



ES6

Nuevos métodos



Nuevos métodos

Asignación de propiedad de objeto:

Nueva función para asignar propiedades enumerables de uno o más objetos fuente a un objeto de destino (método ***assign***).

Nuevos métodos

Búsqueda de un elemento en un arreglo:

Nueva función para encontrar un elemento en un arreglo.

```
[ 1, 3, 4, 2 ].find(x => x > 3) // 4
```

```
[ 1, 3, 4, 2 ].findIndex(x => x > 3) // 2
```

Nuevos métodos

Repetición de cadenas

Nueva funcionalidad de repetición de cadenas.

```
" ".repeat(4 * depth)
```

```
"foo".repeat(3)
```

Nuevos métodos

Búsqueda de cadenas

Nuevas funciones de cadena específicas para buscar una subcadena.

```
var cadena = "hola cara de bola";  
console.log(cadena.startsWith("ola",1));  
console.log(cadena.endsWith("ola"));  
console.log(cadena.includes("ola"));  
console.log(cadena.includes("ola", 1));  
console.log(cadena.includes("ola", 2));
```

Nuevos métodos

Comprobación de tipo de número

Nuevas funciones para verificar números que no son números y números finitos.

```
Number.isNaN(42) === false
```

```
Number.isNaN(NaN) === true
```

```
Number.isFinite(Infinity) === false
```

```
Number.isFinite(-Infinity) === false
```

```
Number.isFinite(NaN) === false
```

```
Number.isFinite(123) === true
```

Nuevos métodos

Número de verificación de seguridad

Comprobar si un número entero está en el rango seguro, es decir, está representado correctamente por JavaScript (donde todos los números, incluidos los números enteros, son técnicamente números en coma flotante).

± 9007199254740991 o $\pm 9,007,199,254,740,991$

`Number.isSafeInteger(42) === true`

`Number.isSafeInteger(9007199254740992) === false`

Nuevos métodos

Comparación de números

Disponibilidad de un valor estándar de Epsilon para una comparación más precisa de los números de coma flotante.

```
console.log(0.1 + 0.2 === 0.3); // false
```

```
console.log(Math.abs((0.1 + 0.2) - 0.3) < Number.EPSILON); // true
```


Nuevos métodos

Número de truncamiento

Trunca un número de coma flotante a su parte integral, soltando completamente la parte fraccionaria.

```
console.log(Math.trunc(42.7)) // 42  
console.log(Math.trunc( 0.1)) // 0  
console.log(Math.trunc(-0.1)) // -0
```

Nuevos métodos

Determinación de signo de número

Determine el signo de un número, incluidos los casos especiales de cero firmado y no número.

```
console.log(Math.sign(7)) // 1
console.log(Math.sign(0)) // 0
console.log(Math.sign(-0)) // -0
console.log(Math.sign(-7)) // -1
console.log(Math.sign(NaN)) // NaN
```

Referencias

<http://es6-features.org/#NumberSignDetermination>