

07장. 연산자

- 연산자란?

하나 이상의 표현식을 대상으로 산술, 할당, 비교, 논리, 타입, 지수 연산 등을 수행해 하나의 값을 만드는 것

- 피연산자란?

연산의 대상, 값으로 평가될 수 있는 표현식, 피연산자와 연산자의 조합으로 이뤄진 연산자 표현식도 값으로 평가될 수 있는 표현식

즉, 피연산자 = 명사의 역할 / 연산자 = 동사의 역할

7.1 산술 연산자

1. 피연산자를 대상으로 수학적 계산을 수행해 새로운 숫자 값을 만든다.
2. 산술 연산이 불가능한 경우, NaN 반환
3. 피연산자의 개수에 따라 이항 산술 연산자와 단항 산술 연산자로 구분

- 이항 산술 연산자

2개의 피연산자를 산술 연산하여 숫자 값을 만든다.

어떤 산술 연산을 해도 피연산자의 값이 바뀌는 경우는 없고 언제나 새로운 값을 만든다. (부수 효과 X)

+, -, *, /, %

- 단항 산술 연산자란?

1개의 피연산자를 산술 연산하여 숫자 값을 만든다.

이항 산술 연산자와 달리 증가/감소(++/-)연산자는 피연산자의 값을 변경하는 부수 효과가 있다.

1. 피연산자 앞에 위치한 전위 증가/감소 연산자는 먼저 피연산자의 값을 증가/감소시킨 후, 다른 연산을 수행
2. 피연산자 뒤에 위치한 후위 증가/감소 연산자는 먼저 다른 연산을 수행한 후, 피연산자의 값을 증가/감소시킨다.

```
var x = 5, result;

result = x++; //선할당 후증가
console.log(result, x); // 5 6

result = ++x; // 선증가 후할당
console.log(result, x); // 7 7

result = x--; // 선할당 후감소
console.log(result, x); // 7 6

result = --x; //선감소 후할당
console.log(result, x); // 5 5
```

++, --, +, -

• 문자열 연결 연산자란?

+연산자는 피연산자 중 하나 이상이 문자열인 경우 문자열 연결 연산자로 동작
그 외의 경우는 산술 연산자로 동작

```
//문자열 연결 연산자
'1' + 2; // '12'
1 + '2'; // '12'

//산술 연산자
1 + 2; //3

//true는 1로 타입 변환
1 + true; // 2

//false는 0으로 타입 변환
```

```

1 + false; // 1

//null은 0으로 타입 변환
1 + null; // 1

//undefined는 숫자로 타입 변환되지 않는다.
+undefined; // NaN
1 + undefined; // NaN

```

암묵적으로 불리언 타입의 값인 true를 1로 타입을 강제 변환 후 연산을 수행

→ 암묵적 타입 변환/타입 강제 변환

7.2 할당 연산자

우항에 있는 피연산자의 평가 결과를 좌항에 있는 변수에 할당

할당 연산자는 좌항의 변수에 값을 할당하므로 변수 값이 변하는 부수 효과 존재

할당문은 값으로 평가되는 표현식인 문으로서 할당된 값으로 평가

7.3 비교 연산자

좌항과 우항의 피연산자를 비교한 다음 그 결과를 불리언 값으로 반환, if문이나 for문과 같은 제어문의 조건식에서 주로 사용

• 동등/일치 비교 연산자란?

좌항과 우항의 피연산자가 같은 값으로 평가되는지 비교해 불리언 값을 반환

동등 비교 연산자(==) : 느슨한 비교 / 일치 비교 연산자(===) : 엄격한 비교

동등 비교 연산자는 좌항과 우항의 피연산자를 비교할 때 먼저 암묵적 타입 변환을 통해 타입을 일치시킨 후 같은 값인지 비교

일치 비교 연산자는 좌항과 우항의 피연산자가 타입도 같고 값도 같은 경우에 한하여 true를 반환

→ NaN은 자신과 일치하지 않는 유일한 값 (주의!), 대신 Number.isNaN을 사용

- **Object.js란?**

예측 가능한 정확한 비교 결과를 반환

- **대소 관계 비교 연산자란?**

피연산자의 크기를 비교해 불리언 값을 반환

7.4 삼항 조건 연산자

조건식의 평가 결과에 따라 반환할 값을 결정, 자바스크립트의 유일한 삼항 연산자로 부수 효과 없음

삼항 조건 연산자 표현식은 값으로 평가할 수 있는 표현식인 문

조건식 **?** 조건식이 **true**일 때 반환할 값 **:** 조건식이 **false**일 때 반환할 값

첫 번째 피연산자가 true로 평가되면 두 번째 피연산자를 반환, 반대로 false로 평가되면 세 번째 피연산자를 반환

if-else문을 사용해도 삼항 조건 연산자 표현식과 유사하게 처리가능

→ 삼항 조건 연산자 표현식을 값처럼 사용 가능, if-else문은 값처럼 사용 불가능

7.5 논리 연산자

우항과 좌항의 피연산자를 논리 연산

1. **||**(논리합_OR)
2. **&&**(논리곱_AND)
3. **!**(부정_NOT)

7.6 쉼표 연산자

왼쪽 피연산자부터 차례대로 피연산자를 평가하고 마지막 피연산자의 평가가 끝나면 마지막 피연산자의 평가 결과 반환

7.7 그룹 연산자

소괄호로 피연산자를 감싸는 그룹 연산자는 자신의 피연산자인 표현식을 가장 먼저 표현

7.8 typeof 연산자

피연산자의 데이터 타입을 문자열로 반환

string, number, boolean, undefined, symbol, object, function 중 하나를 반환

null값을 연산해보면 object로 반환함(자바스크립트의 첫번째 버그!)

→ null타입인지 확인할 때는 일치 연산자(===) 사용 권장

선언하지 않은 식별자를 typeof로 연산해 보면 undefined를 반환함

7.9 지수 연산자

좌항의 피연산자를 밑으로, 우항의 피연산자를 지수로 거듭 제공하여 숫자 값을 반환

지수 연산자 도입 이전에는 Math.pow()메서드 사용