

Q: Sort an array of strings based on their lengths in ascending order. If two strings have the same length, maintain their relative order in the sorted array.

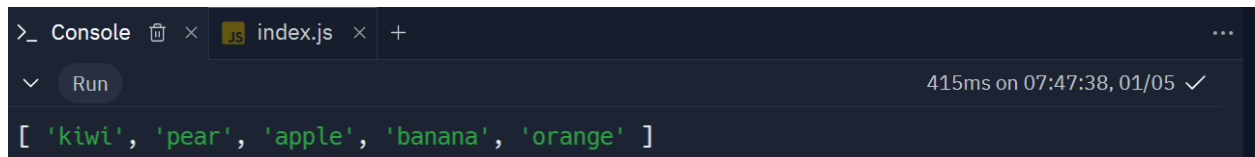
```
const stringArray = ["apple", "banana", "kiwi", "orange", "pear"];

// Custom comparison function for sorting based on lengths
const sortByLength = (a, b) => {
  // Compare lengths
  const lengthComparison = a.length - b.length;

  // If lengths are equal, maintain relative order
  return lengthComparison === 0 ? stringArray.indexOf(a) - stringArray.indexOf(b) :
lengthComparison;
};

// Sort the array based on lengths
const sortedArray = stringArray.sort(sortByLength);

console.log(sortedArray);
```

A screenshot of a web browser's developer console. The top bar shows a 'Console' tab with a trash icon and a file named 'index.js'. Below the bar, there is a 'Run' button and a timestamp '415ms on 07:47:38, 01/05' with a checkmark. The console output is an array of strings: ['kiwi', 'pear', 'apple', 'banana', 'orange'], which are sorted by length in ascending order while maintaining their original relative order.

```
>_ Console × JS index.js × + ...
✓ Run 415ms on 07:47:38, 01/05 ✓
[ 'kiwi', 'pear', 'apple', 'banana', 'orange' ]
```