

JavaScript (ES6) For Framework

4. JS를 움직이는 데이터의 종류

“

자료형 (1) - 일반적 자료형

자료형 (2) - 객체

자료형 (3) - 새롭게 추가된 자료형 : Symbol

자료형

Data

숫자?

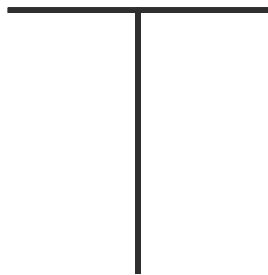
문자열?

Boolean?



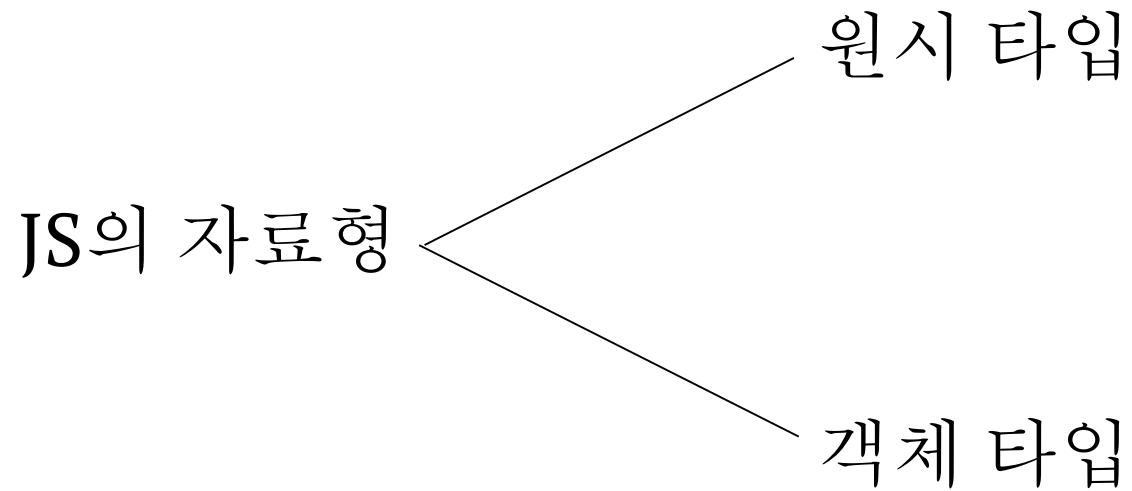
자료형

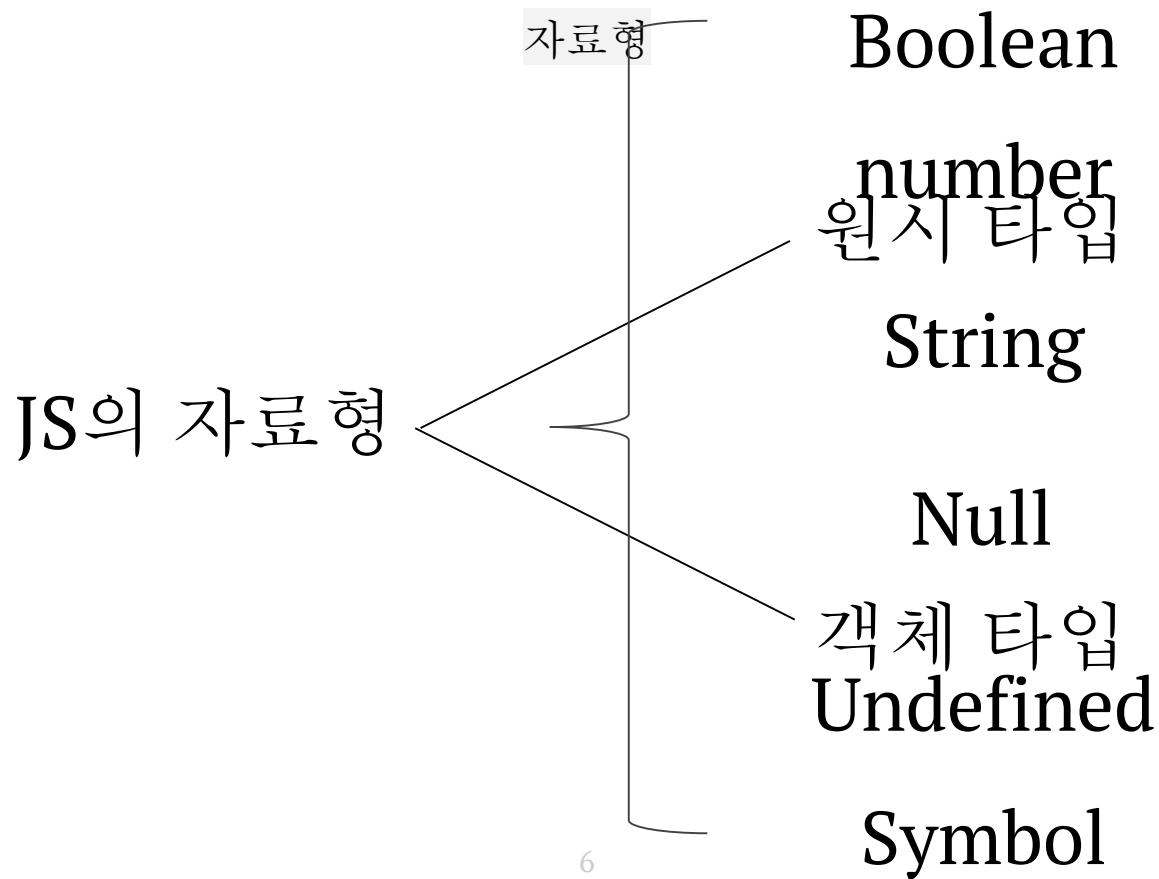
Data 종류 식별 → Data 처리



프로그래밍 단계에서 미리 처리하겠다! = 자료형

자료형





자료형

typeof 연산자

```
var a = 100;
```

Elements Console Sources Network Performance Memory Application Security Audits

top Filter Default levels ▾

Console was cleared

⏪ undefined

> var a = 1;

⏪ undefined

> typeof a;

⏪ "number"

> var b = "문자열";

⏪ undefined

> typeof b;

⏪ "string"

> var c = true;

⏪ undefined

> typeof c;

⏪ "boolean"

> var d = undefined;

⏪ undefined

> typeof d;

⏪ "undefined"

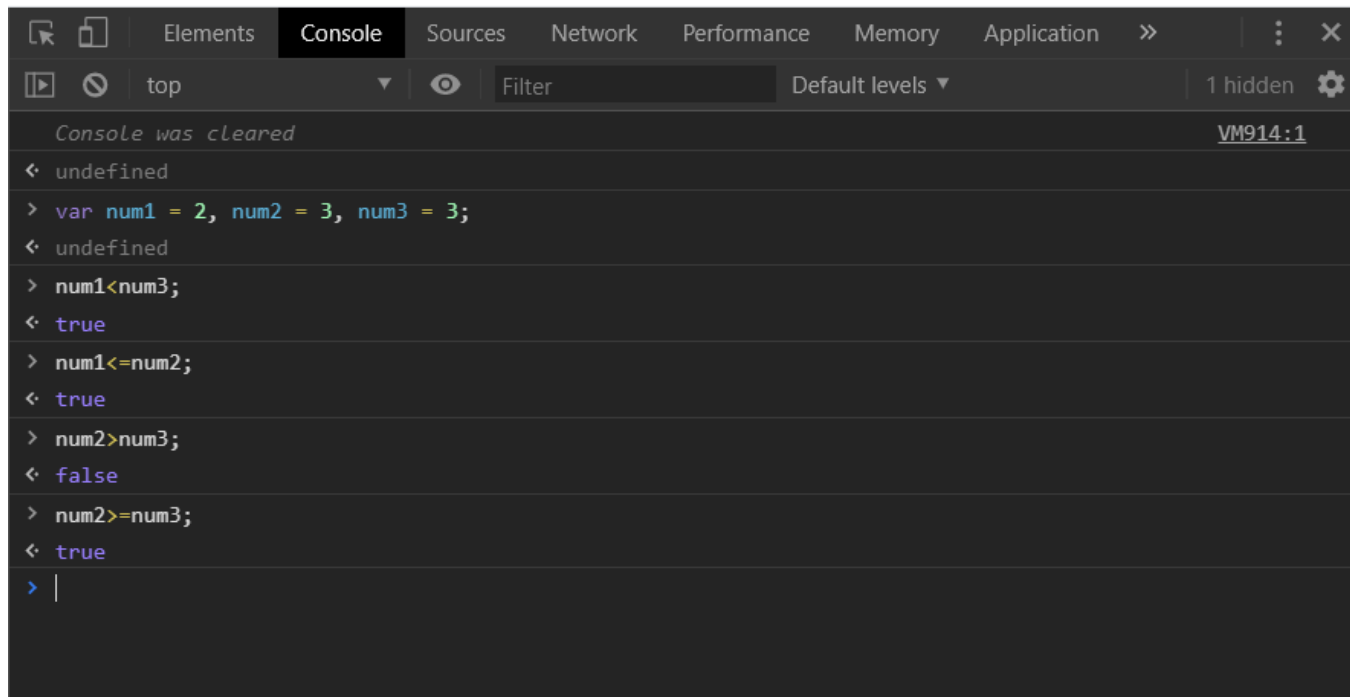
>

Boolean

참 혹은 거짓

```
var bool1 = true;  
var bool2 = false;  
  
console.log(typeof bool1); // boolean  
console.log(typeof bool2); // boolean
```

Number



The screenshot shows a web browser's developer console with the 'Console' tab selected. The console displays the following sequence of commands and results:

```
Console was cleared VM914:1
< undefined
> var num1 = 2, num2 = 3, num3 = 3;
< undefined
> num1<num3;
< true
> num1<=num2;
< true
> num2>num3;
< false
> num2>=num3;
< true
> |
```

Number

동등 연산자, 일치 연산자

동등하다 ==

동등하지 않다 !=

일치하다 ===

일치하지 않다 !==

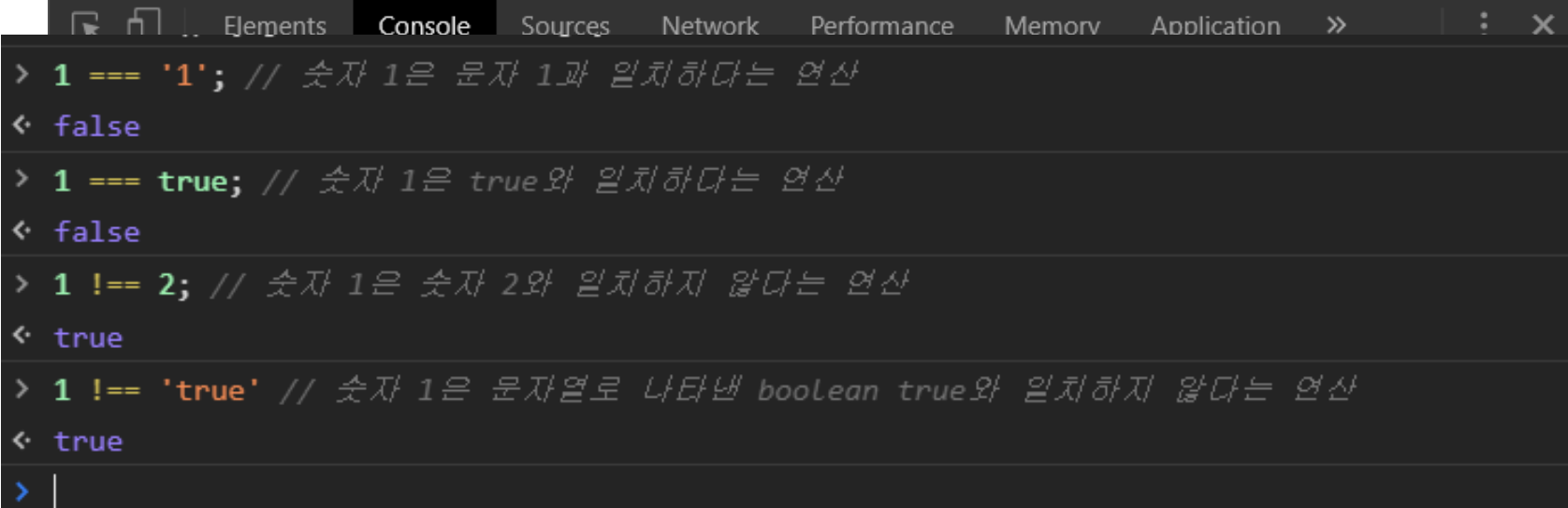
Number

동등 연산자, 일치 연산자

동등연산자 : 자동으로 자료형을 일치시킨 뒤 비교

일치연산자 : 자료형도 비교 대상에 포함

Number



The screenshot shows a web browser's developer console with the 'Console' tab selected. It displays four lines of JavaScript code and their corresponding output results. The code uses strict equality (===) and strict inequality (!==) operators to compare the number 1 with various values. The comments are in Korean, explaining the purpose of each comparison.

```
> 1 === '1'; // 숫자 1은 문자 1과 일치하다는 연산
< false

> 1 === true; // 숫자 1은 true와 일치하다는 연산
< false

> 1 !== 2; // 숫자 1은 숫자 2와 일치하지 않다는 연산
< true

> 1 !== 'true' // 숫자 1은 문자열로 나타낸 boolean true와 일치하지 않다는 연산
< true

> |
```

“

가급적 **일치 연산자**를 사용해주세요!

String

문자열

String

문자들의 배열



String

0	1	2	3	4
1	3	5	7	9

0	1	2	3	4
가	나	다	라	마

String

문자열은 한 번 선언되면
그 문자열 자체는 변경 불가능



Elements

Console

Sources



Network



Performance

Memory


Application

»





top




Filter

Default levels

▼

1 hidden



Console was cleared

VM4508:1

< undefined

> var name = "강민철";

< undefined

> console.log(name[0]);

강

VM4642:1

< undefined

> console.log(name[1]);

민

VM4665:1

< undefined

> console.log(name[2]);

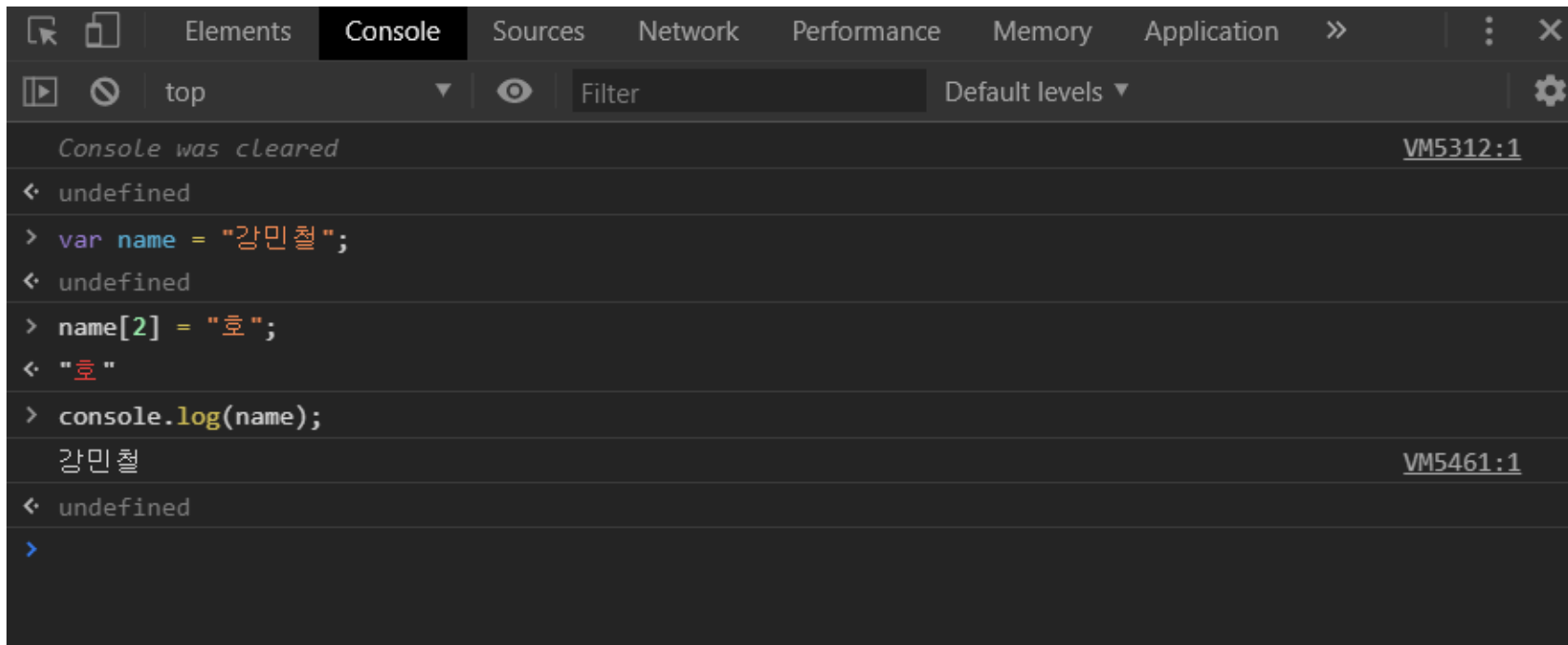
철

VM4676:1

< undefined

> |

String



The screenshot shows a web browser's developer console with the 'Console' tab selected. The console displays the following sequence of events:

- A message: *Console was cleared* (source: VM5312:1)
- Execution of `undefined` (source: VM5312:1)
- Execution of `var name = "강민철";` (source: VM5312:1)
- Execution of `undefined` (source: VM5312:1)
- Execution of `name[2] = "호";` (source: VM5312:1)
- Execution of `"호"` (source: VM5312:1)
- Execution of `console.log(name);` (source: VM5312:1)
- Output of the string `강민철` (source: VM5461:1)
- Execution of `undefined` (source: VM5461:1)
- Execution of `>` (source: VM5461:1)

String

문자열을 다루는 메서드

length	문자열의 길이 반환하는 메서드
search	문자열 중에서 문자열 찾는 메서드
slice	문자열 일부 추출하는 메서드
replace	문자열 일부를 바꾸는 메서드
concat	문자열끼리 합치는 메서드

등등..

null

아무 값도 갖지 않음을 뜻하는 자료형

대소문자 유의

NULL (X)

Null (X)

null (o)

```
> name = Null;  
✖ ▶ Uncaught ReferenceError: Null is not defined  
  at <anonymous>:1:1  
  
> name = NULL;  
✖ ▶ Uncaught ReferenceError: NULL is not defined  
  at <anonymous>:1:1  
  
> name = null;  
◀ null
```

null

null 타입 검사할 때는 === 를 사용하기

```
< undefined
> var a = null; // a라는 변수에 null을 넣기
< undefined
> typeof a; // a의 자료형을 검사하면 null이 나와야 하는데 "object"가 나옴
< "object"
> var a = null; // null 타입 검사는 일치 연산자를 활용할 것!
< undefined
```

undefined

변수를 선언하고 값 할당을 안 한 경우
존재하지도 않은 객체로 접근했을 경우

null vs undefined

명시적으로 없음을 나타내는 null

VS

진짜 아직 없는 undefined

추가 학습 자료

원시 자료형을 유용한 메서드로 감싸는
Wrapper 객체

다음강의

Section 4.

JS에서는 온 사방이 객체

Lecturer Github : <https://github.com/kangtegong/>

수업자료링크 : <https://github.com/kangtegong/develop-javascript>

Email : tegongkang@gmail.com