

Destructuring, Rest Parameters and Spread Syntax

Destructuring sử dụng ở function

```
const handle = ({ a, b = 0, c }) => {  
  return a + b + c  
}  
  
console.log(handle({ a: 1, c: 3 })) // 4
```

Spread syntax

```
const user = {  
  name: 'Duoc',  
  age: 24,  
  ability: ['coding']  
}  
  
// shallow copy  
const cloneUser = { ...user }  
  
// // merge object  
const info = {  
  address: 'Vietnam'  
}  
const mergedUser = { ...user, ...info }
```

Thêm thuộc tính **ability** vào **userPlus** Nếu **user** đã có **ability** thì nó sẽ bị ghi đè

```
const userPlus = { ...user, ability: ['sing'] }
```

```
// concat array  
const list1 = [1, 2, 3, 4]  
const list2 = [5, 6, 7, 8]  
const list3 = [...list1, ...list2]  
  
// shallow copy array  
const cloneList1 = [...list1]  
  
// shallow copy và add 1 phần tử vào cuối  
const list4 = [...list1, 5]
```

```
// shallow copy và add 1 phần tử vào đầu  
const list5 = [0, ...list1]
```

Rest Parameter

```
const handle = (a, b, ...c) => {  
  return c  
}  
  
handle(1, 2, 3, 4, 5, 6) // [3,4,5,6]  
  
// Kết hợp rest parameter với destructuring  
  
const handle = ({ a, b, ...c }) => {  
  return c  
}  
  
handle({ a: 1, b: 2, c: 3, d: 4, e: 5 }) // {c: 3, d: 4, e: 5}  
  
// cũng có thể áp dụng với array  
const handle = ([a, b, ...c]) => {  
  return c  
}  
  
handle([1, 2, 3, 4, 5]) // [3, 4, 5]
```

```
const cake = {  
  name: 'Banh Qui',  
  price: 200,  
  quantity: 100  
}  
  
const { name, ...other } = cake  
console.log(other) // {price: 200, quantity: 100}
```