Navigating the Old and New Ways to Retrieve Last Element in JS

Wondering what is the new way to retrieve last element of an array in JS ? This presentation will provide all the answers you need!

by Gourav Roy

OLD Way - To get Last Item of an Array

The traditional way to select the last item of an array in JavaScript is by using the array's length property.

In this example, myArray.length gives the total number of elements in the array, and myArray.length - 1 gives the index of the last element. This method works reliably and is widely supported across all JavaScript environments.

```
const myArray = [1, 2, 3, 4, 5];

// Old way using array length
const lastItem = myArray[myArray.length - 1];

console.log(lastItem); // Output: 5
```

NEW Way - To get Last Item of an Array

The new way to select the last item of an array in JavaScript is by using the Array.prototype.at method in combination with a negative index. Introduced in ECMAScript 2022 (ES13), the at method allows for more concise when accessing elements at specific indices.

- The Array.prototype.at method is employed to access an element at a specified index.
- A negative index, such as -1, refers to positions counting from the end of the array.
- In this example, myArray.at(-1) returns the last element of the array myArray.

```
// Sample array
const myArray = [10, 20, 30, 40, 50];

// Using Array.prototype.at with a negative index to select the last ite
const lastItem = myArray.at(-1);

// Output the result
console.log("Last item of the array:", lastItem);
```