

JavaScript

구조분해할당 (Destructuring) – 김근형 강사

구조분해 할당(Destructuring)

- 구조분해 할당(Destructuring)
 - 배열이나 객체의 속성을 해체하여 그 값을 개별 변수에 담을 수 있게 하는 JavaScript 표현식을 뜻함.
 - 사전적인 의미는 “~의 구조를 파괴하다 ”란 뜻이며 기존의 변수 할당 방식을 마치 배열처럼 선언하여 [분할 할당]을 하도록 하는 기능이다.

```
[구문]  
let one, two;  
[one, two] = [ 1, 2 ]
```

구조분해 할당(Destructuring)

○ Array 구조분해 할당(Destructuring)

```
let one, two, three, four, five;  
const values = [1, 2, 3];  
  
[one, two, three] = values;  
console.log("A:", one, two, three);  
  
[one, two] = values;  
console.log("B:", one, two);  
  
[one, two, three, four] = values;  
console.log("C:", one, two, three, four);  
  
[one, two, [three, four]] = [1, 2, [73, 74]];  
console.log("D:", one, two, three, four);
```

A: 1 2 3

B: 1 2

C: 1 2 3 undefined

D: 1 2 73 74

```
let one, two, three, four, other;  
[one, , four] = [1, 2, 3, 4];  
console.log(one, four);
```

```
[one, ...other] = [1, 2, 3, 4];  
console.log(other);
```

1 4

```
▼ Array(3) ⓘ  
  0: 2  
  1: 3  
  2: 4  
  length: 3  
  ▶ __proto__: Array(0)
```

구조분해 할당(Destructuring)

- Array 구조분해 할당(Destructuring)

- 변수 교환 트릭

```
let guest = "Jane";  
let admin = "Pete";  
  
// 변수 guest에 Pete, 변수 admin에 Jane이 저장되도록 교환  
[guest, admin] = [admin, guest];  
  
console.log(`${guest} ${admin}`); // Pete Jane
```

- Default Parameter를 활용한 구조분해 할당

```
let [name = "Guest", surname = "Anonymous"] = ["Julius"];  
  
console.log(name);    // Julius  
console.log(surname); // Anonymous
```

구조분해 할당(Destructuring)

○ Object 구조분해 할당(Destructuring)

```
let {one, two} = {one: 1, nine: 9};  
console.log(one, two);  
  
let three, four;  
({three, four} = {three: 3, four: 4});  
console.log(three, four);
```

```
1 undefined
```

```
3 4
```

```
let five, six;  
({one: five, two: six} = {one: 10, two: 20});  
console.log(five, six);  
//console.log(one, two); //error  
  
let {nine, plus: {ten}} = {nine: 9, plus: {ten: 10}};  
console.log(nine, ten);
```

```
10 20
```

```
9 10
```

구조분해 할당(Destructuring)

- Object 구조분해할당(Destructuring)

- Default Parameter 사용

```
let options = {title: "Menu"};  
let { width = 100, height = 200, title } = options;  
  
console.log(title); // Menu  
console.log(width); // 100  
console.log(height); // 200
```

- Default Parameter와 클론 동시 사용

```
let options = {title: "Menu"};  
let { width: w = 100, height: h = 200, title } = options;  
  
console.log(title); // Menu  
console.log(w); // 100  
console.log(h); // 200
```

구조분해 할당(Destructuring)

- Object 구조분해 할당(Destructuring)

- 특정 프로퍼티만 뽑아내기

```
let options = {title: "Menu", width: 100, height: 200};  
  
// title만 변수로 뽑아내기  
let { title } = options;  
console.log(title); // Menu
```

- 객체에서 Spread 를 활용하여 구조분해 할당 하기

```
let options = {title: "Menu", height: 200, width: 100};  
// title = 이름이 title인 프로퍼티  
// rest = 나머지 프로퍼티들  
let { title, ...rest } = options;  
  
// title엔 "Menu", rest엔 {height: 200, width: 100}이 할당된다.  
console.log(title); // Menu  
console.log(rest); // {height: 200, width: 100}  
console.log(rest.height); // 200  
console.log(rest.width); // 100
```

구조분해 할당(Destructuring)

- Object 구조분해 할당(Destructuring)

- 중첩 구조분해 할당

```
let options = {  
  size: { width: 100, height: 200 },  
  items: ["Cake", "Donut"],  
  extra: true  
};  
  
// 코드를 여러 줄에 걸쳐 작성해 의도하는 바를 명확히 드러냄  
let {  
  size: { width, height }, // size는 여기,  
  items: [item1, item2], // items는 여기에 할당함  
  title = "Menu" // 분해하려는 객체에 title 프로퍼티가 없으므로 기본값을 사용함  
} = options;  
  
console.log(title); // Menu  
console.log(width); // 100  
console.log(height); // 200  
console.log(item1); // Cake  
console.log(item2); // Donut
```


구조분해 할당(Destructuring)

- Object 구조분해할당(Destructuring)
 - for ~ of 문에서의 구조분해 - 1

```
let values = [  
  { item: "선물1", amount: { apple: 10, candy: 20 } },  
  { item: "선물2", amount: { apple: 30, candy: 40 } }  
];  
  
for (var { item: one, amount: { apple: two, candy: five } } of values) {  
  console.log(one, two, five);  
};
```

```
선물1 10 20  
선물2 30 40
```

구조분해 할당(Destructuring)

- Object 구조분해할당(Destructuring)
 - for ~ of 문에서의 구조분해 - 2

```
var people = [  
  {  
    name: "Mike Smith",  
    family: { mother: "Jane Smith", father: "Harry Smith", sister: "Samantha Smith" },  
    age: 35  
  },  
  {  
    name: "Tom Jones",  
    family: { mother: "Norah Jones", father: "Richard Jones", brother: "Howard Jones" },  
    age: 25  
  }  
];  
for (var { name: n, family: { father: f } } of people) {  
  console.log("Name: " + n + ", Father: " + f);  
}
```

Name: Mike Smith, Father: Harry Smith
Name: Tom Jones, Father: Richard Jones

구조분해 할당(Destructuring)

- Function parameter 구조분해할당(Destructuring)

- 함수를 이용한 구조분해할당 예제 - 1

```
function total({ one, plus: { two, five } }) {  
  console.log(one + two + five);  
};  
  
total({ one: 1, plus: { two: 2, five: 5 } }); // 8
```

- 함수를 이용한 구조분해할당 예제 - 2

```
// 함수에 전달할 객체  
let options = {title: "My menu", items: ["Item1", "Item2"]};  
  
function showMenu({ title = "Untitled", width = 200, height = 100, items = [] }) {  
  console.log(`${title} ${width} ${height}`); // My Menu 200 100  
  console.log(items); // Item1, Item2  
}  
  
showMenu(options);
```