JavaScript Array Sort

Back to JS page

Table of Contents

- JavaScript Array Sort
  - Alphabetic Sort
    - Example 1
  - Reverse Sort
    - Example 2
  - Numeric Sort
    - Example 3
  - The Compare Function
    - Example 4
  - Find Min or Max with sort()
    - Example 5
  - Document
  - Reference

Alphabetic Sort

The `sort()` method sorts an array alphabetically:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Alphabetic Sort</h4>
<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];

// Sort alphabetically
fruits.sort();

document.getElementById("demo").innerHTML =
  "Original: Banana, Orange, Apple, Mango<br>" +
  "After sort(): " + fruits;
</script>

</body>
</html>
```

Example 1

Result View Example

Reverse Sort

The `reverse()` method reverses the elements in an array. You can reverse after sorting for descending order:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Reverse Sort</h4>
<p id="demo"></p>

<script>
const fruits = ["Banana", "Orange", "Apple", "Mango"];

// Sort first, then reverse
fruits.sort();
fruits.reverse();

document.getElementById("demo").innerHTML =
  "After sort() and reverse(): " + fruits;
</script>

</body>
</html>
```

Example 2

Result View Example

Numeric Sort

The `sort()` method sorts numbers as strings, so `25` comes before `100`. Use a compare function for numeric sort:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Numeric Sort</h4>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];

// Without compare function: sorts as strings (WRONG for numbers)
const wrong = points.slice();
wrong.sort();

// With compare function: sorts as numbers
const right = points.slice();
right.sort(function(a, b) { return a - b; });

document.getElementById("demo").innerHTML =
  "Original: " + points + "<br><br>" +
  "Without compare function: " + wrong + " (wrong for numbers)<br>" +
  "With compare function: " + right + " (correct)";
</script>

</body>
</html>
```

Example 3

Result View Example

The Compare Function

The compare function takes two arguments `a` and `b`. If the result is:

- **Negative**: `a` is sorted before `b`
- **Positive**: `b` is sorted before `a`
- **Zero**: no change

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>The Compare Function</h4>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];

// Ascending: a - b
const ascending = points.slice();
ascending.sort(function(a, b) { return a - b; });

// Descending: b - a
const descending = points.slice();
descending.sort(function(a, b) { return b - a; });

document.getElementById("demo").innerHTML =
  "Original: " + points + "<br><br>" +
  "Ascending (a - b): " + ascending + "<br>" +
  "Descending (b - a): " + descending;
</script>

</body>
</html>
```

Example 4

Result View Example

Find Min or Max with sort()

You can find the lowest or highest value in an array by sorting and accessing the first/last element:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Array Sort</h2>
<p>From w3schools.com, Experiment by Teeratus_R</p>

<h4>Find Min or Max with sort()</h4>
<p id="demo"></p>

<script>
const points = [40, 100, 1, 5, 25, 10];

// Sort ascending to find min and max
const sorted = points.slice();
sorted.sort(function(a, b) { return a - b; });

const min = sorted[0];
const max = sorted[sorted.length - 1];

document.getElementById("demo").innerHTML =
  "Array: " + points + "<br><br>" +
  "Sorted ascending: " + sorted + "<br>" +
  "Minimum value: " + min + "<br>" +
  "Maximum value: " + max;
</script>

</body>
</html>
```

Example 5

Result View Example

Document

Document in project

You can Download PDF file.

Reference

- W3Schools JavaScript Array Sort