5472. Shuffle String

My Submissions (/contest/weekly-contest-199/problems/shuffle-string/submissions/)

Back to Contest (/contest/weekly-contest-199/)

Given a string `s` and an integer array `indices` of the **same length**.

The string `s` will be shuffled such that the character at the `i`th position moves to `indices[i]` in the shuffled string.
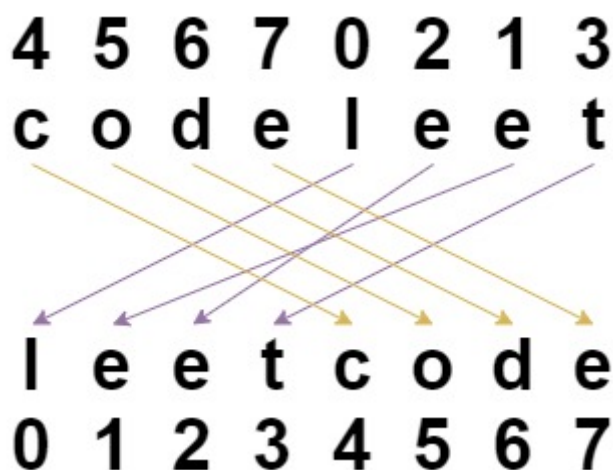
Return *the shuffled string*.

User Accepted:	0
User Tried:	0
Total Accepted:	0
Total Submissions:	0
Difficulty:	Easy

Example 1:


```
Input: s = "codeleet", indices = [4,5,6,7,0,2,1,3]
Output: "leetcode"
Explanation: As shown, "codeleet" becomes "leetcode" after shuffling.
```

Example 2:

```
Input: s = "abc", indices = [0,1,2]
Output: "abc"
Explanation: After shuffling, each character remains in its position.
```

Example 3:

```
Input: s = "aiohn", indices = [3,1,4,2,0]
Output: "nihao"
```

Example 4:

```
Input: s = "aaiougrt", indices = [4,0,2,6,7,3,1,5]
Output: "arigatou"
```

Example 5:

```
Input: s = "art", indices = [1,0,2]
Output: "rat"
```

Constraints:

- s.length == indices.length == n
- 1 <= n <= 100
- s contains only lower-case English letters.
- 0 <= indices[i] < n
- All values of indices are unique (i.e. indices is a permutation of the integers from 0 to n − 1).

| JavaScript ▾ | | ⟨⟩ | ⟳ | ⚙ |

```javascript
 1 ▾ /**
 2     * @param {string} s
 3     * @param {number[]} indices
 4     * @return {string}
 5     */
 6 ▾ const restoreString = (s, indices) => {
 7         let data = [];
 8 ▾      for (let i = 0; i < s.length; i++) {
 9             data.push([s[i], indices[i]]);
10         }
11         data.sort((a, b) => a[1] - b[1]);
12         let res = '';
13 ▾      for (const d of data) {
14             res+=d[0];
15         }
16         return res;
17 };
```

☐ **Custom Testcase**   (Use Example Testcases)

▶ Run   ☁ Submit

Submission Result: Accepted (/submissions/detail/371667398/) ❓

More Details ❯ (/submissions/detail/371667398/)

Share your acceptance!