

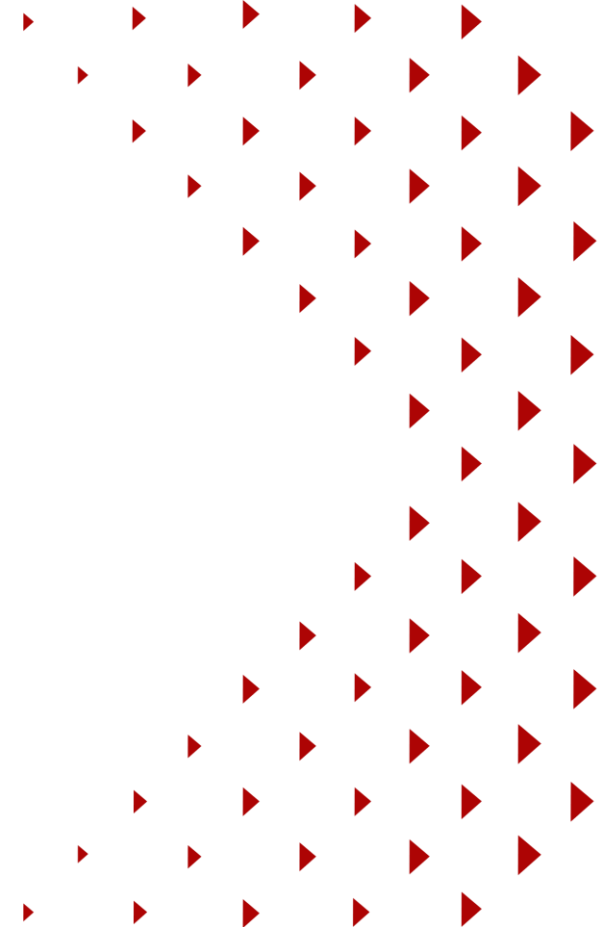


## SESSION 10:

# Array in Javascript

Module: Training Program Preparation

Version: 2.0

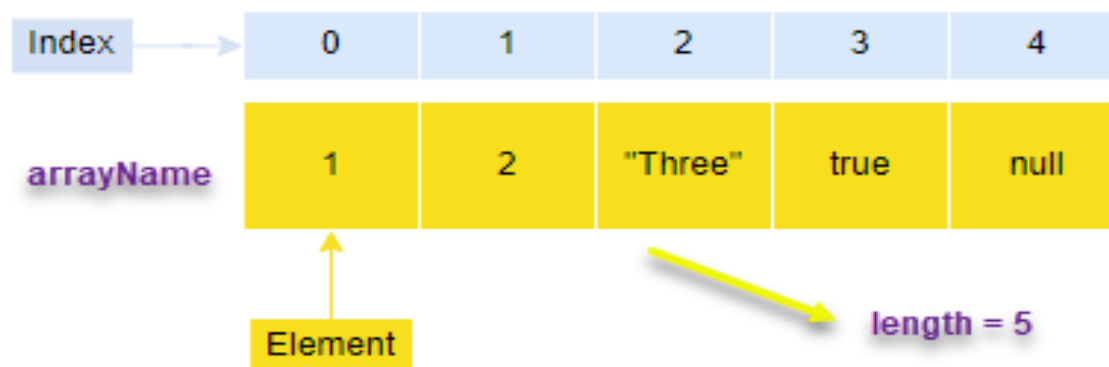


- 1. Array – Mảng**
- 2. Mảng một chiều**
- 3. Mảng nhiều chiều**
- 4. Các hàm làm việc**
- 5. Các hàm làm việc với mảng**
- 6. Duyệt và làm việc với các phần tử của mảng**

# ARRAY – MẢNG

- **Khái niệm**

- Trong JavaScript, array là một kiểu dữ liệu thuộc **kiểu tham chiếu** (reference type) dùng để lưu trữ nhiều phần tử trong một biến duy nhất.
- Array có thể chứa nhiều phần tử có kiểu dữ liệu khác nhau như **Number**, **String**, **Boolean**, **Object**, **null**,... hoặc thậm chí là những array
- Tổng số phần tử trong array được gọi là **length** của array. Các phần tử trong array được đánh chỉ mục **index** từ 0 đến **length - 1**
- Có 2 loại mảng:
  - Mảng một chiều
  - Mảng nhiều chiều



# MẢNG MỘT CHIỀU

- **Cú pháp khai báo**

- **Dùng Array literals**

- Cú pháp

- ```
var | let | const array_name = [item1, item2, ...];
```

- Ví dụ

- ```
const animals = [ 'Cat' , 'Dog' , 'Pig' , 'Tiger' , 'Lion' , 'Bird' , 'Duck' ]
```

- **Dùng Array constructor**

- Cú pháp

- ```
var | let | const array_name = new Array( item1 , item2 , ...)
```

- Ví dụ

- ```
let numbers = new Array(1, 2, 3, 4, 5)
```

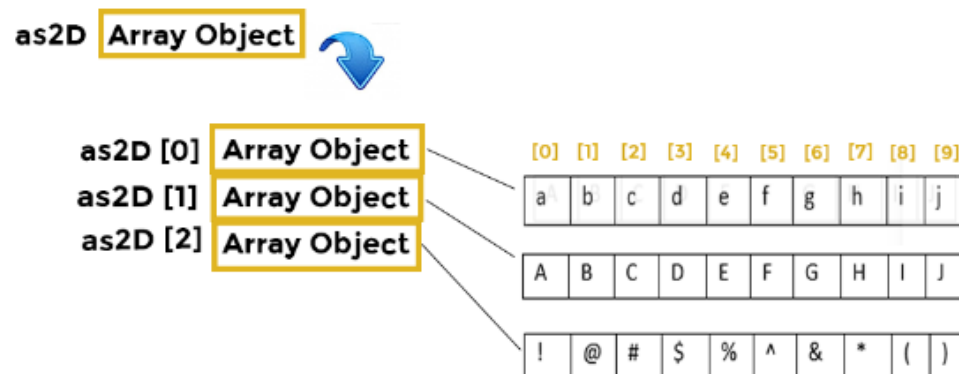
- **Truy cập vào phần tử của mảng qua tên mảng và chỉ số mảng**

- Ví dụ: `animals[1] = 'mouse'`

- **Quy tắc đặt tên mảng:**

- Giống quy tắc đặt tên biến

# MẢNG NHIỀU CHIỀU - 1



- **Khái niệm**

- Mảng 2 chiều là mảng có các phần tử là mảng
- Có thể có mảng 2 chiều, mảng 3 chiều, ...

- **Cú pháp khai báo**

- Dùng Array literals
  - Cú pháp

**const array** = **[**[item1\_1,item2\_1,...],... ,[itemN\_1,itemN-2,...]**]**

- Ví dụ

**const number2D** = **[**[1,2,7],[3,1,8],[6,4,2],[8,5,6]**]**

# MẢNG NHIỀU CHIỀU - 2

- **Cú pháp khai báo**

- Dùng Array constructor

- Cú pháp

```
const array = new Array(  
  new Array(item1,item2,...),  
  ...,  
  new Array(itemN_1,itemN_2)  
)
```

- Ví dụ

```
const number2D = [  
  new Array(1,2,7),  
  new Array(3,1,8),  
  new Array(6,4,2),  
  new Array(8,5,6)]
```

- **Truy cập phần tử mảng**

- `arrayName[row_Index][col_Index]`

	[0]	[1]	[2]
[0]	1	2	7
[1]	3	1	8
[2]	6	4	2
[3]	8	5	6

```
number2D[0][0] = 1  
number2D[0][1] = 2  
number2D[0][2] = 7  
number2D[2][1] = 4
```

# CÁC HÀM LÀM VIỆC VỚI MẢNG - 1

- **push(item1,item2,...)**

- Thêm một hoặc nhiều phần tử vào cuối mảng và trả về độ dài mới của mảng
- Mảng ban đầu sẽ có thêm các phần tử được **push()**
- Ví dụ

```
const numbers = [1, 4, 5, 10] //[ 1, 4, 5, 10 ]
```

```
let numbers_length = numbers.push(6,7) //[1,4,5,10,6,7], numbers_length = 6
```

- **pop()**

- Loại bỏ phần tử cuối của mảng và trả về phần tử đã loại bỏ
- Ví dụ

```
const numbers = [1, 4, 5, 10] //[ 1, 4, 5, 10 ]
```

```
let num_pop = numbers.pop() //[1,4,5], num_pop = 10
```

# CÁC HÀM LÀM VIỆC VỚI MẢNG - 2

- **unshift()**

- Thêm một hoặc nhiều phần tử đầu mảng, và trả về độ dài mới của mảng
- Ví dụ

```
const numbers = [1, 4, 5, 10] // [1, 4, 5, 10]
```

```
let numbers_length = numbers.unshift(6, 7) // [6, 7, 1, 4, 5, 10], numbers_length = 6
```

- **shift()**

- Loại bỏ phần tử đầu tiên của mảng và trả về phần tử đã loại bỏ
- Ví dụ

```
const numbers = [1, 4, 5, 10] // [1, 4, 5, 10]
```

```
let num_shift = numbers.shift() // [4, 5, 10], num_shift = 1
```



# CÁC HÀM LÀM VIỆC VỚI MẢNG - 3

Hàm	Dữ liệu	Cú pháp	Kết quả
arr1.concat(arr2,arr3...): Nối mảng	arr1 = [1,2] arr2 = [3,4] arr3 = [5]	let Children = arr1.concat(arr2,arr3)	children = [1,2,3,4,5]
Array.copyWithin(target,start,end): Sao chép mảng	arr = [1,2,3,4,5]	arr.copyWithin(2,0,2) arr.copyWithin(2,0)	arr = [1,2,1,2,5] arr = [1,2,1,2,3]
Array.fill(value,start,end): điền giá trị từ chỉ số start đến end	arr = [1,2,3,4,5]	arr.fill(6) arr.fill(6,2,4)	arr = [6,6,6,6,6] arr=[1,2,6,6,5]
Array.includes(element,start): kiểm tra element có tồn tại không	arr = [1,2,3,4,5]	let exists = Arr.includes(3) let exists = Arr.includes(3,3)	exist = True exist = False
Array.indexOf(item, start): lấy chỉ số của phần tử item đầu tiên từ trái sang phải	arr = [1,2,3,4,5]	let index = arr.indexOf(6) let index = arr.indexOf(3) let index = arr.indexOf(3,3)	index = -1 index = 2 index = -1
Array.join(separator): chuyển mảng thành chuỗi	arr = [1,2,3,4,5]	let str = Arr.join() let str = Arr.join(" and ")	str = "1,2,3,4,5" str = "1 and 2 and 3 and 4 and 5"
Array.reverse(): Đảo mảng	arr = [1,2,3,4,5]	arr.reverse()	arr = [5,4,3,2,1]
Array.slice(start,end): cắt mảng từ chỉ số start đến chỉ số end	arr = [1,2,3,4,5]	let arrNew = arr.slice(1,3)	arrNew = [2,3]
Array.sort(): Sắp xếp mảng	arr = [2,4,1,,5,3]	arr.sort()	arr = [1,2,3,4,5]
Array.splice(index,howmany,item1,..., itemN)	arr = [1,2,3,4,5]	arr.splice(2,0,6,7) arr.splice(2,,2)	arr=[1,2,6,7,3,4,5] arr=[1,2,5]

# DUYỆT VÀ LÀM VIỆC VỚI CÁC PHẦN TỬ CỦA MẢNG - 1

- **Vòng for**
  - Ví dụ: Hiển thị các phần tử trong mảng

```
const cities = ['Ha Noi', 'Da Nang', 'Can Tho', 'Ho Chi Minh', 'Vung Tau', 'Qua Nhon', 'Vinh']
```

```
for (let i = 0; i < cities.length; i++) {  
  console.log("index", i);  
  console.log("phần tử trong mảng", cities[i]);  
}
```

# DUYỆT VÀ LÀM VIỆC VỚI CÁC PHẦN TỬ CỦA MẢNG - 2

- Vòng for...of

```
const array1 = ["a", "b", "c"];

for (const element of array1) {
  console.log(element);
}

// expected output: "a"
// expected output: "b"
// expected output: "c"
```

```
const iterable = [10, 20, 30];

for (let value of iterable) {
  value += 1;
  console.log(value);
}

// 11
// 21
// 31
```

# DUYỆT VÀ LÀM VIỆC VỚI CÁC PHẦN TỬ CỦA MẢNG - 3

- Vòng for...in và for...of

```
const array1 = ["a", "b", "c"];

for (const element of array1) {
  console.log(element);
}

// expected output: "a"
// expected output: "b"
// expected output: "c"
```

```
const iterable = [10, 20, 30];

for (let value of iterable) {
  value += 1;
  console.log(value);
}

// 11
// 21
// 31
```

- 1. Khái niệm array (mảng) trong javascript**
- 2. Mảng 1 chiều**
- 3. Mảng 2 chiều và mảng đa chiều**
- 4. Các hàm làm việc với mảng**



# KẾT THÚC

HỌC VIỆN ĐÀO TẠO LẬP TRÌNH CHẤT LƯỢNG NHẬT BẢN