@Code.Clash

JavaScript : Array Methods

Next →

1

@Code.Clash

Hey Everyone 👋

In this Post, we will learn about the JavaScript Array Methods with the help of examples.

Arrays provide a lot of methods. To make things easier.

Do Like, save and Share This Post If You Found This Helpful.

Next →

2

@CODE.CLASH

Popping and Pushing

1. push()

- Adds a new element to an array (at the end).

2. pop()

- Removes the last element from an array.

3. unshift()

- Adds a new element to an array (at the beginning).

4. shift()

- Removes the first element from an array.

```
const month = ['Jan', 'April', 'June', 'Nov', 'Dec'];
const numArr = [602, 3020, 1003, 622, 42, 589, 84];

// *-*-*-*-* This Methods Returns New length of Array *-*-*-*-*
console.log(month.push('October'));
// Expected Output : [ 'Jan', 'April', 'June', 'Nov', 'Dec', 'October' ]
console.log(month.unshift('Feb'));
// Expected Output : [ 'Feb', 'Jan', 'April', 'June', 'Nov', 'Dec', 'October' ]

// *-*-*-*-* This Methods Returns Deleted Element *-*-*-*-*
console.log(numArr.pop());
// Expected Output : 84
console.log(numArr.shift());
// Expected Output : 602
```

NEXT →

3

@CODE.CLASH

Sort Method

1. sort()

- This method first convert element into string and Sort According to element first character, then second and so on...
- Returns Sorted Array.

```
const months = ['Jan', 'April', 'June', 'Nov', 'Dec'];
const numArray = [602, 3020, 1003, 622, 42, 589, 84];

console.log(months.sort());
// Expected output [ 'April', 'Dec', 'Jan', 'June', 'Nov' ]

console.log(numArray.sort());
// Expected output [ 1003, 3020,  42, 589,  602, 622, 84 ]

// *-*-*-*-* For properly sort integer element *-*-*-*-*
console.log(numArray.sort((a, b) => a - b));
// Expected Output [42, 84, 589, 602, 622, 1003, 3020 ]
```

NEXT →

4

@CODE.CLASH

Find And Filter Methods

1. indexOf()

- Returns the first index of occurance of an array element, or -1 if it is not found.

2. lastIndexOf()

- Returns the index of the last occurrence of a specified element in the array.

3. includes()

- Checks if an array contains a specified element or not.

```
const months = ['Jan', 'April', 'June', 'Nov', 'Dec', 'April'];

console.log(months.indexOf('April'));
// Expected output: 1

console.log(months.lastIndexOf('April'));
// Expected output: 5

console.log(months.includes('June'));
// Expected output: true
```

NEXT →

4. find()

- Returns the value of the first array element that satisfies the provided test function.

5. findIndex()

- Returns the index of the first array element that satisfies the provided test function or else returns -1.

6. filter()

- Returns a new array with all elements that pass the test defined by the given function.

```
const prices = [100, 200, 300, 400, 500, 600, 700];

console.log(prices.find((price) ⇒ price < 400));
// Expected output: 100

console.log(prices.findIndex((price) ⇒ price > 400));
// Expected output: 4

console.log(prices.filter((price) ⇒ price ⩽ 400));
// Expected output: [ 100, 200, 300, 400 ]
```

NEXT →

6

@CODE.CLASH

Splicing and Slicing

1. splice()

- This method can be used to perform insert, update and delete operation.
- Return a deleted element

2. slice()

- Slices out a piece of an array into a new array.
- slice() method creates a new array.

```
const month = ['Jan', 'April', 'June', 'Nov', 'Dec'];

// Update from index 0 to 2 step.
console.log(month.splice(0, 2,'Feb','March' ));
// Expected output: ['Jan', 'April']

console.log(month.slice(1, 3));
// Expected output: ['March', 'June']
```

NEXT →

7

@CODE.CLASH

Other Useful Methods

1. concat()

- This method creates a new array by merging existing arrays.

2. toString()

- This converts an array to a string.

3. join()

- This method also joins all array elements into a string, but in addition you can specify the separator.

```
const girls = ['Cecilie', 'Lone'];
const boys = ['Emil', 'Tobias', 'Linus'];

console.log(girls.concat(boys));
// Expected Output :- ['Cecilie','Lone','Emil','Tobias','Linus']

console.log(boys.toString());
// Expected Output :- Emil,Tobias,Linus

console.log(boys.join(' * '));
// Expected Output :- Emil * Tobias * Linus
```

NEXT →

8

@CODE.CLASH

If you have any queries then tell me in the comment box.

NEXT →

@CODE.CLASH

THANKS FOR YOUR ATTENTION

CODE.CLASH

IMTIYAZ NANDASANIYA

LIKE AND SAVE IT FOR LATER