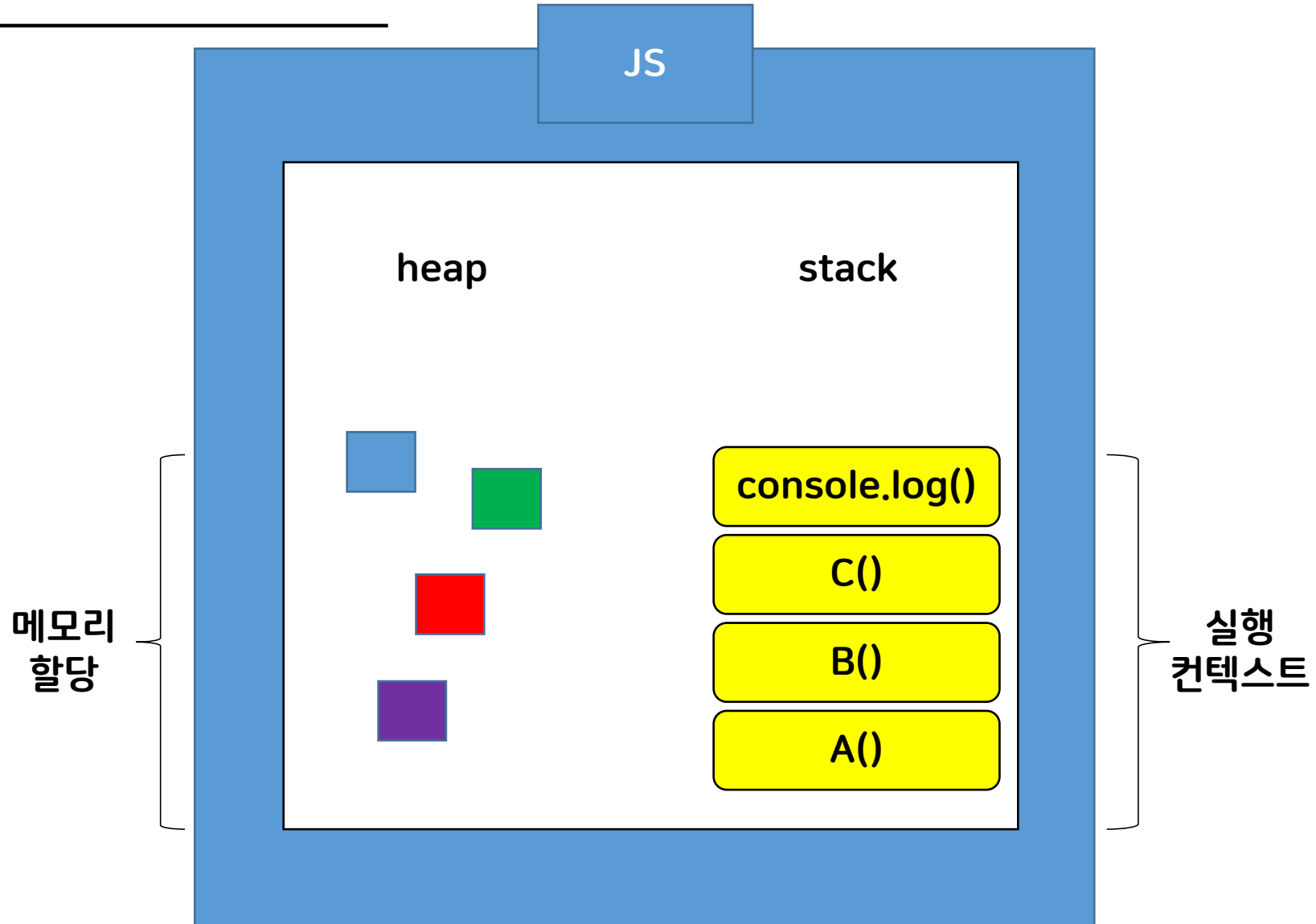


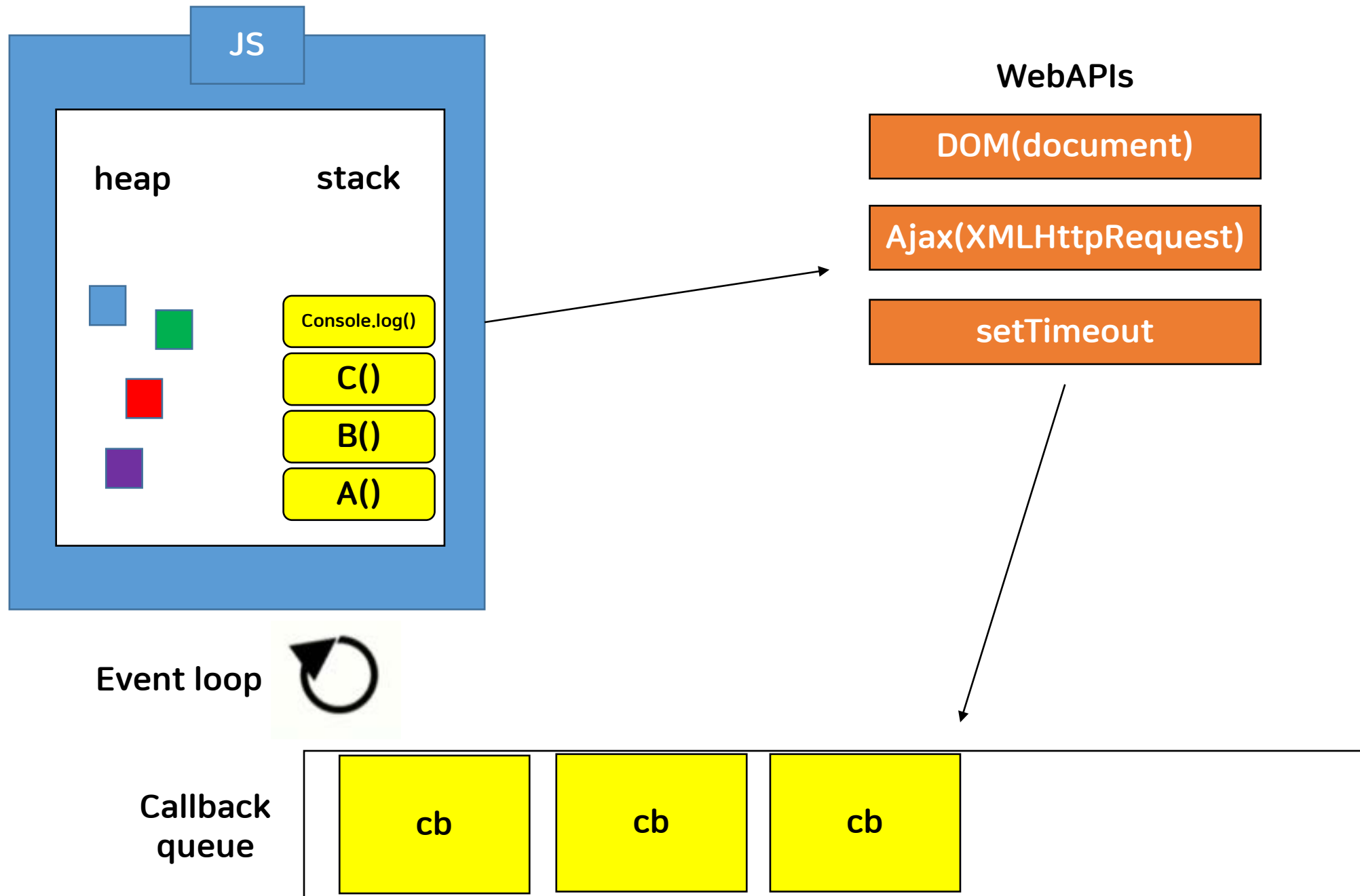
# 자바스크립트 작동 원리

---

20140289 김정호

# V8 런타임 단순화





# 자바스크립트

---

**자바스크립트 : 싱글 스레드 프로그래밍 언어**

- => 싱글 스레드 런타임을 가짐**
- => 한번에 하나의 싱글 콜스택을 가짐**
- => 하나의 프로그램은 동시에 하나의 코드만 실행할 수 있음**

# 콜 스택

```
function multiply(a, b){  
    return a * b;  
}  
function square(n) {  
    return multiply(n, n);  
}  
function printSquare(n){  
    var squared = square(n);  
    console.log(squared);  
}  
printSquare(4);
```

stack



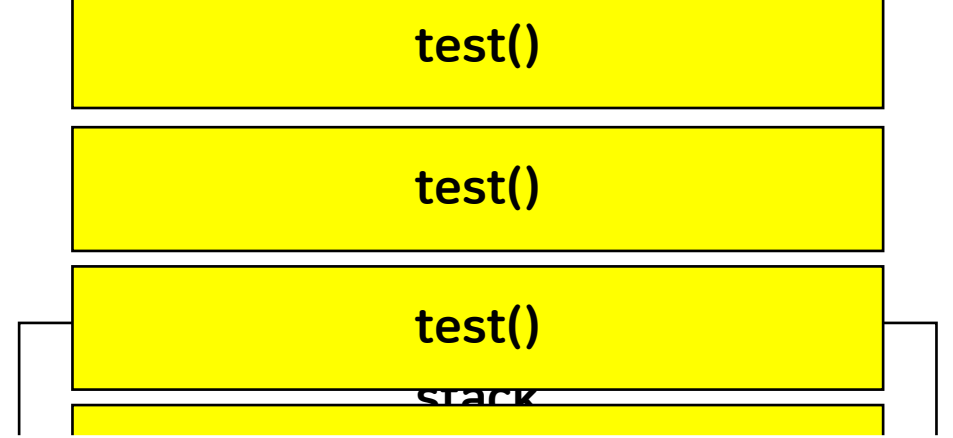
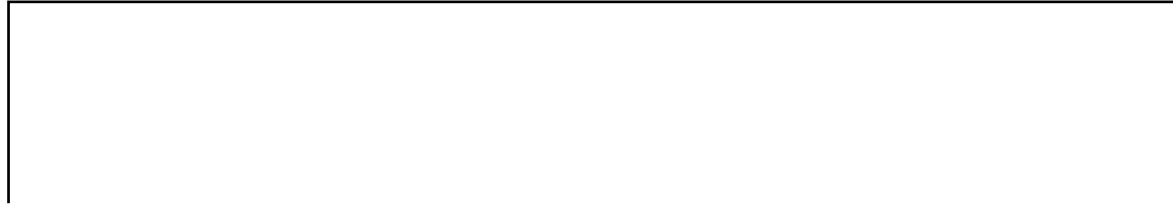
# 콜 스택

---

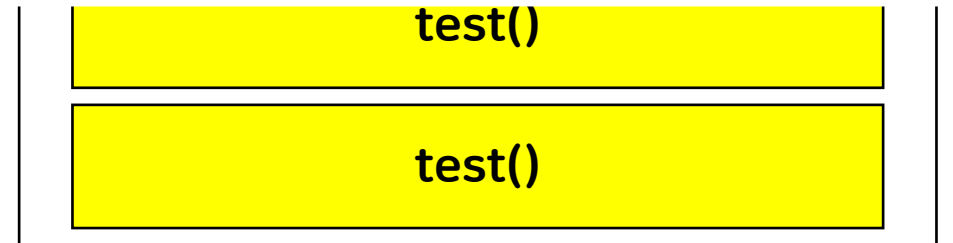
```
function A() {  
    throw new Error('에러 발생!');  
}  
function B() {  
    A();  
}  
function C() {  
    B();  
}  
  
C();
```

에러 발생!

# 콜 스택



RangeError : Maximum call stack size exceeded



# 콜 스택

---

```
var A = $.getSync(' //A.com' );  
var B = $.getSync(' //B.com' );  
var C = $.getSync(' //C.com' );
```

stack



# 비동기 콜백과 콜 스택

---

```
console.log('정호야');  
setTimeout(function() {  
  console.log('응 아니야');  
}, 5000);  
console.log('왜 불러?');
```

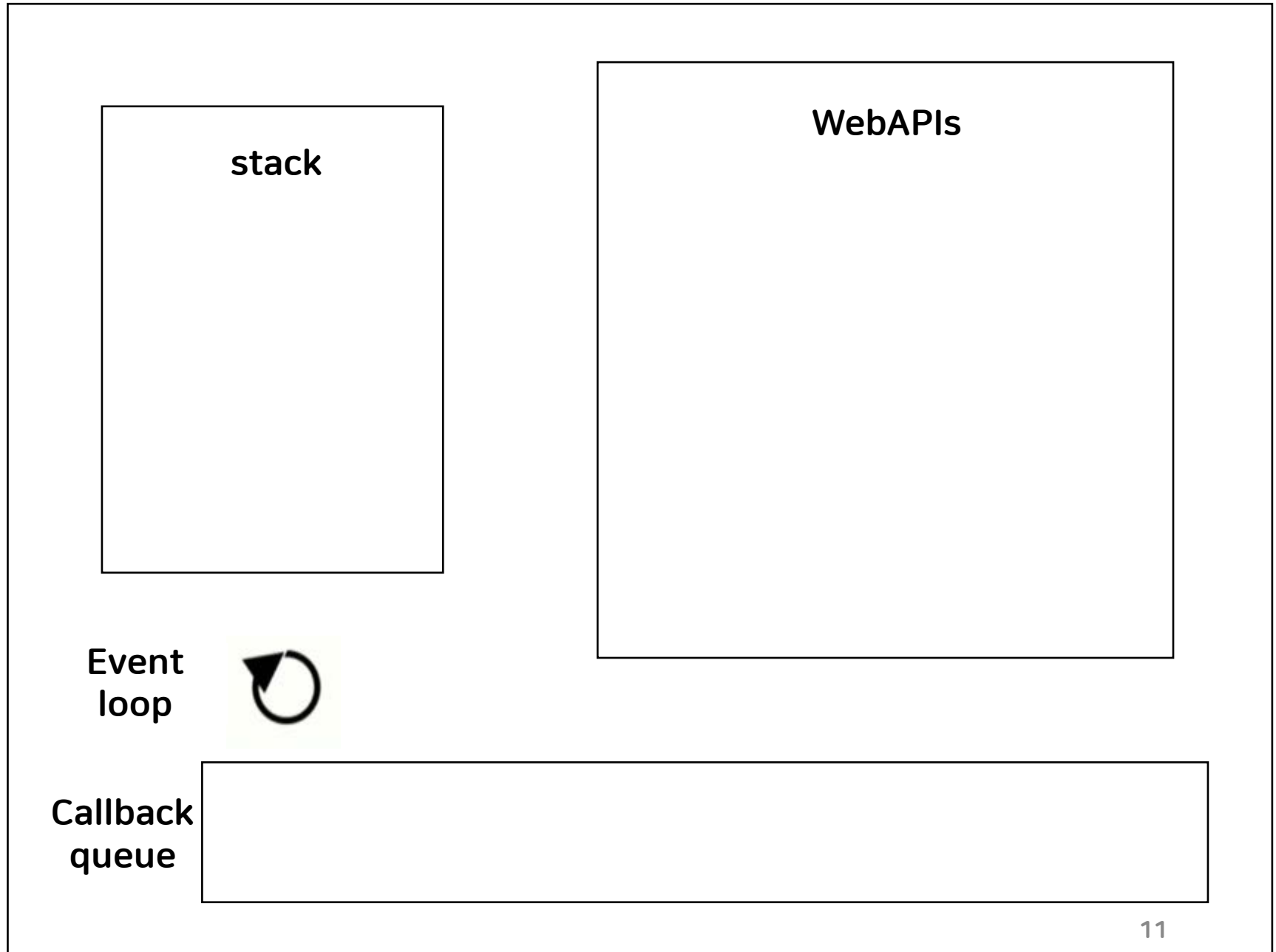
stack

# 콜백 큐와 이벤트 루프

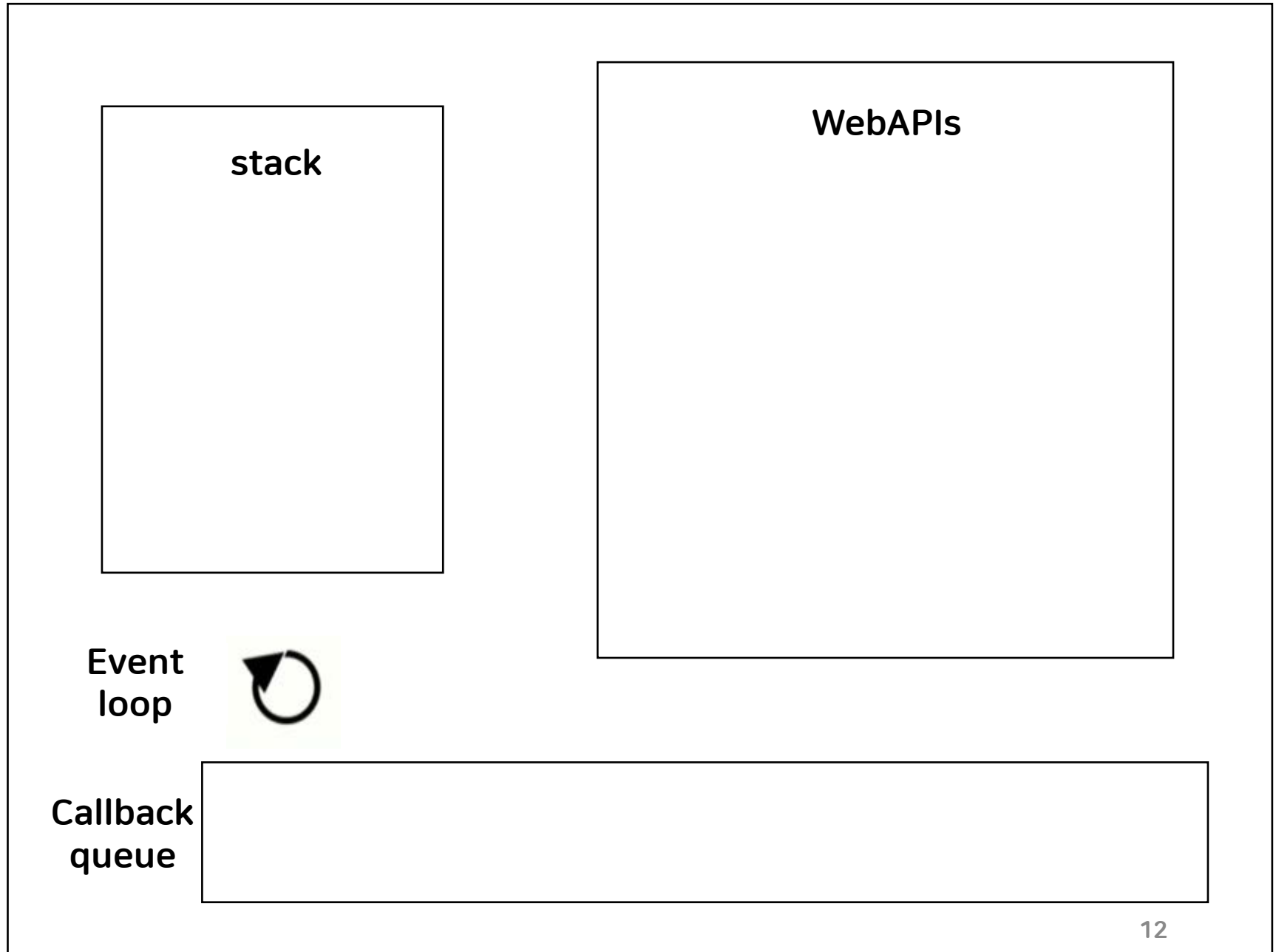
**테스크 큐(콜백 큐) : 콜백 함수들이 대기하는 큐 형태의 배열**

**이벤트 루프 : 콜 스택과 테스트 큐를 주시해서 스택이 비어있으면  
큐의 첫번째 콜백을 스택에 쌓아 효과적으로 실행 할 수 있게 해줌**

```
console.log('정호야');  
setTimeout(function() {  
  console.log('응 아니야');  
}, 5000);  
console.log('왜 불러?');
```



```
console.log('정호야');  
setTimeout(function() {  
  console.log('응 아니야');  
}, 0);  
console.log('왜 불러?');
```



# 렌더링 역시 콜백함수

---

```
var A = $.getSync('//A.com');  
var B = $.getSync('//B.com');  
var C = $.getSync('//C.com');  
  
console.log(A);  
console.log(B);  
console.log(C);
```

**감사합니다**

---