

Join()

1. Method Syntax

- `array.join(separator)`

2. คำอธิบายความสามารถของ Method

- **join()** คือ วิธีสร้างและส่งคืนสตริงใหม่โดยเชื่อม element ทั้งหมดในอาร์เรย์ และ คั่นด้วยเครื่องหมายจุลภาคหรือสตริงตัวคั่นที่ระบุ (หากอาร์เรย์มีเพียงหนึ่งรายการ รายการนั้นจะถูกส่งคืนโดยไม่ต้องใช้ตัวคั่น)

3. วิธีการใช้งาน Method

- `array.join(separator)`
- แปลง elements ทุกตัวที่อยู่ใน array ให้เป็น string และนำ elements ทั้งหมดมาเชื่อมเข้าด้วยกัน โดยจะส่งค่าคืนเป็น string โดยมี

Separator คือ ตัวคั่นระหว่าง elements จะใส่หรือไม่ใส่ก็ได้ ถ้าไม่ใส่ค่า default คือ comma [,]

4. ตัวอย่างการใช้งาน Method

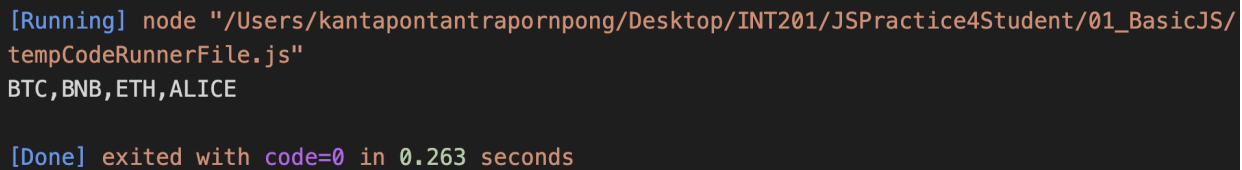
- Example 1

A code editor window with a dark background and three colored window control buttons (red, yellow, green) in the top-left corner. It contains two lines of JavaScript code:

```
let cryptocurrency = new Array('BTC', 'BNB', 'ETH', 'ALICE')  
console.log(cryptocurrency.join());
```

```
let cryptocurrency = new Array('BTC', 'BNB', 'ETH', 'ALICE')  
console.log(cryptocurrency.join());
```

Result

A terminal window with a dark background showing the execution of the code. It displays the command being run, the output of the console.log statement, and the final status of the process.

```
[Running] node "/Users/kantapontantrapornpong/Desktop/INT201/JSPpractice4Student/01_BasicJS/  
tempCodeRunnerFile.js"  
BTC,BNB,ETH,ALICE  
  
[Done] exited with code=0 in 0.263 seconds
```

สร้าง Array ที่เป็น object ชื่อ cryptocurrency

เรียกใช้ method join จะเชื่อม elements ใน array และคั่นด้วย comma[,] ซึ่ง
เป็น separator default

- Example 2



```
var numbers = [120, 127, 138, 148, 161];  
console.log(numbers.join(' / '));
```

Result

```
[Running] node "c:\Users\TUA\Documents\JavaScript\Lab1-10\JS-SourceCode Lab1-10\join.js"  
120 / 127 / 138 / 148 / 161  
  
[Done] exited with code=0 in 0.143 seconds
```

สร้างตัวแปรที่เป็นอาร์เรย์ในชื่อ numbers โดยให้แสดงผลลัพธ์โดยใช้ .join()
และตั้ง separator โดยให้แสดงผลระหว่าง element ด้วยเครื่องหมาย /

- Example 3



```
let people = ['John', 'Jane', 'Max', 'Doe', 'Noon'];  
console.log(people.join(" and "));
```

Result

```
[Running] node "c:\Users\HP\Documents\ปี2 IT\Term1 64\  
John and Jane and Max and Doe and Noon  
  
[Done] exited with code=0 in 0.072 seconds
```

สร้าง Array ที่เป็น object ชื่อ people

เรียกใช้ method join() เพื่อเชื่อม elements ใน array โดยตั้ง separator ให้
แสดงผลระหว่าง elements ด้วย “ and ”

Reduce[]

1. Method Syntax

//arrow functions

```
//arrow functions
reduce((previousValue, currentValue) => { ... } )
reduce((previousValue, currentValue, currentIndex) => { ... } )
reduce((previousValue, currentValue, currentIndex, array) => { ... } )
reduce((previousValue, currentValue, currentIndex, array) => { ... }, initialValue)
```

2. คำอธิบายความสามารถของ Method

- **reduce[]** คือ การรวม element ของ array โดยใช้ฟังก์ชันที่เราระบุเพื่อสร้างค่าเดียว method มีการกำหนดค่าเริ่มต้น และเรียกใช้ฟังก์ชันการเรียกกลับ "reducer" ที่ผู้ใช้ระบุในแต่ละ element ของ array โดยส่งค่าคืนจากการคำนวณใน element ก่อนหน้านั้น

3.วิธีการใช้งาน Method

reduce[] เปรียบเสมือนกับการที่เราเขียน reducer เพื่อเปลี่ยน input ที่เป็น Array ของเราให้เป็น output ที่เราต้องการ โดย reduce[] รับพารามิเตอร์ 2 ตัวคือ callback และ initialValue

Parameter คือ

1. callbackFn

ซึ่งจะประกอบด้วย reducer คือ

previousValue (ค่าที่เกิดจากการเรียกใช้ callbackFn)

currentValue [ค่าปัจจุบัน]

currentIndex [มีหรือไม่มีก็ได้]

array [the array to traverse]

2. initialValue

Return Value : ค่าที่ได้จาก reducer แล้วทำการสะสมค่าไปเรื่อยๆ ซึ่งจะได้ผลลัพธ์ออกมาเป็น single output value

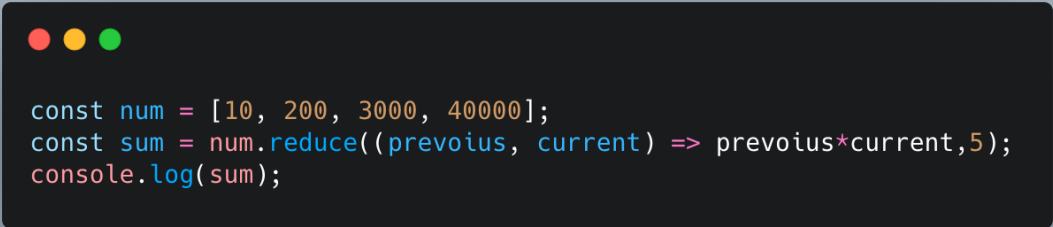
ตัวอย่าง :

Reduce [] ทำการรับ acc,value เข้าไป ซึ่งเราระบุให้ ค่า initial เป็น 0

```
//example
let arr = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
let sum = arr.reduce((acc, value) => value + acc, 0)
console.log(sum)
```

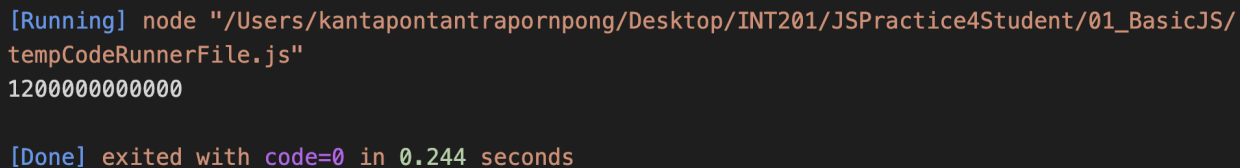
4. ตัวอย่างการใช้งาน Method

//Example 1 :



```
const num = [10, 200, 3000, 40000];  
const sum = num.reduce((prevoius, current) => prevoius*current,5);  
console.log(sum);
```

//output



```
[Running] node "/Users/kantapontantrapornpong/Desktop/INT201/JSPractice4Student/01_BasicJS/  
tempCodeRunnerFile.js"  
12000000000000  
[Done] exited with code=0 in 0.244 seconds
```

สร้างตัวแปรที่เป็น array ชื่อ num จากนั้นสร้างตัวแปรใหม่ชื่อ sum เพื่อใช้เก็บค่า
method reduce ของ num และตั้งเงื่อนไขคือให้ค่าแรกคูณกับค่าถัดไป
โดย ค่าเริ่มต้นคือ 5

//Example 2 :



```
let prices = [40, 30, 32, 50, 15];  
let total = prices.reduce(function(sum, price){  
    return sum + price  
}, 0);  
  
console.log(total);
```

//output

```
[Running] node "c:\Users\HP\Documents\ปี2 IT\Term1 64\INT  
167
```

```
[Done] exited with code=0 in 0.071 seconds
```

สร้างตัวแปรที่เป็น array ชื่อ prices โดยเก็บค่าเป็นตัวเลข จากนั้นสร้างตัวแปรที่
ชื่อ total โดยเก็บค่า method reduce ของ prices และสร้าง function ที่เก็บค่า
sum และค่า price โดยให้ return ค่ากลับเป็นค่า sum + price ไปเรื่อยๆ โดยค่า
เริ่มต้นคือ 0

//Example 3 :

```
let members = [
  { id: "A001", name: "KANTAPON" },
  { id: "A002", name: "KITAYOD" },
  { id: "A003", name: "UDOMCHAI" },
  { id: "A004", name: "HIRUN" },
  { id: "A005", name: "SIRIWAT" }
]

let structuredMembers = members.reduce((result, member) => {
  result[member.id] = member

  return result
}, {})

console.log(structuredMembers);
```

//output

```
//output
{
  A001: { id: 'A001', name: 'KANTAPON' },
  A002: { id: 'A002', name: 'KITAYOD' },
  A003: { id: 'A003', name: 'UDOMCHAI' },
  A004: { id: 'A004', name: 'HIRUN' },
  A005: { id: 'A005', name: 'SIRIWAT' }
}
```

สร้างตัวแปรที่เป็นอาร์เรย์ชื่อ member ซึ่งเก็บค่าที่เป็น object ไว้

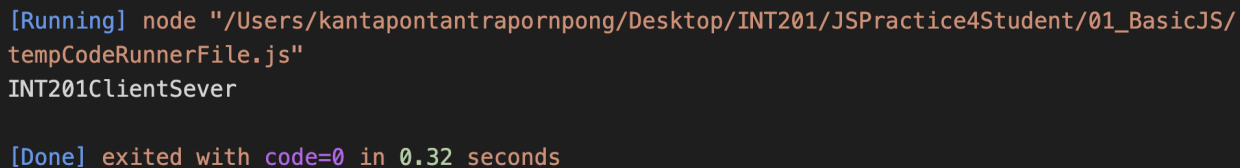
สร้างตัวแปร structuredMembers เพื่อเก็บค่า method reduce ของ join โดยใส่เงื่อนไขให้ return ค่าออกมาเป็น id และภายใน id ประกอบด้วย id และ name ที่ตรงกัน

//Example 4 :



```
const character = ['NT', 201, 'Client', 'Sever'];  
const subject = character.reduce((before, next) => before+next, 'I');  
console.log(subject);
```

//output



```
[Running] node "/Users/kantapontantrapornpong/Desktop/INT201/JSPractice4Student/01_BasicJS/  
tempCodeRunnerFile.js"  
INT201ClientSever  
[Done] exited with code=0 in 0.32 seconds
```

สร้างตัวแปรที่เป็นอาร์เรย์ ซึ่งเก็บค่าที่เป็น String และ number ชื่อ character และสร้างตัวแปรชื่อ subject เพื่อเก็บค่าที่เรียกใช้ method subject ของ character โดยกำหนดค่าก่อนหน้า และค่าถัดไป ตั้งเงื่อนไขให้นำมาบวกกัน และใช้ค่าเริ่มต้นคือ 'I'