Practice Exercises: Adding Elements

Each exercise below is designed to help you master different aspects of using splice() for adding elements. Try to solve them yourself before looking at the solutions!

1. Add 'Wednesday' between 'Tuesday' and 'Thursday':

javascript
Copy
```
let days = ['Monday', 'Tuesday', 'Thursday', 'Friday'];
```

```
// Your code here
```

Problem Analysis:

- Current array has a missing day (Wednesday)
- Need to insert between 'Tuesday' and 'Thursday'
- 'Tuesday' is at index 1, so we want to insert at index 2
- We don't want to remove any elements, just insert

Strategy:

- Use splice(2, 0, 'Wednesday')
    - 2: insert after 'Tuesday'
    - 0: don't remove any elements
    - 'Wednesday': the element to insert

2. Insert three numbers (4, 5, 6) after index 3:

javascript
Copy
```
let sequence = [1, 2, 3, 7, 8, 9];
```

```
// Your code here
```

Problem Analysis:

- We need to insert multiple elements (4, 5, 6)
- They should go after 3 (index 2)
- Current sequence jumps from 3 to 7
- We want to maintain numerical order

Strategy:

- Use splice(3, 0, 4, 5, 6)
    - 3: position after the number 3

    - 0: don't remove any elements
    - 4, 5, 6: multiple elements to insert

3. Add 'Spring' at the beginning of the seasons array:

javascript
Copy
```
let seasons = ['Summer', 'Fall', 'Winter'];
```

```
// Your code here
```

Problem Analysis:

- Need to add at the start of the array
- First season (Spring) is missing
- No elements should be removed
- Want to maintain seasonal order

Strategy:

- Use splice(0, 0, 'Spring')
    - 0: insert at the beginning
    - 0: don't remove any elements
    - 'Spring': element to insert

4. Insert 'purple' and 'orange' between 'blue' and 'green':

javascript
Copy
```
let rainbow = ['red', 'blue', 'green', 'yellow'];
```

```
// Your code here
```

Problem Analysis:

- Need to insert two colors at once
- Insert point is between 'blue' and 'green'
- 'blue' is at index 1
- No elements should be removed

Strategy:

- Use splice(2, 0, 'purple', 'orange')
    - 2: position after 'blue'
    - 0: don't remove any elements
    - 'purple', 'orange': multiple elements to insert

5. Add 'guitar' at index 2:

javascript
Copy
```
let instruments = ['piano', 'violin', 'drums', 'flute'];
```

```
// Your code here
```

Problem Analysis:

- Need to insert at a specific index (2)
- Current element at index 2 ('drums') should move right
- No elements should be removed
- Just inserting one element

Strategy:

- Use splice(2, 0, 'guitar')
    - 2: specified index position
    - 0: don't remove any elements
    - 'guitar': element to insert

Practice Exercises: Removing Elements

These exercises focus on different ways to remove elements using splice(). Each one demonstrates a unique aspect of the method's capabilities.

1. Remove two elements starting from index 1:

javascript
Copy
```
let fruits = ['apple', 'banana', 'orange', 'grape', 'kiwi'];
```

```
// Your code here
```

Problem Analysis:

- Need to remove 'banana' and 'orange'
- Starting point is index 1 ('banana')
- Want to remove exactly 2 elements
- Remaining elements should shift left

Strategy:

- Use splice(1, 2)
    - 1: start at 'banana'
    - 2: remove two elements

- Returns: ['banana', 'orange']

2. Remove the last three elements:

javascript
Copy

```
let numbers = [1, 2, 3, 4, 5, 6, 7];
```

```
// Your code here
```

Problem Analysis:

- Need to remove last three numbers (5, 6, 7)
- Can use negative index or length-based index
- Want to keep first four elements
- No new elements to add

Strategy:

- Use splice(-3) or splice(4, 3)
  - -3: start 3 from end OR 4: start at index 4
  - 3: remove three elements
  - Returns: [5, 6, 7]

3. Remove the first element:

javascript
Copy

```
let colors = ['red', 'blue', 'green', 'yellow'];
```

```
// Your code here
```

Problem Analysis:

- Need to remove just the first element ('red')
- Similar to shift() but using splice()
- Remaining elements shift left
- No new elements to add

Strategy:

- Use splice(0, 1)
  - 0: start at beginning
  - 1: remove one element
  - Returns: ['red']

4. Remove all elements after index 2:

javascript
Copy
```
let planets = ['Mercury', 'Venus', 'Earth', 'Mars', 'Jupiter'];
```

```
// Your code here
```

Problem Analysis:

- Keep first three planets
- Remove everything after 'Earth'
- Index 2 is 'Earth'
- Can use splice without length parameter

Strategy:

- Use splice(3)
    - 3: start after 'Earth'
    - No second parameter: remove all remaining elements
    - Returns: ['Mars', 'Jupiter']

5. Remove one element at index 3:

javascript
Copy
```
let names = ['John', 'Jane', 'Bob', 'Alice', 'Tom'];
```

```
// Your code here
```

Problem Analysis:

- Need to remove 'Alice' at index 3
- Only remove one element
- Remaining elements shift left
- No new elements to add

Strategy:

- Use splice(3, 1)
    - 3: start at 'Alice'
    - 1: remove one element
    - Returns: ['Alice']

Solutions

Adding Elements Solutions:

Days solution:
javascript
Copy
```
days.splice(2, 0, 'Wednesday');
```

1. `// Result: ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday']`

Sequence solution:
javascript
Copy
```
sequence.splice(3, 0, 4, 5, 6);
```

2. `// Result: [1, 2, 3, 4, 5, 6, 7, 8, 9]`

Seasons solution:
javascript
Copy
```
seasons.splice(0, 0, 'Spring');
```

3. `// Result: ['Spring', 'Summer', 'Fall', 'Winter']`

Rainbow solution:
javascript
Copy
```
rainbow.splice(2, 0, 'purple', 'orange');
```

4. `// Result: ['red', 'blue', 'purple', 'orange', 'green', 'yellow']`

Instruments solution:
javascript
Copy
```
instruments.splice(2, 0, 'guitar');
```

5. `// Result: ['piano', 'violin', 'guitar', 'drums', 'flute']`

Removing Elements Solutions:

Fruits solution:
javascript
Copy
```
fruits.splice(1, 2);
```

1. `// Result: ['apple', 'grape', 'kiwi']`

Numbers solution:
javascript
Copy
```
numbers.splice(-3);
```

2. `// Result: [1, 2, 3, 4]`

Colors solution:
javascript
Copy
```
colors.splice(0, 1);
```

3. `// Result: ['blue', 'green', 'yellow']`

Planets solution:
javascript
Copy
```
planets.splice(2);
```

4. `// Result: ['Mercury', 'Venus', 'Earth']`

Names solution:
javascript
Copy
```
names.splice(3, 1);
```

5. `// Result: ['John', 'Jane', 'Bob', 'Tom']`

Tips for Practice

1. Always make a copy of your array before practicing splice() operations
2. Remember that splice() modifies the original array
3. Practice with both positive and negative indices
4. Try combining adding and removing operations in a single splice() call