<https://medium.com/@leokao0726/%E6%B7%BA%E8%AB%87-js-sort-%E5%88%B0%E8%83%8C%E5%BE%8C%E6%8E%92%E5%BA%8F%E6%96%B9%E6%B3%95-1035f5b8cde8>

**sort 背後的排序方法**

Chrome V8 引擎的 source code 所實作背後的運作原理：

function InnerArraySort(array, length, comparefn) {  
 *// In-place QuickSort algorithm.*  
 *// For short (length <= 10) arrays, insertion sort is used for efficiency.*  
 if (!IS\_CALLABLE(comparefn)) {  
 comparefn = function (x, y) {  
 if (x === y) return 0;  
 if (%\_IsSmi(x) && %\_IsSmi(y)) {  
 return %SmiLexicographicCompare(x, y);  
 }  
 x = TO\_STRING(x);  
 y = TO\_STRING(y);  
 if (x == y) return 0;  
 else return x < y ? -1 : 1;  
 };  
}

註釋寫著他是**用「Quick Sort」，而當陣列長度小於 10 時，為了提升效能會改用「Insertion Sort」**。

如果不懂 Quick Sort 和 Insertion Sort ，可以參考以下連結，影音並茂的方式說明十分明瞭：

Quick Sort : [點我](https://www.youtube.com/watch?v=5nXrEBhBFpU) | Insertion Sort : [點我](https://www.youtube.com/watch?v=DfloPvgptJA)

// 自定義排序 (Custom Sort)

// sort() 可以傳入函數參數 compareFunction，可以用來自訂排序的邏輯，陣列會根據 compareFunction 函數的回傳值來做排序依據。

// compareFunction(a, b) 函數接受兩個參數，分別表示兩個元素值怎麼做比較，然後傳回一個數字，可能是正數、0 或負數：

// a=50,b=100

// compareFunction(a, b) 回傳值如果小於 0 (負數)，表示 a 排序在 b 前面

// compareFunction(a, b) 回傳值如果等於 0，表示 a 和 b 排序一樣位置不動

// compareFunction(a, b) 回傳值如果大於 0 (正數)，表示 b 排序在 a 前面

//ECMAscript 標準沒規範當 0 相等的時候，哪個元素排先哪個元素排後，每個瀏覽器的實做可能會不一樣。