JavaScript Array **some()** Method

Where at least one element makes the cut!

RAJA MSR
Full Stack Engineer

rajamsr.com

JavaScript array **some()** method

The **some()** array method checks whether at least one element meets the given condition.
It will return **true** if at least one element meets the condition; otherwise, it will return **false**.

Searching array elements

The some() method can also be used for searching array elements.

```
const languages = ["JavaScript", "Java", "C#", "TypeScript", "F#"];
const stringToSearch = 'C#'

const hasCSharp = languages.some(e => e === stringToSearch);
console.log(hasCSharp)
// Output: true
```

Checking if any element in an array passes a test using **some()**

You could use the **some()** method to check if any elements in an array are even numbers.

```
const numbers = [3, 5, 7, 4, 9, -11, 15];
const hasAnyEvenNumber = numbers
    .some(e => e % 2 === 0);
console.log(hasAnyEven)
// Output: true
```

Checking if an array contains a certain value

You could use the **some()** method to check if the month exists in the month names array.

```
const months = ["Jan", "Feb", "Mar", "Apr", "May"];
const monthToSearch = 'Mar'

const exists = months
    .some(e => e === monthToSearch);
console.log(exists)
// Output: true
```

Validating a form with required fields

Sometimes, default parameters can be a bit shy and not show up when you expect them to.

```
const formInputs = [
  { name: 'username', value: 'johndoe' },
  { name: 'email', value: '' },
  { name: 'password', value: 'p@ssword1' },
  { name: 'confirmPassword', value: 'p@ssword1' },
];

const hasEmptyFields = formInputs
    .some(input => input.value.trim() === '');

if (hasEmptyFields) {
  console.log('Please fill in all the required fields.');
} else {
  console.log('Form validation is success.');
}
```

Filtering an array based on multiple conditions

To check whether any products with prices less than $50 are available in stock from a list, you can use the code example shown below:

```
const products = [
  { name: "T-Shirt", price: 20, inStock: true },
  { name: "Jeans", price: 50, inStock: false },
  { name: "Sneakers", price: 80, inStock: true },
  { name: "Backpack", price: 40, inStock: false },
];

const hasAffordableAndInStockProducts = products
    .some(product => product.price < 50 && product.inStock);

console.log(hasAffordableAndInStockProducts);
// Output: true
```

Checking if an array contains any element from another array

Using the some() function along with the includes() array method, you can compare array elements with another array.

```
const tokens = ['Hello', 'JavaScript','onerror'];
const restrictedWords = ['script', 'html', 'onerror','alert'];

const hasRestrictedWords = tokens
  .some(u => restrictedWords.includes(u));

if (hasRestrictedWords) {
  console.log('Input string contains restricted word.');
} else {
  console.log('Input string is sanitized.');
}
// Output: "Input string contains restricted word."
```

Fixing JavaScript some() is not a function error

If you apply some() method to a non-array type, then it will throw "some is not a function" error.

```
const greet = 'Hello JavaScript!';
console.log(greet.some(e => greet.length > 0))
// Output: TypeError: greet.some is not a function
```

```
const greet = 'Hello JavaScript!';
if (Array.isArray(greet)) {
  console.log(greet.some(e => greet.length > 0))
}
else {
  console.log('Input type is not an array.')
}
// Output: Input type is not an array.
```

Limitations of the JavaScript array some() method

Using the some() method only determines if one or more elements meet a certain condition. It does not determine which elements meet the condition.

```
const numbers = [1, 3, 5, 6, 7, 9];
const hasEvenNumber = numbers
      .some(n => n % 2 === 0);

console.log(hasEvenNumber)
  // Output: true
```

Note: Using some() functions, we can’t find or select which number (in this case, 6) satisfied the condition from the array.

Thank You!

Over to you: Have you used JavaScript array **some()** method? Leave your comment.

rajamsr

rajamsrtweets

rajamsr.facebook

www.rajamsr.com

I would appreciate it if you could repost this!