

```
what's wrong with this js code?  const numString =  
document.querySelector('input').value  
const numArray = numString.split(":")  
for (let num of numArray) {  
  num = parseInt(num)  
}
```



Your JavaScript code has a **scope issue** inside the `for...of` loop. Here's the problem:

Issue: Reassigning `num` Doesn't Change the Array

When you do:

```
js  
  
for (let num of numArray) { num = parseInt(num); }
```



You're modifying the loop variable `num`, but this doesn't change the actual values inside `numArray`. JavaScript passes primitive values (like strings) **by value**, so `num` is just a copy of each array element, and reassigning it doesn't update the array.

Fix:

You need to update the array elements directly using `.map()` or a `for` loop with indexing:

✅ Using `.map()` (Preferred)

```
js  
  
const numString = document.querySelector('input').value; const numArray =  
numString.split(":").map(num => parseInt(num, 10));
```



✅ Using a `for` Loop with Index

```
js  
  
const numString = document.querySelector('input').value; const numArray =  
numString.split(":"); for (let i = 0; i < numArray.length; i++) {  
  numArray[i] = parseInt(numArray[i], 10); }
```



Now, `numArray` will contain actual numbers instead of strings.

Extra Fix: Ensure `input` Exists

Make sure `document.querySelector('input')` actually exists before accessing `.value`, or else you may get a `null` error. You can add a safety check:

```
js  
  
const inputElement = document.querySelector('input'); if (inputElement) {  
  const numArray = inputElement.value.split(":").map(num => parseInt(num,
```



```
10)); }
```

Let me know if you need further clarification! 🚀