonar una estructura de datos tan compleja como ésta? Sería una locura...

```
▼ {nombre: "Nymeth", apodo: "La piromántica", raza: "Humana", genero: "Femenino", edad: 24, ...}
  ▼ afiliaciones: Array(2)
    ▶ 0: {nombre: "La llama azul", organizacion: "Escuela mágica", ubicacion: {...}}
    ▶ 1: {nombre: "Logia de los siete", organizacion: "Eruditos", ubicacion: {...}}
      length: 2
    ▶ __proto__: Array(0)
  apodo: "La piromántica"
  clase: "Hechicero/a"
  corrupcion_caos: 20
  edad: 24
  genero: "Femenino"
  ▼ historia: Array(3)
    0: "párrafo1"
    1: "párrafo2"
    2: "párrafo3"
      length: 3
    ▶ __proto__: Array(0)
  ▼ idiomas: Array(2)
    ▶ 0: {idioma: "Común", nivel: 100}
    ▶ 1: {idioma: "Draconido", nivel: 70}
      length: 2
    ▶ __proto__: Array(0)
  ▼ inventario: Array(3)
    ▶ 0: {tipoObjeto: "Arma", rareza: "epico", nombre: "Báculo de las tormentas", peso: 1.5, valor: {...}, ...}
    ▶ 1: {tipoObjeto: "Armadura", rareza: "epico", nombre: "Aro del búho", peso: 0.6, valor: {...}, ...}
    ▶ 2: {tipoObjeto: "Armadura", rareza: "poco_comun", nombre: "Anillo de luz", peso: 0.1, valor: {...}, ...}
      length: 3
    ▶ __proto__: Array(0)
  mana_actual: 84
  mana_max: 84
  monedas: {oro: 100, plata: 20, cobre: 12}
  naturalDe: {ciudad: "Tremton", region: "", nacion: "Imperio de Jayndor"}
  nivel: 3
  nombre: "Nymeth"
  peso_max: 30
  raza: "Humana"
  vida_actual: 21
  vida_max: 21
  ▶ __proto__: Object
```

Clonando un JSON: Deep Clone

Para realizar una copia es recomendable hacer uso del **Deep Clone o Deep Copy**, que consiste en convertir en cadena el objeto a clonar usando **stringify()** y, posteriormente, convertirla a un nuevo JSON totalmente independiente con **parse()**.

```
a = {
  name: "Ana",
  age: 27,
  attributes: [
    "Smart",
    "Creative",
    "Curious"
  ]
}
▶ {name: "Ana", age: 27, attributes: Array(3)}
b = JSON.parse( JSON.stringify(a) )
▶ {name: "Ana", age: 27, attributes: Array(3)}
b.age = 22
22
b.attributes.push( "Beautiful" )
4
b
▼ {name: "Ana", age: 22, attributes: Array(4)} ⓘ
  ▶ attributes: (4) ["Smart", "Creative", "Curious", "Beautiful"]
  ▶ name: "Ana"
  ▶ __proto__: Object
a
▼ {name: "Ana", age: 27, attributes: Array(3)} ⓘ
  ▶ age: 27
  ▶ attributes: (3) ["Smart", "Creative", "Curious"]
  ▶ name: "Ana"
  ▶ __proto__: Object
```