

# 13장. 스코프

## 13.1 스코프란?

- 스코프란?

식별자가 유효한 범위(모든 식별자는 자신이 선언된 위치에 의해 다른 코드가 식별자 자신을 참조할 수 있는 유효 범위가 결정)

- 식별자 결정이란?

자바스크립트 엔진은 이름이 같은 두 개의 변수 중에서 어떤 변수를 참조해야 할 것인지 결정

즉, 스코프는 식별자를 검색할 때 사용하는 규칙

```
var x = 'global';

function foo() {
  var x = 'local';
  console.log(x); //local
}

foo();

console.log(x); //global
```

x는 이름이 동일한 식별자이지만 스코프가 다른 별개의 변수이다.

따라서 스코프는 충돌을 일으킬 수 있기에 프로그램 전체에서 하나밖에 사용할 수 없다.

== 식별자인 변수 이름은 중복될 수 없다.

## 13.2 스코프의 종류

전역 코드: 코드의 가장 바깥 영역(전역 스코프, 전역 변수)

-전역에 변수를 선언할 시 어디서든 참조 가능

지역 코드: 함수 몸체 내부(지역 스코프, 지역 변수)

-지역 변수는 자신의 지역 스코프와 하위 지역 스코프에서 유효

### 13.3 스코프 체인

스코프는 함수의 중첩에 의해 계층적 구조를 갖게 된다.

→ 중첩 함수의 지역 스코프는 중첩 함수를 포함하는 외부 함수의 지역 스코프와 계층적 구조

외부 함수의 지역 스코프(=중첩 함수의 상위 스코프)

- **스코프 체인이란?**

스코프가 계층적으로 연결된 것

실행 컨텍스트의 렉시컬 환경을 단방향으로 연결한 것

- **렉시컬 환경이란?**

코드가 어디서 실행되며 주변에 어떤 코드가 있는가

상위 스코프에서 유효한 변수 → 하위 스코프에서 자유롭게 참조 가능

하위 스코프에서 유효한 변수 → 상위 스코프에서 참조 불가능

⇒ 스코프 체인으로 연결된 스코프의 계층적 구조 == 부자 관계로 이뤄진 상속과 유사

### 13.4 함수 레벨 스코프

- **블록 레벨 스코프란?**

모든 코드 블록이 지역 스코프를 만드는 특성

- **함수 레벨 스코프란?**

var 키워드로 선언된 변수는 오로지 함수의 코드 블록(함수 몸체)만을 지역 스코프로 인정

## 13.5 렉시컬 스코프

- **동적 스코프란?**

함수를 어디서 호출했는지에 따라 함수의 상위 스코프를 결정하는 것

- **렉시컬 스코프 또는 정적 스코프란?**

함수를 어디서 정의했는지에 따라 함수의 상위 스코프를 결정하는 것