.splice()

Array Method in JavaScript

Edition Series

Cute Kitty CheatSheets

Second Edition

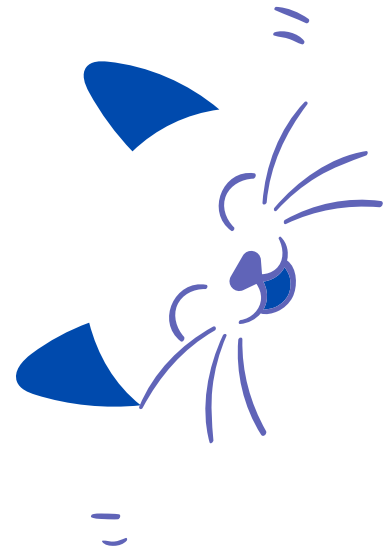
PART I

MUTATING methods

Follow us meow! /kittycat kittycat-tech

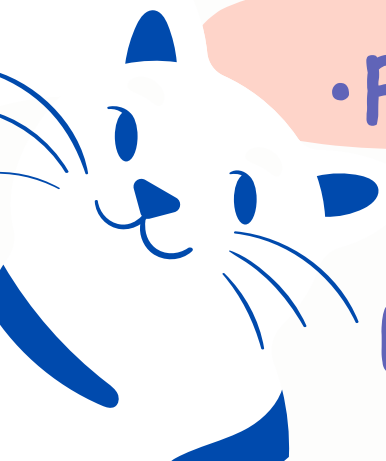
Part 1

Mutating



They **Mutate** the **original array right away** when executed

- .copyWithin()
- .reverse()
- .splice()
- .sort()
- .fill()
- .unshift()
- .pop()
- .shift()
- .push()



Follow us meow! in/kittycat

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice method

MUTATING method

For Deleting Elements

Using 1 argument only:
We use this to **delete all** the pre-existing elements, starting from the index you provided.

Arguments

1ST

First Argument is the **index** of the element, that you want to be your **Start Index** of deletion. All other indexes after this index will be deleted together with their values.

```
let MeowArray = ["Coding","with","Meow", "is", "fun!","Cutie","Kittycat!"]
MeowArray.splice(1)
console.log(MeowArray)
```

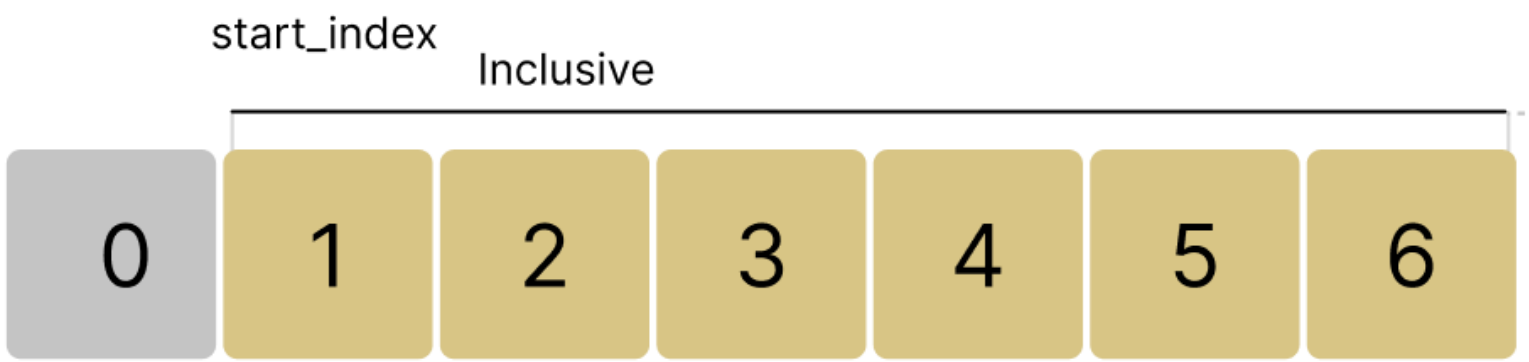
```
▶ ['Coding']
```

Follow us meow! /kittycat kittycat-tech

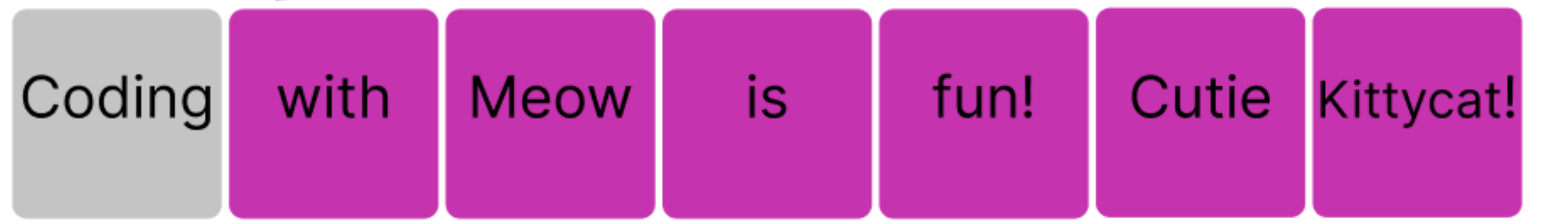
References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub || Marouane Souda

.splice(1)

.splice(start_index)



**From here,
Delete all elements**



deleted!

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice method

MUTATING method

For Deleting Elements

Using 2 arguments:
We use this to **delete** **specific** pre-existing elements

Arguments

1ST

First Argument is the **index** of the element, that you want to access **for deletion.**
In case you would want to delete multiple indexes, it will be your **Start Index** amongst the other indexes.

2ND

From your Start Index, how many indexes do you want to delete in total?
That will be your **count** for your **Second Argument.**

```
let MeowArray = ["Coding","with","Meow", "is", "fun!","Cutie","Kittycat!"]
MeowArray.splice(3,2)
console.log(MeowArray)
```

```
▶ (5) ['Coding', 'with', 'Meow', 'Cutie', 'Kittycat!']
```

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice(3,1)

.splice(start_index,count)

start_index

Inclusive

0	1	2	3	4	5	6

Delete this one only

Coding	with	Meow	is	fun!	Cutie	Kittycat!

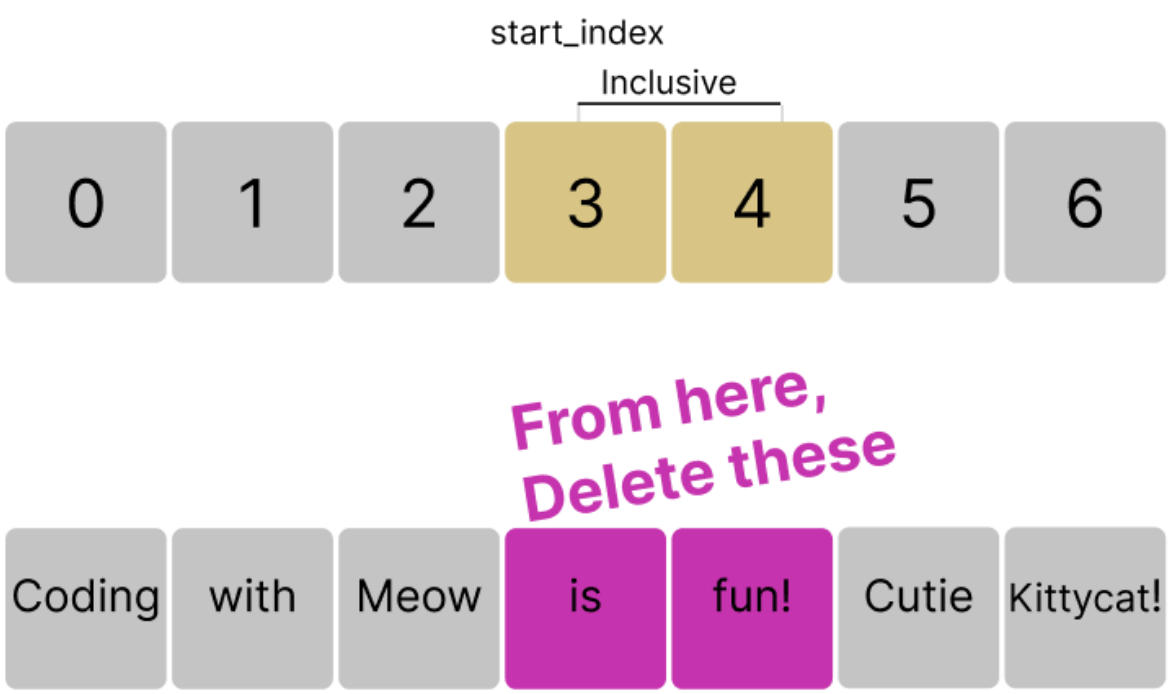
Coding	with	Meow	fun!	Cutie	Kittycat!

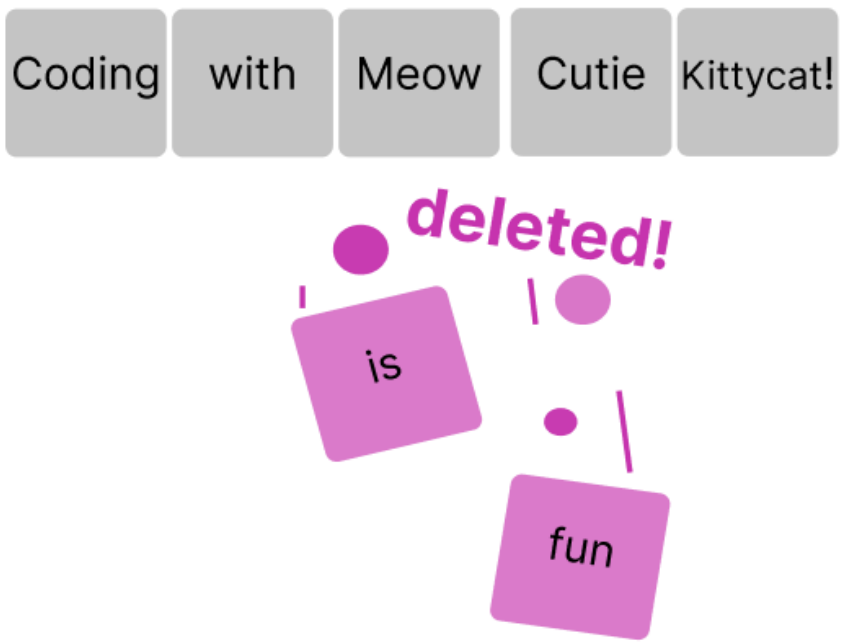
Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice(3,2)

.splice(start_index, count)





Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice method

MUTATING method

For Adding New Elements

3, 4, or more arguments:

We can use splice to **add** element/s **in the middle** or **anywhere** in the array. The new element/s just takes the index position and push the other pre-existing elements to the right.

You can add as much as you want using 3rd, 4th arguments and so fort.

```
> let MeowArray = ["Coding","with","Meow", "is", "fun!",]
  MeowArray.splice(4,0,"ultra","super")
  console.log(MeowArray)
  ▶ (7) ['Coding', 'with', 'Meow', 'is', 'ultra', 'super', 'fun!']
```

Next Page: Arguments explanation

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

Arguments: .splice(1st, 2nd, 3rd, 4th,..)

For Adding New Elements

1ST

First Argument is the **index** of the element, that you want to access **for adding/inserting new element.**

In case you would want to add more elements, it will be your **Start Index** where to start adding new elements.

2ND

From your Start Index, how many indexes do you want to delete in total?

Since you don't want to delete any elements in this case, Zero will be you **count** for your **Second Argument.**

3RD

Third Argument is **the element,** that you want to **ADD or Insert.**

This new element will be **inserted** on the **Start Index** of your first Argument. **Nothing gets deleted** in this process.

4TH

+

You can add your **4th argument** or more arguments as much as you want.

Remember that these will be the **new elements** that you want to **add in addition** to third Argument.

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice(4,0,"ultra","super")

.splice(start_index, count, Add_new, Add_new)

start_index

0	1	2	3	4
Coding	with	Meow	is	fun!

ADD new elements

Coding	with	Meow	is	ultra	super

added!

Coding	with	Meow	is	ultra	super	fun!

new length: 7

0	1	2	3	4	5	6

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice method

MUTATING method

For Deleting then Adding New Elements

3, 4, or more arguments:

We use this to **remove** the specific existing elements **then add new** elements declared in Argument 3 and so fort.

```
let MeowArray = ["Coding","with","Meow", "is", "fun!","Cutie","Kittycat!"]
MeowArray.splice(4,3,"full","of","cuteness","meow")
console.log(MeowArray)
```

```
▶ (8) ['Coding', 'with', 'Meow', 'is', 'full', 'of', 'cuteness', 'meow']
```

Next Page: Arguments explanation

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

Arguments: .splice(1st, 2nd, 3rd, 4th,..)

For Deleting then Adding New Elements

1ST

First Argument is the **index** of the element, that you want to access **for deletion.**
In case you would want to delete multiple indexes, it will be your **Start Index** amongst the other indexes.

2ND

From your Start Index, how many indexes do you want to delete in total?
That will be your **count** for your **Second Argument.**

3RD

Third Argument is **the element,** that you want to **ADD or Insert** to replace the removed element/s.
This new element will be **inserted** on the **Start Index** of your first Argument.

4TH

+

You can add your 4th argument or more arguments as much as you want.
Remember that these will be the **new elements** that you want to **add in addition** to third Argument.

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

.splice(4,3,"full","of","cuteness","meow")

.splice(start_index, count, Add_new, Add_new..)

start_index Inclusive

0	1	2	3	4	5	6

From here,
Delete these 3

Coding	with	Meow	is	fun!	Cutie	Kittycat!

Coding	with	Meow	is

then ADD new elements

Coding	with	Meow	is	full	of	cuteness	meow

added!

new length: 8

0	1	2	3	4	5	6	7

Follow us meow! /kittycat kittycat-tech

References: https://developer.mozilla.org/ || MIT xPro Full Stack Course Modules, Prof. Zahid || https://betterprogramming.pub

Follow us meow! /kittycat kittycat-tech

What do Edition Series mean, meow?

First Edition

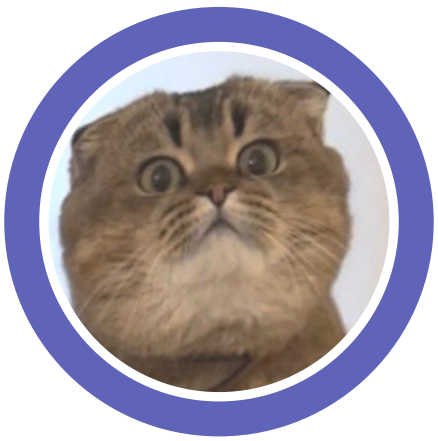
The most basic version for newborn babies in programming.

The goal of this edition is for the newbies to have a smooth onboarding and to digest easier the basic concepts.

Second Edition (We are here, meow)

This is the material is for those who have already grasped their knowledge and just got on board for programming.

It consists of basics and a bit more foundation than First Edition, usually with a step by step thinking method.



Third Edition

This is for a bit advanced version, for beginner level.
Usually a combination of multiple topics,
a deeper understanding of the topics together
and a real-life usage of the concepts.

Website: kittycat.tech (coming soon)

Hi amazing you!
Let's be friends!
Add Meow!

/kittycat

kittycat-tech

Sweet and friendly

(I code randomly with paws)

Website: kittycat.tech (coming soon)