

삼육대학교 SW융합교육원 김진호

# 자바스크립트 기초교육

## 05. spread syntax (스프레드 문법)

5-1. rest parameter

5-2. spread syntax

5-3. array and object copy

## ❖ 5. spread syntax

### 5-1. rest parameter (나머지 매개변수)

#### 5-1-1. rest parameter (나머지 매개변수)

- 함수에 정해진 인수보다 적게 전달 되면 undefined, 정해진 인수보다 많이 전달 되면 해당 인수를 무시하고 기능한다.
- 이 때 나머지 매개변수 ... 를 사용하면 매개변수를 한데 모아 배열에 담을 수 있다.
- 유의할 점은 나머지 매개변수는 항상 마지막에 있어야 한다.

# ❖ 5. spread syntax

## 5-1. rest parameter (나머지 매개변수)

### 5-1-1. rest parameter (나머지 매개변수)

```
function merge(msg1, msg2) {  
    return msg1 + msg2;  
}  
  
console.log(merge('안녕하세요.'));  
console.log(merge('안녕하세요.', '반갑습니다.'));  
console.log(merge('안녕하세요.', '반갑습니다.', '제 이름은 홍길동입니다.'));  
  
/* 함수에 정의된 인수는 2개이고  
정해진 인수보다 적게 전달 되면 undefined, 정해진 인수보다 많이 전달 되면 해당 인수를 무시하고 기능한다.  
이 때 나머지 매개변수 ... 를 사용하면 매개변수를 한데 모아 배열에 담을 수 있다. */  
  
function mergeAll(...args) {  
    let message = '';  
  
    for(let arg of args) message += arg;  
  
    return message;  
}  
  
console.log(mergeAll('안녕하세요.'));  
console.log(mergeAll('안녕하세요.', '반갑습니다.'));  
console.log(mergeAll('안녕하세요.', '반갑습니다.', '제 이름은 홍길동입니다.'));  
  
/* 유의할 점은 나머지 매개변수는 항상 마지막에 있어야 한다.  
function func(arg1, ...args, arg2) {} -> (X)  
function func(arg1, arg2, ...args) {} -> (O)  
*/
```

## ❖ 5. spread syntax

### 5-2. spread syntax (스프레드 문법, 전개 문법)

#### 5-2-1. spread syntax (스프레드 문법, 전개 문법)

- rest parameter처럼 매개변수 목록을 배열로 가져오는 것과는 반대로 배열을 통째로 매개변수에 넘겨주는 기능이다.
- 하나로 뭉쳐 있는 여러 값들의 집합을 전개해서 개별적인 값들의 목록으로 만든다.
- 사용 대상은 for ... of 문으로 순회할 수 있는 이터러블에 한정된다.

## ❖ 5. spread syntax

### 5-2. spread syntax (스프레드 문법, 전개 문법)

#### 5-2-1. spread syntax (스프레드 문법, 전개 문법)

```
console.log(`가장 큰 값 : ${ Math.max(10, 30, 20) }`);    // 30

let arr = [10, 30, 20];
console.log(`가장 큰 값 : ${ Math.max(arr) }`);    // NaN

/* Math.max()는 배열이 아닌 숫자 목록을 인수로 받기 때문에 배열의 경우
원하는대로 동작하지 않는다.
이와 같은 경우 스프레드 문법을 사용할 수 있다.
함수를 호출할 때 ...arr을 사용하면 arr이 인수 목록으로 확장 된다. */
console.log(`가장 큰 값 : ${ Math.max(...arr) }`);    // 30

let arr1 = [10, 30, 20];
let arr2 = [100, 300, 200];

// 배열 객체 여러 개 전달 가능
console.log(`가장 작은 값 : ${ Math.min(...arr1, ...arr2) }`);
// 일반 값과 혼합해서 사용 가능
console.log(`가장 작은 값 : ${ Math.min(1, ...arr1, 2, ...arr2, 3) }`);
```

# ❖ 5. spread syntax

## 5-2. spread syntax (스프레드 문법, 전개 문법)

### 5-2-1. spread syntax (스프레드 문법, 전개 문법)

- 배열 병합에도 사용 가능하다 - concat 보다 간결

```
let merged = [0, ...arr, 2, ...arr2];  
console.log(merged);
```

- 이터러블 배열 변환
- 스프레드 문법은 for ... of 와 같은 방식으로 내부에서 이터레이터(iterator, 반복자)를 사용해 요소를 수집한다.

```
let str = "JavaScript";  
console.log([...str]);           // 문자열을 배열로 변환  
console.log(Array.from(str));  
  
// Array.from()도 동일하게 기능하지만 Array.from()은 이터러블 객체 뿐만 아니라  
유사 객체 배열에서도 사용할 수 있어 무언가를 배열로 바꿀 때 보편적으로 사용한다.
```

## ❖ 5. spread syntax

### 5-3. array and object copy (스프레드 문법을 이용한 배열, 객체 복사)

#### 5-3-1. array and object copy (스프레드 문법을 이용한 배열, 객체 복사)

##### ■ 배열 복사

```
let arr = [10, 30, 20];
let arrCopy = [...arr];
console.log(arr);
console.log(arrCopy);
console.log(arr === arrCopy); // false
arrCopy.push(50);           // 복사본 배열에 값 추가
console.log(arr);
console.log(arrCopy);
```



## ❖ 5. spread syntax

### 5-3. array and object copy (스프레드 문법을 이용한 배열, 객체 복사)

#### 5-3-1. array and object copy (스프레드 문법을 이용한 배열, 객체 복사)

- 스프레드 문법의 대상은 이터러블이어야 하지만 스프레드 프로퍼티 제안은 일반 객체를 대상으로도 허용하고 있다.

```
let obj = { name : '홍길동', age : 20 };
let objCopy = { ...obj };
console.log(obj);
console.log(objCopy);
console.log(obj === objCopy); // false
objCopy.age = 30;           // 복사본 객체에 값 수정
console.log(obj);
console.log(objCopy);
```

## ❖ 정리하기

- 스프레드 문법과 나머지 매개변수의 구분
- **스프레드 문법** : 배열을 목록으로 확장해주므로 ...이 함수 호출 시,  
배열이나 객체 리터럴 내부에 사용된다.
- **나머지 매개변수** : 인수 목록의 나머지를 배열로 모아주므로 ...이 함수 매개변수의 끝에 있다.
- 두 가지 문법을 함께 사용하면 매개변수 목록과 배열 간 전환을 쉽게 할 수 있다.