Array.prototype.findIndex()

Summary: in this tutorial, you will learn how to use the JavaScript Array `findIndex()` method to find the index of the first element in an array that satisfies a test.

Introduction to the JavaScript Array findIndex() Method

The `findIndex()` method allows you to find the index of the first element in an array that satisfies a provided function.

Here's the syntax of the `findIndex()` method:

```
findIndex(callbackFn)
```

The `findIndex()` method iterates over the elements of the array and executes the `callbackFn` function on each element.

If a matching element is found, the `callbackFn` returns a *truthy* value and stops searching. If no matching value is found, the `callbackFn` function returns a *falsy* value.

The `findIndex()` method returns the index of the first matching element or -1 if no matching element is found.

The `callbackFn` function has the following syntax:

```
callbackFn(element, index, array)
```

The `callbackFn` takes three arguments:

- `element` is the current element in the array.

- `index` is the index of the current element being processed.

- `array` is the array that the `findIndex()` is called on.

thisArg

The `findIndex()` method has a second optional argument called `thisArg`:

```
findIndex(callbackFn, thisArg)
```

The `thisArg` is used as `this` inside the `callbackFn` function.

JavaScript Array findIndex() examples

Let's take some examples of using the JavaScript Array `findIndex()` method.

1) Basic findIndex() method example

The following example uses the `findIndex()` method to return the index of the first occurrence of the number `7` in the `numbers` array:

```
let numbers = [1, 5, 7, 8, 10, 7];
let index = numbers.findIndex((n) => n === 7);

console.log({ index });
```

Output:

```
{ index: 2 }
```

How it works.

First, define a `numbers` array that has six numbers:

```
let ranks = [1, 5, 7, 8, 10, 7];
```

Second, find the first occurrence of the number `7` in the `numbers` array using the `findIndex()` method:

```
let index = ranks.findIndex((n) => n === 7);
```

The callback function returns `true` if the element in the `numbers` array is `7` or `false` otherwise:

```
(n) => n === 7
```

Third, display the `index` in the console:

```
console.log({ index });
```

2) Using the findIndex() method with complex conditions

The following example uses the `findIndex()` method to find the index of the first occurrence of the number `7` after the second element in the `ranks` array:

```
let numbers = [1, 5, 7, 8, 10, 7];
let index = numbers.findIndex((n, i) => n === 7 && i > 2);

console.log({ index });
```

Output:

```
{ index: 5 }
```

3) Using the Array findIndex() method with an array of objects

The following example uses the `findIndex()` method to find the index of the first product whose price is greater than `1000` :

```
const products = [
  { name: 'Phone', price: 999 },
  { name: 'Computer', price: 1999 },
  { name: 'Tablet', price: 1255 },
];
```

```
const index = products.findIndex((p) => p.price > 1000);

console.log({ index });
```

Output:

```
{ index: 1 }
```

Summary

- Use the JavaScript Array `findIndex()` method to find the first element that satisfies a given test.