

# JavaScript vs jQuery Parameter Order Cheat Sheet

---

## QUICK REFERENCE

---

Vanilla JavaScript: (ELEMENT, INDEX)

---

jQuery: (INDEX, ELEMENT)

---

## METHOD COMPARISON TABLE

---

Method	Parameter Order	Example
Vanilla JS	(element, index)	<code>array.forEach((element, index) =&gt; {})</code>
jQuery .each()	(index, element)	<code>\$('#div').each((index, element) =&gt; {})</code>
jQuery \$.each()	(index, element)	<code>\$.each(array, (index, element) =&gt; {})</code>

## MNEMONICS TO REMEMBER

---

Vanilla JS: "What, Where"

---

- **What** (element) first
- **Where** (index) it is

jQuery: "Position, Thing"

---

- **Position** (index) first
- **Thing** (element) at that position




## VISUAL MEMORY AIDS

---

Vanilla JavaScript Pattern:

---

text

```
[ ,  ,  ].forEach( (FRUIT, POSITION) => ... )
```

↑            ↑  
element    index

jQuery Pattern:

---

text

```
$('.fruit').each( (POSITION, FRUIT) => ... )
           ↑           ↑
           index      element
```

## SIDE-BY-SIDE EXAMPLES

---

### Processing an Array:

---

javascript

```
// VANILLA JS - Element first
['a', 'b', 'c'].forEach((element, index) => {
    console.log(element, index); // 'a' 0, 'b' 1, 'c' 2
});

// JQUERY - Index first
$.each(['a', 'b', 'c'], (index, element) => {
    console.log(index, element); // 0 'a', 1 'b', 2 'c'
});
```

### Processing DOM Elements:

---

javascript

```
// VANILLA JS - Element first
document.querySelectorAll('div').forEach((element, index) => {
    console.log(element, index); // <div> 0, <div> 1
});

// JQUERY - Index first
$('div').each((index, element) => {
    console.log(index, element); // 0 <div>, 1 <div>
});
```

## COMMON PITFALLS & FIXES

---

### Wrong (Mixing Patterns):

---

javascript

```
// ❌ DON'T - Using jQuery order in vanilla JS
array.forEach((index, element) => { /* WRONG! */ });

// ❌ DON'T - Using vanilla order in jQuery
$('div').each((element, index) => { /* WRONG! */ });
```

### Right (Correct Patterns):

---

javascript

```
// ✅ DO - Vanilla JS order
array.forEach((element, index) => {
  console.log(`Element ${element} at position ${index}`);
});

// ✅ DO - jQuery order
$('div').each((index, element) => {
  console.log(`Element at position ${index}:`, element);
});
```

## QUICK CONVERSION GUIDE

---

### Going from jQuery to Vanilla:

---

javascript

```
// jQuery
$.each(array, function(index, element) { });

// Convert to Vanilla
array.forEach(function(element, index) { });
//           ↑           ↑
//           SWAPPED! SWAPPED!
```

### Going from Vanilla to jQuery:

---

javascript

```
// Vanilla
array.forEach(function(element, index) { });

// Convert to jQuery
$.each(array, function(index, element) { });
//           ↑           ↑
//           SWAPPED! SWAPPED!
```

## QUICK CHECK QUESTIONS

---

When you see a callback function, ask:

1. Am I using jQuery or vanilla JavaScript?
2. If jQuery → (index, element)
3. If vanilla → (element, index)

## COLOR-CODED REMINDER

---

<div style="color: blue; font-weight: bold;"> Vanilla JS: (ELEMENT, INDEX) </div><div style="color: green; font-weight: bold;"> jQuery: (INDEX, ELEMENT) </div>

**Print this and keep it by your desk! The inconsistency is tricky but becomes automatic with practice.**