

JS-중급 8

☰ 태그	
📅 날짜	@2023년 6월 2일

```
arr.sort()
배열 재정렬
배열자체가 변경되니 주의

let arr = [1,5,4,2,3];
let arr1 = ["a","b","c","d","e"];
let arr2 = [27,8,5,13,12];

arr.sort();
arr1.sort();
arr2.sort();

console.log(arr)
console.log(arr1)
console.log(arr2)

// [object Array] (5)
[1,2,3,4,5]
// [object Array] (5)
["a","b","c","d","e"]
// [object Array] (5)
[12,13,27,5,8]

arr2는 제대로 정렬되지 않는다.
시작 수가 1과 2가 맨앞으로 왔기 때문

let arr2 = [27,8,5,13,12];

arr2.sort((a,b)=>{
  return a-b;
});

console.log(arr2)

// [object Array] (5)
[5,8,12,13,27]

-----
Lodash 라는 라이브러리를 쓰면 편함

_.sortBy(arr);
```

```
reduce : 인수로 함수를 받음  
(누적 계산값, 현재값 ) => {return 계산값 };
```

```
// 배열의 모든 수 합치기  
let arr = [1,2,3,4,5];  
// for, for of, for each  
let result = 0;  
arr.forEach((num) => {  
    result += num;  
});  
console.log(result)
```

15

reduce를 사용을 하면 한번에 처리할 수 있다.

```
// 배열의 모든 수 합치기  
// prev : 누적 계산값, cur : 현재값  
let arr = [1,2,3,4,5];  
  
// for, for of, for each  
  
const result = arr.reduce((prev, cur)=>{  
    return prev + cur;  
},0)
```

```
console.log(result)
```

15

```
let userList = [  
    {name : "Mike", age: 17},  
    {name : "Tom", age: 20},  
    {name : "Mino", age: 30},  
    {name : "Hongsi", age: 5},  
    {name : "SBS", age: 27},  
    {name : "Jane", age: 45},  
];  
  
let result = userList.reduce((prev, cur) => {  
    if(cur.age > 19) {  
        prev.push(cur.name);  
    }  
});
```

```

    }
    return prev;
  }, []);

console.log(result);

// [object Array] (4)
["Tom", "Mino", "SBS", "Jane"]

```

구조분해할당(Destructuring assignment)

구조분해할당(Destructuring assignment) : 구조 분해
할당 구문은 배열이나 객체의 속성을 분해해서 그 값을
변수에 담을 수 있게 하는 표현식

배열 구조 분해

```
let [x,y] = [1,2];
```

```
console.log(x); // 1
```

```
console.log(y); // 2
```

```
-----
let users = ['mike', 'tom', 'jane'];
```

```
let [user1, user2, user3] = users;
```

위 코드는 아래와 같다

```
let user1 = users[0];
```

```
let user2 = users[1];
```

```
let user3 = users[2];
```

```
console.log(user1); // 'mike'
```

```
console.log(user2); // 'tom'
```

```
console.log(user3); // 'jane'
```

```
-----
let str = 'Mike-Tom-Jane';
```

```
let [user1, user2, user3] = str.split('-');
```

```
console.log(user1); // 'Mike'
```

```
console.log(user2); // 'Tom'
```

```
console.log(user3); // 'Jane'
```

배열 구조 분해 : 기본값

```
let [a, b, c] = [1,2];  
c는 undefined
```

```
let [a=3, b=4, c=5] = [1,2];
```

```
console.log(a); //1  
console.log(b); //2  
console.log(b); //5
```

배열구조분해 : 일부 반환값 무시

```
let [user1, , user2] = ['mike','tom','jane','tony'];
```

```
console.log(user1); // 'mike'  
console.log(user2); // 'jane'
```

배열구조분해 : 바꿔치기

```
let a = 1;  
let b = 2;
```

```
[a,b] = [b,a]
```

객체 구조 분해

```
let user = {name : 'Mike', age : 30};
```

```
let {name, age} = user;  
// let {age, name} = user; 순서를 바꿔도 동일하게 동작
```

```
console.log(name); // 'Mike'  
console.log(age); // 30
```

새로운 변수 이름으로 할당

```
let user = {name : 'Mike' , age : 30};  
// let {name, age} = user;  
let {name : userName, age : userAge} = user;
```

```
console.log(userName); // 'Mike'  
console.log(userAge); // 30
```

기본값

```
let user = {name : 'Mike', age : 30};
```

```
let {name, age, gender} = user;  
// gender는 undefined
```

```
let { name, age, gender = 'male'} = user;
```

```
let user = {name : 'Mike', age : 30};
```

```
let user = {  
  name : 'jane',  
  age : 18,  
  gender : 'female'  
};
```

```
let { name, age, gender = 'male'} = user;
```

```
console.log(gender); // 'female'
```

객체로 부터 받은 값이 undefined일때만 기본값을 받는다.