

S1 กลุ่ม 2

concat() : ใช้สำหรับสร้าง Array ใหม่จาก การรวมตัวแปร Array 2 ตัวขึ้นไป โดยที่ตัวแปร Array ที่นำมารวมกันนั้นยังเหมือนเดิม ไม่มีการเปลี่ยนแปลงใด ๆ

1. method syntax

```
concat()
concat(value0)
concat(value0, value1)
concat(value0, value1, ... , valueN)
```

2. วิธีใช้ : สามารถใช้ได้โดยการพิมพ์ .concat() หลัง array ตัวที่ต้องการใช้ จากนั้นจึงใส่ array ที่ต้องการเชื่อมลงไป

พารามิเตอร์

- valueN คือ array ที่ต้องการนำมาต่อท้าย array ตั้งต้นเพื่อให้กลายเป็น array ใหม่

3. ตัวอย่างการใช้งาน

ตัวอย่างที่ 1

```
const array1 = [1,2,3]
const array2 = [4,5,6]
const array3 = array1.concat(array2);
//จะได้ output คือ [1,2,3,4,5,6]
```

ตัวอย่างที่ 2

```
let a = ['a', 'A']
let b = ['b', 'B']
let c = ['c', 'C']
let abc = a.concat(b, c)
console.log(abc);
//จะได้ output คือ [ 'a', 'A', 'b', 'B', 'c', 'C' ]
```

ตัวอย่างที่ 3

```
let fruit = [['mango','durian']]
let animal = [['dog','cat'],['dove'],'frog']
let fruitAndAnimal = fruit.concat(animal)
console.log(fruitAndAnimal)
//จะได้ output คือ [ [ 'mango', 'durian' ], [ 'dog', 'cat' ], [ 'dove' ], 'frog' ]
```

map() : ใช้สำหรับการเรียก Function หรือ ทำ Function ภายในตัวเอง โดยที่ map จะทำหน้าที่ในการ loop element ภายใน Array เข้าสู่ Function นั้น และ return ค่าออกมาเป็น Array

1. method syntax

```
// Arrow function
map((element) => { ... })
map((element, index) => { ... })
map((element, index, array) => { ... })

// Callback function
map(callbackFn)
map(callbackFn, thisArg)

// Inline callback function
map(function callbackFn(element) { ... })
map(function callbackFn(element, index) { ... })
map(function callbackFn(element, index, array){ ... })
map(function callbackFn(element, index, array) { ... }, thisArg)
```

2. วิธีใช้ : พิมพ์ .map() หลัง array ที่ต้องการ ซึ่งภายในวงเล็บจะใส่ฟังก์ชัน

พารามิเตอร์

- element คือ element แต่ละตัวที่อยู่ใน array
- index คือ index ของ element ที่ถูกเรียกใช้งาน
- array คือ Array map() ที่ถูกเรียกใช้

3. ตัวอย่างการใช้งาน

ตัวอย่างที่ 1

```
let arr1 = [1, 2, 3]
let result = arr1.map(x => x * 2)
console.log(result);//[ 2, 4, 6 ]
```

ตัวอย่างที่ 2

```
let number = [4, 16, 64]
let result = number.map(Math.sqrt)
console.log(result);//[ 2, 4, 8 ]
```

ตัวอย่างที่ 3

```
let number = [1, 2]
let resultStringNumber = number.map(x => {
    let result = x * 2
    return result.toString()
})
console.log(resultStringNumber); //[ '2', '4' ]
```