



## Ch.5 표현식과 문

### ▼ ★★★값

값은 식(표현식)이 평가되어 생성된 결과를 말한다.

값은 다양한 방법으로 생성할 수 있다. (식으로 생성할수도 있고 등등) 가장 기본적인 방법은 리터럴을 사용하는 것이다.

### ▼ ★★평가

평가란 식을 해석해서 값을 생성하거나 참조하는 것을 의미한다.

### ▼ ★★★리터럴

사람이 이해할 수 있는 문자 또는 약속된 기호를 사용해 값을 생성하는 표기법을 말한다.

자바스크립트 엔진은 코드가 실행되는 시점인 런타임에 리터럴을 평가해 값을 생성한다. 즉, 리터럴을 값을 생성하기 위해 미리 약속한 표기법이라고 할 수 있다.

### ▼ ★★★표현식

표현식은 값으로 평가될 수 있는 문이다.

문법적으로 값이 위치할 수 있는 자리에는 표현식도 위치할 수 있다.

리터럴은 값으로 평가되므로 리터럴도 표현식이다.

변수 식별자를 참조하면 변수 값으로 평가된다. 식별자 참조는 값을 생성하지는 않지만 값으로 평가되므로 표현식이다.

표현식은 리터럴, 식별자, 연산자, 함수 호출 등의 조합으로 이뤄질 수도 있다.

### ▼ ★★★문

문은 프로그램을 구성하는 기본 단위이자 최소 실행 단위다.

문의 집합으로 이뤄진 것이 바로 프로그램이며,  
문을 작성하고 순서에 맞게 나열하는 것이 프로그래밍이다.

문은 여러 토큰으로 구성된다.

문은 선언문, 할당문, 조건문, 반복문 등으로 구분할 수 있다.

## ▼ 🌟 토큰

토큰이란 문법적인 의미를 가지며 문법적으로 더이상 나눌 수 없는 코드의 기본 요소를 의미한다. p54(e88)

## ▼ 🌟🌟🌟 표현식인 문 vs 표현식이 아닌 문

표현식인 문은 값으로 평가될 수 있는 문이며 표현식이 아닌 문은 값으로 평가될 수 없는 문이다.

표현식인 문과 표현식이 아닌 문을 구별하는 가장 명료한 방법은 변수에 할당해 보는 것이다. (표현식인 문은 값으로 평가되므로 변수에 할당할 수 있다. 하지만 표현식이 아닌 문은 변수에 할당하면 에러가 발생한다.)

변수 선언문은 값으로 평가될 수 없다. 따라서 표현식이 아닌 문이다.

하지만 할당문은 값으로 평가될 수 있다. (할당된 값으로 평가된다.) 따라서 표현식인 문이다. 할당문은 값처럼 변수에 할당할 수 있다.

이러한 특징을 활용해 여러 변수에 동일한 값을 연쇄 할당할 수 있다. p80(e114) 예제 07-11

```
var a,b,c;  
  
a = b = c = 0;  
  
console.log(a,b,c); // 0 0 0
```

+0개 이상의 문을 중괄호로 묶은 코드 블록 뒤에는 세미콜론을 붙이지 않는다. (if문, for문, 함수 등의 코드 블록 뒤에는 세미콜론을 붙이지 않는다.) 코드블록은 언제나 문의 종료를 의미하는 자체 종결성을 갖기 때문이다.

+문이 완료되서 반환하는 값 = 완료값 p58(e92)