

* destructuring

```
var {body} = document;
```

K	V
:	
<u>body</u>	<u>200</u>
:	

~~200~~
<body> — </body>

document.getElementsByTagName("body")[0]

body == document.body ==

* function



function prototype (C/C++)
= method signature (Java)

function 함수명 (파라미터, 파라미터, ...)

Function body {
 `문장1;`
 :
 return 표현식;
}

리턴 : "aaa", 20, true, {}-[], []
변수 : a, score, sum ...
식 : a + "hello", a * 2, 함수호출 ...

함수호출
(함수 선언시켜놓은 것)
⇒ 함수호출 (인자, 인자, ...);
 ↖ ↗
 아주먼드 (argument)

* 아규먼트와 파라미터

f(10);

function f(a, b) {
 arguments = [10]
}

• 모든 함수에 접근 가능한 Built-in 변수
• 아직 접근 가능한 대상이 없음

* 아규먼트와 파라미터

$f(10, 20);$

function $f(a, b) \{$

\equiv

}

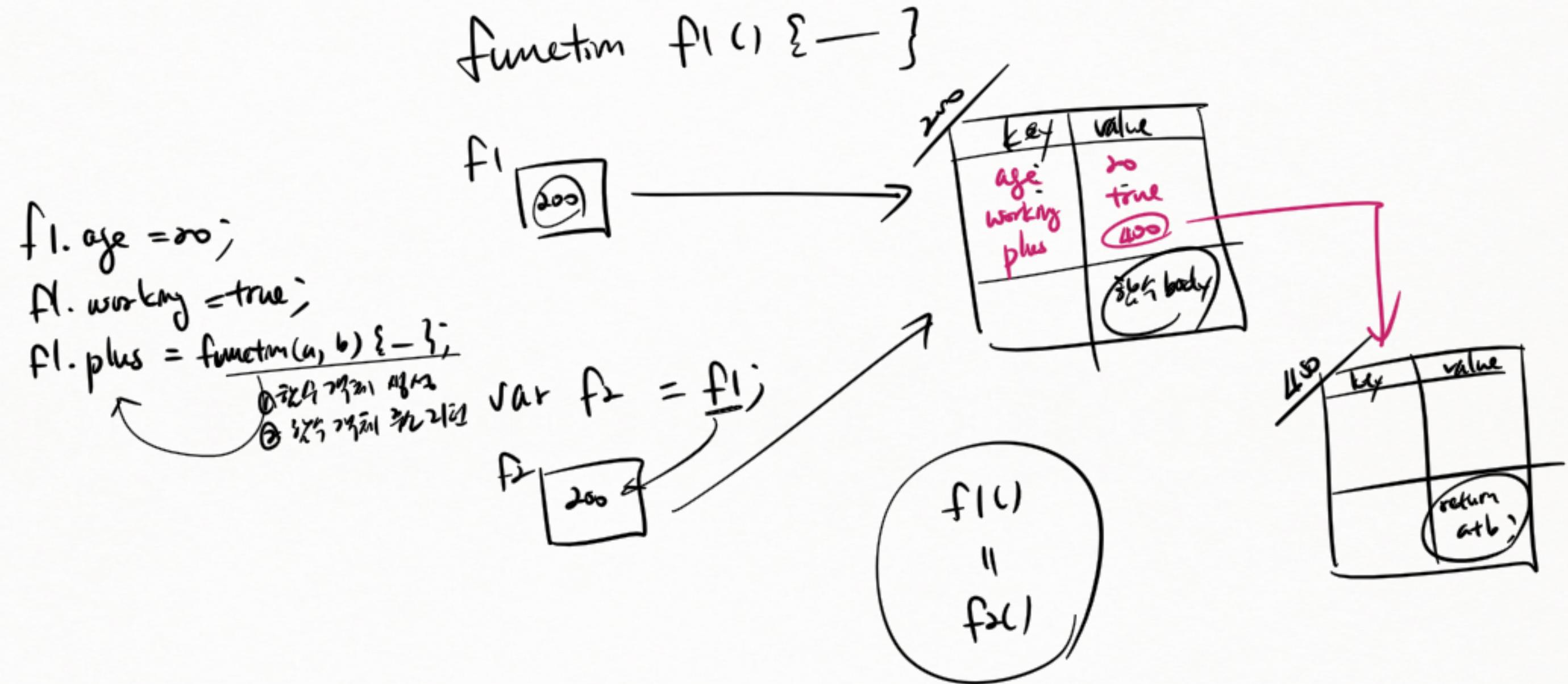
arguments = [10, 20]

* 아규먼트와 파라미터

$f(10, 20, 30, 40);$

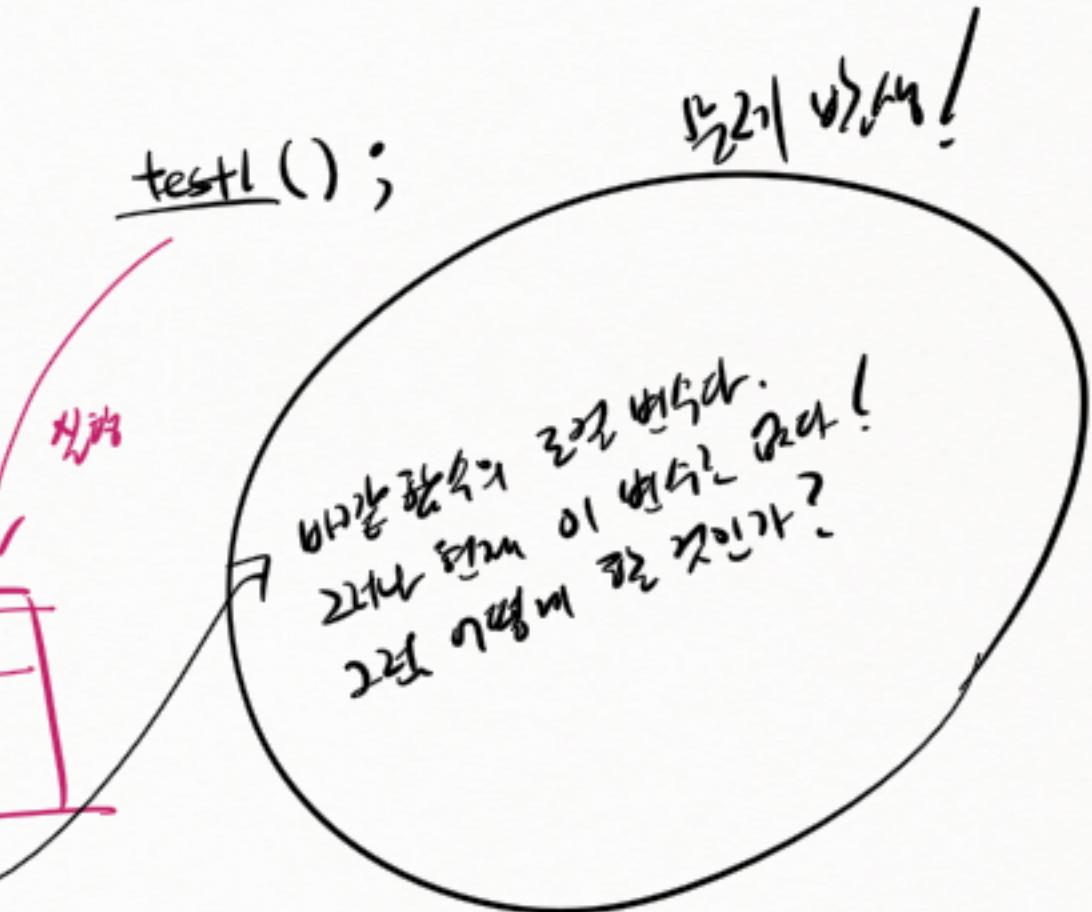
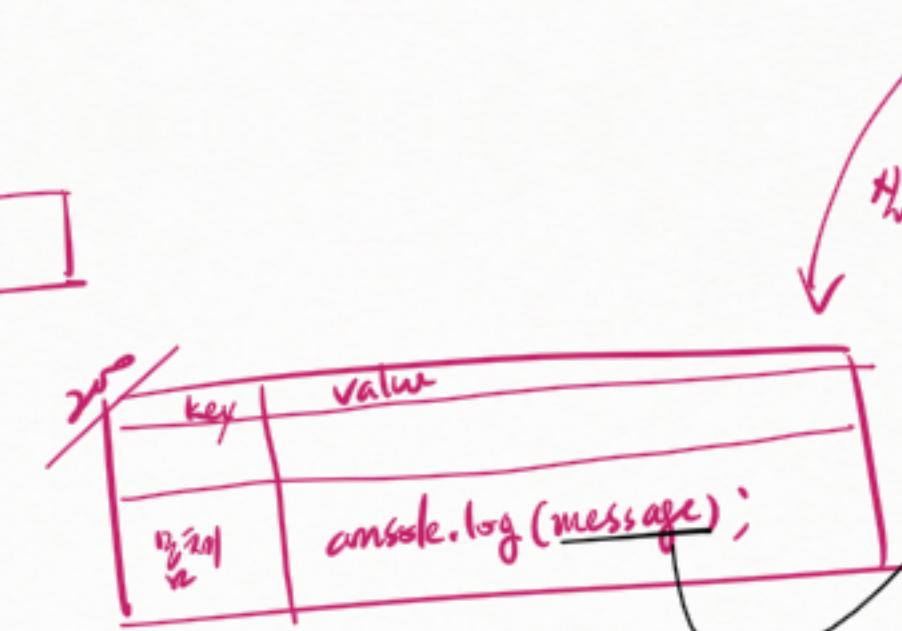
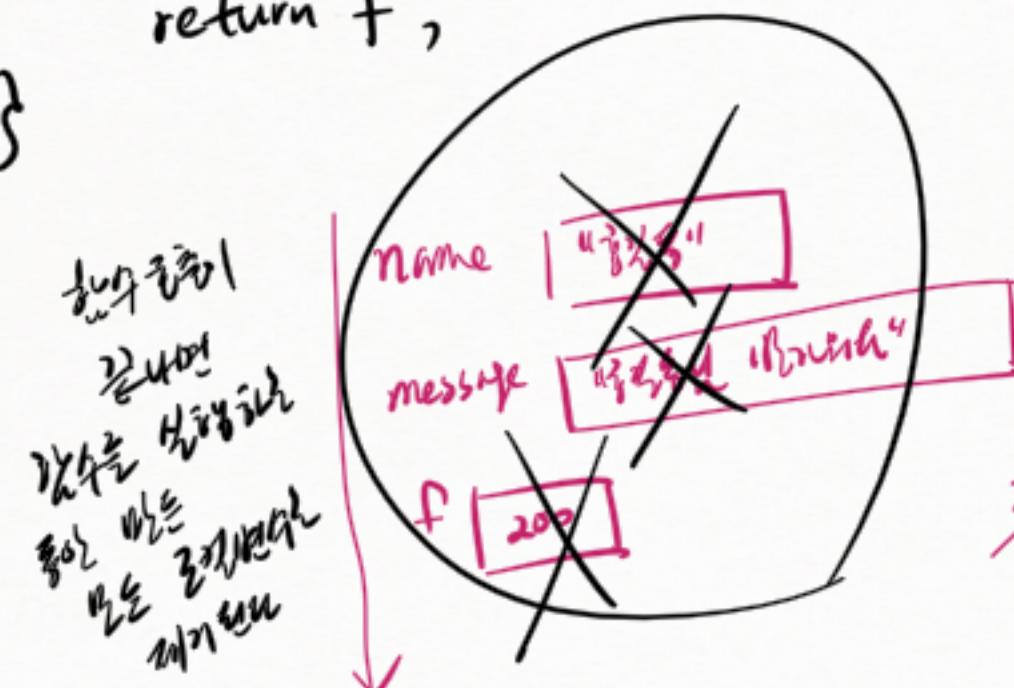
function $f(a, b)$ {
 \equiv arguments = [10, 20, 30, 40]
}

* 흡수와 레퍼런스



* closure

```
function createGreeting(name) {  
    var message = name + "님 반갑습니다";  
  
    var f = function() { console.log(message); };  
  
    return f;  
}
```



```
var test1 = createGreeting("Hello");
```

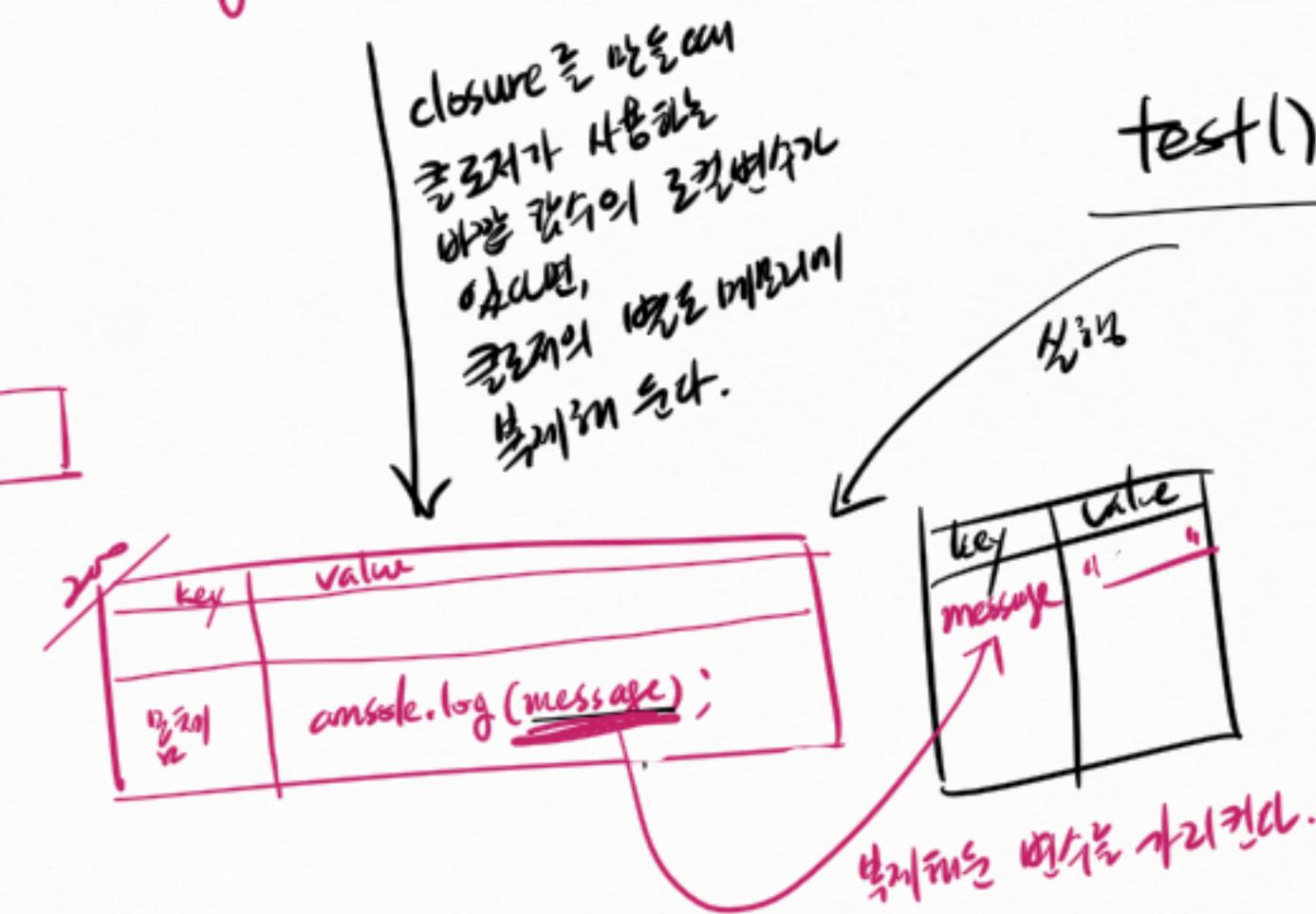
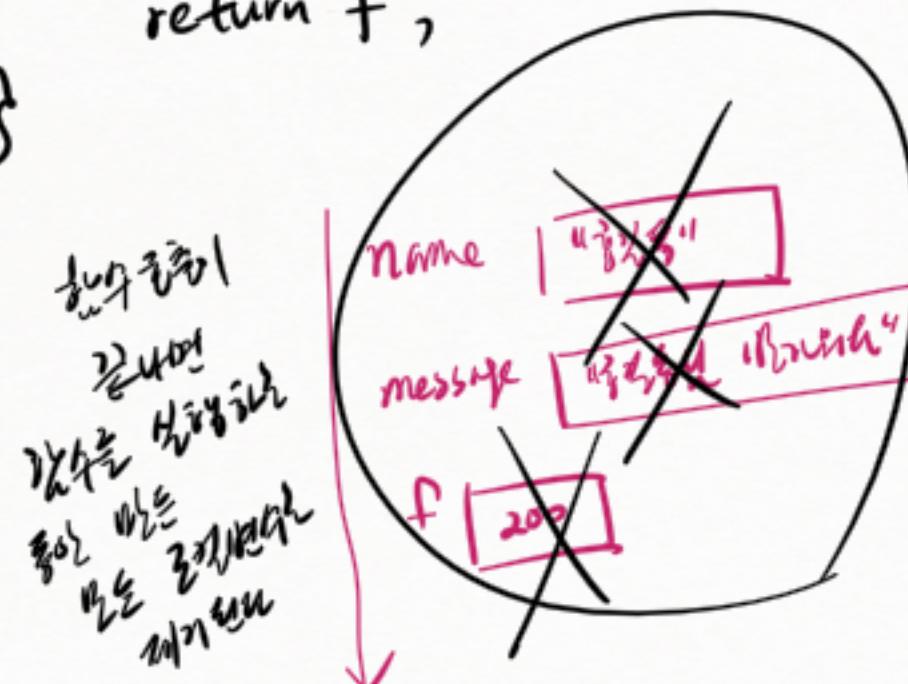
test1 → 200

200

- name
- message
- *

* closure : 1867년 3월 22일 토요일
1871년 2월 3일 일요일!

```
function createGreeting(name) {  
    var message = name + "님 환영합니다";  
  
    var f = function() { console.log(message); };  
  
    return f;  
}
```



```
var test1 = createGreeting("김민수");  
test1  
200
```

- name
- message
- *

test1();

1867년 3월 22일
message 김민수
200
환영합니다!

* closure 例

test(1)
test2()

var test1 = createGreeting ("Hello");

test1

200

key	value
	console.log(message);

closure 例
variable = 3210392 と
key value

key	value
	"Hello World"

var test2 = createGreeting ("Duck");

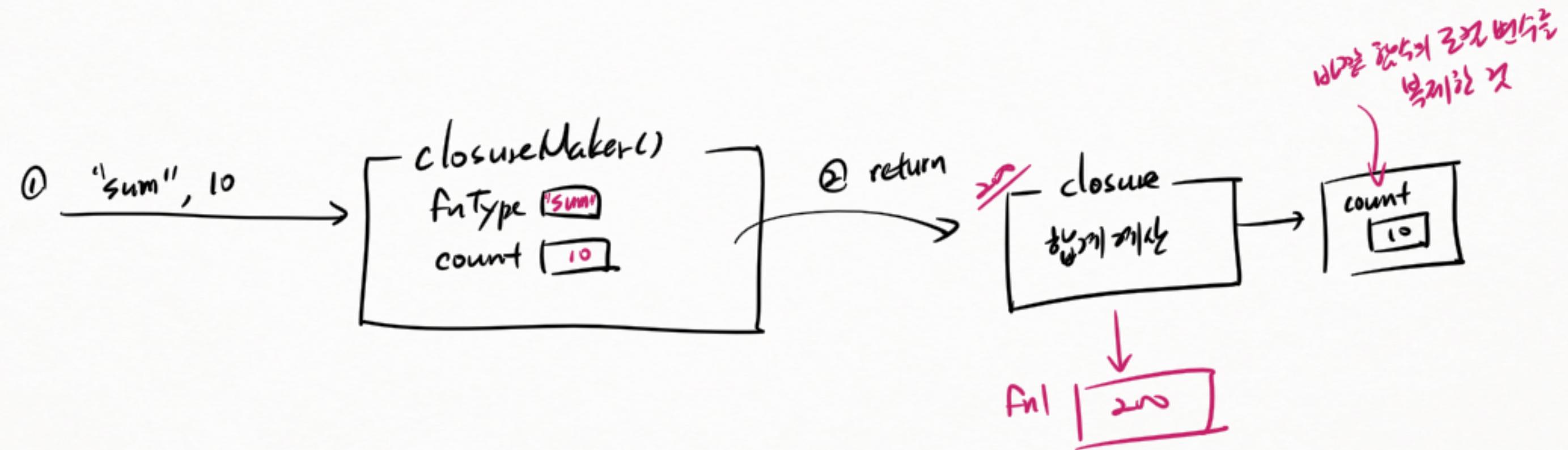
test2

300

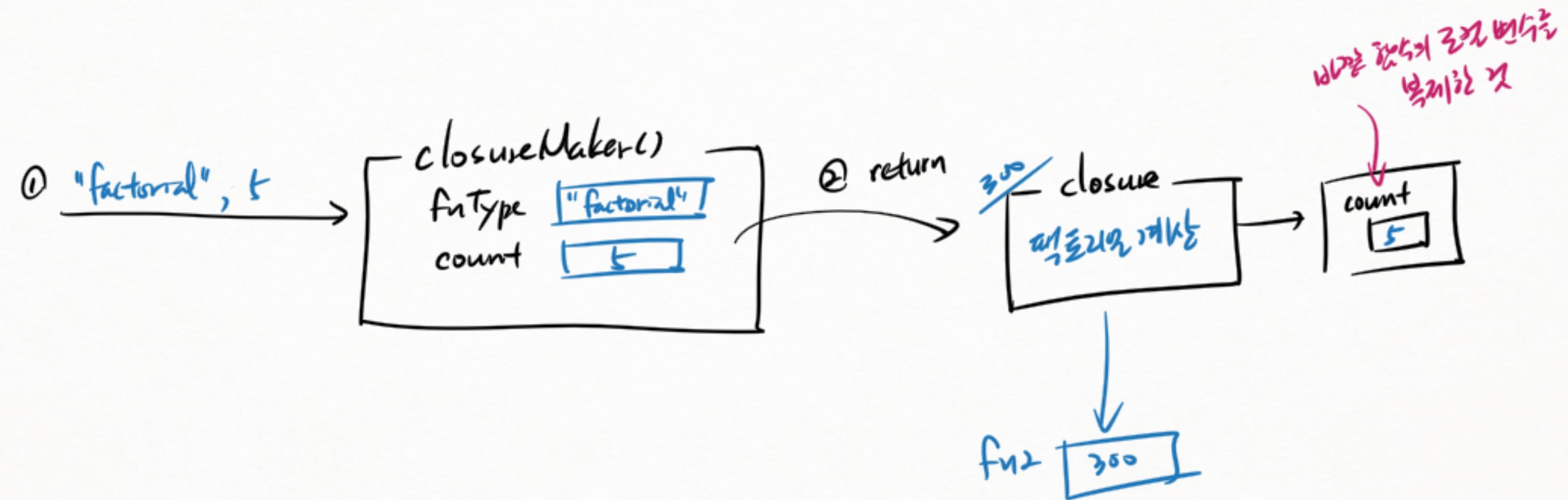
key	value
	console.log(message);

key	value
	"Duck is cute!"

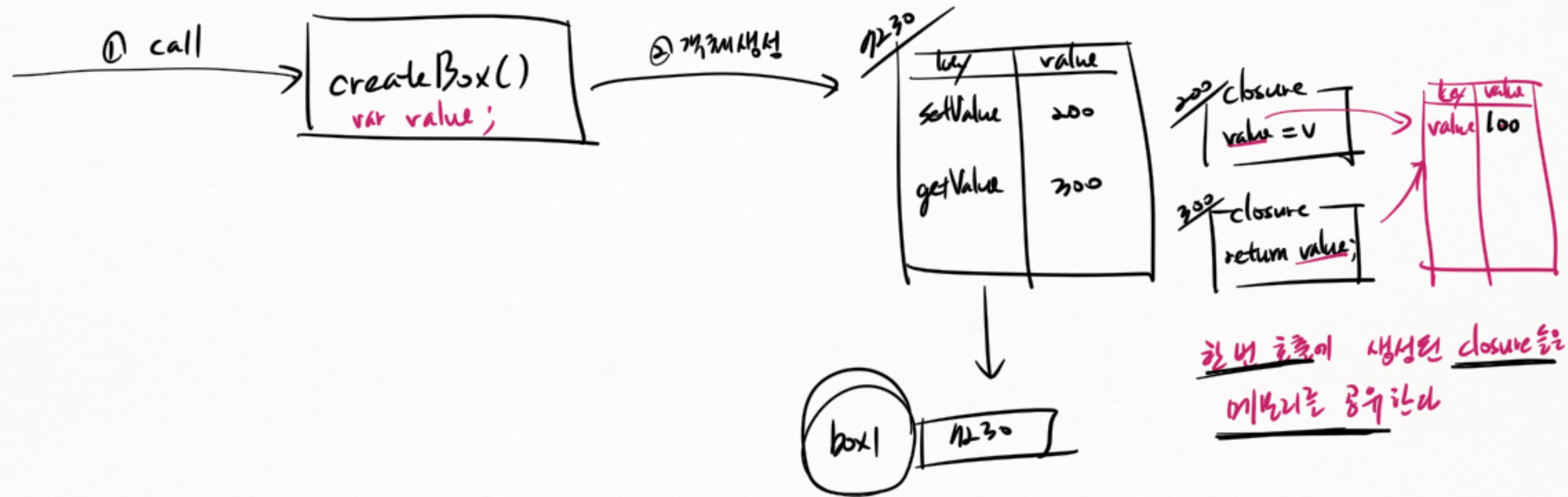
* closure № II



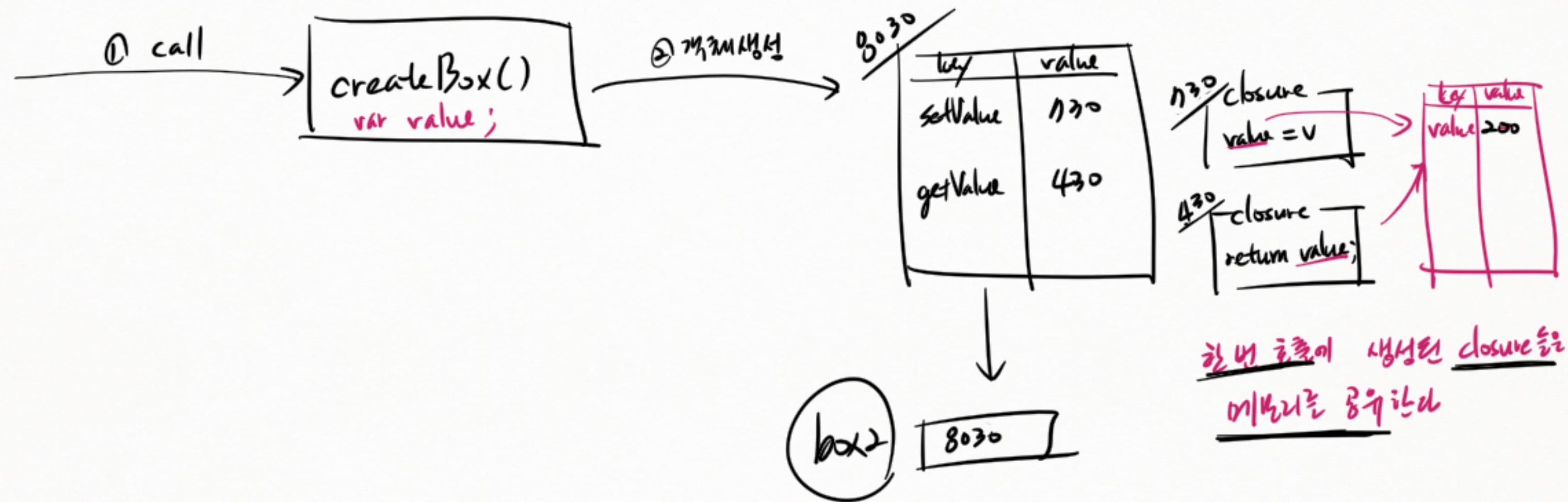
* closure № II



* 4th Ⅲ : box1



* 4th Ⅲ : box 2



```
function() { return "안녕" }  
          ↓  
console.log( (값주기)() )  
          ↑  
값주기를 가지는 함수 호출  
  
          ( ) => "안녕"  
          ↓  
console.log( (값주기)() )
```