

```
import { sum } from "./script.js";
```

```
// เป็นการนำเข้า variable หรือ function จากไฟล์ที่ต้องการ
```

```
// จะใช้ในที่นี้คือ การรับ function sum() จาก script.js
```

```
console.log("case 1");
```

```
let fun0 = sum(); // เรา execute function sum จะเป็นการ initial
```

```
// ค่าเริ่มต้นและ return function inner เพื่อเตรียมบวก
```

```
// ตอนนี้ fun0 คือ function inner พอเรา execute ก็จะเป็นการเอาค่าที่ส่งไปบวกเพิ่มในตัวแปรและ return ผลรวม
```

```
console.log(fun0(0)); // 0 : ใส่ 0 เพื่อดูว่าตอนนี้ข้อมูลด้านในมีเท่าไรโดยไม่ต้องบวกอะไรเพิ่ม
```

```
//เรากรอกข้อมูลตัวเลขหลายตัว inner จะเอาตัวเลขทั้งหมดมาบวกกับผลรวมปัจจุบันเอาไปเก็บไว้และ return ผลรวม
```

```
console.log(fun0(5, 6, 8)); // 19
```

```
console.log("case 2"); // ทำแบบค่าติดลบ
```

```
let fun1 = sum(-8);
```

```
console.log(fun1(0)); // -8
```

```
console.log(fun1(5, 6, -8)); // -5
```

```
console.log(fun1(20)); // 15
```

```
console.log("case 3"); // ทำแบบค่าเป็นทศนิยม
```

```
let fun2 = sum(-8.2);
```

```
console.log(fun2(0)); // -8
```

```
console.log(fun2(5, -8, 8.2)); // -3
```

```
console.log(fun2(8.6)); // 5.6
```

export { sum } // เป็นการเตรียมส่ง variable หรือ function ไปให้ไฟล์อื่นที่ต้องการเรียกใช้

// ในที่นี้จะเตรียมส่ง code ไปให้ main.js

// sum function เอาไว้หาผลรวมโดยจะรับ parameter ตัวเดียวเป็นค่าเริ่มต้นก่อนจะเอาค่าอื่นมา
รวม

function sum(num1 = 0) {

// เมื่อ execute function sum() จะเก็บค่าเริ่มต้นโดยมี default parameter เป็น 0

let result = num1; // inner รับ parameter เป็น rest parameter ชื่อ num2

// โดย inner จะเอา num2 มาหาผลรวมแล้วเอามาเก็บไว้ใน result

// จากนั้นก็ return result ก็คือผลรวมล่าสุดออกมา

function inner(...num2) {

// closure function (สามารถเข้าถึง local variable,function ของ sum ได้)

result += num2.reduce((prev, cur) => prev + cur); // หาผลรวมแล้วเก็บไว้ใน result

return result;

}

return inner;

}

ผลลัพธ์

```
[Running] node "d:\INT201\2010264\main.js"  
case 1  
0  
19  
case 2  
-8  
-5  
15  
case 3  
-8.2  
-3  
5.6  
  
[Done] exited with code=0 in 0.122 seconds
```