



5강 - 형 변환

형 변환(type casting)이란?

암묵적 형 변환

문자열 타입으로 변환

숫자 타입으로 변환

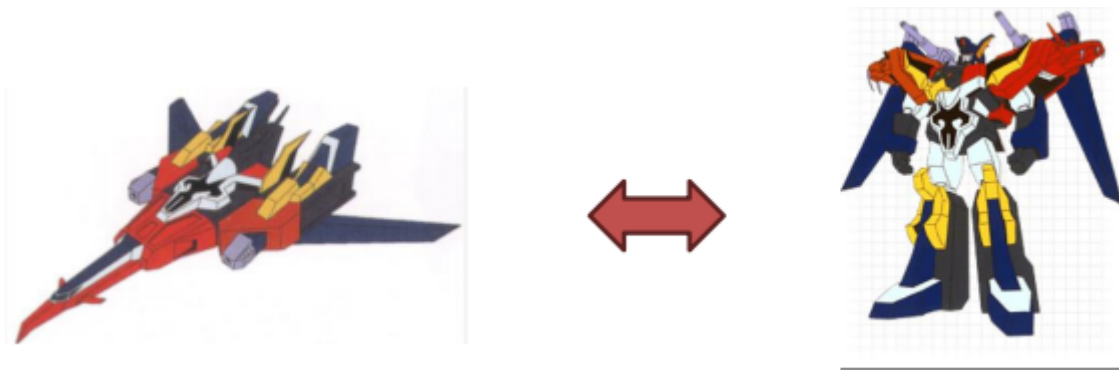
논리 타입으로 변환 - Truthy, Falsy

명시적 형 변환

문자열 타입으로 변환

숫자 타입으로 변환

형 변환(type casting)이란?



- 자바스크립트의 모든 값은 타입을 가지고 있습니다. 타입은 개발자의 의도에 따라 다른 타입으로 변환할 수 있습니다.
- 개발자의 의도와는 상관없이 표현식을 평가하는 도중에 엔진에 의해 암묵적으로 타입이 변환되기도 하는데 이를 **암묵적 형 변환**(implicit coercion)이라고 합니다.
- 개발자가 의도적으로 타입을 변환할 수도 있는데 이를 **명시적 형변환**(explicit coercion)이라고 부릅니다.

암묵적 형 변환

문자열 타입으로 변환

```
var ex = 1 + '2';
```

- 위 예시의 +연산자는 피연산자 중 하나가 문자열 타입이므로 문자열 연결 연산자로 동작합니다.

- 자바스크립트 엔진은 이 때 문자열이 아닌 정수 1을 문자열타입으로 **암묵적으로 변환**합니다.

```
//숫자 타입의 암묵적 문자열 변환
```

```
0 + ''    // -> "0"
```

```
10 + ''   // -> "10"
```

```
NaN + ''  // -> "NaN"
```

```
//논리 타입의 암묵적 문자열 변환
```

```
true + ''  // -> "true"
```

```
false + '' // -> "false"
```

숫자 타입으로 변환

```
var x = 10 - '3';
```

- 위 예시의 - 연산자는 뺄셈 산술 연산자이므로 문자열 3을 숫자로 **암묵적으로 변환**하여 7이라는 결과값을 만듭니다.

```

//문자열 타입의 암묵적 숫자 변환
+' '      // -> 0
+'0'      // -> 0
+'1'      // -> 1
+'hello'  // -> NaN

//논리 타입의 암묵적 숫자 변환
+true     // -> 1
+false    // -> 0

//null
+null     // -> 0

```

논리 타입으로 변환 - Truthy, Falsy

```

if ( ' ' ) console.log('OK');
if ( true ) console.log('OK');
if ( 0 ) console.log('OK');
if ( 'hello' ) console.log('OK');
if ( null ) console.log('OK');

```

- 'OK'가 몇 번이나 출력될까요??
- 정답은 2번째, 4번째 줄이 실행되어 2번 출력됩니다!
- 자바스크립트 엔진은 논리 타입이 아닌 값을 **Truthy값(암묵적으로 참으로 평가)** 또는 **Falsy값(암묵적으로 거짓으로 평가)**으로 구분합니다.
- **undefined, null, 0, NaN, ""(빈문자열)**은 모두 **false**로 판단합니다.

- 그 이외에는 모두 true로 판단합니다.

명시적 형 변환

문자열 타입으로 변환

- 문자열이 아닌 값을 문자열 타입으로 변환하는 방법

1. String 생성자 함수를 사용

```
var a = 10, b = 20;  
var result = String(a) + String(b);
```

result: 1020

2. toString 메서드를 사용

```
var x = 30, y = 40;  
var result = x.toString() + y.toString();
```

result: 3040

숫자 타입으로 변환

- 숫자가 아닌 값을 숫자 타입으로 변환하는 방법

1. Number 생성자 함수를 사용

```
var m = '10', n = '5.15';  
var result = Number(m) + Number(n);
```

result: 15.15

2. parseInt, parseFloat 함수를 사용

```
var m = '20', n = '3.14';  
var result = parseInt(m) + parseFloat(n);
```

result: 23.14

논리 타입으로 변환

- 논리 타입이 아닌 값을 논리 타입으로 변환하는 방법

1. Boolean 생성자 함수를 사용

```
Boolean('hello');    // -> true  
Boolean('');         // -> false  
Boolean(0);          // -> false
```

2. ! 부정 논리 연산자를 두 번 사용

```
!!'hello';    // -> true  
!!'';        // -> false  
!!0;         // -> false
```

단축 평가 (short circuit)

단축 평가 표현식	평가 결과
true anything	true
false anything	anything
true && anything	anything
false && anything	false

첫번째 truthy를 찾는 OR연산자

```
alert( 1 || 0 ); // 1 (1은 truthy임)  
  
alert( null || 1 ); // 1 (1은 truthy임)  
alert( null || 0 || 1 ); // 1 (1은 truthy임)  
  
alert( undefined || null || 0 ); // 0 (모두 falsy이므로, 마지막 값을 반환함)
```

첫번째 falsy를 찾는 AND연산자

```
// 첫 번째 피연산자가 truthy이면,  
// AND는 두 번째 피연산자를 반환합니다.  
alert( 1 && 0 ); // 0  
alert( 1 && 5 ); // 5  
  
// 첫 번째 피연산자가 falsy이면,  
// AND는 첫 번째 피연산자를 반환하고, 두 번째 피연산자는 무시합니다.  
alert( null && 5 ); // null  
alert( 0 && "아무거나 와도 상관없습니다." ); // 0
```

//OR 연산자

```
'Hello' || 'World'; // -> 'Hello'  
' ' || '안녕'; // -> '안녕'  
'잘가' || ' '; // -> '잘가'
```

//AND 연산자

```
'Hello' && 'World'; // -> 'World'  
' ' && '안녕'; // -> ''  
'잘가' && ' '; // -> ''
```