

Ch.6 데이터 타입

▼ 쓸데이터 타입

값의 종류

▼ 쓸쓸쓸자바스크립트 8개의 데이터 타입

데이터 타입별로 확보해야 할 메모리 공간의 크기도 다르고 메모리에 저장되는 2진수도 다르며, 읽어들여 해석하는 방식도 다르다. p59(e93)

- 원시타입
 - 。 숫자 타입
 - o BigInt 타입
 - 。 문자열 타입
 - 。 불리언 타입
 - ∘ undefined타입
 - o null 타입
 - 。 심벌타입
- 객체타입
 - 。 객체, 함수, 배열 등

자바스크립트는 객체 기반의 언어이며, 자바스크립트를 이루고 있는 거의 모든 것이 객체다. 7가지 원시 데이터 타입 이외의 값은 모두 객체 타입이다.

▼ 숫자타입

p60(e94)

- 숫자타입의 값은 배정밀도 64비트 부동소수점 형식을 따른다. 즉, 모든 수를 실수로 처리.
- 2진수, 8진수, 16진수를 표현하기 위한 데이터 타입을 제공하지 않기 때문에 이들 값을 참조하면 모두 10진수로 해석된다.

▼ 부동소수점

부동소수점(Floating Point)은 실수를 표현하고 계산하기 위해 컴퓨터에서 사용하는 수의 형식입니다. 고정소수점(Fixed Point) 과 달리, 부동소수점은 소수점의 위치를 움직일 수 있어 매우 큰 수와 매우 작은 수를 효율적으로 표현할 수 있습니다. 부동소수점의 이름도 소수점이 "떠다닌다(floating)"는 의미에서 유래했습니다.

부동소수점의 구성 요소

부동소수점 숫자는 일반적으로 다음과 같은 세 부분으로 구성됩니다:

- 1. 부호(Sign): 숫자의 양수 또는 음수를 나타냅니다.
- 2. **지수(Exponent)**: 숫자의 스케일(규모)을 나타냅니다. 이는 소수점의 위치를 결정합니다.
- 3. **가수(Significand 또는 Mantissa)**: 숫자의 유효숫자 부분을 나타냅니다.

IEEE 754 표준

가장 널리 사용되는 부동소수점 표현 방식은 IEEE 754 표준입니다. 이 표준에는 단정밀도(Single Precision)와 배정밀도 (Double Precision) 두 가지 주요 형식이 있습니다.

단정밀도 (32비트)

- 1. **부호 비트**: 1비트
- 2. **지수부**: 8비트
- 3. **가수부**: 23비트

배정밀도 (64비트)

- 1. **부호 비트**: 1비트
- 2. **지수부**: 11비트
- 3. **가수부**: 52비트

부동소수점 숫자의 표현 방법

부동소수점 숫자는 다음과 같은 수식으로 표현됩니다:

 $(-1)^{sign} \times 1.fraction \times 2^{exponent-bias}$

- sign: 부호 비트 (0이면 양수, 1이면 음수)
- fraction: 가수부
- exponent: 지수부
- bias: 지수부의 바이어스 값 (단정밀도의 경우 127, 배정밀도의 경우 1023)

예시

숫자 42를 배정밀도 64비트 부동소수점으로 표현하면 다음과 같은 과정을 거칩니다:

- 1. 42를 2진수로 변환: 42는 1010102
- 2. 정상화: 1.01010 × 2^5
- 3. 부호 비트: 0 (양수)
- 4. 지수: 5 + 1023 = 1028 (2진수로 10000000100)
- 이 값을 결합하면 64비트로 다음과 같이 표현됩니다:

▼ 💥 💢 문자열타입

p61(e95)-

- 자바스크립트의 문자열은 원시 타입이며 변경 불가능한 값이다.
- 💥 💥 탬플릿 리터럴
 - 。 런타임에 일반 문자열로 변환되어 처리된다.
 - 。 멀티라인, 공백 가능
 - 일반 문자열은 개행이 허용되지 않는다. 일반 문자열 내에서 공백, 줄바꿈 등을 표현하려면 백슬래시\로 시작하는 이스케이프 시퀀스를 사용해야 한다. p63 (e97)
 - 표현식 삽입 `\${표현식 } `
 - 표현식의 평가 결과가 문자열이 아니더라도 문자열로 타입이 강제 변환되어 삽입된다.
 - p111(e145) 표현식의 평가 결과를 문자열 타입으로 암묵적 타입 변환한다.

`1 + 1 = \${1 + 1}` // "1 + 1 = 2"

▼ 🜟정의

p66(e100)

자바스크립트의 undefined에서 말하는 정의란 변수에 값을 할당하여 변수의 실체를 명확히 하는 것을 말한다.

▼ null 타입

변수에 값이 없다는 것을 의도적으로 명시하고 싶을때는 null을 할당한다.

변수에 null을 할당하는 것은 변수가 이전에 참조하던 값을 더이상 참조하지 않겠다는 의미다. 즉, 이전에 할당 되어 있던 값에 대한 참조를 명시적으로 제거하는 것을 의미하며, 자바스크립트 엔진은 누구도 참조하지 않는 메모리 공간에 대해 가비지 콜렉션을 수행할 것이다.

함수가 유효한 값을 반환할 수 없는 경우 명시적으로 null을 반환하기도 한다.

▼ 💥심벌타입

심벌 값은 다른 값과 중복되지 않는 유일무이한 값이다.

주로 이름이 충돌할 위험이 없는 객체의 유일한 프로퍼티 키를 만들기 위해 사용한다.

심벌 이외의 원시값은 리터럴을 통해 생성하지만 심벌은 <u>Symbol함수를 호출해 생성한다.</u> 이때 생성된 심벌값은 외부에 노출되지 않는다.

▼ 쓸쓸데이터 타입의 필요성

• 값을 <mark>저장</mark>할때 확보해야 하는 <mark>메모리 공간의 크기</mark>를 결정하기 위해

- 자바스크립트 엔진은 데이터 타입에 따라 정해진 크기의 메모리 공간을 확보한다.
- ECMAScript 사양은 문자열과 숫자 타입 외의 데이터 타입의 크기를 명시적으로 규정하고 있지는 않다. 따라서 문자열과 숫자 타입을 제외하고 데이터 타입에 따라 확보되는 메모리 공간의 크기는 자바스크립트 엔진 제조사의 구현에 따라 다를 수 있다.

• 값을 <mark>참조</mark>할때 한 번에 읽어 들여야 할 <mark>메모리 공간의 크기</mark>를 결정하기 위해

。 특정 값이 저장되어 있는 메모리 공간의 선두 셀을 찾아갈 수 있도록 메모리 셀의 개수(바이트 수)를 알아야 한다.

• 메모리에서 읽어 들인 <mark>2진수를 어떻게 해석할지</mark> 결정하기 위해

데이터 타입에 따라 값의 해석이 달라질 수 있다. 메모리에 2진수, 0100 0001을 숫자로 해석하면 65지만 문자열로 해석하면 'A'다. p70(e104)

▼ 쓸쓸쓸동적 타입 언어

자바스크립트는 값을 **할당하는 시점**에 변수의 타입이 동적으로 결정되고 변수의 타입을 언제든지 자유롭게 변경할 수 있다.

자바스크립트의 변수는 선언이 아닌 할당에 의해 타입이 결정된다. (타입 추론)

그리고 재할당에 의해 변수의 타입은 언제든지 동적으로 변할 수 있다.

이러한 특징을 동적 타이핑이라 하며 자바스크립트를 정적 타입 언어와 구별하기 위해 동적 타입언어라 한다.

+정적타입 언어는 변수를 선언할 때 변수에 할당할 수 있는 데이터 타입을 선언해야 한다.

+컴파일러 또는 인터프리터는 심벌 테이블 이라고 부르는 자료 구조를 통해 식별자를 키로 바인딩 된 값의 메모리주소, 데이터 타입, 스코 프 등을 관리한다. p70(e104)

Ch.6 데이터 타입 4