

[JS] 변수, 자료형, 연산자

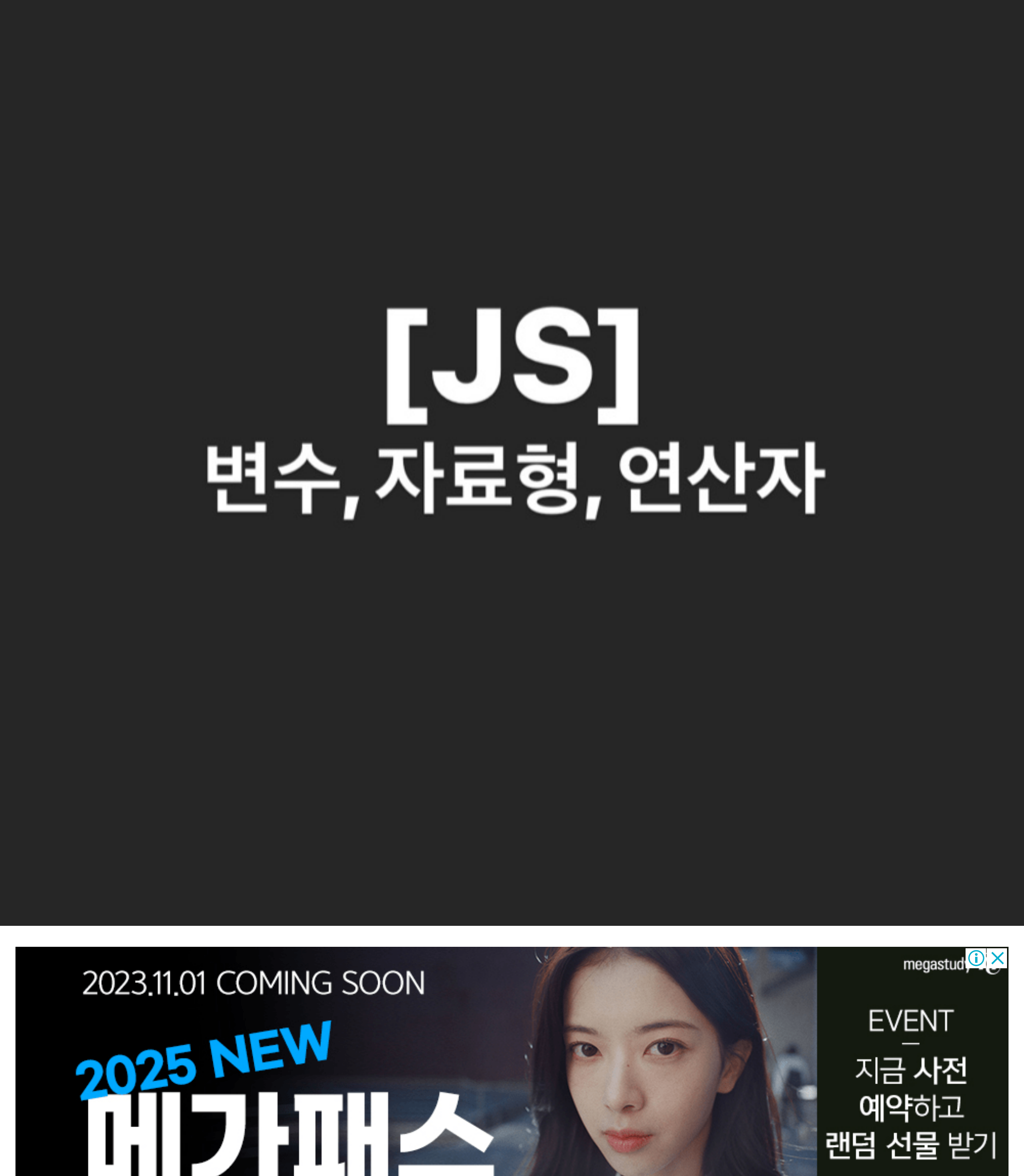
fttxld3 · 2022년 1월 14일

js 변수 연산자 자료형

JS

▼ 목록 보기

9/20



1. 변수

= 들어올수있기에 변할 수 있는 값(재할당가능)

변수를 선언하는 규칙 세 가지

- 이름은 의미있게 짓는다
- 여러 단어를 연결한 변수 이름을 낙타 모양으로 만들어 준다
- 변수 이름의 첫 글자는 반드시 문자나 밑줄(`_`), 달러 기호(`$`)로 시작해야 한다. (달러기호는 잘 사용하지않음)

변수 선언과 값/식 할당

- let 다음에 변수 이름을 적어서 변수를 선언
- 변수 오른쪽에 = 기호를 붙이고 오른쪽에 저장할 값이나 식을 작성 (변수 선언과 값 할당을 동시에 할 수도 있음 `let age= 20;`) (변수는 선언과 동시에 값할당을하여 초기화하는 습관을 가져야한다!!)

```
function age() {  
  //함수식이 들어감  
}
```

함수 안 괄호로 받는 것을 매개변수=파라미터=인자값=아규먼트

2. 자료형(Data Type)

= 컴퓨터가 처리하는 자료의 형태

1)기본형(Primitive Type) : 더이상 나뉘질 수 없는 single item

= number, bigint, string, boolean, undefined, null, symbol

- number(숫자) = 따옴표 없이 표기한 숫자를 나타냄
 - 정수(소수점X, - 표현 방법에 따라 10진수, 8진수, 16진수)
 - 실수(소수점 O, - 자바스크립트에서는 정밀한 실수 계산을 못 함)
- bigint = 아직사용하지말것
 - `const bigint = 1234567890123456789012345678901234567890n;`
 - 숫자의 표현범위 : `-2**53 ~ 2**53`
- string(문자형) = 적은따옴표나 큰따옴표로 묶어 나타냄
 - 숫자도 따옴표로 묶으면 문자형이 됨.
 - 따옴표 안에 따옴표를 넣어야 할 경우 `' '` , 또는 `" "` 처럼 사용
- boolean(논리형) = 참(true),거짓(false) 두 값만 가지고있는 유형
 - 조건을 확인할 때 많이 사용
 - 거짓(false) = 0, null, undefined, NaN, "(빈문자열)"
 - 참(true) = any other value
- undefined = 자료형을 지정하지 않았을 때의 상태 (예) 변수를 선언만하고 값을 정의하지않았을때
 - 변수가 undefined은 ' 처음부터 변수에 값이 할당되지 않았다 ' 는 의미
- null = 값이 유효하지 않을 때의 유형(빈값을 할당)
- symbol = create unique identifiers for objects
 - map 등 식별자가 필요하거나 동시다발적으로 콘크리트하게 일어날 수 있는 코드에서 우선순위를 주고싶을때 정말 고유한 식별자가 됨

2)복합형

- array(배열) = 하나의 변수에 여러값을 저장
 - 배열의 인덱스(index)는 0부터 시작
 - 배열에 있는 값을 가져오려면 배열 이름과 대괄호(`[]`) 안에 인덱스 사용.
 - .length로 길이를 구할 수 있다

```
let season = ["봄", "여름", "가을", "겨울"];  
season[0] //봄
```

- object(객체) = 함수와 속성이 함께 포함된 유형
 - 여러 자료를 중괄호(`{ }`)로 묶은 것
 - 키(key)와 값(value)을 한 쌍으로 여러 자료 저장

```
let info = {  
  firstName: "Hyun",  
  lastName: "Lim",  
  age: 25,  
  address: "Seoul"  
}
```

2-1. 자바스크립트 자료형의 특징

- 느슨한 자료형 체크(weak datatype check) -> typescript로 보완된것이 나옴
- 자바스크립트는 미리 변수의 자료형을 지정하지 않음
- 변수를 지정하고 원하는 값을 할당만 하면 됨

3. 연산자

- 사칙연산자 = `+`, `-`, `*`, `/`
- 나머지연산자 = `%`
- 증감연산자 = `++`, `--` (1만큼 증가,감소시킴)

- 전위 증가연산자 :;이 끝나기 전에 이미 증가가 되어있음
- 후위 증가연산자 :;이 끝난 후 증가가 됨

```
let a = 10;  
console.log(a++); // 10  
console.log(a); // 11  
console.log(++a); // 12
```

- 할당 연산자(복합대입 연산자) = `+=`, `-=`, `*=`, `/=`, `%=` (예)`a%=b` `a=a%b`

- 변수에 값을 할당하는 연산자
- 사칙 연산자와 조합해서 사용할 수 있음

5. 연결 연산자

- 문자열과 문자열을 연결
- + 기호 사용

6. 형변환

- 숫자형과 문자형을 더하면 숫자를 문자열로 인식함
- 곱하거나 나누기, 나머지 연산에서는 문자형 자료를 모두 숫자로 자동 인식함

※ 연산자와 피연산자
`currentYear - birthYear + 1`
`currentYear, birthYear`, 1은 연산 대상이 되기 때문에 ‘**피연산자**’라고 부름
피연산자를 제외한 더하기, 빼기 같은 것을 ‘연산자’라고 부름

Hyun
FrontEnd Developer (with 구글링)

이전 포스트
[JS] 자바스크립트란?

다음 포스트
[JS] 조건문, 반복문

애니메이션 2기 방영 기념
신인 모험가의 단독 행동은 위험천만 클린에
소속되어 힘을 길러라
고블린 슬레이어
열기 >

0개의 댓글

댓글을 작성하세요

댓글 작성

관심 있을 만한 포스트

Python 출력, 입력, 연산자

python 기초 파이썬의 출력 >print("hello")
>print("hello") 문자열이나 문자 출력은 줄바꿈표
(\n) 또는 쉼표줄표(,)이다. >print(1) 아무것도 불리
지 않고 숫자만 쓰면 정수나, 실수를 출력 할 수 있
다. >print(1+...
2020년 4월 9일 · 0개의 댓글
by bungouk6829

JavaScript - Logical Operator

JavaScript - 논리연산자(Logical Oper...

논리연산자 (Logical Operator) 논리 연산자는 보
통 Boolean값과 함께 쓰이며, Boolean값을 반환하
며 JavaScript가 지원하는 논리 연산자는 아래와 ...
2019년 8월 21일 · 0개의 댓글
by surim014

[JavaScript] - null 병합 연산자 '??'

null 병합 연산자(nullish coalescing operator)
??를 사용하면 짧은 문법으로 여러 피연산자 중 그 값
이 '확정되어 있는' 변수를 찾을 수 있다. a ?? b의 ...
2021년 1월 20일 · 0개의 댓글
by seize

C# ?? 및 ??=, ?. 연산자

C#에서 C(피연산자가 null 이 아닌 것으로 평가되었
을 때만 멤버 액세스 ? 또는 오소 액세스 ?[]) 연산을
피연산자에게 적용하며, 그렇지 않으면 null 을 반 ...
2020년 6월 16일 · 0개의 댓글
by jinuku

[13강] 후위 표기 수식 계산

(A + B) * (C + D) 연산자가 피연산자들의 사이에
위치 AB + CD + * 연산자가 피연산자들의 뒤에 위치
후위 표현식을 왼쪽부터 한 글자씩 읽어서 피연산자 ...
2020년 7월 10일 · 0개의 댓글
by inyong_pang

[Flutter] Dart 문법 정리 (type / op...

내가 보려고 정리하는 다투 문법

2021년 7월 15일 · 1개의 댓글
by dosilv

Unit 5-9. 숫자계산, 변수, 출력, 비교연산...

정수 int 실수 float 함수 complex 산술연산자 변수 =
값 (문자열은 ") 변수 1, 변수 2 = 값 1, 값 2 파이썬은
None 자료형. 타 언어는 null 주: 할당 연산자 사용
시에는 값이 들어있는 변수 사용해야함(input()) 변수
= input()

2020년 2월 28일 · 1개의 댓글
by joygoround

스택의 응용 - 수식의 후위 표기법

[12강] 스택의 응용 - 수식의 후위 표기법(P...

연산자가 피연산자들의 사이에 위치(A + B) * (C +
D) 연산자가 피연산자들의 뒤에 위치 AB + CD + *
중위 (A + B) * (C + D) 후위 A B + C * 이는 괄호는 스...

2020년 7월 10일 · 1개의 댓글
by inyong_pang

자바스크립트 null 병합 연산자와 옵셔널 체...

Nullish coalescing operator (null 병합 연산자)
nullish coalescing operator ??는 nullish한 값
(null 혹은 undefined)을 판별하여 or 연산을 하...

2020년 6월 27일 · 3개의 댓글
by yejinh

JavaScript :: 증감 연산자(+++, --) Feat...

JavaScript의 증감 연산자(increment++,
decrement--)와 함께, 전위 연산자(prefix
operator), 후위 연산자(postfix operator)에 대...

2021년 5월 26일 · 3개의 댓글
by lamhayoung

Javascript Object literal

[JavaScript] 객체 리터럴(object literal)

자바스크립트는 객체 기반 프로그래밍 언어이며, 원
시 값을 제외한 나머지 값은 모두 객체이다. 객체는 0
개 이상의 프로퍼티로 구성된 집합으로, 프로퍼티...

2021년 11월 11일 · 2개의 댓글
by nxnaxx