



TRUNG TÂM HUẤN LUYỆN LẬP TRÌNH  
KHAI GIẢNG KHÓA HỌC LẬP TRÌNH **PHP** - LẬP TRÌNH **RUBY**



Bui Thai Ha @habt42

Follow

★ 433 👤 8 📝 24

Published Feb 22nd, 10:50 AM - 3 min read

TABLE OF CONTENTS

👁 456 💬 0 🔗 1

Slice

Splice

Spread syntax

Kết luận:

SUGGESTED ORGANIZATIONS

...



PHP Education Team

📝 12 👤 40 📑 153



Mino

📝 2 👤 2 📑 8



Decipher Zone Softwares

📝 0 👤 0 📑 4



JavaScript



# Cùng tìm hiểu về slice, splice và cú spread syntax trong Javascript

Bài viết đc dịch lại từ: <https://medium.freecodecamp.org/lets-explore-slice-splice-spread-syntax-in-javascript-e242a6f21e60>

Cả slice và splice đều dùng để thao tác với mảng, hãy cùng tìm hiểu xem chúng hoạt động thế nào nhé.



## Slice

Hàm slice nhận vào 2 tham số **Tham số thứ 1**: là vị trí bắt đầu lựa chọn. Ví dụ

```
var arr1 = [1,5,8,9];  
arr1.slice(1); // [5,8,9]
```

Nó sẽ trả về các phần tử từ vị trí thứ nhất ( số 5 ) **Tham số thứ 2**: là vị trí kết thúc lựa chọn. Nếu bạn ko truyền tham số này, nó sẽ trả về các phần tử từ vị trí bắt đầu đến vị trí cuối cùng của mảng.

```
var arr1 = [1,5,8,9];  
arr1.slice(1,3);
```

```
console.log(arr1.slice(1,3));  
//[ 5, 8 ]
```

Nếu bạn truyền vào một số âm, vị trí sẽ đc đếm từ cuối mảng đổ xuống

```
var arr1 = [1,5,8,9];  
console.log(arr1.slice(-2));
```

VIBLO

Posts

Questions

Discussions

Search Viblo



→ Sign In/Sign up

**Lưu ý: Slice luôn trả về các phần tử đc lựa chọn từ mảng và ko làm thay đổi mảng đó.**



+3



```
var arr1 = [1,5,8,9];  
arr1.slice(2);  
console.log(arr1);  
// [ 1, 5, 8, 9 ]
```



## Splice

Hàm splice có thể nhận nhiều tham số cùng lúc.

**Tham số thứ 1:** vị trí mà phần tử mới hoặc phần tử đã tồn tại sẽ đc thêm vào/xóa đi.

Nếu giá trị này là âm, vị trí đó sẽ đc đếm từ cuối mảng đổ xuống. **Tham số thứ 2:** Số các phần tử để xóa bỏ khỏi mảng bắt đầu từ vị trí ở trên. Nếu là 0 thì sẽ ko xóa phần tử nào cả, còn nếu ko đc truyền thì nó sẽ xóa tất cả các phần tử bắt đầu từ vị trí ở trên.

```
var arr1 = [1,5,8,9];  
console.log(arr1.splice(1,2));  
// [ 5, 8 ]
```

**Tham số thứ 3 đến thứ n:** Giá trị của phần tử mà bạn muốn thêm vào mảng

VIBLO

Posts

Questions

Discussions

Search Viblo



→ Sign In/Sign up

```
console.log(arr1.splice(1,2,'Hi','Medium'));  
// [5,8]
```



+3



**Lưu ý:**

- Splice chỉ trả về những phần tử đã đc xóa khỏi mảng.
- Splice sẽ làm thay đổi mảng gốc





```
var arr1 = [1,5,8,9];
arr1.splice(1,2,'Hi','Medium');
console.log(arr1);
// [ 1, 'Hi', 'Medium', 9 ]
```

## Spread syntax

Cho phép một đối tượng kiểu iterable như array epxression hoặc string có thể đc trải ra ở những chỗ mà có thể truyền 0 hoặc nhiều nhiều tham số ( lúc gọi hàm ) hoặc phần tử ( trong array ). Hoặc một object expression có thể đc trải ra để truyền 0 hoặc nhiều cặp key-value

Ví dụ:

VIBLO

Posts Questions Discussions

Search Viblo



Sign In/Sign up

```
console.log(arr2);
// [ 5, [ 1, 3, 6, 7 ], 8, 9 ]
```



+3



Ví dụ tôi muốn nó thành một mảng kiểu như [ 5, 1, 3, 6, 7, 8, 9 ] Tôi có thể dùng spread syntax để làm như sau:



```
var arr1 = [1,3,6,7];
var arr2 = [5,...arr1,8,9];
console.log(arr2);
//[ 5, 1, 3, 6, 7, 8, 9 ]
```

Một cách sử dụng rất thông dụng khác là để copy một mảng:

```
var arr = [1, 2, 3];
var arr2 = arr;
arr2.push(4);
console.log(arr2);
// [ 1, 2, 3, 4 ]
console.log(arr);
// [ 1, 2, 3, 4 ]
```

Tôi đã thêm 4 vào arr2, nhưng nó cũng khiến arr bị thay đổi theo. Chúng ta có thể sử dụng spread syntax để xử lý kiểu đã nêu

VIBLO

Posts Questions Discussions

Search Viblo



Sign In/Sign up

```
var arr = [1, 2, 3];
var arr2 = [...arr]; // giống arr.slice()
arr2.push(4);
```

+3



```
console.log(arr2);  
// [ 1, 2, 3, 4 ]  
console.log(arr);  
// [ 1, 2, 3]
```



## Kết luận:

Hàm slice sẽ:

- Trả về những phần tử đã đc chọn từ mảng
- Nhận vào 2 tham số
- Ko làm thay đổi mảng gốc

Hàm splice sẽ:

- Trả về những phần tử đã đc xóa khỏi mảng
- Nhận vào nhiều tham số
- Làm thay đổi mảng gốc

VIBLO

Posts

Questions

Discussions

Search Viblo



➔ Sign In/Sign up



### Related

ECMAScript - ES6 Là gì?  
Overview ES6

Trần Văn Mỹ

16 min read

👁 17031 🗨 38 💬 8 ⬆ 49

Trở thành Functional  
Programmer - Phần 4

Ta Duy Anh

17 min read

👁 1311 🗨 8 💬 0 ⬆ 4

mấy "lệnh" xử lí chuỗi mảng  
mà hay bị nhầm với nhau:...

Mòe

3 min read

👁 469 🗨 4 💬 0 ⬆ 6

Tìm Hiểu ECMAScript 6 - P2

Tran Dat

8 min read

👁 594 🗨 3 💬 1 ⬆ 4

### More from Bui Thai Ha

Tự build một thư viện  
validate form bất đồng bộ...

Tạo một drag and drop list  
trong React mà ko dùng bất...

React Hooks, tại sao chúng  
ta nên sử dụng chúng

Dependency Injection là gì,  
và khi nào thì nên sử dụng...

VIBLO

Posts

Questions

Discussions

Search Viblo



➔ Sign In/Sign up

### Comments


Login to comment

## RESOURCES

Posts  
Questions  
Videos  
Discussions  
Tools  
System Status

Organizations  
Tags  
Authors  
Recommend System  
Machine Learning

## SERVICES

 Viblo Code  
 Viblo CV

## MOBILE APP



## LINKS



© 2019 Viblo. All rights reserved.

[Feedback](#) [Help](#) [FAQs](#) [RSS](#) [Terms](#)

DMCA  PROTECTED

