



함수

김민지



목차



1. 함수
2. 함수 선언문
3. 함수 표현식
4. 함수 표현식과 선언문의 차이
5. 지역 변수 & 전역 변수
6. 매개 변수 & 전달 인자
7. 함수의 반환값
8. 함수 이름 짓기
9. 화살표 함수 기본

1. 함수

```
const pluckDeep = key => obj => key.split('.').reduce((accum, key) => accum[key], obj)

const compose = (...fns) => res => fns.reduce((accum, next) => next(accum), res)

const unfold = (f, seed) => {
  const go = (f, seed, acc) => {
    const res = f(seed)
    return res ? go(f, res[1], acc.concat([res[0]])) : acc
  }
  return go(f, seed, [])
}
```

함수: 프로그램을 구성하는 주요 구성요소

함수를 이용하면 중복 없이 코드를 여러 번 호출 가능하다



2. 함수 선언문

```
function showMessage() {  
    alert( '안녕하세요!' );  
}  
showMessage();
```

함수 선언문: 함수 선언 방식을 이용해 함수를 생성하는 방식

`function 함수이름(매개변수){ }`



3. 함수 표현식

```
let sayHi = function() {  
  alert( "Hello" );  
};
```

함수 표현식: 함수를 생성하고 변수에 값을 할당하는 것처럼 함수가 변수에 할당

4. 함수 선언문과 표현식의 차이

JavaScript ▾

```
// 함수 선언문  
function sum(a, b) {  
  return a + b;  
}
```

```
// 함수 표현식  
let sum = function(a, b) {  
  return a + b;  
};
```

1. 문법
2. 자바스크립트 엔진이 함수를 생성하는 때
3. 스코프


5. 지역 변수 & 전역 변수

```
function showMessage() {  
  let message = "안녕하세요!"; // 지역 변수  
  
  alert( message );  
}  
  
showMessage(); // 안녕하세요!
```

함수 내에서 선언한 지역 변수 local variable
함수 내에서만 접근 가능

```
let userName = 'John';  
  
function showMessage() {  
  let message = 'Hello, ' + userName;  
  alert(message);  
}  
  
showMessage(); // Hello, John
```

함수 외부에 선언된 전역 변수 global variable
함수 내부에 외부 변수와 동일한 이름의 변수가
있다면, 전역 변수가 가려진다.



6. 매개 변수 & 전달 인자

```
function showMessage(from, text) { // 인자: from, text
    alert(from + ': ' + text);
}

showMessage('Ann', 'Hello!'); // Ann: Hello! (*)
showMessage('Ann', "What's up?"); // Ann: What's up? (**)
```

Parameter 매개변수 : 함수 선언식에서 괄호 사이에 있는 변수

Argument 전달인자, 인자, 인수 : 함수를 호출할 때 매개변수에 전달되는 값

6. 매개 변수 & 전달 인자

함수 호출 시 매개변수에 인수를 전달하지 않으면 값은 **undefined**

```
function showMessage(from, text = "no text given") {  
    alert( from + ": " + text );  
}  
  
showMessage("Ann"); // Ann: no text given
```

함수를 선언할 때 기본값을 설정해주면 해결



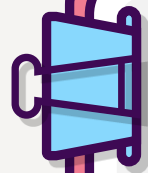
7. 반환값 *return*

```
function sum(a, b) {  
  return a + b;  
}
```

함수를 호출했을 때 함수를 호출한 그곳에 특정 값을 반환하게 할 수 있다.

이 특정 값을 반환 값이라고 부른다 : return value

- return은 함수 내 어디서든 사용 가능
- return을 만나면 함수 실행은 즉시 중단되고 함수를 호출한 곳에 값을 반환



8. 함수의 이름짓기

함수는 어떤 동작을 수행하기 위한 코드를 모아둔 것.
함수의 이름은 어떤 동작을 하는지 설명할 수 있어야 한다.

- "show"로 시작하는 함수는 대개 무언가를 보여주는 함수입니다.
- "get..." - 값을 반환함
- "calc..." - 무언가를 계산함
- "create..." - 무언가를 생성함
- "check..." - 무언가를 확인하고 불린값을 반환함

9. 화살표 함수 기본

화살표 함수 arrow function

```
let func = (arg1, arg2, ...argN) => expression
```

```
let func = function(arg1, arg2, ...argN) {  
  return expression;  
};
```

함수 **func**는 화살표(=>) 우측의 **표현식(expression)**을 평가하고, 평가 결과를 반환