

Front-end

JavaScript

JS. Arrays

- with
- toSorted
- toSpliced
- toReversed

JS. Arrays

Метод **with** является эквивалентом копирования использования квадратных скобок для изменения одного элемента массива. Таким образом, вместо прямого изменения массива, как показано ниже:

```
const languages = ['Russian', 'English', 'Deutsch'];  
languages[2] = 'Chinese';  
console.log(languages); // ['Russian', 'English', 'Chinese']
```

Вы можете скопировать массив и внести изменение:

```
const languages = ['Russian', 'English', 'Deutsch'];  
const updated = languages.with(2, 'Chinese');  
console.log(updated); // ['Russian', 'English', 'Chinese']  
console.log(languages); // ['Russian', 'English', 'Deutsch']
```


JS. Arrays

Метод **toSorted** возвращает новый отсортированный массив.

```
const languages = ['Russian', 'English', 'Deutsch'];  
const sorted = languages.toSorted();  
console.log(sorted); // => [ 'Deutsch', 'English', 'Russian' ]  
console.log(languages); // => [ 'Russian', 'English', 'Deutsch' ]  
console.log(Object.is(languages, sorted)); // false
```


JS. Arrays

Метод **toReversed()** меняет порядок элементов в массиве на обратный, как и метод `reverse()`, но возвращает новый массив

```
const languages = ['Russian', 'English', 'Deutsch'];  
const reversed = languages.toReversed();  
console.log(reversed); // ['Deutsch', 'English', 'Russian']  
console.log(Object.is(languages, reversed)); // false
```

Теперь вы можете использовать `toReversed` или `toSorted`, чтобы скопировать массив и изменять копию вместо исходного.

JS. Arrays

Метод **toSpliced** немного отличается от своей исходной версии splice.

Метод splice изменяет существующий массив путем удаления и добавления элементов по указанному индексу и возвращает массив, содержащий удаленные элементы из массива.

toSpliced возвращает новый массив без удаленных элементов и с добавленными элементами. Вот как это работает:

```
const languages = ['Russian', 'English', 'Deutsch'];  
const spliced = languages.toSpliced(2, 1, 'Chinese', 'Spanish');  
console.log(spliced); // ['Russian', 'English', 'Chinese', 'Spanish']  
console.log(Object.is(languages, spliced)); // false
```


JS. Arrays

Если вы используете `splice` из-за его возвращаемого значения, то `toSpliced` не будет полностью совместимым заменителем. Если вы хотите получить удаленные элементы без изменения исходного массива, то вам следует использовать метод `slice`.

К сожалению, у `splice` и `slice` разные аргументы.

`splice` принимает индекс и количество элементов после этого индекса, которые нужно удалить.

`slice` принимает два индекса - начальный и конечный. Если вы хотите использовать `toSpliced` вместо `splice`, но также получить удаленные элементы, вы можете применить `toSpliced` и `slice` к исходному массиву, вот так:

```
const languages = ['Russian', 'English', 'Deutsch'];  
  
const startDeletingAt = 2;  
  
const deleteCount = 1;  
  
const spliced = languages.toSpliced(startDeletingAt, deleteCount, 'Chinese', 'Spanish');  
  
const removed = languages.slice(startDeletingAt, startDeletingAt + deleteCount);  
  
console.log(spliced); // ['Russian', 'English', 'Chinese', 'Spanish']  
  
console.log(removed); // ['Deutsch']
```