

JavaScript Arrays Cheatsheet

Inhalt:

- Elemente Hinzufügen, Entfernen & Ersetzen
- Länge, Elemente & Indizes Bestimmen
- Arrays Kombinieren & Neue Arrays Erstellen

Elemente Hinzufügen, Entfernen & Ersetzen

.push() – Elemente ans Ende hinzufügen

Fügt ein oder mehrere Elemente am Ende eines Arrays hinzu.

Beispiel:

```
var obstSorten = ["Navel", "Clementine", "Valencia", "Bergamotte"];  
obstSorten.push("Jaffa");  
console.log(obstSorten);  
// ["Navel", "Clementine", "Valencia", "Bergamotte", "Jaffa"]
```

.pop() – Letztes Element entfernen

Entfernt ein einzelnes Element vom Ende eines Arrays.

Beispiel:

```
obstSorten.pop();  
console.log(obstSorten);  
// ["Navel", "Clementine", "Valencia"]
```

.unshift() – Elemente am Anfang hinzufügen

Fügt ein oder mehrere Elemente am Anfang eines Arrays hinzu.

Beispiel:

```
obstSorten.unshift("Satsuma", "Sauer");  
console.log(obstSorten);  
// ["Satsuma", "Sauer", "Navel", "Clementine", "Valencia", "Bergamotte"]
```

.shift() – Erstes Element entfernen

Fügt ein oder mehrere Elemente am Anfang eines Arrays hinzu.

Beispiel:

```
obstSorten.shift();  
console.log(obstSorten);  
// ["Clementine", "Valencia", "Bergamotte"]
```

.splice() – Elemente entfernen, hinzufügen oder ersetzen

Elemente entfernen:

```
obstSorten.splice(1, 2);  
console.log(obstSorten);  
// ["Navel", "Bergamotte"]
```

Elemente hinzufügen:

```
obstSorten.splice(3, 0, "Blutorange", "Seville");  
console.log(obstSorten);  
// ["Navel", "Clementine", "Valencia", "Blutorange", "Seville", "Bergamotte"]
```

Elemente ersetzen:

```
obstSorten.splice(2, 1, "Hamlin");  
console.log(obstSorten);  
// ["Navel", "Clementine", "Hamlin", "Bergamotte"]
```

.slice() – Einen Teil eines Arrays kopieren

Beispiel:

```
console.log(obstSorten.slice(1));  
// ["Clementine", "Valencia", "Bergamotte"]  
console.log(obstSorten.slice(1, 3));  
// ["Clementine", "Valencia"]
```

Ein neues Array mit slice() erstellen:

```
var neueObstSorten = obstSorten.slice(0, 1);  
neueObstSorten.push("Zitrone", "Pomelo");  
console.log(neueObstSorten);  
// ["Navel", "Zitrone", "Pomelo"]  
console.log(obstSorten);  
// ["Navel", "Clementine", "Valencia", "Bergamotte"]
```

Länge, Elemente & Indizes Bestimmen

.length – Anzahl der Elemente ermitteln:

```
console.log(obstSorten.length);  
// 4
```

.includes() – Prüfen, ob ein Element vorhanden ist:

```
console.log(obstSorten.includes("Clementine"));  
// true  
console.log(obstSorten.includes("Hamlin"));  
// false
```

indexOf() – Index eines Elements finden:

```
var zahlen = [12, 30, 4, 12, 4, 8, 25, 8];  
var index = zahlen.indexOf(4);  
console.log(index);  
// 2
```

Arrays Kombinieren & Neue Arrays Erstellen

.concat() – Arrays verbinden:

```
var artenOrangen = ["Navel", "Clementine", "Valencia"];
var artenGrapefruit = ["Marsh", "Flame", "Ruby Red"];
var zitrusfrüchte = artenOrangen.concat(artenGrapefruit);
console.log(zitrusfrüchte);
// ["Navel", "Clementine", "Valencia", "Marsh", "Flame", "Ruby Red"]
```

Spread-Operator (...) – Arrays zusammenführen

```
var alleZitrusfrüchte = [...artenOrangen, ...artenGrapefruit];
console.log(alleZitrusfrüchte);
// ["Navel", "Clementine", "Valencia", "Marsh", "Flame", "Ruby Red"]
```

Spread-Operator für Funktionen nutzen:

```
var schritte = [4350, 8098, 2444, 2536, 9875, 4456, 2453];
console.log(Math.min(...schritte));
// 2444
```

.filter() – Bestimmte Elemente herausfiltern

```
var gefilterteOrangen = zahlen.filter(function (item) {
    return item > 10;
});
console.log(gefilterteOrangen);
// [12, 30, 12, 25]
```

.map() – Elemente verändern

```
var doppelteZahlen = zahlen.map(function (item) {
    return item * 2;
});
console.log(doppelteZahlen);
// [24, 60, 8, 24, 8, 16, 50, 16]
```

.join() – Array in eine Zeichenkette umwandeln

```
var alphabet = ["a", "b", "c"];  
console.log(alphabet.join("?"));  
// "a?b?c"
```