



## 3강 - 함수 기초

함수란?

함수의 정의

함수의 호출

함수를 사용하는 이유

매개 변수 (parameter)

인수 (argument)

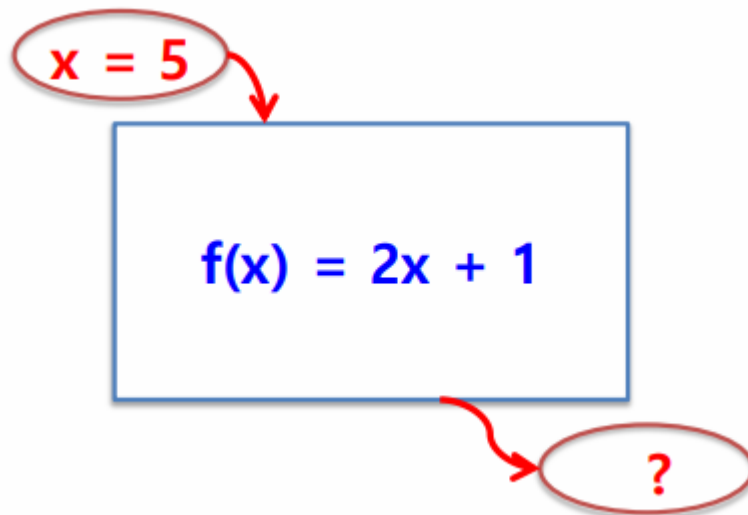
인수가 매개변수보다 적게 전달된 경우

매개변수에 기본값 설정하기 (ES6)

반환문 (return)

return은 함수의 탈출문!

### 함수란?



- 함수란 어떠한 로직을 코드 블록으로 감싸서 하나의 이름이 있는 실행단위로 만든 것을 말합니다.
- 수학의 함수처럼 어떤 데이터를 함수에 전달하면 특정값을 반환하는 형태로 만들어져 있습니다.

## 함수의 정의

```
function 함수이름 ( [매개변수 선언, ...] ) {  
    실행할 코드를 작성하는 곳  
}
```

## 함수의 호출

```
//함수 정의
function add(num1, num2) {
    return num1 + num2;
}

//함수 호출
add(10, 20); //30 반환
add(50, 20); //70 반환
add(-10, 20); //10 반환
```

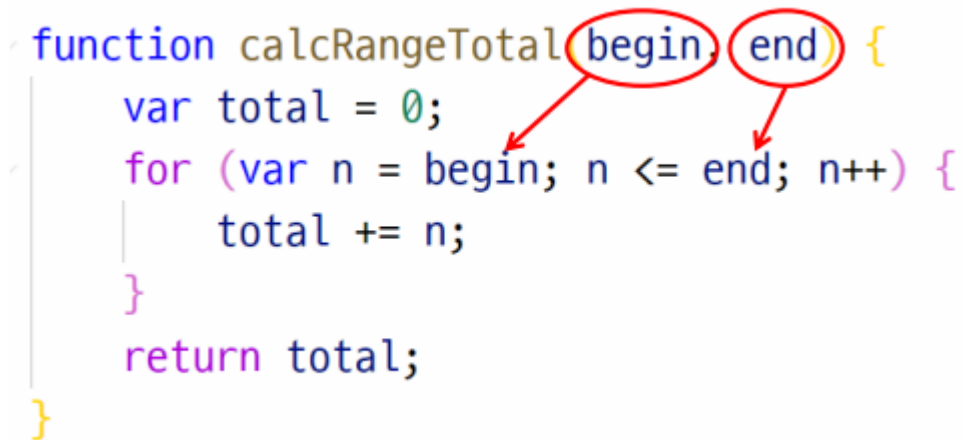
- 함수를 정의했다면 정의한 자체로는 사용이 불가능합니다.
- 함수는 반드시 호출(call)을 통해 기능을 동작시켜야 합니다.
- 호출 시에는 함수의 이름과 함께 함수에게 전달할 값을 소괄호 안에 넣어줍니다.

## 함수를 사용하는 이유

- 함수는 미리 코드를 모아 놓고 필요할 때마다 여러 번 호출할 수 있습니다.
- 즉, 코드 실행 시점을 개발자가 결정할 수 있고 여러 번 재 사용이 가능합니다.
- 그리고 함수에 이름을 붙임으로써 함수의 역할을 잘 설명할 수도 있습니다.
- 따라서 코드의 모듈화로 인해 유지 보수에도 도움이 되며 재 사용성이 높아진다는 측면에서  
유용합니다.

## 매개 변수 (parameter)

```
function calcRangeTotal(begin, end) {  
    var total = 0;  
    for (var n = begin; n <= end; n++) {  
        total += n;  
    }  
    return total;  
}
```



- 매개변수란 함수 실행을 위해 필요한 데이터를 함수 내부로 전달할 목적으로 사용하는 변수입니다.
- 모든 함수가 매개변수가 필요한 것은 아닙니다. 이를테면 사람은 일기를 쓰기 위해 펜이라는 매개체가 필요하지만 일기를 읽기 위해서는 다른 도구가 없어도 되는 것처럼 말이죠.

## 인수 (argument)

```
calcRangeTotal(1, 10);
```



- 인수란 함수 외부에서 함수를 호출할 때 함수가 실행하기 위해 필요한 값을 함수 내부의 매개변수에게 전달하는 값입니다.
- 인수는 값으로 평가될 수 있는 표현식이어야 합니다.

## 인수가 매개변수보다 적게 전달된 경우

```
function add(x, y) {  
  return x + y;  
}  
  
add(2);
```

- 인수가 매개 변수보다 적게 전달되면 할당되지 않은 매개변수의 값은 undefined로 지정됩니다.
- 따라서 위와 같은 예제는 x에 2가 전달되나 y에는 아무것도 전달되지 않아 y에 undefined가 할당되며 즉 결과는 x + undefined의 값인 NaN이 리턴됩니다.

## 매개변수에 기본값 설정하기 (ES6)

```
function add(x=1, y=2) {  
    return x + y;  
}
```

```
add(3, 6); //9 반환  
add(10); //12 반환  
add(); //3 반환
```

## 반환문 (return)

```
function calcRangeTotal(begin, end) {  
    var total = 0;  
    for (var n = begin; n <= end; n++) {  
        total += n;  
    }  
    return total;  
}
```

- 반환문은 return 키워드와 반환값으로 이뤄진 반환문을 사용해 실행 결과를 함수 외부로 전달합니다.
- 반환값은 표현식으로 만들어야 합니다.

**return은 함수의 탈출문!**

```
function add(x, y) {  
    return x + y;  
    console.log('hello'); //실행 안됨!  
}
```

- 함수는 return을 만나는 순간 즉시 종료됩니다.
- 따라서 위와 같이 return이후에 코드를 적게 되면 실행되지 않습니다.