JavaScript Sorting Arrays

- description : JavaScript Sorting Arrays
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-04-27

The source of this article

JavaScript Sorting Arrays

Sorting an Array

`sort()` 메서드는 배열을 알파벳 순서로 정렬합니다.

```
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML = fruits; //
Banana,Orange,Apple,Mango

function myFunction() {
  fruits.sort();
  document.getElementById("demo").innerHTML = fruits;   //
Apple,Banana,Mango,Orange
}
```

Reversing an Array

`reverse()` 메서드는 배열의 요소를 뒤집습니다.

이를 사용하여 배열을 내림차순으로 정렬할 수 있습니다.

```
// Create and display an array:
let fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo1").innerHTML = fruits;    //
Banana,Orange,Apple,Mango

function myFunction() {
```

```
  fruits.sort();            // First sort the elements of fruits
  document.getElementById("demo2").innerHTML = fruits;  //
Apple,Banana,Mango,Orange
  fruits.reverse();         // Them reverse the order of the elements
  document.getElementById("demo3").innerHTML = fruits;  //
Orange,Mango,Banana,Apple
}
```

Numeric Sort

기본적으로, sort() 함수는 값을 문자열로 정렬합니다.

이것은 문자열에서 잘 작동합니다 (“Apple”은 “Banana”앞에 옵니다).

그러나 숫자를 문자열로 정렬하면 “ 2 ” 가 “ 1 ” 보다 크기 때문에 “ 25 ” 가 “ 100 ” 보다 큽니다.

이 때문에 sort() 메서드는 숫자를 정렬할 때 잘못된 결과를 생성합니다.

비교 기능(compare function)을 제공하여 이 문제를 해결할 수 있습니다:

```
let points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;   // 40,100,1,5,25,10

function myFunction() {
  points.sort(function (a, b) { return a - b });
  document.getElementById("demo").innerHTML = points; // 1,5,10,25,40,100
}
```

동일한 방법을 사용하여 배열을 내림차순으로 정렬합니다:

```
let points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;   // 40,100,1,5,25,10

function myFunction() {
  points.sort(function (a, b) { return b - a });
  document.getElementById("demo").innerHTML = points; // 100,40,25,10,5,1
}
```

The Compare Function

비교 함수의 목적은 대체할 수 있는 정렬 순서를 정의하는 것입니다.

비교 함수는 인수에 따라 음수, 0 또는 양수 값을 반환해야 합니다.

```
function (a, b) {return a-b}
```

sort() 함수는 두 값을 비교할 때, 값을 비교 함수로 보내고 반환된 값 (음수, 0, 양수)에 따라 값을 정렬합니다.

결과가 음수이면, a가 b보다 먼저 정렬됩니다.

결과가 양수이면, b가 a보다 먼저 정렬됩니다.

결과가 0이면, 두 값의 정렬 순서가 변경되지 않습니다.

예제

비교 함수는 배열의 모든 값을 한 번에 두 개의 값 (a, b)으로 비교합니다.

40과 100을 비교할 때, sort() 메서드는 compare function (40, 100)을 호출합니다.

이 함수는 40-100 (a-b)을 계산하고, 결과가 음수 (-60)이므로 정렬 함수는 40을 100보다 낮은 값으로 정렬합니다.

이 코드 스니펫을 사용하여 숫자 및 알파벳 정렬을 실험할 수 있습니다.

```
let points = [40, 100, 1, 5, 25, 10];
document.getElementById("demo").innerHTML = points;   // 40,100,1,5,25,10

function myFunction1() {
  points.sort();
  document.getElementById("demo").innerHTML = points; // 1,10,100,25,40,5
}
  function myFunction2() {
  points.sort(function (a, b) { return a - b });
  document.getElementById("demo").innerHTML = points; // 1,5,10,25,40,100
}
```

오션, Javascript, Sorting, Arrays, box-sizing

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:javascript:javascript_note:js_array_sort&rev=1619501502

Last update: **2022/03/10 19:52**