

Top 14 JavaScript Array-Methoden

.push()

-> Fügt ein oder mehrere Elemente am Ende eines Arrays hinzu.

```
let arr = [1, 2];  
arr.push(3); // [1, 2, 3]
```

.pop()

-> Entfernt das letzte Element eines Arrays.

```
let arr = [1, 2, 3];  
arr.pop(); // [1, 2]
```

.shift()

-> Entfernt das erste Element eines Arrays.

```
let arr = [1, 2, 3];  
arr.shift(); // [2, 3]
```

.unshift()

-> Fügt ein oder mehrere Elemente am Anfang eines Arrays hinzu.

```
let arr = [2, 3];  
arr.unshift(1); // [1, 2, 3]
```

.slice()

-> Extrahiert einen Teil eines Arrays in ein neues Array.

```
let arr = [1, 2, 3, 4];  
let newArr = arr.slice(1, 3); // [2, 3]
```

.splice()

-> Fügt Elemente hinzu, entfernt oder ersetzt sie im Array.

```
let arr = [1, 2, 3];  
arr.splice(1, 1, 4); // [1, 4, 3]
```

.map()

-> Erstellt ein neues Array durch Transformation jedes Elements.

```
let arr = [1, 2];  
let doubled = arr.map(x => x * 2); // [2, 4]
```

.filter()

-> Erstellt ein neues Array mit Elementen, die eine Bedingung erfüllen.

```
let arr = [1, 2, 3];  
let even = arr.filter(x => x % 2 === 0); // [2]
```

.reduce()

-> Reduziert ein Array auf einen einzigen Wert.

```
let arr = [1, 2, 3];  
let sum = arr.reduce((a, b) => a + b); // 6
```

.forEach()

-> Führt eine Funktion für jedes Element aus (ohne Rückgabewert).

```
let arr = [1, 2];  
arr.forEach(x => console.log(x));
```

.find()

-> Findet das erste Element, das eine Bedingung erfüllt.

```
let arr = [1, 2, 3];  
let val = arr.find(x => x > 1); // 2
```

.findIndex()

-> Findet den Index des ersten Elements, das eine Bedingung erfüllt.

```
let arr = [1, 2, 3];  
let index = arr.findIndex(x => x > 1); // 1
```

.includes()

-> Überprüft, ob ein Array einen bestimmten Wert enthält.

```
let arr = [1, 2, 3];  
arr.includes(2); // true
```

.sort()

-> Sortiert die Elemente eines Arrays in-place.

```
let arr = [3, 1, 2];  
arr.sort(); // [1, 2, 3]
```