

Problem 2677: Chunk Array

Problem Information

Difficulty: Easy

Acceptance Rate: 84.51%

Paid Only: No

Problem Description

Given an array `arr` and a chunk size `size`, return a **chunked** array.

A **chunked** array contains the original elements in `arr`, but consists of subarrays each of length `size`. The length of the last subarray may be less than `size` if `arr.length` is not evenly divisible by `size`.

Please solve it without using lodash's `_.chunk` function.

Example 1:

Input: `arr = [1,2,3,4,5]`, `size = 1` **Output:** `[[1],[2],[3],[4],[5]]` **Explanation:** The `arr` has been split into subarrays each with 1 element.

Example 2:

Input: `arr = [1,9,6,3,2]`, `size = 3` **Output:** `[[1,9,6],[3,2]]` **Explanation:** The `arr` has been split into subarrays with 3 elements. However, only two elements are left for the 2nd subarray.

Example 3:

Input: `arr = [8,5,3,2,6]`, `size = 6` **Output:** `[[8,5,3,2,6]]` **Explanation:** Size is greater than `arr.length` thus all elements are in the first subarray.

Example 4:

Input: `arr = []`, `size = 1` **Output:** `[]` **Explanation:** There are no elements to be chunked so an empty array is returned.

****Constraints:****

* `arr` is a string representing the array. * `2 <= arr.length <= 105` * `1 <= size <= arr.length + 1`

Code Snippets

JavaScript:

```
/**
 * @param {Array} arr
 * @param {number} size
 * @return {Array}
 */
var chunk = function(arr, size) {

};
```

TypeScript:

```
type JSONValue = null | boolean | number | string | JSONValue[] | { [key:
string]: JSONValue };
type Obj = Record<string, JSONValue> | Array<JSONValue>;

function chunk(arr: Obj[], size: number): Obj[][] {

};
```