



# Instagram clone coding

Front - end part



## Kind-tiger (심 선범)

- 제주코딩베이스캠프 7기 대상
- 탈잉 튜터
- 카인드패밀리 캐릭터 제작
- 디자인나스 프로그래밍 강사

# How to ?

인스타그램 직접 코딩 x  
Front 에서 필요한 핵심 이론 정리

1. html 태그의 요소
2. CSS 의 규칙
3. JavaScript 이론
4. 인스타그램 뜯어보기!

# Final

프론트엔드 별거없네..?

# 1. HTML 태그의 요소

( 인라인 & 블록 )

42953

홀수 or 짝수

Inline ( span )	Block ( div )
가로로 배치	세로로 배치
width, height 적용 x	width, height 적용 o
창크기 영향 받음	창크기 영향 받지 않음
Web ranking	별도의 width지정이 없을경우 Width : auto;

# Floors

( 층수에 대한 개념 )

링크



## 2. CSS 규칙

Cascading style sheets

cascading

static (default)	relative	absolute	fixed
기준점	자신	부모	브라우저
층수	1	2	2
가운데정렬	Margin: 0 auto;	left: 50%; top: 50%; margin-left: -width/2 px; margin-top: -height/2 px;	
필수작성요소	x	x축 1개 y축 1개	

```
“ I’m your mother ”
<browser>

  <html>

    <body>

      <1>      “ Mommy...ㅏㅏ ”

        <2> </2>

      </1>

    </body>

  </html>
```

# Flex box

## ( 플렉스박스 )

flexbox는 뷰포트나 요소의 크기가 불명확하거나 동적으로 변할 때에도 효율적으로  
요소를 배치, 정렬, 분산할 수 있는 방법을 제공하는 CSS3의 새로운 레이아웃 방식이다.  
flexbox의 장점을 한 마디로 표현하면 '복잡한 계산 없이 요소의 크기와 순서를 유연하게 배치할 수 있고,  
정렬, 방향, 순서, 크기 등을 유연하게 조절할 수 있기 때문에 별도의 분기 처리를 줄일 수 있다.

# Flex 속성

( attribute )

parents

( **flex-container** )

- \_ flex-direction
- \_ flex-wrap
- \_ justify-content
- \_ align-items
- \_ align-content

child

( **flex-item** )

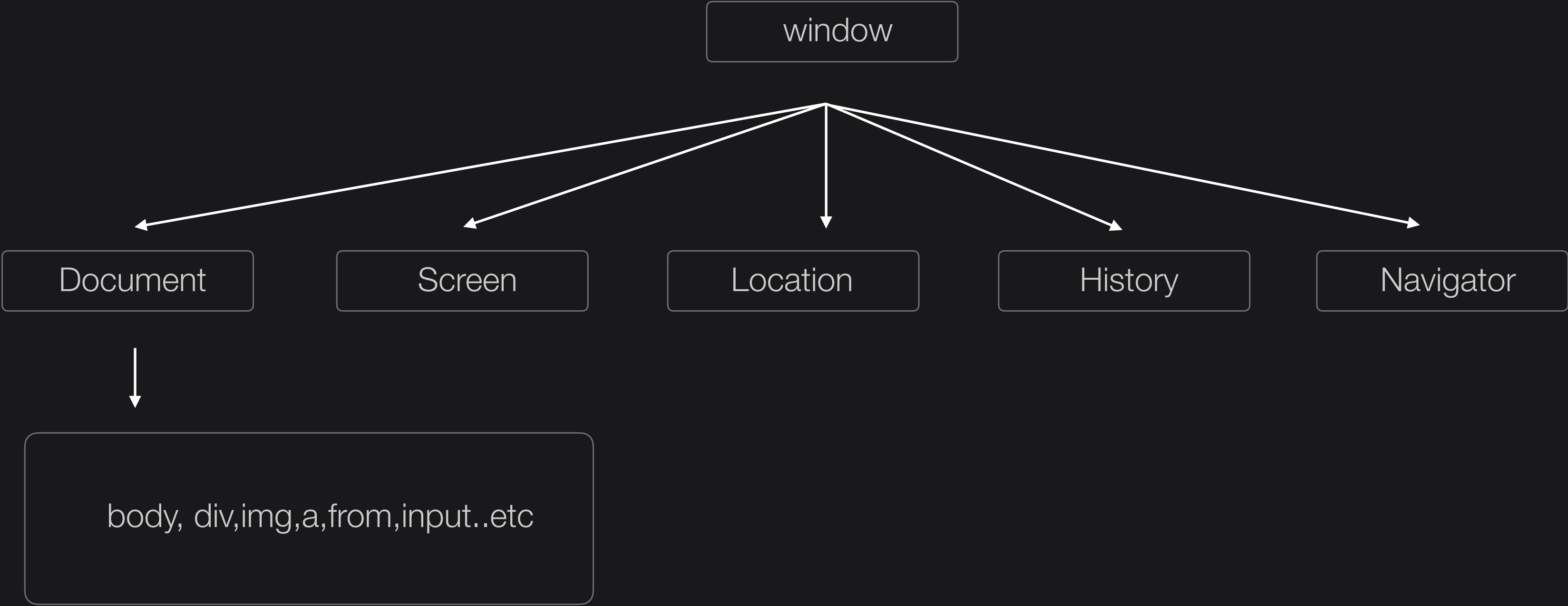
- \_ flex
- \_ flex-grow
- \_ flex-shrink
- \_ flex-basis
- \_ Order

# 3. JavaScript

2015 ECMA - 262 ( ES 6 )

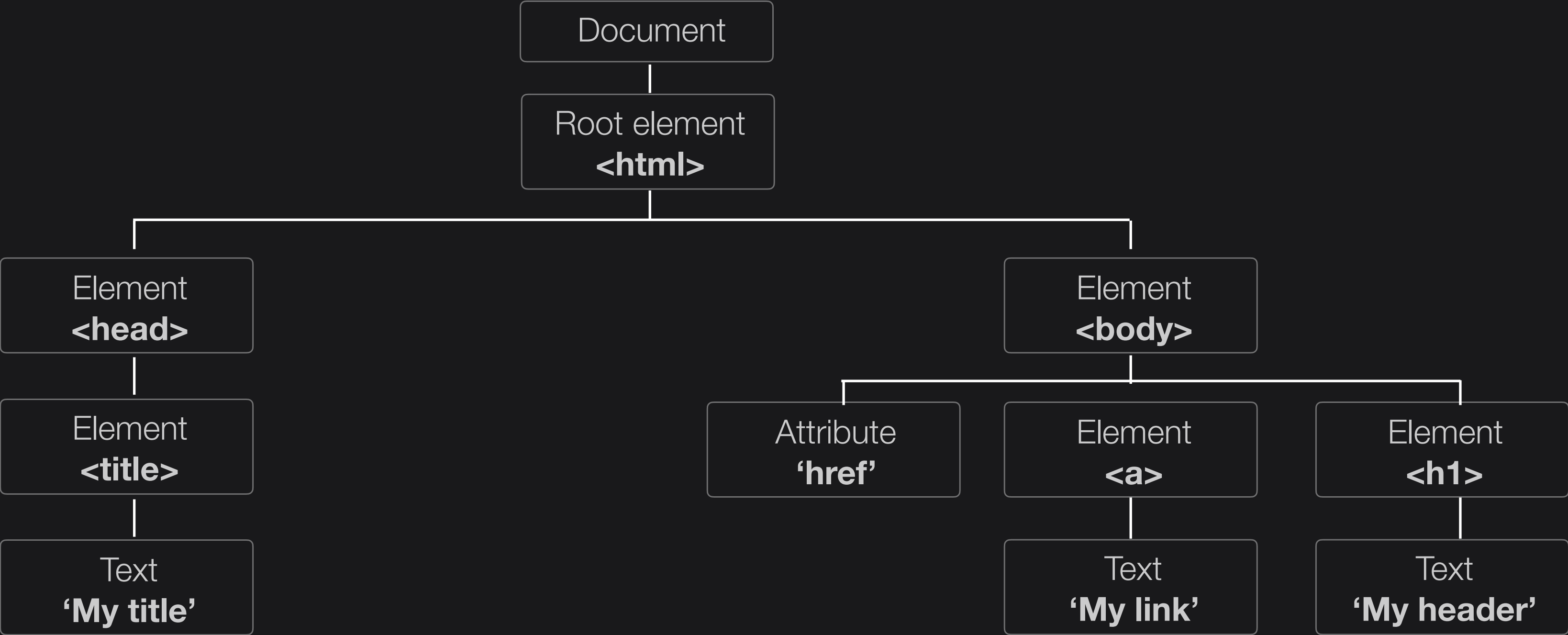
# BOM

(Browser Object Model)



# DOM

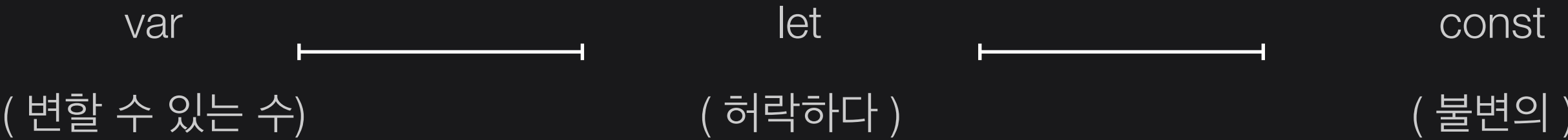
(Document Object Model)





# Variable & Constant

(변수 & 상수)



# COMPOSITION

( 구성요소 )

Expresstion  
( 표현식 )

+

Statement  
( 문장 )

+

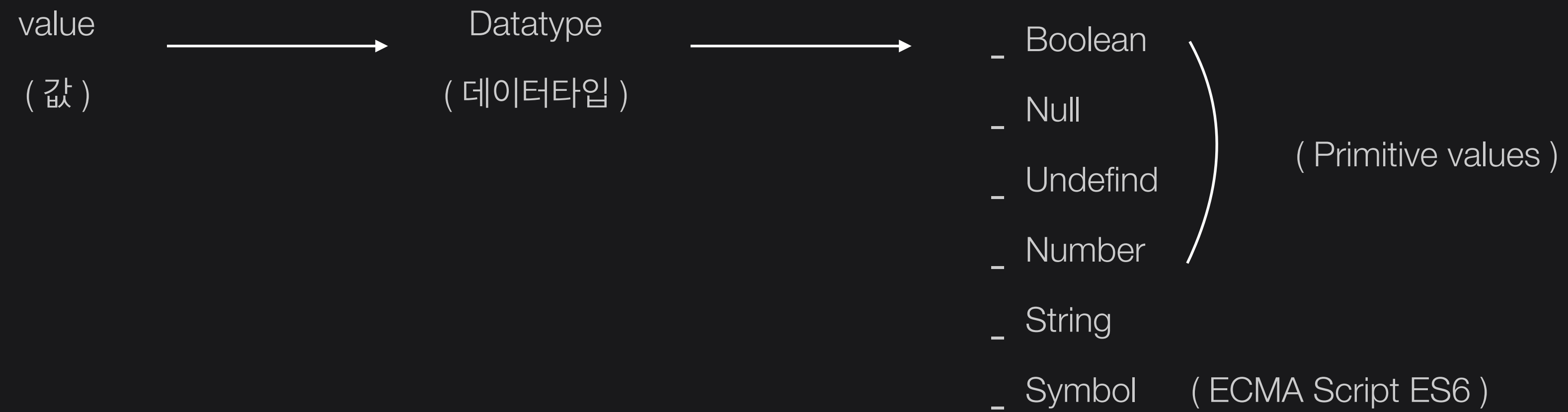
Function  
( 함수 )

=

Program  
( 프로그램 )

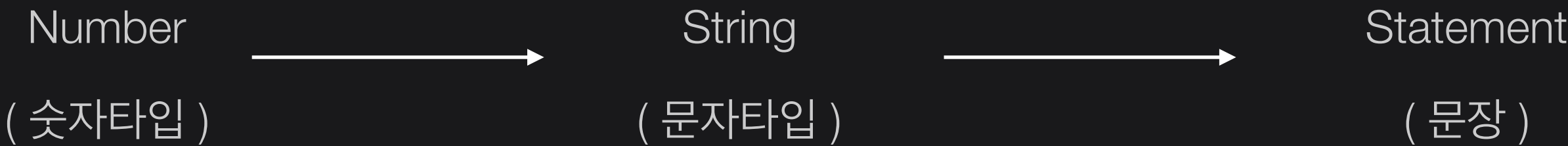
# EXPRESSION

(익스프레션)



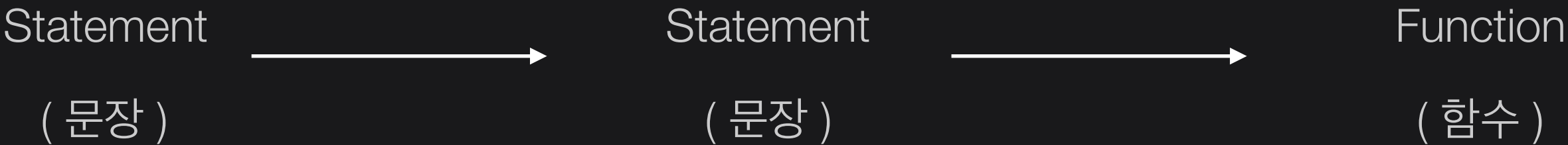
# STATEMENT

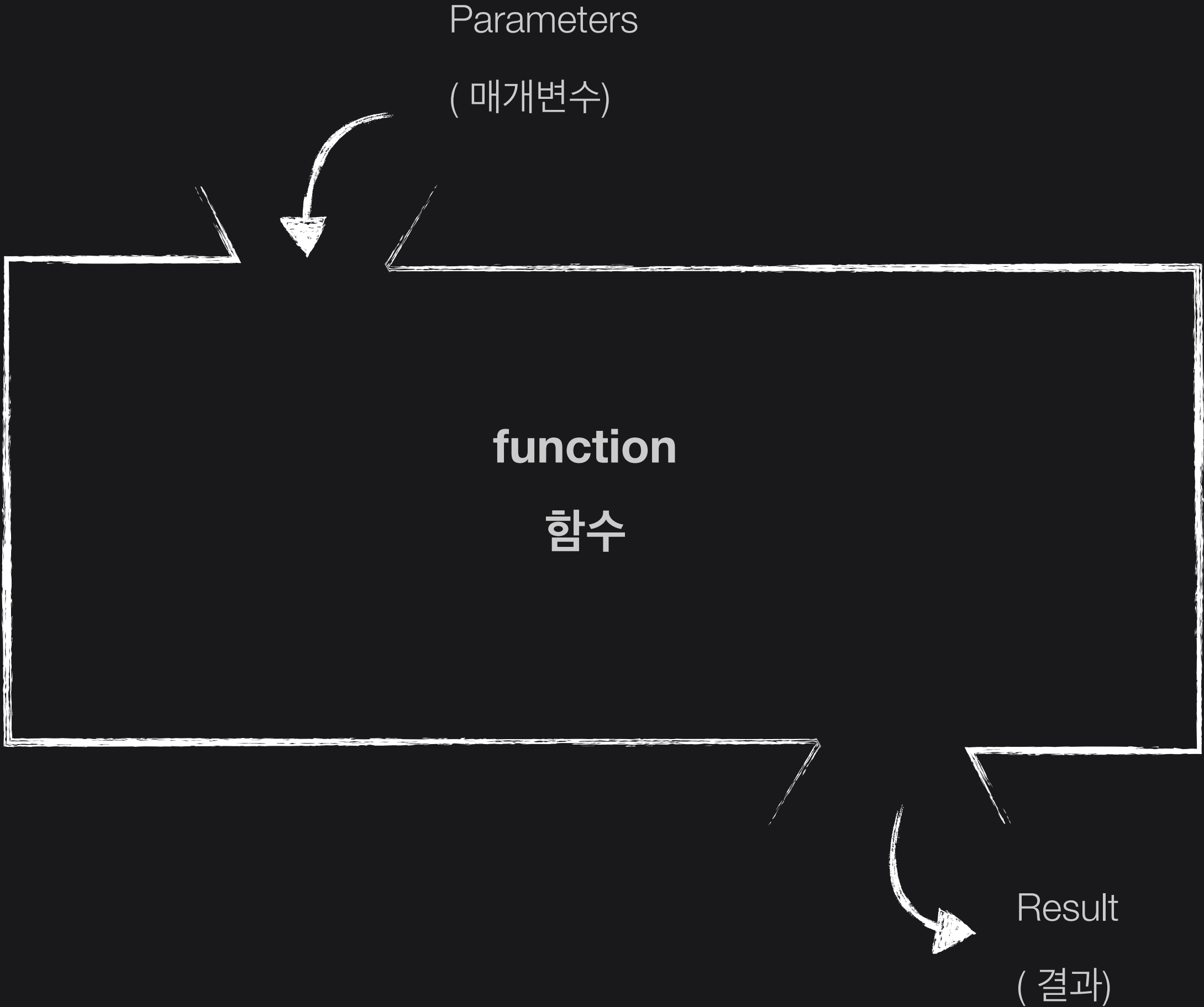
(스테이트먼트)



# FUNCTION

(함수)





# 용어

( dragon fish )

Assignment  
( 할당 )



argument  
( 인자 )



Parameters  
( 매개변수 )

# 조건문

( if door )

어떤 특수한 상황에 맞는 특정 작업을 수행할때 사용하는 “문” 법

## if 문

if  
else if  
else

## switch case 문

switch  
case  
break



# if 문

( if door )

**if** ( condition ) { ( statement1 ) } **else ...**

↓                      ↓

조 건                      명 령 문

true & false



**if** ( condition ) {  
    ( statement1 )  
**}** **else** {  
    ( remainder )  
**}**

# switch

( switch case)

```
switch ( expression ) {
```

```
    case : ( value1 )  
        break ;
```

```
    case : ( value2 )  
        break ;
```

```
    default : ( remainder )
```

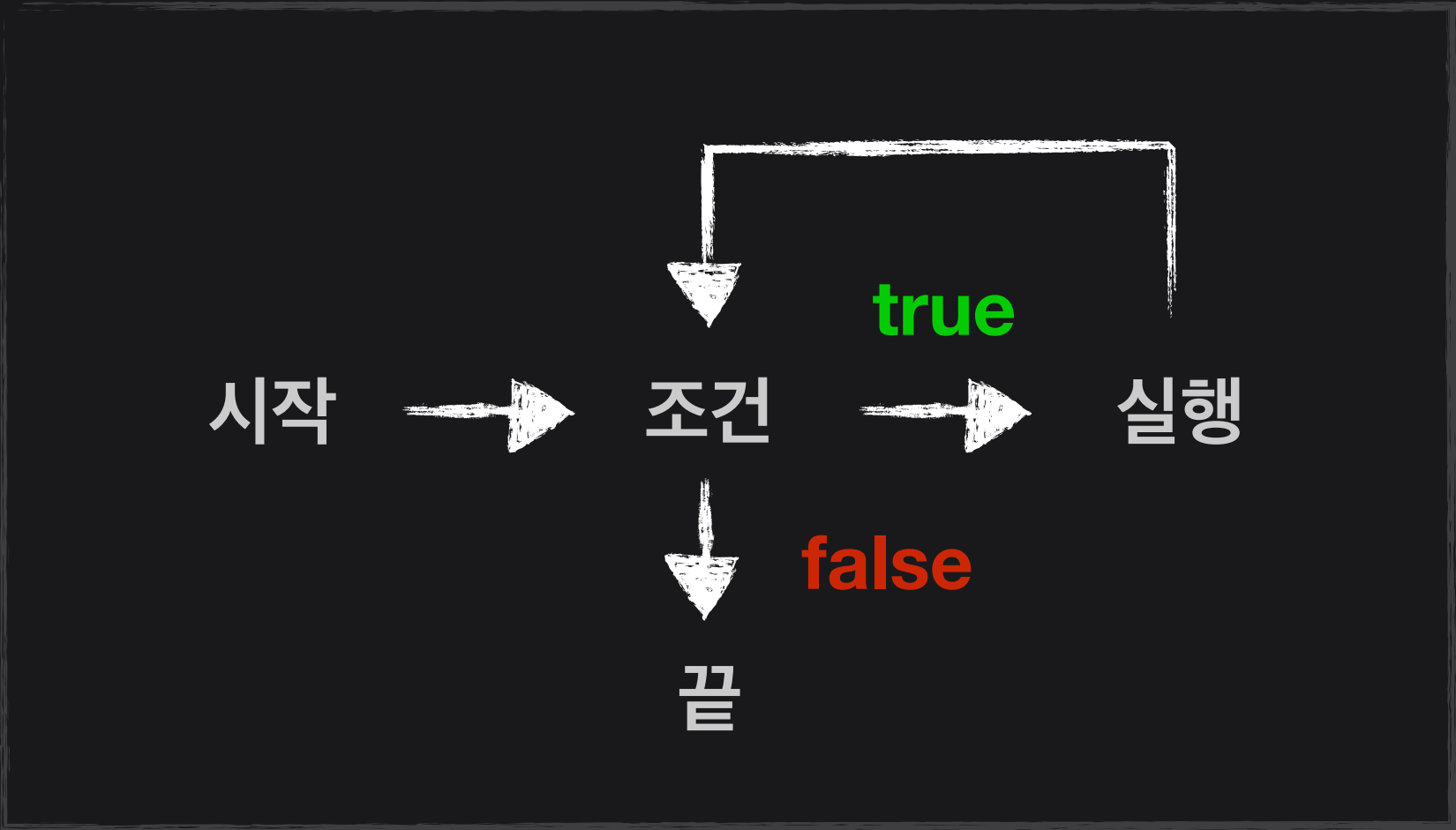
```
}
```

# 반복문

( loops )

특정 작업을 일정 이상의 값으로 반복설정

for



while

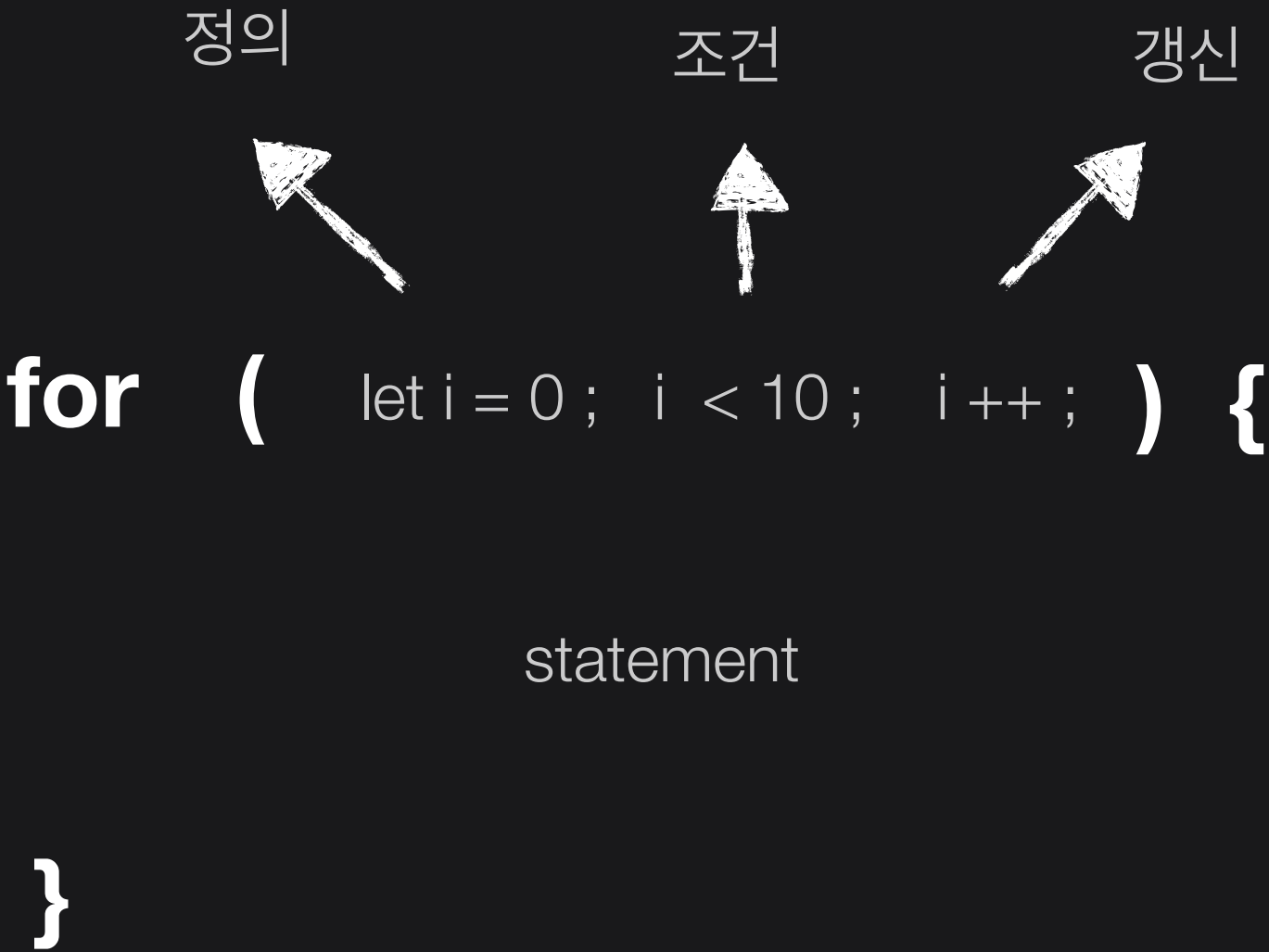
# while

( 반복문 )

```
let i = 0  
while( ( true ) ) {  
    Statement  
  
    i ++  
}
```

# for

( 4 )



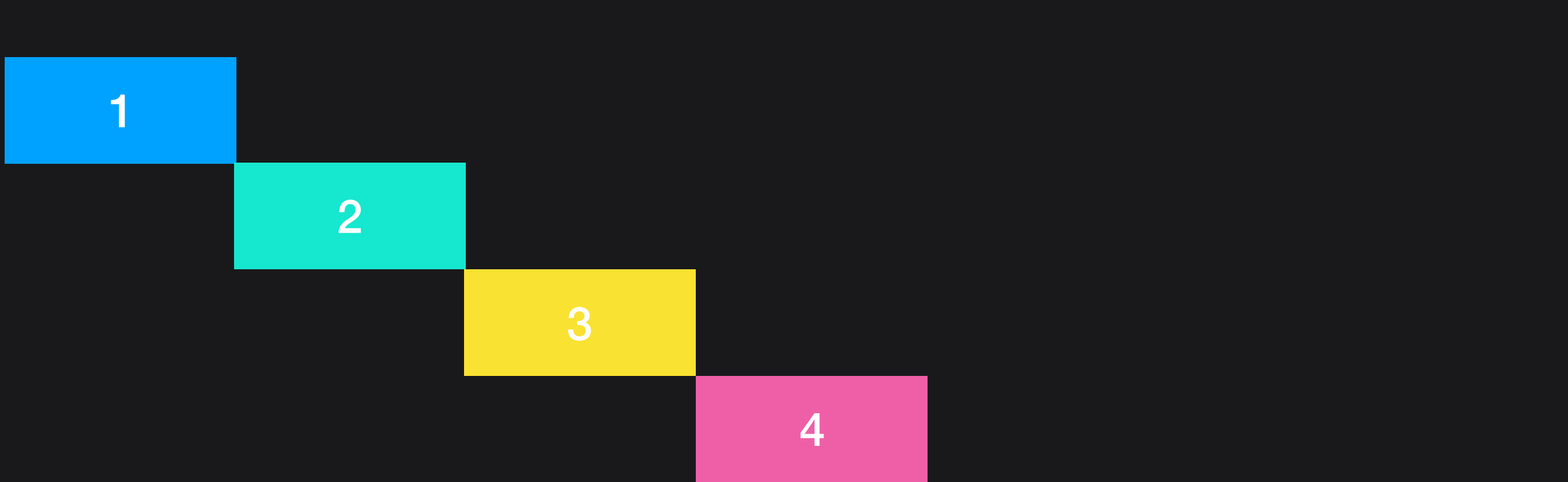
# event listener

( 이벤트 리스너 )

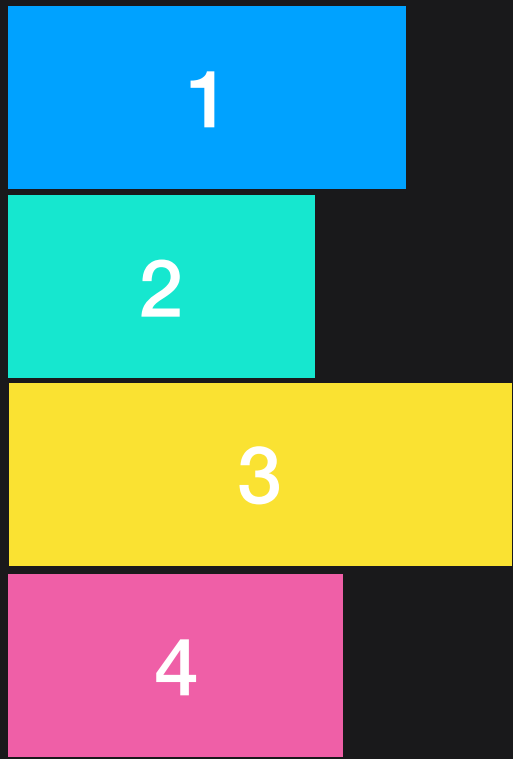
# 비동기통신 ( **ajax** )

( Asynchronous Javascript And Xml )

동기적  
( sync )



비동기적  
( async )





**Type :** get, post 등의 전송 방식을 결정

**Url :** 접근할 url을 입력

**Data :** 전달할 파라미터 값을 설정

**dataType :** 파싱할 파일의 형태를 입력 ex) json, html, xml

**Success :** 성공할 경우 불러올 콜백함수, 반환되는 코드 ( html, xml)기타 등등

**Error :** 실패할 경우 불러올 콜백함수

**Complete :** 성공 또는 실패 뒤에 실행할 콜백함수

# insertAdjacentHTML

( mdn 문서 )

# Final

프론트엔드 별거없네..?

쌔유



