JS JavaScript TUTORIAL

JavaScript Array Methods

This section provides you with the JavaScript Array methods that allow you to manipulate arrays effectively.

Section 1. Array properties

- length property – show you how to use the length property of an array effectively.

Section 2. Adding/removing elements

- push() – add one or more elements to the end of an array.
- unshift() – add one or more elements to the beginning of an array.
- pop() – remove an element from the end of an array.
- shift() – remove the first element from an array.
- splice() – manipulate elements in an array such as deleting, inserting, and replacing elements.
- slice() – copy elements of an array.

Section 3. Finding elements

- indexOf() – locate an element in an array.
- includes() – check if an element is in an array.

- find() – find an element in an array
- findIndex() – find the index of an element in an array.

Section 4. High-order methods

- map() – transform array elements.
- filter() – filter elements in an array.
- reduce() – reduce elements of an array to a value.
- every() – check if every element in an array passes a test.
- some() – check if at least one element in an array passed a test.
- sort() – sort elements in an array.
- forEach() – loop through array elements.

Section 5. Manipulating Arrays

- concat() – merge two arrays into an array.

Section 6. Creating Arrays

- of() – improve array creation.
- from() – create arrays from array-like or iterable objects.

Section 7. Flattening arrays

- flat() – flatten an array recursively up to a specified depth.
- flatMap() – execute a mapping function on every element and flatten the result.

Section 8. Arrays to Strings

- join() – concatenate all elements of an array into a string separated by a separator.

Section 9. Advanced Operations

- Destructuring – show you how to assign the elements of an array to variables.
- Spread operator – learn how to use the spread operator effectively.

Section 10. Accessing elements

- at() – access array elements using both positive and negative indexes.

Section 11. Reversing elements

- reverse() – reverse the order of elements in place and return the same array with the elements in the reversed order.
- toReversed() – reverse the order of elements of an array and return the new array with the elements in the reversed order.

Section 11. Multidimensional Array

- Multidimensional Array – learn how to work with multidimensional arrays in JavaScript.