Challenges 03 Arrays 💩 = [🍔, 🍟, 🍪].reduce(eat)

multiply(arr)

write a function named multiply that accepts an array and returns the total of all the numbers in the array multiplied together. 🎁🎁🎁🎁🎁🎁🎁 Bonus -- works on a nested array

includesCopy(arr, searchValue)

write a function named includesCopy that accepts an array and a search value. Returns true if the array includes the given value. else returns false

inventory(arr)

Define a function, inventory, that accepts a multi-dimensional array of items. inventory should return a new, flat array. Each element in the new array should be a sentence about each of the items. [["chair", ["comfortable", 100]]] => "The comfortable chair is 100 dollars."

camelCase(str)

write a function named camelCase that takes a string as a parameter and returns the camelcased version of that string

joiner()

Define a function, joiner, that accepts up to two arguments:

```
// array
// seperator (string, optional)
// joiner should return a string with all of the elements from the array joined
together. The separator should separate the joined elements:

// joiner(['let's', 'make', 'a', 'list'], ' '); // => "let's make  a list"
// joiner(['a', 'b', 'c'], '+'); // => "a+b+c"

// If separator is undefined, use ',' as the default separator.

// joiner(['Peter', 'Paul', 'Mary']); // => "Peter,Paul,Mary"
```